ESD-TR-68-115

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author)

The MITRE Corporation
Bedford, Massachusetts

2a. REPORT SECURITY CLASSIFICATION

UNCLASSIFIED

2b. GROUP

N/A

3. REPORT TITLE

THIRD CONGRESS ON INFORMATION SYSTEM SCIENCE AND TECHNOLOGY

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

N/A

5. AUTHOR(S) (First name, middle initial, last name)

Donald E. Walker (Editor)

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
April 1968	384	352

8a. CONTRACT OR GRANT NO.

AF 19(628)-5165

b. PROJECT NO.

5050

c.

d.

9a. ORIGINATOR'S REPORT NUMBER(S)

ESD-TR-68-115

9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)

N/A

10. DISTRIBUTION STATEMENT

This document has been approved for public release and sale; its distribution is unlimited.

11. SUPPLEMENTARY NOTES

N/A

12. SPONSORING MILITARY ACTIVITY

Electronic Systems Division, Air Force Systems Command, L. G. Hanscom Field, Bedford, Massachusetts

13. ABSTRACT

The papers contained in this volume were prepared for discussion at the Third Congress on Information System Science and Technology. The Congresses, established as biennial events, were formed to increase communication among scientists, engineers, and military personnel in the science and technology of information systems. The Third Congress, scheduled for the 21st and 22nd of November, 1966, was postponed in response to a presidential directive to curtail such expenditures. However, the timeliness of the papers makes it desirable to issue them in their present form. Introductory remarks by the session chairmen have been included where possible to show the context in which the papers were to have been discussed.

DD FORM 1473

INFORMATION SYSTEM SCIENCE AND TECHNOLOGY

Papers prepared
for the
THIRD CONGRESS

Edited by
DONALD E. WALKER

Sponsored by
AIR FORCE ELECTRONIC SYSTEMS DIVISION
and
THE MITRE CORPORATION

Thompson Book Company, Washington, D.C.

Academic Press, London

Printed in the United States of America.

Sole agent for entire world (except the Americas):
ACADEMIC PRESS
Berkeley Square House
Berkeley Square
London, W. 1

Library of Congress Catalog Card Number 67-23866

Printed in the United States of America

THIRD CONGRESS ON INFORMATION SYSTEM SCIENCE AND TECHNOLOGY

Congress Directors

Paul G. Galentine, Jr., Colonel, USAF — James H. Burrows

Executive Managers

Charles A. Laustrup, Colonel, USAF — Joseph Spiegel

Program Chairmen

Donald E. Walker
Ward C. Low

Session Chairmen

John H. Rhinehart
J. C. R. Licklider
Marlin G. Kroger
William C. Pratt, Brigadier General, USAF
Donald L. Drukey
Edward M. Bennett
Thomas P. Cheatham, Jr.
John B. Bestic, Major General, USAF
Norman Waks
Robert M. Fano
Gordon T. Gould, Jr., Major General, USAF
Richard M. Longmire
Ruth M. Davis
Samuel N. Alexander
Irving K. Cohen
Paul G. Galentine, Jr., Colonel, USAF

PREFACE

The papers contained in this volume were prepared for discussion at the Third Congress on Information System Science and Technology. The Congresses, established as biennial events, were formed to increase communication among scientists, engineers, and military personnel in the science and technology of information systems. The Third Congress, scheduled for the 21st and 22nd of November, 1966, was postponed in response to a presidential directive to curtail such expenditures. However, the timeliness of the papers makes it desirable to issue them in their present form. Introductory remarks by the session chairmen have been included where possible to show the context in which the papers were to have been discussed.

It has been customary to issue the papers for the Congresses in a preprint form and to devote the sessions exclusively to discussions of their content. In contrast, three of the sessions at the Third Congress were planned with the expectation that papers would emerge as products of the discussions rather than as stimuli for them. Two of these sessions are represented by papers, but the third session is identified only by summary.

CONTENTS

XIV TEXT PROCESSING SYSTEMS

XV LABORATORY SIMULATION OF TACTICAL SYSTEMS AND THE QUEST FOR CRITERIA

XVI NEW DIRECTIONS FOR AUTOMATED INFORMATION SYSTEMS

INFORMATION SYSTEMS AND OPERATIONS ANALYSIS

JOHN H. RHINEHART, *Chairman*

Session Participants:

LEWIS C. CLAPP
IRVIN V. VOLTIN
CARL HAMMER

Some brainware problems in information systems and operations analysis

by LEWIS C. CLAPP
Computer Research Corporation
Newton, Massachusetts

INTRODUCTION

The basic theme of this paper is quite simple; namely, we do not always have sufficient experience to extrapolate the characteristics of small-scale information systems to produce the effective large-scale systems which are in demand today. Our problem does not rest only with the hardware and software techniques which are needed to build these systems, but also with the brainware considerations which must come long before the hardware and the software can be intelligently integrated to produce the required system.

Lack of communication between the client, that is, the man who will use the system, and the system designer is the first aspect of the brainware problem. Often the client does not know how to state his information problem in terms that the system designer can use and vice versa. In fact the client and the system designer may be using identical words and talking about distinctly different things. While everyone associated with the development of an information system must share the burden and responsibility for communication during the initial systems analysis, the information system designer should be the person most capable of handling the problem. But as practitioners of information science, we often fail to communicate accurately even among ourselves.

We would not put a blindfolded man into a car and force him to drive down the Los Angeles Freeway at 80 miles an hour. Yet, we may be performing an analagous act as we develop large-scale management information systems for the man who is ill-prepared to use such tools wisely and with appropriate human judgment.

Consider, for example, the manager who enters his office in the morning, reads reports, examines computer displays, and then makes major decisions based on information processed by a large-scale computer information system. If this manager is typical, he will have but a vague idea of how the information was gathered, fed into the computer, and then processed to produce the reports upon which he bases his judgment. What goes on in the computer is a black mystery to this manager. Our typical manager is in the dark first, because he doesn't understand what the system really does and therefore he cannot use the results selectively and second, because the system may have inherent faults for which he cannot make proper allowances. Poor communication of problems, objectives and techniques lies at the root of the difficulty.

It is both natural and necessary for each field to develop its own peculiar technical terms and jargon to facilitate communication among workers in that field. But the jargon of our field does not excuse us from our responsibility to communicate with our customers. Indeed, it is rather remarkable to see words and concepts used in an imprecise and sometimes misleading manner in an industry whose main concern is the processing of information and the communication of data. Consider, for example, the four key words in the title of this session: Information Systems and Operations Analysis.

Each word individually is so broad and vague that it transmits very little information about the subject at hand. In fact "information" and "system" are two of the most overworked words in the English language, preceded in frequency of usage only by the pronoun "I." Even after the words are combined in pairs we are not much better off. What is an information system, or operations analysis for that matter? And considering the title as a whole: Are we talking about (1) operations analysis on information systems, (2) information systems for operation analysis, or (3) information systems and operation analysis applied to some other, as yet undefined, thing? We use words such as these as if there were universal consensus on their meaning, while privately each of us has conjured up a different definition. Meanwhile that manager, who will be the ultimate user of the "system," looks on impressed by our vocabulary but confused about what we are doing for him.

Perhaps, two sets of vocabulary are needed, one for internal communication within the field, and a second to facilitate clear and simple communication with the man who pays the bills and has to live with the product of our labors. It might even turn out that the layman's vocabulary could be used profitably among workers within the information science field.

What do we mean by information?

Several years ago a young M.I.T. student took his girl friend (who was not a science major) to hear a lecture on vacuum technology by the great physicist Enrico Fermi. As they were strolling home the young physicist asked his friend if she had learned anything from the lecture which had clearly been over her head. "Yes," she replied, "I learned two things. One, Professor Fermi cannot draw a straight line and two, the vacuum is where the air is not." As with the vacuum, it is easier to explain what information is not, rather than what it is.

Our difficulty in producing an operational definition of the word "information," may be due in part to the fact that the concept is dynamic and changing constantly. Only a short time ago information to a computer man meant numeric or printed data (e.g., words of text). Gradually our ability to handle other types of information increased. Today, for example, we are making good progress in processing many types of information including graphical data such as line drawings, photographs, and even analog signals (e.g., acoustic signals). With these new techniques for handling specific problems came the need for novel types of input/output devices to handle these other types of information. The result has been important gadgets such as display scopes and light-pens, Rand Tablets and film readers. Our progress in the area of applications has been impressive, but what of our achievements in the area of fundamental theory?

There have been several attempts made at developing a useful mathematical description of the information handling process but these have not, in general, met with much practical success. For example, the operational definition of information in Shannon's Information Theory is too restrictive to be useful in the information processing field (except for problems associated with information transfer and communication). To the best of our knowledge, no one has yet developed a completely satisfactory theory of information processing. Because there is no strong theoretical basis for the field, we must rely on intuition, experience and the application of heuristic notions each time we attempt to solve a new information processing problem. Frequently, we may conceive of several alternate solutions to the problem, but we do not have the techniques to determine which of these solutions is optimum in the general case.

Although we have labeled this field which has to do with the processing of information as "information science," we should recognize that it is not a science in the strict sense of the word as are mathematics and physics. There are no axioms or premises from which all conclusions must unavoidably and logically follow. There aren't even simple criteria to determine the success or validity of a new technique in a real information system. Each new system is a unique and special case. In planning the system, the skillful designer may select from a number of techniques which were successful in earlier systems. But unless the new application is quite similar to the old, the designer cannot be certain that the new system will function as originally conceived. He will generally rely on his ability to make small changes in the design as the system is constructed and as he observes the behavior of the working system. Usually these design changes (sometimes called field improvements) can be handled at relatively low expense at the software level, but sometimes the required changes must be performed on the hardware at great cost. Thus, "information science" is more of an art than a science.

Information system design would be less artful and more scientific if the process of extrapolating techniques from the small to the large system could be carried out with a high assurance of success. But we frequently run into the problem that techniques which work well in smaller systems where the data files are quite manageable, will not work effectively when extended to bigger systems which must cope with larger quantities of data. For example, many information retrieval techniques, which seem very promising when tested with small files containing only a few hundred items, break down when tested with larger files containing tens of thousands of items. In the information systems business, we lack one of the pillars of the scientific method; the reproducibility of results. The physicist knows that water will boil at 212°F, whether the quantity of water is one ounce, one quart or 500 gallons, but the information analyst has no assurance that a technique which works for 50 items will still work with a file of 5,000 entries. In information processing we have to be content with probabilities rather than certainties.

Therefore, the process of constructing an information system generally boils down to a cut and dry procedure. The system is first designed, implemented and then tested. During the testing phase errors and inconsistencies are found and then eliminated. Once again the system is tested and the process is repeated until either the system works to everyone's satisfaction or the designers are thoroughly exhausted. Consequently, information systems evolve as successive improvements to earlier less refined systems. Although excellent systems may result from the process, we must recognize that this type of design process is time consuming, inefficient, and costly.

The solution to this dilemma will not come about by any great innovations in hardware or software. These are only gimmicks which reflect the state of our brainware. If our concepts are sound, coherent and well thought out, the software and hardware which mirrors these concepts will be useful. But if our brainware is haphazard and confused, this will be reflected in the information systems which result.

The state of operations analysis

In many respects operations analysis is in the same predicament as information systems design. Here we shall assume that the objective of operations analysis is (1) to gather information about some operation system generally boils down to a cut and dry pro-potential problems, and (3) to analyze the probable results of proposed solutions to these problems. The operations analyst has a number of tools at his disposal among which include, techniques of operation research, mathematical programming, critical path analysis and simulation. There is no general theory for solving operations analysis problems but rather there exists a collection of techniques which have worked well for special cases in the past. Here again there are no general principles or axioms which the operations analyst can apply to any new problem area. He must, instead, skillfully adapt techniques which have apparently worked for earlier problems.

Abstruse mathematics is not in itself the answer to our difficulties. The complex systems generally encountered in the real world do not lend themselves to neat mathematical formulations and in most cases the operations analyst is forced to reduce the problem to simpler terms to make it tractable. This is often done by building a simplified model of the real world process under study. Frequently this model is simulated on a computer to handle the high data volume in a reasonable period of time. It is important to verify that the solution obtained by use of the model corresponds to an acceptable solution in the real world which is considerably more complicated than the model.

In this paper we are concerned with union of operations analysis and information systems. The data about the process under observation is gathered, processed and displayed by means of the information system. Just as the operations analyst had to make some approximations about the real world process, the data in the information system itself may be an approximation to the real world because of the lack of complete information. First, all of the information about the real world process may not be available on a timely basis. Second, the hardware portion of the system has inherent limitations, it usually cannot store all of the information that is available or process it in a reasonable period of time. Third, we may not have developed suitable software techniques for processing all of the information even if it were available and could be handled by the hardware. Thus, we are usually in the position of having only incomplete information about the real world process at our command.

Given these two imperfect or approximate models of the real world process, one the result of the limitations of information science, the other due to the inherent approximations of operations analysis, there is a possibility that the combined errors will lead to serious deficiencies in the total system. It is as if we were simultaneously wearing two pairs of eyeglasses and trying to look at the world. Unless the two pairs of glasses are carefully matched to one another, our view must be distorted. And if we rely too heavily on the total system in our planning and decision making, we deserve the fate which must inevitably follow.

Can we improve our brainware?

The problem with our brainware involves fundamental concepts and principles and there can be no simple road that we can follow to reach the desired destination. Newton did not arrive at his three laws of motion by saying "I think I shall try to understand the fundamental nature of mechanics in the universe, today." But a study of the way that fundamental notions came about in many of the sciences can at least help to guide us.

First, we must realize that it is the basic notions that we are after. We must not allow ourselves to be distracted by cute devices or clever computer programs. The programs and the devices are important and worthwhile, but they will usually not help us with tomorrow's problems. We should be willing to invest the time and money in a pursuit of basic understanding that may have little payoff in the next few years and which may be absolutely essential five years from now.

Second, we must be patient. Given a new problem we should learn to control our urge to jump to the solution. Implementing the first system that comes into our heads is definitely the most costly way to go about solving an important problem. We should first consider the simplest, perhaps almost trivial, ideas which might be a solution to the problem. By understanding why the simple solution cannot work, we are then ready to examine the next simplest idea. Gradually, we will arrive at a more complex solution which meets all the requirements and which solves the problem. And who knows, in some cases the simplest idea may even work!

Third, we must develop techniques for the objective comparison and evaluation of systems. What makes system A better than system B? How much better? Was it worth the extra cost? Answers to questions such as these are important if we hope to build a basic foundation for a science of information processing. If the operations analyst and the information system designer will join hands to develop criteria for handling these kinds of questions, this will be a major step forward to improving our brainware.

Hardware considerations for information systems and operations analysis

by IRVIN V. VOLTIN
The National Bureau of Standards
Gaithersburg, Maryland

INTRODUCTION

The development of hardware for Information Systems has been very slow. The first computers were designed as batch processors and now, approximately 15 years and four model changes later, the designers *know* what is required in the way of hardware. In the meantime, users have designed information systems requiring far more sophisticated hardware design. It is quite obvious that software is leading hardware by at least one model change. This could be the reason for emulators performing so well. There are growing indications that computer design and utilization have reached their infancy.

Throughout the years thru-put has been used as the yardstick for measuring computer performance. Perhaps a slight change to this measurement would give us a clearer picture of hardware capability and performance. Information flow would be a more appropriate term and would give us a better understanding of multi-programming, multi-processing, batch processing, conversational and inquiry capability. In order to give a more vivid image of information flow, a comparison to the flow of liquid through a pipe may best illustrate this point.

Batch processing

Because past emphasis was placed on batch processing, the hardware design provides for a better information flow. Assuming that the internal memory can be illustrated as a pipe, the cross sectional area is proportional to the memory size. The input and output can be illustrated as fluid pumps on each end of the pipe and the processor itself can be illustrated as a filter in the pipe near the output pump. In order to process a batch job, the input pump is turned on until the pipe is full or all the fluid has been pumped into the pipe. The rate at which the fluid will reach the output pump is dependent upon the input rate and the efficiency of the filter. Pipe lines are a good illustration of this type of processing because they first pump through one type of fluid and, when all of that fluid has been pumped through the pump lines, they then pump another kind of fluid through this same pipe. As can be seen, this is the most efficient way of handling very large tasks, and it can be seen that many small tasks will not be handled efficiently because of the need to clean all of the fluid out of the pipe before the next task can start.

Multi-programming

Multi-programming can be considered as consisting of a number of thin-wall pipes inserted inside the main pipe which is the total internal memory. The reason for using these pipes is that the input pump and/or the output pump does not have sufficient capacity to utilize the filter full time or fill the main pipe. This liquid can be used in each of these smaller pipes. As each smaller pipe is filled, the filter is switched to that pipe. In this way the filter is shared amongst all the pipes. The space between the pipes is lost capacity. In order to help alleviate this waste, dynamic relocation came into being. The area that is lost due to the thickness of the thin-wall pipes is executive overhead* and this varies from system to system. The pressure which must be maintained in the main pipe in order to keep the smaller thin-wall pipes from bursting is the memory protect feature or guard mode.

Conversational mode

The term "time sharing" is not used here because I am not aware of any hardware system which is not used by more than one person. Conversational mode consists of as many small pipes as possible inserted in this large main pipe. In order to converse, the filter must be readily available so that it can re-

*Executive overhead is defined as the time used by the executive or monitor to determine status of various programs, perform the necessary responses and pass control to the proper program.

spond to any input pump which is started. At the present time, the input pumps have very limited capacity. This is an area in which a great deal of work is required. Conversational mode could be considered as an extension of multi-programming. Multi-programming already allows for interrupt processing. Therefore, it appears that a different compiler is all that is needed from a multi-programming environment to conversational mode. At the present time there is much to be done in the area of designing hardware in order to reduce the thickness of the thin-wall pipes (executive overhead).

The conversational mode can be considered as an experimental pipe line in which the designer varies the input pump and observes the characteristics of the fluid in the output pump in order to determine what variations or modifications are required.

Multi-processing

Multi-processing provides more than one filter in the system. It can be used for back-up in case a filter failure occurs . . . it can be used as an augmentation to another filter in case finer filtering is required . . . it can be used in multi-programming and conversational mode where additional filters are required in order to respond to the numerous pipes. Perhaps the most widespread use will occur when a system is used in a combination of batch, multi-programming and conversational mode. The most obvious use will take place where there is a large number of independent pipes, and these pipes can share input and output pumps. This is a complete reversal of the batch-processing mode. Perhaps that is why hardware designers are scratching their heads in trying to come up with a good multi-processing design. Let us hope it will not take 15 years and four model changes to get there.

Compiler hardware

Some very good compilers have been written along with the not-so-good and just plain no-good ones. The very good ones tend to be multi-pass compilers. After these many years there is no excuse for anyone to have numerous versions of the same compiler. The "fast compile-slow execute" and the "slow compile-fast execute" gimmick is the greatest hoax ever perpetrated on the computer users. This results from a definite lack of brainware. Compilers require good input and output pumps along with a rapid filter change. The first pass requires a coarse filter to insure that the filter does not become clogged and impede a rapid, smooth flow. On each successive pass, a finer filter is used until the desired purity is achieved. The output pump is connected to the input pump during this cyclic process. This places a requirement on the hardware for a large rapid-transfer auxiliary memory . . . one that can keep the pipe filled to capacity and both pumps working near maximum capability.

Hardware changes

Each model change has seen a major change in the circuitry, logical design, and manufacturing process of the central processor. During this same period there has been only performance change in the peripheral equipment. Source data equipment has remained practically unaltered. There have been just enough changes to the central processor to keep the user excited, but when he is finally able to evaluate the capability of the hardware, he sees that it falls short of its hoped-for capability. The major change for the better has been the obsolescence of tape-oriented systems and the emphasis on drum-oriented systems with mass storage capability. Multi-programming has existed for a number of years and seems to have matured. The drum-oriented systems have proven themselves in satisfying the hardware requirements for good compilers.

Hardware expectations

The major new hardware improvements will be in source data readers and displays for man-machine interaction. Otherwise there will be a leveling off of the population of computer systems. It is hard to conceive of a great expansion in management information systems unless there is a breakthrough in source-data automation. The time is rapidly approaching when computer manufacturers can no longer rely upon their circuit designers as the major source of processor improvements. Major improvements in performance will depend much more on systems and logical designers. These design engineers will, of necessity, become more user-oriented. It should be easy for them to become at least as user-oriented as the system programmers. One look at the software generated by the computer manufacturers should prove this point. Up-grading and enhancing sections of the system without disturbing the rest of the system will be the goals. Software will have to be as modular as hardware so that as the hardware is modified, the software can be modified in a parallel operation. We can expect improvements or modifications of the hardware and software as often as every year. Computer manufacturers should be able to respond to changing user requirements annually instead of every four years as they have in the past.

SUMMARY

In summary, the statement may be made that hardware is that which makes writing software nearly impossible, and if it were not for brainware, it would be impossible. Until recently the hardware used in information systems has received more attention than the software. Everyone seems to have been enamored with the words microsecond and nanosecond. Other terms which have intrigued people are real-time and time-sharing. Yet the definition of each term is applicable to everything that is done by a computer. Computer hardware will have to pass through another generation in order to reach maturity. This is evident from the fact that software and applications have been leading hardware by about three years. This probably is one of the reasons why systems software is usually so late. The programmers are trying to provide software which is far ahead of the hardware capability. This could account for the popularity of the IBM-7090 family. The software was written for the IBM 704 and 709 and therefore the next generation of hardware properly matched the software. If this is valid, why then would we have to wait until IBSYS 13 in order to get a reasonable operating system? Much publicizing has been done about the third generation hardware. Therefore, the following description may be appropriate: Third generation hardware is that hardware which allows you to use first generation software in order to achieve second generation performance.

Software considerations for information systems and operations analysis

by CARL HAMMER
UNIVAC, Division of Sperry Rand Corporation
Washington, D. C.

INTRODUCTION

The design and development of Management Information and Control Systems (MICS) has received considerable and ever-increasing attention since the introduction of general-purpose, digital, electronic computers. The stated purpose of all such systems is to provide management with timely, accurate, and pertinent information at a reasonable cost so that better decisions can be made with shorter reaction times. Many published studies indicate that these goals are not as readily achieved as would be apparent from the large number of systems in existence. In many cases they don't even provide a solution to the simpler problem of Management Information Systems (MIS), despite the many clichés, such as "Integrated Systems" and "Total Systems Approach," with which the field abounds. In fact, most such systems provide, at best, only dated but voluminous outputs.

This state of affairs is better understood if we examine the software aspects of the problem which result from the brainware efforts expended thus far towards its solution. Historically speaking, all Management Information Systems thus far were developed to cope with *specific* problems; as a consequence, many special-purpose languages have come into being, applicable only to certain hardware configurations and to specific problem formulations. We can find at least a hundred separate installations using as many different languages, many of which were heralded as the panacea to solve all MIS or MICS problems. Nevertheless, a good case can be made for management information and control systems designed to provide considerably more than current data and selected information. Such advanced systems are not yet in operational use because the pertinent software needs are not yet fully understood. However, it is safe to predict that future EDP users will indeed be able to make intelligent use of intelligence systems, to paraphrase Norbert Wiener. With the advent of more sophisticated hardware for displays and man-machine communications, the long-range outlook for the necessary software looks bright indeed. Given the appropriate brainware support e.g., operations research, such systems will be able to provide not only factual data, but also statistical extrapolations, decision functions, and even the calculated risks associated with alternative courses of action contemplated by management.

Fundamentals of management information and control systems

This paper is concerned with the software aspects of information and control systems. In order to attain the necessary perspective, we shall briefly examine the framework within which this software exists, that complex and interwoven pattern of hardware, software, and brainware—not necessarily in that order. The interplay between these three elements cannot be ignored; it has caused a continued increase in the complexity of the hardware, information system languages, and types of applications. As a matter of fact, present systems are mostly information systems; however, some attempts have been made to develop the control aspects but these efforts have not gone much beyond the conceptual or experimental stage.

Management information and control systems are concerned with the following:

- The environment which is to be monitored and/or controlled.
- Sensors, as well as input, output, and display devices to provide interface and feedback.
- Communication subsystems which handle the flow of information, data, and control commands (if any).
- Local and central electronic computer subsystems which process and store information and produce the required outputs.

The software for management information and control systems, therefore, must handle with dispatch all of the functions implied both by the hardware and the intended application. This software is in part strictly hardware-oriented, differing from machine to machine and from system to system. For example, executive or operating systems, so necessary to efficient operations, fall into this category. There will also be software which is strictly problem-oriented; several programming languages on various levels will provide us here with good examples. Finally, there will be software which encroaches on both of these areas, such as random access file retrieval or communications-oriented software. In the subsequent sections of this paper we shall examine critically the present state-of-the-art of this software and we shall also take a look into what may become available in the future.

Information and control system software

The purpose of all information systems is the collection, storage, retrieval, and dissemination of timely information, pertinent to specific activities coming within the purview of management functions. All of these activities are best carried out under control of an executive system.

As indicated earlier, we shall separately consider the software which is essential to the operation of such systems and its application-oriented counterpart. This approach proves rather fortuitous since the former is usually developed and supplied by the hardware manufacturer while the latter originates at the users' organizations, at least conceptually.

Programming for specific applications is generally a problem-oriented task, thus, scientific users have always been much better off than the system science specialists, because of the availability of FORTRAN or ALGOL. On the other hand COBOL and its esoteric derivatives are not yet fully developed or optimized. There have come into being dozens of special problem-oriented languages for management information systems—we shall give a sampling later. The source of these languages can often be traced to special hardware features available, or to the need for meeting special operational requirements. Of course, there is also the factor of human vanity associated with the "invention" of a new language and its acceptance on a greater than infinitesimal basis.

Recently, a whole host of new problems has caused new and special language demands, with the introduction of time-sharing and its sophisticated input/output devices. In this connection, projects MAC and SKETCHPAD at MIT have probably received the greatest publicity; similar work is also going on at many other installations. The languages developed under these projects are intensely device-oriented and there appears little hope that a Universal Display and Output Language (UDOL—everybody has the right to make up acronyms!) will be agreed upon or can be developed within the foreseeable future.

Thus, programming for information systems today, and even tomorrow, will be largely a matter of using what is available, or of developing what is needed. The feeling persists that few programmers ever master any of the languages available to such an extent that they make full use of their total power. Rather, many programmers seem to make, what appears at times, very unreasonable demands for augmentation or enhancement of existing languages, thereby destroying their universal and compatible character. This point was most recently under discussion by H. Oswald (1964) for so well established a language as FORTRAN; needless to say that the picture is even more chaotic in the more general case of system science languages.

Computer operating systems

The efficient operation of large, electronic data processing systems requires extensive and powerful executive operating systems. The reason for this need is strictly economical: best use of the very expensive hardware can only be made by holding human intervention to a minimum. The early operating systems were cybernetic in character; they made great demands upon the ingenuity of the console operator and upon the dexterity and skill of other personnel handling tapes and cards. Even under the most favorable circumstances and with highly trained personnel, such systems could never exploit the full power of the machines.

By contrast, the advent of multiprogramming and multiprocessing computers has greatly changed this picture. We have learned how to organize modular hardware such that all programs are under control of an executive monitor. The design goal has been the more effective utilization of the hardware; we have also learned that such software systems were difficult to construct unless the hardware had certain features to match the needs of the monitor system. A good deal of brainware has been expended to make executive monitors as self-contained and as powerful as possible. Some of the key features that have gone into their design concern:

- *Job schedules.* A good monitor system must recognize and properly react to stated or implied priorities, ancillary device requirements, and many other elements handled by human dispatchers.
- *Equipment allocation.* The monitor, especially in the multi-processing environment, must keep track of memory modules and peripheral devices and it must schedule their use in an optimum manner.
- *Console communications.* The monitor system must provide the operator with on-line information

about the state of all jobs and about the size of the queues awaiting allocation or release of equipments. It must also allow for human intervention where desired.

- *Remote device operations.* The monitor system must service all remote demands for input, inquiry, printer or display output. Data communication between the central computer and off-site peripherals must be handled exactly as if these stations were on-site. This is especially true for time-sharing and conversational operations.

Multiprogramming systems

Efficient system operation implies the continuous running of programs; a typical software system which was developed to do exactly that is the multiprogramming system available for the UNIVAC 1107 machine.

In this system all slow-speed devices are buffered to the central processor by way of a large random access drum. The executive system is always ready to serve interrupts from any one of the 16 input-output channels and to go to work for the device(s) on those channels. If an on-line device requests attention, a symbiont program facilitates the transfer of information between that device and its assigned drum buffer area. For example, printer output from any program is simply transferred to the drum at the time of its generation. Later, the print symbiont program transfers the output from the drum to the printer, one line image at a time. This transfer rate is governed by the speed of the printer and it does not interfere with the much faster central processor which carries on with other work in the meantime. In fact, this other work may include other symbiont programs, servicing card readers or punches, other printers, or even remote devices.

Figure 1 illustrates a typical day in the life of such

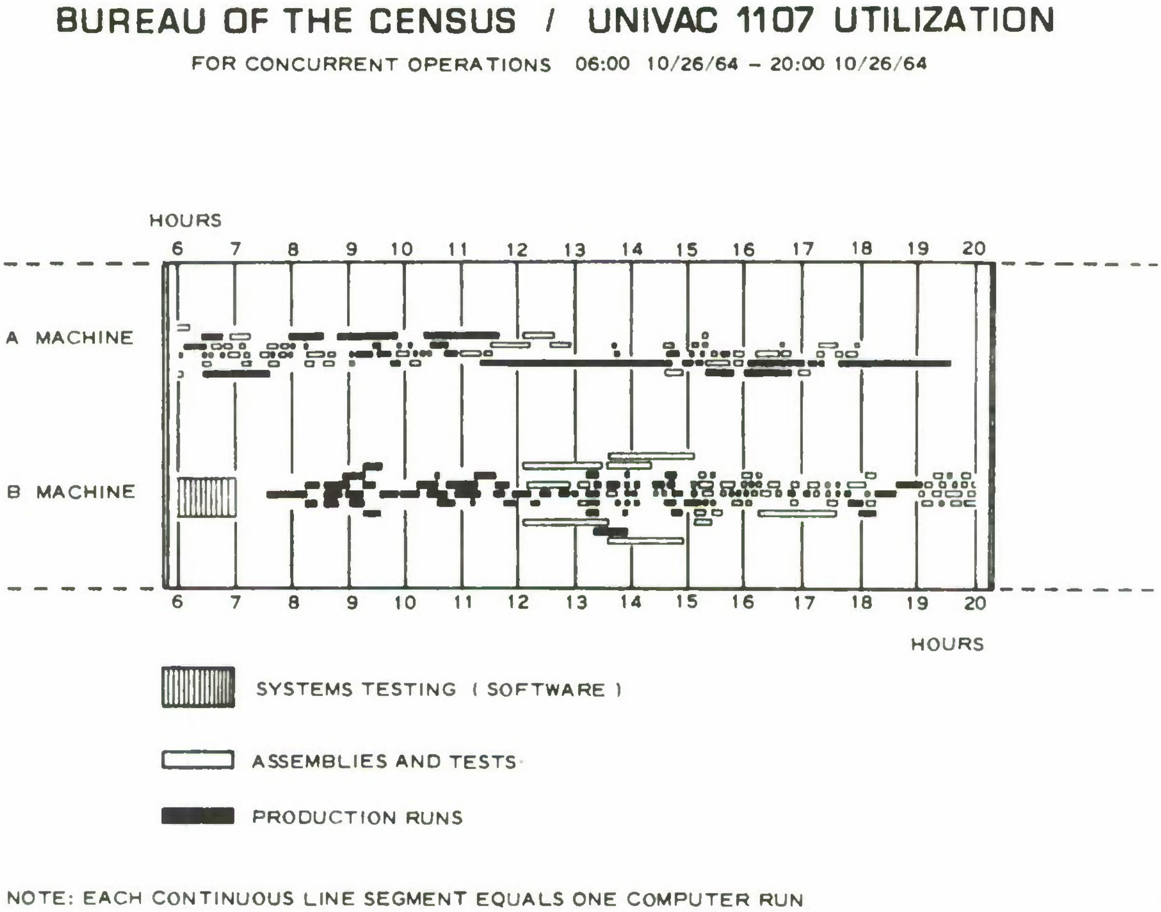


Figure 1

an operating system for two machines "A" and "B" currently in use at the U. S. Bureau of the Census. Starting at 0600 hours, production runs of various lengths were interspersed with test programs; a maximum of nine such runs appear concurrently at 1400 hours. The key to the operating efficiency of this system lies in the fact that the hardware of the machine is capable of immediate response to a large variety of internal and external interrupts. Thus it appears to the casual observer that the on-line devices seem to operate at full speed while the central processor is working at full capacity. In reality, the on-line operations are not simultaneous since the machine executes only one internal instruction at a time—hence, the terminology of concurrent operations or multiprogramming. However, all of the subject programs, including the symbionts, reside in main core memory for the duration of their activity and the executive system serves only to direct the traffic and to initiate or terminate these codes whenever required.

Multiprocessing systems

The ultimate in the true time-sharing approach is the multi-processing system. In a typical such system, the UNIVAC 1108, several large processors share a large core memory and many high-speed random access drums while control is exercised through re-entrant codes which reside in core.

Jobs are entered through the card reader, from magnetic tape, or from multiplexed remote terminals. All incoming information is first transferred to mass memory where it enters a job queue. In the case of batch jobs, the system will assign facilities when available, otherwise the job remains briefly in the waiting line. Jobs with low priority improve their relative position in the queue simply by waiting it out; thus they are eventually brought to the attention of the system and do not "fall down into the electronic crack." The system will keep many jobs in core, dynamically reallocating memory space as they terminate; several reentrant codes may be in execution by several processors at the same time. The codes for real-time jobs remain resident for their entire duration; batch processing codes are read in, executed and overlayed by other batch codes.

Time-sharing of multiple processors is not only more economical but it provides for greater resistance to partial or temporary hardware outages. For example, if one of the processors were in maintenance, the performance of the total system would only be degraded but service to the user is not interrupted. The relevant software for such systems is very complex; executive systems having a hundred thousand lines of code on drum and ten thousand lines permanently residing in core are typical of the effort required to realize all the benefits which can accrue through the use of such a system.

Application languages

The variety of languages (and their dialects) available today for Management Information Systems is illustrated by three representative examples, showing the versatility of special programming languages, semantic difficulties, or the foreboding of things to come, as in the case of computer graphics and time-sharing.

A general information processing system language

Information processing in management information systems implies the discovery and collection of facts, their organization and storage in a computer system, their selective retrieval upon demand, and a synthesis process to compose replies to queries and to present the output in an appropriate format. Such a system has been developed by the Naval Command Systems Support Activity (NAVCOSSACT) in Washington; similar systems, called ADAM and COLINGO, were developed by the MITRE Corporation.

A major part of the system generates and updates data files. Reference is permitted to specific data fields or to logical pieces of data, allowing for operations on the selected fields. The system recognizes fixed and variable length items, both repeated and non-repeated. Consecutive items may be combined into sets and sets may be aggregated into logical records. The operational capability of this file maintenance system is achieved through the use of macros which simulate a two-address variable word length computer whose memory designations include master, transaction, and summary records. There exists also a code conversion macro LOOKUP whereby encoded and decoded data may be placed into or extracted from the file.

Another important part of the system deals with information retrieval. Inquiries are made by means of statements which resemble "ordinary" English. These queries use logical operators such as AND, OR, NOT, EQ(ual), N(ot e)Q(ual), L(es)S(than), and macros for FILE, SECURITY, TITLE, SEARCH, OUTPUT, SORT, and LIBRARY.

This tape-oriented system has been operational for some time; a random access file version has also been developed. Some of the typical problems encountered in the design of this system deserve mention. Many users expected that the system would be "all things to all people" and they were very disappointed when one or the other feature did not meet their needs exactly. Certain users were able to enforce their requests for systems modifications with the result that design and development are continuing and "permanent" functions. However, the intelligent use of the system was found to depend mostly upon proper counselling of prospective users by the operating NAVCOSSACT group.

The integrated date store language

The engineering parts list problem deals with the fact that an original product is a thing made out of things which, in turn, are made out of other things. Freguently hundreds of interlocking parts lists are needed to describe a single item, such as an electronic computer. However, no matter how involved it is, the product structure must be representable in a computerized information system such that simple parts lists can be extracted from it. Random access storage is the only known methodology whereby structured information systems with more than one dimension can be developed. The key element in the design of such a system is an incidence matrix defining the bi-directional logical relation between elements in the structured lists. IDS is such a system, available with General Electric's large-scale computers. Its basic file maintains two chain structures. The first "Call-Out" chain allows penetration of the structure to the more detailed level; it contains submaterial records from which one can proceed to the next lower level. The second "Where-Used" chain defines the upward relationships for a material record and where it is used in the manufacture of a higher level item.

The software for this system is an extension of COBOL. It introduces such elements as "RETRIEVAL VIA CALC CHAIN" and "RANDOMIZE ON MATERIAL-ID." Other typical commands such as "RETRIEVE SUBMATERIAL RECORD" or "REPLACE QUANTITY-REQUIRED FIELD" perform file maintenance.

Experience with the IDS system shows that it eliminates redundant data and saves up to 40 per cent of the random access file space required by more conventional designs. In addition, operating times have been reduced up to 50 per cent because efficient buffering techniques and data blocking are applied uniformly to all IDS programs.

Computer graphics

On-line graphical control and display of computer-processed information will prove extremely effective in improving both ease and speed of man-machine communications because of its natural appeal to the human mind. The most widely publicized programs, under project SKETCHPAD, were developed at MIT. They allow the user, among other things, to draw into the scope any of three views of a solid object while the computer prepares its digital representation. The requisite graphical utility packages can be linked with other computer programs.

A list structure processing system to perform such tasks was developed at Lincoln Laboratory under the name of CORAL (**C**lass **O**riented **R**ing **A**ssociation **L**anguage). This language consists of sets of operators for building, modifying, and manipulating list structures which are developed as rings in the classical mathematical sense. Each element in the ring contains a forward pointer to the next element. One element in each ring is designated as the starting element and all other elements are subordinate to it. Subordinate elements contain a second pointer pointing backward to the starting element. A set of class operations is available with CORAL. For example, PUT statements can be used to build up rings until an element is encountered which belongs to two rings. Then a special type of connection, NUB, is introduced to tie the two rings together.

The program has received many interesting uses. For example, during flowchart programming the user can take a look at his console after calling up the flowchart compiler. Then he can construct global flowcharts on his scope by calling up an empty block, labelling it, and connecting it to another block. Computational steps and conditional tests can be entered via the keyboard, to be displayed within the designated block. In this manner it is possible to make a rough layout of the global flowchart first and then to proceed with work on the details but always with the whole context in view.

A sampling of information system languages

There does not exist today a standardized, universal Information System Language and it is unlikely that such a language will be developed in the near future, for several reasons. First, the system concept is still largely in the development stage; and second, hardware problems arise from mass storage requirements, data transmission needs, and central computer types. There have been as many approaches to information system languages as there have been problem statements. The approach in the past has been to develop systems and languages to meet specific requirements and to learn by taking small, incremental steps. The following sampling illustrates the variety of languages which are in use today:

ACSI-MATIC

This source language is used in an intelligence data processing system for the office of the Assistant Chief of Staff for Intelligence. It runs on the Sylvania 9400 computer using a large Telex disk file for random access storage of data.

AIRS

The Automatic Information Retrieval System on the IBM 7090 uses a keyboard scheme to search through technical literature.

ALERT

The Automated Linguistic Extraction and Retrieval Technique is an information handling program which specifies, collects, and retrieves large volumes of information.

BASEBALL

This program was written in IPL V language. It scans phrases, looks up words and idioms in a stored dictionary, sets up a list of attribute-value pairs, and tallies the extracted data. It was originally developed to answer sports inquiries.

HAYSTAQ

This program was written for the SEAC to assist with the search of chemical patent files according to specific topological and structural relationship.

INFRAL

This language was developed at the National Biomedical Research Foundation. It permits construction of textual materials, such as bibliographies or abstracts, from coordinate indexed materials. This language is an adaptation of COBOL with ALGOL type statements.

MOBL

This Macro Oriented Business Language is used in data processing applications in conjunction with the Macro Instruction Compiler Assembler of the SOS IBM 7090 system.

SDI

This Selective Dissemination of Information program scans large textual files and composes reports or summarizes information for management use. The original version of this program was written in FORTRAN II.

WRU

The Western Reserve University information retrieval system converts textual information to magnetic tape and searches this type against encoded questions.

VIP

This Variable Information Processing system is suitable for small, nonformalized files which can be searched in plain text language. Mnemonic codes, abbreviations, or plain text language are used in this IBM 7090 program which was developed at the Naval Ordnance Labratory, Corona, California.

SUMMARY

Modern computer technology provides an excellent basis for the design of efficient information systems. However, to exploit the full power of the hardware, a big investment in brainware and well-planned software packages is needed. The latter cannot be developed independently of the former and the present state of the software art shows clearly how much work remains yet to be done. The present emphasis is on operating systems and special languages because of the great variety of problems that have been submitted for analysis. No universal language for information systems has been developed thus far and it appears doubtful that such an effort will be made shortly. Rather, many special information system languages, each more sophisticated than their predecessors, will be developed in the years to come. Hopefully, this trend will lead eventually to the design of a very general information system language. In the meantime, users must learn to place emphasis on the exploitation of existing software and languages rather than to declaim their alleged shortcomings. A more positive and forceful approach by responsible management in this direction would be very beneficial. As a by-product, it would also tend to increase the intelligent use of our not-so-intelligent but very costly machines.

REFERENCES

1 C. W. BACHMAN *Software for random access processing.* Datamation 36-41 April 1965

2 A. B. BARNES, A. A. BRIGGS, J. GAUSS and A. RESNICK *SDI-5, An advanced system for selective dissemination of information.* Proc. ACM 19th Natl. Conf. Phila., Pa. August 1964

3 J. BELZER and W. GOFFMAN *Theoretical considerations in information retrieval systems.* Comm. ACM 7, 7 439-442 July 1964

4 J. F. BURLINGAME *Information technology and decentralization.* Harvard Bus. Review November/December 1961

5 S. G. CAMPBELL *Man/machine information transducers.* Computers and Automation September 1965

6 J. F. CAPONIO and T. L. GILLUM *Practical aspects concerning the development and use of ASTIA's Thesaurus in information retrieval.* J. Chem. Doc. 4, 1 5-8 January 1964

7 B. B. CLAYTON, E. K. DORFF, R. E. FAGEN and J. D. JOHNSON *Remote time sharing of a centralized 6600.* CDC Manual 55-0123 December 1964

8 J. DEARDEN *Can management information be automated?* Harvard Bus. Review 128-135 March/April 1964

9 J. DEARDEN *How to organize information systems.* Harvard Bus. Review 65-73 March/April 1965

10 E. W. DIJKSTRA *On the design of machine independent programming languages.* Annual Review in Automatic Programming Vol. 3 27-42 Pergamon Press. New York 360 pp. 1963

11 A. S. DOUGLAS, N. M. GRAY, K. NYGAARD and D. STARYNKEVITCH *Symposium on programming languages.* Proc. IFIP Congress 62, Munich 518-523 North Holland Co., Amsterdam, Holland 1962

12 M. K. EVANS and L. R. HAGUE *Master plan for information systems.* Harvard Bus. Review 92-104 January/February 1962

13 J. B. GOODENOUGH *A lightpen-controlled program for online data analysis.* Comm. ACM 8 130-134 1965

14 M. GREMS *A survey of languages and systems for information retrieval.* Comm. ACM 5, 1 43-46 January 1962

15 T. E. JOHNSON *Sketchpad III: A computer program for drawing in three-dimensions.* Proc. AFIPS 1963 Spring Joint Comput. Conf., Las Vegas, Nev. 347 Spartan Books, Washington, D.C. November 196 3

16 W. B. KEHL, J. F. HORTY, C. R. T. BACON and D. S. MITCHELL *An information retrieval language for legal studies.* Comm. ACM 4, 9 380-389 September 1961

17 T. E. KURTZ and K. M. LOCHNER, JR. *Supervisory systems for the Dartmouth time-sharing system.* Computers and Automation October 1965

18 J. C. R. LICKLIDER *Man-computer partnership.* Intl. Science & Technology 18-26 May 1965

19 E. J. MENKHAUS *Interloc: Control where the action is.* Business Automation pp. 47-53 July 1966

20 J. C. MILLER *Conceptual models for determining informaiton requirements.* Proc. AFIPS 1964 Spring Joint Comput. Conf., Wash., D. C. 609-620 Spartan Books, Washington, D.C. April 1964

21 *National information center.* Hearings, H. R. 1946 Ad Hoc Subcommittee on Natl. Research Data Proc. & I R Center 88th Congress, Wash., D.C. Vol. 1 (679 pp.), Appendix (508 pp.) U. S. Govt. Printing Off., Wash., D. C. 1963

22 A. G. OETTINGER *A bull's eye view of management and engineering information systems.* Proc ACM 19th Natl. Conf., Phila. Pa. August 1964

23 H. OSWALD *The various FORTRANS.* Datamation 25-29 August 1964

24 C. D. PARSONS *Sir: A statistical information retrieval system.* Proc. ACM 19th Natl. Conf., Phila., Pa. August 1964

25 A. J. PERLIS *Procedural Languages.* J. Spiegel and D. E. Walker (Eds.) Information System Sciences: Proceedings of the Second Congress. Spartan Books, Washington, D.C. 1965

26 N. J. REAM *On-line management information: part one.* Datamation 27-30 March 1964

27 N. J. REAM *On-line management information: part two.* Datamation 39-50 April 1964

28 L. G. ROBERTS *Machine perception of three-dimensional solids.* Lincoln Lab. Technical Report 315 22 May 1963

29 L. G. ROBERTS *Graphical communication and control languages.* J. Spiegel and D. E. Walker (Eds.) Information System Sciences: Proceedings of the Second Congress. Spartan Books, Washington, D.C. 1965

30 L. G. ROBERTS *A graphical service system with variable syntax.* Comm. ACM 9 173-176 1966

31 G. SALTON *A document retrieval system for man-machine interaction.* Proc. ACM 19th Natl. Conf. Phila., Pa. August 1964

32 G. SALTON *Data manipulation and programming problems in automatic information retrieval.* Comm. ACM 9 204-210 1966

33 J. G. SANTESMASES, H. L. BROWNSON, H. BORKO, F. LEVERY and H. E. PETERSEN *Symposium on advanced methods in information storage and retrievat.* Proc. IFIP Congress 62, Munich 294-298 North Holland Co., Amsterdam, Holland 1962

34 R. F. SIMMONS *Answering English questions by computer: a survey* Comm. ACM 8 53-70 1965

35 D. E. SPARKS, M. M. CHODROW and G. M. WALSH *A methodology for the analysis of information systems.* Information Dynamics Corp. Wakefield, Mass. May 1965

36 I. E. SUTHERLAND *Man-machine graphical communication system.* Proc. 1964 Spring Joint Computer Conference Spartan Books, Washington, D. C.

37 I. E. SUTHERLAND *Sketchpad: a man-machine graphical communication system.* Proc. AFIPS 1963 Spring Joint Comput. Conf., Las Vegas, Nev., Spartan Books, Washington, D.C. November 1963

38 P. H. THURSTON *Who should control information systems?* Harvard Bus. Review 135-139 November/ December 1962

MAN/COMPUTER INFORMATION INTERCHANGE TECHNIQUES

J. C. R. LICKLIDER, *Chairman*

Session Participants:

J. BRUCE FRASER
WILLIAM R. SUTHERLAND
GLEN C. CULLER

On the role of natural language in man-machine comunication

by J. BRUCE FRASER, *1st Lieutenant, USAF*
Electronic Systems Division, Air Force Systems Command
Bedford, Massachusetts

INTRODUCTION

In a conference such as this where heavy emphasis is placed on the man-machine aspect of the use of computers, the discussion very often touches on the use of a natural language such as English for communicating with a computer-based system. And almost as often the comments heard are that natural language access is too complicated, too inefficient, and not necessary. This may very well be true. But such judgments are not usually based upon examination of or experience with a natural language access capability but rather on prejudices often held over from school English courses, misconceptions born of the ill-conceived and unsuccessful attempts at mechanical translation, and a lack of understanding of how such a capability might be structured and operate. It is my purpose in the following pages to sketch out the framework for such a natural language system, to indicate how it might operate and hopefully thereby provide a better basis for further discussion of the merits of the development and subsequent implementation of a natural language access capability. The system I will discuss is hypothetical though much of the development work has already been carried out; furthermore, this organization is not the only tenable one at this stage and, of course, not necessarily the best. What I do wish to emphasize here, however, is that the role of natural language in the environment of the digital computer seems very often not placed in the proper perspective; the following is an attempt to retify this somewhat.

It is clear at the outset that the decision to even consider a natural language as an access vehicle is based on a number of not unrelated factors. The type of problem being attacked by a user is of paramount importance. Using English, one cannot expect to communicate information about second order differential equations to a machine; rather the language of this branch of mathematics will prove far more tractable. Similarly, one cannot expect to talk about the design of Mach 2 airfoils unless he uses a language specifically tailored to graphical design. Accordingly, an evaluation of the relevance of natural language should be considered in the context of those problem areas which involve, primarily, symbol manipulation with relatively little computation; for example, information about inventory, status of forces, airline schedules, and so forth.

A second very crucial consideration concerns the type of user. A frequent system user, especially one who has some familiarity with the operation of computer systems and who is familiar with the content and structure of the information stored in the data base of the system, might be expected to waive the opportunity to use a natural language and prefer a set of macro instructions which he has designed to suit his individual needs. On the other hand, the infrequent user who knows very little about the system might readily grasp at the opportunity to use his everyday language to communicate with a computer system which would otherwise be beyond his ability.

Still another consideration involves the environment in which the user operates. The air traffic controller at a large metropolitan airport cannot afford the luxury of typing into the system a question such as "What holding patterns are now available?" when he recognizes that two incoming aircraft are following the same pattern and one must alter its course immediately. However, the man responsible for ordering more typewriter ribbons for a large organization might conveniently query the system with "How many type 43 carbon ribbons do we have left?"

Finally, we can expect a natural language system to be reasonable only when the user is operating in an on-line (and thus presumably time-shared) mode. There seems to be little advantage in using English to solve a problem, get an answer, or put in some information, when the user is required to wait hours or even days for the result and his next opportunity to pursue the issue further.

In short, it seems reasonable to consider the use of natural symbols for accessing a system which manipulates symbols (facts, numbers, documents) and which involves little numerical processing (except, of course, some frequent computations precoded into subroutines which could then be called on), where the user is essentially a non-programmer and not thoroughly conversant with the system, where the response time, though important, is not critical, and where the user can communicate with the system in an on-line fashion using a typewriter and perhaps a display. (In the subsequent discussion we will not include the use of a display though it should be clear that it does not significantly alter the sense of the arguments.) To illustrate such a natural language access capability we are using a hypothetical system which has in its data base the information contained in the *Official Airline Guide,* that is, information about the schedules and accommodations on flights in the continental United States. We will assume that operation of this system will be on-line and that users will be secretaries and businessmen interested in determining available flights and (with the requisite additional capabilities) making the appropriate reservations. We will first present the organization of the system, then discuss its operations and indicate how the user might expect to ineract with it.

Organization of the system

We can represent the organization of a language access system, be it of the artificial language or natural language type, by the schema in Figure 1.

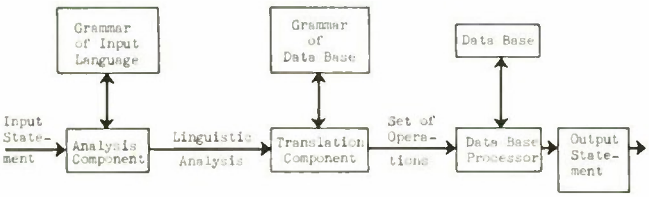


Figure 1

There are essentially two components of interest here: (1) the analysis component and (2) the translation component. The analysis component takes an input statement and analyzes it in terms of the grammar of the access language; the result is the *linguistic analysis* of the sentence. Such an analysis is a representation of the sentence specifying (among other things) syntactic information about the sentence such as what elementary units (such as words) the sentence is composed of, which of these elementary unit sequences function in this sentence as constituents in larger constructions (such as a noun being part of a noun phrase), how the constituents are combined in the sentence, and what relations they bear to one another (such as a relative clause modifying a noun phrase). Figure 2 illustrates an oversimplified syntactic analysis of the sentence "Can you fly from Boston to Chicago?"

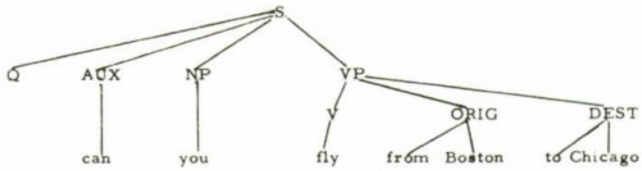


Figure 2

Ideally, the linguistic analysis of an input sentence would provide not only the syntactic analysis but also the semantic interpretation of the sentence. Thus, in the example sentence the notion of *possibility* would be associated with the word *can* since it functions as an auxiliary verb (as opposed to when it functions as a noun as in *tin can*), the notions of *to ride on an airplane* and *to glide through the air under one's own power* would be associated with the verb *fly* (as opposed to when it functions as a noun), and so forth. There is, however, no semantic theory sufficiently well developed for incorporation into a system such as we are discussing; therefore, we must be satisfied with a syntactic analysis.

The translation component has as its input the linguistic analysis of the input sentence and in terms of this derives the appropriate set of operations to be carried out on the data base. (These operations are executed by the data base processor which is of no concern to us here.) Simply stated, the translation component has the task of mapping linguistic structures onto sets of operations to be carried out on data structures. This requires that the nouns and adjectives of the input sentence be correctly associated with the appropriate rows and columns of the data base files (or in case the information designated by a noun or adjective involves the result of computation on the data structure, the appropriate computational subroutines) while the verb of an input sentence is mapped onto the appropriate operation(s) on the data structure. If the sentence contains qualifying adverbial modifiers *(e.g. tomorrow or after* 4 *p.m.)* or if a noun phrase is modified by a relative clause *(e.g. a flight which leaves Boston after* 1 *p.m.)* the translation component must recognize this and determine the appropriate operations.

To see what is involved in this, let us look first at the example data base shown in Figure 3. The information in this example file structure has been taken from a recent issue of the *Official Air-Line Guide* and structured in the following way: each city which is the destination of a scheduled flight is listed under the heading *Destination—Name;* the time zone of this city, the limousine cost from the airport to the center of the city, and a list of the names of the cities *(Origin—Name)* from which flights to this particular destination origi-

Destination File Structure

Destination											
Name	T-Zone	Lim. Cost	Origins								
			Name	T-Zone	Fares	Flights					
						D-Time	A-Time	Name	Class	Stops	A/C
Chicago	CDT	$2.00	Boston	EDT	F—60.85	8:00AM	9:15AM	AA-57	F/Y	0	B-727
					Y—50.85	4:00PM	5:14PM	TW-437	F/Y	0	B-727
						5:30PM	7:50PM	TW-101	F/Y	1	C-880
			New York	EDT	F—52-30	7:00AM	9:17AM	UA-103	F/Y	1	B-707
					Y—43.00	7:30AM	8:27AM	TW-421	F/Y	0	B-727
						10:00AM	11:12AM	AA-251	F/Y	0	C-990
						8:00PM	9:17PM	AA-297	F/Y	0	C-990
New York	EDT	$1.75	Boston	EDT	F—16.50	7:10AM	7:58AM	EA-51	F/Y	0	B-727
					Y—15.60	10:00AM	10:57AM	NE-53	F/Y	0	B-727
						5:05PM	5:50PM	EA-109	F/Y	0	B-727
						9:20PM	10:18PM	TW-15	F/Y	0	B-707

FIGURE 3

nate are also indicated. Associated with each origin city is its time zone, the fare schedule (first class and tourist) between it and this particular destination city, and the list of scheduled flights between these two cities. Associated with each flight is its departure time, arrival time, the airline name and flight number, the class of travel, the number of stops, and the type of aircraft. Those familiar with the actual *Airline Guide* will recognize that the organization of the information in Figure 3 is not very different from that in the actual publication.

This data base, like every other, consists of certain pieces of information which have been structured such that some of the relationships among the pieces are explicit, some implicit. For example, while each destination has a time zone in which it lies and is the end point of a number of flights, only the former information is explicitly stated in the example destination file structure; the latter information must be obtained by searching through the data base. The particular data base shown in Figure 3 has been organized so as to be maximally efficient for obtaining the answers to a certain class of input questions. That is, there are clearly some questions which are simple to answer in terms of this data organization and some which, though possible, are extremely difficult to answer. To illustrate this point, consider what is involved in determining the answer to the question, "Can I fly from Boston to Chicago?" To ascertain whether there is any such flight scheduled, it is only necessary to find Chicago under the heading *Destination—Name* and then look to see if Boston is listed under the heading, *Origin—Name*. If so, then the answer is yes; if not, then the answer is no. Similarly, for the question, "Can I fly from Boston to New York City nonstop?" the answer can be found by simply locating New York as the destination, Boston as the origin, and checking if there are any flights which have "0" under the heading *Stops*.

But it is simple to ask quite reasonable questions about flight information the answers to which cannot be easily determined. Even the question, "How many stops does American Airlines Flight 57 make?" cannot be answered immediately because the data are not arranged according to airlines and then by flight numbers but rather by destination, then by origin, then by flights. then by origin, then by flights. Similarly, questions like, "What is the quickest flight to take from New York to Los Angeles?" "Does TWA or American fly more often between Reno and San Francisco?" "Does Eastern fly to the same cities as United?" and "What is the fastest way to fly from New York to Tallahassee, Florida?" (there is no direct flight between these two cities) cannot be easily answered, not because the information is not available in the data base, but because the information is not organized for such questions. And there are many reasonable variations on the data base organization depending, of course, on where the emphasis is being placed. It is quite conceivable that a travel agency would organize the same informations in an entirely different way if it were continually being asked questions like, "When do I have to leave Boston on Sunday in order to make a four hour stopover in Reno, yet be in Los Angeles by midnight?"

To see what is involved in the actual interpretation of an input question, consider the sentence "Can you fly from Boston to Chicago?" which has the syntactic analysis shown in Figure 2. It is clear that the adverb of origin *from Boston* modifies the verb *fly* and not, for example, the subject noun phrase *you*. The interpretation of the question is not whether someone who is from Boston can fly to Chicago but whether it is possible for someone who happens to be located in Boston to fly between the two cities. And in general, the syntactic analysis of a sentence will represent those constituents which have a close relationship to each other (*e.g.* a relative clause modifying a noun phrase or a time adverbial qualifying the action specified by the verb) as being dominated by the same constituent. In the above sentence, the constituent verb phrase (VP) dominates both the verb and the two adverbials. Of course the fact that two constituents are dominated by the same higher constituent does not imply that they share a particular type of relationship; for example, the relationship between the subject noun phrase and the verb phrase is certainly different than that shared between the verb and the adverbial of origin. And it turns out that each occurrence of the same syntactic construction cannot be interpreted in the same way. The question "Can I return to New York from Boston after 11 p.m.?" is ambiguous; it can be understood as asking whether it is possible to arrive in New York after 11 p.m. on a flight originating in Boston, or whether it is possible to leave Boston after 11 p.m. on a flight heading for New York. For the first interpretation, the operation on the data base would be to determine for *Destination—Name* equal to New York and *Origin—Name* equal to Boston if there is an arrival time later than 11 p.m.; for the second interpretation, the locations are the same but the departure time must be later than 11 p.m.

Clearly the ways in which the time adverbial modifies the verb *return* and its meaning have caused this ambiguity and the difference of translation even though the syntactic structure, as we have analyzed it, is the same for the two cases. And as the input sentence becomes more complicated such as including relative clauses, the translation becomes more complicated as well. For a sentence such as "Can I go to Chicago on a flight which leaves Los Angeles after 11 p.m.?" the translation component must recognize that when a city name follows a verb like *leave, depart, go, take off,* the city specified is to be located in the column containing origins for a particular destination, in this case Chicago, and that when a time adverbial such as *after* 11 *p.m.* follows one of these verbs, the column containing departure times will be relevant rather than the one containing arrival information.

However, even without a well developed semantic theory and in spite of the sorts of difficulties briefly touched on above, an effective translation algorithm can certainly be designed for a given data base structure and a given grammar. The number of linguistic types (as opposed to tokens) is finite and in fact, as we shall argue below, relatively small. Each type can, if necessary, be treated individually in terms of the given data base. One interesting question is, of course, how general the translation can be made. For example, the operations related to answering the question "Does any airline fly between New York and Miami?" are almost the same as "Does TWA fly between New York and Miami?" the only difference being that in the second case the entry in the *Name* column of the Flight information is not free but must be TWA. To the extent that this sort of collapsing of operations can be extended, the translation algorithm can be made more efficient. Similarly, by treating verbs such as *arrive, land, come down,* as the same sort of verb, additional simplification of the algorithm can be affected. How compact the algorithm can be made is certainly an open question at this point and it depends to a certain extent on the class of input sentences.

But perhaps the more important question is just how fast such natural language analysis and translation can be accomplished. If the fairly superficial syntactic analysis of a sentence which is produced by a syntactic analyzer of the Kuno-Oettinger type (Kuno, 1967) provides sufficient information to a translation component, then we can expect sentences like "What flights which originate in Boston go to Los Angeles after 11 a.m.?" to be analyzed in a few seconds. If a deeper syntactican analysis is required—as would appear to be the case—then an analysis procedure of the sort developed by Petrick (1965) or The MITRE Corporation (Zwicky, et al., 1965) which handles a class of transformational grammars will be required. The above sentence taes anywhere from a few seconds to a few minutes when analyzed by these procedures. Of course the programming for these systems was done to maximize experimentation not speed, and one could certainly anticipate a considerable improvement in analysis time in a routine programmed for production work. Even so, the time involved is significant. Furthermore, no estimate on the translation time is available since no one—to my knowledge—has developed and implemented such a capability. Thus it may well be that the amount of machine time involved in processing a natural language input makes the system impractical. On the other hand, what is lost in machine time may be more than recovered in personnel efficiency.

Operation of the System

On this note, let us turn to the role of the user in such a system. Certainly if a natural language system such as this hypothetical one is to be economically feasible the user must not only find the system convenient to use but he must be able to improve his performance over that using a faster query language-type system or using some computerless approach. For it is not clear that from the user's point of view the system will appear all that appealing. There are a number of problems which a user will encounter, some of which the user of a query language system never meets. Let us now examine some of these.

First of all, it is quite clear that no class of users will want to use all of English in working in any particular problem area since not all sentence types will apply (one would not expect to ask "Can I persuade American Airlines to fly me to Los Angeles?") and certainly only a very small portion of the total vocabulary of English will be relevant. In fact, if one begins asking questions about some area such as airline schedules, he finds quickly that, except for exaggerated paraphrases, there are relatively few ways to ask any particular question (for example, "Can I fly from New York to Boston?" "Is it possible to fly from New York to Boston?" "Are there any scheduled flights from New York to Boston?" or "Does any airline fly from New York to Boston?") and that many of the sentence types used to ask for one type of information can be used (with a change of vocabulary) to ask for other information. Consequently, it seems reasonable to assume that a quite "habitable language" (Cf. Watt, 1966) can be defined which does not involve the full range of English sentence types but rather a small subset of them. But this of course brings up the question of how the user of the system knows what part of English is available to him.

The language available to a user cannot, practically, be presented to him as a list of the available English sentence types since for any language consisting of more than a few dozen sentences the user cannot easily memorize this list and will be spending much of his time finding out what he can say and how he can say it. (Furthermore, although the generalization of the user's language will be reflected in the grammar rules which describe it, it is extremely doubtful that the average user could utilize these rules to permit him to recognize the acceptable sentences of the language.) Nor can the user be presented with a list of formats into which he can plug the appropriate names, attributes, and values—the technique used in the query language command and control systems. Here, as above, this approach involves a long list of the possible formats and the possible vocabulary. Nor does an on-line language teaching capability seem feasible for a language of any complexity. It appears that what the user for a natural language access system must do is essentially lear what restrictions have been imposed on the language that he already knows (say, English).

What, then, is the sort of information which the user might be given so that he can learn the limitations on his language? First of all, he might be told the types of questions that can be asked of the system. For example, it is not possible to ask metaquestions of the system, questions such as "Can you answer a question of the form . . .?" or "How do I ask a question about . . .?" or "Do you have information involving . . .?" Furthermore, he might be told to avoid asking questions involving matters of evaluation as in "What would be a good flight to take from Boston to Chicago?" or "What is the best way to go from New York to Los Angeles?" Here the user is not asking for factual information but rather a judgment on the part of the system; but only if such a judgment capability is part of the system, will such questions be included in the access language.

The two types of questions mentioned above deal with the kind of information required from the system. In these cases the issue is *what* is asked for, not *how* the question is asked. Elliptical questions, those questions requiring circumstantial knowledge of time, place, etc., as in "Are there any flights to New York on Friday?" or "When does the next plane leave for Reno?" might pose one type of problem. Here the user would have to know that the machine would make certain assumptions about the information which is lacking and what these assumptions could be. The actual syntactic construction and the vocabulary of the sentence might pose another difficulty. The user might have to learn, for example, that he could use active questions ("Can I fly from Boston to Chicago on TWA?"), passive questions ("Is the flight at 6 p.m. from New York to Miami operated by American Airlines?"), existential questions ("Is there a flight between Los Angeles and San Francisco after 11 p.m.?"), and sentences in which an adverbial is questioned (*how much?, how many?, how far?, how long?, where?, what kind of?,* etc,) but not questions using *have* as the verb ("Does United Airlines have a flight from Washington, D.C. to Detroit") or querying maxima and minima ("What is the fastest way to get from New York to Los Angeles?"). Besides these syntactic limitations, the user will be constrained as to vocabulary. He might have to realize, for example, that the verbs *fly, travel,* and *go* are part of the vocabulary of the restricted language but that the verbs *peregrinate, trek,* and *cruise* are not. Or that *airplanes* and *plane* may be used but *aircraft* may not. Or perhaps he will have to know that when talking about a time period he must use the word *within* rather than *in,* as in *within two hours.* The point here is that no matter how extensive the vocabulary of the language is made, there will always be words lying outside of it and limitations on word usage.

Coupled with the problem of a limited English lexicon is the problem of word meaning. In query languages, each word has only one meaning. A word is treated unambiguously by the system, and the user learns once and for all how the system will interpret a particular word. Since there is only a limited vocabulary, memorizing the single meaning for each word is not a difficult task. For the natural language user, however, there will be a great tendency to use the same word in different contexts with different interpretations. For example, the verb *to be* conveys the notion of equality in the sentence "Logan is an airport," the notion of attribution in "Boston is large," of location in "Boston is in Massachusetts," of arriving in "The flight will be late," and of remaining in "The plane will be here for four hours." Similar problems might arise with other verbs such as *have* and *get* and with many nouns, adjectives and prepositions which have more than one meaning which can be applicable in asking questions about a particular set of information. If the system can recognize and handle the various meanings, then of course no problem exists; if, however, the user must be constrained to using certain words in one or two specific senses, then these constraints must be observed.

Looking at this problem of multiplicity from a slightly different point of view, there is the problem of the ambiguous sentence. We distinguish two different types of ambiguity: structural and lexical. The first is characterized by the possibility of analyzing a sentence syntactically in more than one way. Thus, the sentence "Does TWA or Pan American fly from Boston to Chicago?" has both the interpretation "Does either TWA or Pan American or both fly from Boston to Chicago?" and the interpretation "Which airline, TWA or Pan American flies from Boston to Chicago?" Similarly, the sentence, "Will the tickets be collected by the stewardess?" can be asking if the stewardess will collect the tickets or if someone will collect the tickets over there near (by) the stewardess. The second type of ambiguity, lexical ambiguity, is characterized by a string of words having a single syntactic analysis but having at least one word, which, in terms of this syntactic analysis, has more than one interpretation. The sentence "Can I fly on Eastern from Boston to Chicago after 6 p.m.?" is subject to the interpretation "Is there an Eastern flight operating between Boston and Chicago which leaves Boston after 6 p.m.," and "Is there any room on an Eastern flight operating after 6 p.m., between Boston and Chicago?" The ambiguity is due in this case to the two interpretations of the model verb, *can*. And, of course, there are sentences like "Can I fly on Eastern from Boston to Chicago or Los Angeles?" which contain both types of ambiguity. If the constraints on the language available to the user require that all sentences—at least in the sense that they are relevant to the data base—be unambiguous, then must this be reflected in the input question? Or can such ambiguity be tolerated and multiple answers provided with an indication at the way each was arrived at?

In the foregoing we have mentioned some of the problems which confront the user of a natural language access system: the types of questions he may ask, the amount of information he must provide in the question he is asking, the syntax and vocabulary he may use, and the types of ambiguities which may arise when he attempts to formulate an acceptable sentence of the language. It may or may not be possible to require the user of such a system to keep in mind all of these constraints on his language when he is attempting to use the system. But even if such awareness about the subset of English available could be mastered by the user it would clearly be to the user's advantage, and thus permit him to be more efficient, if he could be relieved of as many of these considerations as possible. What we are suggesting here is that the user should not be ignorant of the limitations on the access language but that he would be more efficient if he could expect assistance from the system itself whenever he makes a mistake.

How then could the system assist the user to formulate a well-formed input question? (Recall that this entire system would be operating in an on-line, time-shared mode and that there would be ample opportunity for convenient and frequent interaction between the man and the system he is using.) Probably the simplest error is the misspelling of a word or the use of a word not included in the lexicon of the language; in either case when the analysis routine begins scanning the sentence in order to associate the possible grammatical categories with the actual English words, no entry would be found for this particular word. The user could then be immediately notified that such and such a word is not currently in the dictionary, asked to check the spelling, and if he finds it to be correct, to indicate synonyms of this word. Thus, if the user had included *aircraft* in his input question and was given this response, he could, upon checking the spelling and deciding it was correct, indicate that *aircraft* = *plane* or that *aircraft* = *airplane*. Assuming that one of these synonyms were in the dictionary, the analysis procedure would then proceed further. At this same point in the analysis routine, if the user had used a word such as *have* or *maximum*, or *fastest*—words which might be in the dictionary but which are not to be used because they involve types of syntactic constructions which are not handled by the grammar—the system could make a comment to the user, saying in effect that the use of this paritcular word is prohibited for such and such a reason and that he should rephrase the question. The user could, at this point, indicate to the system, that he prefers to use certain words in a specific way and that the system should realize that when he puts in a question. Thus each user could establish his own dictionary and use the words in it as he has indicated. In fact, entire idioms and abbreviations could be defined by a given user and the system would interpret them appropriately for that user.

As the analysis routine proceeds to determine possible syntactic structures for this particular sentence, it might turn out that some particular strings of words remain unanalyzable after all possible analyses have been attempted. That is, the vocabulary was appropriate but the user had formulated a sentence with a syntactic construction not handled by the rules of the grammar or this particular string of words did not represent a grammatical construction of English at all. In either case, the user could be notified that the string in question is unanalyzable and be requested to use another construction in stating the same question, that is, provide a semantic paraphrase of the original question. Notice that the same procedure which rejects excluded syntactic constructions would reject a sentence in case it contained a word which, though in the dictionary, could not be used in that particular construction. For example, if the verb *be* is permitted with noun phrases and adjectives but not preceding locative adverbials, then the response to the question "Is Logan Airport in Boston?" could indicate that *be* cannot precede a locative adverbial. Here again, the assistance to the user does not involve any great amount of additional effort on the part of the system since it must do all of this work anyway in order to eventually arrive at the linguistic analysis (analyses) of the sentence.

Metaquestions (described before) would be rejected because their syntactic constructions are not handled by the rules of grammar. However, no distinction would be made by the grammar between this type of question and one which has as its goal the answer to a question from the system, which question, if couched in another syntactic pattern could be answered. Thus, metaquestions would be treated by the analysis routine as syntactically ill-formed. However, because the syntax of metaquestions is so limited ["Do (can) you answer questions of the form . . .?" "How do I ask a question about . . . ?" "Do you know (have information [data]) about . . . ?"] when the analysis routine determines that it is unable to syntactically analyze the input sentence the string might be examined in terms of the metaquestion syntax and the user informed that such a question type has been detected and is inappropriate.

If the analysis routine has finished analyzing the input string and has determined that there exists a structural ambiguity, the user could be notified that such an ambiguity exists and then be provided with some or all of the analyses of the sentence, so that he could indicate which interpretation he intended. If a lexical ambiguity were detected, the system could simply indicate to the user that some particular word was being used with the following possible definitions and request that the user indicate which one he intended. It should be clear that only after the linguistic analyses of the sentence have been determined is it reasonable to query the user about lexical ambiguity. Certainly there will be many words in the input sentence which have more than one meaning but many of the irrelevant meanings will be discarded during the analysis procedure. Thus, although the preposition *in* can mean *within* both spatially and temporally, as well as *at* in the sense of location, there would be no point in stopping the analysis routine and asking the user which of the three interpretations of *in* he intends in a sentence like "When does flight TWA 32 land in Boston?" since the analysis routine would automatically determine that only the locative interpretation were possible here. The question of elliptical sentences is somewhat more difficult to resolve since in many cases there are no linguistic indications that some pertinent information is lacking. For example, the sentence "When does the next flight leave for New York?" is perfectly well formed syntactically and semantically though to provide an answer requires one to know the date and location of the speaker as well as flight information. Problems of ellipsis can only be resolved in terms of the information contained in the data base of the system, its structure, and how the linguistic analysis of the input sentence is mapped onto operations on the data base. However, when it is known what type of information is missing, for example the location of the speaker, then some standard assumption could always be made and conveyed to the user. If he dislikes the assumption as presented to him he could then be free to correct the machine and it could then proceed to determine the answer to the user's question.

What, then, is the role that natural language can play as an access language for computer-based systems? We have suggested that one of the most reasonable environments to place such a capability would be where the infrequent system user interacts with the system in an on-line fashion as he requests pieces of information from the data base (we have ignored the question of how this information originally was entered or how the data base is updated). But the processing time for the syntactic analysis of an English sentence is relatively long (at least an order of magnitude) compared to the analysis of query language systems such as those described in Barlow and Cease (1965) and Spitzer, Robinson, and Neuse (1965) and the time to translate the resulting analysis into the appropriate operations on the data base of the system cannot be even estimated at this time since little effort has been expended in this area. On the more positive side, we suggested that the availability of English to the infrequent user might sufficiently increase his efficiency over other available methods to compensate for increased machine time even though he faces a multitude of stumbling blocks like word usage, ambiguity, acceptable syntactic form of sentence, and so forth. But even if the machine can quickly detect errors on the user's part and point them out to him, it is not at all clear that a user will desire or tolerate such interaction. In short, no definite conclusions can be drawn at this time. The concept of using a natural language to communicate with a machine is certainly an appealing one and I feel that there is at least some possibility of achieving success in the narrow environment defined at the beginning of this paper.

But before any attempts are made to design and implement such a system, I think two efforts should be undertaken. First, the entire area of translation from linguistic structure to data structure should be carefully studied to determine (among other things) just how detailed a linguistic analysis is necessary, whether translation need wait to begin until the entire linguistic analysis is produced, what the relationship is between the type of information requested and the organization of the data, what it would entail to build a translation algorithm which could accommodate linguistic analyses from a class of grammars (just as Petrick's (1965) procedure can analyze any sentence of a language described by a class of transformational grammars) and, similarly, accommodate a class of data structures, and so forth. Once this has been done—or is at least well under way—a translation algorithm should be designed and programmed to determine at least the order of magnitude of time involved. This programming should be done to maximize efficiency for clearly if translation time is excessive, such a system has little chance of being economically practical.

Second, an experiment should be conducted where a group of potential system users are told they have such a system at their disposal and can use the teletype to communicate. The reaction of these users who are not actually receiving feedback from a natural language system but rather from a combination of diagnostic programs and the experimenters can then be studied as parameters such as language subsets, scope of questions, and content of data base, are altered. Certainly if such experiments show promising results, we can be guardedly optimistic. But when, and only when, both efforts show favorable results do I think we will be in a position to seriously consider design of a natural language access capability for a computer-based system.

REFERENCES

1 A. E. BARLOW and D. R. CEASE *Headquarters, United States Air Force Command and Control System Query Language* J. Spiegel and D. Walker, Eds. Information System Sciences: Proceedings of the Second Congress Spartan Books, Inc., Washington, D. C. 1965

2 J. F. SPITZER, J. G. ROBERTSON, and D. H. NEUSE *The Colingo System design philosophy* J. Spiegel and D. E. Walker (Eds.) Information System Sciences: Proceedings of the Second Congress Spartan Books, Inc., Washington, D. C. 1965

3 S. KUNO *Computer analysis of natural languages* To appear in Proceedings of Symposium on Mathematical Aspects of Computer Science New York 1967

4 S. R. PETRICK *A recognition procedure for transformational grammars* Unpublished M.I.T. Doctoral Dissertation Cambridge, Mass. 1965

5 W. C. WATT *Habitability* Technical Report Technical Operations Research, Burlington, Massachusetts May 1966

6 A. M. ZWICKY, J. FRIEDMAN, B. C. HALL and D. E. WALKER *The MITRE syntactic analysis procedure for transformational grammars* AFIPS Conference Proceedings: Fall Joint Computer Conference 27, 317-326 1965

Language structure and graphical man-machine communication

by WILLIAM R. SUTHERLAND
*Massachusetts Institute of Technology Lincoln Laboratory**
Lexington, Massachusetts

Computer graphics has become an important means of man-machine information exchange. Unlike conventional computer languages, graphical languages have received little study, and their formal properties have not been examined in depth. The lack of precise ways to formulate and represent graphical language fundamentals impedes the use of graphical techniques in many problem areas.

The term "graphical language" has been used in a computer context with at least two meanings (Ross, 1964; Teager, 1964). First, the term has been applied to a light-pen and push-button type of language used to control an interactive system such as Sketchpad (Sutherland, 1963) or DAC-I (Jacks, 1964). By manipulating the light-pen as a pencil and using buttons for commands, a user of this kind of on-line language causes a picture to appear on the computer's graphic display. The second kind of graphical language is the picture language of the output itself.

Use of the light-pen and push-button kind of control language circumvents some problems inherent in direct pictorial data input. Well-known deficiencies in current pattern recognition techniques make it difficult to use pictures as computer input data. Instead of presenting the computer with input data in the form of an already existing picture, we can provide explicit instructions for constructing the picture. The extra information available about how the picture was constructed makes the computer's task of interpreting a picture easier. For example, the computer's knowledge that two terminals are connected by a line would be derived not from the presence of the line in some drawing, but from the user's command, "CONNECT A LINE BETWEEN THESE TWO TERMINALS." Replacing the analysis of a complex situation by the synthesis of the result from simple components is a well-known and useful technique.

Statements in either a control language or a picture language can be well-formed or ill-formed. A typed control statement like "DRAW FROM (X1, Y1) TO (X2, DELETE)" is incorrect because the operation DELETE appears in place of a numerical parameter. A picture of a flow chart might be legal only if a continuous flow path exists; a missing flow connection between two boxes would make the flow chart ill-formed. Rules for distinguishing between legal and illegal constructions are commonly called the syntax rules of a language.

At present, we know considerably more about the syntax of control languages than we do about the syntax of pictures. A control language is similar to a written language in that it consists of a linear sequence (in time) of inputs. The properties of linear languages are well understood and formalizing their syntactic rules is not difficult (Floyd, 1964). A picture language is two-dimensional, and as yet we have no general method of formalizing its syntax. A number of investigators are working on the problem,* but to date usable results are not available.

Since the linear syntax form of a control language is well understood, there is no theoretical difficulty in creating programs that will accept control inputs and recognize their form. Several approaches to this task have been reported (Lang, 1965; Roberts, 1966). The basic structure of a graphics system is similar to that of a compiler. Both accept a linear language input, recognize a construct, and take appropriate actions; one case creates machine code and the other manipulates a picture. Once the features of a control language are fixed, creating an input recognizer is a well specified task. The system designer is faced with choices as to which operations to include in the control language, and how to structure operators and their parameters. When these decisions have been made, known formalisms are adequate for describing the control language syntax and for creating the input recognizer.

*Operated with support from the Advanced Research Projects Agency.

*A partial list includes references 4, 5, 7, 8, 9, and 14.

On the other hand, the syntactic properties of picture languages pose certain difficulties to a graphics system designer. Two simple examples will serve to illustrate how different applications require different levels of complexity in picture languages. First, consider an interactive graphics system used as a simple drafting assistant. The object here is to use the control language for synthesizing a neat looking picture on the display. Picture syntax is not relevant in this case since anything we can possibly draw is legal. Second, consider a system designed for flow chart programming. In this case the user will draw a flow chart on the computer display and then expect the computer to compile a program from the picture. The syntax of flow chart pictures is relevant since the user may request that code be produced from an ill-formed flow chart. The computer program that compiles machine code from a picture must be able to recognize and reject ill-formed flow charts. For both these cases control language syntax is relevant; the control language must be well specified and contain operations suitable for drawing pictures.

To determine that a picture is syntactically correct, one might use an analysis program that would check an entire picture. While such a checking program is feasible in many special cases, its construction would be assisted by a formal notation for expressing picture syntax. For conventional written programming languages, one can automatically create an input recognizer by formally describing the language to a "recognizer-creating" program (Feldman, 1964). At present, there is no comparable way of describing and constructing a recognizer or checker for pictures.

Whenever a constructive input language is used, the task of checking a picture may be combined with the construction steps used for creating the picture. The tentative results of each step may be checked for correctness before being made a permanent part of the picture. Advantages in operating this way include immediate feedback on errors, and the use of the human response time between steps for picture error checking. Perhaps the most important benefit derived is the reduction of error checking into smaller, more easily understood parts. The inclusion of this kind of step-by-step error checking in a graphics system is conceptually simple. At the time the actions in a construction operation are defined one must provide a means for determining that the pictorial result of this new step is correct.

Regardless of how picture checking is accomplished, however, one difficult part of the system designer's task still remains. For each application area, it is necessary to provide detailed criteria for accepting a picture as correct. Any rules used must be formulated carefully enough so that one can write a computer program embodying them, and for many applications this will not be trivial.

Let us assume that we can create a graphics system that can determine whether a picture is correctly or improperly constructed. In some applications this will be sufficient. However, we would often like to use the picture to represent and manipulate non-pictorial concepts. Therefore, the computer must be able to interpret the meaning we ascribe to a well-formed picture. One way the picture could be used is as data to a program; a circuit diagram drawn on a scope could serve as input data to a circuit simulation program. The program using the picture as data will determine the meaning assigned to it by the computer. Pictorial information, for example a flow chart, could also be interpreted as a program. The pictorial procedure can in turn operate on other data which need not be pictorial. Thus, we could draw a program which could make an on-line typewriter into a sophisticated desk calculator. Numerical answers would be derived from typed inputs by the drawn flow chart program. The flow chart is only a picture, however, and the meaning given to it by the computer is defined by the programs which use the flow chart for instructions.

In creating programs which use a picture in a more than pictorial fashion, a critical factor is the system designer's understanding of the conventions used to give meaning to a drawing. Computer and user must share a common understanding about the picture being displayed; this can only be accomplished by the system designer who translates a user's conventions into computer programs. People use the basic conventions for standard mechanical drawings without much thought; however, the task of creating a system to control a machine tool from a mechanical drawing is non-trivial because as a minimum it requires a thorough understanding of the conventions used for dimension lines, center line, auxiliary views, cross sections, etc. Similarly, creating a flow chart interpreter or compiler requires a careful analysis of flow chart conventions. Before systems can be developed for many application areas, a surprising amount of effort must be devoted for formulating the details of the kinds of drawings to be used.

Like natural written languages, natural picture languages are often imprecise or even inconsistent. They generally have developed without the need for a careful analysis of their properties and characteristics; people just learn and use them. We must not expect picture languages to be as simple and tractable as standard programming languages. The natural and intuitive features which make pictures difficult to formalize are precisely those which make them valuable for communication. The lack of formalisms for describing picture languages imposes a real burden on the system designer who must create the programs for working with seemingly inconsistent pictures. When at last we are able to state in precise terms the meaning of a picture and the rules for forming definitions of correct and incorrect pictures, the task of creating a graphics system for any particular application should be considerably simplified.

REFERENCES

1 J. A. FELDMAN *A formal semantics for computer oriented languages* PhD Thesis Carnegie Institute of Technology June 1964

2 R. W. FLOYD *The syntax of programming languages* A survey, IEEE Transactions EC-13, 346-353 August 1964

3 E. L. JACKS *A laboratory for the study of graphical man-machine communication* AFIPS Conference Proceedings 1964 Fall Joint Computer Conference, 26 343-350 Spartan Books, Washington, D.C. 1964

4 R. A. KIRSCH *Computer interpretation of English text and picture patterns* IEEE Transactions on Electronic Computers EC-13, 363-376 August 1964

5 L. J. KRAKAUER *Syntax and display of printed format mathematical formulas* MS Thesis MIT 1964

6 C. A. LANG, R. B. POLANSKY and D. T ROSS *Some experiments with an algorithmic graphical language* MIT Report ESL-TM-220 August 1965

7 W. A. MARTIN *Syntax and display of mathematical expressions* MIT Project MAC Memo MAC-M-257 29 uJly 1965

8 R. NARASIMHAN *Labelling schemata and syntactic descriptions of pictures* Information and Control 7, 151-179 1964

9 R. NARASIMHAN *Syntax directed interpretation of classes of pictures* Communications of the ACM 9, No. 3, 166-173 March 1966

10 L. G. ROBERTS *A graphical service system with variable syntax* Communications of the ACM 9, No. 3, 173-176 March 1966

11 D. T. ROSS and C. G. FELDMAN *Verbal and graphical language for the AED system* MIT Progress Report Project MAC Technical Report No. 4 May 1964

12 I. E. SUTHERLAND *Sketchpad: A man-machine graphical communication system* AFIPS Conference Proceedings: Spring Joint Computer Conference 23, 329-346 1963

13 H. M. TEAGER *Real time man-machine communication with graphical languages* First Congress on the Information System Sciences ESD-TDR 63-474-5 January 1964

14 W. C. WATT *Morphology of the Nevada cattlebrands and their blazons, part one* National Bureau of Standards Report No. 9050 10 February 1966

TACTICAL COMMAND AND CONTROL: FIELD SYSTEMS

MARLIN G. KROGER, *Chairman*

Session Participants:

ROGER M. LILLY, *Brigadier General, USAF*
ROBERT F. WORLEY, *Major General, USAF*
S. R. BROWN, *Rear Admiral, USN*

Introductory Remarks

MARLIN G. KROGER
North American Aviation, Incorporated
Anaheim, California

Those of you who attended the Tactical Information Systems panel session at the Second Congress on the Information System Sciences might recall that one of my approaches to being a panel moderator is to reverse the normal procedure of having the audience ask questions of the panel and, instead, have the panel ask questions of the audience.

Teaching by asking questions is known as the Socratic Method because Socrates used it very effectively. History teachers have been known to shun this method of teaching because they recall that Socrates died of an overdose of hemlock, presumably administered by some of his "pupils." This thought does not bother me too much because my reason for asking questions of the audience is to get answers, not to infuriate people by making them think. In any event, Information Sciences Congress audience members obviously like to think.

Having thus reassured ourselves, let us consider some questions that might be appropriate for discussion in the unclassified environment of this Congress. One such set of questions relates to the generation of requirements for tactical information systems:

1. How can we develop requirements for systems which provide effective capability for self-contained tactical C^3 operations but which are still consonant with the trend toward centralization of command? Can you suggest better methods than are now employed? For example, do the people who understand operational problems also have an adequate feel for technical capability? Are they able to obtain the resources necessary to find out what they need?

2. Assuming that operational people in tactical elements do isolate a problem that can be solved by the application of technological ingenuity, can they persuade the service laboratories and system design elements to work on those problems? Is the interchange of information between these groups adequate? How could it be improved?

3. What is the role of nongovernment capability, profit-seeking and nonprofit-seeking, in the development of requirements? The Air Force in its "L" systems has made considerable use of such groups. The Army in its ADSAF (Automatic Data Systems within the Army in the Field) program is following a similar approach. The Navy, on the other hand, has primarily used blue-suit capability—a notable exception being the CINCPAC system. CINCUSNAVEUR is designing its automated operations information system almost entirely with in-house capability. What are the pros and cons of this?

4. When the above problems have been resolved and a useful tactical system has been designed, can it be successfully procured in tactical quantities in a timely fashion under present ASPR procedures? Does the two-step procurement process really work and eliminate the low priced but incompetent bidder? Does the Contract Definition Phase approach fit tactical systems as well as strategic?

5. What about the interchange of information between tactical units of different services? Should one service take the lead in developing systems for all services in certain tactical information system areas? What is the impact of international agreements such as NATO standards?

Another set of questions* relates to problems with current tactical command and control systems.

*Adapted from a previously unpublished set of questions provided by H. L. Shoemaker as a part of a tactical command and control study in which this author also participated.

1. If current programs are not satisfying the need or are not progressing rapidly enough, what are the problems?

a. Inadequate technology—hardware, software, system design, reliability, maintainability, weight and size?
   Is it available?
   Is it used in current programs?
b. Inadequate funding? Service programming? DOD approval?
c. Inadequate Statement of Requirement? Doctrinal difficulties? Failure to assign task? Coordination problems?
d. Inadequate planning or direction of projects? Development competence? Programming and control? Technical direction? Responsiveness to realistic needs? Over-sophistication? User acceptance? Capability versus cost tradeoff studies? Improper technical approach?
e. Inadequate progress? Poor execution? Too "global" in character? Management or approval problems?
f. Are major problem areas being neglected? Effects of ECM? Survivable communications? Interfaces with other systems?
g. Orientation for today's operations? For future anticipated enemy situations or our own capabilities?

2. Will current interface studies and standardization efforts achieve compatibility of tactical command and control systems?

a. Is compatibility currently understood?
b. How far should standardization be carried? Commonality of equipments? Electrical interconnection? Different standards to different classes of equipment? Disposal of existing equipment? Data exchange parameters (message formats, report formats, common data elements, data bases, computer programs, file structure)?
c. What are costs of standardization? What are benefits?
d. Are costs or impact of standardization adequately considered before imposition of standards?

3. What about software evolution considerations?

a. Can the model O software be expected to last the entire life of the system?
b. If the answer to a. is "no," how are changes to be controlled? Completely by system user? How should they be justified? With tradeoff analysis for each evolutionary change provided to management and funding agencies or by arbitrarily allocated software improvement budgets?
c. How can similar tactical systems in various parts of the world be kept similar enough and compatible enough to support current force structure package concepts? Should central programming facilities be established for similar systems to provide updating and improvement through new programs? How can this be policed? Can field units be given any freedom to modify software and procedures?

The following two papers by senior military personnel provide excellent insights into current systems design approaches. Questions listed above for which you do not find answers in these papers may be considered your homework assignment, class.

The test and evaluation of large-scale information processing systems in the army

by ROGER M. LILLY, *Brigadier General, USA*
Automatic Data Field Systems Command
Fort Belvoir, Virginia

INTRODUCTION

A brief narrative description is presented of the test and evaluation environment surrounding the Automatic Data Field Systems Command located at Fort Belvoir, Virginia. The paper begins with a description of the organization and continues with a discussion of those areas which have been determined as testing-critical, such as retesting of software modifications and the problem of maintaining external interfaces with the other services as their requirements, equipments, and systems continue to evolve. Problems are posed with no attempt made to delineate solutions. The paper is offered as a means to stimulate discussion during the Panel Session.

ADSAF and the automatic data field systems command

Because of the unique mix of doctrine and hardware involved in developing and fielding ADP systems, a unique organization, the Automatic Data Field Systems Command, was established to carry out the following missions: research, development and implementation of ADP techniques and systems into tactical units within the Army in the field to assist in command and control functions. These missions are delineated in detail in the Department of the Army Implementation Plan "Automatic Data Systems within the Army in the Field (ADSAF)." As the Commanding General of this organization, which is a merger of the Command and Control Information Systems Group (CCISG) of the Combat Developments Command and the Command and Control Information Systems 1970 Project (CCIS-70) of the Army Materiel Command, I report directly and concurrently to the Commanding Generals of both the Army Materiel Command and the Combat Developments Command. In addition, that portion of my unit which is in Germany is under the operational control of the Seventh U. S. Army for the conduct of an experiment being conducted within that unit. Department of the Army Staff coordination is done directly through the ADSAF Management Office of the Assistant Chief of Staff for Force Development (ACSFOR).

The Automatic Data Field Systems Command is organized functionally on a worldwide basis. The three major systems under development are the Tactical Fire Direction System (TACFIRE), Tactical Operations System (TOS) and the Combat Service Support System (CS3).

TACFIRE is designed to automate selected current artillery functions. These include, for example, ammunition and fire unit status, fire planning, target intelligence, tactical fire control, technical fire control, artillery survey, and meteorological data. These functions have, for the most part, been field tested, and the justification—in terms of cost effectiveness—has been completed. Fielding is scheduled for the 1970-73 time frame.

TOS is designed to assist in certain functions of operations, intelligence and fire support coordination. Representative functions include friendly unit status, task organizations, road networks, tactical troop movement, barrier planning, radio frequency allocation, and engineer tactical operations. Currently, the major effort in TOS development is centered with the Seventh U. S. Army in Germany. With this development as a basis, the TOS will be subsequently implemented Army-wide by the Automatic Data Field Systems Command.

CS3 is designed to assist in selected personnel, administrative, and logistical functions. Included in these are unit readiness reporting, stock control, materiel readiness reporting, ammunition service, transportation service, personnel management, strength accounting, military pay, medical services, casualty reporting, Military Police services, graves registration, maintenance services, and logistic administration. CS_3 will be tested in the Continental United States at Fort Hood, Texas, and will be implemented Army-wide during the 1970-1972 time frame.

This, briefly is a functional description of the ADSAF systems.

The Army test organization

The Automatic Data Field Systems Command is responsible for managing the development of the ADSAF systems. The Electronics Command (ECOM) is responsible for performing hardware and basic software acceptance and design testing, while the Test and Evaluation Command (TECOM) is responsible for performing all functional, user-oriented system tests. Coordination for large-scale troop tests is normally obtained through the Continental Arms Command (CONARC), which supplies all troops for troop tests.

ADSAF employment, scope, and interfaces

Some of the units in the communications nets covered by these three systems will have only card transceivers while others will only have a mixed-format message entry device. Communication itself is by radio or wire over the established communications nets. Each installation which has a computer will be dedicated to precisely one of the three systems. Because of the differing requirements of the three systems, the types of equipment which will serve each will be different.

Though the characteristics will vary between the three kinds of equipment, each computer will be large-scale and will be surrounded by a significant number of selected peripherals, depending on the functions and on the echelon of employment. For instance, each of the three systems calls for at least two kinds of computer: a fast computer with large immediate access and bulk memory for rear echelon and large installation employment, and a somewhat slower computer with smaller memory for employment in more forward areas and smaller installations. The forward areas have a more localized mission and require greater mobility of the computer systems than do the rear areas with their large over-views in mission and infrequent relocations.

These pairs of configurations within each ADSAF system have a requirement for direct computer-to-computer communication. To a lesser extent, there will also be some direct communication between the ADSAF systems themselves. And, depending on the local situation, there will be times when direct data exchange will be made with other Army tactical data systems and with the Air Force, Navy, Marines, and sometimes NATO. Interfaces, therefore, will play a large role in the testing and in the productive life of ADSAF.

ADSAF production/maintenance philosophy

The hardware portion of this philosophy is established by existing Army procedures and the ADSAF plan makes no radical changes. In essence, the total plan is: one or more contractors will produce the hardware and the software for the system and will insure that internal and external interfaces match properly. After the hardware is in production and field issue of the hardware-software package has begun, system maintenance will be taken over by the Army. This means both hardware repair and software modifications and corrections will be Army responsibility. Certain kinds of hardware maintenance will be at organization level with standard support available from depots in the event of failures requiring skills not found in the organization. All software modification will in general be by analysts and programmers located at a higher centralized echelon in an effort to retain standardization of procedures and interfaces and reduce the requirements for specialized technical personnel. Tests for maintainability will thus be significant.

The systems will not all reach the field simultaneously, but in three steps. TACFIRE will be fielded first follow by TOS and CS3, both of which will be undergoing a field concept test and evaluation during the TACFIRE development. This means that the TACFIRE interfaces will have to be the test standard for ADSAF. It also means that simulated inputs from the other two systems which conform to these interfaces will have to be available during TACFIRE testing.

When TACFIRE is being tested, the majority of the equipment undergoing test will be new to the Army inventory. Software and system testing of information systems will be relatively new to Army test agencies.

This, then, is a brief statement of the ADSAF environment when the testing phase is entered.

Objectives of test and evaluation; approaches for attaining these objectives

The objectives of the test and evaluation of large-scale information systems (or for that matter any system) in the Army are familiar, i.e., to insure that the product obtained from development meets or exceeds all applicable Army standards for quality of performance in order that only reliable, useful, maintainable, cost-effective and safe systems reach the Army in the Field.

It is currently envisioned that each new piece of equipment and every configuration in each system will undergo the complete classical series of tests from *Engineer Design Tests,* which are to collect design data, confirm preliminary concepts and calculations, and determine compatibility of components, through *Re-*

search and Development Acceptance Tests, conducted by the developing agency to insure that the specifications of the development contract have been fulfilled, to an integrated *Engineering Test/Service Test (ET/ST)* on the entire software-hardware configuration in the field under realistic physical and procedural environmental conditions under the control of an independent test agency. This test sequence is controlled by a Coordinated Test Plan (CTP) which has as one primary objective the task of insuring that there is no duplication of tests.

In parallel with this sequence of test events, the software will be undergoing a similar, though less well-defined sequence of tests. The complete software package must be available for incorporation into the integrated ET/ST. Likewise, portions of the software package (or simulation packages which produce similar functions) must be available on a phased basis to support the hardware test sequence.

The end result of this series of tests is a so-called "type-classified" system which is ready for production and troop issue. Items are classified as one of several standard types when they have been adopted as suitable for Army use; when they are acceptable as assets to meet operational requirements; are authorized for inclusion in equipment authorization documents; and are described in published adopted item lists. The minimum standard acceptable for ADSAF systems is *Standard A* (STD A): the most advanced and satisfactory items currently available to fill operational requirements. Hopefully, the question of production-suitability should have been resolved very early in ET/ST so that systems can be produced and ready for issue shortly after type-classification is approved. Whether or not this can be done is an open question and is undergoing study.

The problem of software maintenance

Because of the complexity of these systems, the ET/ST Sequence is expected to require at least a year, with type-classification coming a month later. This is a significant amount of time, and it becomes even more significant if requirements are such that software must be type-classified. Current regulations require extensive retesting after each modification before an item can be re-type-classified through "Confirmatory" test. A Confirmatory test (Type I) is a test or investigation of a system after type classification as standard and using early production models, to insure that required modifications not previously tested are acceptable for issue.

Now, change is the by-word of information systems. The ADSAF Systems are to be designed from the outset with provisions for ease of modification and extension. They will be placed in environments which are characterized by change and will by their very nature contribute to this change. It is to be expected, therefore, that almost from their first week of employment, the ADSAF systems manager will begin to receive requests for system modification. This is no reflection on system design adequacy nor on user guidance. It is primarily a product of long production lead-times and the progressive Army approach to incorporating those procedural changes which are required to keep pace with technological development. In short, then, the ADSAF systems will impact heavily on Army procedures, and the functional software will, as a result, have to be modified to reflect the changes produced. Each time a change is made there is the possibility of a new requirement being produced which will create the need for an additional modification. This is the expected environment. Of course all modifications will be analyzed for future impact and this will be reflected in the redesign in order to reduce the number of phases through a time-consuming test cycle.

New additions will also be a consideration, but they are a rather different problem. For many of them no usage experience will be available to temper design plans. Many will require changes in several portions of the software package. An example of this is the addition of a new type of round in the ammunition inventory. Not only a new technical fire direction task, but a new weapons effects and perhaps even a new firing procedures package may be required to make the system fully responsive to the addition of this round.

The problem then simply becomes the following: how can testing lead-times be reduced to permit the incorporation of software revisions and modifications within a meaningful time after their requirement is identified and approved? This is almost equivalent to the following question: does software require type-classification? Or is there a different way of fielding software with assurance of its quality, accuracy, and safety? Does a Confirmatory (Type I) test constitute an adequate test for reissue of programs with modifications?

Evaluation criteria

Of special interest is the determination of the actual criteria which should be applied to a software package to determine its efficiency or even its operational suitability. The testing of the software portions of large-scale electronic equipment is a relatively new experience for the Army. There is little backlog of experience in this type of testing on which to draw. The TACFIRE system will be the first large-scale tactical data system to be tested by the Army, and, because of this, great pains will have to be taken when describing individual test objectives, procedures, data, evaluation criteria, and results.

Of course, one area of evaluation which is important and quite apart from testing is the evaluation of procurement prososals. Here again the Army is generally inexperienced though every day this experience broadens. Many of the areas which currently have relatively undefined value when related to "efficiency" during proposal evaluation will be easier to describe precisely, weight properly, and evaluate adequately after the Army has experience with fielding several large-scale tactical information systems.

Environmental tests

Such tests present a peculiar problem to the testers of the ADSAF systems for a number of reasons:

1. It is difficult to obtain and train truly experienced personnel to operate a brand-new system during an operational test.
2. It is difficult to write a scenario which accurately simulates the conditions of field use for such a large system.
3. Because of the scope of each of these systems within the Army in the Field, accurate test of the system in use would require the services of at least a type-Corps. The requirements in men and materiel for the extent of service or troop tests make this somewhat impractical and expensive.
4. The balance point between a scenario which inadequately simulates actual system procedural environment and a full-scale troop test is not easy to define. What confidence does one have that test results obtained from a simulated test will accurately reflect system behavior during troop employment?
5. What part does partial system performance degradation play during the testing cycle? How is it evaluated?

Complete solutions to such problems are not yet in existence within Army test agencies. Until they are, we must be overly conservative and systems will tend to be over-tested.

One area where over-testing is unlikely is the area of reliability. When one asks for a particularly large MTBF with 95 per cent confidence, either one has to test a single system for a prohibitively large number of hours or test multiple systems for a correspondingly shorter period of time. Normally these tests occur relatively early in the test cycle and hence in the production cycle, so that it is unlikely that more than one system will be available. It is equally unlikely that sufficient hours of test will be available sequentially to test the confidence requirement. What, then, is the trade-off?

There are other areas where complete testing prior to fielding is impossible. But the Army wants assurance of the safety margins of a system before it is fielded. These questions and others like them remain to be answered.

SUMMARY

In this short and essentially non-technical discussion, I have tried to convey some idea of the test and evaluation framework in which the ADSAF systems find themselves. Personnel within the Army are seeking, and finding, answers to these questions daily, but it is realized that even when the systems are fielded, there will remain unanswered questions. Our problem then becomes that of reducing any adverse impact of such unanswered questions.

The role of the user in the procurement of information systems

by Robert F. Worley, *Major General, USAF*
Tactical Air Command
Langley Air Force Base, Virginia

The political requirement for centralized decision-making at highest governmental levels, aided and abetted by the remarkable progress in electronics for data processing, has reached all branches of government, causing a tremendous amount of effort and money to be directed toward the refinement of information systems. Most information systems, whether in industry or government, are used most of the time for management's timely enlightenment so that courses of action called for by analysis of the data passed through the system, may be planned. Since information is obviously the primary commodity in any information system, the differences between various systems are determined by what is to be done with the information, where it must flow, the characteristics of the hardware, the nature of the software, and the environment, both physical and electronic, in which it is to function.

It is relatively simple to develop specifications for fixed information systems. Complexity, sophistication and information flow can be established with environment not a serious consideration. The processed data are used in a fixed or specialized pattern and flexibility is not paramount. Technical interfaces can be predetermined and communications lines do not change. In contrast, few, if any, of these static characterictics are found in a tactical air control system (TACS). The conceptual and physical characteristics of a tactical command and control system differentiate it from any other type of information system. The developer of a tactical command and control system quickly discovers that converting these characteristics into hard and software, surfaces an almost infinite number of conflicting design problems. The purpose of this paper is to discuss some of these design problems in terms of conceptual and physical characterictics of the system and, using the 407L procurement, to point out the role of the user during the translation of this design into hardware.

Conceptually, information handling in any command and control system is a closed loop involving a series of repeating steps, recycled to obtain a desired detail or level of data. Steps in this cycle may be described as observation, interpretation, integration and two-way communication of information. This cycle forms the nucleus for the functional attainment of command, communications and control within a C^3 system. *Command* assists the commander in performing intellectual tasks such as memory, interpretation of new information with respect to accumulated information, recognition of a pattern or meaning in a complex of data, and quickly projecting a course of action as required for the decision-making process. Inserted between command and control is the *communications* function. It provides the transition from one to the other. Flexible communications enable the system to impinge on the real world, to operate internally, and to connect with other similar systems. The *control* function allows the application of command, whether execution is centralized or decentralized. Control is the action interface with weapons systems. Even though the C^3 system itself owns no weapons and is thus a distinct entity from other tactical weapons systems, the information passed through it is a weapon in itself, the use of which will make or break an operation.

All three of these functions are common to any military information system, but the tactical system has one distinctive requirement uncommon to the rest: it must accommodate these functions with mobile equipment. Its design must be based on the premise that it will be required to operate anywhere in the world, under any climatic conditions, and under various political and military constraints. This calls for the epitome of mobility and flexibility; it calls also for maximum reliability. These physical characteristics seriously impact upon the degree of sophistication and complexity permissible, which in some cases determines the quantity and quality of information and the rapidity with which it can be handled. Thus, the balance between physical and conceptual characteristics is much more acute than with a fixed information system.

Furthermore, since the trend is toward a higher and higher level of control, these field or tactical command and control systems must fit into a national command and control environment and the degree of sophistication must be compatible with the conceptual and the physical characteristics of lateral and higher command and control systems, at least at points of interface. The mobile command and control systems might be thought of as forming part of a hypothetical worldwide network of command and control systems of all types—military, political, economic, or what have you—all of which are tied together to form a national control system. While it may be argued that conceptually the entire network would be the same, the physical appearance and environment of this network changes as we move from the Washington level to the jungles of Viet Nam. Thus, *conceptual* design of a tactical command and control system centers about information flow but includes such other considerations as integration into a national system, interface with lateral systems, military doctrine and tactics, cost effectiveness, etc. *Physical* design centers about the packaging of the electronics, communications, and data processing into modular configurations that are mobile, both air and surface, yet which provide needed operational capability. The fundamental problem, then, in developing a tactical command and control system is to determine the optimum point between conceptual requirements for information flow, and physical restrictions dictated by mission environments, realizing that we must often make "tradeoffs" between the two parameters.

Moreover, the physical and conceptual parameters whose balance we must seek are not static entities. For example, not all conflicts will be like Viet Nam. Not all of the physical, geographic, and political environments will be like Viet Nam. These environments, whatever they are, are not explicitly predictable in either their combination or in their impact on system operation. Conceptually, if it were possible to have a standard tactical commander and a standard tactical situation, the definition of the quantitative and qualitative spectrum of information that we need to process would be less complex. But a cursory examination of tactical mission requirements makes it crystal clear that the problem is massively intricate.

The tactical control system must be dynamic, adapting to changing needs. Conceptually and physically, it must be capable of evolving with scientific advancement and it must be more responsive than any enemy system. *Conceptual* development appears likely in the areas of symbology, information stimulus and response, information requirements, and programming of information flow. The trend in information-flow *hardware* is toward greater speed, more capacity, and size/weight reduction. These trends, hopefully, will be accompanied by increased reliability.

Thus, the goal of overall system design is a tactical command and control system composed of hardware that is lightweight, mobile, reliable, and modular; that will gather, process, communicate and display the large volume of information needed in today's tactical operations; and that is capable of worldwide operations. To meet this goal, the designer of the system must thoroughly understand the user's requirements and the user must be cognizant of existing and predictable technical capabilities.

The term "user" refers primarily to the operational echelon which actually uses the data produced by the system. At the top, the prime user is the commander of the combat force. At lower echelons, users include intermediate commanders, personnel who operate the equipment, and large segments of the commanders' staffs involved in production and use of the information processed by the systems. In very broad terms within the Air Force, the tactical command and control users are the tactical operating commands—TAC, PACAF, and USAFE.

It has been a truism that the user determines the product; that hardware procurement begins with a stated requirement from a user. And this is normally the way a procurement cycle does commence, though the user may have been prompted, at least initially, by guidance from a higher echelon or through learning of advances in the state-of-the-art in industry.

Historically we can say that whereas aircraft weapon systems have been designed primarily around machines, and man adapted to them, a command and control system should be designed around the man and the machinery adapted to him. Further, aircraft specifications can be expressed in such physical terms as weight/thrust ratios or energy maneuver curves, but such physical terms will not adequately describe the human engineering for tactical C^3 systems. The command and control system and its operator form as personal a man-machine relationship as can be imagined. If the user does not participate during the complete system development cycle or if his participation is sterile and artificial, the resulting command and control product will be simply a machine—sterile and artificial—and not a *true* command and control system. It is this man-machine relationship, coupled with the complex of conceptual and physical variables cited above, which makes the role of the user so important in the design and development of the system.

This user role—and his participation with the engineers and technicians who are planning and developing new hardware for tactical command and control—has been repeatedly illustrated in Air Force System Program 407L. The user has been an active member of every stage of this procurement, from the statement of basic requirements to testing of finished hardware. During this active participation, the user and all other members of the procurement cycle have engaged in a process of continual refinement and clarification of needs and technological feasibility. The result has been, perhaps, a new form of symbiosis between the user, the technician, and industry which may well mark the path for future development of all command and control systems and perhaps other weapon systems.

The 407L life cycle began with the user production of a specific operational requirement (SOR) for new C^3 hardware. This document, published in September 1964, showed an early systemic approach to command and control by combining several previously unrelated hardware requirements—mobile command centers, lightweight radar, mobile air base equipment, new communications complexes—into a single, definitive user requirement for improved command and control equipment. Despite the inclusiveness of the document, however, there was nothing else to indicate that this was anything other than a standard beginning for a normal procurement action.

However, in February of 1965, tactical C^3 users and technicians participated in a worldwide advanced tactical air control system study (ATACS) which would form the basis for many of the more definitive hardware parameters for 407L. The study was a user-sponsored, engineer-attended, and joint user-technical product. It represented an attempt early in the system life cycle to orient all agencies along similar paths. More important, though, was the tacit recognition that the user could no longer sit back and write operational requirements that specify technical parameters too far beyond the realm of capability, nor could he state requirements that do not take full advantage of recent scientific change. For many years, the military requirements writer was not responsible for precise statements of requirements. Today he is. The ATACS study was conceived as an effort to satisfy this demand for precision.

The study relates Air Force doctrine, operations, and organization to tactical command and control equipment and capability, even though the study was primarily conceptual, and dealt almost exclusively in terms of information flow. User inputs defined the numerous terminals of the command and control system and detailed the information flow between these terminals in the form of *what* data are needed, *who* needs them, and in *what* form. Perishability and priority of data were also considered. The question of interface between the TACS and external agencies served to point out difficulties in defining system limits. This problem led to new definitions and changes in requirements. In several areas, the study served to identify new problems which could seriousl yaffect later procurement. Most important, perhaps, was the user-technician rapport established early in the procurement cycle by a study, *not* a part of the procurement system.

Programming of new command and control equipment was expressed in the proposed system package plan (PSPP) for 407L, published in May 1965, by Air Force Systems Command, as a normal part of the procurement cycle. Closely related to the ATACS study, the PSPP discussed a *system* of many pieces of hardware, all related, but which could be procured through *different* contracts from *different* manufacturers at *different* times. Thus, phasing of these procurements was a most critical action. The PSPP was a *combined* effort of the technical agency Electronic Systems Division (ESD) of Air Force Systems Command and the user. Although it was previously normal procedure for the user to contribute most or all of the *operations section* of such documents, user-representatives participated in the preparation of *all* sections of this document—overall programming, operations, logistic, manpower, personnel, budget, etc. Thus, the user and the engineer could evaluate mutual impact on the program of such factors as delivery schedules versus training requirements and operational capability versus available manpower. The result was a schedule which realistically related operational needs, technical feasibility, and industrial capability. *Corporate preparation* of the publication rather than mere coordination after the fact insured that the technician understood operational requirements and the user understood technical feasibility and cost.

Detailed physical parameters were introduced into the cycle during the preparation of various requests for proposals (RFP) to industry for specific hardware items. Each RFP has been developed as a joint user-technical publication, and meetings with industry for discussions on these proposals have been joint user-technical meetings.

Each formal guidance meeting scheduled by ESD and hosted by the participating industry was heavily user-attended. Here, the user was able to give *negative* guidance as to specific undesirable designs or approaches in addition to *positive* guidance provided by written specifications.

All of these interactions, as with the ATACS study and the PSPP, generated changes which have been reflected by revisions to basic specifications. For example, initial requirements for a lightweight 3-D radar were not in consonance with available technology, i.e., the specified range and altitude parameters could be obtained only in a relatively heavy set. However, trade-off between mobility and performance was found feasible and desirable. In this case, the user preferred mobility over some of the performance parameters, and a revised set of specifications evolved. Similar trade-offs have been made in additional areas of 407L as mutual understanding of operational needs and technical capability has grown. Usually the question of change has revolved about such factors as timeliness of delivery or sophisticated performance versus weight or reliability. There has been no set pattern of responses. The user, however, has been an active participant in each decision and has, on several occasions, revised his original specifications to agree with new, evolved concepts or changed technology.

The degree of user-participation that I have described during some of the beginning processes of procurement has been equally true for each other step of the system life cycle. Source Selection Boards have been equally user-technical staffed. Testing and monitoring of the design of all hardware, beginning with the first article configured for inspection (FACI) and extending through Category I, II and III testing, have and will continue to have detailed participation by the potential user of the system. In general, the user has been an active participant in every step of the procurement cycle. These interactions have generated changes. The quantity of changes—call it refinement of requirements—has in itself been far greater than usual, but beyond that is the important innovation that the user has been willing and able to participate in each of the mediations incidental to the changes.

Beyond the scope of the normal procurement cycle, several other actions have also played a significant part in 407L C^3 procurement and further emphasize the specialized nature of this C^3 life cycle. One such action is the series of special meetings, sponsored by the 407L Systems Project Office. These quarterly planning conferences have been attended by all military agencies that are now, or will be involved with the new TACS equipment. Conferees include representatives from DOD, USAF, TAC, PACAF, USAFE, AFSC, AFCS, AFLC, and ATC. The full range of problems relating to 407L are discussed, including programming of all areas to insure timely accomplishment of production, supply procedures, personnel training and manning, necessary organizational or operational or any other unforeseen changes. Host for the conference is rotated among the using commands and the 407L SPO. These meetings have served to emphasize the wide scope of tactical command and control and have allowed the face-to-face interaction of the many agencies that are involved in its development, procurement, testing, and operational use.

Another such action is the change in emphasis on tactical command and control by the user, best evidenced by organizational changes within the Tactical Air Command. The number and level of staff operational personnel directly involved with tactical command and control has been dramatically increased. These operational personnel have been combined with Air Force communications and technical specialists into a *single* staff agency in order to provide an integrated and emphasized approach to tactical command and control. Additionally, a new tactical command and control unit, the 602nd Tactical Control Group, consisting of over 1,400 personnel, has been formed within TAC substantially raising our capability in this area.

What I have been describing is the tactical command and control procurement system as it really exists. There is a new awareness among *all* agencies that deal with tactical command and control, of special requirements or procedures for its development. I have emphasized that the user *must understand* the complete procurement cycle and *must participate* in the whole cycle. He *must communicate* with other people who work in the cycle. This user-participation involves user-obligation and expertise. Other participating personnel are similarly obligated. The technician must understand the needs of the user; he must insure that he includes *operational* expertise as a continuous ingredient in the amalgam of talents that contribute to the final shape of the system.

Beyond this, the dynamics of our time require continuous consideration of many different variables. The most obvious factor that impacts on the system life cycle is technological change, but other variables such as cost, mission priorities, political and social changes, and conditions of conflict combine and interact to necessitate a continuous re-evaluation of requirements and hardware. In a sense, our evolutionary C^3 system mutates even as it is being produced, so that, as much as possible, we receive operable hardware that is as current as the state-of-the-art will allow, while considering such factors as timeliness, operational capability, and cost.

Our requirements statements must be developed in consonance with today's technology to meet tomorrow's military need, and must provide for evolution into future systems. The answer to this challenge is in a system where frequent reviews of dynamic factors are accomplished and where the system is allowed to evolve *while* it is being designed and produced. The specialized conceptual and physical characteristics of a tactical information system coupled with the peculiar man-machine relationship of any true command and control device make the role of the user dynamic, unique, and absolutely essential during equipment development.

MANAGEMENT OF MILITARY INFORMATION SYSTEMS WITHIN THE FRAMEWORK OF P.L. 89-306

WILLIAM C. PRATT, *Brigadier General, USAF, Chairman*

Session Participants:

JOSEPH F. CUNNINGHAM
NORMAN J. REAM
EDMUND D. DWYER
MORRIS DANTZKER, *Colonel, USA*
JAMES M. BROGAN, *Captain, USN*
GORDON L. BOWLBY

Session Summary

The "Brooks Bill" was passed by the Senate during the final days of the first session of the 89th Congress. It had been passed by the House earlier in the same session. The Bill was signed by the President during the first week of November 1965 and became law (P. L. 89-306).

This legislation provided specific authorities to: the General Services Administration for the procurement, utilization, and disposition of automatic data processing equipment, including administration of an ADP revolving fund; the Department of Commerce (National Bureau of Standards) for the development of data processing standards, the conduct of research in computer sciences, and provision of technical assistance to Federal agencies; the Federal agencies to retain their responsibility for the determination of their individual automatic data processing requirements, including the development of specifications for and the selection of the types and configurations of equipment needed, and the use to be made of the automatic data processing equipment and components; and the Bureau of the Budget for exercising policy and fiscal control over the management of the Federal ADP program.

The management programs of the BOB, GSA, and NBS are being expanded and intensified to provide a greater measure of central policy direction, coordination, and guidance to the Federal agencies in the development of computer-based systems and the acquisition and use of ADP equipment, and to provide a more concentrated effort in dealing with Government-wide problems that involve external relationships with the computer industry, American Standards Association, and others.

The purpose of the session is to review the status of the above described BOB, GSA and NBS management programs including future expansions now foreseen, together with their impacts on the management of military ADP systems.

COMMAND SYSTEM SIMULATION AND DESIGN

DONALD L. DRUKEY, *Chairman*

Session Participants:

THOMAS S. MCFEE
PHILIP R. VANCE
ANDREW E. WESSEL

Introductory Remarks

DONALD L. DRUKEY
System Development Corporation
Santa Monica, California

By way of introducing our four papers, I would like to give some comments that I had planned to make as introductory remarks had the Third Congress been held.

First, I believe that our authors share, perhaps more than I had hoped at the time we established our group of experts, in our opinions as to how much can be done and where. I think that we are all indebted to Tom McFee for the phrase, "Command Support Systems," to describe what we are really planning to discuss. Let me outline my personal position as to what I believe we can agree on with respect to that portion of a Command Support System which can be designed and implemented outside the framework of an individual command. These functions are typically tools which are used in implementing the actual operational programs which are part of the Command Support System. Many of them do not interface directly with using personnel. Where they do, they can usually be designed so that the specific language used in the interaction can be (and should be) tailored to the needs of the specific using personnel. These capabilities provide the following functions:

1. Monitoring the operation programs which actually carry out the desired tasks (the executive program).
2. Storing and retrieving data. This capability will provide for determining what information is to be stored, from what sources it arises for actually storing the data, and for retrieving data. Two elements are important in describing the data and their sources. The first is a capability for adding or deleting elements of information from the files and the second is providing technical or administrative capabilities to delimit those individuals who may change the file structure. When data are to be inserted into the files, they may arise from processing of reports or telecommunications entering the headquarters or as the result of operations carried out within the headquarters. In either case, the system must provide the capability to accept data from an appropriate source. It must provide a capability to change the source or conditions under which information may be accepted and it must limit, by administrative or technical means, those individuals who can so change the system. The system must provide a powerful retrieval capability to retrieve information from these files. To accomplish this, a capability to actually carry out the retrieval is required. This must be supplemented by a control mechanism to limit which information may be retrieved by what individuals and again there must be a mechanism for identifying who may change the system.
3. Processing the information which has been retrieved from the files in moderately complex manner. This capability includes both the processing tools and a set of delineators which determine who may add processing capabilities

to the system or may modify the existing capabilities.

4. Tools (compilers, debugging, editing, etc.) for producing new programs to be added to the existing complement, together with the necessary controls to prevent unauthorized changes in the system's program structure.

It seems to me that within this framework a command system can be implemented respective to the user requirements and which meets Mr. McFee's constraints with respect to the design of the command and control system. I believe that the specifics interacting with programs which I have just discussed will typically be hidden from most of the users in the real system, such that the requirements to tailor the language used for performing these functions to the command is not really necessary, which I, therefore, believe also conforms to Mr. Wessell's objections to laboratory-designed subsystems. Since Mr. Vance appears to be more convinced than I of the feasibility of carrying out laboratory designs, I feel that he also concurs. The element around which we had planned to center our discussion was, therefore, precisely who and how and when do the parameters, which are left open by the system components I have just described, become established.

The other question is the extent to which having built the system in the laboratory facilitates the development process in the field. I believe that it at least halves the time required to get a system operating in response to the requirements of the command.

The final question which I had planned to raise, if there were time, is a discussion as to whether the capabilities that I have just described might not facilitate the problems of intersystem communication and, particularly, of the fact that the over-all, worldwide command system is not implemented as a monolith but rather grows by evolution of its independent parts. I submit that the capabilities required to provide the needed flexibility within the command also go a considerable way towards solving those same problems for the inter-command communications problem.

Laboratory design of command and control systems

by Donald L. Drukey
System Development Corporation
Santa Monica, California

INTRODUCTION

The question I would like to address is the extent to which it is feasible to design a command and control system for a military headquarters at the facilities of the developing organization. My opinion is that, because of variations between military headquarters and the close user involvement required in the design of many system functions, it is not possible to develop an entire system in the laboratory. However—and this is the major point of my thesis—there are certain common elements of such systems which are known in advance and which will provide a set of general-purpose tools that can much shorten the process of implementing a command and control system at the user's facility.

For this discussion, the term "command and control" should be qualified to that of "command." By "command" I mean those functions of a headquarters that are concerned with planning and carrying out military operations a day or more in advance, monitoring the operations as they unfold, and making changes in the plan to meet the exigencies of the situation. I specifically exclude the control, or "real time," function which is addressed to producing direct controlling instructions to weapons and defensive systems. A great deal of the confusion with respect to command and control may be due to an unwillingness to admit that these tend to be separate functions. The control problem is, of course, far from trivial. However, for present purposes, I am addressing the problems of a headquarters which either has a small contingent responsible for such real-time control or which relegates this area to subordinate headquarters.

The problem

The functions that I want to discuss are those of providing support to the commander and his staff in their day-to-day and longer-term planning. The reaction time for these operations is typically measured in hours or even longer. It is primarily to personnel operating on such a time cycle that command systems automation can provide real support. These personnel are concerned with data problems—in particular, with problems that arise when one has large volumes of information that are or may be pertinent to the decisions to be made. Often the information required to assist in making these decisions is in the headquarters but is unavailable, because it is inaccessible or because it cannot readily be put together in the form that is responsive to the particular problem.

An automated command system, then, should provide the capability for receiving inputs to such a file of information, for storing it away, for updating and validating it as appropriate, for retrieving information from the file, and for performing the required operations on that information to make it addressive to the problem at hand. This last requirement is crucial to an adequate solution to the command problem. *Simply providing large files of information, even with a relatively prompt but inflexible retrieval means, can result only in flooding the decision makers with large amounts of largely irrelevant data.* We need the capability to retrieve flexibly and to process the data into a form easily assimilated by the using personnel.

Characteristics of command headquarters

Let us inquire whether there are underlying attributes that are common to many headquarters and that are invariant as the personnel within a headquarters change or as the mission of the headquarters changes. I assert that there are such constant factors and that by addressing ourselves to them we can produce a system which, prior to installation in the headquarters, has solved many of the problems that have proven very time consuming and difficult in the past when done in the operational context.

As we compare various headquarters, certain differences are immediately apparent. Almost no two headquarters have files with the same structure, although some elements of the files may be widely shared between headquarters. The amounts of information stored and received in a headquarters are highly variable. The processing of data to make them meaningful to the using personnel is highly dependent on the mission of the headquarters and on the personalities and capabilities of the headquarters personnel.

On the other hand, certain characteristics are constant throughout the community:

- large quantities of information must be filed;
- the precise nature of the retrievals can only partially be specified in advance;
- over time, different elements assume different importance in the files, and
- data selection and formatting must be responsive to changes in personnel and in mission.

Considering the problem historically, and focusing on the large L-systems, we find that they consist of three elements in addition to the personnel: a complement of hardware for the communications, computing, and display functions; an aggregate of computer programs to control the operations of that hardware; and the procedures used by operational personnel to direct the hardware and computer programs. It is largely through these procedures that the user can exercise whatever flexibility remains within the hardware and software environment.

More often than not, the press of system development schedules has left the developers with no choice but to provide a system that leaves little flexibility for the users or, if it is there, provides it in an extremely awkward way. We find that the hardware, particularly the computing elements, contains an inherently large but finite degree of flexibility, which is constrained by the software. When a headquarters needs to adapt to changing situations, it attempts to modify its procedures and to utilize what little flexibility remain in the hardware/software system. The personnel are limited in their ability to exploit this flexibility by the extent to which they know and understand the operations of the hardware and software system. Typically, they do not understand them well enough to make on-the-spot adjustments effectively, so they end up by making entirely manual fixes and bypassing those parts of the system which they cannot readily modify to suit their own needs.

Generalized software support

Of the complement of computer programs that operates in a command headquarters environment, a very large portion, sometimes well over half, is concerned with tools for producing and checking out computer programs—in short, for the means of modifying the software part of the system. The remainder consists of operational programs that provide for retrieval of data and for processing these data in response to the specific requirements. The general package that I would propose be developed in advance of implementation of the system at a headquarters will take care of the support functions and will provide a number of the tools which make the retrieval and processing of data easy. On the other hand, I do not believe that much can be done in the laboratory to actually tailor these tools to the operational needs of the headquarters.

These operational programs have to be jointly worked out with the actual using personnel for two reasons: first, user personnel find it difficult to formalize their requirements for a capability which they may as yet only dimly understand, and second, many of the requirements, particularly for output formats, are rather personal and are not an invariant to be settled once and for all.

What, then, can be done before a system is installed and completed at the user's site? I believe that flexible hardware can be provided with a memory large enough for an initial installation and capable of expansion as the requirements increase with time (which they invariably do). Then, I believe that we can provide basic software which the user will treat as though it were a part of the hardware, that is, something which is maintained by the supplier and which the user does not need to modify. This basic software holds the answer to the flexibility problem—by providing the capability for the user to easily implement the operational portions of his system and to modify them with little or no support from the professional programming community.

The ingredients that belong in such a software complement include:

- A time-shared executive which permits the user to carry on program development and experimentation concurrent with his operations, and to link a number of remote data sources to his computer.
- A general-purpose data base handling system which makes it easy for his operational personnel to specify which data to include in their data base, the form in which they wish to insert those data, and the modification to these descriptions as circumstances indicate. This data base handling system would also permit easy, flexible, user-oriented retrieval of information, and would provide for convenient loading and updating of the data base.
- A reporting and display system which makes it easy for operational personnel to specify formats for reporting and for display of information and provides a simple complement of easily used arithmetic capabilities for processing data retrieved from the data base prior to display or report preparation.
- A higher order language capability for producing

efficient operating programs specific to user needs.

- A set of conveniently usable tools for debugging and checking out programs both from the standpoint of the nonprogrammer, data-oriented user and from the standpoint of the professional programmer.

All these capabilities are within today's technology. We have systems in existence today in a variety of locations which provide each of these capabilities. Some of them are further developed than others. Programming languages and executive systems are in pretty good shape. The general-purpose data base handling systems and report generating and display systems, while they exist, tend to be poorly human-engineered in their present forms for an operational environment. To really satisfy the requirements, they should be simple and straightforward enough so that a reasonably intelligent user, familiar with the data that he wishes to process, can learn to operate the system in, at most, a week or two of part-time instruction. We seem to be on the verge of being able to provide that capability.

The primary limitation to our progress with such systems is not lack of technical facilities but the fact that the systems are not yet convenient enough to attract sizable numbers of operational users. As soon as we get more such users we can shake down the systems faster and turn them into really useful tools. Then we can begin implementation of a general-purpose software package having these basic capabilities. Such a package would shorten the development cycle for tailoring the command and control system to the actual requirements of a headquarters by a factor of 2 or more, and the impact on the ability of that headquarters to evolve its own system would be even more substantial.

Installation of the generalized software

I would like to discuss the mechanism by which such a capability could be phased into the operations of the headquarters. Let me postulate that the hypothetical headquarters is now operating with a system utilizing an IBM 1410 and uses that headquarters' variant of the 473L System. The system consists of a set of files, a set of the basic procedures supplied through 473L essentially unmodified, a second set of procedures that have been adapted with minor modifications for the uses of the headquarters, and, finally, a significant collection of programs that is specific to the headquarters and was developed there to meet its own needs.

The basic software package that I have described previously will provide the equivalent capabilities of the unmodified programs from 473L. Typically, these are the programs associated with inputting, storing, and updating information, and retrieving from the data base. The phasing in of the new system will consist initially of replicating most of the externals of the capabilities available to the staff personnel within the framework of their old system. After the staff is convinced that the new system, operating in parallel with their old system, provides the same or comparable capabilities and is easy and straightforward to use, the old system will be phased out and the process of modifying the new system to exploit its enhanced capabilities will begin.

Replicating the old system starts with the ability to store the information contained in the files of the old system and to retrieve from that store. It should be a straightforward process to generate file descriptions for the categories of information contained within the old file by a system such as the data base system described above. Taking the actual file of information from the 1410 System and converting it to the new data base system is somewhat more complicated. Although this requires the services of a professional programmer who understands the structure of the old file, it is not a terribly complex task and is one that has been done frequently. Since the new system makes it easy to change the structure of the files, it seems desirable to initiate the changeover by preserving the old file structure as seen by the data user (not the programmer) within the new system, and adding capabilities and evolving them only after the new system had demonstrated its ability with the old structure.

Next, the process of trying to replicate the capabilities of the operational programs for the 1410 System within the new system can begin, starting with the less complex ones. Much of the complexity of the old programs relates to the now straightforward tasks of information retrieval and display. These functions can usually be readily accomplished with the general-purpose, user-oriented tools provided in the new system. This process should be undertaken jointly by the implementing organization and personnel familiar with the old system so that reasonable tradeoffs can be made between slavish copying of the old query capabilities and different but simpler capabilities available through the new system. As the capabilities come into being they will be tried in parallel with the old capabilities and it can be demonstrated that they do (or do not) produce the same result. Personnel will become familiar with the new and slightly different ways of doing business and may even, at this early stage, begin to experiment with modified ways of performing some of their tasks.

Finally, the process of converting the remaining, and typically command-specific, programs to the new system begins. Personnel from outside the headquarters will probably be needed to supplement the headquarters staff for this task. This conversion should begin slowly, since a great deal of programming will probably be required and it would be a shame to tie this too firmly to the old mode if the new capabilities offer greater power. I would recommend that the operational programs be priority-ordered and the most important ones tackled more or less one at a time so that the learning process can be brought to bear on later portions of the conversion process as early as possible. Because the time-shared executive enables several users to have access to the system at one time, there need be no interference between development and implementation.

CONCLUSION

The process I have just described holds many potential problems with respect to reliability. It is very important that the general-purpose tools provided by the new system be extensively checked out before the conversion process is attempted. Inadequacies of the tools necessarily reduce user acceptance of the new system. This is particularly true of the executive system which can clobber the parts of the operational program that the command staff might like to exploit.

In conclusion, I should point out that there is a very serious problem, particularly in the Air Force, in bringing such a package into being. In principle, it is ESD's charter to produce such tools but, for whatever reason, it has proven virtually impossible for ESD to achieve the funding support required to contract for such efforts. Until one, or preferably more than one, effort to provide such capability occurs, this will be talk and not action and we will never be able to find out whether these concepts held by many of us are valid.

A case for the right

by THOMAS S. McFEE*
Executive Office of the President
Washington, D. C.

"All twelve violins were playing identical notes. This seems to be unnecessary duplication. The staff of this section should be drastically cut. If a larger volume of sound is required, it could be obtained by means of electronic apparatus." (Author unknown, 1955)

INTRODUCTION

I quote part of this study because it is an excellent example of what can happen when classical analysis methodology is applied by an outside analyst. It was contained in the report of a work study engineer after his study of a symphony concert at the Royal Festival Hall in London. This report was probably based on interviews, visits, simulations, and other activities which can be done in a laboratory environment with a minimum amount of disruption to the user's environment.

I realize it is an exaggerated point; but it is the outcome which we must continually guard against when we try to design automated support for an organization without a thorough understanding of the inner workings of that organization. I use it only to open my part of the discussion on Command Systems Simulation and Design.

If you look at the agendas for the last two Information System Science Congresses, you will notice that we have had sessions similar to this; and if you read back through the papers that were presented and remember the discussions at each of these sessions, you will find that they cover a wide range of subjects.

The ends of the spectrum

The Congress discussions and papers, as well as the trends in system design technology over the last six years, have been controversial. There seems to have been two general philosophies, and they have been quite divergent and, in fact, have ranged somewhere between two ends of a wide spectrum.

At one end of this spectrum has been the philosophy that because of many problems—administrative, technical, security, etc.—it is going to be impossible to allow the technicians to become intimately involved in the day-to-day operations of a command or corporation. Therefore, the only alternative is to conduct surveys, interviews, study documentation and organizations in the abstract and to make heavy use of simulation in a laboratory environment. All of these are used to produce an ultimate flexible system with many general purpose characteristics which can be installed in a customer's environment. Without any connotation to any current political philosophies, and for purposes of this paper, let us call this the "left" of the spectrum.

The best example that I could find of this "left" position was reported in a trade journal. I have deleted any reference to the well-known systems design organization involved.

"Secondly, . . . hopes to avoid the sad experience learned from SAGE. After SAGE was installed, the operator discovered that information he often needed was not available to him. It could not be requested, could not be displayed, nor was the computer able to produce it. By a series of new developments not yet disclosed, . . . plans to design flexibility into the system and into the programming of the system, that will make it automatically responsive to new, unforeseen requirements as they emerge, the system will be able to change itself as it operates." (Mason, 1963)

At the other end of this spectrum (the "right") is the philosophy that only through an involvement with the user, and with the user himself participating to the fullest with the designer in his own environment, can any progress be made. The advocates of this posi-

*On loan to the Office of Science and Technology from the Weapons System Evaluation Group, Department of Defense. The ideas presented in this paper are the personal opinions of the author and do not necessarily reflect the position of the Office of Science and Technology or the Department of Defense.

tion propose that the technician move into the user's environment. They advocate that it takes a continuous process in which the operator and the technician work together learning each other's problems as a team effort, while they develop and define areas for support and feasible ways of doing the job from a technical standpoint.

I happen to feel that this was closer to what Dr. Fubini meant in his opening address to this Congress two years ago:

> *"The purpose is not simply defined by simple statements. I am saying to you gentlemen—do not overplan; do not design the software as if you know what the user wanted; bring the user right into the middle of your machine and make him work with it, and you'll be surprised at how good the man is. . . . This is why the DOD has recently changed the rules on military systems. We try not to deal with Weapons Systems alone, but to bring the users in. We want to increase interaction—continuous interaction between the user and the doer."* (Fubini, 1964)

The philosophies at each end of this spectrum have their advantages and their disadvantages. I need not point these out to this audience as most of you have been involved in systems development somewhere along this spectrum. In fact, the very purpose of this session is to discuss the extent to which success can be achieved and how far towards the "left" or the detached, laboratory approach, designers can operate. Of course, to be practical, somewhere between these two divergent views there must be a trade-off point. Taking the particular command, the particular improvement desired, and the particular financial, personnel, technical limitations which have been imposed, one can arrive at a "best fit" to the situation.

My position in this paper is to support the "right" —if not to convince you completely, at least to swing the pendulum as far in this direction as I can. I am speaking from a number of years of experience working with various users in their environment, or trying to convince others that this is the only way to work. I have been attempting to bring about an improvement in the development of various support systems to assist users in performing their jobs.

I have not always been successful, and I cannot come before you with a list of spectacular, technically elegant, multi-computer systems that have been designed applying this technique. In fact, it is difficult to measure accomplishments in this area, and some have a distorted view as to what is an accomplishment; but the only valid measure of success is an improvement seen through the user's eyes. Here, I have succeeded.

Some qualifications

There are some definitions and a few qualifications that should be made early in this paper. First, I will limit my comments to what is known today as *command system* design. To make sure that you understand that my comments are meant to apply to the design of systems to support organizations that perform command functions, I would like to discuss very briefly some of these functions.

Command functions involve broad analyses of strategic problems. This involves allocating resources; alerting, committing and assessing the capabilities of both your own and enemy forces. These functions require the gathering of large amounts of information, aggregating this information, analyzing and processing it in many different ways and distributing it throughout an organization. Command functions deal with ambiguous circumstances. Command functions involve questions like: What are the courses of action open to my enemy or competition? Command functions involve planning.

A staff must modify, suggest, and define the command functions as new situations and problems arise. When a plan is selected, command functions include issuing of orders. They include monitoring the execution of these orders against the plan. Finally, intelligence may start the cycle over again, generating a new command, a new plan. In summary, command functions are concerned with the total management of the resources of a command or a corporation. Organizations that perform these functions are the command systems themselves.

I am addressing my comments concerning the Support Systems that can be designed to assist these organizations to perform better these functions. In fact, a much better name would be Command *Support* Systems. If we did this, it would help us realize that we are not designing a Command System or replacing, or even augmenting one that is in existence. We are only providing support to an already existing system that is already performing the functions discussed above.

I can speak from experience in the military environment. I have a feeling that the things we are discussing here today are just as pertinent to the industrial management environment at the command function level as it is to the military. Discussions with friends in industry have substantiated this; but I will let you make your own extrapolation to the industrial environment.

Secondly, I wish to make clear that I am not talking about research in the information systems area. If I were, I think I would take a completely opposite view. One of the real problems with operating in the "right" mode is the tendency of this environment to inhibit major and long-range innovation. The user has a job to do. He is going to be involved in his present position, probably for only a short period of time. He wants to see some payoff that will help him. He, naturally, is going to take a dim view of any long-range or any non-mission-oriented research. Yet there is a very great need for research in the command informations systems area. We have much to learn about model building, gaming and simulation, retrieval, data analysis techniques, indexing, document storage, display methodology—to name only a few. Software research efforts need to continue, and we definitely need realistic problems and realistic data to use in these research efforts. But the user's environment is not the setting for this type of work.

I am not too concerned about the status of this type of research because work in the areas that I have just described can be done quite well in the detached laboratory environment. In fact, there is much research of this nature being sponsored for application on non-military problems. Government, libraries, educational groups, commercial and industrial groups, all have a need for this research.

Another reason I am not concerned about this research stems from the fact that I have not found problems in the military that are so different from problems confronting the rest of the world that they require independent research efforts for exclusive military applications. These basic problems are in existence everywhere and realistic problems as well as realistic data can be drawn from less sensitive areas. In fact, from a technical standpoint, even more challenging problems are in existence outside the military environment. This is not to say that the military does not have a need and an obligation to sponsor basic research in command information systems area—it just should not be done in the user's environment.

Now probably not many people will argue with me on this subject, so why did we get in the position of attempting or advocating research in the user's environment? It seems to have come from a problem of semantics, and this was brought about by some artificial definitions which we had to develop in order to get around some serious funding problems. We have not in the past, nor do we have today, any really good guide lines as to how much to spend on development, and how much to spend on research, hardware, etc. This problem is not unique to the military command area and is one which seems to plague research managers everywhere. But in this area, because the technology was new and managers did not have the understanding of the complexities involved, we have been able to get away with some colossal misnomers. We have had some tremendous developmental efforts which have been called research for funding purposes; and some tremendous research jobs which have been funded under developmental budgets.

Unfortunately, these misnomers have done no one any good. It has caused apprehension on the part of some users who were expecting an operational system, and I am sure it has drawn away some excellent research talent and relegated them to menial developmental tasks.

I cannot help but remember a definition in this area that I overheard at the 2nd Congress at the Homestead. If you remember, implicit programming was the number one item for discussion. They said, "You have heard of problem-oriented languages and you have heard of machine-oriented languages, and procedure-oriented languages—well, implicit programming is a fund-oriented language."

I offer no solution to these problems of funding, and I hope some of the other sessions either at this Congress or those that will follow will try to tackle this problem.

So I am talking about the development of an operational system—not research. A certain amount of experimentation for immediate system improvement is necessary in the user's environment and is the only place where it can be done. But research is better done elsewhere. I will assume that the research has been successfully accomplished, and this research is readily available to the development effort.

In summary then, we are discussing development, not research; command support systems, not command systems; command functions, not control or tactical; and we realize that we have some serious funding problems.

Keep in mind that my prime purpose for this paper is to convince you that developmental work can only be successfully performed within the user's environment on a day-to-day basis, with his staff, and that I hold little hope for any successful developmental work being done in the detached laboratory which will have any real payoff or any possible application when later installed in the user's environment. I plan to use the following method to convince you of my position. In my experiences I have come across some very basic characteristics of the organizations at the command function level. Most of these characteristics are probably quite obvious to you all regardless of whether you have been working on the inside or the outside. What probably is not so obvious to the outsider is the importance that intimate knowledge of these characteristics plays to the development of any system that would support the organization.

The hypothesis that I will leave for you is that only through the type of relationship which I am advocating is one ever going to be able to determine these characteristics and the parameters necessary for design of a successful system. As I explore a few of these characteristics, ask yourself, first of all, the necessity of knowledge of these characteristics in the design of a system; and second, the feasibility of determining these relative parameters on the basis of interviews, laboratory simulation, questionnaires, and documentation. These are the only things that are available to my friends on the "left."

The new environment

Probably the most important characteristic of organizations that needs to be examined in the design of a command support system is the environment within which these organizations have to operate. Although much has been written on the subject, I feel that few people on the outside have a real feeling of the depth and size of the effect both in organizations and facilities that has resulted from the change in environment over the last 10 to 20 years.

I am talking about the complexity of the new military responsibility caused by the impact that modern technology has had upon our command environment. Modern complex weapons systems and new technology of warfare have contributed greatly to this change. The number and complexity of our weapons systems, the speed of transmission of information, and the advances in communication and transportation technology, have brought closer together the nations of the world. This has caused an abrupt change in our basic concept of operations. Our command organizations have continually striven to evolve and change their ways of operating to keep pace with the changes in environment. The change has had such a far-reaching effect that only those organizations have been able to survive which could change their objectives, their methods and their structures rapidly enough to keep pace with the changing environment.

Not only has this change come about over the past 20 years, but the situation is so dynamic that the environment changes on a day-to-day basis. Our national objectives supporting a basic foreign policy position have been known to change on a moment's notice, affecting many thousands of people and producing a rippling throughout all agencies of the government. It may have not only military and foreign policy implications, but profound political effect, both in this country and in others. Entire countries in which we have had little or no interest, have come into the spotlight overnight. Obscure Army Master Sergeants in a far away country become leaders of national import. Events which in the past have had little or not effect on military or political operations, all at once become of great importance at national level. In order to survive in an environment such as this, people, machinery, communications, procedures, organizational arrangements, have had to be as dynamic and responsive as the environment itself. The organizations that are with us today have adapted to this environment. Many that haven't are no longer with us or have been relegated to positions of unimportance and we hope will some day disappear from disuse, because as long as they remain out of step they can only work against the interests of the government.

What I have just said you have heard before. But I emphasize again that the effect that this has on command support system development must not be fully appreciated by all of the computer industry or I wouldn't be seeing proposals that contain hardware with Cuba buttons or software that requires modifications to change the Cuba button to a Dominican Republic button. Our command organizations today are changing so rapidly that even if they do have time to make up a simple organization chart, it is obsolete by the time it is off the press. Documentation of procedures is a rarity. Some have stopped printing phone books, and the travel office has become one of the best places to find out "who is in charge."

The implications that this has to our systems design is basically this: If we want to know anything about the environment of an organization, we had better move in and live with it. They cannot afford the luxury of stopping long enough to tell system designers what is going on. And if they try to send a user representative to the laboratory, he can tell us only what it was like when he left, and then only one man's view of the organization.

How they work

Organization is the next basic characteristic of commands. One must find out how they work. Information on this aspect of a command is vital for a system design. But where do we go to find it? The classical way of doing this is to look at organizational charts, charters, directives, executive orders, statements of procedures, rules and regulations, and any other formal documentation on how an organization works. Now even if the changing environment discussed above were not true and organizations did not change and evolve as rapidly as they do —and if the above things were in existence and were up-to-date, how far can they go to help us in understanding the organization and the operation of a command? Based upon my experience, I can say that they are almost useless, *or* at the most, they lay only a very broad groundwork for what can turn out to be a tremendous data collection effort that is necessary to find out how an organization *really* works. But the things mentioned above are the things that are available in the outside laboratory environment, and are some of the things that one will collect from brief interviews or questionnaires.

Probably the most important single thing that I have learned is that if we are to provide a command with a support system or support aids, we must have a fairly intimate knowledge of the inner workings of an organization. Many of the intimate and close-working relationships that exist in the command environment become known only after long association with the command. These relationships are vital to the exercise of command and must be dealt with in any command system designed to support that commander.

The formal documentation of organization and procedures many times tells only a small part of the total story. Many of the procedures, if they are not obsolete, are impractical and people do not use them. Many procedures just don't work and people have learned this and they have long since fallen into disuse. Some procedures just take too long and people have found that they can get away without using them.

Completely outside the formal organization is the informal or back-door method of operation. These are the personal relationships, the temporary innovations that prove effective, the cross-organizational communication channels, the by-passes, both up and down. All these have been developed to meet better the challenge of the changing environment. The larger the organization, the more of these back door routes are developed. Some of these are legal and definite improvements to these procedures, and soon they may even be formalized. These, with a little digging, are not too hard to discover. Most users are proud of their innovations and when they come to know you they will be glad to share them with you.

The others are the ones that are almost impossible to discover—the illegal operating procedures. When I say "illegal," I mean, of course, in relation to their formal charter, etc. Innovation in this area is important if an organization is to survive. Knowledge of these procedures is as important to the system design as the legal ones, but this knowledge does not come easily. Before a commander or his staff is willing to share any of these procedures with an outside group, he must be convinced of their intentions—convinced that they are on his side. This confidence does not come from an occasional visit or an interview, but can only come from a long term personal involvement with each other. He handles these procedures with utmost caution because, if they were cut off or in many cases formalized, he could not do his job and his organization would soon be out of business. What we are discussing is his very future, his career, and what is more important, the actual well-being of our country.

Some of my colleagues may argue as to whether knowledge of these factors is really necessary to the development of a system. To this I can only say that without the knowledge of the detailed workings of an organization, it is almost impossible to measure progress. Unless we are able to determine how an organization really works and be able to determine where we are in relation to where we're going, we have no way of knowing whether we-re improving the situation. All too often I have seen groups recommend changes which they thought were improvements based upon formal organizational arrangements only to find out, too late, that these changes and many more had long since been made. Without this intimate knowledge we have no way of knowing what is realistic, what is relevant, what is practical, or how large a step we should make in improvement. In fact, in a practical sense, we need to know *who* really can make the decisions as far as continuing the system design effort.

The decision process

Another organization characteristic is the basic decision process itself. I assure you that I have a completely different view today of how decisions are made in command level organizations than I had some years ago. This is also something that one cannot pick up secondhand. This is something that one has to see, to be involved in, to have a real understanding of how decisions are made and more particularly, the use of information in this process. This is the aspect of the process that is of most concern to system designers. The ultimate system pictures the decision-maker sitting in front of a screen, a console, or in a board room, listening and viewing the most accurate, up-to-date, well-organized, carefully filtered information—making various decisions which are then handed down to his staff. In reality, this is not the case at all. But how this process really works in a particular organization, what information they use, when they use it, and what they do with it, is vital to the design of any information system. It is, of course, the key to how the organization works.

Some have tried to gain an insight into this decision process by studying records of processes in various crises. Here one has tried to determine who made the decision, what the decision was, when the decision was made, who was involved in it, and what information they used. This process, in general, has met with almost complete failure. The reason for failure is important. It has failed because it had been assumed that decisions were made at one particular time, based on a collection of the best information available at that time; while in reality, the decision process involved the whole organization and many other groups. The decision, although it may be managed by a commander, involves his entire staff. This staff is continually making decisions. This staff is recognizing, filtering, sorting, editing, selecting, translating, inputting and outputting, making decisions on what to throw out, what to leave in, what to get more of, whom to coordinate it with, and whom to inform. Decisions are being made as to relevance and importance. Information that has been stored in libraries and information that has been accumulated in the staff's own minds maybe many years before, are being called upon and used. Mistakes are being made. Data are being translated and summarized. Data are being misinterpreted. Many specific human processes are taking place. Many of these are difficult to understand, much less record and study. The staff is trying to judge, is trying to relate, is trying to remember, is trying to predict, infer, create, associate—all of these involve decisions. And when we put these all together and final decisions are made, they may be based on many other factors, prejudices, agency prerogatives, jealousies and the like, both relevant and irrelevant. Much of this information in decision-making progress is unrecorded. Much of it is done verbally. It's not only the commander for whom we must provide this command support system - it is his total organization.

I repeat, study of an organization and its decision-making process as described above is not something we do secondhand. It is something that takes a long time, close involvement, mutual trust—even to get a partial picture of what really goes on within an organization.

But these things are the very parameters needed in the design of an information system. Without them, information requirements are lacking. Without them, information flows, storage capacity, loading limits and other necessary and very vital inputs for the design of a system are meaningless.

The importance of information

Although the concept of information and its importance to both an individual's and an organization's existence has been known for a long time, we seem to have shied away from considering this as a basic design parameter as if it were something almost immoral. Nevertheless, it *is* very clear that the possession of information is the very essence of power in command level organizations. Show me the man who has exclusive possession of information in an organization, and I will show you the man who is the most powerful in his organization, regardless of his position. Although most people will agree with me on this subject, few will admit it openly. Ignore it if you want, but it is a fundamental fact and a real characteristic of organizations at this level and must be dealt with by any system designer.

The effect that information systems have on the distribution of information, and therefore on power, is of course, obvious. Any change or disruption in access to, possession of, or distribution of information will be resisted by those who may think they will lose power by this innovation. Intimate knowledge of an organization, how it works and who possesses information, is necessary. The ability to be the sole possessor of a piece of information and to gain access to the boss because of this information is so important to organizations at this level that a system designer would do well to remember this when he proposes unlimited, on-line access across organizational lines to various lateral as well as subordinate units. No commander in his right mind will allow such a system to operate unless he *knows* he has full control over who has access, over what he gets, and when he gets it. Security is only a minor problem compared to what I am discussing.

These are not things that are talked about in the open—they are not things that you will come across by a casual association with an organization—they are things you must dig for. Unless the user has been so involved in the development of his own system that he can assure himself that his information is being protected, he will not use the system regardless of how well it works. Actually, this characteristic of an organization can work to the system designer's advantage. Automated support can provide the user with more, better, and faster information, and provide better controls and better access to information than he can possibly hope for without automation. You cannot just *tell* him this. He has to see it. He has to be convinced that he can use it to his advantage.

Flow of information and exercise of authority

One must also understand the differences between flow of information and the execution of command authority. Unless a system designer can determine the differences between flow of information for these two purposes, he cannot properly design a system to support the command. This distinction is not easy to understand; and it is not just the system designer that has troubles with this one. The user needs real help here too. Professor Oettinger describes this problem very well:

While there are legitimate reasons to guard privacy, at least part of this concern arises from a mistaken confusion of information gathering with the exercise of authority. Clearly, the opening of information lines up, down and across, would legitimize a leaping over organizational boundaries that, while essential for real accomplishment, is done nowadays only at official risk and peril. Organization lines reflect lines of authority, but while knowledge is power, the gathering of information is not the exercise of authority. It seems, therefore, perfectly proper for a manager to leap several levels down in search of answers, or for a subordinate to leap across organizational lines and occasionally over his boss' head, so long as decisions and orders travel by normal channels and care is taken to protect legitimate confidences such as, for example, actual salary figures." (Oettinger, 1964)

Resistance to this method of operation is natural, but I feel it comes more from misunderstanding of the differences between the two information flows and that they must and can be separated. How to do this requires some real ingenuity and detailed knowledge of how the organization works.

This method of operation is part of the New Environment and it is here to stay. Dr. Hollomon pointed out that:

"President Kennedy . . . has insisted not only on the right, but the necessity to talk to those who are informed, and not only to those who, by some quirk of accident, occupy positions of authority." (Hollomon, 1964)

This was not a peculiarity just of the Kennedy Administration, but has become a method of operation throughout the government. It actually started with the change after World War II when this country assumed a global concept of operations. There is a need for current information in each agency of the government at all levels. Who needs it, when, in what form, etc., are inputs for system design. As you can see, this is not just an internal problem for a particular command. To gain an understanding of the interagency, intercommand communication problems, you have to be there when it happens.

Command resources

When we discuss resources of a particular command, we normally think of financial resources. When we are writing a proposal, we always have in the back of our mind, regardless of the job we've been asked to do, the question of just what is realistic from a financial standpoint, and try to hit within that ballpark. There are many other resources of an organization which are normally overlooked. Some of these resources we can discover quite easily, but others can only be discovered by operating in the user's environment.

We need a knowledge of the staff itself, its education, its background, its training, its abilities and its limitations, its technical qualifications and experience, and many other intimate factors and characteristics. Although this information is not needed at the level of a particular individual, it is needed in general to help determine the best way to provide support systems for an organization. Answers are needed to questions such as: How far can one go in formatting input? What procedures and methods are convenient? What is realistic within current and projected personnel limitations? In my role as a systems evaluator, I have seen systems that would take Philadelphia lawyers to operate them when only seamen are available. I have seen systems that would require full colonels "mark sensing" to make them work. In organizations at this level, we don't retrain people for a particular job. Unlike SAGE, we can't build a simulation lab and train new console operators. We are dealing with busy and senior people. At this level we don't restrict them for hearing or visual deficiencies because this has little effect on their ability as commanders, but these deficiencies have a lot to do with how they use a system.

A characteristic of organizations at this level is that they are top-heavy with high-level people. There never seems to be enough clerks, typists, and enlisted men to go around. There are some very important reasons why this is true, and some equally important reasons why this is not likely to change. I think this comes from the fact that there is not nearly as much routine, delegatable work going on at this level as one might think. Systems designers very often overlook this fact.

Knowledge of the resources of a command are necessary if we are to determine what are some of the trade-offs between operational convenience and utility. It is necessary if we are to do any cost benefit analysis and develop realistic alternatives. I have not yet found a way by which I can determine these characteristics of an organization from the outside. The problem of self-criticism is very important here. Command level organizations are very sensitive to criticism. They keep their mistakes to themselves. Self-analysis is almost unknown except in a very small segment of an organization. This is not only true due to the psychological and personal aspect of this analysis, but also due to the political and security aspect.

Security

Now let's look at the last characteristic of organizations at this level that we will discuss—security. This is something that has plagued system designers and all organizations that have attempted to use support from outside groups in designing and building systems. I realize that it has been used as an excuse many times to inhibit innovation. But in spite of the abuses of security, it is real indeed. I have never been convinced that it is a problem peculiar to military or intelligence groups, because the problem is broader than one of national security, in its strictest sense.

And if we look at it in this broader context, I think we will see why it is such an important problem. We will also see why it is one of the characteristics of organizations at this level which makes it almost impossible to use groups outside of the administrative and security control of the particular command. To get around some of the security problems, we have tried to design some systems using simulated data, sanitized scenarios, and hypothetical situations. We, of course, have run into the pitfalls of the technician working on the outside not knowing what is reality and therefore not knowing if his ersatz data is realistic. When we tried to have the user provide us with realistic data, he did not know enough about the technical characteristics of the system to be able to provide technically complete data. We have had to build and design based on what we thought it was like on the inside. I see no acceptable technical solution to this problem. I have yet to find a system that was fully checked out on simulated data that didn't hang up the first time I put in a real message.

While working within the user's organization, many of these problems can be solved. There are areas where security and administrative problems prohibit the user from delegating his responsibility to an outside group, or even from presenting information outside his immediate area. In the development of our tactical systems, we were able to get a pretty complete knowledge of a small portion of the total system. But system designers at the command level become intimately involved with the totality of information which is known to only a few, even inside the organization. This information could never be released to a group of outside technicians. A commander does not release this information to just anyone, regardless of his security clearances. But by becoming involved with his organization, working on a day-to-day basis trying to solve the technical problems, a mutual trust is developed and certain administrative controls can be established, allowing the technician to have access to information that could not be generally released.

The technician must, in turn, give up some of his freedom in return for possession of this information. He finds himself in a position where it is difficult to speak or publish reports about his work. He can easily find himself restricted in his technical approach and required to attack the "brush fire" problems part of the time. But the access to this information and the reward that a technical person gets from the chance to use this information in the application of advanced technology, and the personal satisfaction that he gets from seeing his development being applied to real problems, providing real solutions, far outweigh any of these restrictions.

Not everyone will work in this environment. But I am convinced enough of its importance to try to convince others that this is the only approach to making a useful contribution to the problem. The "other way" is not very rewarding or productive.

CONCLUSION

In summary, I have tried to explore some of the characteristics of command level organizations that make operating in the "left" mode very difficult. There are, of course, many others, and I have only briefly touched on the problems that the "right" mode of operation can have. This is the subject of a whole paper itself. In all fairness, I must close on the point that system designers are not the only people that need convincing on this method of development. Many of the users and a lot more of the financial managers still do not support this position.

REFERENCES

1 EUGENE G. FUBINI *The Opening Address* in J. Spiegel and D. E. Walker (Eds.) Information System Sciences. Proceedings of the Second Congress Spartan Books, Washington, D.C. p. 3 1965
2 J. H. HOLLOMON *Letters to the Editor* Science Vol. 143, No. 3605 p. 429 January 31, 1965
3 JOHN F. MASON *National military command system moves forward* Electronics April 24, 1963
4 ANTHONY G. OETTINGER *A bull's eye view of management and engineering information systems* in Proceedings of the 1964 ACM National Conference Philadelphia, Pa.
5 UNKNOWN AUTHOR American association of university professors Bulletin Vol. 41, No. 3 Autumn 1955

The development of a standard system

by PHILIP R. VANCE
The MITRE Corporation
Bedford, Massachusetts

INTRODUCTION

Don Drukey has asked me to discuss the "extent to which a command and control system can be designed on the basis of interview and laboratory data and other facts which can be gathered without actually implementing the system in the customer's environment." I should make it clear at the outset that I am addressing the problem of command system development, and that many of my comments would not apply to the subject of control systems.

We might ask: Why after six to eight years' experience in the development of command systems are we still addressing this sort of topic? The answer is that we have been plagued by serious technical and management problems in every command system that we have attempted to implement, and have yet to find a really satisfactory approach to system development. However, a promising new approach is evolving within the Air Force, based upon the concept of a standard command system.

Most of the impetus for the generation of this concept has come from the activity associated with the installation of a 1410 system at five of the major air commands. This system was an outgrowth of the early stages of 473L and is referred to as the Air Force Interim Command and Control System (AFICCS). The AFICCS approach has been to standardize on the hardware configuration and the support software—which includes utility routines, information retrieval and file building routines, an executive program, etc. Changes to these items are rigidly controlled by Air Force Headquarters. However, the individual users are free to develop specific operational programs within the constraints of the standard hardware and support software package. The concept is simple: Standardize on the capabilities that are required by all users such as the storage and retrieval of information, and the tools for building the special computer programs to meet command-unique requirements. This concept, I am sure, will carry over to the follow-on standard system; and the capabilities will be extended to include such features as a time-shared executive that will permit multiple users at remote stations to access the data processor simultaneously, an improved programming language, better program debugging aids, and the like. We can also expect heavy emphasis on the capability to upgrade the data processing installation and still retain the existing programs, without resorting to recoding or to awkward emulation techniques.

There are two distinct aspects to the standard system approach. One is the technical development of the standard system itself. The other is its application to the unique needs of each command. My contention is that a different approach is required for each aspect of the problem. I would like to discuss these problems in terms of requirements definition, system development, and laboratory support. My opinions are based partly upon a recently completed survey of our experience in the development of the last generation of "L-Systems," and partly upon direct involvement with several of them.

Requirements

In the past the initial command system requirements statements were based upon *ad hoc* studies characterized by discontinuity of effort and superficial analysis. The analysis efforts were often directed more toward supporting the development of the official requirements documents — QOR, SOR, PSPP, etc. — than toward providing a meaningful input to the system designer. These statements typically contained broad, and sometimes vague definitions of requirements. On this basis specific equipment configurations were tailored for each command, but the details of application to functional problems were often left as an exercise for the software contractor and the user. Six- to seven-year lead times were not uncommon for the implementation of the systems; and the rather hazy functional requirements did not provide a good basis for a detailed projection of the system configuration and the required component characteristics.

The standard system approach attempts to avoid the problem of tailoring the basic system components to each command's unique functional requirements by orienting the design toward the basic information processing requirements of many users, and by providing the capability to expand the system as the user's requirements develop. My opinion is that the initial design will be based upon data collected on the basis of interviews with the users, and upon a technical assessment of the state-of-the-art.

A method for identifying specific functional applications of the standard system is also needed. My feeling is that an on-going analysis and requirements-definition activity should be established within each using command. For lack of a better name, I will call this an operational system improvement program. Such a program would be controlled by the user, and directed toward identifying the operational improvements needed to enhance mission performance, regardless of whether they be implemented through simple improvements to manual operations, procedural changes, or the application of data automation techniques. The primary objectives of this activity would be to state requirements in a meaningful level of detail, and to establish overall priorities for implementation. This activity would provide a growing definition of the information processing requirements of the user. It would be conducted in the user's environment.

System development

Our experience has shown that many of the problems encountered in the acquisition of command systems were directly attributable to developmental hardware and software items. In the early conceptual and design stages, the tendency was to overlook the developmental problems, to assume that off-the-shelf components were available, and to schedule accordingly. However, in spite of the desire to make exclusive use of off-the-shelf components, all of the systems have included developmental hardware items. In some cases, they were required; and in other cases, they were inadvertently chosen during the source selection process. These developmental items were almost always late, or were unreliable when first delivered, or both. The software was generally recognized as developmental from the outset, although there was a general tendency to underestimate the size and complexity of the effort. In addition, software production was often plagued by equipment problems, and sometimes by inadequate facilities for program checkout. In general, the overall impact of component development problems was far less severe when the acquisition program provided a full system prototype before attempting implementation in an operational environment.

The task of developing the standard system is primarily a technical one, requiring a strong system engineering effort, and should be the responsibility of the developer. My feeling is that the design and acquisition activities should be centered around a full-blown system prototype. No hardware or software item, whether off-the-shelf or developmental, would be scheduled for use in the field until it had passed thorough acceptance tests in the prototype environment. Maintenance and revision of elements of the standard system would also be centrally controlled by the developer, using the system prototype as a testbed. The testbed would also be used as a vehicle for training military personnel in the technical aspects of the systems operation.

In the terms of Mr. Drukey's question, the activity I have just described is not appropriate for the user's environment, nor for the laboratory. It is an engineering activity that stands between the two. In fact, one of its purposes is to ensure that laboratory items are brought to a reasonable state of engineering development before being considered for use in the field.

Laboratory support

So far I have not identified a role for the laboratory in the generation of requirements, or in the development of the standard system. Is there a role in the application of that system to the unique requirements of individual users? The key to the answer hinges on how well the laboratory can simulate the environment in which the applications must ultimately blend. My feelings on this subject are colored by two efforts which I followed closely, and which provided an opportunity to compare the development of data processing applications in the laboratory and in the field.

The first effort is the Experimental Planning Facility that was developed by MITRE, and sponsored by the 473L SPO. The main purpose of the activity was to gain some insight into the problems of automating the planning function. It was also expected that the first-hand experience gained in integrating the data files, computer programs, and input/output devices would be an important by-product, since 473L was being implemented without the benefit of a prototype development program. We selected airlift planning as the problem vehicle, and established an informal working relationship with MATS to develop the operational requirements. The experimental system was implemented on the old SAGE XD-1 computer at Hanscom Field, and the initial capabilities were successfully demonstrated in early 1962. The system was later extended to include support to the planning of tactical fighter/bomber deployments requiring inflight refueling.

The question is: How much did the experimental activities help 473L? Actually, the effort impacted the 473L program much sooner than we had anticipated. The initial work of developing the operational requirements for the experimental facility influenced the specifications for an interim data processing system for the Air Force Command Post that was also being sponsored by the 473L project. However, the actual experimental work had very little impact. The experimental system suffered from a sterile environment, since it could not compete for the user's attention with the interim system that was being developed in his own facilities.

Although the experimental aspects of the laboratory program had little effect upon 473L, it did have some important side effects. The experience gained in implementing the experimental system led to the general purpose data management concept as personified in the ADAM and COLINGO systems, provided a testbed for some early experimental work in PACCS, and educated a small group of technical people in one aspect of the military planning problem. Ironically, Project 492L (USSTRICOM) reaped more of the benefits than did 473L—the project that funded the effort. The COLINGO system was implemented at USSTRICOM, and two of the people that were trained in the process of developing the experimental facility were able to make an important contribution in the development of an improved planning system at that command. This brings us to the next effort that I would like to discuss.

The USSTRICOM planning system that I just mentioned was developed in the user's environment. It is used to generate combined Air Force and Army force lists with a time-phased deployment schedule for contingency operations; and to test the feasibility of the schedule on the basis of various combinations of airlift and sealift capabilities. A significant factor in the successful development of the system was that the MITRE technical people worked as members of a military team, with one of the key users leading the effort. The MITRE people were first indoctrinated in the USSTRICOM planning process by assisting in the generation of some actual contingency plans. They then *assisted* the military planners in developing an improved planning procedure that would be amenable to data processing support. Finally, they took the lead in developing the specifications for the data processing support system. The computer program was produced and checked out by military programmers, and has made a real contribution to the STRIKE planning operation. How could it miss? Every step of the way the user was either leading the development or approving design decisions. The system had built-in acceptance.

On the basis of experience, I must conclude that command-unique data-processing applications are best developed with user participation, and in his environment.

By restricting myself to a discussion of the development and application of the standard system, I have given a rather pessimistic picture of the role of the laboratory. Actually, I believe that laboratory experiments can be catalytic in developing more imaginative application of data-processing techniques to military problems. The example of the Experimental Planning Facility illustrates the importance of laboratory work in terms of stimulating and channeling technological development, although it also points out some of the difficulties in attempting to justify it in terms of solving a specific user's problem on a schedule. It appears that laboratory efforts should not be focused on short-term solutions, nor should they be expected to produce products that can be transported directly to the field. Perhaps the most effective way to inject the laboratory experience and technical developments into the field, and to bring the key user problems and environmental characteristics to the laboratory, would be to rotate technical people between applications projects within the laboratory and field assignments within the user's environment.

The impact of the new technology on command system design

by Andrew E. Wessel*
The RAND Corporation
Santa Monica, California

A few years ago S. M. Genensky and I wrote a paper titled, *Some Thoughts on Developing Future Command and Control Systems.*** This paper briefly states the position taken in that earlier paper in order to raise some questions as to whether the newer capabilities for "on-line" interactions between users and automated systems have outmoded our previous thinking on the subject of command system design.

In brief, the earlier paper argued for a version of an "on-site" development and design philosophy supported by a military service center which would provide the appropriate specialists on loan to the given user command. Do the newer on-line software packages, the so-called "friendly systems" which are either currently or about to be available, permit the design and development of command systems to be returned to the laboratory? As Don Drukey stated in his letter of invitation to participate on this panel, the question is "the extent to which a command and control system can be designed on the basis of interview and laboratory data and other facts which can be gathered without actually implementing the system in the customer's environment."

The heart of the matter has not changed—can we determine the system requirements in sufficiently realistic detail to permit us to produce a system design capable of being implemented in the time period for which the obtained requirements are appropriate?

I quote my previous views on this point:

> . . . *Even with significant user participation, it has not been possible for the system developer . . . to obtain a detailed description of user's needs and operational requirements that could be translated into a coherent functional design and satisfactorily guide the system designer in the long-term development of command and control systems. . . .*
>
> *Furthermore . . . any such description [of the operational requirements] is of necessity not invariant with respect to time. . . . [Before] completion of a major system development that might satisfy the specified operational requirements, changes in national policy, military strategy, weaponry, roles and missions, and advances in technology would have outmoded the developmental system.*
>
> *Historically, the attempt to resolve the basic difficulties in obtaining operational requirements for command and control systems led to the creation of special agencies within the military services and, recently, within the Department of Defense. The overall management responsibility for the development, acquistion, and delivery of command and control systems was located in such agencies. In general, the eventual system user had representatives in the appropriate agency and was thereby able to participate in the system design phase. In this maner, the system user could introduce up-dated functional requirements that often resulted in program-design changes. However, compromise was inevitable. In order that the established delivery schedules might be met, great pressure was applied to freeze the system design early so that hardware procurement and development could begin. Soon after the selection of the contractors there was a tendency to fix upon an initial operational capability, reserving changes for the second phase of complete operational capability. It was quickly discovered that most aspects of the system, both [the] hardware and software, soon became cast in concrete. At this point, introduction of even relatively small program-design changes, however important these changes might be, proved costly in terms of both funding and delivery time. As this experience became the common pattern in the broad systems approach to command and control developments, the need for a change in developmental philosophy became apparent.*

I see no reason to retract any of these statements. However, the next quotation from the earlier paper raises some interesting questions.

> *While one aspect of the military approach to the development of the command and control systems has always been to involve the user as deeply as possible in the system design phases, another aspect has been essentially technological. In light of the difficulties described above, some of the military services have sought to develop general purpose data processing equipment and general purpose computer programming. This technological effort has been directed towards the development of the equipment and associated software packages flexible*

*Any views expressed in this paper are those of the author. They should not be interpreted as reflecting the views of The RAND Corporation or the official opinion or policy of any of its governmental or private research sponsors. Papers are reproduced by The RAND Corporation as a courtesy to members of its staff.

**P-2941-1, October 1964, The RAND Corporation.

enough to permit a resultant system to be transformed by the user as his needs changed and became better known through operational experience. Whatever future technological developments may bring, it is clear that current equipment and computer programming techniques continue to impose severe difficulties upon the system user. A major time- and manpower-consuming effort is still required for the user to achieve an operationally useful system given the delivered hardware and software packages.

While there are remote indications that technological developments, such as implicit programming, may make the user's task somewhat easier, one must remember that similar hopes were raised by technology a decade ago. It is realistic to note that no currently visible magic technological wand appears likely to be able to solve all of our command and control developmental problems.

Let us take three somewhat different approaches to this issue:

1. Do currently available software packages permit us to deliver to a user systems which can readily (on-line, machine-aided) be modified to create useful, functional software on site?

Direct experience with one such package, JOSS, and knowledge of some of the work sponsored by ESD and ARPA (ADAM, LUCID, Lightning, GENISYS, and GENISYS revisited), lead me to reassert that currently available equipment and computer programming technology continue to impose severe difficulties upon the system user. A major time- and manpower-consuming effort is still required for the user to achieve an operationally useful system, given the currently deliverable hardware and software.

2. Do current developments in this technology indicate solutions to this problem which will be available by 1968-70?

The hopes that were raised 10 years ago have not been fulfilled in several user commands. However, it appears that on-line, machine-aided, "friendly"-to-the-user capabilities for on-site functional programming can be reasonably extrapolated from current technology *if the appropriate developmental approach is taken.*

3. Can we hope to develop in the laboratory such user-oriented software for use with the next round of data automation equipments, or must it be done on site?

Any answer to this question must be based, at least in part, on opinion. We can hardly infer success from past failures, but neither should we expect that laboratory-developed software will find no external use. What we can, and I believe should, do is to hedge our bets by opting for on-site development of specific portions of the required software package.

First, such on-site development would by its very nature involve some of the potential users quite early in the program, providing the technical developers with continuing and much needed user inputs. Equally important, the "education" of the user would procced hand-in-hand with his participation. As an outgrowth of this, we would be in a position to obtain meaningful user support for other appropriate developmental programs. Finally, and most important, user-oriented software would have been developed in time to take advantage of the next generation of data processing equipment. For these reasons, and in connection with the more general and abstract laboratory software developments, it is my hope that the on-site software development program described below will be initiated in time to produce results by 1968.

The pilot program should do the following:*

- Retain and re-emphasize the "lead-user" concept shown to be so valuable in the 1401-1410 Air Force interim system development.
- Insure that the newer technologies, particularly the potential capabilities of the "friendly," on-line, time-sharing technologies will be implemented and explored in a user-command environment and that the successful results will be made available to other Air Force commands.
- Provide a direct means for the Air Force to stimulate and influence the software and hardware manufacturers to see that such techniques are developed with Air Force command system needs clearly in mind.
- Make available to the Worldwide Military Command System (WWMCS) software and hardware which has been proven to be reliable and operationally useful.

The pilot program is to provide certain capabilities, described below, which are to be available at a selected user command already possessing a 1410 interim command and control system. I do not believe that the pilot program should attempt to duplicate the capabilities of the existing 1410 system. Instead, it should concentrate on providing the following working capabilities at the user site:

- On-line** file construction and data entry: to permit individual (non-programmer) users to build, maintain, and transform personnel files by using ordinary logical categories and operational terminologies. In addition, in accordance with standard operating procedure (SOP) to be estab-

*I have chosen to describe the characteristics of the pilot program in terms of an Air Force oriented program tied to the already existing ADP phase of the 1401-1410 interim Air Force systems. However, with suitable change of nomenclature, I feel the program could be made suitable for application within any of the military services.

**Throughout this discussion, a time-shared system is assumed which permits several users to have simultaneous access to the available pilot program capabilities.

lished, this is to permit individuals to use, build, maintain, and transform organizational files. Computer-aided guidance in typical, though selected, operational language is to be provided.

- On-line input/output format construction: to permit the individual users to construct data entry and output formats with a freedom at least equal to that now available with the RAND JOSS system. Computer-aided guidance to facilitate the introduction and definition of new operational terminology in terms acceptable to the machine is to be provided.
- On-line file retrieval and data output: to permit the individual users to retrieve entire files or selected portions (assuming SOP) and to retrieve data previously entered by the individual using any of the constructable data output formats. Computer-aided guidance is to be available to facilitate this process.
- On-line/off-time transfer of 1410 system files and data: to permit individual users to transfer, in accordance with SOP to be established, computerized 1410 data for the individual users in the pilot program system. (This would be two-way transfer in an area to be explored but not required initially.) Computer-aided guidance is to be available.
- On-line product request aids: to permit individual users to request lists of pilot program products, and to request the instructions and/or rules for using any product available on the lists (a product is a software package offering the capability to accomplish a specific operational task such as building a given file, special kinds of arithmetic processing data, or display construction, etc.). Machine-aided guidance is to be available to facilitate this process.

The first, second, and third items are working capabilities to be available for the initial phase of the pilot program. Probably the fourth item and especially the fifth item will have to be developed on site.

Other results to be expected from the pilot program apply to the following problem areas:

- Standardization.
- Technological development.
- User/technical developer relationship.
- Education of the staff.

The recently held Air Force Command and Control Users Symposium† clearly indicated that guidance was required for at least the above topics. It is possible to provide some general guidance as to these and other related issues, but detailed answers can best be obtained only by vigorously pursuing the joint user/technical developer pilot program described above. It is a matter of obtaining sufficiently realistic detail to provide a picture which is clear enough to make basic decisions. For example, "How much of what kind of standardization?" is a matter that requires investigation within the user context of a pilot program. Furthermore, the development of standard techniques and procedures is expected to be one of the outputs of a successful pilot program. These standardized techniques and procedures, with proven operational utility in at least one user command, would provide tested standard products for use and adaptation at other user commands.

As to technological development, while we know in general the kind of state-of-the-art development that would be of value to command systems, the details are again lacking. I believe that they will continue to be lacking, or will be invalid if they are not obtained by a carefully conducted pilot program in a user context.

All this bears on the question of the user/technical developer relationship. It seems obvious that both the user and the technical developer must be involved in the pilot program. We must provide the means for the technical development agency to become aware of the technical deficiencies that are brought to the surface during the implementation of the pilot program. The participation of the technical development agency will both accomplish this and help to create the willingness and capability to correct those deficiencies. It will also help to obtain support and funding for the necessary development programs. It also seems clear that the technical developer and the user still have to learn to work more effectively together. A joint user-technical developer pilot program under the auspices of the user command would do much toward meeting these needs.

To educate high-level military staff in the art of electronic data processing, words can only do so much. As was said often at the Symposium: A product is worth a hundred thousand words. As early as possible, the pilot program must develop products which are useful in the eyes of at least the battle staff of the given user command. I do not think it unreasonable to assume that products regarded as useful by one battle staff will be regarded as useful by others. And if the battle staff is "sold," they are the natural salesmen to the commander. Further, I believe that a significant effort must be made to see that the story of the pilot program is widely disseminated and that demonstrations are held at the user command conducting the pilot program with appropriate staff from other commands attending and participating.

To sum up, it appears that current technology offers a

†Air Force Command and Control Data Automation Users Symposium, December 6, 7, 8, 1965, The RAND Corporation, Santa Monica, California.

potential software development which would permit on-site, on-line functional programming. Such a development would permit direct user construction and adaptation of man-machine routines as a normal on-site activity. Success in this area, at least in the sense indicated by the pilot program and in conjunction with other continuing laboratory developments, would ultimately permit the return of system design to the laboratory, even though the path back to the laboratory leads through an on-site development program.

ON-LINE MAN COMPUTER INTERACTIVE SYSTEMS

EDWARD M. BENNETT, *Chairman*

Session Participants:

JOHN W. SWETS
JACK MINKER
JOHN K. SUMMERS
EDWIN L. JACKS
RAY E. CHAPMAN, *Major, USAF*
JEROME D. SABLE
KEITH W. UNCAPHER

Introductory Remarks

EDWARD M. BENNETT
The MITRE Corporation
Bedford, Massachusetts

The initial drive to achieve an interactive and direct working relation between the many organizational users of a computer and the computer itself first appeared in the design of the SAGE Air Defense System, now over a decade old. Since then, with the success of that venture behind them, advocates of direct on-line man-computer interaction have traveled far, preaching the likely benefits of on-line user techniques in business, engineering, architecture, education, and any other serious human endeavor involving logical processes.

Gradually over the past decade, we have begun to accumulate a diversity of experiences with systems designed to emphasize man/computer interaction. Some of the designers of these systems are passing into third generation developments, others are actively experiencing the results of initial system design efforts, and still others are just now entering the arena of on-line interactive processing for the first time.

This session planned to bring together a group of system engineers and designers with some significant current commitment to interactive processing. The subject of discussion would have been related to matters of value, utility, economics, and the advantages and disadvantages of on-line interactive processing. The emphasis was to be placed on the appropriate role of such processing in the variety of services supporting the military community.

The session participants were invited to supply technical descriptions of the systems in question to be available at the time of the meeting. Because of the postponement only one such contribution was submitted in suitable form for publication.

AESOP—A final report: a prototype on-line interactive information control system

by John K. Summers and Edward Bennett
The MITRE Corporation
Bedford, Massachusetts

INTRODUCTION

AESOP (An Evolutionary System for On-Line Processing) is an experimental on-line information control system realized in the computer facility at The MITRE Corporation. It serves as a prototype for a class of management or command information systems capable of giving the user as much on-line control over system performance as possible. It is a display-oriented system in that a cathode ray tube provides the primary means for man/machine communication. In a system of this orientation, a light pen provides the primary means of user control. This control is not limited to that level of the organization responsible for programming the system, but applies upward to the highest level of executive personnel interested in obtaining direct access to the system.

The AESOP system is concerned with those categories of problems that can be characterized by large amounts of data to be stored, retrieved, processed, manipulated and changed. The system design criteria include the following considerations. First, it is intended for operation by users whose knowledge of data processors covers a wide spectrum. The range of intended users includes top-level managers, operations analysts on their staffs, and system programmers. Such a wide range of users imposes unique requirements. Some of the system capabilities must be simple to use and should entail an absolute minimum of training. Other features should allow the staff operations analyst to make immediate and personal use of the data processor as an analytical tool. The second design consideration acknowledges the need to meet changing and new requirements as they are imposed. Features are incorporated which enable the systems programmer to change the system on-line.

A display and a lightpen provide the primary means of man/machine communication. The display represents a very good means of communication because it can be used not only to present the data and those commands required for operation on the data but also instructions on how to use the system. By simply pointing his lightpen at a selected option, the user tells the data processor what to do.

The AESOP-A prototype* operates on an IBM 7030 (STRETCH) computer (65K memory with 64-bit words) and an IBM 353 disk storage unit with a capacity of two million words. The four user stations each consist of an on-line Data-Display Inc., display console (DD-13), a photoelectric lightpen, an on-line typewriter, and a Stromberg-Carlson 3070 medium-speed printer. The AESOP system is designed to take advantage of the range of capabilities implied by the 7030 central processor and the user station equipment. While taking advantage of the range of equipment capabilities implied, the design philosophy is tailored to respond to the needs of management and command users.

The design of a smooth man/machine interface is not an easy task. Design considerations associated with the development of a display-oriented system will be covered first, followed by a review of the salient aspects of the software architecture and a description of the user's view of the system.

Design philosophy

A keystone of the AESOP design philosophy is the evolutionary approach to the development of the system. As a particular capability was implemented and tested, the insights gained from its use were applied to the development of new capabilities. In particular, this approach was applied to the design of the display programs in the system. The display formats and procedures for man/computer interaction can proceed only so far at the designer's desk. The

*An earlier model of AESOP was reported in E. M. Bennett, E. C. Haines, J. K. Summers, AESOP: A Prototype for On-Line User Control. *AFIPS Conference Proceedings: Fall Joint Computer Conference*, 1965, 27, 435-455.

detailed sequence of user actions and choices for display interaction were thoroughly discussed and specified before construction of the data processing programs was initiated. However, as soon as a capability was operational and tested on-line at the display, additional design refinements frequently became obvious.

One concrete example of this concerns the development of skeleton messages for syntax control (see Syntax Control section). The system prevents syntax errors by the user; if he tries to make an error, it indicates the legal set of actions that can be taken. Experience and insights gained from operation of the system show that a better design is one that indicates to the user what set of actions are legal and, if an error is made, merely provides an error feedback signal.

Experiences such as this led to the following mode of action in software development. When the essential parts of a new interactive display capability were implemented, the feature would be tested and design adjustments identified. Then the additional features needed to complete the capability were programmed and made operational.

Intra system communication

The evolutionary design philosophy applied to the AESOP system imposed a similar requirement on the design of the data processing programs to ensure that new capabilities could be easily incorporated. This was solved by using the AESOP User Language not only as the means of communication between man and the data processor but also as the means of communication between program modules in the system itself. This led to the organizational structure of computer program modules shown in Figure 1. This figure shows how the program modules feed requests into the retrieval program for information from the system data base.

The retrieval program was constructed to interpret messages composed by the user on either the typewriter or, by means of the communication tree, on the face of the display. Any other programs that require data from the retrieval program construct messages identical to the user's and pass them to the retrieval program. The retrieval program does not know whether the messages it processes are user or compute program-generated. Thus the display program that generates a display of 30 lines of data obtained from the data base also composes and sends 30 retrieval messages to the retrieval program. The concept of intra-system communication was further extended to permit many of the lightpen actions to be converted internally into AESOP User Language messages.

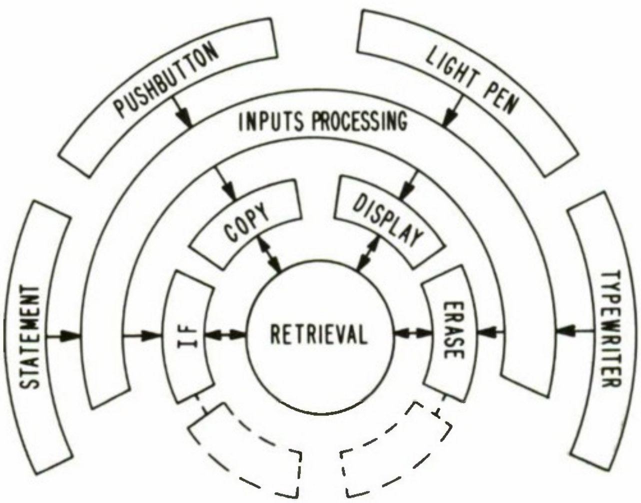


Figure 1—Simplified schematic of program module organization.

This approach has several benefits. New programs can be added which simply call upon already-existing programs to perform operations in a specific sequence. Consider, for example, the COPY program which transfers data from one data base file to another. In so doing, it first uses the retrieval capabilities of the retrieval program to obtain the required data and then uses the updating features of the retrieval program to place the data in another file. The COPY program itself does little processing other than message analysis and message sequencing.

The technique of intra-system messages also contributes to the problem of interface specifications between program modules. In defining the User Language, the program input interface specifications also are defined.

The simplicity of the intra-system messages concept is a definite asset. As large computer systems evolve, their organization tends to become too complex to be remembered by any one of the system builders. As a result, the coordination required among the system builders increases tremendously. Any design approach that tends to maintain simplicity within the system architecture will postpone the day when the size and complexity of the system makes it difficult for the system builder to understand and alter the system to meet current operational requirements.

System Languages

Five different programming languages were used in the construction of the system:

STRAP	STRETCH Assembly Language
SMAC	STRETCH Macro Language
FORTRAN	

TREET*	List Processing Language
TAP	Macro Language for the List Processor

All of the programs sit in a FORTRAN environment to take maximum advantage of the FORTRAN utility system provided with the 7030 software system. The TREET list processor contains both an interpreter, especially useful for program checkout, and a compiler. The programs are debugged on-line in the interpretive mode. When they are accepted for operational use, they are compiled to provide greater speed of operation.

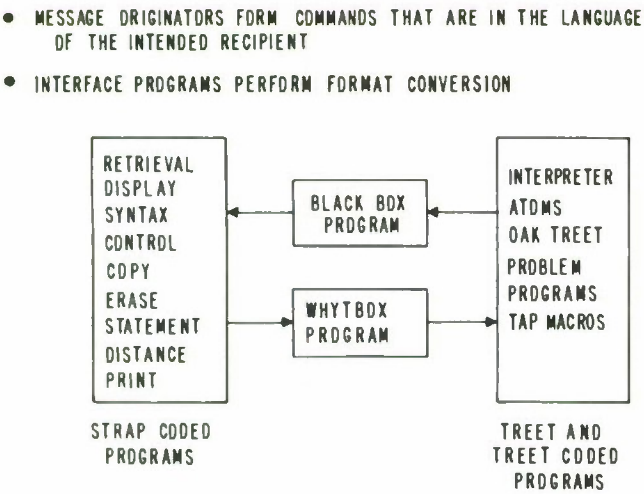


Figure 2—Communication paths between conventional and list processing

Applications programs written in TREET can access and modify the data base in the same manner as other programs in the system. A schematic representation of the method of communication between the list processor and the conventionally coded programs is shown in Figure 2. Communication is a two-way process in that either party may originate messages. The originator composes messages in the language and syntax of the intended recipient. The message then is passed through the format conversion programs (WHYTBOX and BLACKBOX in the figure) to convert these messages into an internal format acceptable for use by the recipient. This approach led to an economical interface in the construction of a hybrid system that employs both conventional and list processing programs. The same approach was used to allow communication between programs written in the FORTRAN language and programs written in other languages.

*The TREET language is described by E. C. Haines in The TREET List Processing Languages; Bedford, Mass., The MITRE Corporation; SR-133, April 1965.

Space allocation

Several applications programs have been overlaid on the basic AESOP system. The demands for space created by their inclusion could not be satisfied by the available core memory in the STRETCH computer. To satisfy this requirement, the memory was logically divided into two areas (Figure 3), one containing the AESOP executive and basic programs and the other the problem programs. Programs, such as the retrieval program, that are used by all problem programs are stored permanently in core. All problem programs are stored on disk. Whenever a specific program set is required to process a request, the programs in core are transferred to disk and the necessary programs on disk are transferred into core.

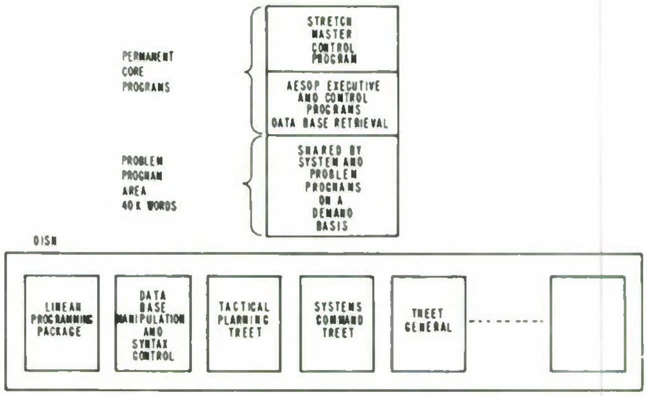


Figure 3—Core memory allocation

AESOP data base characteristics

AESOP is a data base-oriented system; this characterization can be applied to both its basic programs and to its applications programs since these are primarily concerned with retrieving data from the data base, performing operations upon it, and storing the results into the data base. Since the AESOP on-line capabilities are dependent upon the data base structure, it is essential to first describe the organization of the data base.

The data base contains an arbitrary number of files where a file is a named collection of information about similar entities. The entities in a file are called objects; each object is identified by a name or by a number which corresponds to its relative position within the file. The various kinds of information which can be stored about an object are called properties. Each piece of information stored for an object is the property value of some property in the file.

File displays present requested data to the user, and are arranged in matrix form as shown in Figure 4. Each row contains data for one object in the file; each column contains values for one property in the file and is headed by the name of the property to which

Figure 4—File display format

the values belong. The leftmost entry in each row is a line number which indicates the relative position of the object in the data base file. A line number can be used in place of the object name in any input message. The entry to the right of the line number generally is the object name; it can, however, be omitted. The remainder of the entries in each row contain property values.

A maximum of 30 rows of data can be displayed at one time. Many data base files, however, contain more objects than can be displayed in 30 lines. Such files are divided into pages where each contains 30 objects. A page number shown in the upper right corner of the display (see Figure 4) is associated with a given subset of the objects in a file. The first 30 objects in the file are on page 1, the second 30 on page 2, and so forth. A section number appears below the page number and is associated with a given subset of the properties in a file. The number of properties in a section varies; it is determined primarily by the size (length) of the property values and names. In general, the properties in a particular section are logically related.

On-line input capabilities

The AESOP system provides two different ways for entering inputs on-line: typewriter and lightpen. There is a set of legal typewriter messages which spans the complete range of user actions (see the Appendix). Every system function can be activated by inputs from the typewriter; and many of these can be activated by lightpen inputs.

In general, lightpen inputs are easier, faster, and less error-prone than the typewriter inputs. Traditionally the lightpen has only been used to make simple requests or inputs. In the AESOP system, however, the lightpen is used to compose complex file modification and data retrieval messages.

The light-pen is a photoelectric sensing device capable of detecting information on the display scope. The user focuses an aiming circle (a ring of light) on information on the display scope and depresses a switch on the handle of the lightpen. This user action causes information to be transmitted to the program; the program can then determine what datum on the scope face was lightpenned.

The face of the display scope is subdivided into four areas: the center or main screen display, the upper margin display, the left margin display, and the right margin display. The main screen display area is 12 inches square. The margin areas are 12 inches by 3 inches. Data displayed in the upper margin area contain information concerning legality and results of actions taken on other areas of the display scope. Both the right and left margins contain commands from which the user selects the functions to be performed. The upper right margin contains five commands, each of which defines a mode of operation for lightpen inputs at the display: TABULAR, TREE, FILE MANIP, ERASE, and COPY. The last four modes (see Syntax Control section) are used to compose messages with the lightpen. Each time the user activates a mode, a set of commands associated with that mode appears in the lower right margin of the display. For the TABULAR mode, the commands are:

Action	*Right Margin Command*
Next Page	• NEXT PAGE
Previous Page	• PREV PAGE
Next Section	• NEXT SECT
Previous Section	• PREV SECT
Restore	• RESTORE
Display Line	• DISP LINE
Execute Stored Statement	• DO
Print	• PRINT

The commands in the right or left margins are executed by firing the lightpen at the command displayed in the margin. The DISP LINE command allows the user to request an object display for any object in a file display. The DO command allows a user to execute a stored statement in one of the statement files (see Syntax Control section). The PRINT command allows the user to obtain from the Stromberg-Carlson printer, a hard copy output of any display (except trees) shown on the main screen area of the display scope.

The lower left margin contains commands to provide a shortcut method by which a user can display

Page 1, Section 1 of a file without composing the full input message. Through use of these commands the user can build a list of 10 file names in the upper left margin. This feature is useful for both the casual and the steady users of the system.

An example of the display before a list is selected is shown in Figure 5a. When the lightpen action is taken on SET FILES in the left margin, the display changes to present a list of all files in the system as shown in Figure 5b. Each file name selected by lightpen action appears in the upper left margin as shown in Figure 5c. A maximum of 10 file names may be selected in this manner. When the command, FINISH, is lightpenned, the previous file display is restored (see Figure 5d). Whenever a user now fires the lightpen on any file name in this left margin, Page 1, Section 1, of the selected file will be displayed.

Figure 5a—Initial file display and margin commands

Figure 5b—File index from SET FILES action

Syntax control

In the TREE mode a communication tree is available to compose messages for requesting displays,

Figure 5c—Construction of file name index in margin

Figure 5d—Left margin after FINISH action

changing data values, or for retrieving information from the data base. It is a very useful device for the casual user since it serves as a visual aid for remembering the syntax of the AESOP User Language and, at the same time, provides a look-ahead feature to guide the user down the tree. In addition, certain problems associated with typewriter inputs are eliminated. Some of these problems are: (a) remembering message formats, (b) remembering the spelling of file names, object names, and property names, and (c) typing accuracy.

Assume, for example, that the user desires to change the Circular Error of Probability (CEP) for a weapon against a specific target type (see Figure 6). Lightpen action on the word TREE in the upper right margin of the display initiates the tree mode and causes the communication tree to be displayed as illustrated in Figure 7. This tree contains the syntax of a subset of the legal messages of the AESOP User Language.

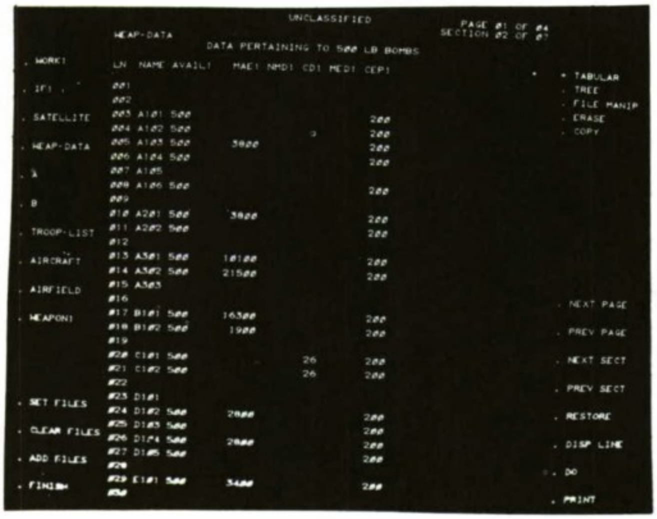


Figure 6—WEAP-DATA file display with weapon code data

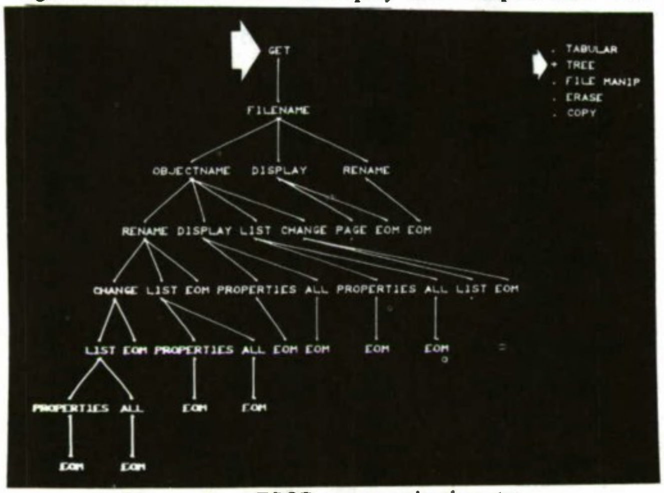


Figure 7—AESOP communication tree

The rules for using the tree are very simple. A user can lightpen only those options appearing at the top of the tree. A legal message is formed by proceeding down a path joined by vectors. Initially, GET is shown at the top of the tree (Figure 7). When the lightpen is fired on GET, the display changes (see Figure 8a), and GET now appears above the tree to form the beginning of the message the user is constructing. At this point, the user has only one option; he must lightpen FILENAME. If the user lightpens any other word on the tree ERROR flashes at the top of the display.

If the user has taken a legal action, the resulting display (Figure 8b) contains a dynamically generated index of the current files in the system. The user chooses a particular file by firing his lightpen on its name. He is next presented (Figure 8c) with three options: RENAME, which will permit him to change the name or names of one or more entries in the WEAP-DATA file; DISPLAY, which will permit display of a particular page and section of the file; and

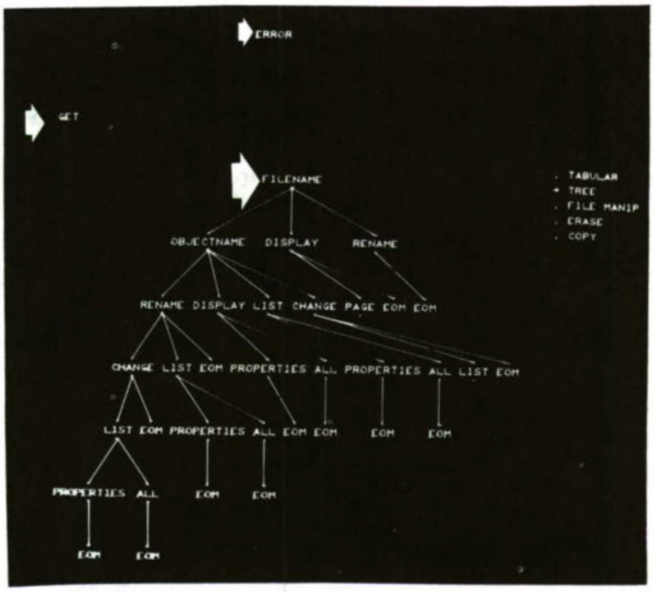


Figure 8a—Result of light-pen action on GET

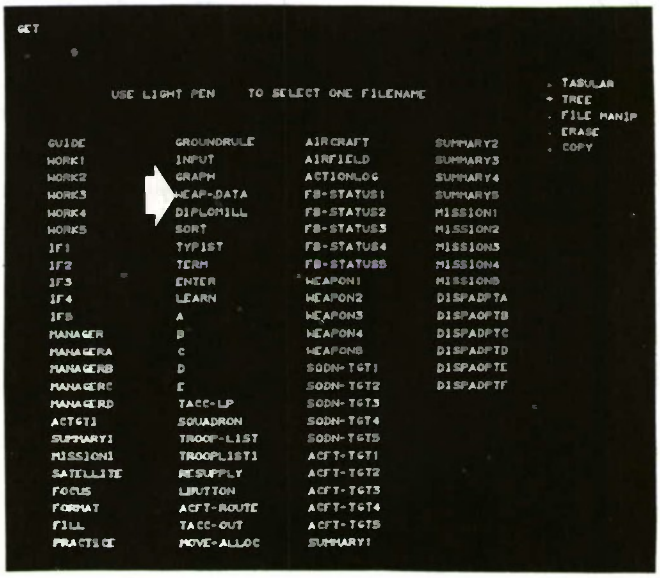


Figure 8b—Result of FILENAME action—list of data base files

OBJECTNAME, which permits the user to choose a particular file entry in order to retrieve or change some of its properties.

When the lightpen is fired on OBJECTNAME, an index of the objects in the file is generated (Figure 8d). Since, in this example, the target type to be changed is A201, the lightpen is fired on this label. On the resulting display (Figure 8e), we can choose to rename this target-type, or to display some or all of its properties, or to change the stored value of some of its properties. When the lightpen is fired on CHANGE, an index of the property names is generated (Figure 8f). Since the property we wish to change is the CEP, the lightpen is next fired on CEP1.

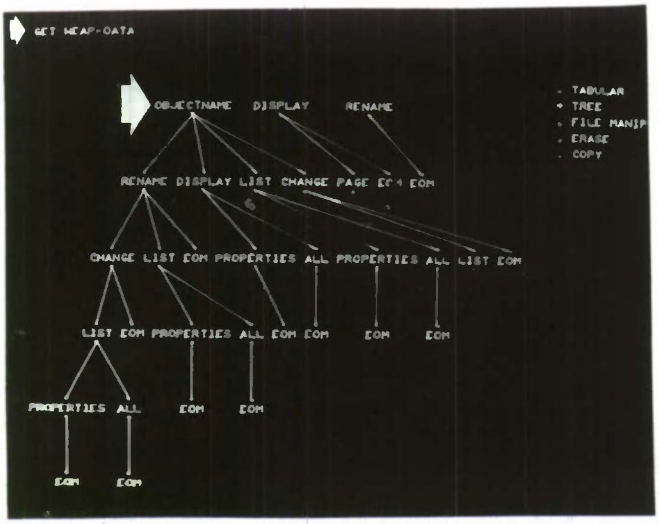


Figure 8c—Next set of tree options

Figure 8f—Result of CHANGE action—list of properties in WEAP-DATA file

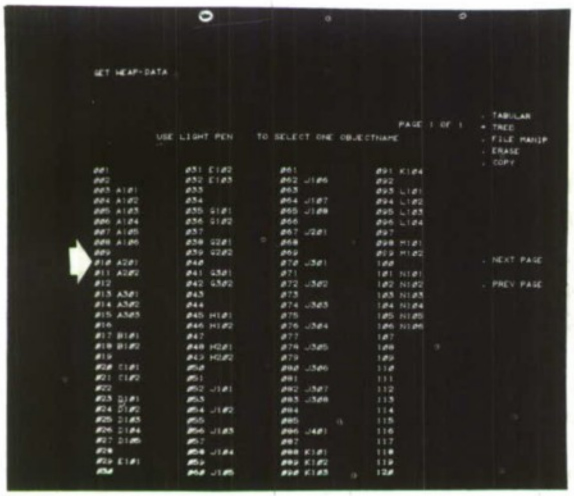


Figure 8d—Result of OBJECTNAME action—list of objects in WEAP-DATA file

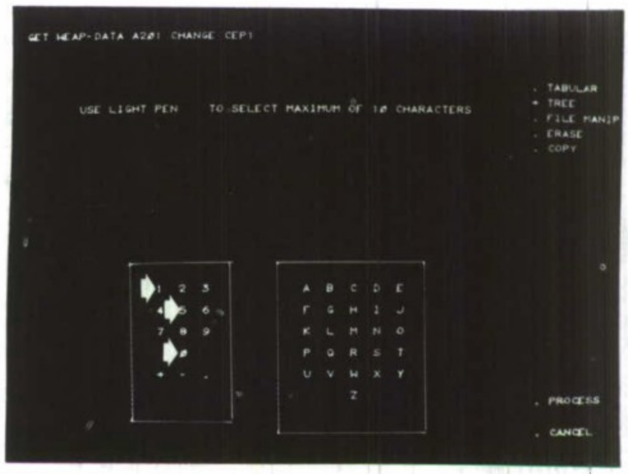
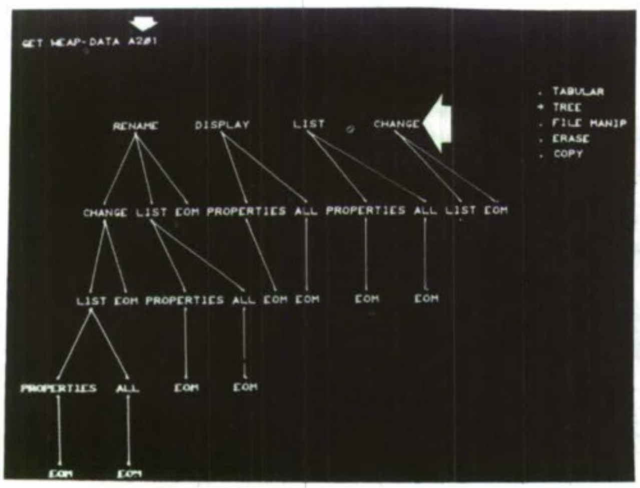


Figure 9a—WORD COMPOSER



Figre 8e—Next set of tree options

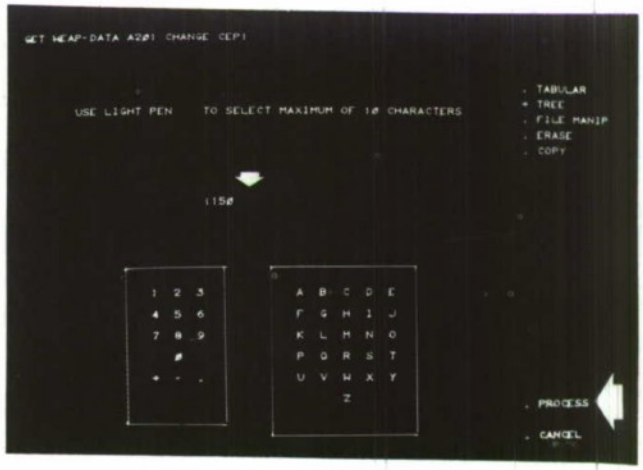


Figure 9b—150 in character accumulator

A word composer (Figure 9a) is next displayed. The word composer permits any alphanumeric symbol up to a maximum of 10 characters to be generated by successively firing the lightpen on the characters in the matrices. As each character is fired upon, it appears in the character accumulator between the two parentheses. Firing upon the numerals 1, 5, and 0 in succession, generates the value of 150 (Figure 9b). If a mistake is made while a word is being composed, a lightpen action upon the word, CANCEL, in the right margin causes the character accumulator to be erased so that a fresh start can be made. When the light-pen is fired on PROCESS the symbol in the character accumulator is added to the message being composed at the top of the display (Figure 9c). At this time additional properties may be selected in order to change their values or the message may be terminated by firing the lightpen on EOM (end of message) in the right margin. The resulting display with the changed data value is shown in Figure 9d.

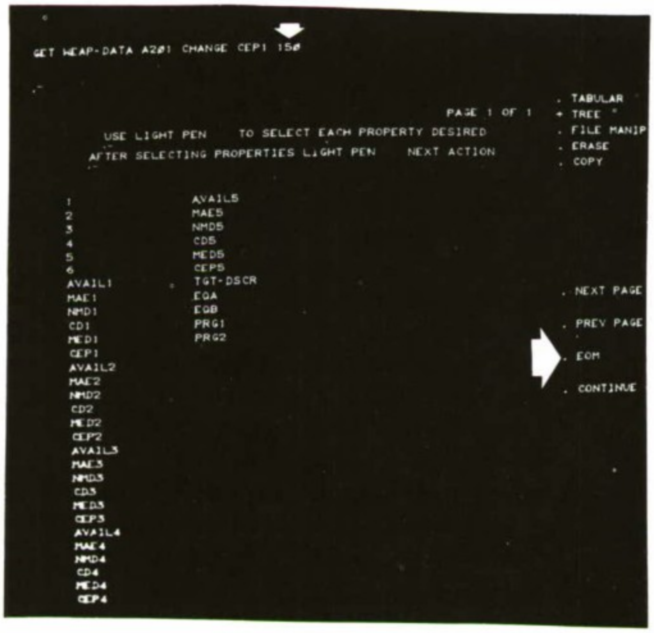


Figure 9c—Result of PROCESS action

When he is composing an input message the user often requires the file display in order to see the data to be modified or manipulated. In such cases, another form of syntax control is used; the COPY mode is an example. When COPY is lightpenned, the commands associated with this mode appear in the lower right margin (see Figure 10a).

When one of the first five commands is lightpenned, the skeleton message for that COPY option appears in the upper margin display area. This message is a legal input message, where a string of dots represents each parameter to be inserted by the user. A pointer indicates which parameter the user must fill

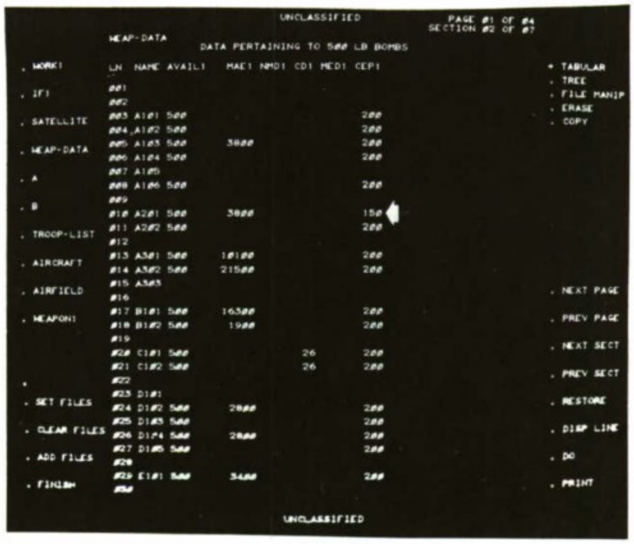


Figure 9d—Result of EOM action—file display with changed value

in next. This message will be used to copy the data from the Airfield file in Figure 10a into an individual working file in the system. Figure 10b shows the message with the first two parameters filled, i.e., the Airfield line numbers and properties specified. The ALLPROPS option indicates that all property data is to be copied. These parameters were filled in by firing the lightpen on the line numbers and on the appropriate commands in the lower right margin. In Figure 10c the empty Work file is shown with all the parameter values inserted for the skeleton message. The Airfield data will be loaded into the Work file starting at line 1, with the properties of the Airfield file mapped one for one into the Work file. Fir-

Figure 10a—Skeleton message from COPY action

ing the lightpen on EOM in the lower right margin causes the message to be processed and produces the results shown in Figure 10d. A private copy of the public data stored in the Airfield file has been made. Changes and alterations can be made to this data without affecting the public data which other users may need.

The remaining lightpen modes, ERASE and FILE MANIP, provide a set of data manipulation actions that work in a manner similar to that illustrated for the COPY mode. The right margin commands for the FILE MANIP mode are shown in Figure 11a and the construction and result of a SWAP message in which lines of data in the file display are swapped are shown in Figures 11b and 11c. The COPY mode allows the user to bring together in one file data from many different files. The user may then rearrange and format the data into the form for a report, using the FILE MANIP and ERASE modes. A hard copy of the display can be obtained on the Stromberg-Carlson medium-speed printer located next to the display console.

Figure 10b—Parameter insertion of airfield line numbers & properties to be copies

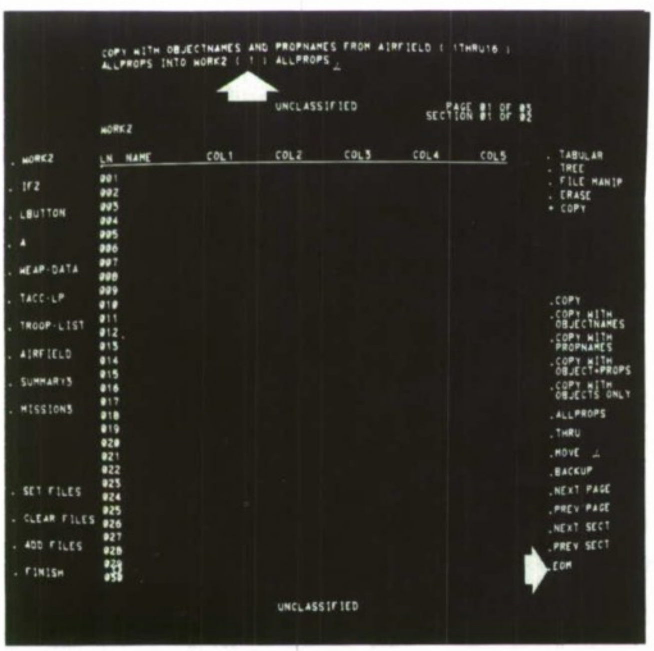


Figure 10c—WORK file & completed input message

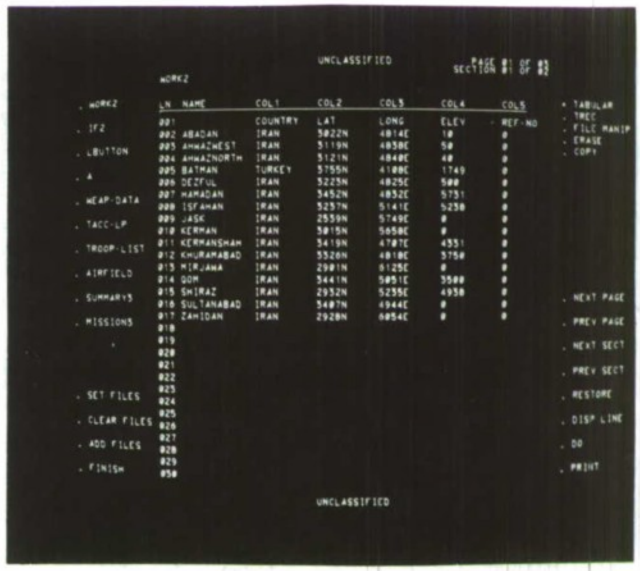


Figure 10d—Result of EOM action—data transferred into WORK file

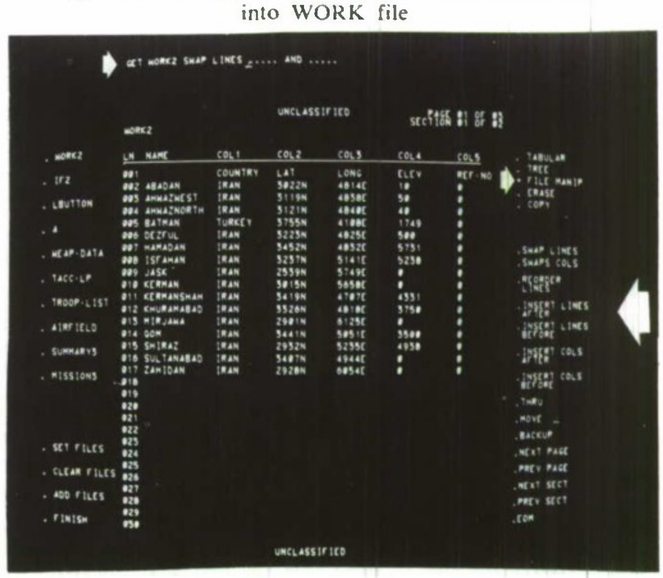


Figure 11a—Skeleton message & right margin commands for FILE MANIP mode

It often happens that a user repeats certain sequences of actions over and over. The messages for these actions can be stored in the AESOP data base as Stored Statements and executed by lightpen actions. The use of Stored Statements is illustrated in Figure 12a by means of a subsetting operation on the Satellite file. Statements can be stored in the data base in files named by a single letter of the alphabet. The B file is shown in Figure 12b. The statements are stored in a line. To look at the complete contents of a line, the lightpen is fired first upon DISP LINE in the right margin and then upon the line number concerned. Figure 12c shows the statement, named Select, that will be operated. This statement contains two system messages, separated by the word, EOM. First, the Satellite file is displayed, and then all of the satellites launched between 1959 and 1962 are selected and stored in a new file. The output file is shown in Figure 12d before the statement is executed.

The Stored Statement is initiated by a lightpen action on the DO command in the Statement file display followed by a lightpen action on the line number of the statement. When Select is executed the selective retrieval program changes the output file so that it duplicates the source file format and, in addition, contains the results of the subsetting operation (see Figure 12e). Stored Statements can be created online, stored in the data base, and operated over and over.

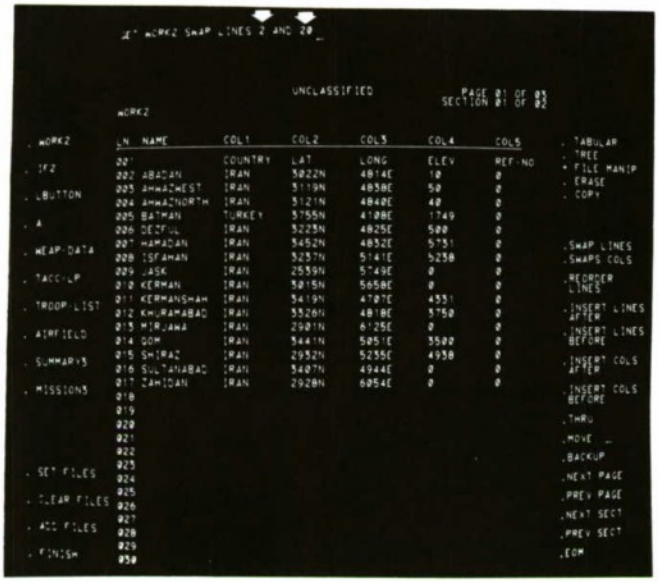


Figure 11b—Completed SWAP message in upper margin

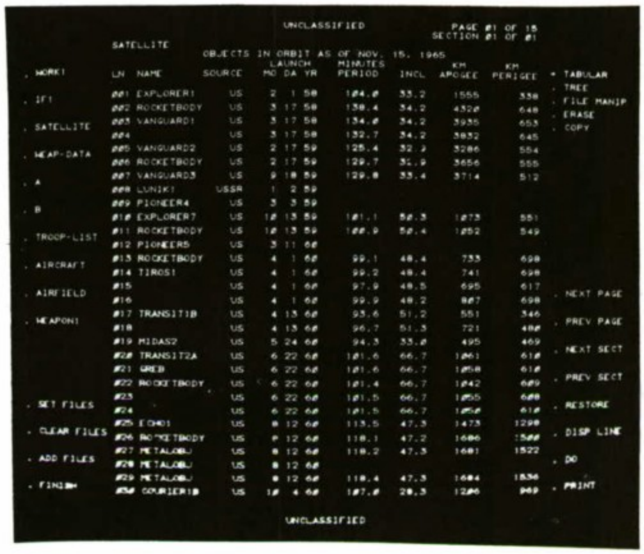


Figure 12a—SATELLITE file display

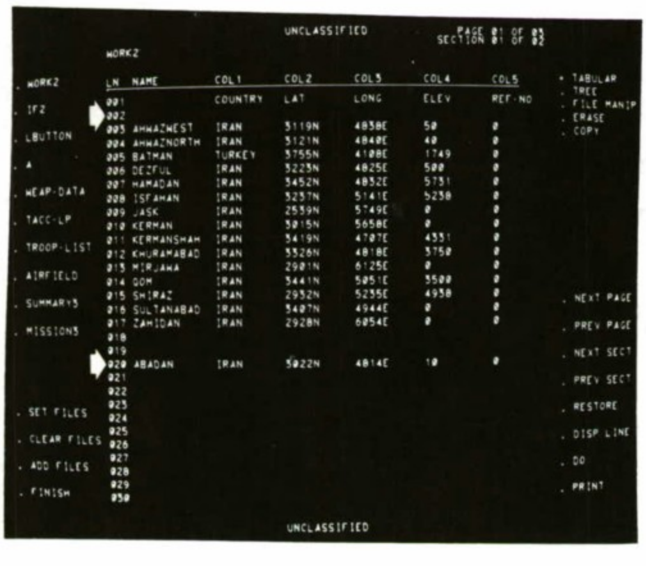


Figure 11c—Result of EOM action—two lines of data swapped

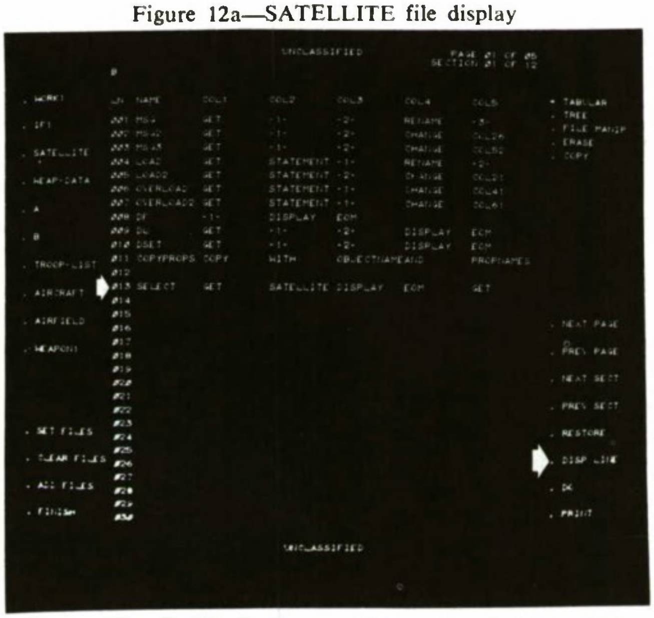


Figure 12b—B file display containing stored statements

Procedure generation

Other methods for processing system data are by either executing or modifying established routines or by constructing new routines using the lightpen and display. For these purposes, a routine called OAK-TREET is called into operation. OAK-TREET, a tool for on-line programming and debugging, is written in TREET, a list processing, procedure-oriented language. There are three modes of operation: OAK, DEBUG and SYMBOL. OAK is used to construct and execute procedures; DEBUG permits the user to display and modify previously constructed routines; and the SYMBOL mode is used to modify individual symbols and their properties. The lightpen serves as the basic input device for all three modes.

The displays are divided into four areas: left margin, top margin, right margin and center display areas (see Figure 13a). The center display contains the work

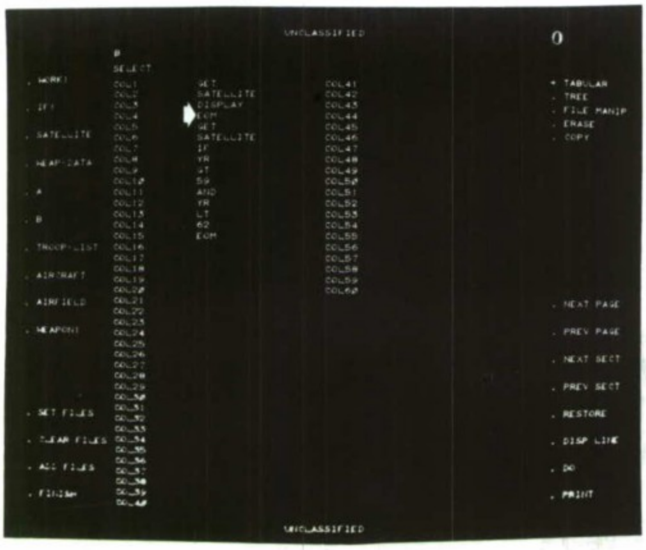


Figure 12c—Object display of statement to be operated

Figure 12d—Initial display of the IF file

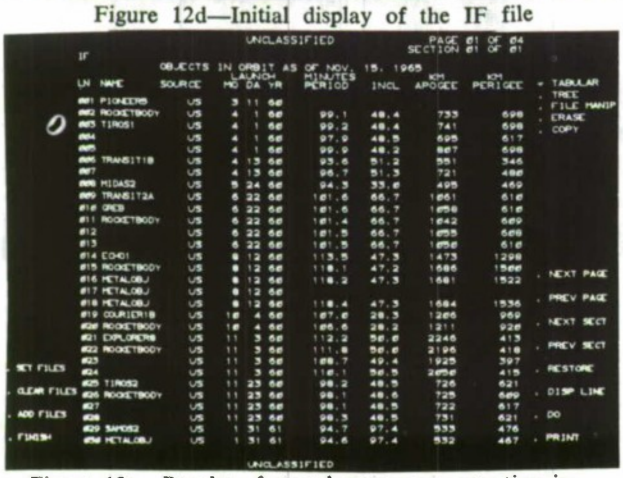


Figure 12e—Results of stored statement operation in the IF file

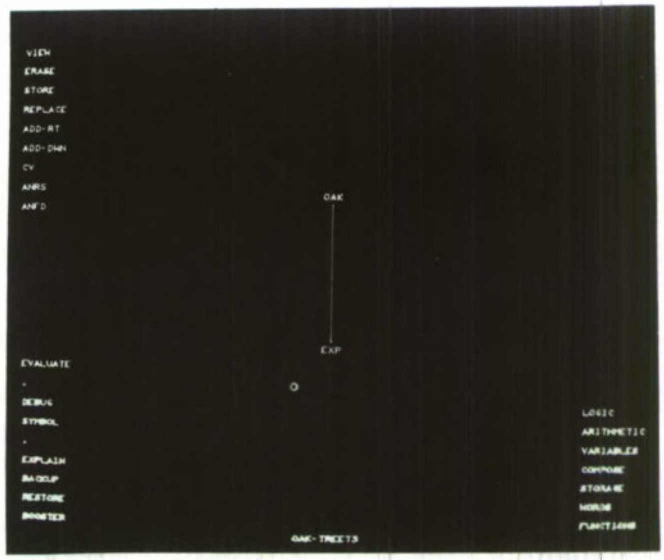


Figure 13a—Initial OAK display

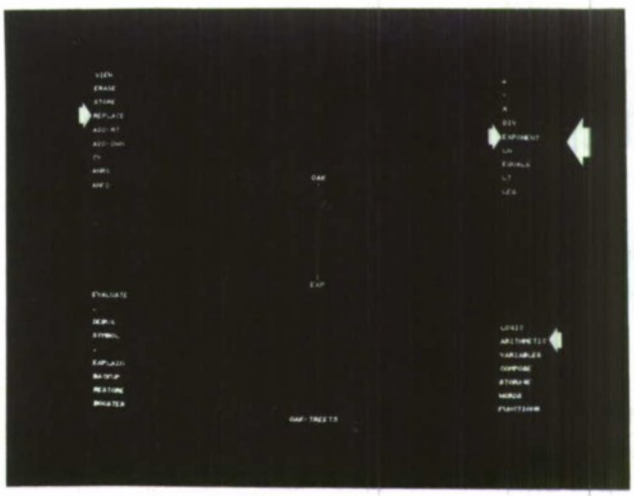


Figure 13b—ARITHMETIC operators in upper right margin

space for construction of algorithms and routines. Algorithms and routines called into the work space are represented as tree structures. The commands in the lower left margin are used to define a mode and to operate on the work space as a whole. The commands in the upper left margin determine how subsequent light-pen actions in the work space are interpreted. A lightpen action on one of these commands causes the command to appear in the command box area in the top margin. The lower right margin contains a set of data classes. When a lightpen action is taken on a data class, a set of operators associated with the data class is displayed in the upper portion of the margin. For example, when a lightpen action is taken on the data class, ARITHMETIC, its set of operators appears in the top right margin as shown in Figure 13b.

A simple arithmetic operation can be used to illustrate the OAK mode. First, REPLACE is fired on by the lightpen and appears in the command box to show that it is the active command. Next, a light-pen action is taken on the word, EXPONENT, in the arithmetic operators which causes the symbol to be placed in the data box area in the top margin, as shown in Figure 13c. A light-pen action on EXP (expres-

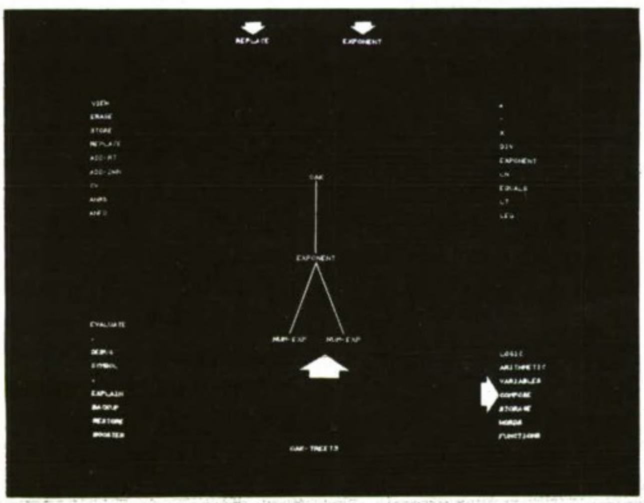


Figure 13c—EXPONENT algorithm in workspace

sion) in the work space replaces it with EXPONENT. EXPONENT has two limbs to show that it requires two numerical expressions as parameters for its operation. The data class, COMPOSE, is fired on by the lightpen to generate the numbers for the algorithm. The right margin display allows a user to compose any desired alphanumeric symbol by successive lightpen actions on the characters (see Figure 13d). As each character is lightpenned, it is placed in a character accumulator above the Word Composer. Lightpen actions on 2 and on PROCESS place the 2 in the data box shown in Figure 13d. When a lightpen action is taken on one of the nodes of the tree, the node is replaced with the 2. In a similar manner, 10 is constructed and replaces the second node of the tree (see Figure 13e). The procedure now constructed is "2 raised to the 10th power." The routine is executed by a lightpen action on EVALUATE in the lower left margin, and the routine and its results are printed out on the console typewriter.

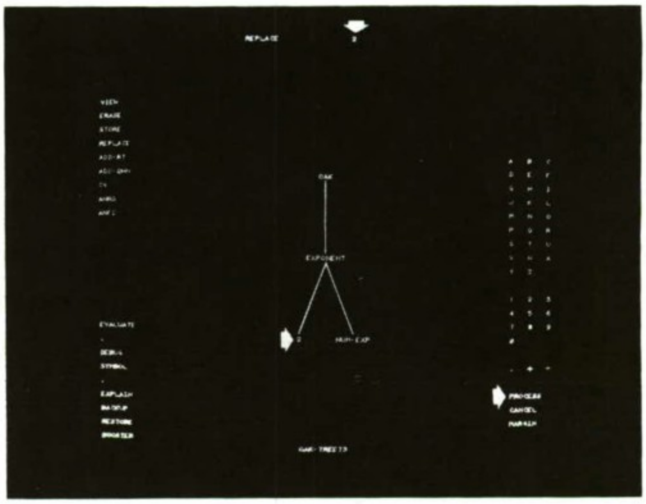


Figure 13d—OAK-TREET word composer right margin display

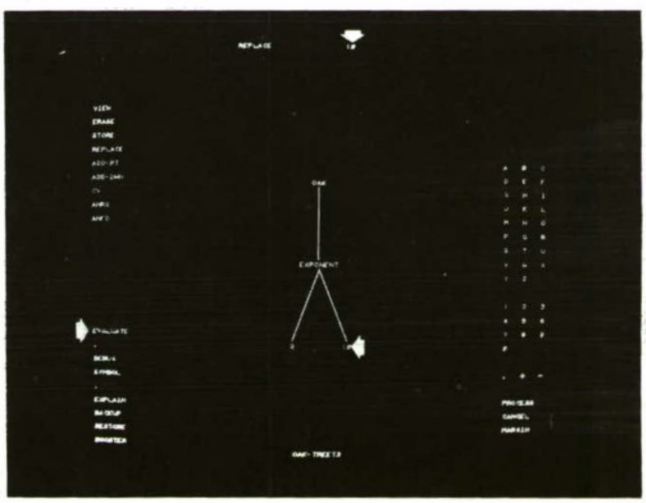


Figure 13e—Numerical values inserted in the tree

The OAK capability permits even more complicated expressions to be constructed. If the entire tree representation of a routine cannot fit on the display, then the workspace can be assigned to display some subexpression of the routine. Figure 13f illustrates

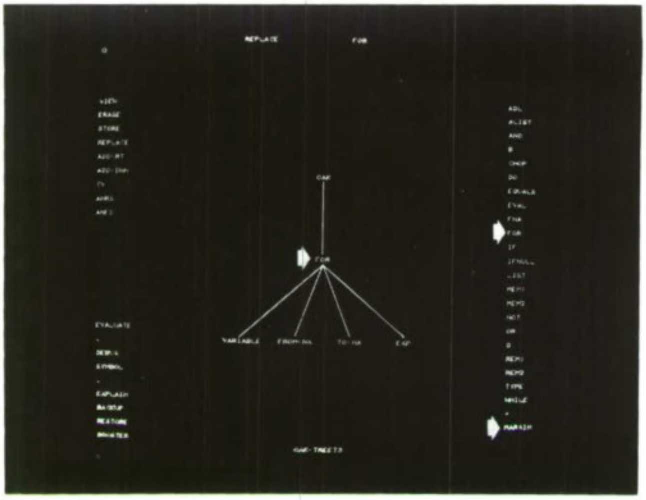


Figure 13f—LOGIC operators in right margin & FOR expression in workspace

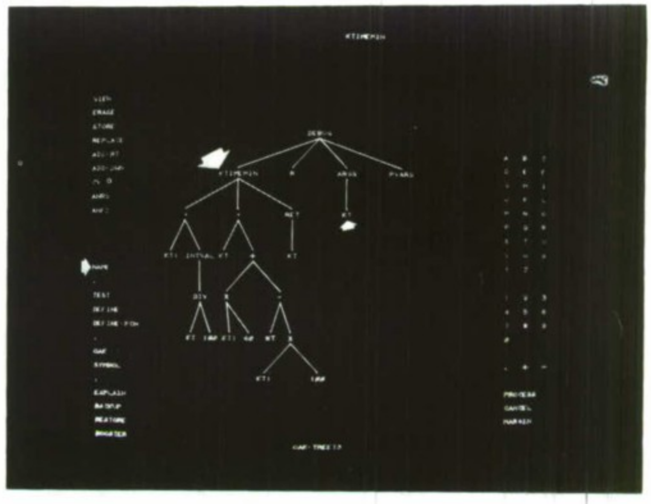


Figure 15a—KTIMEMIN routine in DEBUG mode

the set of logical operators in the right margin and one of these operators, FOR, in the workspace. In this case, the list of operators fills the right margin; hence, the special operator, MARGIN appears at the bottom. A lightpen action on MARGIN restores the data class display.

The detailed structure of existing routines or programs can be examined and modified in the DEBUG mode. In addition, new routines can be generated in this mode. The initial display is called by a lightpen action on DEBUG, in the lower left margin. Changes in the workspace area and in the lower left margin commands for the DEBUG mode are shown in Figure 14. A routine displayed in the DEBUG mode is shown in Figure 15a. The routine, KTIMEMIN, converts military time, e.g., 1330, to time in minutes. Using the word composer, the name of the routine is placed in the data box and displayed when a light-pen action is taken on NAME in the left margin.

The left limb of the tree shows the actual code for the routine written by the programmer. The next limb, R, shows that this routine is a type R routine in that it requires a fixed number of input arguments. The ARGS limb shows that this routine has one input argument, KT. If bound variables had been included in this routine, they would have been placed on the next limb under PVARS. To illustrate how a routine is modified, a check can be added to the routine to test for a given time greater than 24 hours. An IF

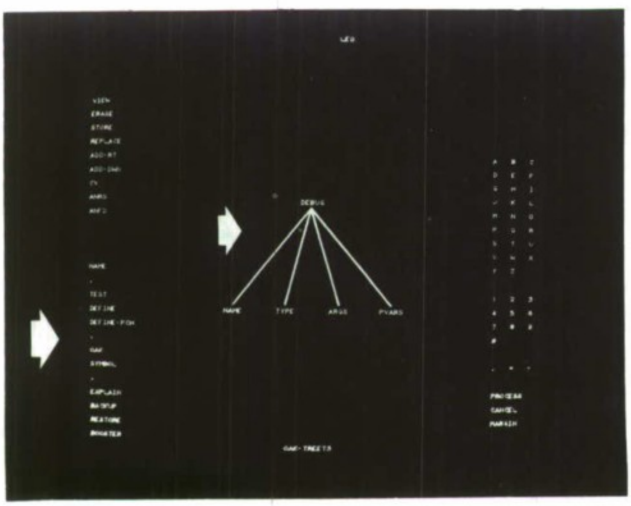


Figure 14—Initial DEBUG mode display

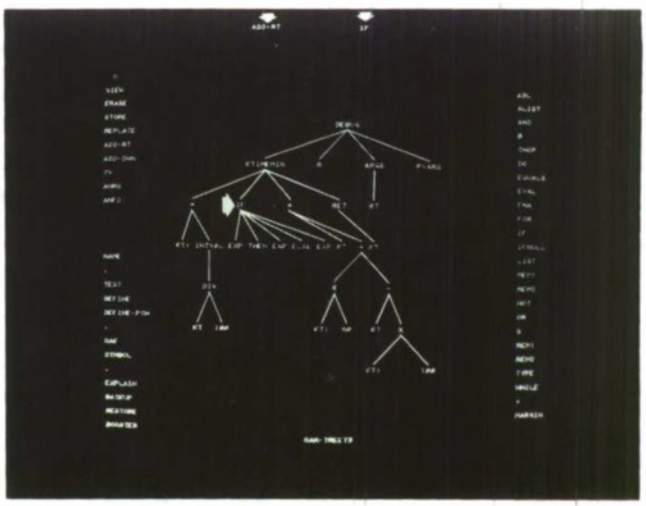


Figure 15b—IF expression added to the routine

statement is added to the routine in Figure 15b, by means of the ADD-RT command. The ADD-RT command causes an expression to be added on the same level and to the right of the node lightpenned. The IF expression in this figure indicates the parameters that must be added for its proper execution.

Some of the steps in constructing the new check for a time greater than 24 hours are shown in Figures 15c and 15d. First, the conditional test phrase is

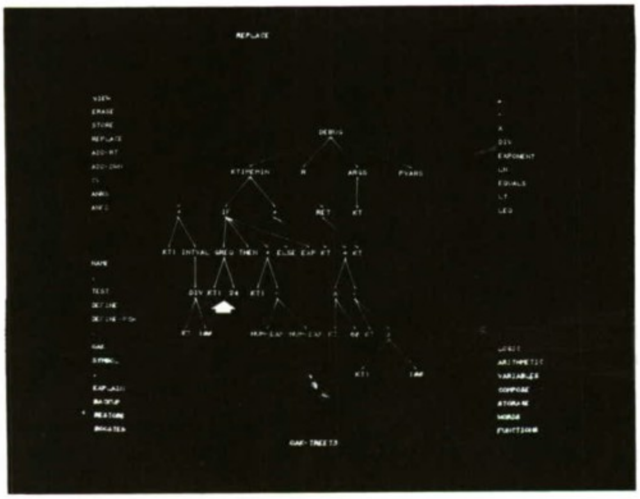


Figure 15c—Conditional test inserted

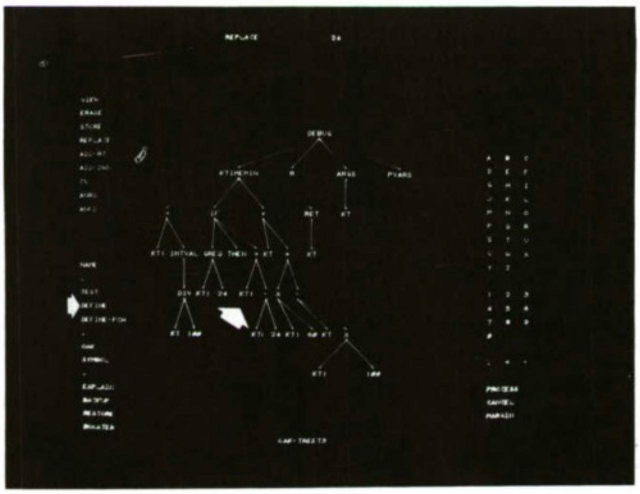


Figure 15d—True condition inserted & else limb erased

inserted. Then the next part of the IF expression is inserted to change those input values that are greater than 24. Finally, the ELSE limb is erased, since the test is needed only for a true condition. The modified routine replaces the original routine by means of a light-pen action on DEFINE in the lower left margin.

An on-line debugging tool such as this is a valuable aid in reducing the elapsed time between the requirement for a change to the system programs and the time when the new or modified routine is operational.

Every TREET symbol has a unique value. This value can be either another symbol or a list of symbols. A symbol can also have properties associated with it. Each property, in turn, has a value which is a symbol or a list of symbols. The SYMBOL mode deals directly with any symbol used in TREET and permits one to modify the symbol and its value and/or its properties. The SYMBOL mode is entered when the lightpen is fired on SYMBOL in the lower left margin, and the workspace and the margin are again altered for the new mode (see Figure 16a).

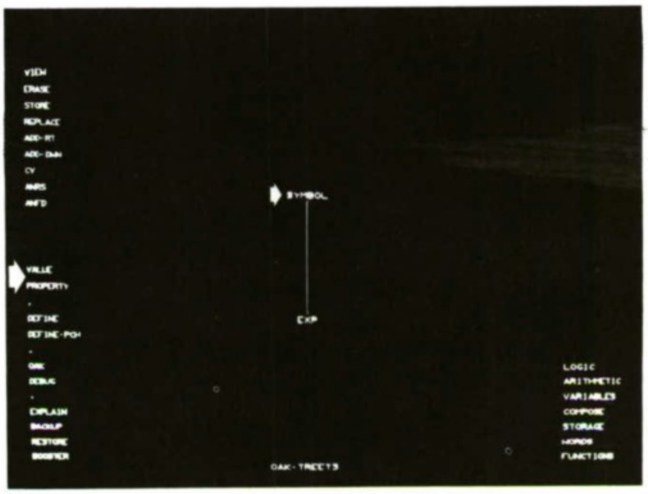


Figure 16a—Initial SYMBOL mode display

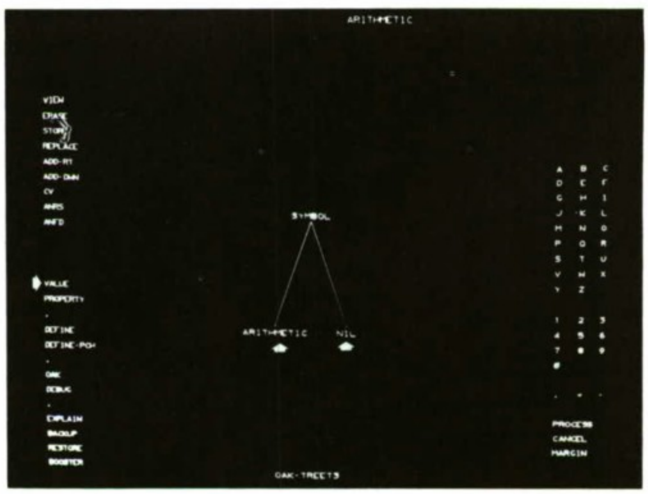


Figure 16b—ARITHMETIC symbol expression in workspace

The contents of the margins in the OAK-TREET displays are properties of system symbols. These symbols control the information displayed in the margins. A user can change the commands, data classes, and operators available to him by changing the property list associated with the system symbols.

For example, the value of a property of the symbol ARITHMETIC can be modified by expanding the list members of the property RIGHT MARGIN (RTMG) for the symbol. In essence, the right margin for the ARITHMETIC data class will be changed to include more operators. With ARITHMETIC placed in the data box, a lightpen action on VALUE causes the symbol and its value to be displayed in the work-space (see Figure 16b). In this case, the value is NIL. The ARITHMETIC property RTMG is brought into the work space when a light-pen action on PROPERTY is taken and after the symbol RTMG is placed in the data box (see Figure 16c). This display

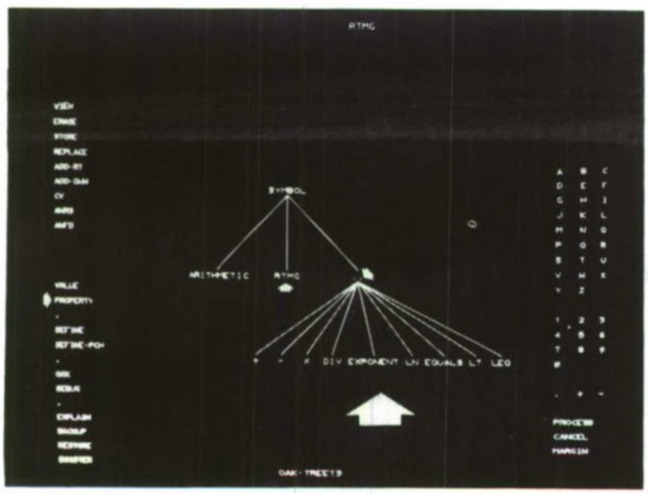


Figure 16c—RTMG property and its list brought into the display

now shows that ARITHMETIC has a property RTMG whose value is a list. The fact that it is a list is indicated by the dominating node of two dots. The list members are the operators of the ARITHMETIC data class that were displayed in the right margin. Now the list is modified by adding other members: FACTORIAL, SIN and COSINE. Each new member is placed in the data box and brought into the work space by the ANRS command.

The ANRS command specifies that a node is to be added as a sister to the right of the node lightpenned. After the SIN member is added to the list (see Figure 16d), the FACTORIAL and COS members are added but not displayed. Instead, as Figure 16e illustrates, a dot is placed to the right of the SIN node to indicate that there is not enough room to display the other known members. Although these members are not displayed they are, in fact, included in the list

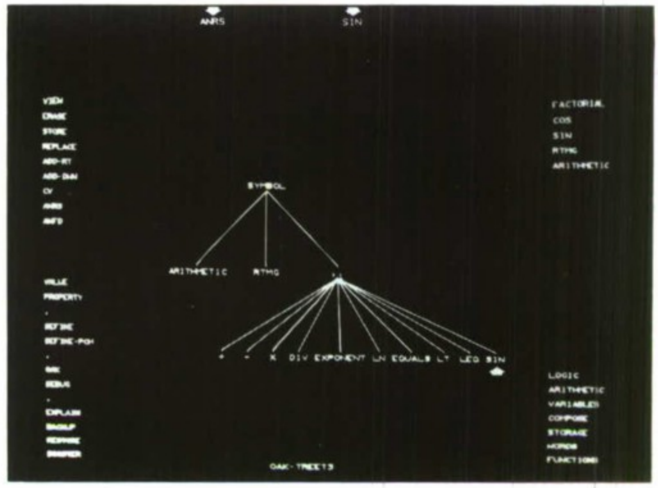


Figure 16d—SIN added to the list

since the upper right margin for the ARITHMETIC data class shows all the operators after a light-pen action is taken on the DEFINE command in the lower left margin (see Figure 16e).

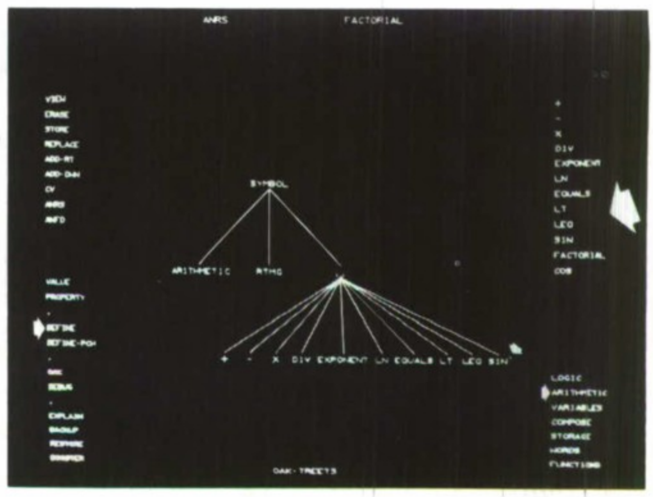


Figure 16e—Result of DEFINE action

Applications

This report has described the general purpose features of the AESOP system. The availability of the general purpose features has eased the task of embedding special purpose applications programs within the system. A number of applications program have, in fact, been placed in the system. For example, an on-line linear program for deployment planning has been included. All of the input parameters reside in data files so that the user may use the general purpose features, *viz.*, file displays and data updating, to view and change the input parameters, respectively. When the linear program is operated, the results are placed in output data files. The user views the solution in a file display and can obtain printed copies of it. If the solution is not satisfactory, the user can change the input parameters and request another solution. This mode of operation puts the operations analyst on-line with the computer in an interactive, problem-solving mode.

ACKNOWLEDGMENT
Joy Dawson, David Ewer, John Glenn, E. C. Haines, Marlene Hazle, Robert Kuljian, Peter Naumnik, Carla Mae Richards, and Dorothy Snider designed and programmed the AESOP system.

APPENDIX

AESOP Users Language

Typewriter Message Notational Conventions

Upper Case Words

Underlined Words

Upper case words which are underlined are key, action defining words which must appear in the actual typewriter input message.

Words Not Underlined

Upper case words which are not underlined are optional. Their presence or absence in a typewriter message does not affect the meaning of the message. They can be included to enhance readability; they can be omitted to reduce typing time.

Lower Case Words

Lower case words describe the kind of information to be inserted by the operater at that point in the message.

EXAMPLE: GET filename . . .

Punctuation

Brackets

Brackets are used to indicate portions of a message which may be included or omitted. The presence or absence of these portions does affect the meaning of the message.

Brackets are nested. A bracketed portion of a message may not be included unless all bracketed portions of which it is a subportion have been included:

EXAMPLE: . . . PAGE number [SECTION number]]

"SECTION number" may not be included if "PAGE number" is not included.

Braces

Braces are used to indicate alternative inputs. One of the alternatives must be chosen. An alternative extends along a horizontal path until the right (as opposed to left) brace is encountered; one cannot switch paths within a set of braces.

EXAMPLE: . . . { ALL / property list }

Parentheses

Parentheses are used in the COPY messages to enclose the object specification. They must appear in the message and be separated from data by spaces.

Abbreviations and Run-On Words

Abbreviations and run-on words, used in message descriptions, are listed below together with their meanings:

(a) filename—the name of a data base file
(b) object—in all cases, *either* the name of an object (object name) or the number of an object (object number).
(c) objectspec—specification of what objects are to be operated on. An object specification can be any combination of the following:
  1. Object name
  2. Object (line) number
  3. Ascending range of object numbers, written #→#. Eg. 1→20 specifies object 1 through object 20.
(d) #—line number
(e) property list—a variable length list of property names.

Key Words Followed By Dots

Two "phrases", the LIST phrase and the CHANGE phrase, appear as optional portions of several messages. For the sake of simplicity, they are denoted by *LIST* ... and *CHANGE* ... except in the messages in which they are the principle elements. They are defined as follows:

LIST ... means LIST [{ ALL / property list }]

CHANGE ... means CHANGE property name$_1$ property value ... property name$_n$ property value$_n$

Message Explanation

Message 14 enables one to change the value of a single property for several objects. The number of values following the property name must equal the number of objects specified in the object spec.

Message 16. The objects to be operated on are specified in the object spec. The data to be erased is specified by one of the following:

ALL	specifies the object name and all property values
ALLPROPS	specifies all property values but not the object name
OBJECTNAME	specifies the object name only
property list	specifies the values of the properties listed

Message 30 causes the current versions of the data base files to be saved on magnetic tape.

A. *Data Retrieval Messages*

Display Request Messages

(1) GET filename DISPLAY [PAGE number [SECTION number]]
(2) DISPLAY PAGE number
(3) DISPLAY SECTION number
(4) RESTORE
(5) GET filename object DISPLAY [{ALL | property list}]
(6) DISPLAY LINE number [{ALL | property list}]

Hardcopy Output Request Messages

(7) GET filename object LIST [{ALL | property list}]

Booster System Typewriter Input Messages

(8) PRINT DISPLAY
(9) GET filename PRINT
(10) FIND DISTANCE FROM {latitude longitude TO {latitude longitude | filename object} | filename object TO {latitude longitude | [filename] object}}

B. *File Modification Messages*

(11) GET filename object CHANGE property name$_1$ property value$_1$ ... property name$_n$ property value$_n$ [RENAME new object name] [LIST....]
(12) GET filename {object RENAME | RENAME object} new object name [CHANGE....] [LIST....]
(13) GET filename RENAME object$_1$ new object name$_1$ .... object$_n$ new object name$_n$
(14) GET filename objectspec CHANGE property name value$_1$ ... value$_n$
(15) GET filename object {ADD | SUB} integer$_1$ property name$_1$ .... integer$_n$ property name$_n$ [LIST....]

C. *Erase Message*

(16) GET filename objectspec ERASE ALL | ALLPROPS | OBJECTNAME | property list

D. *Line Move Operations*

(17) GET filename SWAP {LINE | LINES} {# AND # | #→# AND #→#}
(18) GET filename REORDER {LINE | LINES} #→# AS # # # . . .
(19) GET filename INSERT {LINE | LINES} # | #→# | # # # . . . {BEFORE | AFTER} #

E. *Column Move Operations*

(20) GET filename objectspec SWAP {COL | COLS} propertyname AND propertyname
(21) GET filename objectspec INSERT {COL | COLS} propertylist {BEFORE | AFTER} propertyname

F. *Copy Message*

(22) COPY FROM filename (objectspec) {ALLPROPS | propertylist} INTO filename (objectspec) {ALLPROPS | propertylist}

(23) COPY WITH OBJECTNAMES FROM filename (objectspec) INTO filename (objectspec)

(24) COPY WITH OBJECTNAMES FROM filename (objectspec) { ALLPROPS | propertylist }
INTO filename (objectspec) { ALLPROPS | propertylist }

(25) COPY WITH PROPNOMES FROM filename (objectspec) { ALLPROPS | propertylist }
INTO filename (objectspec) { ALLPROPS | propertylist }

(26) COPY WITH { OBJECTNAMES AND PROPNAMES | PROPNAMES AND OBJECTNAMES } FROM filename (objectspec)
{ ALLPROPS | propertylist } INTO filename (objectspec) { ALLPROPS | propertylist }

G. *Selective Retrieval Message*

(27) GET filename [objectspec] IF propertyname { LT | LEQ | EQ | GT | GREQ } value [{ AND | OR } propertyname { LT | LEQ | EQ | GT | GREQ } value . . .]

H. *Statement Message*

(28) DO filename object [$parameter_1$ ..... $parameter_n$]

I. *Miscellaneous Messages*

(29) DATE dd-dd

(30) SAVE

TACTICAL COMMAND AND CONTROL SYSTEMS COMPATIBILITY

THOMAS P. CHEATHAM, JR., *Chairman*

Session Participants:

VERNON L. MICHEEL, *Captain, USN*
MARCUS C. JORDAN, *Colonel, USA*
KENNETH C. DEMPSTER, *Major General, USAF*
EARL E. ANDERSON, *Brigadier General, USMC*

Introductory Remarks

THOMAS P. CHEATHAM, JR.
McDonnell Company
St. Louis, Missouri

I am most pleased to have this opportunity to participate in the Third Congress on Information System Science and Technology and to talk to you about one of my favorite subjects—"Tactical Command and Control Systems Compatibility." I feel that it is worthwhile to devote some time to this subject from the point of view of the Department of Defense. "Tactical Command and Control" is being discussed, and programs developed for implementing systems, at all levels in all four of the Military Services. Of course, today's tactical forces possess command and control capabilities. However, a principal goal of our present DOD effort is to make command and control more effective by relating them to "real time" in the "real world." The term "Tactical Command and Control" as used here covers the total capability of a commander to order his forces and control his weapons. This capability consists of the staff and facilities at the joint tactical command headquarters, the communications to his component commanders, the staff and facilities of subordinate commands, and so on down through the various echelons to the combat forces.

Looking at it in another way, the tactical command and control capability is the sum of the human contribution, the contribution of machines, the organizational structure in which both work, and the procedures used to perform the tasks which are, themselves, usually dictated by the assigned mission. In this structural organization, men are assisted in these tasks by communications systems, computers, input-output equipment, display devices (from simple "grease pencils" to sophisticated automatic displays) and all the other paraphernalia incident to the accomplishment of the task. It is this particular interrelationship which should make our subject of interest to a "Congress on Information System Science and Technology."

Our primary area of concern is related to the words "SYSTEMS COMPATIBILITY." Here, we are concerned not so much with the "Intra-Service" systems compatibility problem (which also exists) but with the "Inter-Service" systems compatibility problem. The problem of inter-Service systems compatibility is generated where these "systems" function in support of military forces operating "jointly" or as components of a Joint Task Force. This is the type of operation currently being conducted in Vietnam. Warfare such as this is characterized by its highly fluid nature. Units are shifted rapidly and may be committed to widely dispersed offensive and defensive operations. The associated tactical command and control systems capability will, accordingly, be stretched and strained over the extended areas of the operation.

Tying together the characteristics of complexibility, great flexibility, diversity and mobility, all normally inherent in the conduct of tactical operations, is no small challenge. Providing "Inter-Systems compatibility" for the command and control of this complicated machinery is an area all too often overlooked in the normal preoccupation with the individual tasks of designing and building the "individual" weapons systems, control systems and command systems needed by the tactical forces.

In order to view inter-systems compatibility in its proper perspective, it has been necessary to broadly define what the phrase means and then to examine both the operational and technical aspects of this definition. Since command and control systems and the associated communications (commonly referred to as C^3 systems) are basically "Information handling systems," achieving compatibility between them has been treated as a process of *accomplishing an effective information exchange between them.* The goal is to provide the "commander" at each specific echelon with the information required to aid him in arriving at timely decisions or to accomplish his required command functions. Although this may be an easy statement to make—it is a far more difficult task to accomplish. For example, compatibility between systems should be included as a requirement for system design only where necessary for intercommunications in joint operations. Therefore, someone has to be charged with first analyzing this specific problem from a "joint operational standpoint" in order to specify such things as:

Who must communicate with whom.
What information must be exchanged.
The character of the information.

It was also quite apparent that tactical C^3 systems embrace a wide variety of systems and, therefore, compatibility between such systems in a "joint environment" cannot have a singular meaning. In some cases, systems perform similar functions; in other cases, the functions performed may be different but related. An analysis shows that different information transfer requirements can be associated with each of these types of relationships. This implies a recognition of the use of different information "sources" and "sinks" within systems, as well as the different kinds of information being transferred within the context of being "compatible." The appropriateness of being compatible becomes the key consideration. This involves such matters then as timeliness, detail, accuracy, and other such descriptors of information content to be meaningful. This problem was referred to the JCS by the Secretary of Defense in July 1964. Captain Vernon Micheel of the JCS Joint Command and Control Requirements Group will shortly tell you where we stand now in relation to this particular problem area.

If we look at the JCCRG action in analyzing the "operational" requirements for joint systems compatibility, as the first step, then we should look at the actions of the JCS Joint Standardization Group as the second important step. The JSG is primarily concerned with the technical aspects of inter-systems compatibility. This aspect of the problem is primarily communications oriented since the information transfers between systems take place as signal transmissions. This area also is not completely under the purview of an "individual" system designer since his system is always only one "end" of any required inter-system communications link involved. Without advance agreement between the two individual system communications points on all of the required technical characteristics (channel characteristics, modulation techniques, synchronization schemes, character coding and formatting, etc.) not only can the required message transmission be blocked from getting through; it may not even be able to get started. Tactical data systems standards, as the name implies, represents actions being taken to standardize on some of these variables when the inter-system exchange of data between tactical systems is required. Lt. Col. Marcus Jordon of the JCS Joint Standardization Group for Tactical Communications and Control Systems will cover this particular problem area, and discuss the efforts of the Joint Standardization Group in the related area of ensuring compatible tactical communications equipment for use in the C^3 systems.

A third important step in our efforts to solve the systems compatibility problem was taken in March 1966 when Deputy Secretary of Defense directed the establishment of the Joint Service Office for Advanced Tactical Command, Control and Communications. This office, in itself, has no directive authority. It is really a very small, full-time group of Service technical experts serving as a coordinating committee to assist the Director of Defense Research and Engineering. The primary group activity is focused on the research and development programs of the Services to help achieve tactical systems compatibility by seeing that equipments are developed for use by more than one Service, where possible, and recommending for consideration the initiation of new developments, if necessary to improve Service operations and wherever possible to augment and improve joint operations. Putting it another way, the JSO assists DDR&E by performing detailed reviews of the on-going or proposed Service R&D programs in the tactical C^3 area in an attempt to determine that unnecessary duplication does not exist and that the Service considers all possible technical, as well as operational compatibility aspects. Major General Kenneth C. Dempster, Director of Operational Requirements and Development Plans, DCS/R&D, Hqs. USAF, will cover this area.

Finally, we have the problem facing the Military Services in the actual creation of the tactical C^3 systems. Designing and building a tactical command and control "system" is a complex and difficult undertaking. Managing the birth and evolution of one is even more difficult. It was a difficult enough problem when the manager was faced with the task solely of creating a "system" that would accomplish the desired goals of its intended user. When we compound this problem by adding in the requirements for maximum use of standardized, readily supportable "common" components; the desires of the "user" to obtain equipment that never fails in use, requires little or no space, consumes practically no power, weighs next to nothing, can be moved anywhere, at any time, by any means, and *above all*, meets all the requirements for inter-system compatibility in a joint tactical environment, we can see that the Military Services have a very difficult task. The critical factor is not one of technology alone, but rather one of insuring the proper balance between the "users," the "joint requirements," the "technical manager," and the actual builder. General Earl E. Anderson, Deputy Chief of Staff (Research, Development and Studies), Hqs. USMC, will cover this very difficult problem area.

In conclusion, I would like to once again point out that the requirement for inter-systems compatibility in the tactical command and control environment has been well documented, aggressively supported at top DOD levels, and is being addressed by many agencies throughout the Military Departments. It is, however, a very complex and challenging task. It cannot be solved by "dictum" alone; it will take place by strong and intelligent evolution. Similarly, solutions to the broad problem of inter-system compatibility are not to be found alone in the specific development of equipments or standards. The solution will come only through an understanding of the whole problem across the board and then by the complete and wholehearted cooperation of all parties involved, not only within the military family, but those representatives of industry and the scientific community such as represented here today.

Our goal is clear, our task difficult. The final measure of our success, however, may only be determined in the field. Since, when all is said and done, actual inter-systems compatibility is achieved only when the respective systems are operationally deployed and the required information exchange *actually takes place*. We cannot afford to fail.

The role of joint command and control requirements group in tactical command and control system compatibility

by VERNON L. MICHEEL, *Captain, USN*
Joint Chiefs of Staff
Washington, D. C.

INTRODUCTION

It has long been recognized that a commander needs the best command and control system he can get. It wasn't surprising that the Services were quick to apply advances in information system technology to improve their command and control systems.

It wasn't long before it was recognized that there was a need for compatibility at the national level. In January 1960, the Joints Chiefs of Staff (JCS) undertook a study leading to the development of a comprehensive plan for a joint military command and control system. In the implementation of the plan, joining into a single system each of the command and control systems of the Services, Commanders in Chiefs of unified and specified commands (CINCs) and supporting Department of Defense (DOD) agencies, there was a serious compatibility problem between the systems because each system had been developed to meet individual requirements.

It took a bit longer to recognize that the Services were going their individual, separate ways on tactical command and control without any coordinated effort being made to insure that the tactical systems could exchange necessary information during joint operations. To forestall a compatibility problem in the tactical area, the Secretary of Defense, in July 1964, requested the Joint Chiefs of Staff to conduct a study which would give consideration to:

- The degree of commonality of operational, procedural and functional goals of service systems then under development.
- The doctrinal, data exchange, and procedural standardization necessary to facilitate compatibility when these systems are automated.
- The suitability of developing formal requirements for deployable joint task force command and control facilities, which would satisfy the requirements of the 1968-75 time frame.

Joint command and control requirements group in tactical command and control compatibility

Within the Organization of the Joint Chiefs of Staff (OJCS), the Joint Command and Control Requirements Group (JCCRG) under the Director, Joint Staff, is the central point of contact for World-Wide Military Command and Control System (WWMCCS) matters. The assignment of the study on problems associated with tactical command and control systems was a new field for this Group but was a logical assignment considering the Group's past experience in developing the Concept of Operations for the WWMCCS and the Master Plan for the National Military Command System (NMCS).

So, JCCRG, together with Service representation, proceeded to develop a study which would provide a solution to the problems the Secretary of Defense had posed. The result was the World-Wide Tactical Command and Control Study, August 1965. While this study did not provide the answer to all the problems posed by the Secretary of Defense, it was a start in the right direction and formed the basis for continuing programs.

The rationale used in approaching the compatibility problem was basically to go back to the operational requirements of the user.

First, determine, by Service, the operational tasks and functions of the user which must be supported by the command and control system. (Land maneuver, close air support, amphibious operations, etc.)

Second, determine the operational tasks and functions common to more than one Service.

Third, suggest appropriate areas for standardization. (Terms, language, procedures, etc.)

Fourth, determine the supporting command and control compatibility requirements. That is: Who must talk to whom? What information must be exchanged?

What is the character of the information? Note that the study is not hardware oriented.

By following this rationale it was felt that the analysis would support conclusions and recommendations which would lead the way to the point where the technicians could take over and give technical character to the solutions to the problem.

Early in the development of the study it became apparent that doctrinal differences between the Services for certain operational tasks might form a block to securing agreed solutions in the study. In order to prevent this block from occurring, a separate study was conducted by the Joint Chiefs of Staff to determine the unresolved (divergent) issues between the military Services in the tactical command and control area. The study revealed only one divergent issue which the Joint Chiefs of Staff were required to resolve; the concept for airspace control. Specifically, the degree and manner of control of airspace over the combat zone. Once this problem was resolved no other doctrinal issues prevented the accomplishment of the study.

The mechanics used in expanding the rationale was to:

a. Compile and compare the operational tasks and functions of the Services in joint operations.

1 A total of 12 broad operational tasks were addressed.

(a) Land combat
(b) Close air support (CAS)
(c) Air strike/interdiction (less close air support)
(d) Air defense/AAW (Interceptor)
(e) Air Defense (SAM)
(f) Airborne/Air
(g) Air Reconnaissance/Surveillance
(h) Arty fire/Naval gunfire support
(i) Air space mgt/air traffic regulations
(j) Amphibious
(k) Anti-sub warfare
(l) Search and rescue

2 Each of the above tasks was evaluated by determining:

(a) Where the information (requests, intelligence, orders, assessment, etc.) appears geographically or organizationally in each Service command and control system.
(b) The immediacy of the commander's need for information ("short term," "long term").
(c) The veracity or reliability of information in terms of class (I greater than 0.95%; III about 0.65%).
(d) The degree of detail required (how far down—platoons).
(e) When information is required (demand timing, e.g., 3 per half hour).

b. Determine the compatibility requirements, that is, provide solution to the interface problem and information exchange requirements.

Further evaluation was made on the above functions/information to determine the flow of information among Service forces operating jointly. Diagrams were produced to show "boundary crossings" on the kinds of information flowing to and from the Service systems.

c. Analyze the tactical command and control system in joint operations from the data compiled in order to:

1. Determine which tasks are accomplished by more than one Service and establish need for compatibility among the Service systems.

2. Assess the approximate extent of functional commonality which Service tactical command and control systems would probably need to possess in order to provide effective support.

3. Recognize the operational requirements of the supporting tactical command and control systems, and, from guidance provided therein, ultimately to derive their technical design.

4. Isolate similarities and dissimilarities in terms, titles and procedures in order to determine areas suitable for standardization.

The conclusions which followed from the study were:

a. The operational, procedural, and functional goals of all the Service tactical command and control systems have a high degree of commonality.

b. The survey of the operational information flow across the "boundary line" between Service forces operating jointly and between such forces and the joint headquarters controlling them are first approximations of user requirements.

c. There is a requirement for compatibility among future tactical command and control systems. However, the requirement for inclusion of compatibility considerations within system design is to be implemented only in those systems having a predictable requirement for compatibility with other command and control systems.

d. Many terms and titles as related to joint tactical operations should be standardized.

e. There is a requirement for standardization in automatic data exchange of message formats, message transmission procedures, and message transmission characteristics.

f. It is highly desirable to standardize procedures employed by different Services in accomplishing the same tactical operational task in joint operations.

g. There is a need for a joint agency to be responsible for necessary compatibility, standardization and commonality of tactical command and control systems.

h. The World-Wide Tactical Command and Control Study can form the basis for development of technical standardization criteria to insure compatibility of future tactical command and control systems.

The following recommendations were generated by the conclusions.

a. That the World-Wide Tactical Command and Control Study be referred to joint groups for:

1. Development of standardized procedures, terms, and titles.

2. Coordination of qualitative operational requirements for tactical communication-electronic equipments when a problem of compatibility exists.

3. Development of standardized message formats, procedures, and transmission characteristics.

4. Development of technical standardization criteria.

b. That the Joint Chiefs of Staff establish improved procedures for ensuring inter-Service coordination in the development, acquisition and operation of tactical command and control systems.

As a result of the approval of the Study by the Joint Chiefs of Staff and the Secretary of Defense, the following occurred:

a. The World-Wide Tactical Command and Control Study was approved for use by the Services as a statement of basic military operational requirements.

b. The Chief, JCCRG, was appointed chairman of a Joint Tactical Command and Control Procedures Standardization Working Group composed of Service representatives to:

1. Establish and coordinate a standardization program for operational procedures, terms and titles.

2. Develop inter-Service coordination procedures for development, acquisition and operation of tactical command and control systems.

3. Review and, as appropriate, update the WWTCCS.

4. Review and make recommendations for the resolution of unresolved tactical command and control systems compatibility problems brought to the attention of the Joint Chiefs of Staff.

c. The remaining recommendations were approved and are being implemented within the Directorate for Communications-Electronics, J-6.

Continuing actions are now going on within the JCS, principle effort on the operational side by JCCRG and on the technical side by Director, Communications-Electronics Directorate, J-6. The Tactical Command and Control Procedures Standardization Working Group (TCCPSWG) from JCCRG and the Joint Standardization Group for Tactical Communications and Control Systems (JSG/TCCS) from J-5 have been and are working hand in glove coordinating throughout the process of arriving at standards.

Under JCCRG the TCCPSWG developed the Inter-Service Coordination Procedures which were promulgated to the Services. The TCCPSWG is establishing a number of Standardization Field Panels (SFP). These panels meet for a period of time in the field, each panel addressing a specific operational function area such as Close Air Support, Air Strike/Interdiction, Air Intercept, etc.

The panels are composed of officers, selected by their parent organizations because they are highly skilled in the operational tasks which the panels are to examine. They are selected from all the Services and certain unified and specified commands and have a working familiarity with their Service's doctrine and operational requirements. These officers have no other duty than to study and analyze the operational task involved and to assemble a document containing the most operationally useful, mutually concurred in, command and control procedures, terms, and titles that their combined experience in joint operations can produce. With this combination of professional competence so precisely focused on a task, the resulting conclusions (derived from mutual agreement) have the highest possible certainty of being acceptable as US standards.

Completed SFP reports are forwarded to the TCCPSWG for review, approval and processing through the Services and Joint Staff. Approved terms and titles will be incorporated into the Joint Dictionary by Director, Personnel Directorate, J-1, both in the main body of the dictionary and in glossaries (by functions) to be added to the dictionary. In addition a new JCS Pub,—(X) for the present, will be developed which will contain all the standardized procedures.

The standardized items are translated or transcribed by J-6 into technical terms and standards and placed in JCS Pub. 10.

SUMMARY

In summary, the problem in compatibility was recognized by the Secretary of Defense and studied by the JCS and the Services. The resulting study did not answer all the problems but directed attention to a means of solving the problem. Assigned Responsible Agencies were specified in selected fields of endeavor: JCCRG in requirements for operational compatibility, standardization field panels and resulting JCS publications; J-6 in the technical field—to recommend standardization and compatibility criteria for tactical communications and control systems to be incorporated into JCS Pubs. 10 and 11; and DDR&E to coordinate, review and make recommendations on Service proposed

R&D programs to insure required inter-Service operational compatibility among future tactical command and control systems, in accordance with criteria established by the JCS. Thus, the Joint Chiefs of Staff have developed programs which will provide the tactical commander with a compatible command and control system. One program will improve the mutual exchange of information by establishing world-wide standards for procedures, terms and titles. Another will develop criteria for the technical exchange of information between the various systems, both manual and automated. The third will insure Inter-Service coordination in the development, acquisition and operation of future tactical command and control systems.

The joint standardization group for tactical communications and control systems

by Marcus C. Jordan, *Lieutenant Colonel, USA*
Joint Chiefs of Staff
Washington, D. C.

INTRODUCTION

The Joint Chiefs of Staff have given increasing recognition to the vital importance of compatibility of tactical command, control, and communications systems and equipment during the past several years. Compatibility of equipment has always been recognized as a requirement for joint operations. The increasing interdependence of forces engaged in military operations and the continuing introduction of more technically advanced systems and equipment, however, necessitate greater high-level attention being given to ensure compatibility and the appropriate amount of commonality. The Joint Standardization Group for Tactical Communications and Control Systems (JSG/TCCS) is one activity within the Organization of the Joint Chiefs of Staff which has been given a significant role in ensuring compatibility and commonality of tactical command, control, and communications systems.

In April 1964 the Joint Chiefs of Staff were requested by the Secretary of Defense to "develop standardization and compatibility criteria that can be used in the selection of equipment for worldwide tactical communications and control systems of the military services."[1]

Already in existence at that time were numerous civilian and military activities engaged in standardization and compatibility in the C-E field. These included, among others, the Military Communications Systems Technical Standards Committee, the Military Communications-Electronics Board, and various groups under each of the Military Services, as well as international standardization activities. Each of these activities was interested in compatibility of tactical communications and control systems to some extent; however, there was an apparent need for the Joint Chiefs of Staff to establish criteria for standardization and compatibility. This need was made more urgent by the knowledge that compatibility problems would be tremendously increased as the newer and more sophisticated systems in development were perfected for introduction into the tactical forces.

The JSG/TCCS was created within the framework of the Joint Chiefs of Staff to recommend these standardization and compatibility criteria. This Group, which functions under the Director for Communications-Electronics of the Joint Staff, was provided membership from each of the military services as well as the joint staff and certain defense agencies. It was charged with recommending:

- Criteria to insure necessary compatibility among existing systems and between existing and future systems if required.
- Standards applicable to existing compatibility requirements and for the development of future systems.
- Equipment commonality consistent with individual operational requirements.
- Solutions to problems of interface between non-tactical and tactical communications and control systems when required.[2]

Excluded from consideration by the Group were operational requirements and considerations of mission and doctrine.

In establishing this Group the Joint Chiefs of Staff recognized that one tactical communications and control system to meet the needs of all Services, while it might be desirable, was probably unrealistic and unattainable and that the Services had invested heavily in systems now in use, that these systems will continue to be used for some time, and they will be phased-out over a considerable period beyond that.

Tactical Communications and Control System Standards

An obvious area in which jointly agreed standards were required was for the exchange of real-time digital data between automated air defense communications and control systems of the various services. JCS Publication 8[3] states that organizational arrangements for accomplishing air defense functions must provide compatible electronic coordination and control means, operationally connected, when air defense forces of the various services operate within a region. With this established requirement the first major activity of the JSG/TCCS was to develop JCS Publication 10, *Tactical Communications and Control Systems Standards.* This publication was approved by the Joint Chiefs of Staff and the Secretary of Defense and published in May 1966.

The Tactical Communications and Control Systems Standards (TACSTANS) contained in JCS Pub. 10:

- Are developed for systems and equipment applicable to functional areas in which the need for compatibility has been validated as essential by the Joint Chiefs of Staff.
- Are based upon the philosophy that the interface between tactical systems should exploit the maximum capability of sensors and processors to provide precise information interchange.
- Utilize system characteristics previously approved for Service use where these characteristics meet the joint requirements.
- Define interface standards in adequate detail to guide the design and/or procurement of systems and equipment so that inter-operability in the field will occur without modification or degradation.
- Include message format standards designed to support established doctrine and known requirements.[4]

The standards in JCS Pub. 10 are designed to be complementary to those in Mil. Std. 188.[5] The data elements and codes are being further processed for standardization under the DOD Data Elements and Data Codes Standardization Program. In the development of these standards, international standardization requirements, particularly those of NATO, which are the responsibility of the Electronic Data Transmission Working Party (ELDATRAWP), were also given consideration.

Chapter I of JCS Pub. 10 contains technical standards for air defense and aircraft control which will be used by all US tactical systems which have a requirement for inter-Service data exchange. The tactical digital information links (TADILs) for which standards are currently approved are shown in Figure 1. For each of these TADILs, characteristics have been standardized. For example:

TADIL A — NETTED DIGITAL DATA LINK UTILIZING PARALLEL TRANSMISSION FRAME CHARACTERISTICS AND JOINT STANDARD MESSAGE FORMATS AT 2400 BPS.
A — 1 — INTERCONNECTING NAVAL AND GROUND ENVIRONMENT SYSTEMS.
A — 2 — INTERCONNECTING NAVAL UNITS.

TADIL B — POINT TO POINT DIGITAL DATA LINK UTILIZING SERIAL TRANSMISSION FRAME CHARACTERISTICS AND JOINT STANDARD MESSAGE FORMATS AT 1200 BPS.
B — 1 — INTERCONNECTING SAM WEAPONS SYSTEMS AND GROUND ENVIRONMENT AIR DEFENSE SYSTEMS.
B — 2 — INTERCONNECTING GROUND ENVIRONMENT AIR DEFENSE SYSTEMS.

TADIL C — TIME DIVISION DIGITAL DATA LINK UTILIZING SERIAL MESSAGE FRAME CHARACTERISTICS AND JOINT STANDARD MESSAGE FORMATS AT 5000 BPS.
C — 1 — FOR CONTROL AND RECTORING OF INTERCEPTOR AIRCRAFT.
C — 2 — FOR AIR TRAFFIC AND LANDING CONTROL.
C — 3 — FOR SURFACE TARGET RECTOR CONTROL.

Figure 1—Tactical digital information links (TADILS) from JCS pub. 10

- modulation and signal,
- frequency setting, accuracy and stability,
- timing,
- control code structure,
- procedural signals, and
- communications channel.

Chapter II contains messages approved for joint usage, together with their characteristics and a statement concerning which messages are used on which TADILs. There are 16 messages for use between control stations and between control stations and surface to air (SAM) weapons systems on TADILs A and B, and 31 control and 16 reply messages for use between control stations and aircraft or airborne weapons system on TADIL C. For example, one standard message is designed to report the position, identity and track number of an air track on a coarse scale. Included in the publication also are a number of messages used by the individual services in performing their air defense missions but which have not as yet been agreed upon for joint service usage.

Work is continuing within the Joint Standardization Group for Tactical Communications and Control Systems on developing additional standards for application both to air defense and other tactical functions to be performed by automated techniques. Certain of the messages now reserved for use by individual services are also being processed for joint approval. Standards for a low-speed TADIL (TADIL-D) have been developed and are being processed for approval of the Joint Chiefs of Staff.

The results of the Worldwide Tactical Command and Control Study conducted by the Joint Command and Control Requirements Group of the Organization of the Joint Chiefs of Staff and the follow-on effort to standardize procedures, formats, terms, and titles for use in joint operations will provide the basis for the continuing development of JCS Pub. 10. With the operational requirement for compatibility as well as the amount and kind of information that requires interchange established, the /JSG/TCCS will be able to effectively develop the standardized rtansmission characteristics and message formats and transmission procedures to insure compatibility of automated systems where compatibility is required. The further development of these standards is obviously a major under-taking and will represent a significant part of the activities of the /JSG/TCCS for several years even after all required input information has been developed, approved, and received by the Group.

Tactical Communications Planning Guide

The other major activity of the JSG/TCCS is continuing development and maintenance of JCS Publication 11, *Tactical Communications Planning Guide*. This guide is a direct outgrowth of a comprehensive study of tactical communications made by the Joint Chiefs of Staff in 1965 at the request of the Deputy Secretary of Defense. The main objectives of the study were to:

- Ascertain and recommend means of increasing commonality and compatibility of communications equipment used by tactical forces.
- Recommend means of decreasing variety and quantity of tactical communications equipment.
- Recommend means of limiting crowding of the radio frequency spectrum.
- In each of the above to give due consideration to dollar and manpower costs involved.[6]

The study was conducted over a six-month period by a study group of approximately 20 personnel from the Services, the joint staff and defense agencies under the chairmanship of Major General W. T. Smith, Deputy Director, Communications-Electronics, Organization of the Joint Chiefs of Staff. A part of the report of the study group addressed policy and procedural matters affecting commonality and compatibility. It included discussions, conclusions and recommendations covering

- communications doctrine,
- planning and programming,
- operational requirements,
- defense standardization program,
- procurement,
- radio frequency spectrum, and
- communications security.

A second part of the report of the study group was entitled *Tactical Communications Planning Guide*. This part was designed to be the first edition of a document that would be maintained and improved upon by the Joint Chiefs of Staff in the future. The purpose of this publication is to provide guidance to the Services in tactical communications equipment planning and to assist the Office of the Secretary of Defense in reviewing tactical communications matters. The concept of this guide was aproved by the Joint Chiefs of Staff and the Secretary of Defense, who, in January 1966 approved the Tactical Communications Planning Guide as a general guide for use in planning future tactical communications and requested that it be updated for use in appropriate 1966 program reviews.[7] The updated edition was approved by the Joint Chiefs of Staff for promulgation as a JCS publication and forwarded to the Secretary of Defense in April 1966.

The guide contains four major chapters covering Categories of Tactical Communications and Compatibility Requirements, Current Equipment Compatibility, Operational Requirements, and Technological Compatibility Objectives. It should be emphasized that this document is concerned at this time solely with tactical communications and not with any of the multitude of other aspects of tactical command and control. Even within the area of tactical communications, coverage is selective in the chapters dealing with operational requirements and current equipment.

Operations conducted by joint forces involve various combinations of closely interrelated operational tasks and numerous functional applications of communications necessary for task accomplishment. Compatible communications equipment must be made available wherever intercommunication is required. The requirement for compatibility can be stated in at least three ways: in terms of the operational task, in terms of the communications function, and in terms of the technical characteristics of the equipment. A technical categorization was considered to be most appropriate for the guide; accordingly, Chapter I identifies a number of technical categories within which compatibility is required.

In the Worldwide Tactical Command and Control Study, developed by the Joint Command and Control Requirements Group, 12 broad operational tasks are identified and requirements for intercommunications between units engaged in performing these operational tasks specified. Based upon this information, the technical categories of communications compatibility, and a similar categorization of the applications which communications perform, a series of matriees were developed which describe:

- Who must intercommunicate with whom in performing each broad operational task.
- Why they must intercommunicate (i.e., for what communications functional application).
- How the intercommunication can take place (which of the technical categories of communications compatibility).
- When intercommunication is required (in support of which broad operational task).

Figure 2 is an example of one entry from one of these matrices.

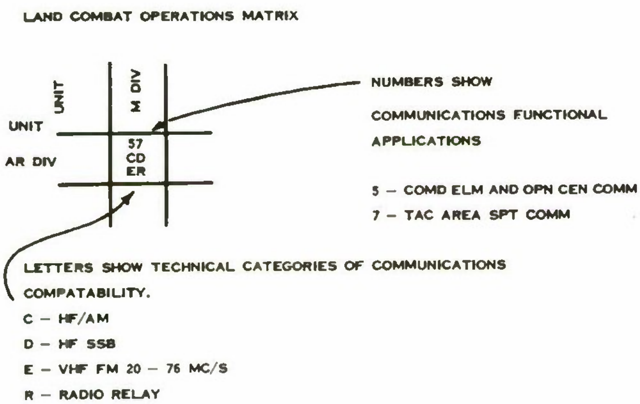


Figure 2—Matrix example from JCS pub. 11

In Chapter II a total of over 700 major items of tactical communications equipment either in the active inventory or in development with operational use indicated in the near future are categorized by technical categories of communication compatibility. Each of the more than 100 Current Equipment Compatibility Sheets lists the compatible equipment in a category and gives an indication of the extent of commonality. Figure 3 is an example of one of these sheets.

In Chapter III the operational requirements documented by the Services in Qualitative Material Requirements, Specific Operational Requirements, etc., are categorized in a manner similar to that employed for current equipment, and analyzed to determine the extent of joint interest in each requirement. About 100 requirements were documented by the Services. 10

EXTRACT CURRENT EQUIPMENT COMPATIBILITY SHEET NO. D14
HF SSB RADIO TRANSCEIVER, VEHICULAR

NOMENCLATURE	FREQUENCY	MODULATION	RECOMMENDED REPLACEMENT/ STANDARD
AN/GRC–106	2–30 MC/S	A1, A3, A3A, A9c	STD
AN/MRC–83	2–30	A1, A3A, F1	STD
AN/TRC–75			STD
AN/MRC–95	2–30	A1, A3A	SN/VSC–2
AN/PRC–47	2–12	A2, A3A	NOTE 1
NEW DEVELOPMENT:			
AN/VSC–2	2–30	A1, A3, A3A, A9c, F1	PLD STD

NOTE 1: AN/PRC–47 IS A 100W STD FOR U. S. MARINE CORPS.
AN/PRC–62 WILL BE A 20W REPLACEMENT FOR U. S. ARMY.

Figure 3—Example of current equipment compatability sheet from JCS pub. 11

per cent of these were considered to be only of interest to a single service; the remaining 90 per cent were considered as joint requirements (a total of 34) based on expressions of interest by one or more services in a requirement doeumented by another service. For each of these joint requirements a cover sheet incorporates an analysis of the degree of joint interest in the requirement and a summary of future action to be taken on the requirement. Because of the rather lengthy process required for complete inter-service coordination of requirements, the joint requirements in the present version of the guide are listed as "interim"; however, action is under way to complete the coordination and remove the interim designation.

Chapter IV contains technological objectives which are considered important to achieving optimum compatibility of tactical communications in the future. These objectives are of two types—general, which are applicable to more than one of the technical categories, and specific, which are applicable to only one category.

Much development work remains to be done in each of the areas addressed in JCS Pub. 11. In addition, continuing maintenance will be required as technology and requirements change. These functions of development and continuing maintenance will be accomplished through the mechanism provided by the JSG/TCCS.

Reorganization of Joint Standardization Group for Tactical Communications and Control Systems

The Joint Chiefs of Staff have recently approved a reorganization of JSG/TCCS. The purpose of this reorganization is to provide for continuing development and maintenance of JCS Pubs. 10 and 11 and to permit the Group to more readily recommend equipment compatability criteria and technical and procedural standards for tactical command, control, and communications systems in support of joint operations. The permanent organization and membership of this Group is now as indicated in Figure 4.

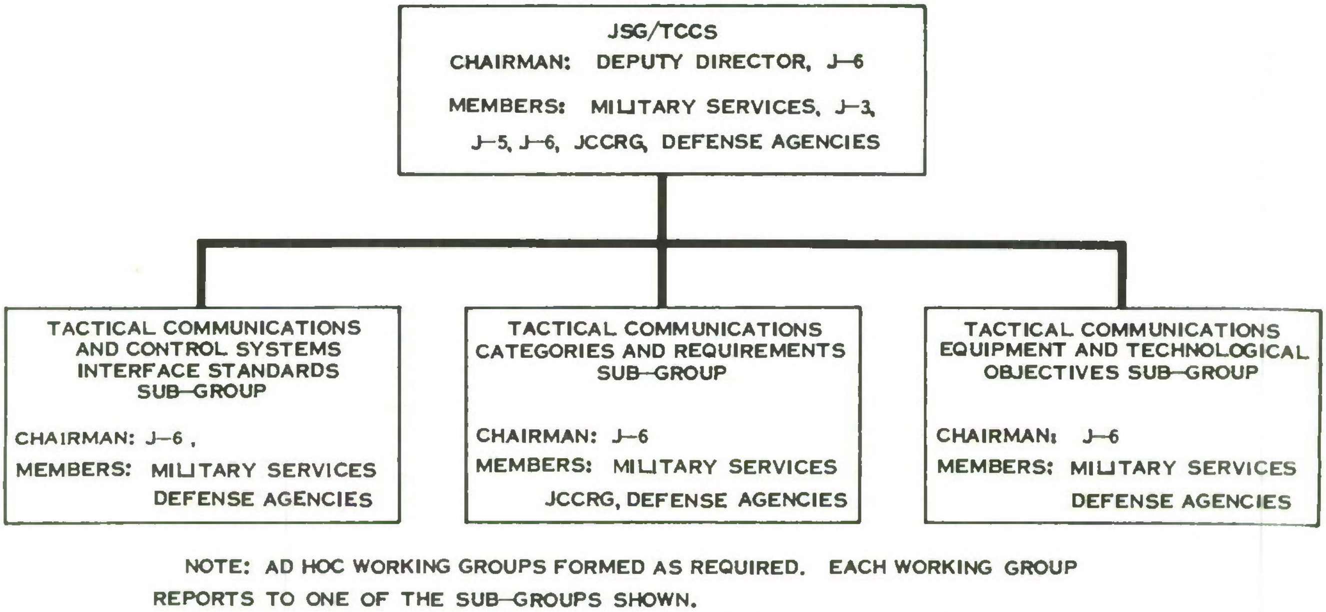


Figure 4—Permanent organization of joint standardization group for tactical communications and control systems

In addition to the permanent organization shown a number of working groups are actively engaged in working on individual actions.

In general terms, the objective of Tactical Communications and Control Systems Interface Standards Sub-Group is to define standardization and compatibility criteria for information exchange between tactical communications and control systems of the military services. The work of this sub-group will be largely devoted to continuing the development of JCS Pub. 10, *Tactical Communications and Control Systems Standards*, and to related international standardization efforts. The Tactical Communications Categories and Requirements Sub-Group and the Tactical Communications Equipment and Technological Compatibility Objectives Sub-Group both have basic objectives related to the further development and maintenance of JCS Pub. 11, *Tactical Communications Planning Guide*. Over-all guidance and coordination is provided by the parent group.

REFERENCES

1 Secretary of Defense Memorandum for the Chairman, Joint Chiefs of Staff *Worldwide tactical communications and control* (Secret) 18 April 1964

2 Joint Chiefs of Staff *Worldwide tactical communications and control* (Secret) 8 May 1964

3 Joint Chiefs of Staff Publication 8 *Doctrine for air defense from overseas land areas* May 1964

4 JCS Publication 10 *Tactical communications and control systems standards* May 1966

5 Department of Defense *Military communication system technical standards* MIL-STD-188B 24 February 1964

6 Deputy Secretary of Defense Memorandum for the Chairman, Joint Chiefs of Staff *Tactical communications* 8 December 1964 and Deputy Secretary of Defense Memorandum for the Chairman, Joint Chiefs of Staff *Tactical communications study group* 22 January 1966

7 Secretary of Defense Memorandum for the Chairman, Joint Chiefs of Staff *Worldwide tactical command, communications, and control* (Confidential) 7 January 1966

IMPACT OF AUTOMATED INFORMATION SYSTEMS UPON ORGANIZATION AND MISSIONS

JOHN B. BESTIC, *Major General, USAF, Chairman*

WILLIAM E. KUNTZ, *Captain, USN*
BERT S. HARRIS, *Colonel, USAF*
BENJAMIN ERDMAN
ROBERT E. HARSHBARGER
MARVIN L. MANHEIM

Information in the military operational environment

by WILLIAM E. KUNTZ, *Captain, USN*
Defense Communications Agency
Washington, D. C.

During the past two and a half decades improved capabilities in electrical communications and in automated data processing have received extensive application in support of military operations.

At the risk of providing to the information system scientist merely another glimpse of the obvious, this paper is intended to review some of the facets and consequences of applying current technology in these fields to the solution of long standing military operational problems.

Information has always been, and continues to be, the primary tool of military command. Information serves as the basis for the commander's decision-making, and as the means for conveying his intentions to his command and to others.

Consequently, the acquisition and dissemination of information have preoccupied military commanders as long as they have existed.

Information, in the context of the military operational command environment, is selected, evaluated data, relevant to a specific situation in point of timeliness and content. Data are not information until these data are perceived by the commander to be valid, and of value. Furthermore, data cannot be transformed into information until the problem to be addressed has been defined and classified. Until this happens, no one can know what are the "pertinent facts"; one can only know data. Definition and classification determine which data are to be considered relevant. Analysis and perception determine which data are to be considered of value. In the military operational environment, the means or method of transferring these data is the classical "communications system."

With information—that is, relevant valid data—essential to military command, this component subsystem for the transfer of data, plus the mechanical and electrical methods available to select, correlate and display useful data for analysis, is essentially the military's equivalent to the scientist's "information system."

An effective military communications/information system must keep not only the commander but all levels of a command adequately informed of changes in the military situations which affect them. To accomplish this, personnel of the system's organization, both data handlers and data originators, must know what data to collect and to process, and the data users must be able to delineate their requirements for information.

The effectiveness of a communications/information system in supporting a command depends not upon the amount of data flowing in the system, but upon the amount of relevant data the commander receives and the degree to which the system can transfer that relevant data to potential users.

Historically, military communications have been virtually synonomous with the exercise of authority. Further, from the commander's point of view, his organization is primarily a communications network.

One can conjecture that primitive military operations were successfully conducted by a commander who relied solely upon his own sensory inputs, recalled relevant experiences from memory, performed mentally the comparative analyses required under the circumstances, and communicated his intentions or his will directly to subordinates (and to adversaries) by invoking appropriate action.

Proxy observations and the necessity to extend the scope of the primitive military commander's influence dictated a need for assistance in achieving the desired flow of information, and the role of the military communicator was born.

As the complexity of the military commander's operations further increased, two types of delegation occurred: staff subordinates, to assist the commander in performing his functions of data correlation and analysis, and line subordinates (*i.e.*, subordinate commanders) to assist the commander in the communication of his intentions.

To provide for mutual understanding, to standardize behavior, and to minimize information exchange requirements, a third delegation of a sort occurred in the emergence of predetermined and prepositioned policies, plans, doctrines, and procedures, to govern the performance of both the staff and the line hierarchies, in the absence of direct communications with the commander.

As military weapon ranges and lethalities increased, as time available for response to changing situations seemed to dwindle, as mobility and methods of communicating improved, as political, economic, and social implications of military action became more obvious and critical, significant improvements in mechanical and electrical data handling capabilities were sought by the military commander to assist him and his staff and line subordinates in coping with the increasing flow of data and the increased need for more information associated with more complex military operations.

The communications/information systems which have evolved in support of the command structure in military organizations differ among each other in technical configuration and reflect subtle differences in military roles and missions, in the attitudes of the military commanders served, and in the manner in which the command echelon is structured.

These differences fall generally into one of three categories or types, probably best described by analogy. One type of system might be characterized by a "public utility" concept. Responsibility for the service and support thus provided to the command is shared between the supporting organization and the using command itself. The line of demarcation of this responsibility falls roughly at the point data enter into and leave this "public utility" type system.

Its advantages are obvious. The "public utility" aspect of its operation is governed by relatively concise technical principles and its management and operators can be, and are usually, technically oriented. Its behavior can be carefully structured and its technical performance predicted in the presence of appropriate standards. It is obviously economical of resources.

The major disadvantages of this first type of system are also obvious. The "public utility" aspect of the system and its personnel rely upon advance planning, policy, doctrine and procedures; there is little, if any, flexibility, obligation or capability to react promptly to the dynamics of the using command's changing requirements, Moreover, responsibility for "writer to reader" delivery of data is vested in more than a single organization.

A second configuration employs the same foregoing "public utility" concept for trunking and interswitch networks, but with a significant organizational difference. An individual, usually a direct representative of the commander, is designated the command communicator and assigned the primary responsibility for "writer to reader" delivery of data. He is technically competent to interface with the "public utility" system into which the command has access. In addition, he controls both the terminal portion of the system serving the commander and the staff, and may pre-empt a predetermined portion of the "public utility" network resources. He maintains the close liaison with the commander and staff necessary to observe and modify system performance to match the dynamics of the command situation, and when possible, to foresee and act upon changing conditions before they occur.

Advantages include improved "writer to reader" delivery time. Studies have shown that, other conditions of the communications/information system being fixed, the greatest potential improvement in message handling times can be achieved in the terminal portions of the communications/information system. Therefore, with the manager of these data handlers reporting directly to the commander, the performance of the terminal portion of such systems appears to be the critical feature which determines the advantage of this configuration.

The second configuration also has the same disadvantage as the first configuration, if to a lesser degree, of inflexibility to respond to user's changing requirements. However, the role of the commander's direct representative has proven to be highly useful in obtaining exceptions to doctrinal constraints to improve overall system responsiveness.

The third configuration has essentially "private line" characteristics. This is the familiar dedicated communications concept. Its advantages are obvious. It provides the ultimate in foreshortened "writer to reader" time. It can be engineered to be available when needed. It seldom saturates.

The disadvantages of this type of configuration are also well known. It is more prodigal of resources than either of the other two types and is usually less survivable in comparison with integrated switched networks. More importantly, however, from the viewpoint of both line and staff command hierarchies, this configuration has the inherent characteristic of comparting data and information to an inacceptable degree, and may bypass subordinate echelons of the command.

At the outset, it was mentioned that these several configurations also reflect the viewpoints of the commands served. Without belaboring this aspect of military communications/information systems, it may be observed that those highly structured military organizations which operate with well established policies and doctrines find the "public utility" concept acceptable. Those organizations whose missions include a capability to react in extremely short times favor the "private line" concept.

There appears also to be a direct relationship between the "depth of control," the length of the chain of command, and the type of communications/information system configuration employed. In shallower organizational hierarchies, dedicated systems appear more frequently.

Further, command organizations that traditionally are highly structured with detailed doctrine appear to be completely distrustful of the reliability of data flow in any actual operational situation.

Conversely, less systematic organizations seem to place a high degree of faith in system reliability for the conduct of military operations in the face of the enemy.

The term "automated information system" has been used in connection with specific military applications of a wide range of capabilities. Consequently, there have arisen major semantic difficulties reflecting the differences in viewpoints of the sponsoring agency, the system designer, the system implementer, and the operational user.

Some military automated information systems, intended to support an operational military commander and his staff, have been labeled "command and control systems." One definition of this aggregate of capabilities states: "A command and control system is . . . a composite of equipment, skills, and techniques which, while not an instrument of combat, is capable of performing the clerly defined function of *enabling* a commander to exercise continuous control of his forces and weapons in all situations by providing him with:

- the information needed to make operational decisions, and
- the means for disseminating these decisions.

A complete system includes all subsystems, related facilities, equipment, material, services, and personnel required for operation of the system, so that it can be considered a self-sufficient unit in its intended environment."

This all-inclusive view of a command and control system would properly encompass, for example, an automated weapons control subsystem, although its contribution to the information needs or decision-making capability of the commander, or its demands upon his decision-making capability, may be relatively small. Operating as essentially closed loops, from target to weapon back to target, most automated weapon control systems are characterized by integral sensing, selection of an alternative in conformance with pre-established doctrine or by online, subordinate level decision, and automatic commands to the weapon to destroy a specific target.

Such weapon control systems normally require only a simple decision (input) by the commander to initiate or to terminate their use. Their data outputs to the commander are generally restricted to periodically reporting their status, targets engaged, resources remaining.

For purposes of this discussion, automated military command and control systems will be considered those intended to satisfy requirements of the commander and his staff for mechanical and electrical assistance in carrying out the following military functions:

- Sensing significant perturbations in the situation or environment.
- Storage, retrieval, transmission of data.
- Manipulation and analysis of data.
- Development of alternatives.
- Dissemination of decisions.

Several significant presumptions are inherent in so describing automated military command and control systems:

- It presumes a universal understanding, and acceptance, of the content and context of the command function.
- It presupposes an equally universal understanding of command's line hierarchy.
- It assumes a differentiation between data and information.

Technically, its design includes both a communications capability and a data processing capability.

Radical changes in scope and methodology are taking place in modern business management. Automated data are modifying the content of managerial jobs at all levels, the roles of middle- and lower-level managers, and the entire structure of decision-making within the enterprise.

To the extent that military operational command may be similar to business management, equally radical changes might be anticipated in military command methods and organizations. Improved capabilities of communications and awareness of data processing applications in business have prompted military commanders to review their situations and seek to obtain an automated command and control system. These reviews also have fairly consistently seemed to indicate the desirability of centralizing the data—a consequence of a "total system" approach, popular since about 1960.

The total system implies that *all* data are collected, transmitted, processed, and distributed to potential users.

In highly structured situations, in which virtually all contingencies are foreseen, in which preplanning is thorough and the resulting plans and doctrine are established, the total system approach can be approximated.

The thermostatic control is an example of such a total system. Temperature and humidity ranges can be maintained within pre-established limits; even a record of environmental changes and system responses could be obtained very simply, from which performance analyses and logistic support requirements could be derived.

However, a review of military organizations involving extensive communications and data manipulation reveals that the total system approach must be used with caution when one considers the nature of military activities.

The total system concept presupposes acquisition of data direct from source by means of communications capability integral to the automated information system. Data so acquired may or may not be available to intermediate echelons of the military organization.

Assuming the technically optimum case, wherein data from a source are simultaneously available to all command echelons, unilateral decisions by higher headquarters based on these data will deprive the higher command of the judgment of intermediate echelons.

In the case wherein the data sources differ for individual commands or for functional subsystems serving a single command, problems of ambiguity arise.

In the event that the upward flow of data follows the command hierarchy and is forwarded only after evaluation in the context of the forwarding headquarters, the ensuing delays are usually considered unacceptable by the sponsoring agency, the system's designer, and the system's implementers. The operational users' viewpoint is usually ambiguous.

What seems to be neglected is that, in the military, the command hierarchy is and remains a multilevel pyramidal arrangement of headquarters, each differing from the others. On the same echelon, the difference between commands lies in specialization and functional cognizance. Between echelons of command, the differences in contexts stem from perception and scope of interest. This hierarchy is a result of pragmatic empiricism, in the presence of constraints of communications and the limits of an individual's ability to cope with issues in the time available. The constraints of communications and individual perception which are, in part, responsible for the hierarchy also contribute to the demand for filtered, evaluated data (*i.e.*, information) between successive levels of command.

At higher echelons of military command, there is relatively less problem-solving, in which the situation is postulated, requirements are evident, and the decision, tactical. In this context, problem-solving implies a single optimum solution. The situation is given, and requirements are evident. The optimum solution is the most economical adaptation of available resources.

Rather, higher command echelons are faced with the necessity of determining the situation, even of changing the situation; of determining resources, either what they are, or should be; of developing alternative courses of action or options, which are seldom black or white and may include doing nothing. Decisions are strategic in nature. Further, there is a continuing requirement to identify missing information, which in turn generates a requirement for judgment, and for determining the degree of precision allowable or acceptable in the decision.

At higher command echelons it is a rare situation indeed in which all needed information is available and there is one optimum solution. In fact, wherever analysis of the situation leads to this comforting conclusion, one may reasonably suspect the solution of being little more than a plausible argument for a preconceived idea.

Intermediate and lower echelons of command can contribute directly to definition and analysis of the higher headquarters problem, to the development of alternatives, and to the conversion of any decision to effective action; and can make these contributions with due consideration of the risks, economy of effort, timing and limitation of resources as understood by these subordinate headquarters.

It is not apparent to me that any automated command and control system designed to date has adequately taken into account the characteristics of the military environment described above.

For example, one impact of automated information upon the military command hierarchies might be illustrated from considering the consequence of the introduction of radar in the U. S. Navy early in World War II. Initially viewed as a substantial improvement in the sensory range of the naval lookout, its potential contribution as a source of data and information to the command echelon was not recognized by the designers nor the military at the outset.

Because radar provided not only detection, but target location, its use was promptly pre-empted for target location, its use was promptly pre-empted for dedicated application to weapons control problems, often to the detriment of its improved "lookout" or initial detection function.

Later, when radar's potential for providing data for a multiplicity of uses was recognized, two major changes, one technical and one organizational, occurred:

- Technically, the basic concept for presentation of data had to be changed from the A-scope to the plan position indicator, the PPI.
- Organizationally, a new group of operator-technicians emerged at the staff level further to process the data now available in light of on-going operations, not only of their own command but of adjacent commands as well.

One might have hoped that the trend taken in establishing this new group would have enjoyed universal acceptance of the obvious benefits it could demonstrate.

Actually, the notion that data should be available to several potential users as a basis for each user's contribution to overall command analysis and decision was far from universally accepted by functionally oriented staff elements. On the contrary, today, 25 years after the introduction of radar one finds military organizational arrangements still geared to unilateral use of dedicated sensing systems, to the exclusion of other potential users in the command who may either duplicate the data collection and processing capability, or do without it.

Sonar is another example of a space sensing system viewed primarily as a dedicated data input for ASW weapons control, rather than as a potential source of data for rapid collation with other undersea space sensing systems and intelligence sources, in both strategic and tactical applications.

A more recent, and probably more widely recognized example of the impact of an automated military command and control system on the military command hierarchy is SAGE.

Disregarding service orientated opinions as to the merits of its technical design, SAGE has had a profound impact upon military organization and mission. It provided a massive detection, data collection, data processing, and data communications capability which made technically feasible a highly centralized command organization. In turn, this new command undertook the detailed management, on a near real time basis, of the air defense mission for the entire North American Continent—a span of control of military operations never previously envisioned.

Current policies for acquisition of automated command and control systems to support military command seem to stipulate:

- Evolutionary improvement of capabilities, exploiting technological advances as they occur and operating experience gained in copying with changing situations.
- Flexibility in system performance to anticipate possible future changes in the operational environment.
- Compatibility with associated automated information systems of coequal, subordinate or superior echelons in the command hierarchy.

Implementation of these policies can lead to the procurement of off-the-shelf ADP equipment and software, the latter subsequently modified as necessary in an effort to tailor the off-the-shelf ADP capabilities to evolving requirements.

One obvious consequence of the introduction of this type of automation into a military environment is its impact upon organization: An entirely new organizational entity is required (but seldom acquired) to provide for data management and computer program implementation and maintenance. The National Military Command System Support Center, operated by the Defense Communications Agency in support of the National Military Command System, and the Naval Command System Support Activity operated by the Navy, are two examples of these relatively new organizational adjuncts to command.

Understandably, such supporting units seek to achieve an in-house capability to convert information requirements of command into usable computer programs to produce useful information.

However, the ADP operator-technician-programmer has had no significant decision-making experience upon which to base his interpretation of user demands. Moreover, the user group is inherently distrustful of any data system with which they, as individuals, are unlikely to have had any extensive personal experience.

One hears criticism of the technical ADP staff as lacking "operational experience." This criticism should be interpreted more precisely as this ADP staff lacks "decision-making experience" *at the command level being served.* Issues of risks, stakes, and alternatives are implicit in this view.

On the other hand, senior decision-makers at the command level are equally unlikely to be conversant with capabilities and limitations of ADP, either now or in the immediate future.

Further, provision of mass data systems on an incremental basis, using hardware and software designed primarily for industrial or business applications, appears to presuppose similarity between the techniques of business management and of military operational command.

As pointed out previously, there are marked differences in the operational contexts at successive levels in the classical military command hierarchy. Decision can range from problem-solving solutions at the lower echelons, to determination and election of alternative courses of action at the higher levels. Techniques to support this spectrum of activity will seldom be similar throughout, and data requirements will not be similar at each level of command.

Nevertheless, automated command and control systems using off-the-shelf ADP are being introduced to support higher headquarters. These systems generate requirements for large amounts of data, stored at a central data repository, and efficiency of processing data demands that these data be acquired in standardized formats.

But differences in information requirements at intervening headquarters—not differences in ADP equipment and programs—inevitably demand compromises in reporting formats and in data content, or, alternatively, parallel reporting systems, or independent levies for information upon subordinate or collateral commands.

Next, an effort is made to obtain "all" data for storage at the central higher echelon repository, just in case these data may be needed. The rapid processing of vast amounts of data leads higher echelons of command to believe they possess adequate information upon which more detailed decisions can be based. The command organization at the higher echelons then tries to participate actively not only in strategic decision but tactical problem-solving as well.

To the extent that the higher command actually makes more tactical decisions, there is a diminution of the role of intermediate headquarters, its experience in making decisions is reduced, and, in the event portions of the centralized data acquisition, processing, or dissemination functions fail, catastrophic loss of overall military capability, rather than a more graceful degration of performance, can well result.

Historically, military plans, doctrine and procedures evolved to minimize the otherwise mandatory reliance upon communications, to ensure standard patterns of behavior and timely response by subordinates in given situations, and to provide a residual, though reduced, capability in the event of partial loss of the command hierarchy.

Improved communications and data processing capabilities seem to have considerably lessened the need for highly structured doctrine; in fact, as long as it survives, the capability to obtain more and more data rapidly to determine a situation, rather than to postulate it, further reduces the traditional role doctrine plays in military operations, or substitutes for it a more complex and more rapidly changing modus operandi.

The concomitant ability to communicate tactical decisions directly to the action unit in turn generates the necessity for a higher degree of flexibility of response by the action unit, which in turn generates a requirement for rapid status and response reporting back to the command level initiating the tactical decision, which in turn generates a requirement for more communications and more data processing, ad infinitum.

SUMMARY

To summarize, it is my opinion that military organization is a result of pragmatic development. The principal tool of command is information. The commander has resources, such as weapons, personnel, and vehicles at his disposal, but he cannot manipulate these resources effectively without this tool.

The command echelons have been structured, with staff, line, and doctrinal delegates, to obtain data, to manipulate data to derive information and arrive at alternative courses of action, and to communicate decisions, within the constraints of available or nonavailable communications.

Increased capabilities in communications and in data processing have vastly improved the ability of commanders to acquire data, to have it manipulated, and to transmit derived decisions.

The automated information systems now technically feasible cover a wide spectrum of capabilities, ranging from individual sensor systems generating data for one or a variety of users, through weapons control systems of varying degrees of automatic response, to mass data systems collecting and collating large amounts of detailed data. However, this spectrum of automated information systems has widely divergent characteristics in terms of military utility.

The sensor system's utility, with its design based on relatively simple concepts and known doctrine, is limited only by engineering ingenuity and ability to control the physical environment. At the other end of the spectrum of application, mass data systems have vast technical capability but virtually no doctrinal basis for design and employment, other than conventional business management relationships.

Automated information systems, other than command and control systems, employed in the role of sensor systems to generate data, may have either a relatively little or a major impact upon military organization and mission. It is difficult for either the designer or the user to anticipate the magnitude of this impact.

Higher order automated information systems, such as weapons control systems, even though operational concepts stem from established doctrine or can be stated concisely and any required communications are integral, nearly always have major impacts on both military organization and military mission at the middle echelon level. The designer and user jointly can usually predict the magnitude of this impact.

Automated information systems intended to support higher echelons of command always have a profound, continuing, predictable, but seldom recognized impact upon both military organization and military missions.

New organizations with the mission of ADP management and program maintenance are required within the command. The tactical decision-making capability apparently provided to higher headquarters by these systems tends to obviate the necessity for making decisions, and thus the learning experience, at intermediate echelons in the operational chain of command.

Finally, although the primary purpose of information systems at higher headquarters is to assist in the strategic decision-making process—that is, determination of real-world situations and development of alternatives—automated data processing capabilities continue to be woefully deficient in identifying missing information, exercising judgment, or determining the degree of precision acceptable, in the light of risks and stakes involved and the resources available.

The impact of information systems on centralized vs decentralized command and control functions*

by BERT S. HARRIS, *Colonel, USAF*

and

BENJAMIN ERDMAN
Defense Communications Agency
Washington, D.C.

INTRODUCTION

Te purpose of this paper is to stimulate thinking on the relationship between the information handling functions associated with command and control and changing trends in military management. The intent is neither to discuss the functions of command and control nor to compare the values of centralized and decentralized management and information systems. The paper attempts to illustrate how the character of organizations and related management functions are influenced by information processing technology. Thus it discusses how management practices may change as better information handling capabilities are provided, and how, in turn, these changes further influence the character of the related information handling functions.

In the interest of brevity and in an effort to confine the scope of the paper, many military responsibilities and functions are described only in the context of this paper. Accordingly, these descriptions are not wholly complete and may be erroneous if applied in a different context.

Part one

Present-day military planning is characterized by a broad spectrum of military conflict, widely scattered areas of operations, the involvement of many agencies, and the concept of rapid mobility. For example, the European and Pacific threatres of operation each have large forces and resources continuously assigned to the area. They can be augmented as required with forces and resources normally positioned in the United States. On the other hand, there are potential conflict areas where little or no forces and resources are in place, and for which preplanned packages must be organized and deployed as the need arises.

The concept of mobility packages, and the general capability available for moving large force packages, permits a rapid response to many potential levels and areas of conflict without indefinitely committing forces to each area. While this permits worldwide coverage with less "inventory," it greatly increases the requirement for centralized planning and coordination.

Thus the problem becomes one of competing require ents for finite resources. No one CINC* or Service or Single Manager can unilaterally make the decisions while significantly affect several other Defense organizations. Questions, such as how much of the MAC† fleet should be held in reserve, whether to mobilize CRAF†† aircraft, what the impact would be of borrowing weapons from one theatre to use in another, or the price to be paid for shortening closure times, can often only be answered at the high, common level of the National Command Authority (NCA). It is now recognized that the NCA requires the capability for rapidly assessing feasible alternatives. This obviously has profound implications with respect to information processing and organizational responsibilities.

There are clearly many functions which a CINC, Military Department, Single Manager, or Component

*The thoughts expressed in this paper are those of the authors, and do not represent the position of the Department of Defense or any Agency thereof.

*Commander-in-Chief.

†Military Airlift Command (formerly Military Air Transport Service-MATS).

††Civil Reserve Air Fleet.

can do best. However, resource planning on a worldwide basis, particularly when reaction time is at a premium, must often be done at a higher command level. In this respect, it is important to note that most of the details with which each organization is normally concerned will still be addressed by that organization. It is the policy making, establishing of priorities, and the rapid determination of the role each agency will play in an unplanned environment that must be the function of the highest level.

At a minimum, the information processing capability supporting the NCA level must be able to aggregate, correlate, ana'yze, and evaluate that data and information provided by all of the potentially affected agencies. The question of level of detail of unprocessed data to be forwarded to the NCA is of critical importance, since the end result of the information processing at the NCA level is intended to be the assignment of responsibilities to each of the major organizations involved.

It is not at all clear, however, that this will be where NCA involvement will stop. Given the data processing and communications to acquire and process vast quantities of data, the NCA can, if they wish, effectively monitor and exercise considerable control over the manner in which each organization carries out many of its responsibilities. In the past, experience has shown that the best balance is achieved by retaining only centralized policy making, overall broad planning, and monitoring of operations at the NCA level, and by decentralized control of operations at lower levels. Each supporting organization must therefore have its own information processing capability for detailed planning, and must also contribute to the NCA broad planning role. This means, for example, that the NCA can determine the feasibility and capability of MAC support of a mission, or to assess the impact of utilization of aircraft on closure time, but leaves to MAC the problem of determining and controlling flow schedules and cycling of aircraft in coordination with other affected agencies.

This concept is based on the premise that the single most important problem in both long-range planning and crisis planning is the coordination of all affected agencies. Under noncrisis conditions, the NCA could afford the time required for numerous studies to be made by competent organizations, each feeding information to the other, with many interactions before a final result is attained. Under crisis conditions, or when a multitude of plans must be feasibility-tested in a short time, the classic approach to planning may need to be drastically curtailed.

The most valuable benefit of centralized planning is that a great many alternatives can be considered, *i.e.*, a thorough sensitivity analysis can be performed, in less time than was often required to examine one untested course of action. As noted previously, however, there is an attendant danger. Since communications and data processing technology impose no practical limitation on how much raw data can be acquired and stored, it is possible for the NCA to have in the centralized data base all of the data held by all other organizations. (Often the highly detailed raw data is required to insure that the aggregations of the data is accurate.) But this level of detail also permits the NCA to "second guess" the judgment and actions of operational organizations. Movement schedules can be developed as easily at the NCA level as at any other level. The problem arises therefore in the manner in which parameters are applied and results interpreted.

Part two

Present day technology, *i.e.*, the present state-of-the-art, imposes very little constraint on the determination of centralization vs decentralization, *i.e.*, what functions are carried out by what echelons. This is entirely a matter of the personalities involved, and the willingness to implement a system which will permit the degree of command and/or control interrelationships desired. In other words, any desired organizational structure can be supported from an information processing point of view; however, the cost/effectiveness issue may impose realistic constraints. Management now has at its disposal an impressive array of tools; the question is, how does management choose to use what is available?

The previous point leads to a consideration of what is perhaps the most significant issue. From the broad view-point, what is the impact of information processing potential on organization and mission? There are many facets to this question. The question we really want to ask is, what does information processing capability enable us to do at this time that we either could not do at all before, or could do only with great difficulty, and so what effect does this have on the way we organize and the functions we assign?

First, there is the classic case of using information processing to automate an existing manual system. This is to say that nothing is really changed except that some people are relieved of tedious tasks, or some tasks are accomplished faster. This is a very important application, and the first and most important task of system analysis is to determine whether or not simply automating the manual processing is what is required. For example: Optical character readers can enable direct input of data to a computer, speeding up the process and requiring less personnel; messages can be logged automatically; operational data can be extracted from files much faster than from a multitude of messages and reports, etc. Little if anything changes in the fundamental way that the organization works.

By the same token, it is possible for an information processing system to enable a whole new spectrum of functions to be performed simply because vast quantities of data can be processed at a central location, very quickly, and minute details considered along with all implications, thus bypassing many echelons previously needed for the same information processing. Now there can be a requirement for additional staff elements to perform analyses and help formulate policy, functions which were previously accomplished at a lower echelon or by a special group established to provide these services. There are many examples within the Services and industry of this trend.

It is interesting to note that, in the case where both information processing and decision-making are centralized, there is not necessarily a modification to the existing organization. One of the most significant and obvious manifestations of this is the manner in which the National Command Authority, specifically the President, Secretary of Defense, and the JCS interface more or less directly with subordinate commands of the CINCs. Part of this is due to communications, which enable almost real time reporting of events. Part is also due to the ability to store and process vast quantities of data which previously would have been aggregated by the CINC and forwarded to the NCA as summary reports.

Although the CINCs remain in the direct line of command, data flows directly from their subordinate field commands to Washington. This data is simultaneously furnished to the CINCs and intermediate levels so that they may process it for their own purpose, know as much about the situation as the authorities in Washington, and be prepared to discuss it with them. The availability of all such information in the Washington area tends to generate questions and discussions on matters concerning many echelons of command. This in turn tends to encourage direct discussions between authorities in Washington and the field forces involved. Intermediate echelons should, of course, participate in such discussions. This way of doing business does not necessarily mean that all decisions are made in Washington, but the pressure for rapid and complete answers to questions and the implied requirement for solutions to problems is far greater than before.

Two possible approaches to data reporting and processing might be considered: One is to have all data, however detailed, sent directly to the highest echelon. In this case, not only can the detailed data be aggregated to provide summary reports appropriate at the highest level, but the data can be used by persons not familiar enough with current operational problems to draw valid inferences. For example, planning factors modified by recent operational experience may not be available to accompany the raw data. There is no question that the same detailed data can be used at all levels. The question is what processing capability should be available to the highest echelon?

The second approach is to have the detailed data held in data bases at lower levels, not to be tapped as raw data, but to be processed or partially processed at that or an intermediate level for forwarding to the higher level. Thus, when certain reports were required, the reports themselves, or summarized data, would be available to the higher level on demand. This would avoid having the data available with the attendant danger of using it incorrectly. This also enables the lower echelon to have at its disposal data which can provide the basis for answering more detailed questions which may arise without being pre-empted.

Part three

The organization of command and control functions has a significant influence on the organization of the associated information systems and vice versa. It appears that there are at least five general concepts which might evolve:

- The organization remains the same and the functions remain the same; information processing merely enables more efficient carrying out of existing functions.
- The organization remains the same, but functional responsibilities change, *e.g.*, a lower echelon no longer has the same degree of local control that it once had, some of this control having been assumed at higher echelons.
- One or more lower echelons are eliminated or cut out with respect to being in the chain of command upward, although they still remain in the downward chain.
- The lower echelons can have semi-autonomous local control, providing information properly aggregated to higher echelons, resulting in much closer coordination without giving up all prerogatives.
- The information processing can be centralized with an attendant increase in local control. The only significant organizational change in this instance is the technical support at the higher echelon.

Information systems and the implementing organization

by ROBERT E. HARSHBARGER
Defense Communications Agency
Washington, D. C.

Most discussions of information systems consider the subject in its continuum from source data gathering through the various steps of communications to a central location, automatic screening and manipulation of the data, display or presentation to a user, the user analysis and evaluation process and finally the decision and action response process. To the implementer this represents a series of separate though closely interdependent implementation processes demanding differing techniques, technologies, disciplines and supporting organizations. To concentrate on implementation areas having some commonality of technology and organization, this discussion will address only that segment of organization concerned with implementing the data screening and manipulation process and the process of providing the resulting data or information to the user; in essence, the concern of automatic data handling from the time data is received at a central location through its application to a user problem. Organizational elements concerned with data gathering and reporting, with communications systems to support this reporting and with decision and action response are not specifically addressed although the remarks presented are believed to be generally applicable to these areas of implementing activity.

Any discussion of information systems and their impact on organization and mission or on any other element of management or technology upon which they impinge must be preceded by some descriptive argument reconciling the ambiguities of the term itself. This term, information system, like many others in the current technical jargon bears clear definition only in the mind of the individual applying the term. The vision may focus at any point in the spectrum from the most rudimentary manual methods of collecting data through the varying degrees of complexity as found in large manual systems for reporting or summarizing data, computer assisted systems for providing a data bank, automated systems permitting inquiry of the data bank, analog devices for converting directly observed data to useful information, complex systems of multiple users interacting independently in apparent real time with a large automated central data bank, and on to the vastly complex and interconnected systems of men and machines as may be beheld only by the visionary divorced from practicality. To select from this span of diverse concepts common threads impacting on organization and mission is not possible due to the diversity of the supporting organizations themselves. Rather, in order to focus the discussion to points germane to this Congress, it is necessary to delimit the meaning of the term to some specific concepts and capabilities recognized in current computer automated systems or anticipated in the following generation of systems.

The term, information system, as subsequently referenced will include computer systems containing an alphanumeric data bank accessible in apparent real time by each of a number of users, including automatic response to critical parameter thresholds, to obtain read-out of data or to derive information from such data and providing the means for rapid display of the data or information so derived. While this somewhat restricted definition is not necessarily descriptive of current national level command control data handling systems with which I am closely associated and toward which I will generally direct my remarks, and does not reflect any specifically approved planning toward such goals, it is representative of that minimum system I believe necessary to satisfy current demands at the national level. Indeed, in the absence of well defined goals, an assumed definition is necessary to the implementing organization if chaos is not to reign supreme. The implementing activity is one close to reality, solving today's problems today, making decisions on methods and techniques applicable today but which influence tomorrow's decisions through the impact of economic investment and continuity of applications, and providing system growth and transition to meet daily changing requirements. Orderly and economically sound growth demands well defined goals whether specified or assumed. The remarks which follow will not propound philosophy nor argue justification of systems' being. Rather, they stress the need for clear goals, whatever the justification, and the impact of the existence or absence of such goals on today's implementing organization and activities.

The dynamic growth of information technology during the past 10 years has given rise to a number of automated military information systems of varying complexity utilized in storage and manipulation of alphanumeric data. Existence of these information systems, though generally they may be better described as data banks, has brought recognition of the wide variety and large volumes of data that may be efficiently stored and manipulated. This facility for storage and manipulation has in turn broadened the demand for more and more diverse and detailed data accumulation at various command levels. The past three years have seen unprecedented advances in communications to support this demand and most notably have witnessed the evolution of broad and well disciplined reporting systems oriented toward utilization of computer methods in data transmission and data handling. These mechanisms allow, and indeed today are accomplishing, the accumulation of great volumes of data at many levels of command. Unfortunately our alacrity in producing the means for translating these large data volumes into information assimilable by the interested user has not matched our progress in providing the data itself. I somewhat sanguinely presume that this will be the next revolution in the information systems evolution.

Throughout the history of systems relying on analog methods for obtaining source data the conversion to usable information has been an accepted part of the system. This derives both from the generally unintelligible nature of the source data to the ultimate user and from the great volume of individual data inputs. In the typical radar or sonar application the automatic conversion of source data to a few blips on a scope depicting range and location of detected objects is only an initial step which partially reduces the vast volume of source data. Useful information derives only after extensive distillation of many such observations into some operationally meaningful statement of number of hostiles penetrating, their relative location, their course or closing rate, expected time of closure, and similar factual details on which decisions can be made. Such information, which in no way conveys the vast volume of source data from which it was distilled, is commonly accepted as system output. Similarly, the often referenced Semi-Automatic Ground Environment System initially grew from these same precepts of the analog system. It was early recognized that higher and higher levels of aggregate information were necessary to rapid-response decisions required in the air defense mission. Moreover this aggregation has been carried to the integration of information from widely dispersed centers; information aggregated to a level of context completely removed from the source data. Here again information reflecting in no way the vast conglomerate of source data is accepted by the operational user.

Current alphanumeric information systems have become quite analogous in complexity to these analog systems; data volumes reporting facts and detail of facts have become far too numerous for assimilation by the individual user; data coding makes direct reading almost unintelligible; complexities of the interdependence and interaction of these detailed facts preclude intuitive judgments and decisions; and distillations of these data volumes into more comprehensive levels of information are necessary if decision responsiveness is to be effective. The simple data bank or inventory control approach which accepts source data and subsequently, only on demand, spews forth minutia or summarized data sorted in many ways will provide an initial level of volume reduction comparable to that evidenced by the blips on the radar scope. Automatic distillation to much higher levels of information content are not possible in the data bank approach since neither the hardware nor software associated with this approach adequately support the voluminous arithmetical and logical calculations necessary to the distillation. Even the simplest extension of information level to the sensing of parameter thresholds cannot be effectively incorporated in the data bank environment. Some such sensings that are at best limited include indications that an event has or has not occurred, parameter value has exceeded some specified boundary, rate of parameter value change has exceeded some specified norm or that experienced in a previous time period, or that the fraction of remaining resource capability has fallen below some specified critical level. Yet such minimum extension is necessary to direct the user's attention to areas of critical change or occurrence demanding further investigation or action response; else he must make frequent and time-consuming inquiry into the status of a large number of individual parameter conditions or must peruse voluminous batch processed printouts or paper listings to sift out these more critical changes as they occur. Both time for such random inquiry and the volume of data detail to be perused preclude effective decision response as data inputs continue to grow in size, detail and timeliness. Further extension to information distillates conveying the impact on operations of the interaction of multiple parameters involved in operations planning, status of forces and resources, plan execution and following, and transportation and logistics support can be accomplished only by a comprehensive information system as opposed to a data bank.

Distinct from the personal reasoning above, the implementing organization, as an organization, is indifferent to why a specific concept is accepted. It needs only a generally clear definition of the near term and longer range goals toward which the system is aspiring. These goals must distinguish between data banks and information systems or levels between, must be generally accepted by management and operating officials at all levels, and must be achievable in a reasonable and generally accepted time frame. The impact on the implementing activity of the system definition implied by such goals is very direct in both mission and organization. As regards the data bank definition on the one hand, mission encompasses little more than providing a storehouse for data and distributing the data in various forms on demand to multiple users; the more comprehensive information system, on the other hand, extends mission scope to include development of the necessary tools to permit automatic response as specified conditions are met, to allow multiple users to interact on-line with the data base, to permit complex queries and analyses involving multiple parameters, and to provide rapid display and distribution of significant results. These obvious differences in mission dictate widely varying organization structures, demand quite diverse technologies, require quite different academic disciplines and experience backgrounds, and imply completely contrary views of user function. The implementing organization, not being clairvoyant, requires adequate mission definition if its efforts are to be efficiently directed. The alternative is to hedge against the entire span of possibilities thus ensuring that a large proportion of its efforts are misdirected if not totally lost.

H. D. Benington noted in 1964 in a somewhat broader context that " . . . almost everyone emphasizes evolutionary system design, not fully appreciating, however, the eventual impact that this new approach will have on the role of technologists, the goals of our research, the development of tools, and the organization of the command." At the level of the implementer, evidence does not indicate that two years have added significantly to the appreciation of the impact of this evolutionary approach other than to change the verb tense of Mr. Benington's statement from "will have" to "is having."

Distinction between the evolution of hardware systems, software tools and applications programs is not generally recognized. To the implementing agency this is reflected in a general slowdown or lack of progress in applied research while information systems as referenced herein require a vigorous research program, the basic role of the technologist is virtually ignored as initial increments of the evolution almost totally absorb available resources, development of information systems tools specifically for the evolutionary needs of the military environment receives only minimal interest (other than in communications) defeating the very concept of evolution, and organization at best becomes unstable due to the lack of appreciation of the above. Information systems coupled with an evolutionary approach require that the designer and implementer provide a series of tools developed outside the user environment to ensure a flexible system within which the user can, in his own environment, develop programs and procedures applicable to specific problems. These tools incorporate individually complex control programs, programming languages, inquiry languages, console devices, data transmission systems, mathematical simulation and utility systems, and data management and manipulation systems into a single complex in which they effectively and efficiently interact in a manner easily accessible to the user and the applications programmer. Information systems therefore require that the implementing organization include an element devoted to the applied research, development and implementation of these tools. The complexity of the technology required for this development demands expert technical and logical talent experienced in the disciplines of complex computer systems and strong technical management to ensure compatibility of the many complex elements. This presumes that the evolution is directed toward the goal of an information system rather than toward a data bank which can be provided by well-known technology with minimal developmental requirements. Since organizational elements and organizational competence are not created, but are built over a period of months or years, distinct goals toward which to build are essential. Based on a common understanding applied research can be initiated, acquisition and retention of technological competence can be accomplished based on known discipline requirements, tools serving the continuing system evolution can be developed, and an organizational structure responsive to the technological requirements and to the needs of the user and management can be established.

Information systems bring to the implementer the need for strong centralized control in both the implementation and operation of the system. The many components of the central computer control system are so complexly interrelated that a single manager is necessary during implementation to ensure the efficient functioning of these parts as a total system. Since such a system serves many functional users a single manager must recognize the needs of these many users to ensure that system scope is sufficiently broad, both functionally and in detail, to meet these needs. He must also control any changes or additions to the system since even a minor change in one component may impact on many other components. In operational use strong control is necessary to reflect priorities, determine saturation points, detect and correct system errors, and most importantly to ensure that all users are aware of all available standard algorithms and routines; the latter to prevent multiple duplication of programming and often inefficient programming of frequently used routines which are the two greatest contributors to inefficiency in large information systems.

The implementing activity must provide the organizational element and disciplines to assist the user in the application of these automatic tools to his specific information needs. The problems of the military user have become vastly more complex during the recent years that have embraced the policy of controlled response and have at the same time seen the availability of more and more detailed data relating forces, plans and actions at all command levels. This has required a more complex logical technology in evaluation of the military environment. Man has of necessity learned to cope with this increased complexity and continuously becomes more technically sophisticated in order to do so. The military user is indeed the expert in his own environment; understanding the problems; able to define what information is needed for their solution; capable of defining the general methods, procedures and problem logic necessary for developing applications programs. The implementer need only to guide in the use of the automatic tools and assist in actual programming where required. Yet, too often, the user is coerced into spending valuable time in learning the details of computer programming, usually at a machine language level remote from any application he will ever encounter, in an attempt to unravel the mysteries surrounding his information system. In order to provide the user with the time to properly do the job for which he exists rather than become an expert in noncontributory peripheral areas, the implementer must provide, as part of the user assistance, a mechanism for keeping the user informed of system capabilities and limitations and the procedures for applying the automatic tools available to him. These should be explainable and applicable in technical English, mathematics, and logic without reference to the cryptic language of the programmer and program systems. Presentation must be in a manner which instills confidence in the system, denoting simplicity and ease of application. To provide less will only compound the already complex problems of the user and divert him from his primary functions. Here the technical competence of the implementer and of the scientific community in providing such tools is being tested, not that of the user. Provided adequate tools and succinct guidance in their application the user will advance at a rate dictated by his own requirements.

The phenomenal growth and advancing technology of information systems also brings to the implementer the traditional problems of an expanding organization. Frequent change in organization structure to provide response to new functional areas, success and failure in the investigation and research into new technologies, justification of increased resources, acquisition and retention of competent personnel, training in application of new technologies, and confronting squarely the decision required in the face of uncertainty are all internal problems with which the implementer can cope so long as his goals are well defined. In the absence of such goals each of these areas become, in themselves, a time consuming and wasteful process bent on outguessing the future.

While all of the areas of concern heretofore mentioned impact directly on the implementing organization, none are insurmountable in the presence of clearly established goals. Presuming their existence, one area of significance remains, that is, the need for conveying to the basic research and design activities the requisite changes and advancements in the overall system. The implementer, being continously in close contact with the users in their applications, is the first to recognize the need for major changes or additions to the then existing system components. In an evolutionary environment these requirements for change generally appear in small increments and often are incorporated as minor improvements or as part of system maintenance. They derive from the observations of several organizational elements and emerge as basic design changes only after experience shows the need for some general improvement in overall capability; frequently in the form of a need to make more efficient some processes already in being. A specific element of the implementing organization with the mission requirement of uncovering and defining such needs is necessary if continuing liaison between the design organization and subelements of the implementing organization is to be maintained. The need must be transmitted in some formal fashion from either the user or implementer to the design activity. This separate organization activity will inevitably remain one step removed from the system and the user thus diluting its effectiveness. The manner in which the need for basic design changes derive, the necessary formality of transmitting recognized design needs to elements even further removed from the system, and the necessity of separate organizational elements maintaining detailed knowledge of systems programs and procedures make this process at least cumbersome and time consuming if not inefficient with inadequate response. From the view of the implementer the mission areas of information system design and implementation are not separable where the concept of evolutionary system development prevails.

The current challenge to the scientific community and to the implementing activity, in the continuing evolution of information systems, lies in the full realization of the inundating volume of data rapidly becoming available to the user and the concomitant demand for the tools which will allow the user to automatically translate this volume into higher levels of information assimilable at the decision level. Further, though the tools themselves may be complex, it must be recognized that the view presented to the user should be one of simplicity and ease of use, one which instills confidence in the utility of the tools rather than compounding the problems and impairing the technical efficiency of the user. Equally important is the challenge to management at all levels of recognizing these current demands, generally defining the longer term goals, and of providing the policy guidance and resources through which the tools can be developed and their applications exploited.

An overview of strategic mobility and its implications for design of analysis systems

by MARVIN L. MANHEIM
Massachusetts Institute of Technology
Cambridge, Massachusetts

INTRODUCTION

The purpose of this paper is to establish the basic characteristics of a comprehensive and flexible capability for strategic mobility analysis. This objective arises from the recent establishment within the Organization of the Joint Chiefs of Staff of the Office of the Special Assistant for Strategic Mobility (SASM).

SASM is charged *explicitly* with a wide range of missions, coverning the full spectrum of strategic mobility. SASM is charged *implicitly* with executing his strategic mobility functions with the best analysis technology now available.

This analysis technology has three major components. The primary impetus is that of systems analysis, the generic term for rigorous analysis of difficult problems. The second is the technology of computers, required by the complexity of strategic mobility problems. The third technology is the accumulated knowledge of transport systems analysis, together with the associated mathematical and computer models.

In response to this implicit charge, SASM has under way a major effort to develop a broad analysis capability for strategic mobility. Development of this capability requires maximum use of the systems analysis approach, of computer support, and of transportation system models. The purpose of this paper is to discuss the basic characteristics of this analysis capability.

We begin with a discussion of the strategic mobility problem, in order to clarify the substantive area of concern. Next, we give a relatively cursory discussion of issues in developing models for use in strategic mobility. Third, we discuss systems analysis as a framework and indicate the limitations of this framework and its implications for analysis procedures. Fourth, we discuss briefly the kind of computer environment which can be provided, and the implications this environment has for design of an analysis capability. Finally, we summarize the implications of the preceding discussions by identifying the basic characteristics of a strategic mobility analysis capability.

The arguments of this paper are by no means definitive and final, but are presented with the objective of stimulating discussion in the technical communities concerned with strategic mobility analysis.

I. The Strategic Mobility Problem

A. The triangle of resources, requirements and criteria.

The strategic mobility problem can be represented by a triangle, whose three corners are: strategic movement resources; strategic movement requirements; and criteria for selection of strategic mobility plans (Figure 1).

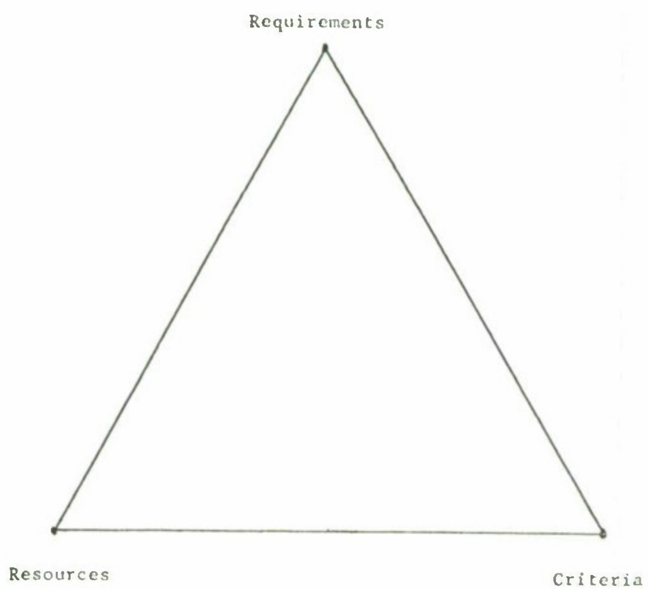


Figure 1 – The strategic mobility triangle

Strategic movement resources are all the transportation resources available for support of worldwide mobility, including military owned, commercial for-hire, and non-military vehicles and networks available under alternative conditions. These are enumerated in Figure 2.

A. *Vehicles*
- Aircraft
  - MAC nucleus
  - Commercial, voluntarily available
  - Civil Reserve Air Fleet
  - Non-U.S. owned
- Ships
  - MSTS nucleus
  - U.S. commercial, voluntarily available
  - Flags of Convenience
  - Other foreign-flag commercial, voluntarily available
  - Reserve fleets
  - Other available
- CONUS and Theaters
  - Rail cars—box, flat, heavy duty flat, POL, passenger
  - Trucks—commercial for-hire, private carrier; military—organic, non-organic, other government-owned
  - Buses—commercial, private carrier, military, other
  - Inland waterways
- Special movements
  - LSD's, LCM's
  - Aircraft carriers and other helicopter transports
  - Floating cranes
- New technologies
  - Ground-effects machines
  - Containers

B. *Networks*
- Installations
  - CONUS origins—home stations, depots
  - Theater destinations—depots, staging areas
- Terminals
  - Water terminals (POE's and POD's)—constructed and LOTS (logistics-over-the-shore)
  - Air terminals (POE's and POD's)
  - Enroute air terminals
  - Enroute sea terminals
- Links
  - CONUS—rail, highway, inland water, coastal
  - Inter-theater—air, sea
  - Theater—rail, highway, off-the road, inland water, coastal

Figure 2—Strategic movement resources

Strategic movement requirements are all movements through the worldwide transportation system which are in support of a strategic plan or which impact on the ability of the transportation resources to support that plan. These include the forces being deployed, including personnel fillers and replacements and resupply, and reverse flows from the theaters, as well as civilian and non-deployment flows within the the theater, within the Continental U.S. (CONUS), or between the CONUS and the theaters (such as existing channel traffic and household goods and dependents movements). These are summarized in Figure 3.

- Units
  - Personnel
  - Equipment
  - Accompanying supply
- Personnel
  - Fillers
  - Replacements
- Resupply
- Reverse Flows
  - Civilian dependents
  - POW's
  - Medical evacuees
- Prepositioning
  - Equipment
  - Supply
- Reinforcements and redeployments worldwide
- Special movement requirements
  - POL
  - Helicopters
  - AID cargoes
- Channel flows
  - Support of military forces
  - Support of civilian economy

Figure 3—Strategic movement requirements

The third element of the triangle, criteria for selection of a mobility plan, includes consideration of the movement times and costs (fixed and variable) associated with alternative strategic mobility plans, and considerations of effectiveness of plans as reflected in the effectiveness of the forces delivered, and vulnerability and reliability. These are further detailed in Figure 4.

B. Problem types

The significance of this triangle is that it focuses on the variety of problem types which can be formulated for strategic mobility analysis. Any given problem can be characterized in terms of the three vertices of the triangle, by indicating which vertices are specified in the problem statement, and which are to be found by solution or analysis of the problem (Figure

5). Various combinations are shown below, with examples:

Time (Arrival at destination)
- Completion of total deployment
- Specified phases: by force packages, or by mode of delivery
- Actual departure, relative to ready date
- Actual arrival, relative to required dates

Costs
- Capital costs
- Operating costs – fixed variable

Effectiveness of forces delivered
- Rate of force buildup
- Unit integrity
- Matchup of cargo and passengers
- In-transit time and travel conditions
- Time left available for training

Flexibility
- Nature and location of constraints on flow
- Actual utilization of movement resources as compared to potential

Vulnerability
- Convoy size and speed
- Dispersion of vehicles and units over time
- Dispersion of vehicles and units over space
- Essentiality of specific links or terminals

Figure 4 – Strategic mobility criteria

(1) Requirements and resources are specified, find performance with respect to the criteria:
  a. What time will it take to deliver the requirements with the specified resources?
  b. What is the cost of using these resources to deliver the requirement?

(2) Requirements and criteria are specified, find the resources required:
  a. Resources required to meet a specified deployment completion time?
  b. Resources required for minimum cost deployment which meets the required time and effectiveness?

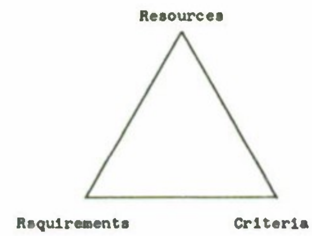
(3) Resources and criteria specified, find the requirements which can be delivered:
  a. Quantity of personnel and equipment which can be delivered in a given time?
  b. Within specified cost limits?

All three problem types arise.

This triangle is offered as a conceptual aid only, for it does submerge many subtle aspects. For example, these problem types do not follow the patterns above completely:

(1) Requirements specified for different time periods, find minimum cost use of transportation resources over all time periods.

(2) All transportation resources specified except for the number of new large aircraft, which is to be found. This is a special case of the more general problem of finding the transportation resources given the requirements and the criteria; even with the restriction on transportation resources, the requirements and criteria must still be specified for the problem to have meaning.



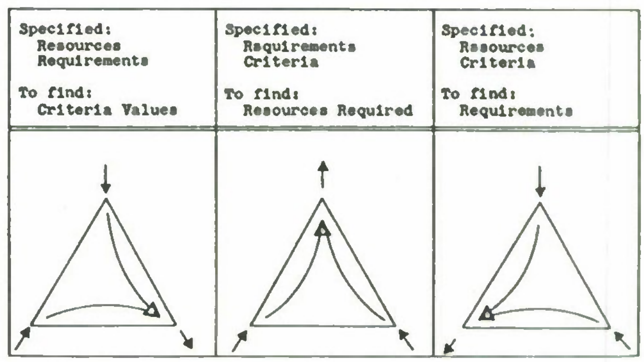


Figure 5 – Problem types

C. Routing and scheduling

The core of the strategic mobility triangle is movement routing and scheduling. At the most detailed level of strategic mobility analysis, a complete detailed movement schedule is actually established, for every movement unit being deployed. This schedule traces the unit from initial origin through port of embarkation (POE) and port of debarkation (POD) to theater destination, indicating for each leg the transportation mode, route, vehicle or vehicles assigned (e.g., ship name, number of aircraft by type), and detailed timing information for arrivals and departures (Figure 6). Such a schedule is the result of routing and scheduling through multiple transport modes.

When the type of analysis does not require the level of detail represented by the complete movement schedule, "routing and scheduling" are still present. Even for the most general capabilities studies, rout-

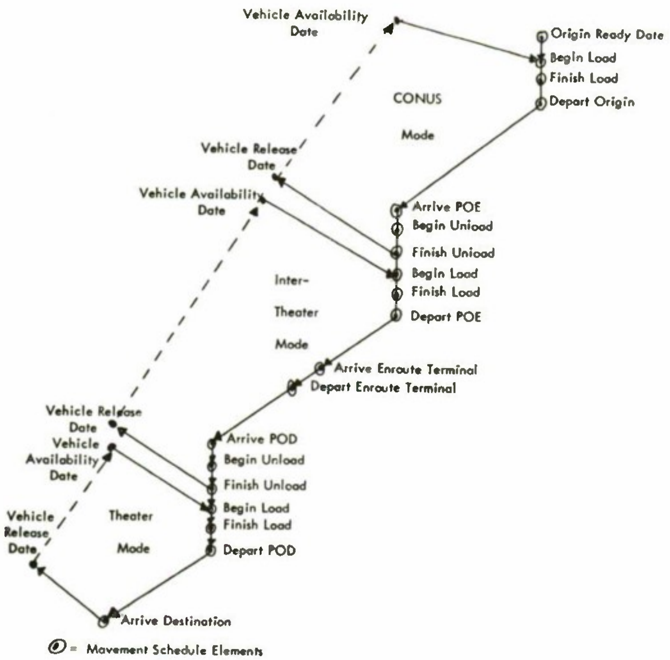


Figure 6 — Detailed movement schedule

ing and scheduling decisions are implicit; any statement about the capability of a particular transportation facility is based upon an explicit or implicit assumption about routing and scheduling within the system. For example, an air channel capacity of 1000 short tons per day is based upon assumptions about types of aircraft, operating policies and routing and scheduling of aircraft and crews over routes in and out of the channel (cycling to maintenance, etc.).

As shown in Figure 7, this has significant implications for the analysis of strategic mobility problems. If routing and scheduling is not done explicitly in the analysis, assumptions about routing and scheduling and their impact on channel and other capabilities must be verified. This argument is particularly important in understanding the relationships among different types of transportation models.

There are four types of routing and scheduling decisions:

(1) Mode selection decisions. These decisions depend upon such factors as vehicles available, time available for the move, route capabilities (highways, rail lines, enroute airfields, etc.), and cost (dollars and strategic value of resources).

(2) Route selection decisions. These decisions can be broken into two parts:

a. Selection of POE's and POD's, which depends upon such considerations as the mode selected for inter-theater lift, port capability, and port vulnerability.

b. Selection of routes for a single mode, which depends on such factors as POE's and POD's chosen, availability of enroute support for vehicles being used, and flexibility (potential for enroute diversion around congested or vulnerable links).

(3) Vehicle selection decisions (within a mode). These decisions depend on such considerations as the number of each type vehicle available at the time and place required, and the total lift capacity of available vehicles as compared to total lift requirements.

(4) Timing decisions. Establishment of movement schedule times depends on the routes selected, speed of vehicles selected, anticipated queues enroute, and whether scheduling is done from the availability date forward (availability date plus travel time equals predicted arrival time at destination), or from required delivery date back (required delivery date minus travel time equals required time of departure from origin).

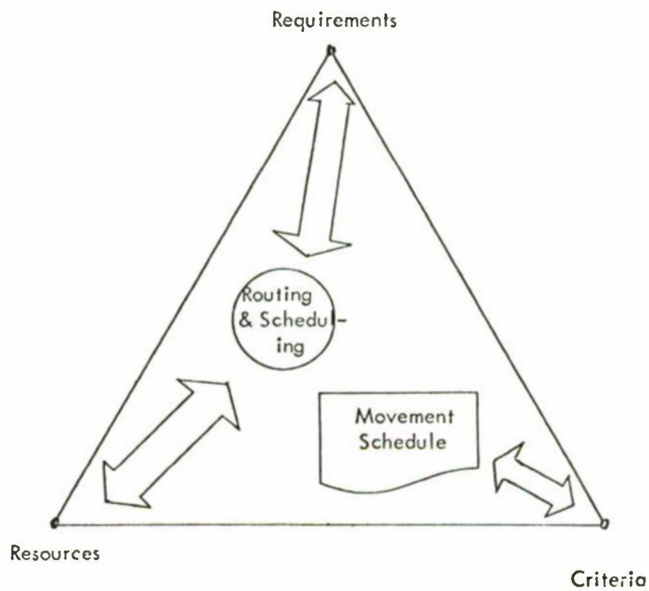


Figure 7 — Routing and scheduling

These routing and scheduling decisions interact greatly. Decisions of each type will be influenced by decisions of other types. Hence, the sequence in which these decisions are developed will strongly influence the movement schedule developed, and the evaluation of capability.

To the maximum extent possible, it is desirable to consider the total transportation system as a single system, not just one part. Decisions cannot be made for any mode in isolation—for example, sealift or airlift. Strategic mobility analysis must deal with factors throughout the system, including the large number of origins within CONUS, various unit readiness dates, various required delivery dates, various tradeoffs between modes and within modes, between routes, between speed of deployment and readiness status, etc., as well as various constraints (POL availability, port throughput capacity, cargo needing specialized handling, etc.).

D. Time frame distinctions

A typical division of strategic planning problems is based upon time frames:

(1) Long-range planning—over five years.
(2) Mid-range planning—one to five years.
(3) Contingency planning—within the current year.
(4) Operational planning and current operations—right now to one to three months.

The triangle of resources—requirements—criteria is valid for each time frame. The dominant aspect of the problem is the relationship of the problem to the triangle. For example, it does not matter whether we are dealing with the transportation force structure of the one-five year range or the over-five year range; what is most critical for the mobility analysis is whether this is a problem of the resources-given or the resources-to-be-found type. Time frame distinctions play a significant but lesser role in determining the analytical techniques required.

Certain general patterns do emerge, in normal planning practice, relating the strategic mobility triangle to time frames. The following types of strategic mobility problems are typical:

(1) *Operational Planning.* Transportation resources and movement requirements are specified generally in detail; the problem is to determine the most efficient application of resources to requirements. The criteria of efficiency are predominantly time and effectiveness of forces delivered.

(2) *Contingency Planning.* Transportation resources are fixed. Planner wishes to explore costs, time, and force effectiveness implications of the most efficient application of the transportation resources to alternative levels and mixes of movement requirements.

(3) *Force Structure Planning.* Several sets of movement requirements are given. Each set of requirements corresponds to a different contingency or set of contingencies. For each set of movement requirements, the least-cost mix of transportation resources required is to be found for alternative levels of efficiency (time and effectiveness of forces delivered). A desirable force structure is determined by "averaging" over all the sets of requirements.

E. Analysis functions

All strategic mobility functions can be summarized under the major functional tasks of Force structure planning, Contingency planning, and Current operations. For instance, the function of monitoring research and development in strategic mobility is primarily concerned with the introduction of new vehicles and transportation technologies in the long-range time frame. The general issue is, how does the performance of the transportation system change with changes in vehicles or other technologies. This question may sometimes be applicable to current operations also. Therefore, in all time frames of analysis there is a general option open to the planner to explore changes in vehicle characteristics.

This discussion of the strategic mobility problem leads to Figure 8. This figure shows how the major functional tasks in strategic mobility lead to decision issues. These issues can be related to options open to the planners, which in turn lead to the basic analysis problem, balancing resources against requirements. The product of analysis is information upon which to base decisions.

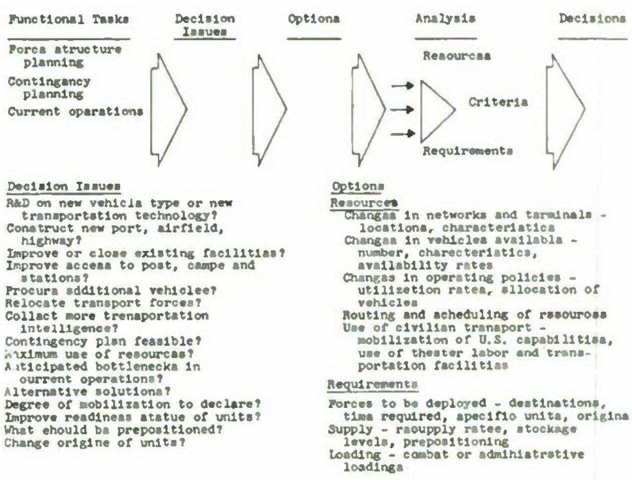


Figure 8—Functions of analysis

F. Conclusions

There is no single type of strategic mobility problem. A system for strategic mobility analysis must accept different formulations corresponding to different aspects of the triangle and different time frames. Inputs to the system will generally consist of resources and/or requirements and/or performance criteria. Even if routing and scheduling is not done explicitly, all factors and assumptions used must reflect explicit consideration of implicit assumptions about routing and scheduling.

II. Modelling Strategic Transportation

The basic concern of strategic mobility is transportation; the core of the strategic mobility problem is routing and scheduling of transport vehicles. Techniques and models developed for other transportation problems are applicable to strategic mobility.

A. Principles of transport systems analysis

A number of general principles for the analysis of transport systems are shown in Figure 9 (a more concise statement of these principles is found in Manheim, 1966 b). These principles have direct application to the problem of modelling the strategic transportation system.

I. Transportation systems=networks+ vehicles.
II. All modes of transportation.
III. All movements.
IV. From initial origins to final destinations, through all modes.
V. Decision variables: short-run to long-run
VI. Transportation-related decision variables—particularly those influencing demand.
VII. Full set of consequences.
VIII. "Market": supply and demand reach equilibrium within the constraining channels of the transportation network.
IX. Comparative analyses must maintain the same set of assumed conditions.
X. Transportation is not an end in itself.

Figure 9—Principles of transport systems analysis

Principle I

Transportation systems are composed of networks and vehicles moving through those networks. Networks consist of nodes and links which connect pairs of nodes. Some nodes may be interchange points between links of the same mode; other nodes may be interchange points between links of different modes, commonly called terminals.

Application to strategic mobility analysis

In strategic mobility, the transportation system of interest is, potentially, the complete world-wide transportation network. A highly abstract model of this network is shown in Figure 10.

The implications of this principle for modelling strategic transportation are sufficiently great that we reserve discussion for Section III-B.

Principle II

All modes of transportation must be considered. (A mode is a particular set of vehicle, supporting way, and control and propulsion systems technologies, together with a broad set of operating policies.)

Application to strategic mobility analysis

The strategic transportation system includes all modes of transportation utilized by the military, in-

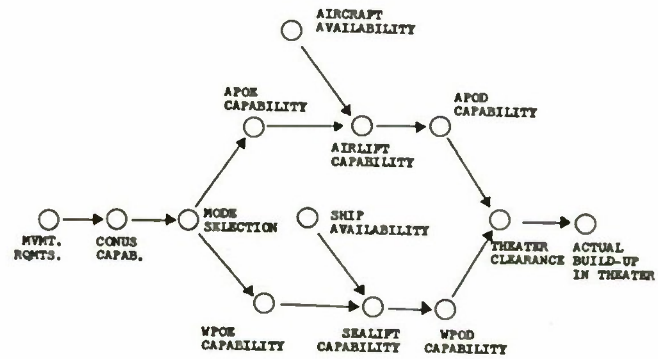


Figure 10—Strategic mobility factors

cluding military-owned air, sea, highway, rail, and terminal transportation capabilities; and the same capabilities in private or other governmental ownership and potentially utilizable by the military. (See Figure 2.)

For strategic mobility analysis, models must be developed for all modes of transportation. Further, models must be able to represent proposed new technologies.

Principle III

All movements through the transportation system must be considered.

Application to strategic mobility analysis

Analysis must consider all movement requirements moving through the world-wide transportation system at the same time: units and accompanying supply being deployed; personnel fillers and replacements; resupply; reinforcements and redeployments world-wide; reverse flows from the theaters; existing or assumed channel flows to support forces already deployed; and flows to support the civilian economics in the CONUS and in the theaters. (See Figure 3.)

Strategic mobility models must be able to simulate the flows of all of these movement requirements, considering the special characteristics and resource needs of each.

Principle IV

Movements must be considered from their initial origin to final destination, through all modes.

Application to strategic mobility analysis

Models of the strategic transportation system must consider the movements of forces and supplies from initial origins (home stations or depots); through the CONUS transportation system to the air and sea ports of embarkation (POE's); through the inter-theater air and sealift systems to the theater ports of debarkation (POD's); and through the theater transportation systems to destinations (staging areas or theater depots). (See Figure 6.) It is not sufficient simply to model the airlift phase. Models must also consider movements from one theater to another or from theaters back to the CONUS.

Models must consider, not only the performance of single modes in isolation, but also interactions between modes, at terminals and other interface points. Models are required which specifically address the relationships between movements into and out of interface points where movements transfer from one mode to another.

Principle V

The set of transportation decision variables should cover the full scope from short-run to long-range options:

a. long-range investments, such as changes in the fixed facilities (networks and terminals), size and composition of available vehicle fleets, and changes in transportation technologies (vehicles, supporting ways, etc.);
b. mid-range options, such as options with regard to the procurement, distribution and allocation of vehicle resources, and network operating policies (routing and scheduling policies);
c. short-range operating decisions, including detailed routing and scheduling, and assignment of vehicles.

Application to strategic mobility analysis

Long-range options include: changes in terminal locations and capabilities; changes in locations and capabilities of origins, including home stations and depots; changes in network configurations and link capabilities; including construction of new transportation links; changes in numbers, types, and operating characteristics of aircraft, ships, railroad cars and other vehicles; research and development of new transportation technologies including vehicles, facilities, and networks.

Mid-range operating policies include: changes in the numbers of aircraft, ships or other vehicles allocated to a theater; changes in readiness status and mobilization rates; changes in routing and scheduling criteria and other operating policies.

Short-run operating options include: routing and scheduling of specific movements, assignments of movement priorities, and detailed decisions at the installation, terminal and transportation operator levels.

Strategic mobility models must have provision for analysts to express particular options of all these types. The models will be used to explore alternative options, and so should be designed explicitly for ease of use for these purposes. This does not mean that every model must incorporate the full set of options, but that every option must be provided for in at least one model in the set available.

Some of these options are illustrated in Figure 8.

Principle VI

In addition to direct transportation decision variables, transportation-related decision variables must be considered, particularly those variables which can influence directly or indirectly the demand for transportation (distribution of demand over space, over time, and/or by type of transportation resources, and level of service required).

Application to strategic mobility analysis

Non-transportation factors which influence the demand for strategic transportation include: uses of prepositioning and forward bases; changes in initial locations and readiness status of movement requirements; changes in consumption rates; changes in composition of the forces in the theater or in tactics; use of local products as opposed to items moved from the CONUS; and ability to do without specific kinds of items and/or forces.

This principle implies that in addition to various transportation models, there must also be models and procedures for expressing the indicated options. For example: changes in movement requirements, in locations and readiness status of requirements; etc.

Principle VII

The full set of consequences of alternative transportation systems policies must be considered, including:

a. dollar costs of capital investments in transportation vehicles and facilities;
b. dollar-valued operating costs;
c. dollar-valued changes in costs born by users of the transportation system;
d. Non-dollar-valued costs born by users of the system (such as transit time, comfort, convenience, flexibility, reliability, vulnerability, effectiveness);
e. non-dollar-valued costs and benefits incurred by non-users of the system;
f. other non-dollar-valued or non-quantifiable aspects of the impacts of transportation alternatives.

Application to strategic mobility analysis

Capital investments in fixed facilities including networks and terminals and in vehicles are direct and obvious, as are also operating cost components such as fuel, maintenance, personnel, etc. Dollar costs to the users include direct transportation costs, as well as such indirect factors as quantities of goods in the pipeline; deterioration or wear and tear of goods; cost of distribution systems, including warehouses, inventory levels and reserve stocks. Non-dollar-valued impacts on the users of the system are primarily aspects of the military effectiveness of personnel and material, as affected by transportation time, reliability, vulnerability, etc. Non-dollar-valued impacts upon non-users include effects on the civilian economy (CONUS or theater) of the military use of transportation resources. The broader non-quantifiable aspects of impacts are the over-all vulnerability, reliability and political desirability of the transportation alternatives.

The strategic mobility criteria in Figure 4 summarize the major relevant impacts.

The purpose of a model is to predict the performance of specified options with regard to relevant criteria. Strategic mobility models must produce information appropriate for predicting the performance of options described above with respect to criteria concerning these impacts.

Principal VIII

A transportation system is a particular form of "market," in which supply and demand reach equilibrium within the constraining channels of the transportation network.

a. A number of "level of service" variables are necessary to define the interrelation of supply and demand.
b. The volume and composition of the demand for transportation depend upon the level of service at which transportation is supplied.
c. The "supply" of transportation, represented by the level of service provided, depends (for given resource inputs) upon the volume and composition of the demand.
d. Determining the level of service at which supply and demand are in equilibrium in a particular context is usually computationally difficult, because of the complexities of the transportation network and of the transportation demands.

Application to strategic mobility analysis

The basic "level-of-service" variables in strategic mobility are identified in Figure 11. For fixed transportation resources, the level of service, as measured in total trip time, for example, will depend upon the volume of movements through the system. For a wide range of low volumes, travel time is relatively constant, but as the level of movement through the system increases toward capacity, travel time increases rapidly—i.e., the level of service depends on demand.

Time
- Total trip time
- Reliability—frequency distribution of trip times
- Time spent at transfer points
- Actual arrival versus desired arrival
- Actual departure versus ready date
- Closure dates of personnel, equipment, and total unit

Cost
- Operating costs
- Fixed costs

Safety
- Probability of fatality
- Probability distribution of accident types

Comfort and Convenience
- Physical comfort
- Psychological comfort

Figure 11—Level-of-service variables

Usual military practice is to establish "firm" movement requirements—that is, to fix demand. The level of requirements is generally (but not always) based upon some initial broad estimate of the demand which can be satisfied at an acceptable level of service—for example, an estimate of the forces which can be deployed in a specified time period. If upon detailed analysis it is determined that the level of service is unacceptable (such as movement times too great), or alternatively that there is excess capacity in the system, then the level of requirements will be adjusted. Although demand is controlled by military planners, it *is* a function of the level of service supplied.

Given a level of demand, represented by the set of movement requirements, determining the level of service is computationally difficult. It is necessary either to do detailed routing and scheduling of movements, or to use some aggregate flow model which reflects the way routing and scheduling is assumed to be done.

Models used in strategic mobility analysis must address explicitly these "market" aspects. The models must be able to predict the various level-of-service characteristics, such as travel time, closure dates, etc., of interest to the planner, for given options for applying the resources against the requirements. The models must be able to indicate to the planner potential changes in his options which will result in a more satisfactory level of service. The models must have provisions for the planner to revise his transportation and transportation-related options to improve the level of service. Because supply and demand interact within the transportation network, models are necessary to provide satisfactory core capability for these functions.

Principle IX

Comparative analyses of alternative transportation systems policies must maintain the same set of assumed conditions. In particular, the same volume of demand must be assumed for transportation system alternatives, or else explicit correction for changes in the level of demand must be made.

Application to strategic mobility analysis

The basic implication of this principle is that assumptions must be stated explicitly, and that alternative analyses must be consistent. The key assumptions in strategic mobility are shown in Figure 12.

Readiness status	–movement requirements (units, supplies)
	–transportation resources
Network limitations	–overflight restrictions
	–canal closures
Strategic Warning Time	
Enemy actions	–hostile actions enroute
	–intra-theater
	–sabotage in CONUS
Weather	
Local populace	–labor availability
	–hospitility
	–transport capability required to support
Supply factors	–consumption rates
	–actual supply levels
Movement capability factors	–maintenance requirements
	–aircraft utilization rates
	–vehicle speeds
	–port clearance times
Reliability of transportation intelligence	

Figure 12—Strategic mobility assumptions

The implications of consistency are best illustrated by an example. Consider the problem of force programming, where the issue is which alternative level of transportation force structure to select. Assume the alternatives have significant differences in movement capability. To analyze the alternatives, the level of movement requirements must remain the same for each alternative force structure. Otherwise, evaluation of the alternative force structures must balance the cost of each alternative transportation force level against the costs and benefits of each of the different levels of movement requirements satisfied.

Principle X

Transportation is not an end in itself.

Application to strategic mobility analysis

The ultimate concern of strategic mobility is the effectiveness of the strategic response to an actual or potential aggressor. There are many alternatives to transportation, including prepositioning, changes in force requirements, changes in movement characteristics of equipment, etc. (See Principle VI). The strategic planner must

The strategic planner must continually evaluate whether transportation is in fact the most effective means of response.

More directly, this principle also means that the mobility planner must continually verify that in his attempt to achieve maximum transportation effectiveness, he does not reduce over-all strategic effectiveness. For example: maximum utilization of transportation capability may require excessive travel time for personnel; or major disparities between arrival times of personnel and equipment of units; or an excessive grouping of lift vehicles or saturation of facilities, thus increasing vulnerability to enemy actions.

B. Alternative levels of detail

In this section, we expand upon Principle I. This principle states that transportation systems consist of networks and vehicles flowing through those networks. Thus, this principle establishes the most detailed level of modelling required, in which every vehicle is moved as an element over each link of the transportation network on its route. This detailed level provides a reference point against which to discuss other levels.

The key elements in modelling a transportation system are:

a. the transportation network—links connected at nodes in a relatively complex fashion to form a network;
   (1) links—highways, sea routes, air routes, railroads, canals, etc.
   (2) nodes—may be enroute terminals—for example, an enroute air terminal; may be storage points—for example, a holding and reconsignment point or staging area; or may be transshipment points or interchange

points between modes—for example, ports of embarkation and ports of debarkation;

b. transportation vehicles—differ in their load carrying characteristics (for example, bulk load capability versus outsize load capability; cube capability versus heavy-lift capability, etc.) and in their operating characteristics (speed, range, requirement for specialized loading equipment, shallow water draft, etc.):

c. movement requirements—movement requirements consisting of personnel and cargo may vary significantly in the characteristics which determine what vehicles can carry them and how they will move through transshipment points; for example: weight, cube, pallet size, clearance dimensions, whether heavy-lift, and other characteristics of loads are extremely important in determining their impact on the transportation system.

The basic problem to be addressed in strategic mobility analysis is the assignment of resources against requirements, as described by the three-cornered triangle. From a computer-systems point of view, we can replace the requirements and resources corners of the triangle by two corresponding files of data:

a. a complete set of movement requirements, defined in detail (origin, destination, date required at destination, date ready to depart origin, movement dimensions, etc.) as appropriate for the specified analysis;

b. a file of transportation resources also defined in appropriate detail (number of vehicles by type, initial position, speed, load carrying capabilities, initial availability time; transportation network capabilities, links, terminals, etc.).

The assignment of resources against requirements can take place at a number of different levels of detail:

a. ton miles of capabilities against ton miles of requirements (in general, or broken down by channel; geographic area; passengers, short tons bulk and short tons outsize; and/or time period);

b. tons per day per channel, by gross types of movement requirement (pax, short tons bulk, short tons outsize);

c. so many vehicles per day capability over each route—this implies being able to do routing and scheduling of vehicles in an approximate manner;

d. detailed assignment of requirements against capabilities—assigning movement units to CONUS modes and vehicles, ports of embarkation, vessels and aircraft, etc., constructing a detailed, comprehensive movement schedule.

The relations between levels of detail are illustrated in Figure 13. A basic problem in designing a system for strategic mobility analysis is to establish the different levels of detail required for stating the movement requirements, for describing the movement resources, and for analyzing the relationship of requirements against resources. Obviously, the level of detail used for requirements should be approximately the same as that used for capabilities, in any particular analysis. The fundamental question is what alternative levels of detail to provide.

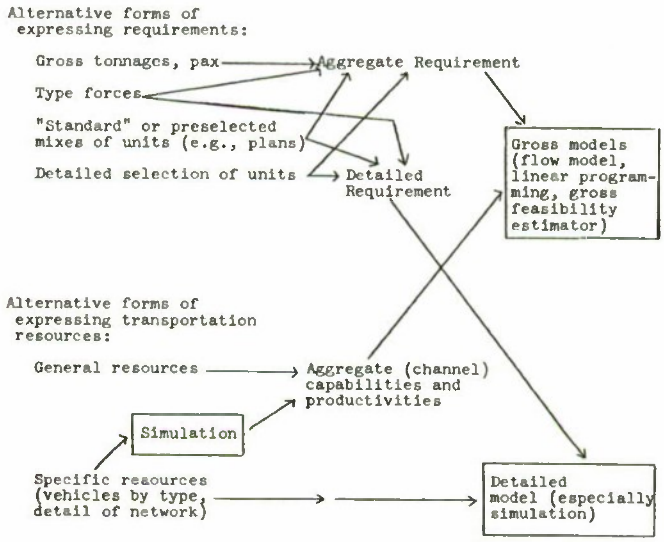


Figure 13—Conceptual relationships between aggregate and detailed levels

Corresponding to the different levels of detail in which requirements and resources files can be expressed, are different levels of models of the transportation system:

a. Total throughput capability: in this, the most aggregate level, transportation resources are represented by their total throughput capability, such as three thousand ton miles per day. The image corresponding to this model is of a pipe through which movement requirements can be passed at the indicated rate. The extent to which the indicated rate takes into account all the subtlety of transportation system elements determines the reliability of this kind of model. In particular, for highly complex networks in which a variety of flows occur more or less simultaneously, and in which the flows have diverse patterns of origins and destinations and have diverse movement characteristics, the reliability of this "throughput" model is highly questionable. On the other hand, a total throughput model may well be appropriate for such steady-state flow problems as represented by flow over a very small number of parallel, non-interacting routes for airlift to a fairly isolated theater, in which the number of vehicles available is relatively well within the capability of enroute support facilities and in which the vehicles are recycled at a steady rate.

b. Network flow model: in this type of model, the transportation system is represented by a network in which each link has a capacity, such as short tons per day. One example is the Ford-Fulkerson flow model (cf. Ford and Fulkerson, 1962). This takes into account the interacting or topological characteristics of transportation networks and so is somewhat more realistic than the gross throughput model. However, like the throughput model, it suffers from assumptions about vehicle capabilities necessary in deriving the channel capacities.

c. Vehicle recycling model: in vehicle recycling models, transportation networks are ignored or represented only as route distances and perhaps route capacities; that is, the interactions of a variety of routes using several common links are ignored. The major focus of the model is on taking a fixed number of vehicles and considering the impact of their recycle times: that is, the time it takes to move them out to the origin to pick up the load, from the origin to the destination, and back from the destination to a repositioning point for pickup of the next load. An example of this is the Airlift Capabilities and Requirements Estimator (ACRE) model of the U.S. Military Airlift Command, or the Airlift Deployment Simulator (AIDS) model of Stanford Research Institute. These models take into account the effects of vehicle availability, at the expense of ignoring network interactions.

d. Comprehensive transportation model: a comprehensive transportation model would take into account both network characteristics, and vehicle recycling and availability characteristics. Such a model might be obtained initially by relating the Ford-Fulkerson flow algorithm to explicit use of recycling models such as ACRE. However, a comprehensive model should be developed which is explicitly designed to consider the flow of discrete vehicles through transportation networks, considering both the topology or interaction characteristics of the network, and the vehicle availability and recycling aspects. Initially, such a model might make fairly gross approximations to a third aspect of the problem, the loading problem (how the detailed components of movement requirements are distributed over or loaded into the available vehicles).

We do not anticipate that such a "comprehensive" model can be developed as a single package. Rather, we expect that various forms of "comprehensive" models will be obtained by linking together components of other types of models. This is illustrated in our discussion of the relationship between optimizing and simulation models following.

Channel flow models and detailed models lie at two extremes of the spectrum of detail. For some kinds of studies, channel throughput capabilities may be a reasonable approximation to detailed networks. These channels may be obtained by aggregating links and nodes in a detailed network and estimating vehicle utilization rates and carrying capabilities to achieve a single numerical flow capacity for each channel. However, to verify such an approximation, and for detailed analysis which explicitly considers the interaction of different modes of transportation and the effects of terminal or interchange capabilities, transportation must be modelled in a way which takes into account the movements of discrete vehicles singly or in units through the detailed topological structure of the flow network.

Thus, the levels of detail in a strategic mobility analysis system will range from detailed movement scheduling models which consider discrete vehicles, to aggregate or gross flow models which consider only channel flow capacities.

C. Need for a variety of models

The overall implications of the preceding discussion are that there is no single model adequate for all strategic mobility analyses. Because of the wide variety of problem types and modelling issues, there must be a variety of models and techniques available. These models will differ in other aspects in addition to level of detail.

Other basic differences among models are:

a. the cost of using a model – in time and/or dollars;

(1) initial set-up cost;

(2) cost for each successive application;

b. accuracy or validity-probability that the model gives the right decision; reasonableness of model as representation of the real world;

c. scope – what planning options and performance measures are within the scope of the model;

d. sensitivity – degree to which results from the model vary with uncertainty in key data;

*For a theoretical treatment of the roles of models in a problem-solving process, see Manheim (1966a).

e. degree of "optimality"—extent to which model produces a solution which is optimal with respect to some criterion.

A strategic mobility analysis system must have a variety of models, differing in detail, in degree of optimality of the solution, in coverage of the problem, in level of computational cost, and in other respects. Every model has its advantages and disadvantages. These models should be available in a single system environment, with routines to allow as complete compatibility as possible such that the models all operate off the same data base. Only after extensive operational experience is acquired with a full set of models, will it be possible to really identify the relative roles of different models. Until then, the analyst must have freedom to experiment with using alternative models in a flexible environment.*

There should be one basic set of data files in a strategic mobility analysis system. That is, any routine which requires files in a non-standard format will have to be preceded by pre-processing routines which convert data from the format of the basic files to temporary files in the formats equired by these routines. This approach, while perhaps expensive in computational time, allows implementation of a flexible environment in which simulation models, linear programming, and other algorithms are used together.

The relationship of linear programming and similar optimizing algorithms to simulation models may take several forms, and illustrates the desired degree of flexibility and interaction among models.

By simulation models we mean models which *may* have partially optimizing components, for example, the assignment of loads to ships. However, a simulation model is characterized by the fact that the result is dependent upon specific values of parameters given by the planner; the result of a simulation is more a prediction of the consequences of the planner's parameter values than an optimal solution of a problem given those parameter values.

In contrast to simulation models, optimizing models such as linear programming or Ford-Fulkerson flow algorithms give a solution which is optimal within the context of the model (which may be extremely limited). For example, optimizing the distribution of flow through a transportation network to achieve maximum total throughput capability; minimizing the total cost of strategic mobility resources to meet given requirements. A simulation approach to this last example would consist of the following: given a specific fleet mix, what is its cost and does it meet the movement requirements. Thus, the simulation model would have to be used many many times, varying the fleet mix each time, to determine the least-cost fleet mix which meets the requirements.

There is a role in a strategic mobility analysis system for each type of model. Some of these roles are as follows:

a. optimizing algorithms used to identify an optimal solution at a general level;
b. simulation models used for detailed evaluation of a particular solution;
c. simulation models used to determine coefficients for input to optimizing models (See Figure 14);
d. optimizing models used as components within simulation models;
e. optimizing models used to determine a starting point for a simulation model.

This variety of potential interactions between simulation and optimizing models justifies the basic design decision in the first paragraph. With one basic set of data files and the development of appropriate data conversion routines, optimizing and simulation models can be tied together in a variety of ways to obtain quite different kinds of strategic mobility models. It is highly important that the system have this capability.

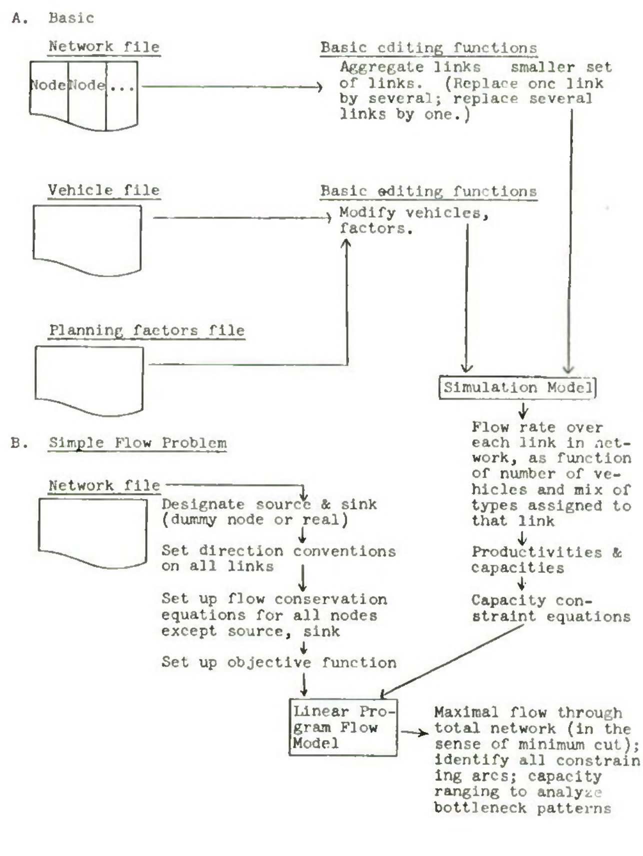


Figure 14—Relation between linear programming and simulation models

D. Implications for system design

The basic point of this discussion is that an analysis capability for strategic mobility must utilize a variety of models. A corollary argument is that the planner must be able to use these models in an interrelated fashion. To amplify:

1. models must be provided for analyzing all modes of transportation,
2. models must be provided for simulating all types of movement requirements,
3. models must be able to consider interrelations among successive modes of transportation as requirements flow from initial origin to final destination, and must specifically address the relationships between flows into and out of terminals and other mode interchange points,
4. models must be able to analyze the full set of transportation and transportation-related planning options, in all time frames from long-range to current operations,
5. models must provide information with which the planner can identify all the significant impacts of the alternatives open to him,
6. models must explicitly recognize the interaction of supply and demand in the transportation market, and must provide the planner flexible tools for exploring the levels of service which can be achieved with alternative levels and mixes of requirements,
7. models must allow explicit variation of key assumptions,
8. models must be available at different levels of detail and with different costs, emphases, and coverages of the options, so that the planner can use the model most appropriate to the analysis task,
9. models must be available in a common environment, related to the same basic data base, so that the planner can use different models in an interrelated manner.

III. The Systems Analysis Approach

A. Ideal approach

The basic theory of systems analysis is shown diagrammatically in Figure 15. Systems analysis works toward this theoretical approach as an objective. Under ideal conditions, this theory implies the sequence of analysis shown in Figure 16.

In practice, even the best systems analyses don't achieve this ideal. Nevertheless, it provides an important and useful conceptual framework (cf Hitch and McKean, 1960, particularly appendix by Enthoven; also Henderson and Quandt, 1958), and an objective to guide analyses.

The difficulties in implementing this ideal approach in practice arise for the following reasons:

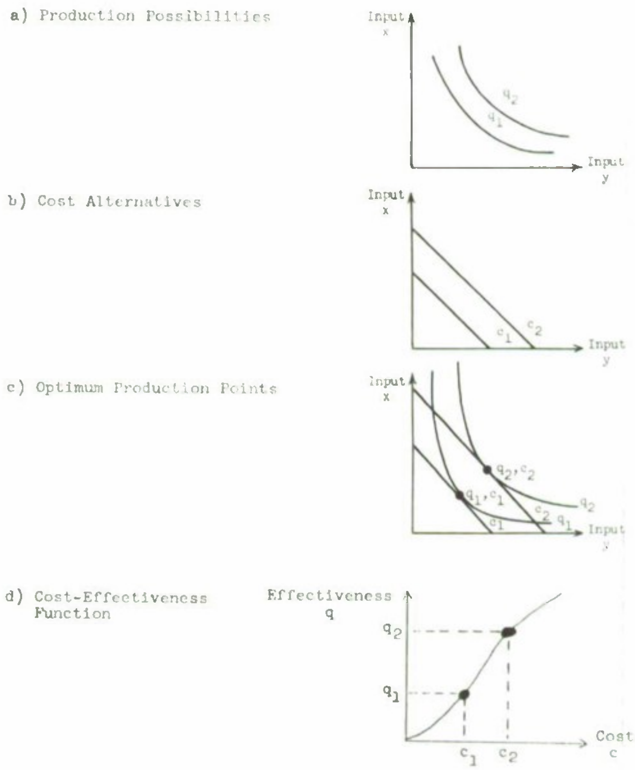


Figure 15 – Systems analysis: theory

a. *Description of space of alternatives:*

The set of all possible alternatives is not easily described in terms of a small number of continuous variables; for example, see the variety of transportation resources identified in Figure 2. Furthermore, the space of alternatives is large, and generation of a single reasonable alternative is difficult – for example, the generation of a single transportation plan which is likely to be feasible.

b. *Measures of effectiveness:*

In general, there is not a single major criterion of effectiveness of a transportation plan, as indicated in Figure 4. All of the measures shown there, and others, are potentially significant at different stages of analysis of a strategic mobility problem. It is not acceptable to assume that all of these measures can be uniformly collapsed into two measures, one of effectiveness and one of cost. Rather, several different measures must be identified separately and so carried through the analysis. Only under rather constrained conditions or with a narrow scope of alternatives can a single measure be used: for example, closure time can be used only when buildup rate, transit time, unit integrity, and other criteria are all satisfactorily met, for a balanced deployment force with resupply assumed in balance.

Define space of alternatives (x,y).

↓

Identify measure of effectiveness q, and measure of cost c.

↓

Define models or procedures to be used to determine, for any alternat've (x,y), (1) its cost, c(x,y), and (2) its effectiveness, q(x,y).

↓

For all alternatives (x,y), determine effectiveness q and cost c.

↓

Construct production possibilities curves: identify all alternatives which produce same effectiveness, for different levels of effectiveness.

Construct cost alternatives curves: identify all alternatives which have same cost, for different levels of cost.

↓

Determine optimum production points: for each level of effectiveness, find that alternative which has least cost.

↓

Construct the cost-effectiveness function, based upon the least-cost alternative for each level of effectiveness.

↓

By examining the cost-effectiveness function, select the optimum alternative as that one with greatest cost for which the marginal increase in effectiveness per unit cost is greater than the minimum acceptable rate of return.

Figure 16—Systems analysis: ideal approach

c. *Models for determining cost and effectiveness:* As we pointed out in Section III, there are a variety of models required in mobility analysis. Given a particular alternative mobility plan, determination of its costs and effectiveness is difficult, complex, and expensive. Cost and effectiveness are not simple analytical functions of the alternatives.

d. *Identification of all alternatives which have same cost or same effectiveness:* Because of the variety and complexity of the alternatives and of the models for evaluating the alternatives, exploration of the space of alternatives to identify equivalances of cost and/or effectiveness is very difficult. A realistic approach must be a "trial-and-error" approach: generate one or several alternatives, determine the corresponding costs and effectiveness, and attempt to infer from the limited set of alternatives the minimum cost alternative for each level of effectiveness. If desired, return to generate more alternatives.

This ad hoc approach is necessary while dealing with major alternatives. However, there is a useful role for highly-specialized explorations of equivalences over specific aspects of the alternatives: for example, small changes in the number or mix of aircraft applied to move the requirements; effect of alternative locations within the US on airlift requirements, in terms of additional aircraft required to meet same delivery times, or exploration of relation between US location and ready date, with airlift available held constant. There are many such partial explorations possible, and the analyst can get important insights into the overall problem through conducting such tradeoff analyses. However, to do these, the majority of variables must be held constant while those under focus are varied; and so the basic problem of finding *major* alternatives with equivalent costs and/or effectiveness must be handled in the ad hoc approach described above.

e. *Determination of cost-effectiveness function:* The cost-effectiveness function represents the set of all alternatives (x,y) such that for any alternative in this set, there is no other alternative with lower cost for the same or greater level of effectiveness, or with greater effectiveness for the same cost. This assumes that the costs and effectiveness of all alternatives are known with certainty, and so for all the reasons identified above is unrealistic. The only feasible approach in problems as complex as strategic mobility is to array the alternatives in order of increasing cost, and apply a dominance check. The dominance check is achieved by considering each alternative in turn, checking that there is no other alternative with lower cost for the same effectiveness or with greater effectiveness for the same cost. Wherever the dominance check is not satisfied, the less desirable of the two alternatives is discarded.

B) Practical approaches to systems analysis

These difficulties do not invalidate the ideal of systems analysis, provided the analysis approach is modified appropriately. The first step is to adopt the sequential process shown in Figure 17. This process emphasizes the need to generate alternatives one by one. Further, q and c may be vector-valued as appropriate, to allow for multiple measures of effectiveness. The idea of a dominance check is easily extended to multiple dimensions.

Define space of alternatives (x,y).

↓

Identify measure of effectiveness q, and measure of cost c.

↓

Define models or procedures to be used to determine, for any alternative (x,y), (1) its cost, c(x,y), and (2) its effectiveness, q(x,y).

↓

Generate an alternative (x,y).

↓

Determine the effectiveness q and cost c for the alternative.

↓

Dominance check: Compare alternative with all others previously generated. One alternative dominates another if the first has higher effectiveness for equal or lower cost than the second, or if the first has the same effectiveness but lower cost. Discard the dominated alternative.

↓

Cost-effectiveness function: Array the undominated alternatives in order of increasing cost. (Check that effectiveness increases with cost.)

↓

By examining the cost-effectiveness function, select the optimum alternative as that one with greatest cost for which the marginal increase in effectiveness per unit cost is greater than the minimum acceptable rate of return.

↓

Compare the optimum alternative found so far with the prospective payoffs of additional analysis. Decide whether to terminate analysis or to generate another alternative.

↓

Figure 17 — Systems analysis: sequential approach

The second possibility is to extend the idea of sequential analysis to multiple levels of detail.* A highly simplified example will illustrate this approach.

Consider a deployment in which only one type of aircraft and one type of ship are available. The alternatives can be expressed as: number of aircraft, allowable load per aircraft in tons per day, number of ships, and capacity per ship per day. (Assume that the load capacities have already been scaled appropriately to tons per day throughput over the specified routes.) The alternatives can also be represented in another way: throughput by air and throughput by sea (per day). Air throughput is calculated as the number of aircraft times the allowable load per day per aircraft. Sea throughput is the number of ships times capacity per day.

For a given force to be deployed, the problem is to find the cost-effectiveness function for the alternatives. The alternatives can be treated at two levels of detail. The "gross" treats alternatives defined in the two variables, air throughput and sea throughput. The "detailed" level treats alternatives defined in the four variables, number of aircraft, aircraft load, nu-ber of ships, ship load. With appropriate models, cost and effectiveness can be determined at either the gross or detailed levels.

The important point is that each gross alternative corresponds to a large number of detailed alternatives: for example, an airlift throughput capability of 1000 tons per day can be achieved by 50 planes each carrying 20 tons per day, 25 planes carrying 40 tons per day, or 40 planes carrying 25 tons per day. Therefore, the costs and effectiveness of gross alternatives can only be estimated roughly; and only for detailed alternatives can accurrate costs and effectiveness be determined. However, the models for evaluating gross alternatives may be much easier and cheaper (time, cost, data, etc.) to use than the detailed models. Therefore, the planner will want to use both levels of detail in a balanced way; he will use the gross level of analysis to identify the general shape of the cost-effectiveness function, and will use the detailed level to determine precise costs and effectiveness for detailed alternatives of most interest.

The general methodology for doing systems analysis in this "hierarchically structured" sequential manner is shown in Figure 18.

C. Implication for system design

The systems analysis ideal provides a general framework of analysis which is not realizable in practice. Design of an analysis capability must recognize that this framework is an objective, but that practical analysis of real problems will fall short of the objective. This implies that the analysis capability must:

a. allow explicit generation of large numbers of alternatives;
b. provide rapid evaluation of alternatives to eliminate quickly those which are likely to be dominated;
c. provide detailed evaluation of those alternatives remaining;
d. allow planner control of explorations of tradeoffs among small sets of variables;

*For a Bayesian Decision Theory model of this process, see Manheim (1966a).

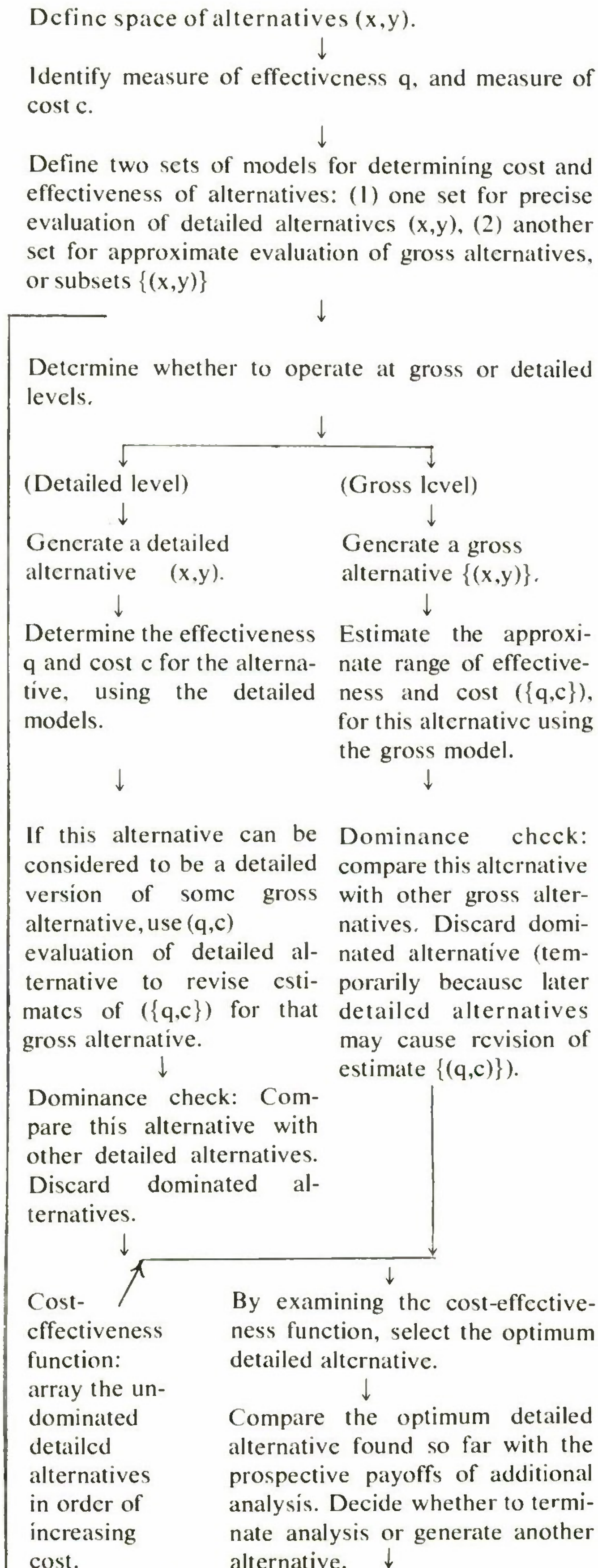


Figure 18 – Systems analysis: hierarchical structure (two-level)

e. provide "record-keeping" facilities to allow the analyst to store the results of his analysis in a way amenable to the systems analysis framework.
f. allow the planner flexibility in using models of different levels of detail at different points in the analysis process (cf. Manheim, 1967).

IV. Analysis Environment

A. Relations between planner and system

In the preceding sections, we have indicated the arguments for a highly flexible interactive relationship between the planner and the ADP system. The purpose of this section is to briefly describe how this can be achieved with current computer technology.

The analysis tasks will range over a wide spectrum in terms of their requirements on the computer system (Figure 19.) There will be many times when a planner simply wishes to initiate the running of a large computational routine, such as a linear programming or network flow model, or a complex simulation model. There will be other times when the planner will want to interact very rapidly and adaptively: selecting items out of one file and ordering them in another, requesting subtotals of various kinds, and performing a variety of manipulations of the data. In the first sense, the planner will be using the computer much as he has traditionally used it: he would approach the computer only to initiate a large processing task, and then again only to receive the results. In the latter case, the planner would be using the computer as almost a sketch pad or notebook, interacting with it as he would with a pencil and ordinary paper tablet.

The majority of planner analysis tasks will be between these two extremes of the spectrum. A typical analysis sequence will probably be a mixture of quick, highly interactive data manipulations; selection of alternative processing routines; execution of processing routines of short, medium, and occasionally significant length; the monitoring of those routines, with consequent changes in parameters as they are executed; and display of results.

The basic operations of the analyst in this system will be using programs and procedures available to formulate alternative movement plans, and to analyze those plans. As pointed out in Section II, all of the analysis tasks, whether concerned with vehicle R&D or network construction, would involve the same basic function, balancing requirements against resources. But in addition to executing this function many many times, the analyst will also wish to explore alternative movement planning procedures, such as constructing planning procedures and larger models out of component models and procedures. The planner will also be concerned with routine and semi-routine maintenance of the data base, including review and familiarization.

Data Access and Manipulations
- Review of file contents (display, print) in variety of formats
- Addition/deletion/revision of data
- Creation of new file(s) from one or more files—merge, extraction
- Change of file formats and sequences

Execution of Computations
- Set-up—designation of data files, parameter values, computational options
- Actual calculations—short bursts, long sequences
- Monitoring long runs and modification of calculations.
- Review of results

Construction of New Computational Procedures
- Linking of modules and naming of composite sequence
- Designation of conventions for data files, parameter values, computational options

Construction of Modules
- Programming
- Debugging
- Compilation
- Testing
- Acceptance into repertory

Systems Programming

Figure 19—Modes of use of the system

There will also be a need for use of the system as an education and training device: some routines in the system will be designed explicitly for programmed instruction in the use of the system.

In design and use of a system such as this, a careful balance between roles of man and machine must be maintained. Wherever possible, the planner should be forced to make explicit judgments about parameter values, criteria, etc., as opposed to building into the logic of a program a particular set of criteria and a particular scheduling approach. (However, "standard" values would be provided so that planner need operate only on "exception" basis to override "standard" values.) This serves several purposes: first, it forces the planner to think about the problem and to become involved with its details, thus trying to impress upon him that he should not treat the computer answer as dogma. Second, it assists in preventing, to the maximum extent possible, the institutionalization of the particular prejudices of one or several planners or programmers.

B) Basic technological capabilities

The basic computer capability necessary to provide the highly interactive support required is the kind of system now known as a multi-user, remote-access, "time-sharing" system. Through sharing the fundamental processing capabilities of the computer simultaneously among a large number of users, this kind of capability can provide access, through consoles at remote locations, to a large number of individuals. Each individual can utilize the computer in a highly interactive way, having the computer respond to him about as rapidly as he can absorb the information. For processing tasks which require significant amounts of computation, the user does not need to stay at his input-output device, but need only initiate the processing and return to receive the results; however, if he wishes, he may monitor the processing, review intermediate results, and make parameter changes.

These systems will have a variety of interactive capabilities, including flexible problem-oriented languages which allow the planner to utilize the computer system very rapidly. Such a computer system can approach the ideal of an electronic "writing tablet" or "sketch pad," in which the computer provides services to the user as rapidly as he can formulate his requirements.

Remote access consoles will be a teletypewriter or other similar capability, and might also have graphic displays and local capabilities for transmitting and receiving facsimile reproductions. In addition, high-speed bulk printing capability will be provided for producing large, voluminous reports and details, and permanent records of analyses.

Such technological capabilities are available and are technologically feasible for implementation at this time.*

C. Conclusions

The previous sections of this paper have established the requirement for highly flexible interactive processing, in order to effectively accomplish strategic mobility systems analysis. Such capability is available through multi-user, on-line, remote-access systems. These systems will provide a highly responsive capability, allowing the planner to explore tradeoffs; to explore alternative plans, including alternative levels of movement requirements and resources; to explore the impact of assumptions; and in general to become highly familiar with the shape of his problem, the alternatives open to him, and the issues.

With this kind of system, there is little danger that the machine will replace the planner: this kind of

*See for example articles on Project MAC and System Development Corporation time-sharing systems in the proceedings of the Second Congress on the Information System Sciences (Spiegel and Walker, 1965).

capability will force the planner to think through his problem and to address the broad issues—what assumptions he makes, what options he explores, what criteria he uses—instead of being concerned almost exclusively with clerical manipulations.

A flexible, on-line computer system for using a variety of models is feasible and practical.

V. Characteristics of a Strategic Mobility Analysis Capability

A. Variety of analysis tasks

The discussion of the strategic mobility problem indicated that there is not just one "strategic mobility problem," but that there are many different problem formulations of interest. The basic issue in strategic mobility is the balancing of resources against requirements with respect to various performance criteria. There are a wide variety of different types of transportation resources, including vehicles and networks. There is not one single criterion for effectiveness of a mobility plan applicable in all contexts, but a variety of potentially significant measures of effectiveness must be considered. The strategic mobility triangle focuses on the variety of problem types in which different elements of the triangle may be stated as given, with the remaining elements to be found by analysis.

The core of the strategic mobility triangle is movement routing and scheduling. Even when the type of analysis does not require the level of detail represented by a complete movement schedule, routing and scheduling issues are still involved. This is particularly important in understanding the relationships among different types of transportation models. Typically, strategic planning problems are divided by time frames into long-range, mid-range, contingency and operational planning, and current operations. The triangle of resources—requirements—criteria is valid for each time frame. Other distinctions of problem types can be made, using the triangle as a base, and corresponding to analysis functions.

The discussion of strategic mobility problems indicates conclusively that there are in fact a variety of problem types.

B. Variety of models

Next, we discussed models required to do strategic mobility analysis. We enumerated basic principles of transport systems analysis and showed their application to strategic mobility.

These principles had a number of specific implications for modeling strategic transportation. In particular, we discussed in detail the relationships between alternative levels of strategic transportation models, ranging from detailed vehicle and network models to gross flow models. We identified possible relationships between these models. The basic and most important conclusion of this argument is that there is no single model which is *the* model to use in all strategic mobility analyses.

C. Systems analysis as a framework

The basic theory of what has come to be known as systems analysis was discussed next. The sequence of analysis implied by this theory was shown to be an ideal which is rarely achieved in practice. A more realistic sequential approach was identified which preserved the essential characteristics of systems analysis, while recognizing the realities of limitations of mobility analysis resources and tools. The systems analysis ideal provides a useful objective for design of an analysis capability, but the analysis capability must recognize that the ideal will not be achievable in practice. A variety of capabilities are identified to provide the analyst the capability for striving toward the systems analysis ideal as effectively as possible. Essentially, in conjunction with the recognition of a variety of models for strategic mobility analysis, the objective of systems analysis argues for a highly flexible analysis capability, in which the planner can modify his analysis in many ways as he explores the scope of his problem.

D. Flexible analysis capability

Discussion of the analysis environment began with recognition of the variety of modes of utilization of an analysis capability by the analyst. The basic technological capabilities which can be provided in a highly interactive system were identified. Essentially, the idea of multi-user, remote-access, "time-sharing" systems is that they provide a large number of individuals access from their desks to powerful, sophisticated computing capabilities. This environment will allow the planner to focus on understanding a particular mobility problem and exploring the issues of that problem, freeing him from burdensome clerical tasks and allowing him to exercise to the maximum extent possible his experience and judgment.

E. Conclusions

We conclude that a strategic mobility analysis capability must be capable of addressing a variety of analysis tasks, must have a variety of models available to be applied to the tasks as the analyst chooses, must provide appropriate tools for organizing an analysis within the systems analysis framework, and must be accessible to the planner in an interactive, responsive, flexible way through multi-user, on-line, remote-access computing capability.

These implications identify a number of major tasks areas to be addressed in constructing such a capability. First, and most obviously, the initial outlines of the required strategic mobility models must be established, and appropriate data collection and model verification initiated. Second, priorities must be established and models implemented. Third, analysts with extensive military experience and/or extensive systems analysis experience must be brought into the system development process to assure development of appropriate models, and to be able to make effective use of those models. Fourth, the appropriate type of computer environment must be implemented as rapidly as possible. Fifth, basic research must be initiated: to develop fundamental understanding of strategic mobility as a function, to develop new kinds of strategic mobility models and analysis techniques, and to develop theoretical and practical understanding of the effective use of a comprehensive and flexible analysis capability.

F. Closing comment

The new user of the computer, as well as the neophyte systems analyst, tends to have an over-simple view of how a computer is used in analysis. In this view, the sequence of operations is linear: assemble the data, feed the data into the program, and summarize the computer output.

In reality, analysis of problems as complex as in strategic mobility involves many different sequences of analysis, and many different uses of the computer. Problem analysis is a process—a process in which additional information is acquired as gaps and errors are discovered in the data base, a process in which models are revalidated and revised, and a process in which the analyst is continually learning more about the shape of the issues.

Strategic mobility analysis must be seen as a process. The design of an analysis capability must address this explicitly.

ACKNOWLEDGMENTS

Important contributions to the ideas presented here have been made by Dr. Joseph H. Stafford of M.I.T. and Mr. Kenneth F. Mineau, Defense Communications Agency, while both were associated with the author at Project STRATMAS, (Strategic Mobility Planning and Analysis System); Colonel Richard C. Rantz, of the Office of the Special Assistant to the Joint Chiefs of Staff for Strategic Mobility (SASM); Defense Communications Agency and its contractors through the Plan for Technical Support to SASM; and Dr. Paul O. Roberts, Harvard University. Many of the basic arguments have grown out of research in transportation planning by the author while at the Department of Civil Engineering, M.I.T., and have been influenced by current research at M.I.T., especially on the Integrated Civil Engineering System (ICES) and on Project MAC (Multi-Access Computer/Machine-Aided Cognition).

This paper was initially presented at the NATO Advisory Panel on Operations Research "Conference on the Analysis of Military Systems of Transportation," Oxford, England in July 1966. Minor revisions to the paper were made under a research project at MIT supported by the Special Assistant to the Joint Chiefs of Staff for Strategic Mobility through Defense Communications Agency, Contract DCA -100-C-67-0008. Of course, all statements made herein are the opinions of the author and do not necessarily reflect the policy of any sponsoring agency or institution.

REFERENCES

1 L. R. FORD JR. and D. R. FULKERSON *Flows in networks*. Princeton University Press 1962.

2 J. M. HENDERSON and R.E. QUANDT Microeconomic theory: a mathematical approach. McGraw-Hill, New York, 1958.

3 C. HITCH and R. N. McKEAN *The economics of defense in the nuclear age*. Harvard University Press, Cambridge, Mass. 1960.

4 M. L. MANHEIM *Hierarchical structure: a model of planning and design processes*, M.I.T. Press, Cambridge, Mass. 1966 (a)

5 M. L. MANHEIM *Principles of transport systems analysis*. In Papers, Seventh Annual Meeting, Transport Research Forum, 1966. (b) Also reprinted in Highway Research Record. Highway Research Board, Washington, D.C. (in press).

6 M. L. MANHEIM *Problem solving processes in planning and design*. Professional Paper P67-3, Civil Engineering Systems Laboratory, Massachusetts Institute of Technology, Cambridge, Mass., 1967.

7 J. SPIEGEL and D. E. WALKER Eds. *Information system sciences: proceedings of the second congress* Spartan Books, Washington, D.C. 1965.

ORGANIZATION FOR THE DESIGN OF MILITARY INFORMATION SYSTEMS

NORMAN WAKS, *Chairman*

Session Participants:

JORDAN J. BARUCH
DONALD C. CARROLL
ZENON S. ZANNETOS
JAMES L. MCKENNEY
FRANK R. ELDRIDGE, JR.
HERBERT D. BENINGTON
JOHN A. BECKETT

Introductory Remarks

NORMAN WAKS
The MITRE Corporation
Bedford, Massachusetts

Three design dimensions

The designer of a military information system has at least three dimensions to his job whose polarities he should consciously appreciate. The purpose of this 3rd Congress session of the panel is to supplement the material provided by a similar panel at the 2nd Congress by illustrating and illuminating these dimensions. Each of the three dimensions will be discussed in turn. Each one, in a sense, is embrasive of those that follow it.

The first dimension

The first and most basic of these dimensions is the intended intelligence content of the information system being designed. The "intelligence" dimension is meant here to be that dimension along which the degree to which human beings are informed, or provided with knowledge/intelligence, by the system is measured. The scale involved is a function of human utility, and may even be negative (i.e., have dis-utility to people). Thus we may distinguish between those information systems on one end of the spectrum of the general class of information systems which have little direct human value and the others. The former might be thought of simply as "data" systems, or systems in which the intelligence content of what is being transmitted is neutral or low in human terms. problems for their designers. Eldridge's paper is, in fact, an example of the serious problems one can run into in designing such a system. In particular, he attempts to illustrate the difficulties one encounters in performing the systems analysis related to such design effort.

Carroll and Zannetos on the other hand, attempt to show some of the difficulties of designing what might be classified as "highly intelligent" information systems. They also try to show conceptually at what point in the spectrum of information systems such a system may first be classified as an "intelligent" system in the first place.

The second dimension

A second scale along which a military information system designer might usefully classify his efforts in an explicit way is in terms of available conceptual and technological state-of-the-art. For military information systems on one end of the spectrum, like weapon or other vehicle control systems, the concepts and the technology may be considered to be available, for all practical purposes. What needs to be appreciated here is only the massive systems engineering effort that must be organized to acquire such systems.*

On the other end of the spectrum, however, are the so-called "command" or "command control" types of military information systems. Here we have a type of system whose available conceptual and technological inventory is low, undoubtedly because they so involve man and his higher order mental processes as a basic part of the system. Significantly, Baruch's

*See D. Israel's paper "System Engineering Experience with Automated Command & Control Systems," elsewhere in this volume, for a detailed treatment of the needed systems engineering effort.

paper, on an entirely different field than the military —*hospital* information systems—amply illustrates that the problem is not peculiar to *military* information systems at this end of the spectrum. Indeed, as he and the literature illustrate, military information systems of this type share quite common needs for concepts and technology with "information" systems being provided to serve a number of other environments or professions like medicine, law, and education.

The third dimension

This last leads to the third dimension by which information systems as a class might be measured, with consequent significance to designers. And that is the purpose, in professional or management terms, intended to be accomplished by the system. Clearly there is a difference between the approach that must be taken to designing a management information system for strategic planning purposes and that which is being provided for various types of control purposes.

McKenney's paper is a detailed case example of an attempt to create the former type of system, focusing on the simulation model involved. The Carroll/Zannetos paper develops some of the designer's concerns when attempting to provide "operating process control" and "planning process control" information systems.

SUMMARY CONCLUSIONS

By way of introduction and synthesis of the individual papers that follow, selected basic conclusions reached by each of their authors are set down here for readers' comparison. No attempt has been made to edit these conclusions, except to list them in descending order of specificity of the problem they address: from the very general to the very pointed.

"We conclude from our brief review that there exists no publicized comprehensive realization of intelligent management information systems."

Carroll and Zannetos

"In total, many of the bits and pieces from which higher level intelligence in management information systems can be fabricated exist. The problem is to assemble these within one organization."

Carroll and Zannetos

"In conclusion, we believe that increased organizational intelligence is possible and greatly to be facilitated by new advances in information technology. Perhaps the greatest progress can be made, however, by simply recognizing that increasing intelligence is a legitimate goal for information systems design, and that there are some straightforward steps which can be taken towards that goal."

Carroll and Zannetos

"The professional-level dedicated system differs both in degree and kind from the general system. The involvement in the user's problem it demands of the system designer represents both a challenge and an opportunity. It is our belief that there are so many professional areas that need such help: education, the law, architecture, libraries, etc., that the more generalization we can do from our specific observations the better off all of society will be. What is obvious in one specialty may well not be in another."

Baruch

"While our past five years of experience have been associated with one hospital within the medical profession, we believe that the four outstanding requirements of which we have become aware are generalizable to other professional level dedicated systems. We have found that our system must be:

1. Responsive
2. Reliable
3. Unobtrusive
4. Modifiable"

Baruch

"Better models are the result of a joint development effort of the individuals who are to use the model as a tool (the planners) and the individuals who are designing and programming it (the modelers)."

McKenney

"The most appropriate . . . reference base . . . for the initial stages of model development . . . is the planners' definition of the pertinent influences in the environment and how he thinks they relate to each other."

McKenney

"This improvement (in the modeling) process seems to be one of continuous redefinition of the panners' concept of the pertinent forces in the environment and growth in the modelers' ability to adequately represent these forces."

McKenney

"An impediment to more adequate rapport between modeler and planner is the state of present computer languages. The language of the program for the model has to be interpreted to the planner. This interpretation creates ambiguities and misunderstandings which limit the effectiveness of present simulations as a tool for most planners. Hopefully new computer languages will allow the modeler and planner to conceptualize the simulation model in the language it is to be programmed."

McKenney

"I will define information systems, here, as any network of facilities that acquires and/or processes data for use in controlling resources. In this sense an information system can be either purely automatic or it can contain manual elements for monitoring, display, control, or other command functions.

"Associated with every decision are both ponderable issues and imponderable ones. The analyst should concentrate on the ponderable ones. The decision maker must consider both.

"There are several alternative approaches open to the systems analyst who, once having identified the principal issues, undertakes to address them. He can collect and analyze raw data or he can build models of the system and synthesize data that will serve to answer the questions or he can do both. The analyst is advised to look at both the hardware and the software, or (the) operational aspects, of the system, in order to develop a balanced context for his study. He should relate the objectives and the missions of a support system, such as telecommunications, to the objectives and missions of the system that it supports.

"The choice of suitable measures of effectiveness is a critical part of the analysis. He should be aware of what the system can do as well as what it should do. By choosing measures of effectiveness that are related to the missions of the system supported he can avoid selecting effectiveness criteria that relate to less important side issues.

"Another critical function of the analyst is the definition, for the decision maker, of suitable alternative systems and methods of operation. The analyst should consider a large enough study context to allow for development of feasible alternatives and trade-offs and to avoid unsuitable sub-optimal decisions. For instance, different time phasings of any one program are important alternatives to be considered. However, the cost and effectiveness of alternative programs are also usually very relevant.

"The analyst should call the decision maker's attention to uncertainties in his study and should, where possible, provide him with sensitivity analyses for important but uncertain phasing, effectiveness and cost factors.

"Finally, he should, where possible, indicate which factors cannot be analyzed quantitatively within the time frame of the study or the context of the data available and, therefore, should be left to the judgment of the decision maker."

Eldridge

A medical information system: some general observations*

by JORDAN J. BARUCH
MEDINET
Watertown, Massachusetts

INTRODUCTION

Since 1962 an intensive research effort has been under way on a collaborative basis between computer technologists at Bolt Beranek and Newman Inc. and hospital personnel at the Massachusetts General Hospital. Under this project, a PDP-1 digital computer has been operated as a dedicated system under a Time-Sharing Executive to provide multiple terminal access at several experimental locations in the hospital. During these past four years, various hospital personnel with a wide range of training have operated the terminals experimentally in such applications as admissions, bed occupancy, the ordering, administration and control of medications and laboratory test reporting.

A large library of programs which fluctuates in size and composition is on easy call from the hospital terminals and permits the various research teams under the direction of Dr. G. Octo Barnett to experiment with and evaluate the use of the terminals, programs and the computer lying behind them in typical hospital procedures.

The computer equipment is located approximately seven miles from the hospital and all connections to it are made by telegraph-grade land lines. The types of programs implemented, the equipment used for their implementation and much of the hospital reaction has already been reported either in the professional literature or in technical reports of the project, a list of which is included at the end of this paper. The four years have, however, made clear the existence of certain needs and desiderata which a medical computer center must fill if it is to serve as a useful dedicated system. It is hoped that the nature of these needs and some of the steps presently being taken to meet those needs on a practical scale may be of use to those readers who are concerned with the implementation of other similar systems.

*The work reported here was done in part while the author was at Bolt Beranek and Newman Inc., Cambridge, Massachusetts. That work was performed under Contract PH43-62-850, and Grant #GM 00263-01 with the National Institutes of Health, U.S.P.H.S. and under a grant from the American Hospital Association.

A matter of definition

In the preceding section we used the words "dedicated system." Such a system, in our case, is dedicated to the service of professionals engaged in the practice of their procession. As such, it has many of the characteristics of an information system used for military command and control, for legal analysis and research or for design. A dedicated system has certain characteristics which distinguish it from the general purpose information system.

Some of the distinguishing characteristics are:

- Users of the dedicated system may generally be assumed to be unfamiliar with—and indifferent to—the operation of a computer.
- While the general purpose system concentrates on supplying computational power, the dedicated system serves more as a data repository-communication system. As a result, data input, retrieval, description and dissemination become major system functions with secondary attention paid to raw computational power.
- The dedicated system serves a reasonably well-defined population dealing with a generally restricted vocabulary. Further, the problem areas are relatively circumscribed at the initial stages of a dedicated system.

While these distinctions may, at first blush, appear to be designers' advantages, in fact they imply certain responsibilities if the long-term needs of the user group are to be met.

In an effort to generalize from the specific case of the medical computer system, let us examine some of the needs which, to us, seem to be characteristic of a dedicated system when used at the professional level. These needs may or may not be self-evident. To us—at the start—their importance was not fully appreciated. They have, however, become the most important guides we have in the second-pass system now under construction.

Needs and desiderata

While our past five years of experience have been associated with one hospital within the medical profession, we believe that the four outstanding requirements of which we have become aware are generalizable to other professional—level dedicated systems. We have found that our system must be:

1. Responsive
2. Reliable
3. Unobstrusive
4. Modifiable

Let us examine each of these needs in some detail. In our section on current approach we will discuss how to meet them.

Responsiveness

A professional using an information manipulation system is generally using it as an adjunct to his own thoughts and decision-making processes. As such, it is important that there be a reasonable match between the system time constant and the time constants of the individual engaged in these processes. Further, the user's satisfaction with the responsiveness of the system is often conditioned by his expectations. These expectations are based on his past experience and his estimate of the "work" involved in what the machine is doing. These two aspects control responsiveness-in-time.

A simple example will illustrate time responsiveness. The nurse or physician engaged in a patient care situation requires answers in a time measured in seconds; the researcher engaged in file retrieval exercises requires his answers in minutes or hours. Systems designed to respond to the researcher much faster than this represent a wasted asset. The "mulling time" required in his overall task is such that the total elapsed time from start of a problem solution to its end will not be markedly decreased by increasing the response speed of the information processing system alone. If, on the other hand, the response time is much slower than the individual's "mulling time" we have the situation of the individual waiting for the machine, losing his train of thought, losing interest in the problem or bypassing the direct inquiry use of the on-line system.

This avoidance of direct inquiry is most clearly demonstrated in the ordinary batch processing system. Here the user commonly asks for many, many tables at any one run. He then modifies his inquiry behavior from inquiring of the data to inquiring of this mass of tables. Time responsiveness avoids this form of behavior.

It is also interesting to note that if a user has grown accustomed to half-second response in an operation he does frequently, then an occasional three-second response is virtually intolerable. We believe, although we cannot yet prove it, that if the three-second response had been there initially, no adverse reaction would have been noted.

A further characteristic of a responsive system is its participation in the activity and education of the user. Responsive in time, the system must also be responsive in kind. Physicians, nurses and researchers must all be assumed to be untrained in the use of sophisticated informational systems; the computer must, therefore, be programmed to participate in their guidance. A responsive system makes the entry of data and the phrasing of analysis requests a simple and nearly foolproof procedure. Further, error correction, procedural inquiry, and mind-changing are all to be facilitated in a truly responsive system.

Consider the data input or capture problem. We can start off by saying that the general format will be a question-answer program, with the machine asking a question and the user supplying an answer. Such a technique is analagous to the pre-printed form. For a system to be responsive, however, we require that it be better than a form. Thus, for example, we will only ask those questions that have a high probability of being pertinent. In a simple "name-address-phone number" program we will ask all the questions, whereas in a more complex program (a situation assessment, for example) we may only ask a small percentage of the total.

Further to its responsiveness, the system must be able to answer questions about itself and its expectations during the above procedure. For example, what format is required for this answer, what did I answer earlier, etc.

Lastly, in our example, the system must provide for the frailty of the human operator. The user must be able to correct his mistakes as he goes along. In general, these corrections will imply new questions to be answered and new places for errors to occur.

While we have here considered only data input, responsiveness in kind must also be evident in the inquiry programs. More will be said about this, however, under modifiability.

Reliable

Outside of the military sphere, the medical community probably has the highest generally accepted requirement on system reliability of any group of users. Failure in a hospital or other medical setting is constantly guarded against and every effort is taken to ensure an ongoing reliable environment. The introduction of automated information handling must in no way reduce the reliability of that environment.

It is interesting to note that operating the computer system in parallel with the older manual systems during early experimental phases seems to be a snare and a delusion. In a dedicated system, which generally implies a stressed system, the introduction of a simpler technique in parallel with the standard techniques rapidly results in the standard techniques being ignored and atrophying. Cautions about the possibility of failure go unheeded if the failure probability is reasonably low. If the failure probability is high, the system will not even be amenable to valid experimentation. We have learned, therefore, that reliability must be built in even at the early experimental system level and that, in an unstructured community, one cannot rely on parallel operation as a safeguard in the early stages of activity.

From our experience we have concluded that system reliability in the medical community must provide for several levels of failure, leading to the term "fail-soft" rather than "fail-safe." We recognize that the level of reliability provided represents a balance between the cost of failure and the cost of providing that level of reliability. In the medical as in the military professions, it is generally terribly difficult to assess the cost of failure; these decisions are thus generally made heuristically at the command level.

One special word about reliability must be entered here and that is the reliability and preservation of the data base itself. The professional-level dedicated system has, as a major advantage, the provision of a communal shared data-base. Much of its strength comes from that common store of information. Representing a major advantage, that data base also represents a major responsibility. One finds that it is most essential to preserve the data base regardless of the kinds of failure that may take place within the system hardware or software. Indeed, I estimate that the social importance of such data bases will eventually lead to major legislation in this area.

Unobstrusive

A profession engaged in the practice of its profession looks for aid in an existing environmental context. Having evolved over the years, there is a high probability that the operating system of that profession is reasonably efficient if not optimum, but lags its technology. The introduction of a major technological change such as an information-handling system will imply a potential for change in some of the operating characteristics of the profession. The discontinuity caused by its introduction must, however, be minimized if the system is to be both effective and accepted. We have found it to be essential that the newly introduced system impose no requirement for the reorganization of the operational environment. The new system must appear to be a natural extension to already existing information handling functions.

Ideally, an unobstrusive information system is one that produces no marked change in immediate function but a marked change in both ultimate capability and the rate at which the operating environment approaches that capability. It is our current hypothesis that users are sensitive to changes in form but not to changes in the first time derivative of form.

Modifiable

It is our belief that the most important single characteristic of an information-handling system in a profession is its modifiability by the practitioners of that profession. It is our major thesis that the introduction of an information-handling system in a profession can increase that profession's capability for growth tremendously. As a corollary, we believe that the introduction of a rigid information-handling system can cripple that growth, making the profession's potential completely unrealizable. Casting the information patterns of a profession in concrete, providing rigid "hard wired" communication paths or trying to anticipate all of the information requirements for the profession represent tremendous dangers. We consider that the information environment must provide a nurturing effect and hence, must facilitate change rather than restrict it.

In reviewing the required kinds of modification, it has become clear to us that modifications should not be intercessionist. One does not wish to create a "priesthood" of programmers or the ilk through which all changes in the information-handling of the profession must pass. Rather, we would like the practitioner himself to be able to experiment with the configuration of his environment, to change the kinds of information it contains, the definitions and the flow patterns of that information. By providing such experimental capability with minimal learning on the part of the practitioner, we will encourage the user to recognize new needs and try to develop new scopes for his dedicated system.

Current approach

The MEDINET Department of General Electric Company is currently engaged in the design and construction of a large system for use by the medical community. The new system aims at serving many organizations that are widely diverse. Let us examine that system for the design approaches being taken to meet the four needs that we have just discussed.

Responsiveness

The current system is being designed around a relatively standard Time-Shared digital computer with three memory elements in a hierarchical structure. As is the case with many other such systems, core is used for running programs, a high-speed drum is used for the exchange of pages and a large disc is used for the bulk storage of libraries, data, dictionaries and so forth. Provision is made for removing patients from active status to inactive status by transferring their records to tape and for making such transfers reversible.

Each program on the library is to have one or more queuing codes incorporated within it to control the minimum service level of that program during busy running periods. The resulting priority chain will thus be a combination of first-come, first-served modified by minimum levels of serviceability associated with each program. Such a system provides a guarantee that no low-urgency program will detract from a high-urgency program.

To provide for high-speed response during the data input phase where the user's speed expectation is high, peripheral computers are used for message control. To provide for stability of response in order that the user's expectation not be built up during periods of low system use, an experimental initial artificial delay has been built into the system to prevent responses shorter than 2½ seconds. Early use of the system will indicate whether this response time is appropriate.

In the area of kind-responsiveness, we are fundamentally concerned with facile data capture. In order for the system to be both responsive and unobstrusive, it is necessary that the entry of data be matched to the professional level of those entering it. Broadly speaking, we deal with two kinds of data entry, extracts from an extensive vocabulary and extracts from a limited vocabulary. In the case of the extensive vocabulary, as encountered in making comments, random notes, and free text, we provide a keyboard entry device. Such a device may be used as an adjunct to dictation or transcription or may be used directly in the operating situation.

By far the greatest quantity of data input in the medical environment, however, is of the limited-vocabulary type. For this purpose a densely coded entry device is provided on each terminal. Operating in conjunction with tables stored in memory and with overlays, a multi-slide projector, and typescripts, the Datacoder permits entry of phrases, expressions and other data macros at a rate exceeding one per second. The vocabulary represented by either a small collection of overlays or a small set of slides is so very large as to permit major branching input programs with very few inapplicable choices.

This kind of system that replaces a check sheet or printed form with the presentation of alternatives whose pertinence to the situation is determined by the program permits the logical entry of large masses of data in a very short amount of time.

Reliability

In order to provide for gross reliability, the computer center, designed to serve a group of hospitals and other users in the medical community, will contain two complete computer systems. The present system design, being restricted to state-of-the-art techniques, calls for reliance on users and operators for malfunction detection. Correction of such malfunction will be made by switching to the standby system while the malfunction is being tracked down. Although the status of running programs may be lost at such a time, they can generally be reinstated by a small amount of user repetition.

In the case of hardware malfunction, such a shutdown and transfer procedure will be generally both necessary and sufficient. In the case of a software malfunction, the ability to resume operation while having a completely frozen total system available for fault analysis makes the correction of software problems a great deal easier than would be the case were only one system available. In order to protect against failure because of power loss, each center is to contain a two-stage standby power system capable of taking over with zero loss of operating time should commercial power fail.

Should any major catastrophic failure occur resulting in either loss of communications, loss of hospital power or failure of both of the computers, we must still ensure that the medical community can function. To provide such assurance in the hospital, we rely on the printed word. Hard copy is generated at each operating terminal in sufficient quantity and with sufficient timeliness to provide the necessary data for patient care and organizational operation. Thus, for example, drug lists, lab reports and other clock-type retrieval programs generate print-outs rather than just scope displays.

Such means should yield the desired fail-soft operation. A typical machine or program failure may cause a one-to-five minute loss of service. A more severe problem may cause a gradual degradation because of a need to rely on print-outs and manual techniques for an hour or more. At no time, however, are we in more danger from information-system failure than from hospital failure.

As mentioned earlier, preservation of the data base is a prime consideration in the system design. To this end, the information on the master disc file is duplicated elsewhere at the computer center. As each entry is made on the running file, the location of that entry and its contents are recorded on non-reversible magnetic tape. The collection of such tapes from time zero would then permit a very lengthy reconstitution of the file were it ever accidently wiped out. Naturally, such an update is impractical. At reasonable periods, the tapes are used to update a second or standby disc. Upon completion of this update, the standby disc is dumped onto tape. The updated standby disc and any tapes subsequent to the last update thus form a complete file. In the case of memory destruct commands from the computer, or a degradation in the accuracy of what is being written that goes undetected for some time, the audit tapes can be edited before updating the standby disc.

There are, of course, many more reliability steps that must be taken in software, hardware and procedures. The above examples (the most interesting to this writer) serve, however, to illustrate their level of rquirement.

Unobtrusiveness

In order that the system be as unobtrusive as possible when first introduced, the initial programs being written for user hospitals are designed essentially to implement those information-handling processes currently in use. This decision causes great strain on the staff since it calls for postponing many significant and obvious improvements. In order to allow users to use words familiar to them rather than learn new ones, we have investigated the language differences among hospitals. We find that the major differences among hospitals are formal rather than substantive ones. We are thus endeavoring to include large multiply overlapped dictionaries in the system. These will permit a high degree of data sharing as the use of the system develops.

Contrary to our earlier expectations, the reasonable introduction of a dedicated system into the medical community appears to be readily acceptable by the practitioners of that profession. By making the system unobtrusive, we are trying to insure that "management" will not use "the machine" as an excuse to bring about other changes that it feels necessary or desirable. There seems no more likely a way of guaranteeing the rejection of a system than by introducing a simultaneous set of procedural changes with the comment, "It has to be done this way because of the machine. . . ." Despite the fact that the system is being made initially unobstrusive, extensive literature is being prepared to provide instruction for the medical community in the system modification. Knowledge of this modifiability stresses the system's subservient role and makes it more acceptable as well as more useful.

Modifiability

In order to permit the members of the medical community to modify the programs that they use, we need to express those programs and store them in an easily understandable language. The rules concerning who may change programs—and what ones they may change—must be specified *ex-system*. They are far too complex in any real situation to permit of simple algorithmic statement.

While it is currently fashionable to seek for a "Natural English Language" form of programming, particularly in inquiry work, as a general cure to programming ills, we have taken a somewhat more realizable approach that recognizes the special nature of our users. The RAND Corporation's experience with the JOSS language and BBN's experience with TELCOMP, a JOSS derivative, has led us to tackle the design and implementation of FILECOMP, a JOSS-like language suitable to medicine, incorporating file manipulation capabilities and an expandable library of system verbs.

Every general program on the stored library will be carried in the FILECOMP language and may be called in that form for modification. Aside from the portions of the program that are incorporated in order to provide data security, the user will be able to control format, program logic, branching, generated communications and inter-program communication to any degree he may desire. He will, of course, also be able to generate completely new programs.

Since a FILECOMP Interpreter is available during such a modification, the user may experiment until he has a program running to his liking. Once the user has a program that satisfies him, he will be able to add it to the library. From then on, he and others who have access to that private library may call it in "run mode" just as they would call any other library program.

Frequently, programs thus modified may be of more general public use. In such cases they may—with the author's approval—be transferred from a private library to the public library. Such a transfer will be accompanied by a rigid quality control procedure that includes testing, evaluation and documentation. It is our hypothesis that medical personnel will not only use this modification capability, but that they will take advantage of the wide public library audience and use it as a form of professional publication. Needless to say, the requirement for modifiability extends not only to programs but to the data, the procedural tables, dictionaries, etc. that go to make up the overall system.

CONCLUSION

While this author's understanding of the epistemological basis for system theory is too limited either to allow him to generalize rigorously from his restricted experience or to describe that restricted experience in scientific terms, it is hoped that the four needs uncovered in our work in the medical community (obvious as those needs may be) and the approach we are taking to cope with those needs may serve as useful inputs to some readers.

The professional-level dedicated system differs both in degree and kind from the general system. The involvement in the user's problem it demands of the system designer represents both a challenge and an opportunity. It is our belief that there are so many professional areas that need such help: education, the law, architecture, libraries, etc., that the more generalization we can do from our specific observations the better off all of society will be. What is obvious in one specialty may well not be in another.

REFERENCES

1 DR. G. O. BARNETT and J. J. HUGHES *The computer in the hospital.* In The New York Times 23 May 1965

2 DR. G. O. BARNETT *Real-time shared on-line digital computer operations.* Journal of Chronic Diseases **10**, 377-386 1966

3 J. J. BARUCH *Progress in programming for processing English language medical records.* In Whipple, H. E. (Ed.), Computation for Cardiovascular Research New York Academy of Sciences Vol. 126, Art. 2 795-804 1965

4 J. J. BARUCH *Computer-based medication system.* To be published in HOSPITALS—Journal of American Hospital Association

5 J. J. HUGHES *The BBN-MGH hospital computer project.* Datamation II No. 12 December 1965

6 J. J. BARUCH *Potential implementation.* In Sass, M. A. (Ed.), Computer Augmentation of Human Reasoning Spartan Books, Washington, D.C. Ch. 9 1965

7 J. J. BARUCH *Hospital automation via computer time-sharing.* In Stacey, R. and Waxman, B. (Eds.), Computers in Biomedical Research Academic Press, 12, 291-312 1965

8 J. J. BARUCH *Information system applications.* In Candra, C. A. (Ed.) Annual Review of Information Science and Technology Vol. 1 Wiley & Sons, New York 1966

9 S. BOILEN, R. A. BOLT, and J. J. HUGHES *An interpretive communications system in the hospital.* In DECUS Proceedings 1964

10 DR. G. O. BARNETT *Hospital computer project* Memorandum Number Six-A, Hardware Status Report Bolt Beranek and Newman Inc. Report No. 1217 January 1965

11 DR. G. O. BARNETT *Hospital computer project* Status Report 1965 Memorandum Eight Massachusetts General Hospital Report revised March 1965

12 DR. G. O. BARNETT *Hospital computer project* Progress Report, Memorandum Nine Massachusetts General Hospital Report revised February 1966

13 DR. G. O. BARNETT *Hospital computer project* Software Status Report, Memorandum Six-B Bolt Beranek and Newman Inc. 1966

14 E. M. WHITE *Hospital computer project* Mid-Year Progress Report, Memorandum 4 Bolt Beranek and Newman Inc. Report No. 975 February 1963

15 E. M. WHITE *Hospital computer project* Memorandum Five Bolt Beranek and Newman Inc. Report No. 1075 December 1963

Toward the realization of intelligent management information systems

by Donald C. Carroll
Massachusetts Institute of Technology
Cambridge Massachusetts
MITRE Corporation
Bedford, Massachusetts
and
Zenon S. Zannetos
Massachusetts Institute of Technology
Cambridge, Massachusetts

INTRODUCTION

Statement of purpose

In reviewing existing statements on what a management information system should provide, we have noted a singular lack of operationally viable goals. "To provide a basis for better decision making" simply does not provide a basis for choice for the system designer. It is to help fill this void that we are motivated. Consequently, our purpose is to propose some new doctrine for management information system design, to state some explicit goals to be sought, and, in so doing, to offer some new perspectives for designers.

What we propose is an information system which aids in continually increasing management's understanding of its environment and in improving management's logic or rationality in dealing with that environment. In short, we are suggesting information systems which will allow management organizations to exhibit more intelligent behavior.

Let us stress that it is the *organization* which behaves intelligently. We are *not* proposing an artificially intelligent machine system, but rather men and machines collectively and cooperatively exhibiting intelligent behavior.

Structure of what lies ahead

In the remainder of Part I we discourse on the nature of intelligence and management and define our framework and terms. In Part II we attempt to provide specifications for intelligent management information systems. We review the current state of the art in light of these specifications in Part III, and in Part IV we propose steps and some research to be undertaken to fulfill the specifications. This is followed by a brief summary and conclusions.

On the nature of intelligence

To provide a rigorous definition of "intelligent behavior" is extraordinarily difficult, but we do wish to offer a working definition, to clarify our use of the term as applied to management information systems. Intelligent behavior is "doing what you do for the right reasons," roughly speaking. According to this definition, intelligence has two facets: first, *understanding* of cause and effect, and second, *logic or rationality in employing that understanding.** Note that we haven't said "doing what is right" in an *ex post facto* sense. Our definition admits of intelligent behavior in the face of uncertainty; it requires neither omniscience nor clairvoyance, and distinguishes between the decision-making process and its outcome.

Let us be more specific about this concept of "understanding." The better one understands a phenomenon, the more accurately he can predict its behavior. And, for purposes of subsequent discussion we shall find it convenient to associate understanding with a predictive *model* of a phenomenon. That is to say, by increasing managerial understanding, we shall mean improving the manager's model† to be a better image of the real world—his models, whether they be ex-

*"Understanding" corresponds (approximately) with the *inductive* aspect of intelligence and "logic" with the *deductive*.
†We are implying many models in the mind of a manager. At any point in time, the models need not be mutually consistent, humans being humans.

plicit or implicit, being his basis for prediction.‡

The logic-rationality aspect of intelligence can be described in this way: given a set of relationships and facts, one can be more or less adroit in exploiting this knowledge for his own ends. There are at least two types of situations in which logical use of knowledge is at least as important as knowledge itself. One is in decision making under uncertainty. Here statistical decision theory is now offering a suitable rationale (although in problems of large size, the computational side of the logic becomes important). The second situation is in system management. The detailed interrelationships among system components may be well understood but overall system behavior is not, either because of dynamic effects or simply because of the combinatorial nature of the full system. Forrester (1962) has offered an approach in the former case, and simulation or heuristic programming some promise for the latter.

We should also emphasize that intelligence is both a relative and dynamic quality of persons (and organizations). We cannot, except arbitrarily, categorize behavior as intelligent or unintelligent, but we can talk about "levels" of intelligence. We are willing to admit that von Neumann was smarter than we, individually and collectively (we have some suspicions about some others as well). Hence, we shall be concerned with organizations which are more or less intelligent. In stating that intelligence is dynamic, we simply mean that one can grow (or even regress) in the qualities associated with intelligent behavior. In other words, one can learn.

In the light of the above, we can now be somewhat more precise in our goal: We want information systems designed to provide for *more* intelligent behavior *over time.*

On the nature of management

Our assumption is this: A management organization manipulates those resources under its control to achieve results in accord with its objectives; this manipulation is based on management's collective understanding of causal relationships among resource inputs, processes, external environmental factors and outputs.

In our view, the main management processes which require intelligence are planning and control.*

Planning, as we construe it, subsumes all consideration of the future course of events, whether the time scale is short or long and whether the process is formal or informal. Planning thus includes searching for future alternative courses of action, selection of goals, specification of procedures to be followed or resources to be acquired and utilized for the achievement of the chosen goals.

We distinguish between *operational planning,* in which the emphasis is on what the organization should be doing in the relatively short run as constrained by the dominant characteristics of its current structure, and *strategic planning,* wherein the emphasis is on possible changes in the dominant characteristics of the structure (physical or organizational) or in major goals. The nature of the planning process entailed is likely to be quite different in the two types, to wit: Operational planning is typically a continuing, systematic process whereas strategic planning is more often *ad hoc* and unstructured; operational planning is intimately linked with control, providing milestones or other goals and receiving current system status, whereas strategic planning is likely to consist of one-shot ("terminal") decisions only loosely linked to the formal control system.

Regardless of emphasis, *planning always involves a model;* the model is explicit in many cases, but is certainly implicit in any activity that projects the future.†

Control

The object of control is to obtain desired behavior or results (often as set forth in a plan). Given the specification of the desired behavior or result in the form of a quantified standard, goal or budget, there are certain processes which go into control. The first is *measurement* of the status or performance of the controlled entity. But the measurement has no meaning until it is juxtaposed with the standard or desired measurement, consequently *comparison* is a second basic process. The third process is *direction,* a process of communication in which the controlled activity receives signals to alter its behavior to obtain closer conformance with that desired.‡

Measurement, comparison, and direction are common to all "feedback" control, but quite often there are additional processes present. *Classification* is one such process. It is required when one measurement relates to several different activities or entities (this is part of the notion of "integrated data processing"). The fact of completion of a task in a manufacturing shop may be reflected in individual worker productivity, the foreman's direct labor expenditure, production progress control, and the like. So within the control cycle there is required association of the measurements with the appropriate "responsible" entities, which is classification.

‡Pounds (1965) has made a strong case for the universality of models in managerial behavior, in "problem finding" as well as in problem solving.

*Our classifications, planning and control, are quite inclusive. They correspond roughly with Anthony's (1965).

†For a thorough discussion see Emery (1965).

‡There are situations in which deviations of one direction only are "bad," deviations of opposite sign therefore receive no correcting signal; if anything, they receive "reinforcement."

Also, implicit in comparison-direction is the necessity for determining the cause of deviation between actual measurement and desired. *Diagnosis* is the term we will use. Diagnosis is a trivial process in some instances, especially in automatic process control. In a household temperature control system, the furnace is always assumed to be the culprit (within the purview of the automatic controller) when actual temperature departs from that which is desired. On the other hand, in a process of any complexity, particularly when probabilistic elements such as human beings play a part, diagnosis can be an extremely complex process. A time overrun on an activity in a PERT network seldom has a simple cause, for example. In manufacturing cost and schedule control systems, allocation of blame between performer and standards estimator has always been a source of argument, to say the least. Diagnosis may occur before direction, in which case causality is considered in formulating the direction, or it may occur as a result of the direction, in which case it is performed by the controlled entity.

Control systems differ as to the specificity of the desired behavior. In simple cases, the purpose of control is strictly *regulative,* keeping performance within reasonable limits. But in other cases, again especially when people are involved, the control system assumes an *educative* role. There are two aspects to this educative role. In cases where direction is readily determinable and diagnosis can be more effectively performed at the activity level, the control signals motivate search (self-diagnosis, so to speak). In cases where diagnosis is performed prior to direction, the control signals encourage desirable aspects of activity by "rewarding" them in some way (or by "punishing" undesirable aspects), thus leading in theory to process improvement, "learning" again. All incentive systems, whether applied to top executive or to workers are educative in purpose.

Control, just as planning, is always based on a model. The model may be a simple expression of formal cause and effect relationships (e.g., furnace yields heat, bad supervision yields unfavorable labor variance), or it may be highly informal, implicit in *post mortem* analyses effected upon major deviations, or it may be an explicit mathematical model.

Hierarchy in planning and control

All management organizations are hierarchical to a degree,* and this implies hierarchy in the processes of planning and control. Even at a single organizational level planning is "senior" to control in the sense that it provides the goal for the control process. Another potential interrelation between planning (particularly operational planning) and control occurs in the diagnostic process. As the control model is revised, presumably the planning model should reflect the new insights. We advocate that this process be formalized.

There are several levels of planning and control in a management organization. We expect to see at the lowest level of the system detailed plans (often in the form of schedules) driving the control of the basic productive physical processes. At higher levels, we expect to see efforts to coordinate (via a plan) the control of multiple interdependent activities. Lower level managers control operating processes, but higher level managers control lower level managers. Anthony (1965) draws a sharp distinction between "operational control" and "management control" (in the latter he is referring to the manager as the controlled entity as well as the controller). We will be particularly concerned with a related aspect, that of controlling the planning process. Consequently, we draw a sharp distinction between *planning process control* and *operating process control.***

On the virtues of intelligent management

Campaigning for intelligent management does not seem controversial on the face of it, but it does beg for some consideration of the economic justification of our particular forms of intelligence. We seek better understanding—more valid models—and we seek better logical understanding. A more valid model means more predictable execution of plans as previously noted. Better logic in developing plans means better alternative courses of action selected, other things being equal. In regulative control, a better model enables reduced variation in performance—output to closer tolerances, for example; in educative control, a better model enables more accurate recognition of desirable behavior and hence more rapid improvement.

The nature of this recognition can be illustrated by the story of B. F. Skinner's "superstitious pigeons" (Holland and Skinner, 1961, p. 88). Several pigeons were placed in separate boxes. A feeding mechanism delivered food to each pigeon every 15 seconds regardless of what the pigeon was doing. After operating in this way for some time, the experimenter observed that one bird was sitting very still, another bowing, another turning around in tight circles, another hopping on one foot, and so on. Each bird repeated its own ritual between feedings. Analogous (presumably civilized) human reaction to indiscriminant rewards or

*We are well aware of even higher levels such as planning process control planning and planning process control control, and so forth. We assert that an information system sufficient for planning process control is sufficient for higher levels as well.

**For a thorough discussion, see Zannetos (1965b).

punishment are frequently found in competitive athletics and, we suspect, in management.

We also will offer subsidiary arguments for formality in planning and control processes. One reason is that while various cognitive agents, namely people, may come and go, in order for the organization to increase its intelligence over time, it must make some provision for recording the accumulated planning knowledge. It thus guards against loss of memory together with the people. A formal model can be stored in "memory" and hence provide continuity in intellectual growth.

Specifications for intelligent information systems

Introduction

We are focusing upon operating process control and planning process control as the central cognitive processes of a management organization. In operating process control, we wish mainly to obtain specified behavior of the activities which comprise operations. In planning process control, we assume primarily an educative goal, that is to say, we wish explicitly to improve the planning process. We will first treat the problem of regulation and improvement in operations and identify therein the potential for structures of different "levels of intelligence," and we will attempt to specify the information system requirements implied for the highest defined level of intelligence. We will then turn to the more difficult problem of planning process control. The difficulty stems from less tangible process goals, less formal processes, and unclear boundaries on the problems being attacked. The characteristics of the control process are the same in planning process control as in operating process control, but the process being controlled is sufficiently different in the former that only a highly intelligent control system will suffice to assure improvement.

Operating process control

There is always a process model which underlies control. It is on the basis of this model that the magnitude and sign (in general, the nature) of the control direction is determined, and, in more complicated systems, the particular agency to receive the signal is determined.

The model can be naive or sophisticated. This is partially a question of the complexity entailed in the model, but in our view, more fundamentally related to the depth to which underlying cause and effect relationships are captured. Applying a polar classification to a continuum, we identify the extremes as *symptomatic* and *causal* control. That there is a continuum of causality should be clear to anyone who has attempted to respond in good faith to a three-year-old's infinite series of "whys." An example of clearly symptomatic control is a wage incentive system used for educative productivity control. Output and reward are directly linked, and little formal attention is paid to *causes* of output (or lack of output) except when major dislocations such as machine breakdowns or material shortages disrupt the process. The implicit assumption is made that high output results solely from energetic or skilled attention to duty by the worker. If this assumption is largely correct, the system works. But if output is affected by a substantial number of causes other than the worker's activity, the system can be acrimonious in its administration and ineffective in its application.* A more causally oriented control system applied to the same problem would attempt to correct output variations for "degree of difficulty" so to speak, by removing the effects of differences among tasks (i.e., more precise standards), differences among materials or material suppliers, differences among machines and the like. It would, in this case, isolate as nearly as possible that portion of output variation truly attributable to the worker. In the extreme case, it would attempt to classify detailed elements of the worker's behavior as being causally related to output and to reward each appropriately, which is to say, it would assist in discrimination.

The close connection between causality and understanding implies that causal control provides a higher level of intelligence than symptomatic control.

Another dimension of classification related to the question of intelligence is the adaptivity of the model. At the lowest level in this case is *reflexive control,* which is based on a fixed model with fixed or externally supplied parameter values. The name derives from the parallel to a reflex in human physical behavior; the effect is that of a "stored response" to stimuli. Habitual behavior is reflexive. Reflexive control is employed in very simple situations such as the house temperature control as well as in highly complicated systems for inventory control based on mathematical models. The salient point is that the system behavior is not easily modified and is certainly not self-adapting to a changing environment.

The next level is *parametrically adaptive control.* At this level the model is *fixed* as to the constituency of variables and parameters and the relationships among them, but the *values* of the parameters are changed as a function of experience or as exogenously supplied data vary.† This type of system is often seen in chemi-

*The battle among workers for "make out," i.e., tasks for which causes other than the workers' effort make good performance easy, is behavior symptomatic of this type of problem.

†Such systems can be thoroughly sophisticated. See Bellman (1961).

cal process control, in which the model relates yield to a variety of input and process factors. The weights placed upon the factors are varied as experience accumulates, providing lagged response to new environmental information. Another aspect of parametric adaptivity is seen in inventory control based on "adaptive smoothing."‡ In this case, the smoothing parameters are adjusted as data are processed in an attempt continuously to obtain a minimum variance forecast.

At a still higher level, we can hypothesize *inductive control,* in which the entire model is subject to change, both in structure and constituency (i.e., functional form) as well as parameter values. What we have in mind here is continuous re-evaluation of the model, diagnosis being the central process. Inductive control, in attempting to establish causality in greater depth, involves formulation of new hypotheses and tests thereof. Indeed, since it is advancing "hypotheses of causes" it parallels closely the Bayesian "prior to posterior" process with highly complex multivariable models.§

We would ask an additional step in inductive control. Since the control model is being adapted, it would seem essential to adapt the related planning model as well. In fact, we will denote inductive systems which provide for direct updating of planning models as *prognostic* as well as diagnostic in purpose.

We know of no examples of *formal* inductive control systems in operation. To clarify the ideas, however, consider the following situation. There was a statistical analysis of yield performed on a mechanical process. The purpose was to relate process yield to various (controllable) input and process factors and ultimately to increase the yield. A rather elegant predictive model was obtained, but the remaining unexplained process variation was still substantial. Additional variables were then studied to little effect until the time of day during which the process was being operated was tested. This variable showed almost dominating significance. Further investigation showed that the third shift superintendent was paying essentially no attention to process yields with predictable effects. While this situation may illustrate either missing the forest due to overzealous tree examination or serendipity (depending on what the responsibilities of the analysts are assumed to have been), it is also a clear example of economically significant increased understanding resulting from an inductive control process. We shall later cite additional cases where the purpose of inductive control is implied in informal systems.

Levels of intelligence in operating process control

As must be clear, we would classify reflexive-symptomatic control at the bottom of our scale and (prognostic) inductive-causal at the top, since the latter provides the potential for achievement of the highest level of intelligence in an organization—for learning, in the general sense.

But let it be clearly understood that we are not advocating wholesale redesign of all operating process control systems to achieve this sort of intelligence. In fact, to the extent that the environment is stable and very well understood, a reflexive control structure may be wholly adequate. After all there are many cases where response to the symptoms also cures the disease. To the extent that constituency and general functional relationships are well understood and stable, parametrically adaptive control may be just right. On the other hand, to the extent that the environment is *not* perfectly understood or is changing, then there exists an argument for inductive control.* As we view the world, the latter category appears to include the majority of system control problems and the majority of activities subject to rapid technological, political, or market change.

We, therefore, advocate intelligent control as a rule, but admit to exceptions and add that all types can co-exist within the same overall system. In fact, an intelligent control system may be viewed as having the capability to sense (through scanning) what level of sophistication should be applied to each situation. As we have already mentioned, planning and control are hierarchical in nature.

Information system requirements for intelligent operating process control

An initial step in establishing causality is to establish *association* between the basic variable or variables of interest and other factors capable of being measured and either corrected for or controlled themselves. Thus, the first step in uncovering the causes of lung cancer has been to establish the association of the incidence of that disease with cigarette smoking. Association is necessary for causality but not sufficient to prove it; it is a first step.† Since diagnosis in most cases occurs after the fact, as with other *post mortem* (or *post*

‡Brown (1963) has a complete description and discussion.

§The "hypothesis of causes" was Bayes' own name for his theorem. Basic references in Bayesian statistical-decision theory are Schlaifer (1959) for the layman and Raiffa and Schlaifer, (1961) for the expert. The process of modifying functional form and variable membership in models is *not* at present included among the problems solved by the Bayesians, but "adapting" parameter values is.

*There is always a question, too, of the economically justifiable depth of diagnosis in inductive control. Since causality is not *necessary* for predictability (it is *sufficient*), it may be optimal to cease searching at some symptomatic level.

†With rare exception, causality cannot be established statistically; proof of sufficiency often requires systematic elimination of all other possible causes or controlled experimentation.

victoriam) activities, it begs for reconstruction, in a flexible way, of the situation when the unexpected occurred. This implies a requirement for a variety of associations among factors, temporal as well as functional, for adequate feeding of the diagnostic process, especially if it is to be carried out before the undesirable event completes its course. Furthermore, a difficulty in establishing association is *confounding,* which is the inability to separate the effects of two or more variables due to overly gross or aggregated measurement. Hence, we require in our supporting information system the facility for functional and temporal association with precision and in detail. To be more helpful this association must take place as soon as possible, on the basis of incomplete information, and be refined as new data are received.

The problem of deciding just which measurements should be maintained is difficult. Potentially relevant data, not just known relevant data, are needed, if the model itself is to be modified. This fact may explain the popularity of parametrically adaptive control models; with them at least the data base is well-defined.

Let us attempt to be more specific about the idea of association. What is needed is a way of finding out the values of a large number of variables which were current at some point in time. Functional association requires linkages among the factors and the basic process measurement. In the yield analysis described above, for instance, all of the measurements of input and process characteristics had to be linked to the yield on a particular batch. The temporal association capability allows for lagged effects, and for dynamic analysis of phenomena in general.

A detailed associative data base is the raw material for inductive control, but additional capabilities are required for the diagnostic elements. Some aspects that are known to be present (but which are not well understood) are *pattern recognition* and *pattern generation*. The former includes the ability to perceive relevant associations and to match a given pattern to observed behavior. Often this involves "normalizing" the data, putting them in a proper format or otherwise transforming them to conform to the pattern or patterns being tested. For example, simply arraying data in time series form normalizes them for certain dynamic pattern matching; in other cases, a graphing of frequency spectra might be required.

Pattern generation is even less well understood, but it clearly involves abstraction and quantitative hypothesis formulation, which is to say, *model building*. The question begged is what is the source of the model. Much opinion suggests that there exist frameworks, general theories, or taxonomies—broad categorizations of phenomena—which suggest detailed models for testing. Freudian psychology, Marshallian economic theory, or more recently, Forrester's "industrial dynamics" (1962), are examples of formal frameworks. In general, however, the totality of one's experience, observation, and education serves as the framework for a human. Pounds (1965) suggests that the process of diagnosis often begins (a "problem is found") when behavior departs from that suggested by one of these frameworks.

An example may serve to clarify what we mean by pattern recognition and generation. Forrester (1962) has cited some instances of self-induced oscillatory behavior in business, one of which was evidenced by inexplicable seasonal demand for a consumer product. By drawing on his general framework he was able to construct a model which (qualitatively) matched that of the (normalized, measured) behavior of the firm. From his model, he was able to deduce that the cause of the seasonal peaks and valleys were the firm's traditional promotional patterns and its customers' anticipation of this pattern.

Pattern recognition, abstraction and hypothesis formulation remain shrouded in mystery as to their precise mechanisms; they are apparently tied up with the very most arcane human capabilities which are often collectively labeled "creativity."* The mystery notwithstanding, we require these faculties as operative elements in inductive control. Also bear in mind that they must be employed in the worst of all possible inferential worlds evidencing as it does probabilistic and dynamically non-stationary behavior and imperfect measuring devices. And, given the mystery, we would expect humans, as opposed to machines, to supply the capabilities required.

Planning process control

When the planning process is brought under surveillance, all of the previously cited aspects of control apply, but there are some new problems to face as well. Some of these can be stated as follows:

- Planning is always based on a model, so in control of planning there is a metamodeling problem. We require a model of the (planner's) modeling process and a model of the employment of the planning model.
- Planning, especially strategic planning, is often based on information about matters external to the firm or organization. For example, predictions of competitors' behavior or general economic conditions are often basic to commercial planning.

*Minsky (1963) and Newell and Simon (1961) have much to say on this. The fact is that, at this point in time, people can do these things very well and machines not well at all. See also Licklider (1965).

The enemy's order of battle occupies a similar position in military planning. Hence, planning process control requires a data base that is not necessarily a convenient by-product of operating process control or otherwise at the disposal of its users.

- In order to establish control, there must be a process goal or standard, in this case an objective purpose for planning. Yet it is not always abundantly clear just what one is attempting to achieve by planning, with the possibly innocuous observation that he is attempting to bring about some order to an otherwise chaotic situation and so structure his problem.
- Planning is frequently intuitive and subjective both as to process and to data. Yet in order to exert control, the subjective estimates and value judgments require quantification and their functional relationship with the planning goal requires establishment.
- Planning, in many cases, looks far into the future. It would be desirable to conduct *post mortem* analysis for process improvement, yet if the planning process controller waits for the future to reveal itself completely, the control cycle time will be too long.
- Planning is typically a group rather than an individual process. We understand little enough about individual behavior but even less about group behavior. Operating also is often a group process, but planning is "groupthink" rather than "groupdo." Again, this is a matter of difficulty in associating cause and effect.
- Correlatively, planning is a task still (if temporarily) performed largely by people. Attempts to observe or to experiment with people often lead to the well-known "Hawthorne Effect," in which the subjects respond to the fact or conditions of the experiment rather than to the environment being studied.* What is even worse, often the environment surrounding the experiment itself changes by the mere fact of being observed.

This list is long enough to be discouraging to the most ardent of idealists; but the alternative of uncontrolled planning should be sufficiently dismaying to make the effort worthwhile. And this list suggests a process sufficiently poorly understood to require inductive-causal control.

*The name stems from some working condition experiments conducted at the Hawthorne Works of Western Electric in the thirties. A group of women workers was submitted to varying lighting, heating and other factors. Regardless of conditions their productivity rose. The experimenters finally concluded that the women were responding to the attention that accompanied the experimentation. Described in Roethlisberger and Dickson (1939).

Information system requirements for planning process control

We clearly require an upgraded data base for planning process control. Its scope requires expansion to include both external data (including forecasts of external variables) and subjective data. By the latter, we mean that the planning assumptions, subjective estimates, and value judgments should be formally recorded. And, of course, we require the same associative facility with these data as we did for operating process control.

What this amounts to is a plea for formal models in planning which we add to those previously voiced by others, notably Emery (1965). The added motivation is the potential here for planning process improvement. To this we add the requirement of formal goals for the planning process.

For control, it is not sufficient merely to evaluate the product—the plan—we require access to the process as it operates. This means somehow capturing the "stream of consciousness" of the planner to obtain his "trace," i.e., the logic he has used to formulate his plan. To tighten the loop in long range planning we need a method for analyzing incomplete returns, to infer on the basis of partial data. And, we require as before a powerful diagnostic facility. Finally, some provision, such as clandestine, unexpected, or constant surveillance must be made to avoid the Hawthorne Effect.

Commentary on the current state of the art

Introduction

We will attempt to review what we perceive to be examples of elements of intelligent information systems which are generally or specifically in operation. We face a problem in so doing in that we suspect that organizations which have achieved higher levels of intelligence in their information systems are probably intelligent enough not to publicize the fact, so the state may not be so primitive as we represent it.

The state of operating process control systems

Control systems for detailed productive processes have been growing in sophistication ever since computers became generally available. In continuous process control, for example, very elaborate formal-model based systems for chemical processing are now common. These vary in their complexity, but most commercially available systems are capable of multivariable control at multiple levels (i.e, they adjust the process to conform with desired values on several variables and also compute the desired values for the variables based on external inputs). Parametrically adaptive systems of at least modest scope are operative as well. However, since these systems are fixed as to model structure, such inductive inference towards model improvement as takes place must be performed externally to the system. These systems are of interest as models for man-machine system control, but while sophisticated in model structure, they offer no guide for model improvement—they do not evolve.

Another area of interesting development is that of detailed job shop production scheduling and control as practiced in Hughes Aircraft (Steinhoff, 1966) and Westinghouse Electric (Trilling, 1964), among other firms. The general structure of these systems is this. A simulation model is fed with inputs of the current order backlog (with routings, processing time estimates, and due dates), the shop configuration (machines, and men), and some decision rules for dispatching the jobs. The model is run and rerun, simulating the future course of events, allowing for adjustments to backlog (i.e., subcontracting), or capacity (overtime, added shifts) and the decision rules. When a "satisfactory" simulation is obtained, the *simulated* start time of each job on each machine is used as the *scheduled* start time for the job in the shop. This schedule provides the plan for production control.

These "finite capacity" schedulers (so-called because they explicitly consider the capacity available at each work station before simulating the assignment of a task) are considerably more complex than the standard "infinite capacity" scheduling systems in which a scheduled date for each task is obtained by dating back from the job due date using "standard lead times" (which allow for direct production time, waiting, transit, setup and the like) for each operation on the routing.*

One problem with infinite capacity schedules is that they are generally not feasible, even in theory. The schedule dates provide crude targets for progress, but unless work station capacity is directly considered, a deviation from schedule may only signify that the schedule was impossible at the outset. Then, deviations resulting from model inadequacy are confounded with true process deviations. This is less a problem with a finite scheduler. A deviation in the latter case generally indicates that something unexpected has occurred such as low productivity, a bad processing time estimate, a material shortage, or failure to follow the scheduled sequence. While causality is not pinpointed, a point of departure is established. (Even in finite capacity schedules, minor deviations tend to compound after a time and schedule infeasibility again rears its head. Potentially, this schedule "decay" can be cured, and discrimination of causes materially improved in on-line, real-time control systems. This possibility is discussed in the next section.)

The more detailed model (derived from the more detailed data base) used in finite scheduling, and the built-in time-based association of resources (work stations) with activities provides the increased control power. In effect, the better model eliminates "noise" from the information system and gives the manager more confidence in the deviation signal. In comparison, the naive infinite capacity schedule tends to "cry wolf," leading to ineffective remedial action.

In conventional accounting control systems the state of the art is rather primitive. Budgets and other standards are frequently almost arbitrarily arrived at and often used on a memorandum basis. Few deviations have any significance and they could easily result from factors totally beyond the aegis of the controlled entity. There is no systematic way of filtering noise from the information system and no aids are provided for causal diagnosis (beyond a superficial level or even for determining significance). This is not to say that managers do not attempt to determine causes of budget overruns, for example but, that such diagnosis occurs separately from the control system (and in some cases, in spite of the control system). Often, because of traditional accounting practices, data are averaged indiscriminately and also "closed-out" destroying the usefulness of the data base for effective managerial control and decision making. We must stress that we are not concerned here with the external financial reporting aspects of accounting in which justification for the conventional practices may be found. Our focus is internal control. For this, at best, in the absence of managerial brilliance, the conventional wisdom in accounting control amounts to symptomatic-reflexive control, the bottom rung of our intelligence ladder. It is small wonder that managerial behavior approximating that of the "superstitious pigeon" is not uncommon.

There are, however, some candles being lit in this area of stygian darkness. The general idea of the "flexible budget," where elementary cause-effect relationships are derived to allow measurement and separation of deviations from plan due to volume of activity ("volume variance") from deviations due to efficiency in labor performance and use of material resources independent of volume, represents an attempt to separate gross uncontrollable (by the production manager in this case) effects from those which can properly be laid at his door. But the accountants outside of limited use of this tool (in the area of manufacturing operations) have not pressed on in this

*Emery (1961) provides a discussion in depth of various alternative scheduling systems including these two.

direction, neither in depth nor in order to apply these techniques to other aspects of operations.

More encouraging are the recent attempts in the Bell System (Harvey, 1966), the military and elsewhere (Black, n.d.; Zschau, 1964) to establish performance standards on the basis of more precise statistical models. For example, suppose there is a multiple plant company with each plant producing comparable products. An electrical utility will serve as an example. No doubt there will be variation among plants in measures of performance, say average delivered cost per kilowatt hour. Some of this variation is attributable to plant management, but a number to factors are outside the control of management, such as fuel costs, age of generating equipment, population density, climate, size of the area served, industrial concentration and classification and so on. Merely to compare plants on the basis of the raw performance measure is patently unfair to the plant manager who has drawn unfavorable circumstances. Mean performance as a function of all of the uncontrollable factors can be predicted for a particular plant on the basis of statistical (i.e., multiple regression*) models. This revised performance figure represents a standard calibrated for "degree of difficulty," so to speak. Performance deviations from this calibrated figure represent true "managed" performance plus a much smaller "unexplained" component, and certainly provide a more accurate basis for learning, reward or castigation.

These control systems, in their separation of uncontrollable causes from controllable, represent a major step towards causally oriented operating process control. While the particular techniques used in the cited studies may have been limited so far mostly to relatively homogenous product oriented industries, the philosophy underlying the use appears impeccable to us, and can be effectively extended, we believe, to many other situations. For example, the causes of variation in overhead costs and central services in general may be estimated in this manner. Another interesting possibility for extension would be to associate the residual variance with identifiable management-controlled as well as extra organizational variables, e.g., work force composition, salary levels, some quantified aspects of operating strategy, competitive reaction, etc. Associations of this sort would lead naturally to diagnosis—inductive control.

We have observed efforts toward diagnosis in system management, particularly in the PERT-based planning and control system employed by NASA in the APOLLO program. First, there is a formal model used for planning and control. Second, the evidently widespread doctrine of "visibility" is employed in project time and cost (and to a lesser degree, technical performance) control. This calls for focusing attention on responsible parties in cases of unfavorable deviation from plan. From discussions with both the system managers and contractors it seems clear that what goes on in the "control rooms" during the *post mortem* project reviews is causally oriented diagnosis to a substantial degree.† The significant point is that the whole information system appears to be oriented towards this process. We feel that "visibility" insofar as it encourages diagnosis, is a useful system design concept.

In addition, several attempts have been made by people at NASA to "calibrate" the bias of the planning estimates of the contractors and thence to correct the plan for this bias as it becomes known, a clearly prognostic exercise.

The state of planning process control

In general, it appears to us that organizations have recognized the need for planning process control for years, but have instituted little or no formal surveillance. For example, there is usually some control exercised over the process of budget preparation in government and industry, in the form of critiques of assumptions and also end-of-fiscal-year post mortems. One of the clearest examples of this is the Westinghouse Electric "Profit Planning" system described by Evans (1959). High level (product department) plans are examined, reviewed, and critiqued on the basis of their assumptions and substance *before* the execution begins, and periodically during the year, the execution is reviewed. Care is taken to separate effects due to poor planning or poor forcasting from those due to poor performance both by dialogue and in the structure of the planning accounts themselves. By classifying costs into "committed" (i.e., fixed, not under management control), "managed" (i.e., discretionary overhead such as management salaries, consultants, computer rental) and "product" (i.e., direct and indirect materials and labor), greater precision in attributing variance to particular gross causes is obtained. But there is no evidence of formal diagnosis being employed in this approach, and the control system is of an *ad hoc* nature. It represents, in a sense, symptomatic planning process control.

Another area in which planning process control has evidently been pursued has been in the military. The classic doctrine of von Clausewitz, which continues to

*Broad coverage of the technique can be found in Ezekial and Fox (1959) and Graybill (1961).

†The first item on the agenda is naturally an inquiry into what can be done to correct the deviation as it exists, only then comes the "Why?" In many situations diagnosis cannot take place until after some action is taken.

influence military planning all over the world, requires the commander to state formally his goals and concepts and "estimate of the situation" at the outset of the planning process. Such a system forces an explicit statement of premises and conclusions and facilitates after-the-fact assessment of blame among assumptions, plan, and execution. Confounding the causes of poor performance is avoided. More recently these traditional methods of military planning have been extended and also enriched by the use of simulation and game-theoretic approaches. While planning process control has been the objective of many of these improvements, it has not, as far as we can assess, become a part of a regular feedback control system. No doubt the necessity for determining strategy on a decentralized basis (in the field) has some bearing on this issue. Also the objectives of warfare are much simpler relatively speaking than those of the firm (we are talking in terms of ease of definition not in terms of execution, magnitude, or significance), so finding a substitute for an *ad hoc* planning diagnosis may not be as critical for the military.

Planning process control has often been employed in military *training;* it is not so clear that it occurs under the pressure of actual operations. But the critiques of maneuvers and large scale training exercises frequently focus on the planning process itself as distinct from operations as executed.

Informal, qualitative planning process control is limited in its effectiveness, again because of the discrimination problem. It is one thing to know that an estimate was bad; it is another to know why it was bad. And, because the planning process itself is relatively unstructured, it is difficult to pinpoint the particular subprocesses that were defective.

As we argued earlier, one way to improve the potential of the control process is to move towards more formal planning models. Rigid plan formats (the "five paragraph" military format, and the Westinghouse chart of planning accounts are examples) and specific procedures are steps in this direction. We believe that more detailed and more complex models—in short, mathematical or computer models will be of even greater value in forcing explicit assumptions and estimates and organizing the process for controllability.

One example of a trend in this direction is the Apollo Project simulation model recently installed for the NASA Office of Manned Space Flight.* Because the planning assumptions are, in effect, inputs to a computer program, they are "visible"; because the planning procedure involves explicit recourse to a computer program to examine alternative procedures, it would be possible to obtain a "trace" of the search process. Hence, the raw material, i.e., the basic "measurements" of the planning process are available. But since the planning horizon for APOLLO is long (at least to 1970), the problem is obtaining feedback on planning results, and one of accurate association. This is a case similar to that facing many non-military organizations, and one which calls for partial data analysis.

Another element of our specifications is being implemented at Westinghouse Electric, namely that calling for expanding the corporate data base to include extra-corporate data.† This also contributes basic measurements for planning process control.

Some conclusions on the state of the art

We conclude from our brief review that there exists no publicized comprehensive realization of intelligent management information systems. We also perceive evidence that intelligence is sought in numerous cases.

Diagnostic control of operating processes seems imminent in restricted environments such as continuous process control and the basic pattern is being established even in such messy discrete process control areas as job shop production control. Employment of well-established statistical methodology holds promise for inductive control at higher operating levels.

Planning process control at present is in some cases performed, but always performed informally. Diagnosis appears to be *ad hoc* and somewhat political in flavor. It is at best qualitatively based and this, coupled with the generally informal nature of the control process, lead us to suspect that its effects are impermanent even when it is effective. But the general trend towards more formal planning models offer opportunity for greater sophistication in control.

In total, many of the bits and pieces from which higher level intelligence in management information systems can be fabricated exist. The problem is to assemble these within one organization.

Steps towards realization of intelligent information systems

Introduction

The total realization of intelligent information systems requires considerable research and development before it can be accomplished. For example, there exists almost no general theory of diagnosis in particular, and inductive inference in general. On the other hand, we feel that some major improvements in the state of the art could be effected simply by recognizing the value

*The details of this system have not yet been publicized. It was designed by Peat, Marwick, Livingston and Company under contract number NASw-1223 and installed this year.

†Described in Burck (1965, p. 113).

of what we have called intelligence and reorienting the information system to the end of acquiring it. Also, new information technology (including modeling approaches subsumed under operations research such as digital simulation and heuristic programming, as well as "third generation" computer technology enabling real-time data processing and time-sharing) now affords some major capabilities which can be exploited for this purpose.

Therefore (with perhaps more alliteration than accuracy) we have defined our steps toward realization as recognition and reorientation, real-time processing, and research.

Recognition and reorientation

Given the desire to increase the intelligence of an organization, there are some fundamental steps that can be taken. In our view, the major discrepancy between typical operating process control systems and those which we want is in the explanatory power of the underlying process models. That is to say, conventional operating process control fails to get at the underlying causes of process variance. Relatively unsophisticated statistical analyses, such as analysis of variance and covariance or multiple regression, can shed considerable light in this area in many situations. And we suspect that simple classification-association would uncover some of the grosser causes. For example, we recently observed a case in which a salesman's pricing misbehavior was uncovered simply by comparing his customer claims experience with that of the rest of the sales force. The basic statistics showed that his claims were far more frequent and, on the average, larger than those of his colleagues. Deeper investigation uncovered the fact that many of his claims were unrelated to damaged, missing, or substandard goods, but were simply his mechanism for granting price concessions to customers (his commissions were not adjusted for claims). We could cite numerous other examples of surprise resulting from attempts to rationalize the causes underlying other performance measurements. It is important that we stress, however, that such rationalization of causes must become part of the information system if intelligent system behavior is to be obtained.

A present-day managerial accounting system has the capability to store raw data and classify them. Classification is done purely on the basis of human intervention, because the system does not have self-organizing characteristics. But given the classification, through a matching process the system can extract differences, which are then reported to management. But the matching process does not at present tolerate any ambiguity. It is deterministic in terms of the classifications assigned.

A difference by itself, of course, does not mean very much. Although "red" variances (debit balances in manufacturing accounts, for example) are automatically considered undesirable and "black" variances desirable, in reality, much more analysis is necessary beyond this stage to determine significance. At best these differences may point to a *potential* problem. The questions that come to mind on observing these accounting variances are mainly of two types: (a) how significant (in a probabilistic sense) are they, and (b) what do they mean?

To enlarge the capacity of the data base and the capabilities of the information systems, one may store in the data base cues for automatic response at the operating level. This response may be purely of the reflexive control type or the result of elementary analysis performed by the system itself. Still, such a system does not allow for any ambiguity, it is deterministic and inflexible. To generate intelligent behavior, the data base must be capable of resolving ambiguity, and possess understanding and learning capabilities.

We mentioned above in conjunction with the output of present accounting systems that the latter do not tell how important are the variances they generate. It is all left to the imagination of the manager. One obvious improvement, therefore, is to introduce probability distributions into the data base. Also decision rules for determining the probabilistic significance of the observed deviations should be included. The system can be instructed to sift through the differences, take remedial action on the basis of prestored cues, or else report the significant variations to the manager—"management by exception." But we need not stop here. We can also in a Bayesian sense review the models which govern the expectation of system behavior, and also update the relevant probabilistic distributions.

To facilitate understanding and improve learning it is not sufficient to have a system which separates the significant from the insignificant variations in performance. Somehow the system must help the manager focus on the underlying cause-effect relationships.

A method of introducing the necessary capability of cause determination in the data base is to store predetermined functional (cause-effect) relationships and explicit decision rules to facilitate their use. Such an arrangement, however, is not very different from reflexive control; it is inflexible and limited in its intelligence. No understanding or inference takes place. We could alternatively "instruct the data base" in the methods of arriving at hypotheses of cause and effect relationships by itself (not a trivial task). This is a more promising avenue because it permits adaptability.

Simple and naive techniques such as statistical variance and covariance analysis, if performed on the accounting variances, can yield useful cause and effect relationships to be stored in the data base for further analysis and testing hypotheses. This type of a system was elsewhere called a *functional accounting* system (Zannetos, 1966d) and many of its characterists and prerequisites for implementation have already been discussed (Zannetos, 1966a, 1964, 1965c). We believe that with the present state of technology and knowledge, the functional accounting system is realizable now. Furthermore, under such a system many facets of the design of organization structures are brought within the purview of the system and resolved analytically for the first time (Zannetos, 1965a).

The major discrepancy in planning process control derives from the informality both of the planning process itself and whatever planning review procedures exist. The initial step, we believe, is to impose some formal requirements on the planning process for purposes of establishing the basic measurements for planning process control. More specifically, we advocate formal planning models, again as part of the information system. It must be granted that formal planning does not imply formal planning process control, but it is the point of departure for setting up the necessary data base for evaluation of performance. Unless there is a systematic method of separating the assumptions and forecasts (the model and its parameters), from the logical deductions as to course of action to be taken therefrom, and then the execution, there is little hope of improvement of the process. Emery (1965) has waxed fervently and at length on this subject. We agree with him.

*Real-time systems: The new information technology**

Our contention is that the new computers offer capabilities that enable much easier construction of intelligent management information systems. Both the quality of the data base and the power of the procedures that can be brought to bear on it can be materially improved.

Consider first the now generally available facility for "on-line, real-time" data processing in general and real-time operational control in particular. Since real-time processing implies up-to-the-minute recording of system-wide individual transactions (status changes), it provides uniquely a *current, global* data base (for operations). In other words, the current status of all operating system activities is known. Furthermore, since all activities are "on-line" to the central processor, access to large-scale computational power can be granted in order to respond to transactions as they arise. This has distinct implications for the situation in which the desired response is in the form of control directions which, recall, are made on the basis of a process model.

*This section of the paper is a partial synopsis of Carroll (1966).

Time-sharing is a product of the same technology, being essentially on-line, real-time computation for multiple users. It provides for man-machine interaction in problem solving without creating idle time. The close coupling thus afforded means that there exists a flexible division of labor between man and machine, the man bringing to the process those attributes in which he excels in close cooperation with the superior computational powers of the machine.

Since real-time processing and time-sharing are based on the same technology, they are mutually compatible and both are compatible with conventional "batch" processing, it has been noted by Carroll (1966). Consequently, we can hypothesize the near-term existence of generalized computers which possess real-time processing and time-sharing capabilities in which:

> . . . *whatever permutation or combination of human and machine problem solving attributes is needed can be supplied with data inputs of whatever quality of currency or scope is desired* (*Carroll*, 1966, *p.* 10).

That these things are good in general is undoubtedly true, but they are particularly useful for the pursuit of intelligence, we assert. In operating process control, we have noted the need for detailed data and functional-temporal association thereof. The global scope and currency of the data in these generalized systems meet this requirement. Furthermore, the availability of these data, coupled with the computational power in real-time, means that quite complex and hence potentially more valid process models can be utilized in control. And finally, we have noted that inductive inference is a faculty limited, for the moment at least, largely to human beings, yet it is the fundamental process of increased understanding. Through the new technology, a human can be closely coupled to operating process information; he can monitor the process and exercise his superior capabilities for pattern recognition, abstraction, hypothesis formulation and test—in short, his inductive powers. He need not await accumulation of evidence; he is on-line to the operating process even though it may be geographically dispersed. As we will observe shortly, this testing capability is very important. Once recognition, reorientation and understanding of the process is established, it may then be introduced into the system itself for increased sophistication and intelligence.

Some of these points can be illustrated by consideration of the application of real-time processing to the job shop production control problems cited earlier. In general, the simulation-based schedules were noted as providing superior control because they provide a more valid model of the process as carried out. Obviously if the model is valid, departures from schedule mean something. But this is true for only a short period after the new schedules are computed. In time, minor deviations from schedule accumulate, machines break down, workers are absent, and as a result, the model (i.e., the schedule) and the real world begin to diverge. This is the "decay" to which we referred previously. In a real-time production control system, detailed decisions on product movement, relating to sequencing and routing of jobs, could be performed by the computer (using the same type of decision rules employed in the simulation). But because the status of the system is continually updated, the decisions are made on the basis of *true current status* as opposed to the *predicted status* used in the simulation approach. Consequently, "decay" would not occur as easily and the resulting control system theoretically would be more effective.* Also, we would have a more useful system for diagnosis because deviations from expected behavior would more likely mean *precisely* that something other than the model is wrong. Analysis of causes could therefore be undertaken without risk of a wild goose chase, and the fact that investigation can take place immediately provides unparalleled opportunity for accurate reconstruction of the "crime," it is conjectured. But good diagnosis does not stop with crime reconstructions. Its greatest value lies in its educational aspects. The more confident one is of cause and effect relationships the stronger pattern association and the faster the remedial action.

Diagnosis is, of course, often a more subtle process than is implied by the running down of variations from plan. It often requires "browsing" through historical data, classifying, normalizing, rearranging and the like. The flexible interaction feature of time-sharing provides great convenience and power for so doing. Being able to think between interactions with the computer is at the heart of the concept of "man-computer symbiosis" advanced by Licklider (1965) among others.

The advantages of this new technology are perhaps even more marked in the planning process control domain. First of all, planning itself is a natural man-machine process, it has been frequently noted (Carroll, 1965b; Emery, 1965. Simply being able to intertwine the heuristically well-endowed, intuitive and subjective planner with his model offers enormous advantages. But the greater advantage may come in the metamodeling process, that is, modeling the planner's behavior for purposes of ultimate improvement (assuming his use of a computer model). Capturing the planner's "trace," the detailed sequence of steps he takes in arriving at a decision, is quite possible. Given the planner's cooperation, it is simply a question of obtaining the hard copy transcript of his session with the computer model, for example. And, of course, the diagnostician is equipped with a unique linkage to the process he is attempting to understand.

Exploitation of the power of these generalized systems for operating process control, planning, and planning process control has been the subject of research by Carroll (1965b) and colleagues at Project MAC.

Research in intelligent information systems

When we view where we stand in relationship to our goal of intelligence, we realize just how little is known about the techniques, the economies, and, broadly, the phenomenon of intelligence.

We list below a few areas of research which we feel represent promising starting points for improvement in this regard.

Data base and information systems for intelligence

One of the prerequisites for the implementation of intelligent systems is an efficient data base. For this we need better understanding of the data required for improved understanding. We have stated the general specifications for detailed and asssociative data; but there are numerous questions begged by this, such as what detail and what means for association. In short, we need some operational specifications (subjected to economic analysis hopefully) and some demonstrably better mousetraps in the data structure domain. There is a dilemma involved in the question of what detail, for example. One simply cannot specify *a priori* what detail or even what variables to measure until unexplained differences, problems (or successes) occur, and hypotheses are generated. What needs to be established is the usefulness, for diagnostic purposes, to provide guidelines on collection of possibly relevant data (as opposed to already known relevant data). In short, some theory is needed.

Some general research in the structure of the associative data bases and the procedures for exploiting this association is in process by Zannetos and Sahin

*The relative effectiveness of real-time versus periodic scheduling was tested by Kogan (1966). In the cases studied, the theory was found to be valid. Of course, we must admit that the comparisons are influenced extensively by the expertise of the one who simulates. Even so, since we are dealing with non-deterministic systems, real-time control will out-perform controls based on fixed theoretical models.

(1966). This can be described briefly (and speculatively) as follows:

The data base requirements for an associative information system will be mainly the same as those of a functional accounting system previously described, with one major difference. Instead of using raw data as the indecomposable modules for storage, manipulation, and causal association, patterns or configurations of data will now be used. The faculties of understanding, we might even say "consciousness," loom into prominence and somehow must be captured and incorporated in the data base by means of these patterns.

To get at the question of procedure, assume that we have an organization with well-established objectives and a dominant (i.e., chosen) plan to accomplish them. Given the dominant plan, we assume that the organization will be able to specify the operations that are necessary to achieve its objectives. Now for each dominant plan there must be a given configuration (pattern) of resource utilization, which will best implement the dominant plan, at least on an *a priori* basis. With the dominant resource configuration established, a dominance ranking of these resources can be made in turns of a one-dimensional index. Such ranking may be in terms of opportunity costs or loss functions. We are only interested in the *dominant* plans, resource configurations, and resources, and in proximate ordinal rankings. (The hypotheses which the system will generate and the search which will follow for testing, all of which are part of the system, will compensate for such approximations.) Furthermore, we are only interested in probabilistic associations.

The next requirement is the association of resources, at the point of acquisition, with the various (major) attributes of such resources. These sets of attributes are given a temporal index and also contain entries indicating the major physical characteristics of resources (among which are cost and capacity information). The attributes of resources are ranked once again according to dominance which obviously is dictated by the dominant plan.

Each one of the resources in the dominant configuration, no matter what its ranking therein, may be the dominant resource in another configuration of subordinate resources, as well as non-dominant member of other configurations. By means of this "dominance" procedure a hierarchy of associations both vertical and horizontal is established.

With the above as a brief description of the system, let us now look at (diagnostic) hypothesis generation at the operating level, because this is one of the greatest attributes we wish to impart to the information system. The signals which trigger hypothesis generation, are of at least three kinds. They may originate in:

- The difference between resource utilization (both quantities and attributes used) as specified in the prior dominant plan (model) and as reflected in operations.
- The dominant resource configuration of a proposed plan, if it does not use the most dominant characteristic of each of the resources proposed for the implementation of the plan. (If the new plan, after search, is still found to be dominant then an updating of the resource-attribute vectors will be necessary to incorporate the latest ranking.)
- The presence of "slack" in some dominant resource which will necessitate a change in the opportunity cost of this resource and a temporary change in the dominance rank. (The system scans for slack in capacity starting from the most dominant resource downward.)

Once the signal is received, on the basis of its content, the system immediately associates at least two patterns with it: the highest hierarchical pattern where the resource appears, whether in a dominant role or not, and the pattern in which it is the most dominant resource. Now the search begins for term by term comparisons of the prior patterns (plan) and those derived from operations or included in the proposed plans, and hypotheses are tested. Depending on the results of these tests the descriptive sets of resource attributes may be rearranged and assigned new temporal designation for subsequent reference. Also, the data base is hierarchically reorganized.

In addition to its diagnostic properties, this system also holds promise for providing information for prognostic purposes. By studying the intertemporal changes in the resource-attribute vectors, the system may now generate hypotheses and test for the existence of patterns of relationships which can be used for planning process control (and also assess efficiency of the decisions by bringing everything back to the point of origin and thus facilitate learning). For this latter task we need systems with man-machine interaction features.

In order to operate efficiently, a system such as the one described here cannot obviously depend on "brute force" or exhaustive sequential search, because of immense combinatorial problems. We suspect that we must use, therefore, parallel search techniques or some hybrid system.* Also parallel processing capabilities are desirable for reasons of efficiency. As for the cues

*Selfridge and Neisser (1963) have commented on the relative merits of the two search strategies.

that trigger pattern retrieval and association, they must not refer to locations of stored messages but to the conent. Finally, we noticed that there is a need for some hierarchical organization of the data base with distributed logic. This relative decentralization allows flexibility for learning and self-organization, but also necessitates functional association of the various modules of the data base. The problem of deciding how much logic is to be distributed and where is not an easy one to solve. We believe, nonetheless, that it is not unlike other organizational problems, so the theory and techniques suggested elsewhere for aiding in the design and evaluation of the organization structure are also applicable in this case (Zannetos and Carroll, 1966; Zannetos, 1965a, 1965b).

Theories of diagnosis and decision-making

We have noted that the "metamodeling" problem of planning process control necessitates modeling the planning process itself. But planning is a decision process, so this amounts to modeling a decision-maker. This has been an active area of research for some years now, notably by students of Simon and Newell such as Clarkson (1962). Other approaches have been studied by Bowman (1963) and his students. However, no research has been directed to modeling the type of modeling process involved in planning and we think this would be useful.†

Moreover, diagnosis, in the sense that we have employed the term, is a poorly understood process at best. There is much work going on in medical diagnosis, but unfortunately for our purpose this is what I might be called "discriminatory diagnosis," in which the relationship of symptoms to diseases is taken as known (probabilistically), rather than "inductive diagnosis" in which no such relationship is available.‡ However, work in medical diagnosis will undoubtedly provide some general insights. Particularly promising is the research of Gorry (n.d.) who is attempting to create a general, diagnostic model ("general" means environment independent—applicable to sick people, sick cars, sick computer programs). His emphasis is on discriminatory diagnosis, but we suspect that the data structures and much of the logic of his procedure are applicable to the less well-structured inductive diagnostic problem as well.

In addition to understanding individual decision behavior, we have noted that planning is often a group process; it is performed by a "team." Team decision making is not well understood. The pioneering work of Radner (1959); Marshak (1955) and Kriebel (1963) brings some organization to this area and the recent work of Clarkson (1966) in descriptive (computer simulation) approaches to group decision-making is directly relevant to the problem. The coming general availability of time-sharing, which enables group cooperative, interactive problem solving and monitoring, should greatly facilitate research in this area.

†Newell and Simon (1963) did incorporate a "planning" mechanism in their "General Problem Solver," it should be noted, but their type of planning and ours are only generically related.

‡What must be supplied in inductive diagnosis is the hypothesis of relationship, a task we have already allocated to the man in the man-machine partnership.

SUMMARY AND CONCLUSIONS

We have specified some features of information systems which are capable of increasing management's understanding of its environment and its rationality in coping with it—its intelligence. In so doing, we have focused our attention first on operating process control, in which intelligence is increased by causal induction, diagnosis of the factors which underlie process behavior. This is conveniently framed as improving the model of the process. We then discussed the higher level problem of planning process control which we noted was typical of higher order intelligence problems. This too involves model improvement, but in this case it is improving the model of what is a modeling process itself.

We have reduced our discussion to size by ignoring some aspects of system design. For example, we have ignored the general dimension of information availability. There exist in this area several issues, to wit: Should information relating to detailed performance of lower level organizational subunits be freely accessible to higher level managers? To what extent should the opposite take place? Should parallel organizational units share data on their status and performance? These are real issues which are related in part to "managerial style" but they also impinge on organizational intelligence. The "multiple split personality" aspect of organizational behavior is, we suspect, intimately linked to the question of dissemination of information.

Another area that we have ignored encompasses the perennial issues of "cost and value" of the generated information. No doubt, trade offs must be established between the cost of the system, detail and purity of information, reality of representation among other features, and the objective as well as the often subjective utility of the results. All these issues we chose to leave outside the purview of this presentation for reasons of expediency and without prejudice.

Within the scope of the general problem we have attacked, we first delineated the basic processes underlying control systems and stressed the significance of the diagnostic process in establishing cause and effect relationships. Understanding of causal dynamics is important to us not so much for its regulative power but for its educative role. Learning and updating of the models used are critical managerial functions especially for the planner.

Although we view planning and control as two hierarchical processes linked together hierarchically, for purposes of exposition, we separated planning process control from operating process control. We advocated control not only of operations but also of the planning process itself so that inefficiencies do not creep incipiently through the fixities of planning.

Formal models are prerequisite before any type of feedback control process takes place. While there is ample evidence of their use for operating process control there is very little indication that they are used for planning process control. Planning, therefore, remains as an *ad hoc* process mostly outside the regular information and control system. But even in the case of operating process control we found by examining the state of the art that the models used are mostly naive and capable of only cursory symptomatic control. Inductive diagnosis for establishing of causalities and changing the model and its parameters whenever necessary is in general missing.

In order to improve our present control systems for operations we advocated associative data bases with pattern recognition and pattern generation capabilities. Functional relationships (cause-effect), probability distributions, and procedural instructions (to be followed upon association) must be part of the data base.

For planning process control, in addition to the above specifications, we argued for system capabilities to recognize patterns in cases where ambiguity exists. This will give the manager sufficient lead time to prevent undesirable consequences, and also update the planning model itself. If the remedial action specifications and the model refinement are part of the control process then such a system we called prognostic.

Finally we suggested some steps for the realization of intelligent information systems. In the area of operating process control we believe that we have made enough progress and also have the technology (real-time systems) to improve significantly the intelligence of our present control systems. As for planning process control, we only speculated on the basis of our ongoing research designs and suggested a procedure for creating dominant patterns which will allow association and automatic hypotheses generation and testing. In addition to more progress in conceptual system design, theories of diagnosis and decision making, we see also the desirability for hardware and software which will allow associations on the basis of content and parallel search, parallel processing.

In conclusion, we believe that increased organizational intelligence is possible and greatly to be facilitated by new advances in information technology. Perhaps the greatest progress can be made, however, by simply recognizing that increasing intelligence is a legitimate goal for information systems design, and that there are some straightforward steps which can be taken towards that goal.

REFERENCES

1 ROBERT N. ANTHONY *Planning and Control Systems: A Framework for Analysis.* Boston: Division of Research, Graduate School of Business Administration Harvard University 1965

2 RICHARD BELLMAN *Adaptive Control Processes.* Princeton University Press 1961

3 WILLIAM A BLACK *Cost Models for Management Control,* unpublished masters' thesis, Sloan School of Management, Massachusetts Institute of Technology.

4 E. H. BOWMAN *Consistency and Optimality in Managerial Decision Making,* in J. L. Muth and G. L. Thompson, eds., Industrial Scheduling. Prentice-Hall Englewood Cliffs 99-112 1963

5 ROBERT G. BROWN *Smoothing, Forecasting, and Prediction of Discrete Time Series.* Prentice Hall Englewood Cliffs 1963

6 GILBERT BURCK, et al. *The Computer Age and Its Potential for Management.* Harper and Row New York 1965

7 DONALD C. CARROLL *Implications of On-Line, Real-Time Systems for Managerial Decision Making,* Paper presented at the Research Conference on the Impact of New Developments in Data Processing on Management Organization and Managerial Work, Sloan School of Management, Massachusetts Institute of Technology. (Also Working Paper 165-66, Sloan School of Management, MIT) 29-30 March 1966

8 DONALD C. CARROLL *Man-Machine Cooperation on Planning and Control Problems,* Paper presented at the International Symposium on Long Range Planning for Management, UNESCO, Paris, September, 1965. (a) (Also Working Paper 145-65, Sloan School of Management, MIT)

9 DONALD C. CARROLL *Simulation Research in On-Line, Real-Time Systems,* Paper presented at the TIMS Eastern Meetings, Rochester, N. Y. 15 October 1965. (b) (Also Working Paper 164-66, Sloan School of Management, MIT).

10 GEOFFREY P. E. CLARKSON *Decision Making in Small Groups: A Simulation Study,* Working Paper 194-66, Sloan School of Management, Massachusetts Institute of Technology 1966

11 GEOFFREY P. E. CLARKSON *Portfolio Selection: A Simulation of Trust Investment.* Prenctice Hall Englewood Cliffs 1962

12 JAMES C. EMERY *An Approach to Job Shop Scheduling Using a Large-Scale Computer,* Industrial Management Review. III (Fall 1961), 78-96

13 JAMES C. EMERY *The Planning Process and Its

Formalization in Computer Models, in Joseph Spiegel and Donald Walker, eds., Information System Sciences: Proceedings of the Second Congress. Spartan Books Washington, D.C. 1965 369 - 390

14 MARSHALL K. EVANS *Profit Planning,* Harvard Business Review, 37 (July-August, 1959) 45-54.

15 MORDECAI EZEKIAL and KARL A. FOX *Methods of Correlation and Regression Analysis.* Third Edition Wiley New York 1959

16 EDWARD A. FEIGENBAUM and JULIAN FELDMAN, eds. *Computers and Thought.* McGraw-Hill New York 1963

17 JAY W. FORRESTER *A New Corporate Design,* Industrial Management Review, VII (Fall, 1965) 5-14

18 JAY W. FORRESTER *Industrial Dynamics.* Cambridge: MIT Press 1962

19 ANTHONY GORRY *The Diagnostic Problem and Process,* Ph.D. Dissertation in process Massachusetts Institute of Technology.

20 FRANKLIN A. GRAYBILL *An Introduction to Linear Statiscal Models.* McGraw-Hill New York 1961

21 EDWARD H. HARVEY *A Study of the Formulation of Operating Cost Models for Engineering of Economic Analysis in the Telephone Industry,* unpublished masters' thesis, Sloan School of Management, Massachusetts Institute of Technology 1966

22 JAMES G. HOLLAND and B. F. SKINNER *The Analysis of Behavior.* McGraw-Hill New York 1961

23 JOHN N. KOGAN *Real-Time Sequencing versus Periodic Scheduling in a Job Shop,* unpublished masters' thesis, Sloan School of Management Massachusetts Institute of Technology 1966

24 CHARLES H. KRIEBEL *On Normative Mathematical Models of Information Systems,* Working Papare 38 - 63, Sloan School of Management Massachusetts Institute of Technology 1963

25 J. C. LICKLIDER *Man-Computer Partnership,* International Science and Technology (May, 1965), 19 et seq.

26 JACOB MARSCHAK *Elements of a Theory of Teams,* Management Science I (aJnuary, 1955) 127 - 137

27 MARVIN MINSKY *Steps Toward Artificial Intelligence,* in E. A. Feigenbaum and J. Feldman (Eds.), Computers and Thought. McGraw-Hill New York 1963 pp. 406 - 452

28 ALLEN NEWELL and H. A. SIMON *GPS, A Program that Simulates Human Thought,* in E. A. Feigenbaum and J. Feldman (Eds.), Computers and Thought. McGraw-Hill New York 1963 pp. 279 - 293

29 ALLEN NEWELL and H. A. SIMON *The Simulation of Human Thought,* Current Trends in Psychological Theory University of Pittsburgh Press 1961 152 - 179

30 WILLIAM F. POUNDS *The Process of Problem Finding,* Working Paper 147 - 65, Sloan School of Management Massachusetts Institute of Technology 1965

31 R. RADNER *The Application of Linear Programming to Team Decision Problems.* Management Science VI. January 1959), 143 - 150

32 HOWARD RAIFFA and ROBERT SCHLAIFER *Applied Statistical Decision Theory.* Division of Research, Graduate School of Business Administration Harvard University Boston 1961

33 F. J. ROETHLISBERGER and W. J. DICKSON *Management and the Worker.* Harvard University Press Cambridge 1939

34 ROBERT SCHLAIFER *Probability and Statistics for Business Decisions.* McGraw-Hill New York 1959

35 OLIVER G. SELFRIDGE and ULRIC NEISSER *Pattern Recognition by Machine,* in Feigenbaum and J. Feldman (Eds.), Computers and Thought. McGraw-Hill New York, pp. 237 - 250

36 HARRY W. STEINHOFF JR. *Daily System for Sequencing Orders in a Large Scale Job Shop* in Elwood S. Buffa, ed., Readings in Production and Operations Management. Wiley New York 1966 155 - 165

37 D. R. TRILLING *The Use of a Job Shop Simulator in the Generation of Production Schedules,* AFIPS Conference Proceedings Vol. 26: 1964 Fall Joint Computer Conference. Spartan Books Inc. Washington, D.C. 1964 277 - 290

38 ZENON S. ZANNETOS and DONALD C. CARROLL, *Information-Technological Pressures in Organizational Design.* Working Paper 219 - 66 Sloan School of Management Massachusetts Institute of Technology 1966

39 ZENON S. ZANNETOS *Managerial Information Systems for Planning and Control* Working Paper 210 - 66, Sloan School of Management Massachusetts Institute of Technology 1966 (a)

40 ZENON S. ZANNETOS *Measuring the Efficiency of Organization Structuress Some Implications for the Control System of the Firm,* Working Paper 117 - 65 Sloan School of Management Massachusetts Institute of Technology 1965 (a)

41 ZENON S. ZANNETOS *On the Theory of Divisional Structuress Some Aspects of Centralization and Decentralization of Control and Decision Making,* Management Science. 12 (December 1965) 49 - 69 (b)

42 ZENON S. ZANNETOS and KENAN SAHIN *Some Thoughts on the Design and Implementation of Associative Information Systems for Planning and Control,* Working Paper 220 - 66 Sloan School of Management Massachusetts Institute of Technologoy 1966

43 ZENON S. ZANNETOS *Standard Costs as a First Step to Probabilistic Controls A Theoretical Justification, an Extension and Implications,* The Accounting Review XXXIX (April 1964)

44 ZENON S. ZANNETOS *Toward a Functional Accounting Systems Accounting Variances and Statistical Variance Analysis,* Industrial Management Review. 7 (Spring 1966) 71 - 83

45 ZENON S. ZANNETOS *Toward Intelligent Management Information Systems,* Working Paper 155 - 66, Sloan School of Management Massachusetts Institute of Technology 1965 (c)

46 EDWIN V. W. ZSCHAU *A Technique for Comparing the Performance Results of Operating Units Within a Decentralized Organization.* Working Paper No. 41, Graduate School of Business Stanford University 1964

Guidelines for simulation model development

by JAMES L. MCKENNEY
Harvard University
Cambridge, Massachusetts

INTRODUCTION

This paper proposes a set of guidelines for developing a simulation model for planning. The guidelines were formulated and tested in the process of developing several industrial simulations. One conclusion of this experience which seems acceptable to most individuals involved in creating analytical abstractions, is that better abstractions result when the decision makers assume an active role in the development of the abstraction. These suggested guidelines have facilitated the involvement of the decision maker in model development in addition to being useful in the development of planning models. The reasons for proposing them at this time is to focus attention on the importance of utilizing the decision maker's concept of his environment, and the necessity that all individuals concerned with the construction of a simulation model understand the developmental nature of the project.

Our guidelines for model building are defined as the attitudes, policies and postulates followed by those involved during the design and development of an abstract representation of an environment. A premise of the guidelines is that better models are the result of a joint developmental effort of the individuals who are to use the model as a tool (the planners) and the individuals who are designing and programming it (the modelers). The guidelines are as follows:

- The simulation model is conducted as a developmental project to aid the planning process.
- An important characteristic of the project is the evolution of goals, uses, and specifications of the model as it relates to the planning process.
- The planner's intuition is the most appropriate reference of the pertinent environment; therefore, it is the role of the model to improve the consistency of the planner's intuition and to make them aware of new information requirements.
- A prime function of the model is to amplify the intuition of the planner by generating a spectrum of analyses for certain codifiable conditions.

There were other guidelines we followed in the development of planning models, but they either seemed unique to the situation or too ambiguous to sensibly defend. An attitude which is essential to all successful model building is the eventual faith in the model as a tentative statement of causal factors influencing the resources to be allocated. There are several published sermons on the necessity of the belief in the order of nature, and we will not dwell on the subject (Ackoff, 1962; Buzzell, 1964). Perhaps the above guidelines can be considered a few of the necessary conditions to engender a planner's conviction.

A discussion of the guidelines

The sole criterion of success in our discussion of simulation models and planners is that the planner utilize the model to improve his understanding of the world in order to allocate resources. An unused model, no matter how elegant, is a failure. It may be that the significant "use" a planner makes of a simulation model is in its development to refine his own concept of his problem. The discussion below deals with how the guidelines serve to encourage the planner to contribute to the development of a model and what the modeler should have in mind to facilitate this contribution.

The involvement of pertinent individuals in the developmental process of a model is critical for two reasons: (1) to generate and distill an appropriate data base from which a pertinent model can be created; (2) to develop a commitment to the model as a professional managerial goal. The first reason is obvious and would probably not require a great deal of the planner's time. However, developing a feeling of commitment is just as critical as "truth" and requires an adequate incubation period for modelers and planners alike.

The planners and modelers should individually consider the simulation project as a means to increasing their understanding of the operation of the firm and thus their influence on the operation. To gain the most insight from the project each individual should work to improve the effectiveness of the model as it relates to his activities.

It is critical that all pertinent personages understand the functioning of the model such that they feel a need to participate in defining aspects of the model germane to their sphere of influence and in responding critically about the formulation of the simulation. Given adequate participation the model can become a unique resource in that it represents the firm as an entity with few organizational boundries. This seems to generate a depersonalized professional attitude in the planner toward the model, which allows the model to be scrutinized for appropriateness in painful detail without bruising individual egos. A successful tactic to obtain this complete involvement is to have the simulation project a responsibility of the top planning executive.

As is obvious, having the boss head the project is but one step toward creating a decent simulation. It is necessary that all individuals concerned have a thorough understanding of the nature and potential of a simulation project. It is suggested that by constituting the project as a research and development venture on the managing process an orientation is provided to the concerned individuals in which all are expected to learn. This learning attitude is helpful when exploring how one can formulate heretofore non-explicit relations. In addition a developmental project by nature should commit a management to a sizeable budget over an extended period of time. The results of this expenditure are uncertain and therefore the project should regularly be appraised as to its effectiveness. This appraisal process is especially important in regard to simulations intended to aid the planning process.

The definition of criteria to evaluate improvement of the planning process is a difficult art and requires experimentation and attention. However, focusing on this aspect of the model's impact provides an appropriate perspective for considering the effectiveness of the model. The appraisal should allow adequate elapsed time for the development of a series of plans concurrent with the implementation of the model. During this time the defense for continuing financial support for the model probably rests on the degree to which it stimulates the management to consider their planning process. After the model is being utilized as an active aid, support should be judged on documental evidence produced by key planners. The model should not be judged solely on appropriateness of results, number of plans evaluated, or mechanics of operation. These can be modified by utilizing different resources to develop the model. It should be judged on how effective it is in improving the planning process. In essence a decent model seems to require a budget for an extended time during which periodic reviews are made of the decision process, not model details.

The evolutionary nature of planning models

The prime reason for a pre-ordained extended life of a planning model project is that the only constant characteristic of a simulation model is change. The product of this evolutionary process is assisted if the changing nature of the model is understood by all associated with the model from the very start of the project. Models with a tradition of change will encourage the planners to attempt to define hazy ideas and to experiment with formulating relationships, as conjectures can be changed if desired. It will also induce the modelers to design their procedures to accommodate changing definitions and specifications.

The most important reason for designing an adaptable simulation model development is the very survival of the simulation. An adaptable and changing model is essential if the model is to be used over an extended period of time. Assuming the model is to operate as an agent for improving the planning process, its main function may well be as a stimulus to search for a definition of what the planner has not included in the model. This improvement process seems to be one of continuous redefinition of the planner's concept of the pertinent forces in the environment and growth in the modeler's ability to adequately represent these forces. The model must continuously reflect the planner's improved concepts or fall into disuse by the decision makers. A successful simulation project for planning will stimulate the continuous growth of the participants as evidenced by an improving model.

The reference base utilized for the initial stages of model development is critical to the goal of generating a useful tool. It is suggested the most appropriate reference is the planner's definition of the pertinent influences in the environment and how he thinks they relate to each other.

A planner with significant budgetary responsibility is assumed to have had an adequate involvement with his environment to have developed an understanding of what elements are critical to the success of his operation. His measures of these elements often range from precise dollar figures to vague intuitive impressions, but all are important and real to him. It seems reasonable to accept the planner's notion of his business as fact and to attempt to substantiate his concept by programming a model to imitate the concept. Normally it is impossible to explicitly define all of the factors a planner considers. In addition individual planners will not be consistent with each other or emphasize the same aspects of the environment. To cope with these concomitant ambiguities, the model should be programmed to codify as many factors as possible with freedom for the planner to modify the impact and range of each variable. Where variables cannot be defined, provision should be made for direct planner influence as he sees fit. The goal of the modeler is to generate an abstraction that adapts comfortably to how the planner considers his resources allocation problem. The process of programming the well-defined variables should involve the planners and modeler in order that both:

- Evaluate the sensitivity of the environment to change in the selected variables.
- Discover new methods of combining variables.
- Resolve inconsistencies or ambiguities between planners and the environment.

This latter process often serves as a basis for data collection to define missing relations or to test in the environment the validity of assumed relationships. A clear definition of how the simulation functions is essential for the mutual consideration of relationships that govern the operation of the simulation.

An overt goal of most simulation projects for planning is that operationally the simulation model is to serve as an analytical tool for the planner. To serve effectively it must be formulated to produce results which are compatible with the planning procedures. Elegant reports can be generated, but care should be taken to well define the limits as well as the potential power of the model as an analytical engine. The modeler and planner should continuously appraise what is more economical and effective for the model to accomplish versus the planner. At this point in time it does not seem economically feasible to model completely an environment as effectively as a good human desiion maker. However, a simulation model can perform quickly and accurately a long involved sequence of well specified events to produce an answer in predefined terms. How the planner will use these answers is important in the development of the model and should be considered at each step of the program. The goal of the model developer is to develop a model which can amplify the planner's insight to a resource allocation problem. At present the method of amplification seems to be a prompt evaluation of a variety of plans under a range of assumed conditions which the planner defines.

Using the guidelines to develop a planning model

How these guidelines might aid the development of a simulation model is presented in the following synopsis of a simulation project in an industrial firm. The firm, a consumer goods producer, with sales in excess of $200 million was planning to enter the European market. This was its first venture overseas and the executives felt a need for improved decision-making procedures to cope with the unknown seemingly more complex situation. The staff aides to the executive committee had suggested a simulation model might assist the planners in allocating capital overseas to assure an orderly and profitable entry into the new market.

A series of seminars conducted by the staff with outside consultants was initiated to explore methods of planning in general and the potential of simulation models in particular. A topic of one of the seminars was the evolutionary nature of simulation models with the expectant change in understanding of their environment by the individuals using the model. This session produced a lively interest in developing a model tailored to the needs of the executive committee. The eventual result was a tentative five-year program for the improvement of the firms strategic planning.[2] A three-year capital budget of approximately $70,000 per year was allocated to the development of a simulation model which was to be defined by the executive committee. Progress review periods were to be held every six months by the executive vice president of sales who was responsible for long-range planning. All vice presidents of the firm and members of the executive committee were active members of the project and agreed to spend up to four hours per week to develop improved planning procedures.

The initial proposal developed in the seminar called for a tentative model to be operational in one calendar year at which time a redefinition of project goals and model specifications would take place. At the conclusion of the seminars the management concluded that the development of a simulation model for planning seemed to be a viable method of improving their planning process and could provide an analytical tool which could aid them considerably. The planning problem was to allocate probable capital resources over an extended time in such a manner as to most effectively enter and become established in the European market.

The initial model, as specified by the executive committee, called for a representation of the necessary resources measured in dollars and elapsed time to produce given quantities of goods and the demand generated by the world environment for the company's products by price and type of product. It was the responsibility of the planners to identify probable gross changes in the environment such as new models or competitive actions and to define possible responses in the use of the firm resources as inputs to the simulation. The model would then give a profit measure for each suggested response. The planners in this project were the vice presidents of the corporation responsible for the Sales, Manufacturing, and Research and Development. There were six active model builders of heterogeneous backgrounds. The group included a market researcher, economist, operations researcher, financial accountant, experienced staff planner and an analyst programmer.

The modelers began the project by attempting to codify what the planners felt were important influences on their planning decisions and what questions they wanted evaluated by the model. To accomplish this the first six months of the project included a series of meetings with each planner to define what he considered the critical aspects of the firm's environment and how they might be formulated in the simulation model. During the time these meetings were being held, a series of definition papers on how the relationships defined by the planners might be combined, what data would be required and what reports would be generated were circulated. These papers served as a basis for seminars with all modelers and planners to discuss what should be included in the simulation model to improve strategic planning. After the seminars small groups would often discuss a specific aspect such as how a new product should be represented in the production process. The seminars were followed by additional two- to three-hour individual planner conferences with two or three members of the group to insure the planner's ideas were accurately represented in the model and to explain how the model was functioning with other planners' definitions.

A continual effort was made by the modelers to define all terms as clearly as possible for inclusion in a glossary that all participants received. The glossary established a common understanding on words all too often not well-defined such as: assumptions, sensitivity, programming, and planning horizon. The glossary aided in educating the planner in a bit of modeling jargon and preventing the modelers from using terms without defining them. It was invaluable in documentation of the model.

Concurrent with the planner conference, data were collected as specified by the planners to define the resources of the organization in a manipulatable fashion for planning purposes. This required the modelers to work with the staff assistants of the planners in an analysis of present measures of what the planners felt to be important. The available accounting data did not prove sufficient and, therefore, new information had to be created and stated in accessible format. For example, data were collected and distilled to develop a productive capacity model which related the total cost and elapsed time of producing a given quantity of product to the mix of products the level of production. The elapsed time required to acquire additional productive capacity or change product mix was defined in accordance with how the manufacturing planner thought the capacity responded.

After six months of discussion and three months of data collection to formulate the planners' concept into a model, the programming of the model for computer manipulation was started. Simultaneous with our programming effort a second series of meetings were held with the planners on how they might utilize the simulation in their on-going planning procedures. It was felt important to maintain the planners' interest in model development and it was conjectured that during the programming process a few revisions in the planners' model would be necessary. The individual meetings soon became formalized into bi-monthly planning meetings to discuss the state of the model and how it might be used to evaluate alternative resource allocations. being considered for future two year periods. These discussions aided the modelers in defining appropriate time units, ranges of accuracy, specific output requirements and potential changes in the input variables. They served to keep the planners informed on the state of the model and its limitations.

As the model entered the final debugging stage the meetings focused more onto methods of testing the model for validity and formulating plans for evaluation by the model. In these later meetings the planners began to develop expertise in explicity defining a feasible range of circumstances which could be tested on the model. This in turn caused the modelers to improve the model's ability to accurately represent a set of conditions. The results of this iterative process was an awareness of the importance of experimental design and new insight to the evolutionary aspect of the simulation project. Most individuals were convinced it was a rewarding experience.

The strong commitment was fortunate as early simulation runs proved to generate quantities of useless output. The first simulations were intended to represent 10 years of sales experience in the international market. The simulations on the average produced rather bizarre sales figures and production demands. The one bright side was that the cash flows resulting from the sales were consistent with past experience. They rediscovered that the process of predicting dynamic demand factors and economic consequences for a 10-year period in a fairly codifiable fashion allows errors to accumulate to bias all results. The planners were not dismayed and suggested procedures which the modelers could incorporate which would aid in the understanding of simulation results. Typical error prevention procedures called for the planners to estimate for the next two, five, and eight years feasible product price ranges, and estimates of production capacity, given the present base of the company. These estimates served as minimum and maximum limits on capacity and sales. The model operated within these bounds to evaluate the proposed price structure, time of product introduction and other aspects of their plan. They then considered the output of the simulation in terms of these limits. If the output indicated the simulation results hit an upper limit and remained there, the planners discounted the answer, because of model deficiencies but would judge that the plan might be a better one than a plan which drove the model to the lower limits. These procedures have afforded a basis for jointly testing plans and their assumptions while evaluating the sensitivity of the simulation model to a variety of inputs in order to invesitgate the model's validity.

Present stage of simulation utilization

There has been an obvious growth in the attitude of the modelers and managers as to what should be in the model and what should be excluded. A few of the original factors included as determinants of available resources of influences on demand have been tested and found unimportant. But of more interest is the number of new factors that seem to be more basic and casual nature than our original factors. Originally population had been considered as a basic variable of demand. They are now considering age distribution, wealth distribution, geographical distribution, and other factors of the economy in a given country to consider its market potential. Many of these factors are still being tested to evaluate whether they are significant in the long run and probably some will be discarded. Continual evaluation of factors in the model including the definition of assumptions and defense or explanation of these assumptions is now accepted by modelers and planners alike. Finally measures specified at the start have been superseded by new ones. Specific dollar requirements and time specifications originally desired as outputs have been replaced by requirements of rate of market penetration or equity growth. In general most measures of performance are more sophisticated than when the project began.

The planners seem to be evaluating alternative plans with the model to support their intuition. They suggest that the model has improved their judgment by testing some variables which heretofore were thought very important and found wanting as indicators of future significant environmental forces. The model development in part has forced the planners to define their time assumptions explicitly and to codify cost assumptions to accommodate manipulation. This has resulted in part of the accounting system changing to accommodate an evaluation of plans rather than a reporting of the accumulated costs of past activities. This change has improved the firms planning procedures and given a better data base for developing an improved model.

At present the model can almost be considered a professional goal for the management of the company as they are committed to its future development. They do not rely upon it for specific decisions, but seem to feel it a useful tool for improving their planning procedures. Perhaps at some future date they will rely upon it as a partner in decision making as well as process improvement.

CONCLUSION

Most modelers are aware of the importance of involving the planners in the development of the simulation model to achieve an operational model. We have had a reasonable degree of success in integrating the planners in the project by using the planner's concept of his environment as the point of departure for the model and stressing the importance of change from the beginning. An impediment to more adequate rapport between modeler and planner is the state of present computer languages. The language of the program for the model has to be interpreted to the planner. This interpretation creates ambiguities and misunderstandings which limit the effectiveness of present simulations as a tool for most planners. Hopefully new computer languages will allow the modeler and planner to conceptualize the simulation model in the language it is to be programmed.

REFERENCES

1 R. L. ACKOFF *Scientific Method.* New York and London John Wiley & Sons, Inc. 1962

2 R. N. ANTHONY *Planning and control systems: a framework for analysis.* Division of Research, Graduate School of Business Administration Harvard University Boston p. 16 1965

3 R. D. BUZZELL *Mathematical models and marketing management.* Division of Research, Graduate School of Business Administration Harvard University, Boston 1964

The use of systems analysis in the acquisition of information systems

by FRANK H. ELDRIDGE
Executive Office of the President
Washington, D.C.

INTRODUCTION

In the past four or five years numerous attempts have been made to apply systems analysis techniques to problems of command control and communications (C^3) in the Department of Defense. Some of these have, it is generally acknowledged, been more successful than others.

One of the principal problems in the early 1960's was the lack of an adequate resource structure for C^3. Hard work since then has, to a large extent, solved this problem. Physical assets, total obligational authority and manpower have been specified in great detail for about 150 command control and communications programs, representing the entire effort of the Department of Defense in this area. Having accomplished this, studies of tradeoffs and alternatives can start from a common data base—a necessary prerequisite for useful cost and effectiveness analyses.

The definition of C^3 mission objectives has, also, been further clarified in the past few years. By the early 1960's the role of C^3 in nuclear warfare had received considerable attention. For instance, a number of models had been constructed showing the effects on mission objectives of various types of thermonuclear attacks leveled against military and civilian headquarters as well as radio and landline communications. From this emerged a better understanding of the use of mixed systems for survival and the need for more rapid and integrated intelligence to meet the needs of different levels of command under various conditions of thermonuclear attack.[1,2]

More recently a great deal of actual experience has been gained in the use of command control and communications systems in crisis situations where the intent is to achieve U.S. political objectives with the minimum possible amount of conflict escalation. Lebanon, Congo, Cuba, Panama, Santo Domingo have all provided opportunities to test our evolving systems and operational methods.

Many of the systems analysis methods developed by the Department of Defense for command control and communications are now being adapted to information system problems of other Federal Departments and Agencies. I will define information systems, here, as any network of facilities that acquires and/or processes data for use in controlling resources. In this sense an information system can be either purely automatic or it can contain manual elements for monitoring, display, control, or other command functions.

Associated with every decision are both ponderable issues and imponderable ones. The analyst should concentrate on the ponderable ones. The decision maker must consider both.

There are several alternative approaches open to the systems analyst who, once having identified the principal issues, undertakes to address them. He can collect and analyze raw data or he can build models of the system and synthesize data that will serve to answer the questions or he can do both. The analyst is advised to look at both the hardware and the software, or operational aspects of the system, in order to develop a balanced context for his study. He should relate the objectives and the missions of a support system, such as telecommunications, to the objectives and missions of the system that it supports.

The choice of suitable measures of effectiveness is a critical part of the analysis. He should be aware of what it should do. By choosing measures of effectiveness that are related to the missions of the system supported he can avoid selecting effectiveness criteria that relate to less important side issues.

Another critical function of the analyst is the definition, for the decision maker, of suitable alternative systems and methods of operation. The analyst should consider a large enough study context to allow for development of feasible alternatives and trade-offs and to avoid unsuitable sub-optimal decisions. For instance, different time phasings of any one program are important alternatives to be considered. However, the cost and effectiveness of alternative programs are also usually very relevant.

The analyst should call the decision maker's attention to uncertainties in his study and should, where possible, provide him with sensitivity analyses for important but uncertain phasing, effectiveness and cost factors.

Finally, he should, where possible, indicate which factors cannot be analyzed quantitatively within the time frame of the study or the context of the data available and, therefore, should be left to the judgment of the decision maker.

Background for a case study

Our office recently had the opportunity to analyze some interesting information systems problems dealing with the telecommunications used for dispatch and control of electrical power networks. I intend to use these as a case study here to show how system analysis is being generally applied to the acquisition of information systems throughout the Federal Government.[3]

In the case under consideration a program decision had to be made whether to buy a microwave system or to use common carrier circuits to control the intertie of electrical power networks in two widely separated areas of the country. Both Government-owned and privately-owned systems were involved. The purpose of these interties is to distribute to power-deficient areas the extra power potential in rivers that is available from the spring runoff of melting snows. A number of electrical power utility companies are participating in interties of this type. Most cases, it turns out, involve many complex technical, legal and economic problems.

It is well known, of course, that public utility companies, in general, differ from other types of businesses. They supply transportation, water, gas, electrical power, telecommunications and other types of services that are required by the entire community. In order to provide a universally high quality of service, to avoid redundancy, and to produce economies of scale, these companies are usually legal monopolies. They are granted franchises to operate in any given area. In return, the Government retains the right to regulate their standards of service and their tariff rates.

Normally, each type of public utility uses services supplied by the others. For instance, the telephone utilities' primary source of power is obtained from the electrical power utilities. In addition, telephone companies maintain standby generators and batteries to provide backup power in case their primary sources are disrupted, such as occurred last Fall in the Northeastern part of the United States.

Many of the power utility companies obtain their telecommunications for primary control of their power networks from the telephone companies and find that this type of service satisfies their requirements. However, many other power utilities throughout the United States currently own and operate microwave telecommunication systems to provide primary communications for control of their power networks. The total investment in microwave systems in the United States for this purpose is estimated at several hundred million dollars and can be expected to increase as large electronic computers are applied in the future to control systems for these types of networks. Backup telecommunications for power control are now provided both by some power line carrier telephones owned and operated by these power utilities and, in many cases, by telecommunication services obtained from the telephone companies.

The power companies that do own and operate their own microwave systems use many arguments for doing so. They state that the cost of owning such a system is lower than obtaining this service from the telephone companies; that the telephone companies are not completely responsive to their requirements, particularly with respect to the reliability and availability of communications that they need for vital links for control of their power system. They point out that the power networks use extremely high voltages which present many safety problems and are difficult to control. They are particularly concerned with the safety of repair and maintenance men working on transmission lines, the stability of the AC power networks, and the proper performance of the new types of DC power intertie transmission lines. They stress the fact that they require overall system responsibility for their power networks. The normal mode of operation for present day systems is necessarily automatic and requires an integrated communications and control system with fast reaction and rapid response to control signals. For instance, in systems containing about 1,000 miles of electrical power transmission lines, surges in the networks, caused by lightning and other disturbances, will build up to a maximum in about half of a second. Therefore, circuit breakers must be operated throughout the system in a few tenths of a second, to prevent these surges from burning out critical components such as transformers, generators and insulators. As more interties are created throughout the United States and Canada, the intertied networks will become larger and more stable. Simulation models indicate that the time constants of the surges will increase. Under these conditions the overall reaction time for circuit breakers will not need to be as fast as for the uncoordinated networks. But the overall control of the larger networks will become more important since, when intertied, larger sections of the country could be blacked out if control of the intertied network is disrupted.

On the other hand, the telephone industry's position is that their companies are public utilities too, and that they have been granted franchises to supply common carrier services throughout specified areas. They have large pools of specially trained, professional communicators for maintenance and operation of existing systems and for R&D, design and engineering of new systems. They have a large dispersed network of communications and a sizable staff of maintenance men and extra equipment that can all be used to reroute and restore circuits that are disrupted in times of disaster. They argue that they have economies of scale that produce cost advantages over smaller privately-owned or Government-owned systems. Further, they say that they fully realize the critical nature of the power utility communications and in recent years have taken special steps to improve their service through, for instance, the use of special operational control rooms for monitoring and managing the restoration of important telecommunication circuits for electrical power control.

ODTM systems analysis

In cases of this type the Office of the Director of Telecommunications Management (ODTM) recognizes and emphasizes the importance of preserving the Government's freedom of choice between obtaining common carrier services or buying Government-owned telecommunications systems. This, it is felt, makes it possible for the Government to take into account all current factors in making a decision on how to obtain satisfactory services for the lowest cost. However, as a general rule, the policy of the Federal Government, as stated in *Bureau of the Budget Circular A-76,* is that the Government should rely on the private enterprise system to supply its needs.

Exceptions to this policy cited in the Circular are cases where:

- Procurement of a product or service from a commercial source would disrupt or materially delay the agency's program.
- It is necessary for the Government to conduct a commercial or industrial activity for purposes of combat support or for individual and unit retraining of military personnel or to maintain or strengthen mobilization readiness.
- A satisfactory commercial source is not available and cannot be developed in time to provide the product or service when it is needed.
- The product or service is available from another Federal Agency.
- Procurement of the product or service from a commercial source will result in a higher cost to the Government.

Interpreting the applicable policy in terms of the case studied, indicates that the Government should lease the intertie microwave system unless one or more of the following conditions exist:

- The telephone companies that proposed to supply the service could not meet the schedule phasing deadline for completion of the electric power intertie.
- The telephone companies could not supply an effective telecommunications system for control of the electric power system.
- The telephone companies' service would cost more than a Government-owned telecommunications system.

The applicability of these conditions formed the basis for our systems analysis of these cases.

Schedule phasing

Analysis of the first condition showed that although the initial investment in the telecommunications for the electrical power intertie would cost only about $2 million, the estimated loss in revenues would amount to almost $0.5 million per month if adequate control mechanisms, including required telecommunications, were not completed by the time the power intertie was ready for operation. Schedule phasing for telecommunications was, therefore, a critical although not an overriding factor in the decision.

Because the telephone companies planned to use existing facilities for part of the required system, and because of preplanning and scheduling flexibility in the growth of their nationwide microwave system, the telephone companies were willing to commit themselves to meeting the operational deadlines for the power intertie at the time the decision was made to proceed with the acquisition of the telecommunications system. The suppliers of equipment for the proposed Government-owned systems, on the other hand, could not guarantee an operational telecommunications system until about 45 days after this deadline, although they felt they might be able to decrease their time requirements if production scheduling, and possible installation delays caused by weather, permitted.

Effectiveness of telecommunications

As indicated previously, one of the most critical functions of telecommunications in this type of task is to provide a means for disconnecting an AC power network if transients or surges threaten to overload and burn out the elements of the network. A simplified diagram of a typical control system is shown in Figure
diagram of a typical control system is shown in Figure 1.

If a transient occurs, an overload detector, such

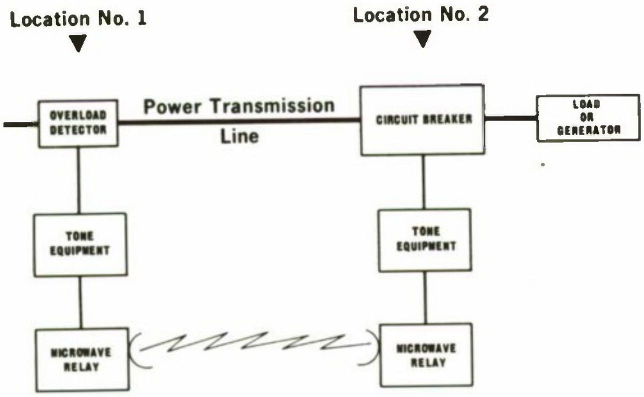


Figure 1—Typical power line control system

as a potential or current transformer, generates a signal that is fed to a tone equipment where it is used to modulate a carrier frequency. The output of the tone equipment is transmitted over a communications system to a tone equipment at another location. Here the carrier is demodulated and the signal used to actuate a circuit breaker for disconnecting power line loads or dropping power generators at that point.

The objectives of the telecommunications portion of this system are, therefore, to insure that elements of the electrical power system will not be destroyed by insufficient rapidity of control response and that telecommunications reliability will not significantly degrade the reliability of the system to either act when action is required or to not act when action is not required.

Measures of effectiveness of the system that were studied, therefore, included:

- Time delays from detection of an overload in power transmission line at Location No. 1 to the completion of the circuit breaker action at Location No. 2.
- Reliability of the system in actuating the circuit breaker when an overload signal is received.
- The probability that noise generated in the system will not accidentally trip the circuit breaker.

It is important to investigate not only the effectiveness and uncertainties of the telecommunications portion of the system but also the overall effectiveness and uncertainties of the system described in Figure 1.

The relative contribution to the overall time delay of the various telecommunication system alternatives under consideration can then be determined. Also it is important to compare the uncertainties in effectiveness of the overall system with the variations in effectiveness produced by the alternative modes of telecommunications in order to determine the relative benefits of introducing each of these alternatives into the system.

Time delays

The time delay in the system, T_o, will be

$$T_o = \sum_{t}^{m} T_n$$

where T_n = the time delay in the *nth* component in system.
m = the number of components in the system.

In other words, the total time delay is equal to the sum of the time delays in the individual components. Typical time delays are shown in Figure 2 for the components of the system indicated in Figure 1.

Our analysis indicated, therefore, that the time delays in the overload detector and the circuit breaker and the uncertainties in these time delays overshadowed the time delays and uncertainties in the communications portion of the system (*i.e.*, the two tone equipments and the microwave link). An obvious conclusion, therefore, is that, if reductions in total time delays and time delay uncertainties are required, improvements should be made in the overload detectors and circuit breakers rather than in the telecommunications portion of the system. For instance, one issue that arose during the analysis was whether wideband tone equipments should be used to reduce total time delays in these components from between 22 and 24 m*s* to between 14 and 16 m*s*.

Overload detector	10 - 16 ms
Tone equipments (two)	14 - 16 ms
Microwave link (200 miles)	3 - 5 ms
Circuit breaker	33 - 41ms
T_o	60 - 78 ms

Figure 2—Time delays in system components

Our analysis indicated that while this improvement would be desirable, high quality voice band communication links and terminals would not improve the quality of the circuit by any significant amount. This point is discussed in more detail in a section entitled *"Noise."*

Another issue that was related to the decision was whether microwave would be required throughout the network, and particularly on the longer links, or whether toll cable links could be used without degrading the system performance. Typical time delays for microwave links are about 2 m*s* per hundred miles of route and for loaded toll cable systems about 6 m*s* per hundred miles of route. It was estimated, therefore, that for control circuits not greater than about 300 miles in length, either loaded toll cables or microwave or a mixture of the two could be used interchangeably without significantly degrading the time delay of the total system. Only in exceptional cases will these types of control circuits be greater than 300 miles in length.

In the case studied, both the Government-owned and the common carrier alternatives proposed to use microwave only. Some additional delays would have to be added to the common carrier circuits because some of the links would pass through telephone company central offices. However, the common carrier network alternative was designed to minimize these types of delays. In no case did the planned total time delay exceed 5ms for control circuits in this alternative. This factor, therefore, was not considered to be significant in reaching a decision to buy or lease the system.

Reliability

The composite reliability R_o of the system shown in Figure 1 will be

$$R_o = \prod_{1}^{m} R_n$$

where R_n = the reliability of the *nth* component of the system.

In other words, the composite reliability of the system will be the product of the reliabilities of the individual components. Typical reliabilities of components are shown in Figure 3 for the system indicated in Figure 1.

Overload detector	0.9950
Tone equipments (two)	0.9995
Microwave link (200 miles)	0.9963-0.9997
Circuit breaker	0.9810
R_o	09719 - 0.9752

Figure 3—Reliability of system components

A study has been made that compared the reliability of 38 Government-owned telecommunication circuits with 46 common carrier circuits over the period of one year. Each common carrier circuit passed through an average of three telephone company central offices. The Government-owned circuits did not pass through any central offices. Figure 4 compares the number of outages per year and the outage durations for these two categories of circuits. Although the average rate of outages was about the same for the two types, the average outage duration of the common carrier circuits is seen to be significantly smaller than the Government-owned circuits. In other words, whenever an outrage occurred it took less time for the telephone companies to fix their circuits than for the Government to fix its circuits. The average reliability of the Government-owned circuits was found to be 0.9963 compared to 0.9997 for common carrier circuits.

Dual routing, using both microwave and cable, can be used to improve the reliability of any telecommunications system, if required. This has been successfully accomplished by the Power Authority of the State of New York and others. The reliability of dual routed circuits, assuming that they are completely independent, can be expressed as follows:

$$R_x = 1 - (1 - R_y)(1 - R_z)$$

where R_y and R_z are the reliabilities of each of the dual circuits. For instance, if $R_y = 0.9963$ and $R_z = 0.9997$, then $R_x = 0.999999$. Another way of looking at this is that it would reduce the expected outage time from 1,945 minutes per year and 158 minutes per year on the individual circuits, to only about 1 minute per year on the dual routed circuit.

However, the important conclusion here is that no significant gain in overall system reliability can be achieved by increasing the reliability of the communications components alone, if the reliability of the overload detector, the circuit breaker and other system components cannot be increased to a comparable level. For instance, in the case studied, if the communications system reliability is increased to, say 0.9999 the reliability of the overall system would still be less than 0.98.

Noise

The composite probability, P_o, that the circuit breaker will not be accidentally tripped by noise in any of the elements of the system is

$$P_o = \prod_{1}^{m} P_n$$

where P_n = the probability that noise generated in the *nth* component of the system will not trip the circuit breaker.

The tests described below were used to determine conditions under which noise, introduced into the system by the telecommunications components, could trip the circuit breaker. Unfortunately, no data were available to determine the probability of tripping the circuit breaker by noise generated in the non-telecommunications components of the system.

Actually, it was planned to use the same type of tone equipment in either the Government-owned or the common carrier system. The main issue, therefore, was whether conventional telephone channels could be

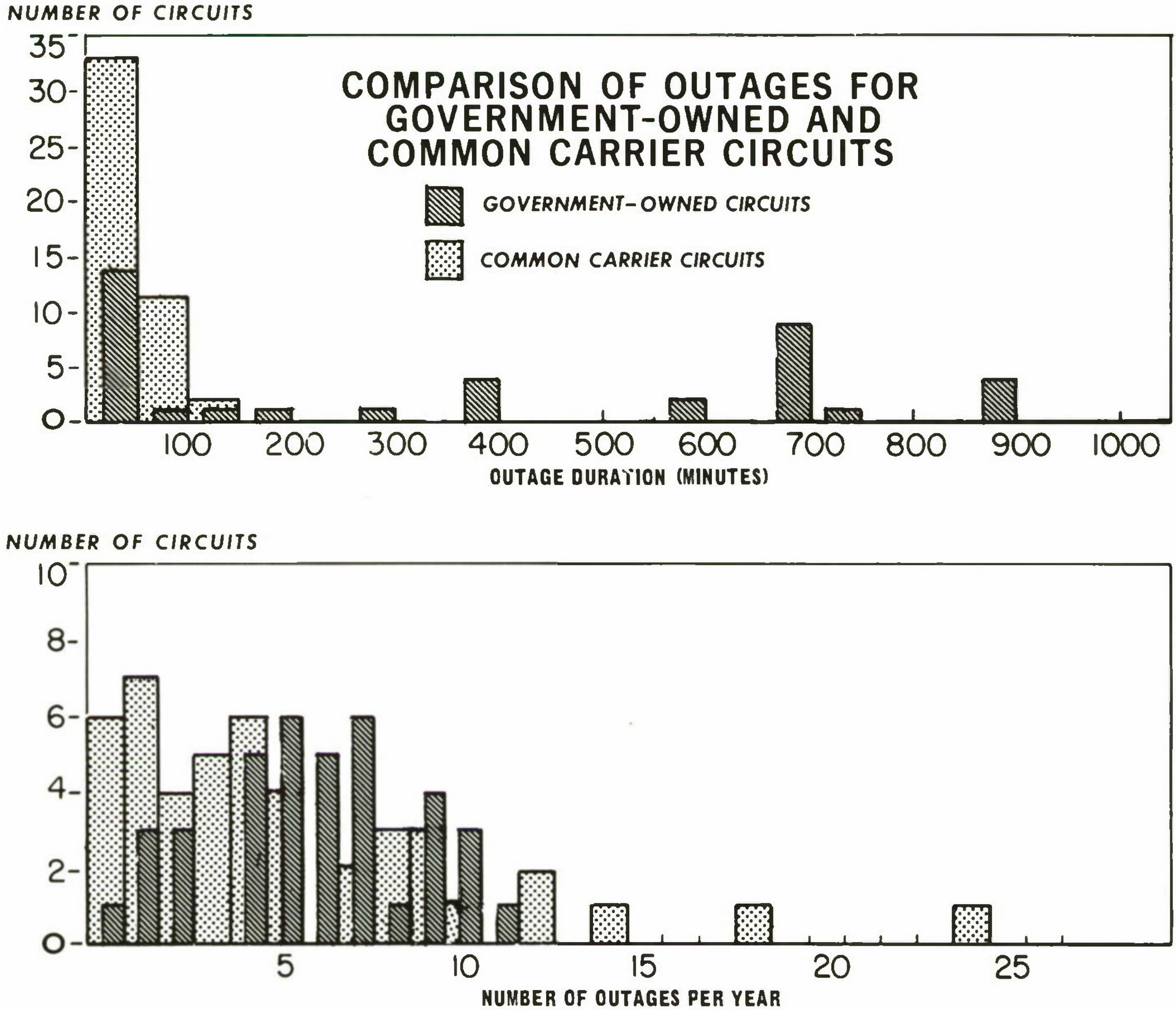


Figure 4—Comparison of Outrages for Government-owned and Common Carrier Circuits

used or whether high-quality, delay-equalized telecommunications links would be required.

The tests were made over a 350-mile circuit. The tone equipment was lined up according to the manufacturers' specifications. Tests were performed to determine squelch activation time and squelch recovery time for both impulse and white noise, the effect of noise on false trip operation of the circuit breaker as well as range of reliable trip operation.[4]

The operation of the system was compared using the conventional telephone channel and the high-quality, delay-equalized channel. The false trip levels were obtained by deactivating the squelch unit in the tone equipment on the receiving end and introducing impulse or flat noise at various points in the system. This noise was introduced at both the output of the transmitting tone equipment and the input of the receiving tone equipment.

It was found that the tone equipment was capable of operating with impulse noise levels in excess of signal levels before false operation occurred. Impulse noise could exceed white noise by several db before false trips occurred. A squelch setting level of at least 7 db below the normal value, for false operation from either white or impulse noise was used as a margin of safety to allow for squelch threshold drift.

With the system maintained at the normal levels a period of operation of four months resulted in only one false operation. This was caused by a frequency shift in the tone equipment resulting from partial power failure rather than from the effects of any noise in the system.

Our general conclusions from these tests were that the common carrier service and the Government-owned system would be equally effective in rejecting noise that might cause false operation of the circuit breaker and that conventional telephone circuits would be adequate for the degree of reliability required by the system.

Cost comparison

The final question posed by Circular A-76 was whether the telephone companies' service would cost more than a Government-owned telecommunications system.

The method of estimating annual costs for the Government-owned system as outlined in Circular A-76 is first to determine the composite life of the system, including buildings, access roads, antennas, transmission facilities, etc. and then to amortize the initial investment over the period of this composite life. In determining the amortization cost, the interest rate must be for the system must be discounted to its value at the equivalent to "the current rate for long-term Treasury obligations for capital items having a useful life of 15 years or more and upon the average rate of return on Treasury obligations for items having a useful life of less than 15 years." Circular A-76 also stipulates that allowances should be included for annual operations and maintenance costs and Federal taxes foregone. In addition it requires that "new starts ordinarily should not be approved unless cost of a Government activity will be at least 10 per cent less than costs of obtaining the product service from commercial sources."

The cost of the common carrier service on the other hand should be based on the existing tariff rates for the service provided, together with any additional costs not covered by the tariffs, such as engineering costs and Government administration of the service contract.[5]

The principal cost issues that arose in the Government-owned case concerned the determination of an appropriate value for the composite life of the system and the estimation of the annual cost of operations and maintenance for the system.

The arguments for Government ownership centered around the fact that a transistorized telecommunications system was specified that would have longer life and lower operations and maintenance costs than the systems previously used.

On the other hand, the telephone companies argued that their quotation must be based on their tariff rate structure which must necessarily reflect the cost of all of their facilities. These included some transistorized equipments but their system is still mainly non-transistorized. Therefore, their quotation did not reflect possible future reductions in tariffs resulting from these types of improvements.

The position of our office was that since both the Government-owned and common carrier systems would employ transistorized equipment the comparisons should be made either on the basis of present overall-experience on composite system life and operation and maintenance for both cases or that both should anticipate decreased costs in the future because of the shift to improve transistorized systems. On these bases the annual cost of the two systems was estimated by our office to be about equal in the two cases, excluding the 10 per cent margin factor stipulated by Circular A-76.

In an attempt to develop a better cost methodology for these types of decisions our office has contracted with the Planning Research Corporation. They are developing alternative means of making lease-versus-buy cost comparisons. One of the methods being studied is to discount the stream of cost differences throughout the life of alternative systems and then derive a "rate of return on investment" as a means of comparing the worth of the systems.

Other issues

The above discussion includes only a few of the more important issues that were considered in this case. There were, in addition, many dealing with legal factors as well as the question of the responsiveness of the various bids submitted.

For instance, this case illuminated a number of tariff problems, including the wide disparity between intrastate and interstate TELPAK service rates. Furthermore, it was found that differences in composite depreciation lives resulting from different mixes of equipment are not reflected in the current interstate and intrastate tariffs.

Another issue that arose during the analysis concerned the legality of leasing circuits in a Government-owned microwave system to privately-owned public utilities rather than requiring the utilities to lease these circuits from the common carriers.

At the present time some confusion seems to exist concerning the accounting of Federal taxes in these cost comparisons. In accordance with the guidance supplied in BoB Circular A-76, Federal taxes foregone should normally be added to the cost of a Government-owned system, if the comparable common carrier service would operate with a profit. An issue raised in the analysis was—if TELPAK offerings do not produce a profit for the telephone company should these Federal taxes be included in the cost of the Government-owned system? The telephone industry argued that the Federal tax applies only to the gross revenue of the company. On the other hand the electrical power utilities argued that the telephone industry is not making a profit on TELPAK and, therefore, that Federal taxes foregone should not be included in the cost of the Government-owned system.

Another important issue arose in regard to system design and engineering costs. Costs incurred by the Government for these types of services have been applied by the electrical power utility to the cost of the common carrier service. The telephone industry claims that system design and engineering services are normally performed by the telephone industry and that costs for these services are covered by the tariffs.

With respect to the responsiveness of the telephone industry bid for the intertie telecommunications, our study concluded from the evidence gathered, that the bid invitation was technically ambiguous in several ways. Many of the principal legal issues that were raised in this case have also arisen previously in similar Federal Government contracts with the telephone industry. They were resolved to the satisfaction of the Government and the telephone industry.

It would certainly be helpful, in the future, if specific policies could be established which would clarify the types of exceptions that make a proposal unresponsive to a telecommunications bid invitation and to outline in more detail the procedures that should be followed when a decision of non-responsiveness results from ambiguities in the invitation to bid.

It appears, further, that standards are needed for the design of bid invitations for common carrier services. It should be recognized that specifications for specialized telecommunications services, such as those designed to meet the needs for electrical power interties, may require careful negotiation between the Government and the telephone industry before the bid invitation is issued.

In looking at the long-range issues associated with power utility telecommunications there is one that appears to be of particular significance. In its bid invitation the electrical power utility specified that microwave circuits must be used throughout the system to reduce transmission time delays. However, in the future, spectrum crowding may force the privately-owned and Government-owned systems as well as the telephone industry to convert to telecommunications cable systems. There is an indication that the power control systems may in some long distance circuits, require faster reaction times than could be supplied by cable systems. However, as indicated previously the extension and intertying of the power grids will increase the critical time constants of the system. The tradeoffs of these factors should be the subject of continuing study as the network configuration and the spectrum-crowding situations change.

Because of the close interdependence of the telephone industry and the power industry, as has been noted above, consideration should be given to the formation of an analytical and policy making task force that can be asked to determine the most effective means of assigning national assets, such as men, materials and spectrum space to the nation's composite telecommunications system including those of both the telephone and the power utilities.

Our office plans to continue its study of these types of cases since we feel that the lessons learned will prove useful in formulating adequate policies for these types of telecommunications issues. In addition, an Intra-Governmental Study Group, with representatives from 12 Federal Departments and Agencies, has been established to assist in developing and coordinating further guidance for these types of decisions as they pertain to Federal telecommunications systems.

REFERENCES

1 CHARLES J. HITCH and ROLAND N. McKEAN *The economics of defense in the nuclear age.* Cambridge, Massachusetts Harvard University Press 1961

2 F. R. ELDRIDGE *Department of defense cost and effectiveness evaluations of command control and communications programs.* Proceedings of the 16th Military Operations Research Symposium Fall 1965

3 D. J. MARIHART and D. L. WYLIE *Southwest Oregon radio system for the BPA* AIEE Paper 56-7000 1956

4 R. P. BARTON *The effect on protective relaying security and reliability of abnormal conditions on communications channels.* Pacific Coast Electric Association 1966

5 GRANT and IRESON *Principles of engineering economy* The Roland Press Company, New York 1960

THE COMPUTER UTILITY AND ITS USER COMMUNITY

ROBERT M. FANO, *Chairman*

Session Participants:

VINCENT E. GIULIANO
EDWARD L. GLASER
MAX V. MATHEWS

Introductory Remarks

ROBERT M. FANO
Massachusetts Institute of Technology
Cambridge, Massachusetts

The future effectiveness of multiple-access computer systems as aids to the individual will greatly depend on how well we will be able to match them to their users, to their individual needs, and to their community goals. There are many aspects to this matching problem. The first and most elementary one is the instructional aspect. Since the main value of a computer system lies in the great wealth and variety of services that it can provide, learning how to exploit them will be a continuing process for all users. In particular a user should be able to learn, and receive help when in trouble, from a remote location, without depending on any next-door expert. The second aspect, somewhat more subtle, is the trust that each user can be expected to develop toward the computer system: trust that it will be available when needed, trust that it will not lose the products of his work, trust that it will protect his work and his private affairs against malicious or negligent actions on the part of other users, and trust that other users are not given economic advantages or other privileges at his expense. The ramifications of these issues go far into the design of computer systems and into their operating policies and procedures. The final and most crucial aspect concerns the reciprocal influence of the intellectual complexion of the community of users and the nature of the facilities offered by the computer system serving them. There are indications that the coupling between system and community may be so strong as to quickly magnify initial characteristics of either of them. For instance, if the system happens to be especially effective for some particular type of work, that type of work will prosper and even better facilities for it will become available. Conversely, activities for which the system is initially poorly suited will be discouraged and may eventually die out. This phenomenon of selective reinforcement may well change in a radical manner the intellectual profile and character of a community.

The importance of properly matching a computer system to its community of users cannot be overemphasized. As a matter of fact, we may well have to think in the future of the community of users as part of the system itself, or, perhaps more appropriately, of the system as the repository of the community's knowledge, and as the major channel for intellectual communication between its members.

A proposed experiment using a new system of documents for communication in the computer science field

by MAX V. MATHEWS
Bell Telephone Laboratories, Incorporated
Murray Hill, New Jersey

INTRODUCTION

This paper proposes an experiment in communication in the computer science field. The experiment consists in the development of a number of vehicles for interchanging information amongst a group of computer users and developers. Each vehicle is designed to satisfy a particular need and the new features of the experiment lie in the putting together of a number of vehicles rather than in the novelty of any particular vehicle. The choice of computer science as a technical area in which to experiment was made for several reasons. There is a clear and great need for better communication techniques in this field. A number of communication vehicles involve the use of computers and computer files so the technology to develop these vehicles is available in the computer user's group. However, although the vehicles described here will be specific to computer science, many of the ideas involved could be generalized to other technical areas.

The particular characteristics of computer science which will be used as a basis for proposing communication vehicles are:

- It is possible to delineate fairly sharply the limits of the subjects in the field.
- The field is changing very rapidly, so that speed in communications is essential.
- There is a real need to disseminate information amongst a group of very interested workers.

In order to make the vehicle seem somewhat more realistic, I will take as a specific environment the development of the second generation time shared computer with remote access. This work has produced a great deal of discussion, soul-searching, confusion, and general exchange of information and thoughts amongst the communities which are involved. Inherent in the process are:

- The development of the computer itself. Substantial changes have been made in the original computing machines as a result of deliberations.
- Writing the system programs which will make the computing machine available to a wide group of users, particularly by means of remote consoles using time sharing.
- Describing these programs and disseminating information about them to the user community.
- Incorporating user generated needs in the computing system.
- Training users.

All these communication requirements are engendered by the new machine and, furthermore, they are typical of any large transition in computer systems.

At present these communication requirements are being "satisfied" in a haphazard and unsatisfactory way. Decisions are made by various committees, but the total community is not kept apprised of the deliberations or made sufficiently aware of the crucial issues. Feedback is insufficient from the general community of users to be effective in directing the planning of the system. After the planning stage, it is necessary to disseminate information about various programs that are being written. Initially, information must go to people writing other parts of the system which interact with these programs; eventually, information must go to the general users of the system. At present, the adequate transmission of this information is too dependent on the writing abilities and the time taken by the individual programmers. Many difficulties and mistakes arise from this poor communication; these range from errors in the system programs themselves to less effective use of the system.

The communication problems in a computer community are made more difficult by the rapidity with which programs in computers change. However, they are typical of communication problems in many other areas of science and technology. And they will become more typical if the pace of technological change continues to increase and the number of people engaged in a particular technology increases.

In order to facilitate this communication I here pose, as examples, seven vehicles, each designed to handle one particular facet of the problem. The vehicles are a newspaper; a computer user's handbook; a set of instructional programs; a person who is a combination of reference; librarian, instructor, and adviser; a set of computer programs; definitive summaries of the state of the computation art, and periodic meeting of a computer user's society where oral papers are given. The rest of this paper will outline these vehicles in more detail.

Vehicle 1—newspaper

A technical newspaper is an essential medium for rapid dissemination information. Ordinary newspapers can serve as a model for this device. However, hopefully, this newspaper will be more accurate than ordinary papers since it will be directed to a smaller community. The paper will be published for news value. Material will be published very rapidly, and the paper itself will not be intended as a permanent record. Thus, every effort will be made neither to save nor to index nor to reference nor to cite material. It is not always possible to prevent things from being saved, particularly in a library environment. However, if the functions of indexing, referencing and citing are not done, then a few piles of newspaper or a few rolls of microfilm will not cause too much damage. Indeed, these may be of value as ultimate source material to future historians.

As an experimental comparison, the newspaper should be published in two ways. It may be presented on a daily or weekly computer file, and distributed over the network of computer consoles. It may be published in a paper form as newspapers are presently put out. A very interesting comparison of these two media for very rapid dissemination of information is thus possible.

Four types of material will typically be in the paper; news items, letters to the editor, articles, and advertisements. I will give some indicative examples of each:

News items—Typical headlines of news items might be as follows: Industry and schools slug it out over foreground versus background computing; computer manufacturer accedes to user demands for hardware page tuner—a new computer is born! Mrs. System-Input-Output pregnant again—how can we replace our irreplaceable programmers? Intrex conference debates—should we go time-shaped or have a separate computer facility for Intrex?

Letters to editor—Typical letters might concern the following subjects; A repartee on PL/1 versue Comit 2 for text-editing; A letter leading for adequate computational facilities for computer music on the new system.

Articles—A typical article by Art Telligence might start: "I just had a great idea. Let's write the new system in Lisp 3 because . . . "

Advertisements Many computer manufacturers would find a newspaper a good medium for advertising. These advertisements might serve a good purpose in keeping the user community up to date regarding equipment.

Vehicle 2—the system user's handbook

This handbook, which would be in the form of a computer file, would give detailed descriptions of the current public programs in the system and the current equipment available to the public. At present it is difficult to maintain up-to-date information about these programs, to disseminate this information and to eliminate previous information which has been disseminated. The handbook file will try to achieve several objectives:

- It will be kept absolutely current. When a "public program" is changed the description in the handbook must be changed at the same time. No introduction of new public programs will be allowed until the handbook changes are completed for concurrent introduction.
- No official printed version of the handbook will exist. Thus changing the one computer file will be enough to officially erase all past versions of a particular set of information. Of course, it is impossible to prevent some people from making printouts of the official file. But they do so at their own risk and, in general, such printed versions will be discouraged.
- The user's handbook will be written and edited for comprehensibility to the "experienced" system user. By necessity, a section describing a new program will be written by the programmer preparing the program. However, to achieve the required comprehensibility, at least one stage of reading and revision by an experienced and critical outside programmer or editor will be required. If necessary, several stages will be done.

The user's handbook will be indexed in depth using all available and useful indexing techniques. Thus it should be possible to locate information with the maximum facility allowed by the current technology. The indexing is most important.

Vehicle 3—instructional programs

The user's handbook is intended as a source of facts for experienced computer users. By contrast, the instructional programs are intended to educate a class of neophytes up to a state of being experienced users. Because of their nature, instructional programs will take longer to develop than the user's handbook. When a major new program is introduced, the handbook will appear first and the instructional program will lag behind a certain time. This time delay is unfortunate and will be minimized by computer editing techniques.

The instructional programs will probably not be classic programs of either the multiple-choice-answer type or the constructed-answer type. For the most part they will be a carefully tested text which will involve many exercises to be done by the student on the computer. These may or may not involve some intervention on the part of a human teacher. The instructional programs will be evaluated by actual use and will be revised as necessary in order to achieve a given level of instruction. The programs will be carefully graded as to the background necessary to take the program. Thus if high school algebra or MAD computer programming or MACROFAP is required as a prerequisite for a given program, this will be clearly state.

The instructional programs will be stored as computer files. A great and perhaps primary advantage so achieved is to allow easy and general editing of the files to keep them up to date. Obsolescence is one of the most difficult problems of instructional programs in rapidly changing areas. Computer technology may contribute a great deal to instructional program technology in this area.

Two modes of administering the instructional programs will be tried, on computer consoles and as printed paper documents. Thus an interesting and important comparison between console distribution and paper document distribution is inherent in this vehicle.

Vehicle 4—Miss Know-It-All

This vehicle will definitely not be on a computer file. It consists of a person who is a combined reference librarian, instructress, psychotherapist, and program counsellor. Such a position might appear hard to fill. I think it will be possible to find a number of women applicants who have the highest abilities but whose personal research has been interrupted through matrimony. These will still have sufficient time and interest to do a job of this nature. Such people will have a number of functions. They will be a source of information about documents and fact retrieval. Here we should be able to get a very good comparison between the annotated catalog and a person for direction on where to find a particular piece of information. It has often been stated that the right person is a much better guide than any catalog, but this assertion has never been really tested, and a comparison could be made in this environment. They will be a guide to students using the instructional programs. Here part of their function would simply be to see that the students have the right background and use the correct instructional programs for the education they need. Another part will be to encourage the students so that they do not extinguish in going through a given program. Still another is to provide some human instruction and to take care of the situations which the written programs do not accommodate. They will provide program counselling. The program counsellor can suggest ways in which various programs should be written, can suggest existing programs or subroutines for doing particular jobs and can provide some assistance or at least moral support in debugging programs.

In general, Miss Know-It-All will provide a human source of information which can be compared with the mechanization which we are imposing on our other vehicles.

Vehicle 5—computer programs

Little need be said about computer programs. They will be stored on computer files and will be the basic information in the system. Some will be almost humanly readable and some will be very difficult to read, depending on the language in which they are written. In many or even most cases it will be desirable to have adequate annotation so that a person can figure out what the program does. It is perhaps useful to divide the programs into three classes:

- **Public programs.** These programs will have very rigid standards of documentation, some of which have already been mentioned in connection with the user's handbook. The program itself, in addition to the handbook, will be required to have a very complete and a very high standard of annotation. Such annotation must be provided with the original program.
- **Semipublic programs.** These programs will be written by various individuals and made available to the public, somewhat at the public's own risk. They would correspond to SHARE programs. The same standards of documentation and annotation will be applied to these as are applied to the public programs. However, the enforcement of these standards will be more lax in that only psychological pressure will be applied to achieve conformity.
- **Private programs.** These are programs intended to be used by one or a very few people. Here, although annotation and documentation still seem desirable, it will be up to the user to do what he wishes. Different people vary in the degree they find most advantageous and in addition the

originator will be the primary person to suffer from any sins of omission.

Vehicle 6—definitive summaries of the state of the computation art

These summaries and reports will be formal published papers which are requested and paid for by a committee of experts which is formed for this express purpose. The committee will meet for one week every six months. (In the Summer the meeting will be held at the National Academy establishment at Woods Hole and in the Winter the meeting will be held alternately in Honolulu and Bermuda. Occasionally a meeting at Lake Como will be held to incorporate European opinions.) The committee will review progress over the last six months, will commission the writing of papers to describe this progress, and will review papers already written. The material already written may either be published, or may be sent back to the author for revision, or may be rejected and sent to another author for rewriting. The highest standards of importance, technical correctness, clarity, and tutorial informativeness will be maintained in these papers. Being asked to write such a paper will be considered a high honor, one which is weighted heavily by academic committees in charge of promotion. Authors will be paid well for their writing and encouraged to take a sabbatical and go to some nice isolated place in order to do an undisturbed and excellent job.

These definitive papers will be carefully referenced, cited, abstracted, and indexed. As the state of the information retrieval art progresses they may well be written in a form to facilitate the machine-retrieval of their facts. This may necessitate annotating the documents with special machine-readable code in order to facilitate "understanding" of the material by a computing machine. The documents will be of such value that the additional effort to make them machine-understandable will be well worthwhile. In any case, the documents will be in machine-readable form and will be available on computer files as well as in published version. They will form part of the world's permanent computer literature.

Vehicle 7—meetings of computer user society

A society of computer users and makers will meet for a few days every six months. These meetings will be patterned after the present meetings of the Acoustical Society of America. Oral papers with published abstracts will be presented. Every effort will be made to achieve the maximum of oral communication between the attendees at the meeting. No attempt at written publication of the entire papers will be made. The attendees will be free to publish the works which they feel are worthwhile in other standard computer publications. Effective oral communication will be facilitated by the newspaper (Vehicle 1) which will give all the participants a common background of information as well as by the various other vehicles to which they have common access. Participants will be encouraged to be highly critical of the papers, discussion after presentation will be encouraged, and in general all forms of oral communication will be pushed. The most damning criticism of a paper will be that it is presented in an unintelligible way; the second that it contains nothing of interest.

CONCLUSION

The seven vehicles described above are in themselves not new. However, putting them together into one integrated system to handle the communications in a particular area is a new experiment. The vehicles which have been described are designed to provide rapid written communication of new items, to provide definitive information about the current technology, to provide instruction in the current art, to provide a human source of reference for all information, to provide the actual technical vehicles (which in this case are computer programs), to provide definitive information in the form of a permanent record of the developments of the field and to provide a vehicle for oral communication in a reasonable environment.

The particular vehicles and the way in which they have been discussed is pertinent to computer science. However, most of the needs of computer science apply to other fields with more or less emphasis depending upon the particular field. Thus, though the vehicles described may differ a bit for another field, a similar set of vehicles would be involved. Certainly the areas of rapid news communication, of a handbook of facts, of instruction, of oral communication and of definitive papers are involved in almost every field.

With the exception of Miss Know-It-All and the oral meetings, all the vehicles are suitable for distribution either on computer files or as pieces of paper, and I have frequently pointed out possible comparisons of these media.

How can an improvement in our technical communications be initiated? Clearly this is a most difficult question, but two thoughts seem pertinent. First, it will be much easier to start with a new system and high standards in a new area than to improve an existing area which already has established habits. Thus, computer science seems a hopeful starting point. Second, the importance of quality and communication in our technical literature must be continually emphasized. The technical experts in a given field must be reminded that they are responsible for the quality of their literature. Nothing any library system, either regular or computerized, can do will help nearly as much as careful writing by authors. In the final analysis, purification of the literature will be determined by the attitudes of the members of the technology. They will create their heaven or hell.

MILITARY COMMAND INFORMATION SYSTEMS

GORDON T. GOULD, JR., *Major General, USAF, Chairman*

Session Participants:

DAVID R. ISRAEL
JOHN R. OTTINA
JOSEPH G. CARLEY, *Lt. Colonel, USAF*
ANTHONY D. RUGARI
CARMINE J. SALVO
JOHN A. EVANS

Introductory Remarks

GORDON T. GOULD, JR., *Major General, USAF*
Headquarters USAF
Washington, D. C.

A decade of pioneering work on Military Command Information Systems has equipped the major military echelons with their first generation of computer assisted command and control systems.

The current complex of individual first generation systems were largely designed and implemented independently and unilaterally as the requirement evolved. As a result, today, military command and control systems, on a world-wide basis, use more than 20 different kinds of computer hardware produced by more than four different manufacturers. The programs for these computers are written in more than 20 different programming languages. At least 10 of these programming languages are each used in only one system. Three of these languages are used in seven or more systems.

Certain factors therefore become apparent to managers concerned with military command and control – whether they be in the military or in industry. Cost effectiveness and operational effectiveness, on a world-wide basis, will be key watch words as we implement the next generation of military command and control systems. Compatibility, commonality, and exchangeability will be emphasized up to but not beyond the point where they degrade the user's military operational capability.

Looking at military command and control with a system point of view, certain judgments can be drawn with respect to exploiting the computer. Remarkable progress is being made in advancing the hardware state of the art. However, a lesser rate of advance is being made in other systems aspects. Also, these aspects are much more difficult to quantify than in the case of the hardware. Examples of the more difficult system aspects are: definition of the user's real needs; responsibilities of the system engineer; relationships between the user and the system engineer; retrieval and display of information to the decision maker; and, design and production of software.

The papers in this section discuss these aspects of exploiting data processing capabilities in military command and control.

System engineering experience with automated command and control systems

by David R. Israel
The MITRE Corporation
Bedford, Massachusetts

INTRODUCTION

This paper addresses itself to some of the key problems arising in the design, engineering, and implementation of automated military command and control systems in which real-time data processing plays a central role. More specifically, I have attempted to extract and summarize the experience obtained during roughly the first decade of the acquisition and use of Air Force command and control systems.

This "decade" may be roughly defined as having started in the early 1950's with the initiation of efforts on the SAGE (416L) air defense system for Continental U.S., and closed in 1966 with a turnover of the NORAD Combat Operations Center System (425L). In between, we have seen completion of the efforts on the SACCS (465L) system for SAC, the 412L air defense system for USAFE, and the BUIC (416M) back-up system for SAGE; we have also obtained significant experience from the on-going efforts on the Headquarters USAF Command System (473L) and USSTRICOM Command and Control System (492L).

While it may be somewhat premature to judge the results of what is essentially the first generation of Air Force command and control systems, it is clear that at least the control systems differ significantly from the more conventional application of computers to scientific problems or business data processing, and that both types of systems have represented a significant increase in technical complexity over previous engineering experience with electronic systems and equipments. It is not surprising, then, that successful system development and implementation—as measured in terms of expected cost, schedules, and performance—has, to date at least, been more the exception than the rule.

The material that follows is written from a specialized point of view; namely, that of a supplier of technical advice, assistance, and system engineering support to the Electronic Systems Division of the Air Force Systems Command, which has been primarily responsible for the acquistion of the systems mentioned. It is not necessarily the viewpoint of the Electronic Systems Division, nor that of the users or operators, nor that of the industrial contractors or suppliers of system components. I have tried to make it as technically objective as possible, but one must expect that there are bounds to a participant's point of view.

This paper, however, is written for a somewhat broader audience: those in the military and government as well as those outside the government who may be concerned with the acquisition of similar systems, for it is clear that many of the problems encountered are not unique to military applications, but are representative of any use of real-time information processing systems.

Accordingly, and with such a broader audience in mind, I have refrained from using much of the specific Air Force terminology and from referring to the pertinent Air Force procedures, regulations, and formats which have come into being over the past decade. Rather, the attempt has been made to provide a discussion independent of reference to specific Air Force jargon, procedures, regulations or usage.

In the sections which follow, frequent use is made of the terms *system design* and *system engineering*. I will use *system engineering* as the broader activity including such functions as design, technical support of procurement and production, and testing. A major point, if not the theme of this paper, is that system engineering must include a host of considerations which are not strictly technical if successful system development and operation is the goal. System design and engineering must attempt to reach an appropriate balance or compromise among the factors of operational requirements, technological capabilities, costs, and schedules.

The system engineering activity must take an extremely broad point of view. Its scope must cover the long time span stretching from initial concept to ultimate operation, and even beyond. Even during acquisition, these systems do not remain static, but evolve to meet new requirements. Systems engineering considerations, as well as encompassing a long time span, must also cover a diversity of topics. It is unwise, if not dangerous, to consider only the equipments and computer programs directly related to the operational functions. Proper system engineering will consider such varied topics as availability, training, and exercising of operators; operational procedures; provisions for the maintenance of hardware and computer programs; the availability and operating conditions of government-furnished equipments; communications integration with other systems; civil engineering construction of buildings and related facilities, and so on.

Beyond this, it is increasingly evident that successful system development is heavily dependent upon the organization and management techniques used to guide the system from concept to field operation.

It is in this broad context of topics stretching over a long time frame that I will discuss some of the design and engineering problems of command and control systems. Within the constraints noted above, instances from the past 10 years of experience will be cited.

In the next section, some of the principal characteristics of command and control systems will be noted. Section II presents comments on pertinent matters of organization and management of the system engineering and acquisition activity. Section III discusses some of the initial considerations which should guide the overall system design; Section IV is concerned with the process and techniques of the design effort. Detailed aspects of hardware, software, and testware design follow in succeeding sections.

As a final introductory remark, let me say that there are no known magic formulas, no known optimum procedures or techniques, nor any 10 easy lessons to successful system design and engineering. My principal intentions are only to point out items which should be considered and to illuminate possible pitfalls. Much of what is presented may seem like little more than ordinary common sense. If so, I can only wish that our foresight had been as good as our present hindsight. We have learned much about system engineering over the past decade; we have learned much less about how to document or apply this experience; and some things we have relearned. I am quite conscious of the difficulty of transferring such experience, and only hope that the material which follows can in some way be of benefit to the engineers and designers of future systems.

I. System characteristics

I do not believe it is possible—or necessarily useful—to establish a completely satisfactory definition of this class of systems. Instead, and as a means of establishing a basis for the subsequent discussion, let me summarize some of the principal characteristics of these military data processing systems:

- They are generally large, as measured in terms of equipment or geography, and usually very costly.
- They are typically made up of many elements or subsystems—computers, displays, communications, personnel, computer programs, data basis, input data sources, and output data users—all of which must be carefully integrated.
- They process large amounts of raw data from diverse sources, converting this to summarized information for men or other machines or systems.
- They must perform the data processing in "real-time" in order that the system response keeps pace with incoming data and required outputs, and they must be available on a continuous, around-the-clock, basis.
- While they normally operate well below their design capacity, a capacity which may never be required except in wartime, they must operate continuously at a high level of readiness.
- They usually replace an existing manual system and, in turn, must operate as a part of some larger military system.
- Despite the introduction of automatic data processing, human operators have a predominant role in the system operation.
- They have generally been designed and implemented by one organization (an acquisition agency) for use by another (the operator or user).
- Because of their size and importance, their development is usually marked by a large number of approval levels and coordination channels.
- They take a significant time to acquire, during which time both the requirements and the available technology may change.
- They must change and evolve during their operational lifetime to meet new requirements and situations.
- They usually defy an *a priori*, detailed, quantitative specification of their performance because of their complexity and the unpredictability of the conditions under which they operate.

It is similarly difficult to define with precision the differences between command systems and control systems. In terms of physical composition, the two types of systems have been quite similar, both involving medium- to large-scale data processors, operator consoles, and communication ties. From a pure hardware engineering standpoint, the problems involved in implementing them are, therefore, quite similar.

The basic differences seem to arise primarily because of the relationships of the system to the job that it is supporting. Control systems have tended to be very much in line with the job, with the individual job descriptions closely coupled to the manner in which the operators interface with the system. In the case of command systems, we currently find more of an off-line relationship between the system and the military job that it is supporting. Many command system user personnel do not accomplish their primary job by sitting at operator consoles throughout the day, but rather, turn occasionally to the system to obtain an output that provides an assist in the accomplishment of a primary task. In other words, the control systems have incorporated a much higher percentage of the tasks involved in accomplishing specified military jobs than have the command systems that have been provided to date. It is not clear that this must be the case, and greater effort needs to be made to increase the percentage of the military jobs directly supported by the command systems.

This loose relationship between the command systems and the military job that they are supporting has process. The functional requirements for control systems tend to be more definitive than those for comin turn had considerable impact on the acquisition mand systems. This fact then makes the translation from requirements to design more difficult for command systems. In this situation, the designer often has little to draw on from his own background and is dependent on the user to supply the detailed description of the job which the system is supporting. This sort of description is seldom available in a documented form, and the designer may be forced to "move in" with the user to gain sufficient knowledge of the job to bring the technology to bear on the problem.

The problems of simulating the system environment during the developmental stages are much more difficult in the case of command systems. No one has yet successfully simulated a command system environment. On the other hand, the environment for such control systems as SAGE and BUIC has been simulated quite realistically by simply tying the command centers into the surveillance and communication networks, or by running the system against recorded surveillance data and simulated interceptor actions. The test and evaluation of control systems, then, has been more straightforward, albeit still a difficult task.

The problems of phaseover to operational status of the two types of systems are quite different. In the case of control systems, the user seems to be more willing to adapt to the system, even to the extent of revising his organizational structure. This has not been true in the case of command systems.

II. Organization and management

In keeping with the stated broad perspective, I will next discuss some pertinent considerations relating to organization and responsibilities, operational inputs, and procurement. This discussion is not intended to be either definitive or complete in its treatment of these nontechnical matters. Rather, I have only attempted to stress those points which are highlighted by command and control experience.

Project organization

It is generally agreed today that if one is to acquire a large new system, a strong, centralized, knowledgeable project organization should be established by and within the government agency with the primary responsibility for system acquisition and development. This seems to be necessary regardless of the type of contractual responsibilities and arrangements made: prime system contractors, associate contractors, or equipment suppliers.

In the material that follows, it will be assumed that *project organization* is synonymous with *acquisition agency* and with *procurement agency*.

This project organization should have a broad charter, both in scope and in time. It must consider the relationship of such problems as construction and facilities, training, maintenance, and exercising to the overall design, since they are likely to exert a strong influence on the system and its eventual performance. It should be charged with coordination of all aspects of the design. It should review and approve all schedules, monitor progress, provide a central and common focus for all system decisions, and excerise the necessary resource management.

This project organization should have a single, enthusiastic, strong-minded leader. It should not be headed by a committee that does not have requisite authority or that can arrive at decisions only by finding the least common multiple. The project organization should have or be directly supported by technical competence equal to the job at hand. As discussed below, this may vary with the contractual/procurement arrangements, but in no case should the project organization find itself incapable of dealing on an adequate technical basis with the contractors.

User inputs and support are equally important, and this is also elaborated upon below.

The location of the project organization is an important consideration. Collocation with the user seems to be desirable especially in the case of command systems, but may not be practical since the project organization may in turn be part of some larger organization handing several system acquisitions. In such cases, the integrity of this broader organization may be important for administrative and other reasons, including common use of certain specialists in various areas. Location of a system test bed or overhead facility (see Section IV) may also be a factor. Where collocation with the user is possible, a joint organization including user personnel has been found to be a very effective way of guaranteeing the user inputs discussed below.

Because the new system may very likely use part or all of the facilities and communications of an existing (manual) system, problems have arisen in the control of the facility and communication contractors.(Even for new communications, the use of user operating and maintenance funds for rental of communications has removed much of the authority of the project organization.) In some cases, the project organization has had little or no direct authority over these contractors. This situation, coupled with tortuous coordination channels, has caused serious problems in obtaining adequate response to communication and facility requirements. For example, in the case of required modifications to the Cheyenne Mountain Complex Facility of the 425L system, requests had to pass from the project office to the user, then through several Air Force levels to the Corps of Engineers to the facility contractor. In the case of communications, requests were forwarded through the user to the telephone companies concerned, and, in some cases, on to subcontractors of the companies. This serial coordination is, of necessity, time consuming, even assuming completely cooperative attitudes on the part of all parties. Adequate arrangements concerning control and direction of the facility and communication contractors, then, should be attempted.

Finally, the project nature of the organization should be stressed. It should be organized for a specific purpose and given the necessary authority and responsibility. It is the antithesis of the stable technical organization formed along conventional lines. An attempt to split up the acquisition and engineering jobs to satisfy some existing functional organization should be avoided, since it is likely to jeopardize the entire endeavor from the outset.

Operational inputs

The arrangements for the project organization should ensure early, continuous, and active representation from the ultimate using and operating commands* This is a requirement in the design process for the following reasons: first, it will provide a direct channel of information concerning changing requirements; second, it will provide the necessary contact with field operating problems; and third, it will provide mutual appreciation of problems by the developer and user.

There is a strong and very dangerous tendency for an engineer to design the system for himself to operate, and generally to operate only for short periods of time and at high, or at least operationally interesting, load levels. The actual operators, on the other hand, are less sophisticated technically and usually must operate the system day in and day out for long periods at very low load levels—levels at which the automatic system generally is not required.

To put it in somewhat cruder terms, the engineer tends to design a sophisticated system which is difficult to operate or which only he can operate. On the other hand, the operator tends to design by adding a number of small fixes to isolated problems. This approach, in turn, results in less capability than might be achieved for the same investment.

The solution lies in close participation by the using command throughout the design and acquisition phases. The participators must be carefully chosen personnel, and their inputs must be controlled so that the requirement does not become merely a "wish book" which cannot be met within the constraints of the allocated resources. The participation should be continuous if waste is to be avoided. Confrontation between the operator and developer at only six-month or longer intervals leads to misunderstandings, unnecessary effort, rework, and often unsatisfactory compromises resulting from tradeoffs occurring too late in time. The operator should be required to concur on all operational design features, particularly those relating to operational personnel and man-machine relationships.

The representatives from the using command should include both planners and operators, with the balance gradually shifting to the latter as the project matures. The necessity to keep key personnel restricted to their day-to-day operational jobs may cause difficulty in achieving proper user representation. The expected rate of system development should take account of such difficulties.

Direct representation from the using command can have two bonus effects. First, it provides for a direct association of the operating command with the system development process, and thus gives the operating command an opportunity to be party to design and development decisions *as they are being made.* In this way, the user gains a helpful understanding of the problems associated with system development. And secondly, it

*In some situations, the using and operating commands may not be the same. This distinction will be ignored here, and the terms *user* and *operator* will be used interchangeably.

may help to alert and prepare the operating command for actually using the new system. This command is generally overcommitted to the operation of an existing system, and finds it difficult to prepare for operation of the new system. If the change is too drastic, the useful operational data of the system is delayed or the system may never be used if the operator is not "prepared."

A problem which has arisen with command systems is that no single or authoritative "user" or focal point in the command has been named, with the result that the whole gamut of the staff agencies must be dealt with. This, naturally, results in a proliferation of system requirements. The need, then, is to find the user; when there are several, someone must be named as the authoritative focal point, and preferably, this should be done by providing him with appropriate and special authority for this purpose.

System engineering capability

It is essential that a capability to accomplish the system engineering and technical services of the type described throughout this paper be provided to the project organization. Beyond its more conventional design and engineering functions, this system engineering capability must also attempt to safeguard:

The user's interest by—

- Ensuring that design specifications are responsive to operational needs.
- Ensuring that test criteria measure operational capability.
- Translating contractor design decisions into operational impact.
- Injecting new techniques, as feasible within the scope of the contract.

The contractor's interest by—

- Ensuring that specifications and user-requested changes are reasonable.
- Ensuring that there is a good technical understanding of the contractor's problems.

The procurement agency's interest by—

- Ensuring that specifications represent a best compromise between user desires and available funds.
- Ensuring that the efforts of several contractors are appropriately integrated.
- Advising on impact of contractor approaches and design decisions.
- Monitoring schedules from the point of view of contractor *actions* rather than reports.

Clearly this activity is not a one-time process which can be completed early in the acquisition period. The system engineering effort which is addressed here is that which is concerned with the entire definition and acquisition phases and proceeds from preparation of definitive performance or design specifications through procurement, installation and test activities to the operational phase of a program. Properly executed, it is a continuous process. Where adequate attention has not been given to such efforts, serious problems of delays, cost overruns, or unsatisfactory performance have resulted.

The system engineering efforts can be obtained from technical personnel on the staff of the project organization, can be contracted from organizations specializing in such services, or can be obtained from the industrial contractors participating in the program. These are not exclusive approaches; the selection or choice generally depends upon the existing situation, and combinations of the three possibilities have been used successfully.

Selection of the procurement method obviously affects the choice of the size and source of the system engineering support. However, regardless of the number or roles of industrial contractors in a system acquisition, a project organization generally has need of additional technical services. Many technical decisions must be made after contracting, regardless of the type of contract. When a major system acquisition is broken into pieces and the project organization acts as the "prime contractor," there is a clear need for broad and deep system engineering to make sure that the pieces can go together again.

Even at the extreme of a fixed-price contract, the acquisition agency must have sufficient technical capability to monitor the contractor's progress, understand the nature of the problems uncovered, and where necessary, assist in the resolution of technical problems. At the time a fixed-price contract is negotiated, the degree of definition of what is required, the specificity of how performance will be measured, and the confidence in achieving the desired capability within cost and schedule goals are not constant and are affected by a number of factors. The degree of definition at contract time is affected by one's ability to communicate requirements and forecast the system environment, the need for state-of-the-art development, the complexity of what one is attempting to acquire, the amount of prior experience with the type of system being acquired, and the process by which one arrived at the contract stage.

Equally important is the ability to define at contract time the manner in which the specified performance once acquired will be managed and demonstrated. Many of the same factors that limit the accuracy of the specification limit one's ability to accurately spell out test or demonstration measures.

The extent to which the factors I have enumerated are present in a fixed-price acquisition determines the level of effort required to provide a degree of confidence that the procurement goals (cost, schedule, and performance) will be met, and to allow the acquisition agency to discharge its responsibilities in a meaningful and timely manner.

In short, and under any circumstances, periodic reviews of the design and production activities are necessary if unfortunate surprises are to be avoided.

Procurement aspects

While it is not the purpose of this paper to explore all the procurement problems associated with military real-time data processing systems, certain of these problems and related considerations are of such direct interest to the system engineering activity that they will be touched on here.

Overall arrangements

In a system program, there is wide variety possible in the types and responsibilities of contractors, methods of selection of contractors, and types of contract. There has been a wide variety of good and bad experience with these different alternatives, and no single alternative is the optimum or best answer in all circumstances. Any can work, and any can fail. One piece of guidance, however, does seem clear: the acquisition agency cannot and must not relax its vigil or diminish its technical interest, regardless of the alternative selected. The requirement exists, under any method chosen, for a strong competent, and responsive technical and managerial capability.

A key decision, then, will be that of how to augment the project organization's "in-house" system engineering capability. The use of independent technical organizations furnishing such service has been noted, and their use is sometimes also termed "in-house."

There are two major approaches to acquisition which help limit (but not eliminate) the need for in-house system engineering support on a fixed-price contract. These approaches help to pin down the contract requirements by providing greater insight into the design method or by providing a basis for hard performance measures.

By use of the "contract definition" approach, involving a funded definition phase with two or more contractors prior to final selection, the majority of in-house engineering effort can be applied before contract rather that after contract. The results of the effort can be a set of design specifications rather than a set of performance specifications. Because the contractors are being paid for their design effort before acquisition begins, they are generally more responsive in providing design details. This is not always the case when one goes out for acquisition with performance specifications and tries to get design specifications as a product of contract negotiation effort. Design details are necessary in most systems (when performance is difficult to demonstrate) to confidently write the fixed-price contract. In this regard, I would like to point out that performance incentives may do more harm than good as a means of satisfying the acquirer's needs if one cannot adequately (clear and unambiguously) pre-specify the way in which performance will be tested and demonstrated.

A second approach to more adequate definition in some situations is the use of prototypes, or the "fly-before-buy" approach. If one constructs and tests a sufficiently comprehensive prototype of a system before acquisition, the risks in acquisition may be reduced.

Even though both of these approaches ordinarily represent significant investments in time and dollars, they do not guarantee that the need for engineering support besides that provided by the contractor will be eliminated. If the system under acquisition is pushing the state-of-the-art, if a complex approach is being used, if the performance is difficult to measure, or if the number of interfaces or amounts of government-furnished equipment are large, the need for extensive in-house engineering effort will still exist.

Under any contracting situation, experience points out the desirability of a separate, independent, test contractor, who is *not* responsible for the original design or any of its implementation, to help assure that the system is properly delivered, installed, checked out, and evaluated. This is especially true in the case of multisite systems.

Software

The software or computer programs generally represent the most costly development item in the system acquisition. As such, their procurement represents a major undertaking. Computer programs can be acquired in several different ways. For example, software can be procured from equipment manufacturers. Software manufacturers not involved in equipment production are also becoming increasingly available, but the use of a company different from that which supplies the system computer can introduce some difficulties relating to release of proprietary information.

Based on the operational job to be done, a proper match of hardware and software must be defined, and neither should be procured without first considering the implication on the other. In cases where the hardware and software represent separate procurements, the hardware is normally selected first.

Software is difficult to procure under fixed-price contracts because of the difficulty of producing detailed performance specifications for these programs. As opposed to hardware, software specifications are most often in the language of the user rather than the supplier. As with hardware, cost should not be the primary or only selection criterion for the software contractor. Experience, past performance, geographical location, and availability of trained manpower should weigh heavily in the ultimate selection.

For various reasons, on several occasions more than one contractor has been involved in the software preparation. As noted in Section VI, there are several major pieces to the software; this has led to a false conclusion that these pieces are completely independent and can be treated as such. In most cases, these split software contracts have led to unfortunate and unpleasant disagreements, regardless of whether there existed a subcontractor or associate contractor relationship. The answer seems to be to avoid split software arrangements if possible. If not:

- A single agency should be assigned overall design responsibility.
- Additional resources should be provided to monitor the separate software efforts.
- A detailed specification of the program interface should be prepared and used as a basis for ensuring program compatibility.
- Special care should be excercised in establishing contractual coverage for the separate software efforts to ensure adherence to interface specifications.
- Special provision should be made for the testing of the integrated software.
- Satisfactory working relationships between the contractors and the project organization should be provided.

As noted in Section VI, some of the necessary software is directly associated with the computer and comprises the essential general-purpose support programs with which to produce the operational and other system-specific programs. These programs are customarily supplied by the equipment manufacturer, but there is little or no standardization in what constitutes this basic support software. Hence it is not surprising that there have been a number of misunderstandings and subsequent disappointments concerning what the computer manufacturer was to deliver in this category. The situation may be worse if the programs are procured from some other source. Accordingly, it now seems best to consider the basic support software as a deliverable part of the computer hardware procurement. Note that scheduling of the delivery of the support software may be critical since it is required for use in the preparation of the system software.

Software procurement should be made on the basis of a plan which considers the longer term software problems. A major consideration is whether the military itself will take over software production or maintenance at some point in time. If so, early introduction of these people for training purposes is indicated.

The problem of correctly estimating the software effort is reviewed in Section VI. A related problem is that of measuring actual progress during the software design and production process itself. A count of the actual instructions written provides only a limited measure, in view of the initial uncertainty in the number required. Further, the actual preparation of the written set of instructions is only part of the job, with far greater uncertainties lying ahead in the various phases of checkout. The result, then, is that progress in the early phases of the software production generally seems to go according to schedule. At about mid-point, delays start to be projected and slippages begin to take place, generally on at least a one-to-one basis (i.e., a month goes by and the delivery date has slipped by at least a month). Finally, a significant slip is introduced and the end date is ultimately reached.

Inability to predict end dates accurately must then be expected (and hence planned for); the only reliable prediction will be one made just before end date is reached. The answer from the procurement point of view is to require in the contractual documents that every attempt be made to delineate significant milestones than can be measured, and to insist upon and provide close monitoring of the progress. This is, unfortunately, easier said than done, and much improvement of procedure and techniques is required here.

Hardware

In view of the significant efforts and experience in the development of digital computers and related hardware for the commercial markets, and the high degree of applicability of these equipments to command and control applications, there has naturally been great stress placed on the use of "off-the-shelf" items. Unfortunately, this term is not precise, and its interpretation has ranged from "available commercial equipment in production," to "equipment in development," to "prototypes," to "new computer designs to be implemented with available modules," and even to mere ideas (or hopes). There is great difficulty in trying to separate fact from fancy.

Because of the long lead time of the computer programs, an early delivery of the computer is generally essential to their timely checkout. Thus it is necessary that great caution be exercised in the procurement of the computer, to ensure its "off-the-shelf" availability. Accordingly, an early (less than, say, 120 days) availability of a computer with the same characteristics as the ultimate production models might be an adequate proof of its being "off-the-shelf."

Generally, each system has the need for some rather unique or special terminal or display equipments, so some amount of development is usually required. The experience here has been very bad, both with delivery and with reliability. Needless to say, then, hardware development should be undertaken only when absolutely necessary. This is a real problem for displays, where the user may want complex multicolor devices that are not truly necessary for successful system operation. This advice is easier to say than to follow, for there is always pressure for gadgets and items that are "better than anybody else's." If the user persists, the acquisition agency must make the projected cost in time and money clear.

Insofar as is possible, then, scheduling should honestly reflect the differences between development items and truly "off-the-shelf" equipment. Where possible, the development cycle should be disassociated from the mainstream system acquisition. Plans should attempt to use available devices until the development items are ready.

It would seem there should be monetary rewards and penalties to provide additional incentives for better contractor schedule performance. Incentive features are certainly no panacea, but they might be effective and provide more usable legal leverage than the threat of termination for default.

Finally, caution should be exercised with hardware procurements based on detailed design specifications unless the procuring agency has assured itself that a product which meets these specifications will provide the desired performance. This requires significant in-house technical effort.

III. Broad or conceptual design

This section discusses some of the initial considerations which should guide the overall design. Detailed aspects of hardware, software, and testware design follow in succeeding sections.

What is the system?

Automated command and control has become extremely fashionable in recent years, achieving some appeal as a military status symbol. This, among other reasons, can lead to the initiation of a system development without a full understanding of what the system is and what it is expected to do. When this happens, the designer, the user, and the taxpayer may later regret the initial precipitate action.

Too often we hear the designer lament the fact that the user cannot supply him with an adequate statement of requirements. When this is said, it usually means that the designer doesn't understand the nature of these systems or the military problem. In practice, the user's requirements are *not* easily expressed in quantitative terms, and they can only be put into meaningful form for procurement when matched with the available technology and possible designs. In many cases, the available technology identifies or even generates the requirements.

Thus, as the very first step in the design effort, the designer and user must come to an understanding of the military problem, and they should prepare and document a matched statement of requirements and the corresponding conceptual or broad design. In doing so, the full implications of the system and its operational use should be explored. Both parties should consider the automated portion of the system as an iceberg, and a significant initial effort should be directed towards understanding and describing what is under the surface.

For example, the implications for manning, training, maintenance, and logistics must be fully understood by the user. Too often we are disappointed by initial claims of savings of operational personnel. I know of no system in which actual personnel savings have resulted, although in most cases the capability (or productivity) of the operations personnel has been greatly increased.

A related question requiring an early answer is: "What constitutes the system?" In particular, what equipments already in use by the manual system are to be employed? In determining this, the operating condition of these field equipments should be realistically ascertained. The early development of SAGE was hampered by the fact that the radars were not considered as a part of the system. In both SAGE and 412L, the actual operating condition of such field equipments was not properly understood, and difficulties resulted.

A careful inventory and field survey is recommended. This applies to both equipments and facilities. The system designer, often an electronics engineer, is quite prone to ignore the seemingly prosaic problems or roads, buildings, power, and air conditioning. Measurements on field equipments under field maintenance are advisable. A radar under field conditions performs quite differently than at the manufacturer's plant. SAGE and 412L experience attest to this.

At this early phase, the interfaces with other systems—present and future—must be considered. An exchange of data with other systems is almost always required, and the problems of data compatibility—in terms of information content, format, and signal levels—must be solved. Not only should these problems be considered at the design level, but responsibility for any changes to achieve compatibility must be clearly assigned.

Briefly then, at the outset there should be a careful consideration and documentation of what the system is, what it will do, what comprises it, and how it relates to other systems.

Evolution

As noted earlier, significant periods of time are involved in the acquisition of these systems. When they do become available, they may not meet the then current requirements and may be very difficult—in terms of time or effort—to modify so as to achieve the desired goals. This has given rise in recent years to a general feeling that the design and implementation of these systems should be pursued on a more evolutionary basis.

The merits of an evolutionary approach are said to be that we can proceed in small, sure steps. It would provide for an early demonstration of capability and would ensure operational experience and feedback before proceeding with a more sophisticated design. Headquarters command systems may be strongly influenced by the personality and desires of the commander himself, and an evolutionary capability would make it possible for individual commanders to tailor or modify the system accordingly. It is further felt that the sophistication and technical difficulties which have characterized the first generation of these systems could be avoided in the future if there were greater use of "off-the-shelf" equipment and a more direct initial mechanization of existing manual operations.

The current emphasis on evolution, however, may confuse what are perhaps two separate ideas. The first is the concept of a time-phased implementation, or what might be termed a planned evolution towards a predetermined design. The second relates to provisions for the unplanned modifications necessitated as a result of operating experience or changes in requirements or the technology.

In this light, the evolutionary approach has some dangers which should not go unrecognized. In particular, a time-phased implementation should not be used as an excuse for avoiding the total system design problem. If only the first increment is planned, then there is a strong possibility that inadequate attention will be given to the design requirements imposed by subsequent steps. Specifically, the universal lack of funds and the influence of ever-present economy drives cause strong pressure in this direction and may force a decision to buy only the equipment or capability required for the first steps and not allow time for sufficient analysis and design of the total, long-range system. Restrictions on building size, power, or air conditioning can be as serious as limited computer capability. Subsequent steps are then very difficult and costly to implement, possibly requiring grossly different equipments and costly retrofits to existing equipments.

Another danger of time-phased implementation is that if the subsequent steps are not very carefully planned, major problems can arise as these steps—involving additional programs and possibly equipments—introduce interference with the operation of the system. A related difficulty is the retraining of operators. In fact, there appears to be a critical mechanization level which should be incorporated in an initial step. This should include the full mechanization of the essential inputs and outputs, and should include the majority of the operator facilities (consoles, keyboards, switches, etc.), even though all of this capability might not be used initially.

It should further be noted that requirements for testing are such that initial system testing is not in any large percentage salvageable for subsequent steps, and a complete series of tests with all attendant costs may be required for the replacement or next model of the system.

Another point is that while direct mechanization of existing procedures may yield an early demonstration of progress, it should be applied with caution, as it may prevent the realization of a vastly improved operation which could be achieved by fully exploiting the automatic data processing capability. (Neither the automobile nor the airplane were products of direct mechanization.)

There is also the question of the use of "off-the-shelf" equipments. This, too, must be approached with caution and reason. Beyond the computers, there is very little that is actually "on-the-shelf," particularly in the sense that it can be applied directly to a system. As noted earlier, it is extremely important to differentiate between "off-the-shelf" and "in the brochure."

Design guidance in this broad area, then, would seem to be as follows:

- The design should contain a reasonable degree of excess capacity and flexibility in all subsystems and elements to permit long-term unplanned evolution. This excess capacity must then be carefully controlled so that it does not dwindle away before it is needed. It should not be used for "frills" which prevent its future use for "needs."
- If the system is to be implemented in steps, the design must consider the resulting system first, allowing sufficient capacity to achieve this goal.

Only when this end design is well understood should the steps in the implementation be delineated.

- The first step should not be too big to prevent an early demonstration of progress, but should be sufficiently large to include those features strongly affecting operator actions and activities as well as those features relating to integration with other systems.
- The design of a step-wise implementation must not ignore the problems of disruption and degradation of current operations.
- The designer should consider with caution the early benefits of a direct mechanization and the applicability and availability of "off-the-shelf" equipments.

Degree of automaticity

The proper degree of automation to be provided in most command and control systems is not easily determined. Many system functions are routine, easily described, repetitive data processing tasks which lend themselves directly to mechanization; other functions clearly require sophisticated choices or decisions which can or should involve human judgment. Unfortunately, however, a larger number of important functions usually cannot be so easily categorized. Here the cost and complexities of automation must be balanced against the problems and difficulties attendant of human operators.

First, the man-machine relationship is severely limited by the cost and complexity of the devices—consoles or large-screen displays—available for the information exchange. The operator can do a satisfactory job only if he is given the necessary data upon which to make his decision. This is surprisingly difficult and costly to accomplish, both in terms of hardware and software.

Second, the involvement of an operator introduces both equipment and human reaction times. Delays can arise both in the computer program and in the display devices. Consoles may introduce several seconds of delay, depending on the storage medium which drives them; large screen displays may involve processing and mechanical transport delays of ten seconds or greater; keyboards provide a flexible input capability, but require significant times for message composition.

Third, the operator may not have the information required to make a better decision than the computer. In SAGE, it was felt that whenever the automatic tracking program got into difficulty as a result of too little, too much, or ambiguous data, the operator would be alerted to correct the situation. An evaluation of one automatic track monitoring scheme later showed that the operators did not, on the average, improve the situation over what the computer would have done if it had been left alone.

The other side of tradeoff is that the automation of complex decisions can be extremely difficult and costly in computer capacity. A satisfactory automatic weapons assignment program must consider a multitude of factors: position, heading, and altitude of bomber aircraft; position and importance of possible targets; locations, status, types, and capabilities of available weapons; and so forth. Other decisions are not as simple to automate as they first appear. Early in the system design there is a disturbing tendency to oversimplify and underestimate, and we are subsequently embarrassed by the computer time or storage consumed by what had been felt to be a simple process.

Let me mention two examples from SAGE. The first is automatic initiation of tracks from radar replies reported by a scanning radar. This was first said to be a simple pattern recognition process: look for radar replies in close proximity on successive scans, allowing some leeway for tracks with low blip-scan ratios. In actual practice, a high level of noise replies requires a very sophisticated process if the wholesale generation of false tracks is to be avoided or if the initiation of real tracks is not to be overly delayed.

Track-while-scan of aircraft is still dismissed by many people as a simple process. Even if they include such considerations as low blip-scan ratios, false tracks, and turning tracks, they generally have ignored the problems of proximate or crossing tracks. This involves another level of design sophistication.

In both cases, it wasn't until a rather detailed knowledge of the actual physical situation became available and extensive computer programming had been performed that a full realization of the cost and complexity was reached.

Finally, decisions as to the degree of automation must be carefully made in light of operator limitations and training problems. The danger of the engineer designing the system for himself to operate is ever present. The performance of many of our automated systems—SAGE and 412L are examples—is overly sensitive to operator qualifications and training. This, in turn, places a very high premium on provisions for continuous exercising and evaluation of these operators. Such exercising and evaluation, as is discussed next, is not without its own costs and problems. An even better solution, where possible, is to provide the system with a meaningful day-to-day job. The integration of air defense and air traffic control has always seemed to be a natural combination. It is a pity that political and other constraints seem to prevent this union.

Self-exercising capability

As previously noted, during peacetime these systems normally operate at rather low load levels, far below design capacity. Since they are quite sensitive to operator performance, the best possible system performance will be attained when required only if a self-exercising capability to maintain and improve operator proficiency has been included as an *integral* part of the overall design. Such a capability, by which the system can be subjected to simulated high loads and special input conditions, will also permit on-going evaluation, checks on system performance, and those demonstrations of system performance which are often required for visitors.

As a part of the system design, then, careful attention should be given to the questions of which environmental conditions and sources of input data should be simulated and how this can best be done. Additional equipments are generally required for this purpose.

If the system has many sources of data, the generation of consistent inputs relating to the same environmental situation may in itself constitute a difficult task requiring data processing equipment. For example, in an air defense system it is desirable to simulate enemy bomber flights as seen by a radar. It is possible to do this in real time by suitable analog or digital simulators at a radar site, or the simulated radar returns can be prepared and prerecorded on a magnetic tape or photographic film which can be scanned by appropriate equipment at the site. Either of these techniques would be suitable for simulation at an individual site. However, if the system consists of many radars with overlapping coverage, and the aircraft will traverse the coverage of several sites, it becomes necessary to simulate a consistent set of radar inputs. This generally requires the coordinated generation of tape or film inputs. This, in turn, entails a significant data processing task in the preparation of aircraft flight paths and the computation of the r, θ data as observed by the radars. A general-purpose computer may be useful or necessary for this purpose.

In any event, is is generally advantageous for training purposes to arrange the simulation so that it is repeatable and situations can be easily reproduced for the same or different sets of operators.

A second problem of exercising is that of concurrent operation with live inputs. Even if it is electronically possible to mix the live and simulated inputs, this may cause extreme confusion and danger unless the computer has some means of segregating and identifying each. This may require excess computer capacity for the extra processing and display programs required to differentiate between the simulated and live inputs, at least to selected operating personnel.

Some simulated inputs cannot be preplanned since they are affected by the system operation and may require real-time data processing and human decisions. During a SAGE training exercise in which a center is removed from the air defense net, voice radio vectoring instructions are monitored by simulated pilots who insert the directed values of heading, speed, and altitude into the computer by special switches. A special computer program uses these inputs to simulate the flight paths of the interceptors. The current positions of these aircraft are then determined, permitting the computation of the simulated radar returns. Thus, system exercising involving non-preplanned, dynamic inputs may require special input facilities, added operating personnel, and additional data processing capability.

Finally, adequate exercising facilities must also provide the means for evaluating the system performance. Rapid feedback to operational personnel is a requirement if they are to understand and correct errors. Depending on the situation, real-time evaluation or post-test analysis of recorded data can be employed. This evaluation and analysis is not without cost, facilities, and design effort.

It should be noted that the facilities required for system exercising are closely related to those required for program and system checkout, shakedown, and test. The facilities should be coordinated and, where possible, combined.

Performance monitoring

As systems become larger and more complex and particularly as the number of input and output subsystems grow, it becomes increasingly difficult to determine whether the overall system is operating properly and to isolate the causes of difficulty. Accordingly, suitable provisions for performance monitoring, trouble detection, and quality control should be included as part of the system design. The requirement for continuous system operation, coupled with the relatively abundant opportunities for subsystem malfunction, means that the designer should not expect that performance checks during preventive maintenance periods will suffice. He must consider the need for dynamic or real-time performance monitoring and diagnosis of malfunctions.

The central computer and the computer program are not major problems in this regard. The majority of their random or intermittent malfunctions can be detected by parity checking, and in many cases an immediate repeat of the computer or program operation can be successfully conducted. Other errors generally cause an obvious system malfunction or complete stoppage. However, the input and output subsystems offer many opportunities for miscalibration, random errors, poor performance, and complete outage in a fashion that does not cause the system to cease operation. Individual communication links, particularly those carrying digital data, can become completely inoperative; individual radars can go out of calibration or can fail to report targets entirely; or individual consoles or manual input devices can fail entirely—all for significant periods of time before the degradation or failure becomes apparent to operating personnel.

Most subsystems have some "built-in" performance monitoring: parity alarms, power lights, etc. However, in many instances the subsystem equipments, their failure indicators, and those who maintain these equipments are remotely located, far from the eventual users of the data which they provide. Experience indicates that attention to the subsystem performance may only occur when the human element comes into play—specifically, when the user complains. Hence an objective of the system performance monitoring should be to provide the user with the tools and information to complain accurately.

The designer should consider the automatic and periodic generation of test messages that can be routed through these subsystems in a systematic manner for checking purposes. When difficulties are encountered, more specific and detailed check messages or techniques can be called upon to isolate the particular source of the difficulty. In many cases, it may be possible to correct or bypass the source of the difficulty temporarily until it can be fixed. Finally, a continuous summary of the status of the key elements of the system should be made available to both operating and maintenance personnel.

Two examples from SAGE might help to illustrate the general ideas. In the first, the radar data processing devices at each radar site were designed with a provision for periodically introducing false targets at predetermined range and azimuth positions, and the central computer isolates and processes these messages to check a number of the processing and communication facilities between the radar and the central computer.

The second example relates to the importance in the SAGE design, as with other netted radar systems, of accurate alignment of the radars in both range and azimuth. If they are misaligned, the generation of multiple data trails and false tracks for a single aircraft may result. Since both the radar and radar data processing equipments can become misaligned during maintenance on the antenna, decoders, etc., a performance monitoring feature was added to the SAGE computer program. With the feature, the computer checks reports from different radars on the same aircraft and determines whether a better match of incoming data would exist if bias errors are assumed in the azimuth data. If it is found that an azimuth correction on a radar improves the consistency of multiple radar reports, this error value is then introduced before processing subsequent reports. The error is also printed out for corrective action at the radar site at the next maintenance period. (Subsequent experiences indicate that automatic correction may be desirable only in critical situations; this feature has the possible negative effect of minimizing the pressure on correcting the source of the error.)

Performance monitoring, then, should be recognized as a system function on a par with the more familiar operational functions. It should not be allowed to slip into the background during the system design. Utilization should be made of the performance monitoring built into existing equipment subsystems, but added hardware and software may be required and this should be reflected into the design of system equipments, computer programs, and communication links.

Continuity and modes of operation

It will be desirable to subdivide this topic into two parts. The first part deals with continuity of automatic system operation under normal or peacetime conditions. The second part relates to alternate modes of operation, generally at lower capacity and capabilities, when key elements of the system have been put out of operation. For ease of reference, normal operation will be referred to as Mode I, an alternate automatic arrangement as Mode II, and a completely manual back-up as Mode III.

Failures of critical system elements will cause an interruption of Mode I operation unless adequate design measures have been taken. The usual solution is the provision of duplicate or duplex equipments coupled with error detection facilities and a rapid switchover capability. In the case of one-of-a-kind equipments, full duplicates would be required; in the case of a multiplicity of units—memory units, display consoles, tape drives, etc.—only a few spare elements might be needed. In the case of consoles, a general-purpose console design rather than special-purpose consoles designed specifically for an operational position should be considered. With suitable program parameter and switch label changes, it is possible to adapt a general-purpose console to individual positions, thereby retaining desired flexibility and permitting rapid replacement or substitutions in event of console failures.

With improvements in equipment reliability, provision of duplicate units, and suitable design, *unscheduled* interruptions of Mode I continuity can be brought down to whatever low level is desired. The designer must not overlook, however, requirements leading to *scheduled* interruptions of Mode I continuity. These include the functions of maintenance, equipment retrofit, program retrofit, and possibly system exercising.

Maintenance is self-explanatory. Equipment and program retrofit may result from initial design shortcomings or from the evolving nature of the system. Program retrofit must include a thorough checkout on the actual machine, and is not as simple as merely changing a tape unit and reading in a new program. As noted earlier, system exercising may require interruption of normal operation unless the live and simulated inputs are properly handled.

It is important not to underestimate the amount of downtime required to perform these functions. This is particularly true in the early months or years of operation where several hours may be required each day. On a long-term basis, daily maintenance or exercising may still be required, with only occasional changes to the hardware or software.

A spare or duplicate unit for each type of element in the system may permit continuity of Mode I operation during limited types of equipment maintenance and retrofit. A full duplex would permit a Mode I operation while performing any of the functions. Reverting to Mode II or Mode III operation is also possible, although the latter is not very desirable.

Duplexing will not, of course, prevent physical destruction of the key elements of the system and the consequent interruption of Mode I operation. Beyond the measures of hardened construction or mobility, some added survivability can be achieved by a dispersed or decentralized design. In some systems, a decentralized design utilizing several data processing centers may be required by economic or other considerations. For example, a single air defense center might be technically feasible for all of the United States, but the communications costs from outlying radar sites would be quite expensive. A completely decentralized design, with a computer center at each radar site, may be equally expensive due to the cost of communications among the centers, but this design is generally more survivable.

When several centers are involved, however, it becomes possible to arrange for a Mode II operation in which one or more centers can "cover" for an adjacent one. Excess data processing capacity and the necessary communication links must be provided.

The last and least attractive possibility in the event of Mode I failure is to revert to a completely manual, Mode III, operation. Needless to say, the capacity and capability are not attractive, and there is the serious economic problem of maintaining and exercising the added operational personnel. It is easily seen that Modes I, II, and III cannot be considered independently. The proper selection and design of these modes has a direct bearing on the system configuration and on almost every aspect of the design.

Overhead facilities

At the early stages of system design and planning, attention should be given to the following non-operational functions:

- Training of operational and maintenance personnel.
- Initial and on-going program production.
- On-going test and evaluation of evolving system changes and additions.
- Generation of exercise materials.
- Reduction and analysis of data recorded during system operation.

These *system* support functions—as opposed to those support functions which must be conducted at each *site*—generally require separate "overhead" facilities, owing to the large computer time requirements and special conditions involved.

The first three functions—personnel training, program production, and test and evaluation—require system-like equipments in system-like configurations, although perhaps not with the added equipments (added modules or full duplexing) required for very high reliability and continuity of operation. The last two functions—generation of exercise materials and data reduction and analysis—can use system-like equipments, but could also use other computers or commercial computing facilities.

Depending upon the specific nature of the system, one overhead facility might be sufficient to handle the first three functions. In large systems this may not be possible because of the requirements on computer time. In SAGE—with about 22 operational direction centers at peak—the number of such overhead facilities has decreased from a high of five, and until 1964 at least one computer was devoted to each of these functions.

An attractive possibility is to use the overhead machine(s) as a part of the back-up or Mode II configuration. In this case and when an operational machine is unavailable, an overhead machine could cease its non-operational functions and join the operational net. This might be considered as a form of duplexing, with the two machines at separate locations.

Overhead facilities, particularly those required for training and program production, assume added importance since they are generally required prior to the operational facilities. An overhead facility for subsequent test and evaluation can be used early in the life of the system for the design verification described in the next section.

IV. Design process and techniques

Volumes have been written on the subject of the different tools and techniques—ranging from probability theory to simulation—which can be employed by the system designer or engineer. How these can best be used and how the design effort should proceed are much more difficult to reduce to writing. Beyond stating that design is both analysis and synthesis, that it must involve feedback, and that it is hardly ever finished, I intend here to mention only two key techniques—design documentation and design verification—which have been found to be of practical utility in many systems.

Design documentation

The need for an early agreement on the requirements and a matching conceptual design has been noted. The design effort starts here and must bring into play full consideration of costs and schedules. Much interplay, much give and take, and much analysis and synthesis may be required. The product should be an operational plan, an employment plan, or some other suitably named document which will serve as the broad plan or prospectus for the system.

At an early point in the design effort, careful thought should be given to the types and levels of documentation to be used. Documentation is one of the key management tools for the design effort, with regard to both the timeliness and the quality of its execution, as well as its conformance to user and other requirements. Documentation, then, is a design technique; it is a key to organizing a design effort and to maintaining design control. The time and effort devoted to generating and maintaining the documentation will be very well spent.

The names of the documents are not important. To quote merely a few: Operational Specifications, Mathematical Specifications, System and Subsystem Performance Specifications, Functional Specifications, System and Subsystem Design Specifications, Computer Program Specifications, and Equipment Specifications. What is important is that there be recognized levels of documentation and that the responsibilities for each be assigned. Too often, the design is not properly documented and hence not available to those who must be brought to bear on the production and implementation effort when the system has been broken down into the smaller pieces required for such activities.

As noted, at the highest level there should be an overall description of what the system will do and how it will be deployed. Next, the overall performance of the system—at least in qualitative terms—should be described, including identification and definition of the principal subsystems. Such system performance specifications might be the vehicle for contracting with a they should be held to meeting subsystem performance prime contractor. If associate contractors are used, specifications. In most cases, the detailed design specification describing what is built will be prepared by the contractors.

In this regard, it should be noted that it is exceedingly important to document a baseline system configuration at the earliest date. This configuration has a first-order impact on facilities, construction, civil engineering, and the government-furnished equipments. The documentation of this configuration, ultimately describing all facilities, equipments, computer programs, and technical manuals in the system, must be adequately maintained and all changes controlled.

Design verification

The second design technique is design verification. By this is meant any reasonable steps which can be taken to validate the adequacy of the design at the earliest possible time. The objective is to avoid costly changes during production or even more costly field retrofits. The latter possibility might involve installation teams waiting unproductively at sites while design problems are diagnosed and solutions generated.

Design verification should start early in the conceptual phase of the design and should be continued, as necessary, until field implementation starts. During this period, the key or critical aspects of the design should be verified, checked, and tested to remove as much uncertainty as possible in the final design. In an air defense system, for example, the tracking logic should be tested under a wide range of realistic conditions; solutions to interception equations should be checked; man-machine relationships and capabilities should be verified, and so on.

Many different methods can be employed for design verification, ranging from complete simulation to live tests with actual equipments. In the former, the computational process, the operators, the environment, and even the system equipments can be simulated; the system computer or some other machine can be used as the vehicle for conducting the simulation; and the entire process can be carried out in a convenient time scale. Live tests, on the other hand, can involve breadboard, prototype, or early production equipments.

Simulation permits an investigation of performance over a wide range of conditions, but generally suffers from the lack of realism. We simulate what we know, not the unexpected, and generally our design has accounted for what we know. Live tests are more difficult and time consuming, and in some instances are too expensive or even impossible to conduct, as for example when the test involves many weapons, ECM, etc. In many cases the methods can be mixed or used to supplement one another, as for example, the use of selective live tests to calibrate a set of simulations.

The methods employed will vary from system to system. The significant point is that design verification should be considered as an essential part of the design process. It should be incorporated into the planning and funding for the system. It should be applied to the critical areas of the design as early and as realistically as possible. It is one of the few bits of insurance that a system engineer can buy.

V. Hardware

Computers

Significant improvements have been made in the speed, capacity, and reliability of digital computers over the last ten years. Today, many machines of proven performance are available, and they exhibit strong similarities in basic design features: order codes, indexing systems, interrupt systems, and ability to handle auxiliary memory devices (drums, tapes, discs). The chief differences are in word lengths, memory sizes, and operating speeds.

For the most part, computers represent one of the few "off-the-shelf" items that the system designer has at his disposal. There are some related hardware design problems, however, primarily in the area of the special-purpose in-out buffers and devices peculiar to the system. As regards the central computer, it is more a question of proper selection of machine and configuration rather than design.

The critical consideration in the selection of a machine is that of adequate capability: storage capacity, in-out capacity, and operating speed. It has already been mentioned that care must be taken to allow excess capacity both for contingencies in the original design as well as for unseen requirements. A machine that looks just big enough at the early stages of design is surely not going to be adequate at the end. A safety factor of two would not seem unreasonable, particularly as the cost of more storage and higher speeds seems to be decreasing.

When comparing the required speed of computation against available machines, one should be careful of high computational speeds achieved by special machine features requiring sophisticated software, either in the assembly/compiler programs or in the operational programs. Many machines can only achieve these high speeds when the program structure permits use of the special hardware features. As a more general point, one should not attempt to economize on hardware by assuming efficient, well-written programs—these are both difficult and costly to achieve. Well-trained, experienced programmers are in very short supply.

The second aspect of machine selection relates to the configuration required to achieve the desired reliability. Until quite recently, if a premium was placed on continuity of operation, a complete duplexing would be required. This was done with considerable success in SAGE. Two machines are available at each direction center, with only one being operational at a time. The machines are connected by a drum through which they can communicate, thereby permitting the second or standby machine to accumulate the dynamic data base required for rapid assumption of responsibility. The high reliability actually achieved from each machine, coupled with a programmed ability to recover from most intermittent errors, has led to very long mean-times between disabling system failures. Several years ago when I last checked, the mean free-time was about 20 days.

Today the computer situation is changing, and the selection of a machine or configuration of machines has taken on several different aspects. This changing situation results from the development of modular machines. In their first appearance, the modularity was restricted to memory capacity and in-out channel capacity. If more of either were needed, added modules could be connected. Today, the modular concept has extended to the central processor itself, and the truly modern machine design includes the capability of employing several processors operating in parallel and sharing the available memory and in-out modules. This permits what has been termed "multi-processing," with several processors operating together on a single job. (It is to be distinguished from "multi-programming," in which one machine works on several different tasks.)

Through multi-processing, modular machines can achieve very high effective operating speeds. However, except for those very real-time systems which might require operating speeds beyond the capability of a single central processor, the advantages of modularity lie other than in the direction of high speeds.

First, of course, is the capability for growth by addition of modules. Second, is the high reliability which can be achieved at a relatively low price; a full duplex is not needed, rather, only one or two spare modules are required for each different element. Further, with proper software design it is possible for individual modules to fail without complete interference with the system operation. This has been termed a "fail-softly" or "fail-gracefully" characteristic. Third, there is the ability to tailor the equipment—that is, the number or modules—to the situation or capacity required at different sites. That is, if a 300-aircraft control system were needed at one site, 6 memory modules might be required; but at a site requiring only 50 tracks, only 2 modules would be needed.

Parallel operation of computers is employed in the NTDS, or Navy Tactical Data System, in which a three-processor configuration is required to do the full fleet air defense task on a major ship. One machine does tracking, another does the intercept calculations, and a third processes display information and performs miscellaneous tasks. Under less than full-load conditions, two machines can be used with different programs to perform the same job at lower capacity. The third machine is then available for maintenance or other support tasks. It is also possible to operate at a one-machine level, and this is done on smaller ships not requiring intercept control capabilities.

The NTDS design, however, does not utilize a common memory; rather, the individual computers exchange information via in-out channels. The French STRIDA II air defense system employs multiple computers, each permanently assigned to specific tasks but sharing some common memory.

The full, multi-processing potential of the recent modular machines has not yet been realized in actual systems. The FAA design for an improved air traffic control system—the National Airspace System—will ultimately incorporate a full modular multi-processing capability, but initial system implementation will be along the more conventional lines.

The software design problems associated with exploiting a full multi-processing capability are quite significant. They include the problem of breaking down the program into small, relatively independent parts, and the design of adequate executive routines to handle the traffic, to sense modular malfunctions, and to manage the assignments and switching of modules. There are related hardware problems. It may be some years yet before it is desirable (or necessary) to spend the money and effort to solve these problems as long as the more conventional high-speed sequential machines in a simple duplex configuration are adequate to the tasks.

Display consoles

The situation with display consoles is quite the opposite from that of computers. There are relatively few "off-the-shelf" equipments, there have been only limited advances in performance or cost over the past ten years, and there are few systems in which the user is satisfied with his display consoles.

The situation is aggravated by the fact that it is extremely difficult to reach an agreement on requirements. The display console is one part of the system in which the operator normally takes a strong interest, and he naturally desires the utmost in performance. Unfortunately, however, he is an easy prey of the "brochuremanship" technique since he does not always appreciate the difference between a paper design and a working model, or between a laboratory model and a field-maintained production unit. The net result, then, is that his requirements are very high: much information; rapidly changing alphanumeric characters; flicker-free presentation; a bright display under high illumination levels; and possibly even color.

The designer finds it difficult to challenge the need, and the problems of complexity, cost, and maintainability are not received sympathetically by the operator. It is unfortunate that it is not generally possible to quickly put together and demonstrate various display capabilities so that the advantage or need of various features could be objectively determined before proceeding with production equipments.

Display consoles, then, represent a most difficult problem for the designer. While it may be impossible to completely satisfy the operator, he must be given a usable and reliable display. Economic factors cannot be ignored, particularly at the present production costs of $30,000 to over $100,000 per console. Added points of caution or consideration include:

- Whenever possible, a standard console design should be adopted, with very minor modifications for specific operating positions.
- Character or symbol sizes should be kept as small as possible within the limits of legibility. Special provisions (perhaps in the software) may be required to prevent overlap of symbology, as might result from adjacent aircraft tracks.
- The ambient lighting environment should be carefully understood or designed, particularly as this may cause reflections on the scope face.
- The general requirement for large viewing surfaces should be balanced against smaller viewing areas coupled with a capability for off-centering and expansion.
- The merits of alphanumeric characters as opposed to a limited symbolic capability should be matched against the differences in equipment complexity and cost.
- In addition to the console-operator interface, attention should be given to the computer-console interface. The display console is usually one of the earliest identified subsystems, but its design must carefully consider the interface with the computer, and particularly with the computer program. The tradeoff between the amount of computer programming involved in display formatting and the complexity of the console itself is not an easy one to make.
- Requirements for background or geography displays must be considered.

Large screen displays

The situation with large screen displays is not very different from that of consoles in that requirements are difficult to resolve, the user generally adds a multiple-color requirement, and the available systems are limited. A large number of techniques ranging from dry processes to wet processes, and from xerographic techniques to theater TV techniques, have been proposed, but only a few have yet reached the stage of working systems.

At the present time, silver halide systems involving projection of images photographed from the face of a CRT are available with processing times of about 10 seconds. The maintenance problems of such systems seem under control. Four-color systems using separate projectors or filters are used in a number of systems; color mixing has not generally been reduced to field practice.

Beyond reliability, a major problem is that of brightness, since the command posts in which these projected displays are used are normally kept at high illumination levels.

Input devices

Special-purpose action switches and general-purpose keyboards are the principal devices by which operators insert data into the system or influence the processing. Switches are quick and easy to use, but pose a problem when many different possible actions are required. In order to retain the simplicity of switch inputs while keeping the number of switches within reasonable bounds, some recent designs have utilized switches capable of performing several different functions by either manual or automatic label changes.

For inputs requiring more flexibility, particularly those involving variable-length items or alphanumeric messages, a keyboard is required. The problems of format and content errors have resulted in sophisticated equipments, generally termed "message composers," to assist the operator. Other designs have relied upon the computer to help the operator in the composition-formatting-error detection-error correction process by a feedback of computer confirmation or error information on printers. Standard teletype machines can then be used as the input device.

For operators working on consoles with plan-position displays, an ability to designate or point out selected positions can be achieved by a photoelectric cell "light gun" or "light pencil" or by movable display circle or "hook" controlled by a "tracking ball" (or "joystick"). Both of these types of devices can also be used as a more general input technique whereby the operator may select and designate quickly with his "pointer" among a number of alternatives presented in an alphanumeric message form on the display console. this technique, an extension of some earlier work on "electronic typewriters," appears to have considerable promise in command systems requiring rapid, lengthy, and flexible man-machine exchanges.

VI. Software

The computer programs, or software, are of central importance, since they direct the data processing equipment, and hence the operation of the system. They usually represent a sizable fraction of the total system design and development effort, and their cost in most systems is comparable to that of the computers themselves.

Nevertheless, the results achieved in this area leave much to be desired. The universal experience is that the magnitude of the computer programming activity, both in time and effort, is grossly underestimated. Further, the inherent potential of these programs for ease of modification has not been realized; in practice, and for a variety of reasons, the operational programs have not been flexible, and to change them has been costly and time consuming.

A first design problem, then, is to recognize the total size and scope of the programming task. In addition to the operational program itself—which, as noted earlier, should contain performance monitoring, data recording, checkout features, etc.—it is necessary to plan for the other programs required in the software production, checkout, test, and installation. These additional programs can be categorized as follows:

- Utility programs necessary for fabricating the operational program. These cover such areas as assembly, diagnostics, tape handling and processing, analysis, documentation, and control. As noted earlier, hopefully many of these programs can be procured with the computer itself.
- Support programs used to expedite the testing of the operational program during fabrication. These would provide the parameter and other data inputs required for such program testing phases as parameter testing of sub-programs, assembly testing of groups of subprograms, program shakedown, and system shakedown.
- The test data reduction programs required to evaluate the performance of the operational program during test phases.

- The operational data analysis programs required to support evaluation of system operation on a longer term basis.

The number and size of the programs in these categories vary with each different system. In SAGE, which pioneered much of the work on the types of utility, support, and data reduction programs needed for real-time systems, a very large effort was expended on the nonoperational programs. Further, the rates at which programs in the different categories can be designed and produced vary significantly.

Production rates on the operational program, in particular, may be an order of magnitude less than on conventional scientific and engineering programs. Because of its size, the operational program must usually be broken down into smaller pieces for individuals to work on. This complicates both the design problem and the subsequent assembly and checkout of the various subprograms. Because the subprograms may refer to or change common items of stored information, and because it is not always easy or desirable to freeze the design of the data tables at an early stage, special techniques—notably the use of common symbolic tags and compools—have been established. This, then, requires that sophisticated special-purpose compliers must be used. Further, since real-time data processing generally requires exploiting the speed of the machine, a good deal of machine-language programming may be required.

The production rates of operational programs are further lowered by the extensive checkout required. Each subprogram must be tested under a variety of input conditions—this is often called parameter testing —and then the programs must be assembled together and checks made on the continuity of operation. Finally, the entire program should be tested under a wide variety of operating conditions.

To illustrate these points, Table I shows some estimates on program sizes and efforts for a proposed real-time data processing system for which considerable related experience is available. All programs were assumed to be produced with a modified assembly program, with the exception of the bulk of the data reduction and analysis programs which would be written for a separate commercial machine using FORTRAN. The "production rate" includes all activities beginning with the program design activities (given program performance specifications) and terminating with the handover of a tested program, including card decks, listings, design and coding specifications, and manuals.

Table I
Program Production Estimates

Category	Size*	Production Rate†	Production Effort‡
Operational	50,000	70	700
Utility	40,000	250	160
Support	40,000	250	160
Test Data Reduction	120,000	1200	
	20,000	250	180
Operational Data Analysis	30,000	250	120
TOTALS	300,000		1320

*Single Address Instructions
†Instructions Per Man-Month
‡Man-Months

In this table, it was assumed that the computer came with the normal repertoire of assembly, loader, trap, and trace programs, and that the special-purpose equipment test programs required to check out and test various equipment subsystems—display consoles, input/output equipments, data links, etc.—would be provided by equipment suppliers.

Table I demonstrates two points of common experience:

- The size of the supporting programs is generally greater than the operational program by a factor of 2 to 5.
- The effort in producing the supporting programs generally equals and may exceed the effort on the operational program.

As an added word of caution, it should be noted that estimators of program sizes are traditionally optimistic (they base their estimates on what they think they themselves could do), yet the program is inevitably written largely by relatively new and unproductive programmers (experienced programmers generally graduate to writing sophisticated compilers or they become managers).

The key lessons in this size and effort area are:

- Identify and plan for all necessary computer programs at the earliest date.
- Do not underestimate the checkout and documentation activities.
- Do not assume program production rates normally achieved in scientific or business data processing programs.
- Expect reduced production rates as the magnitude of the operational program and number of subprograms grow.

A second major software problem is the organization of the operational program into relatively independent modules or subprograms. For example, should display generation functions be distributed among the various subprograms which may affect displays—tracking, identification, weapons assignment, etc.—or should they be grouped into one display generation subprogram? Economy of storage and operating time points to consolidation; flexibility for change points to distribution.

Once the functions of each subprogram module are determined, it becomes necessary to define the precise inputs and outputs, that is, the transfer function of the module.

The fixed and dynamic data storage tables offer many design choices. Here, too, efficiency of storage utilization usually runs counter to the design which offers the greatest flexibility.

A master or executive program is normally required to direct the sequential execution of subprograms, to handle transfers of tables and other data from the high speed memory to and from other storage media, and to handle the in-out and interrupt processing. The executive program must be capable of handling subprograms operating at different rates; some are required periodically, some only on demand. The executive program must also possess high flexibility for addition of new subprograms and for modifications of program sequence or periodicity. In summary, it merits the most careful design, including consideration of the instrumentation and testing requirements imposed during program assembly and checkout.

Throughout the software design, attention must be given to the need for and techniques of adapting a master operational program for use at different sites with individual characteristics or parameters.

During the design process, attention should also be given to the matter of response time, and, in particular, to the elapsed time from an operator input or request to the related computer output or response. Three times are involved: recognition of the input and routing to the proper subprogram, subprogram processing, and the final output processing. Proper design of the terminal equipment can minimize the initial and final steps. Some difficulty has been experienced in several systems with the subprogram processing. Several subprograms may be involved, and if searching of lengthy files is required, surprisingly long response time—on the order of minutes—can result. Priority processing of inputs and careful, and possbily redundant, table design may be required.

As noted, ease and rapidity of modification represent outstanding software design problems. The current lead times of months, and possibly up to a year, for modest field changes are clearly undesirable and tend to belie the name of software. With the availability of larger and faster machines, it is becoming possible to design the program into relatively independent, although less efficient, modular packages. If this is done, new program modules can be added and old program modules changed without requiring a major modification of the entire program. However, one should be wary of going too far to this extreme where ease and rapidity of modification may provide a temptation to unnecessarily modify the system.

A very interesting possibility, adapted in part from business data processing, is the development of general-purpose programs or software. This concept would exploit the similarity of many functional processes in the operational programs, particularly in command systems involving data manipulation. For example, the functions of file updating, file retrieval, message processing, display make-up, or report generation could be programmed in a very general form, and then adapted to specific applications by supplying the detailed descriptors for the files and the data. An extension of this technique would then permit *on-line* compilation of programs by operational personnel to achieve specific needs. Initial research in these areas appears to offer promising results, although not without large requirements for computer storage and operating time.

Finally, the initial program production effort must be carried out with due regard to the subsequent production effort. Current practice and direction is for assumption of these on-going efforts by military personnel. This places stringent demands on the documentation of the initial program, may influence the choice between problem-oriented and machine-oriented compilers, and requires integration of key military personnel in the initial design and production activity.

VII. Testware and testing

Throughout the system design and engineering effort, adequate attention must be given to the testware—the plans, equipments, computer programs, and procedures and techniques associated with the system test activities.

Early definition of the phases of the system test activity is essential. These phases might include design verification, the so-called Category I and II tests relating to acceptance and test at the manufacturer's plan and an initial field site, implementation tests at successive field site, a full-scale operational evaluation, and follow-experimental tests leading to improvements.

Again the names are not significant, and the nature and type of testing will vary from system to system. The significant point is that a decision as to the nature and types of test activity be made at a sufficiently early date to permit a determination of the requirements for special equipments and facilities, special computer programs, test teams, operational personnel, special flight tests, computer time, etc. This should then be followed up with the necessary plans, schedules, contracts, and other arrangements.

The manpower requirements for planning, designing, documenting, conducting, and analyzing a test program in a large system can be sizable, and suitable provisions should be made. The use of a separate contractor not associated with the design or production of hardware or software has considerable merit from the point of view of objectivity. It will also go a long way towards assuring that sufficient attention is given to the entire testware problem. Such a separate and independent test contractor was used in SAGE with considerable success.

Of particular importance is an early recognition that the test plans and schedules must take into account certain facts of system life. Not all tests will be conducted as originally planned; tests will be delayed and disrupted for a variety of reasons; and tests will uncover errors and deficiencies that will require substantial amounts of time for redesign and correction. Optimistic test schedules are not realistic.

In a multi-site system or one involving many remote input-output locations or connections, the sequential phasing of the test activity requires special attention. In all systems, a carefully generated, methodical plan for the availability and test of the subsystems and various sites is critical. To the greatest degree possible, there should be a capability to test and verify subsystem performance independently of the rest of the system. For this purpose, instrumentation—in the form of equipments and programs—must be provided.

The test activity is only meaningful if there are criteria against which measured performance can be checked, and if the system inputs and environment can be controlled when the performance is measured. It is necessary to determine what is to be measured, under what conditions this is to be done, and how much data is to be collected (the size of the sample). None of this is easy to do. Finally, the level of acceptable performance must be decided. This is a particularly difficult but necessary task. It generally requires much compromise and agreement among the designer, tester, and ultimate user.

Verification of test procedures and determination of the test measures should be conducted as early as possible, and can be one of the objectives of a design verification activity. In a multi-site system, Category II tests can provide the performance measures against which successive sites will be checked.

In designing the test activity, consideration must be given to the availability of trained operational personnel. Without adequately trained operators, it may be impossible to conduct useful tests. A corollary problem, of course, is the early availability of trained maintenance personnel.

Finally, conducting tests at a site while maintaining manual operations may pose severe problems. In an air defense system, for example, it is extremely difficult to share the use of search, beacon, and height-finding radars. These potential problems—and suitable operational procedures or equipment modifications—also require early consideration.

All in all, the experience to date has been that adequate provisions for the testware are not usually made, resulting in grossly extended and costly test periods. The technical problems of defining adequate test criteria and obtaining system inputs that are sufficiently controlled (or at least known) to allow meaningful measurements are not yet solved. We still do not expend sufficient time and effort in planning and preparing for the test activity well before it starts.

CONCLUSION

As noted, it is perhaps too early for an objective assessment of the development of the first generation of military real-time data processing systems. From the limited perspective of participating engineers and designers, this paper has attempted to summarize the key system engineering lessons of this experience.

It must be recognized that these lessons are not necessarily unique to military real-time data processing systems, but may generally be true of all large system endeavors. This realization, in fact, may be the most significant conclusion one could reach. Beyond that point, however, I should like to add further emphasis to several items:

- It is important to develop a proper appreciation of the full magnitude and scope of the effort at the start. The system engineer must take a very broad point of view, far beyond the confines of operational equipments and programs. In particular, the design must make provisions for a wide range of essential support functions and activities at the site and system level.
- The management structure and assignment of responsibilities for system acquisition can have a profound effect on the final product. Operational representation and inputs are essential; strong central design control is required regardless of contractual mechanisms and responsibilities.
- The proper matching of man-machine capabilities and the provision of adequate capacity and flexibility for change and growth are outstanding system design problems.
- Documentation and design verification are vital elements of the design process.
- Software problems are invariably underestimated, and much remains to be done to realize the in-

herent potential of these programs for ease and rapidity of modification.

- Testware design merits comparable attention with hardware and software.

Finally, and perhaps needless to say, system engineering is a necessary function in the system development and acquisition process. It does not "just happen"; it must be carefully planned and deliberately applied.

Software lessons learned in military command systems

by JOHN R. OTTINA
System Development Corporation
Santa Monica, California

INTRODUCTION

The first decade of implementing computer programs for military command systems has concluded. Looking back over this period, we can identify many major accomplishments and can enumerate many technical and management problems which have been solved. The first section of this paper presents a review of selected management/development problems that are successfully being dealt with. The second section discusses three other problems which have not yet been fully resolved and proposes techniques for solving them.

Selected review of lessons learned

Early efforts in implementing military command systems taught the developers several lessons which are now well recognized. Further activities are in progress in the military command system community to gather data, define procedures, set standards, *etc.*, in each of these known problem areas which will further refine the solution. A summary of some of these lessons follows. The problem areas reviewed are not all inclusive of those faced or for which a solution was constructed, but are only a subset which to the author seem to be of particular importance.

Cost estimates

Perhaps, one of the earliest lessons learned was that the initial estimates of the software design and development were typically over-optimistic. The complexity of the task and the number of people required to accomplish it were much greater than had been at first perceived. The value of thorough analysis and planning prior to the development of the command system in order to arrive at realistic estimates of cost, manpower, and schedules was quickly recognized. Today many attempts are being made to further validate cost estimation techniques, and as a whole the reliability and validity of schedules and gross manpower estimates are steadily improving. The need for full understanding of the task to be accomplished and for the publication of work plans before making commitments as to delivery schedules, costs and manpower is well recognized.

Timely initiation of software effort

The developers of a command system in many instances realized too late that they had failed to initiate software design early enough in the development of the system. Many of the early systems found that failure to initiate software design early enough caused the total system to be delayed and that the critical path, in many cases, was not hardware but software. The realization of the important part that software plays in the command system is now widespread and not only are schedule delays from this cause being minimized, but in addition many tradeoffs are made possible between software and hardware early in the design and development phases.

Establishment of requirements and control

Early efforts in the development of command systems frequently ran into difficulties because both the software developer and the procuring agency failed to gain a full understanding of the system to be implemented. Furthermore it was not recognized that the environment of a command system changes throughout the development cycle. The developer and the procuring agency failed to establish techniques which would allow concurrence on the system requirements and the initiation of procedures for processing future changes.

Today the picture is much improved. The last few years have seen major accomplishments in these areas. Specifications for the design requirements for the command control system are now produced by the procuring agency. Further detailed specifications are prepared by the developer and are reviewed and concurred upon before the actual production of the program begins. These specifications form the basis for the control of changes. Furthermore, the development of a series of reviews and boards or committees who constantly monitor and approve changes to the system has provided sound procedures for controlling the configuration of the system.

Software design guidance

Some of the initial efforts to develop command systems ran into difficulties because of failure to include sufficient guidance at the outset for the design of the software system itself. Far too much was left to the discretion of the individual constructing each of the separate computer programs. This error became apparent when the pieces of the system were put together. Not only were there inconsistencies and incompatibilities between portions of the system that were developed individually, but the system, when its elements were integrated, had large holes in it and would not function properly. The crystallization of design concepts and principles in the form of specifications has helped to eliminate many of these problems. These specifications have also more readily permitted the software developer to establish for his work, detailed and coordinated programming standards, common terminology and program conventions.

Documentation

Software development does not only mean writing computer programs, but includes a requirement for complete and accurate documentation. In the early days of system development this need was overlooked and no requirements were levied for documentation of the program system, its operation, its modification, etc. Today detailed program specifications, users manuals, flow charts and a wide range of related documents are recognized as being required for system implementation, operation and maintenance. In addition, standards are now being produced by the military which establish the requirements for uniform documentation in all command systems.

Quality control

The failure to include measures for controlling and verifying the quality of the command system software during its development led to a great deal of trouble. Developers and procuring agencies eventually came to recognize this failure and have taken steps to remedy it. A range of in-development tests, test standards, criteria for acceptance, demonstrations, etc., has been implemented as an integral part of command system development. These measures have provided increased quality of the software products, programs, documentation, etc. Some form of quality control has become a recognized requirement by both the developer and procuring agency. Several programs of test criteria, test standards, etc., are being investigated by the community at large.

Experienced personnel

A decade ago one of the most critical problems in system development was shortage of trained, experienced programmers, designers, implementation specialists, etc., to do the software task. Not only were there insufficient numbers of these people, but there was little recognition of the need to involve them throughout all phases of the development of the system. Ten years of working with computers and command systems have changed all this and there now exists a rather large number of experienced professions who are drawn upon to face the problems before us. This is not to say that the supply of these people is adequate or that there won't be serious shortages of trained people in the future. In addition, the need for involvement of software oriented people through all phases of the development of the command system is well recognized. The software developer knows as a matter of routine that he must continue to train and add to this limited people resource.

User's role

For all the advancement in the technology of automation, the developers of early command systems failed to pay sufficient attention to the human element in the system. Complex language, techniques, and methods were used which were strange and incomprehensible to the user. Recognition of this failure has seen the user become more involved in the development phase and the developer more cognizant that his product was meant to be used. In particular the developer has implemented human engineering techniques, user documentation and instruction or training in the use of the system. This application of the systems approach from both ends of the development process has resulted in a narrowing of the gap which would otherwise exist between the designer and user of the system.

Suggested solutions to other problems

The techniques employed to solve the problems described in the previous section have been demonstrated, and are now well known and routinely employed. Continued research and refinement in most areas is in progress. There exists, however, a second class of problems for which the proposed solutions are more obscure, unproved or controversial. Three examples of this class of problems are discussed in the succeeding sections. For each problem a technique which has been successfully employed is proposed as a solution.

Circumventing the unavailability of non-functional software

Typically the computer is available several months before the supporting software. One of the critical problems in the development of the operational computer program is unavailability of this non-functional software. Non-functional software, that is, utility and support computer programs that are required by the developer to construct the operational computer program, must be written in advance of the operational program.

The problem facing the developer of command systems who encounters this situation is how to proceed without the non-functional software. The popular solution attempts to identify those portions of the non-functional software that are required earliest in the development of the system and to commit resources to accomplish those tasks. As a result, the nonfunctional software is implemented on a piecemeal, incremental basis. In most cases, this approach is adequate but it does not eliminate delay, and managers have been led to look for more satisfactory solutions. An alternative solution to the problem is the use of nonfunctional software which has been constructed in higher order language for a prior application on a different computer.

To make this nonfunctional software usable for the new computer, theoretically all that needs to be done is to modify the portion of the compiler that converts the higher order language or the intermediate language to the binary representation required by the new computer. This can be accomplished in two stages. Stage one is to modify the compiler to operate on the original computer as before, but to produce the binary code acceptable to the new computer. Stage two is to use the modified compiler developed in stage one to compile itself, thereby obtaining a "new" compiler which now operates on the new computer. We now go on to recompile the remaining portions of the non-functional software with this "newly" constructed compiler. In actual practice the compiler will require modification as there will undoubtedly be some unique aspects of the input/output function of the computer that must be dealt with. In addition the code generated will not be optimum. This latter factor can be dealt with on a time available basis and will not seriously impede the development of the software for the operational system.

To accomplish the modification of the translator portion of the compiler takes some time. In the meantime, the developer can use the nonfunctional software with the computer other than the one in which it all finally operates. By using a higher order language which is compatible to both the old computer and the new one, actual code generation and logic checks can be made. When the "new" computer equipment is ready the code already generated can be recompiled for the new computer. The amount of scheduled linear time lost in this effort is very small.

The use of the compiler in this way requires detailed plans. Only that nonfunctional software must be used which will be available in the new computer. Well defined, limited programming language, coding conventions, and techniques are also required. In most cases these can be prepared in initial form very early in the development cycle.

Aspects of the approach described have been successfully used in a recent application. The amount of time lost in the recompilation and in verifying the new compiler was less than one week. This accomplishment was made possible by careful preplanning for the recompilation and the saving and rerunning of recorded input and outputs from tests made before the recompilation. The preplanning included carefully documented restrictions on the use of legal higher order statements, declarations, etc. In addition some portions of the command system which were highly computer-dependent were deferred until the nonfunctional software for the computer was available.

In summary, the recompiling technique described above has been successfully used with careful preplanning at a considerable savings in schedule time.

Circumventing the unavailability of hardware

The unavailability of the computer itself, some portions of it, or some interface component may also hinder the development of software for the command system.

One way to alleviate the problem is to simulate the operation of the missing element with the computer. This approach has been used in many instances and has included the simulated operation of the total computer by another computer as well as the simulation of the interfacing hardware. Generally, however, the simulation programs must be developed and this additional work causes schedule delays.

Other approaches are possible. One which has been used is similar to that described in the previous section. Recently a particular configuration of display equipment and computer was scheduled to be available for computer program development only three months before the desired operational use of the command system. Unless the display portion of the command system was verified prior to this three-month period, the operational use date could not be met. There existed elsewhere a similar display configuration which was mated with a different computer. Both computers used, or were scheduled to have available, the same higher order language (JOVIAL). The approach decided upon was to code the command system in the higher order language. Early checkout and display verification was accomplished using the available display system mated with the "wrong" computer. When the operational configuration became available the computer program was recompiled on the "right" computer. This recompiled computer program was then used with the operational configuration of display equipment and the new computer for continued verification. As a result the desired schedule was achieved. The technique described could be employed to circumvent the nonavailability of other pieces of equipment including the computer itself. This technique, however, does not eliminate the need for use of the operational configured equipment. It only delays and reduces the period of time for which it is required.

Again, the same preplanning precaution described in the previous section must be taken. The probability of a successful transfer of a high order coded program from one computer to another computer is significantly increased if the program is constructed with transfer in mind.

Management for change implementation

The developer of software for a command system is faced with the knowledge that during the development process changes to the software will be required. Changes to an in-development computer program can affect both the quality and the schedule. A study of the number of errors corrected in a command system during a one-month period demonstrated that each time a change was introduced the error rate increased. A clear correspondence was found between high error rate and the number of changes. The problem that faces the developer is how to accommodate change without adversely affecting quality or lengthening the schedule.

The impact of a given change varies not only with its size, complexity, and system involvement but also as a function of the stage of development of the software. If a change is identified early in the design phase it may be possible to include it in the command system with no impact on quality or schedule. On the other hand if the change is introduced into the system during later stages of testing it may seriously affect the quality of the problem or delay its completion or both. In general the later a change is introduced in the development process the greater its impact will be.

With this rule of thumb and the knowledge that change is inevitable, two courses of action are suggested in order to manage the impact of change: Make full provision in the design of the military command system for change implementation, and cut off all changes during the later phases of testing and plan for incremental modification packages to accommodate subsequent changes.

Some goals to consider in designing a system to allow for changes include the following:

- Fractionate the system into subprograms so that the functions are as isolatable as possible. That is, try to construct the subprogram so that changes to a function will require changes only to that subprogram.
- Construct central tables containing items, values, parameters, etc., used by more than one program. Thus, if a value is changed the correction need only be made in one place.
- Define with care all communication items between programs and systems. Document these carefully.
- Define carefully rules, conventions and procedures to be used by all programmers to facilitate and expedite understanding. Prohibit involved and tricky programming techniques.
- Insist that programmers provide documented program comments which will permit other programmers to understand readily what is being done.
- If possible, use a higher order language and maintain the program in that language. Do not resort to corrections in binary code if it can be avoided.
- Define tests and results such that if a change is made, it can be easily verified that other functions are still operating properly.
- Where possible, for each parameter define a range within which the parameter can be meaningfully varied.

The second action suggested is to determine a point in time when no other changes will be accepted, and to test thoroughly this system base. Given a tested base, further changes can be accommodated with increased confidence that they will have a low impact on quality and schedules. It is generally convenient to collect a set of these later changes and develop them as a unit.

The combination of the two courses of action has been successfully employed with improvements in the quality of the program and the achievement of greater overall system capability in shorter time than would otherwise have been possible.

Real time display techniques for military command information systems

by ANTHONY D. RUGARI* and CARMINE J. SALVO
Rome Air Development Center
Rome, New York

INTRODUCTION

Rapid growth in complexity of command and control systems has placed severe requirements on associated display subsystems. Existing display technology is generally inadequate, and it is becoming increasingly difficult to meet the operational requirements dictated by present and future systems. Current systems call for displays that can handle large volumes of data in real time and are bright, multicolor, of high resolution and highly flexible. There are no existing displays that can provide this total capability. New techniques must be provided to ensure that the display does not become the weak link in the information system chain.

In order to provide continuing solutions to the display problem, an extensive R&D program has been established in the Rome Air Development Center's Display Branch. This program is multifold and provides for engineering support for systems now in existence or under development, engineering development of display systems for near future requirements, and research of new techniques and ideas to provide a basis for the advanced displays of the future. As a result of this program, several display disciplines have been extensively studied and show promise towards providing displays in consonance with the stated objective of the program. This paper will describe work under way in the following two areas, laser displays and light valve displays.

The recent development of the laser as a practical, continuous, coherent light source has created a new display technology, that of the laser beam display. In concept form, one thinks of the laser beam as being analogous to the electron beam, however, some major differences exist. While the laser beam is an uncharged beam, thus requiring the development of new techniques for its control, it is not constrained to a vacuum environment nor does it require a special screen for the emission or control of light as does the electron beam. These advantages thus prompted further exploitation of the application of the laser to displays. The most general laser display concept is shown in block diagram form in Figure 1. A description of the concept can be found under the heading ***Laser Display***

Light valve displays offer several possible solutions to the display problems mentioned above. Light valve displays include all those devices which use an electron beam to change an optical parameter of a control layer. These changes, in conjunction with special optics, modulate light supplied by an external light source in order to produce a display image. The electron beam effects changes at discrete points of the control layer. In this manner the image projected onto the display screen is an optical replica of the charge image that the electron beam has deposited. By using a light valve one retains the flexibility and simplicity of electron beam control of CRT devices. In addition, one gains in image size and brightness. Image size is controlled by the projection optics, while brightness is limited only by the external light source and the heat dissipation capacity of the control layer. A light valve may be expected to provide in a large bright display, selective address capability, high resolution, high writing speeds and controlled persistence available in CRT's. A description of the RADC light valve program is contained under the heading *Light Valve Displays.*

Laser display

In its simplest breakdown, the laser display consists of three major items, the laser source(s), the light modulators and the light beam deflector. The laser sources which have been considered provide a continuous output in the visible spectrum and are of the gaseous variety. Other sources, such as semiconductor diodes having pulsed outputs in the visible region can be considered when the technology allows. Pertinent characteristics of the laser sources as applied to the display problem are discussed later on.

A great deal of research has been expended on the

*Presently with The Night Vision Laboratory, U. S. Army Electronics Command, Ft. Belvoir, Virginia.

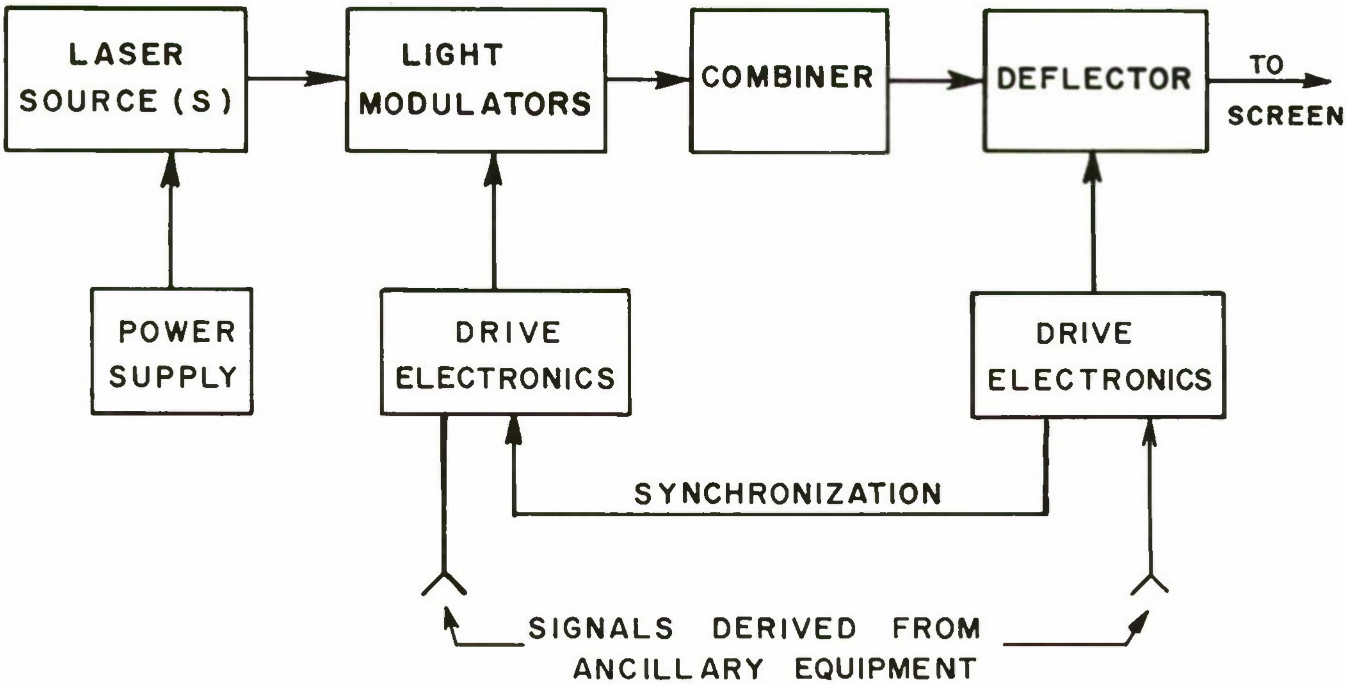


Figure 1—Generalized block diagram of laser display

light modulation problem for various laser applications. The state-of-the-art has advanced to the point where a DC—30 MH_z baseband modulator is practical thereby providing high resolution displays. The theory and application of these devices are also discussed.

The problem of deflecting a photon beam is considerably more difficult than that of deflecting a charged electron beam. Many deflection techniques are under consideration. The electromechanical resonant scanner is described in detail since it appears to be the most promising for achieving a high resolution raster display. Several versions have been built* including a 525 line piezoelectric device operating at 15.75 KH_z, and a 945 line magnetostrictive device operating at 28.35 KH_z.

Finally, it is important to consider the image which results from a laser display. One major observation of the visible laser beam is a "sparkling" sensation to the viewer due to diffraction and interference effects. An experimental model of a laser television system has been built which assisted in making some general observations regarding this effect. The initial version provided a 525 line monochromatic laser image, and a follow-on effort is presently under way to develop a model which will provide a 525 line multicolor laser image.

*The piezoelectric and magnetostrictive scanner were both developed by the Texas Inst. Co. under RADC contracts AF30(602)-3271 and AF 30(602)-3731 respectively.

A. Pertinent Characteristics as Applied to Displays

The laser properties of interest for the display application are:

- Spatial Coherence
- Temporal Coherence
- Color
- Intensity

1. *Spatial coherence*

Conventional light sources emit wavefronts whose spatial phase characteristics cannot be predicted. They are said to be incoherent sources of energy. In contraspect, radio waves and microwaves generated by man, represent coherent sources of energy. A wavefront emitted by such oscillators allows a high degree of phase correlation as a function of time and spatial variations. This fact makes possible the sophisticated design of present day radar equipment.

The laser is a coherent source of light. A gas laser operated in the fundamental transverse electromagnetic mode, TEMoo, can be considered as being a diffraction limited beam. Consider the optical system shown in Figure 2. The diffraction limited diameter is given by:

$$d = \frac{2.4\ f\ \lambda}{D} \qquad (1)$$

where:

d is the diameter of the focused spot
f is the focal length of the lens
λ is the wavelength of the illuminating source
D is the diameter of the lens.

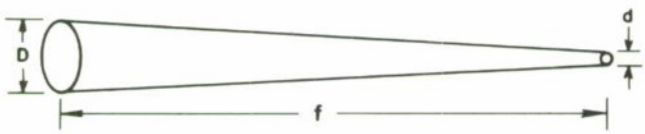


Figure 2—Simple optical system

If the system is illuminated with laser energy of λ equal to .6328 microns with D equal to .127 cm and f = 3.81 cm, the minimum diameter *d* which can be achieved is approximately 46 microns. The percentage of incident energy within this diameter is ≃ 84% (Born & Wolf, 1959). Thus, for a 3 milliwatt laser at .6328 microns (relative visibility factor of the eye for this wavelength is .24), the flux within the focused spot of the above diameter, is approximately 0.4 lumen.

An ordinary light source can be made spatially coherent by placing a very small iris in the path of the beam at some extended distance from the source, allowing only those rays with the desired directionality to pass through. With appropriate collimating optics, one can then make these rays parallel. The main problem is the large loss in luminous flux which must be suffered. The flux (W) from an incoherent source in a diffraction limited beam is given by (Hopkins, 1964):

$$W = BA\,\omega \qquad (2)$$

where:

B is the radiance of the source

A is the area of the Airy disc

ω is the solid angle subtended at the source by the collecting lens.

The diffraction limited diameter of the focused spot was given in Equation 1. The area of the Airy disc is then:

$$A = \Pi \left\{ \frac{1.22\,\lambda\, f}{D} \right\}^2 \qquad (3)$$

The measure of the solid angle (ω) subtended at the source by the collecting lens is the area cut out by the cone from the sphere divided by the square of the radius of the sphere or

$$\omega = \frac{\Pi\, D^2}{4\, f^2} \qquad (4)$$

and combining the preceding equations, we obtain for the flux

$$W = \frac{(1.22\,\pi\,\lambda\quad)^2 \quad B}{4} \qquad (5)$$

If we consider a 1,000 watt mercury arc which has a radiance of 10 W/cm²/steradian in the .5461 micron line (at this wavelength, the conversion from watts to lumens in 620 lumens/watt, the luminous flux is W = .68 × 10^{-4} lumens. Therefore, we can see a four order of magnitude difference between the flux collected from the two sources, *i.e.*, the 3 milliwatt laser and the 1,000 watt mercury arc lamp.

This property also allows us to consider limited aperture devices as practical for both the modulator and deflector components. It can be shown that a reduction in modulator drive from 7,000 volts to 400 volts can be attributed to the spatial coherence property of the laser.

2. *Temporal coherence and color*

The temporal coherence property of the laser implies a very narrow linewidth or monochromaticity of the emitted radiation. From a display point of view, the narrow linewidth means that we can consider the basic laser source as a 100 per cent pure or a fully saturated color. By appropriate choice of the three basic sources comprising a color additive scheme, it is possible to reproduce a wider spectrum of colors than can be achieved with other color display systems. Using the He-Ne laser (output at .6328 micron) and the Argon laser (outputs at .5145 micron and .4880 micron) to provide the three primaries, we can plot the range of colors which may be reproduced. The color triangle which results is shown in Figure 3. A comparison is made to the present day color TV range. It is very evident that a higher frequency blue will be required to provide faithful reproduction of the magenta and purple portion of the color spectrum.

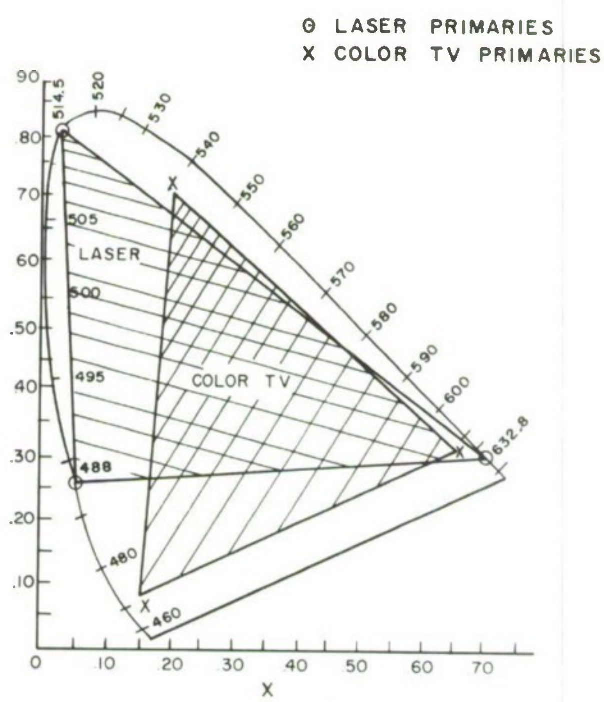


Figure 3—Plot of 3 CW visible laser sources on CIE chart

3. *Power and intensity*

An empirically derived expression which relates the brightness of the display image to the laser power and other system parameters is given by Equation 6.

$$B = \frac{C_1 \; G \; L_p \; T}{A} \qquad (6)$$

where

$$L_p = P \; K_\lambda \; C_2$$

B is the highlight brightness of the area under consideration expressed in foot lamberts;

C_1 is the transmission efficiency of the system giving the luminous flux incident on the screen expressed as a fractional part of the total luminous flux available from the source;

G defines the directionality of the screen relative to that of a lambertian surface, expressed as an integer;

A is the area of the elemental area under consideration, or the area of the minimum resolvable element expressed in square feet;

L_p is the peak luminous flux available from the source, in lumens.

T is the duty cycle or the ratio of the dwell time of the laser beam on the area under consideration expressed as a fractional part of the total available time (the repetition rate of the laser beam on any one element of the display is chosen to be above the critical fusion frequency of the eye, to prevent a flickering image);

P is the laser power in watts;

$K\lambda$ is the relative visibility factor of the eye for the particular wavelength under consideration and

C_2 is the conversion factor from watts to lumens at .5550 microns and equals 680 lumens/watt.

It is seen that the brightness which can be achieved is a direct function of the duty cycle as well as the laser power. This implies that the method of scan and display format are extremely important factors in determining the brightness of the display. The following examples are chosen to demonstrate this point.

Let us consider a one watt Argon laser as the basic source of a monochromatic laser display. The average visibility factor is 0.40 giving a luminous flux of 272 lumens from the source. Then let us impose the conditions that the system has an optical efficiency of 0.5, the display image has an area of 100 square feet, the display image is composed of one million resolvable elements and the screen is a lambertian diffuser having a gain of one. If the deflector provides a raster type scan, all of the resolution elements are accessed in one frame period giving a duty cycle of 10^{-6}. The brightness which would be achieved is 1.35 foot lamberts. If the deflector is flexible enough to allow random or selective accessing of the information to be presented to the viewer, then only those resolution elements containing information would have to be accessed in one frame period. For a military display, this number is on the average about 2.5 percent of the total resoultion elements. Thus the duty cycle would be 4×10^{-5}, giving a potential display brightness of 50 foot lamberts.

Various gaseous lasers are available today which can provide cw visible output. Most notably, for the display application, are the He-Ne laser and the Argon and Krypton ion lasers. Commercial models of the He-Ne laser can provide 100 milliwatts of power at .6328 micron. Commercial models of the Argon laser can provide 5 watts of continuous power at dominantly .5145 micron and .4880 micron and the Krypton ion laser can provide 250 milliwatts of cw power at .6470 micron.

B. Electro-optic (E-O) modulators

There are various mechanisms available for effectively varying the amplitude of the laser beam. The wide bandwidth (DC to 30 MH_z) required to achieve a high resolution display necessitates the use of an electro-optic or similar high speed device. Two categories will be described, one which is referred to as the linear e-o effect and the other as the quadratic e-o effect. The particular effect which applies is a function of the symmetry class of the e-o crystal.

1. *Linear E-O effect*

This effect is exhibited by crystals falling into the $\bar{4}2m$ crystal class. Some representative types are in the class of $X H_2 PO_4$, where X can be potassium or ammonium. This effect has been widely described in the literature (Billings, 1949 a,b 1952, Carpenter, 1950). The classical mode of operation is shown in Figure 4. The dominant characteristic of this classical model is that the direction of the applied electric field is parallel to the direction of propagation of the light beam.

Consider the schematic shown in Figure 4, minus the quarter wave plate. The collimated input beam is converted to linearly polarized light by the first polarizer. With zero electric field applied to the e-o crystal, it is uniaxial and if the direction of propagation of the polarized light is parallel to the optic axis of the crystal, the light propagates undisturbed through the crystal. Since the analyzer and polarizer are oriented with their preferred directions orthogonal with respect to each other, none of the light is transmitted through the analyzer. When an electric field is applied along the z

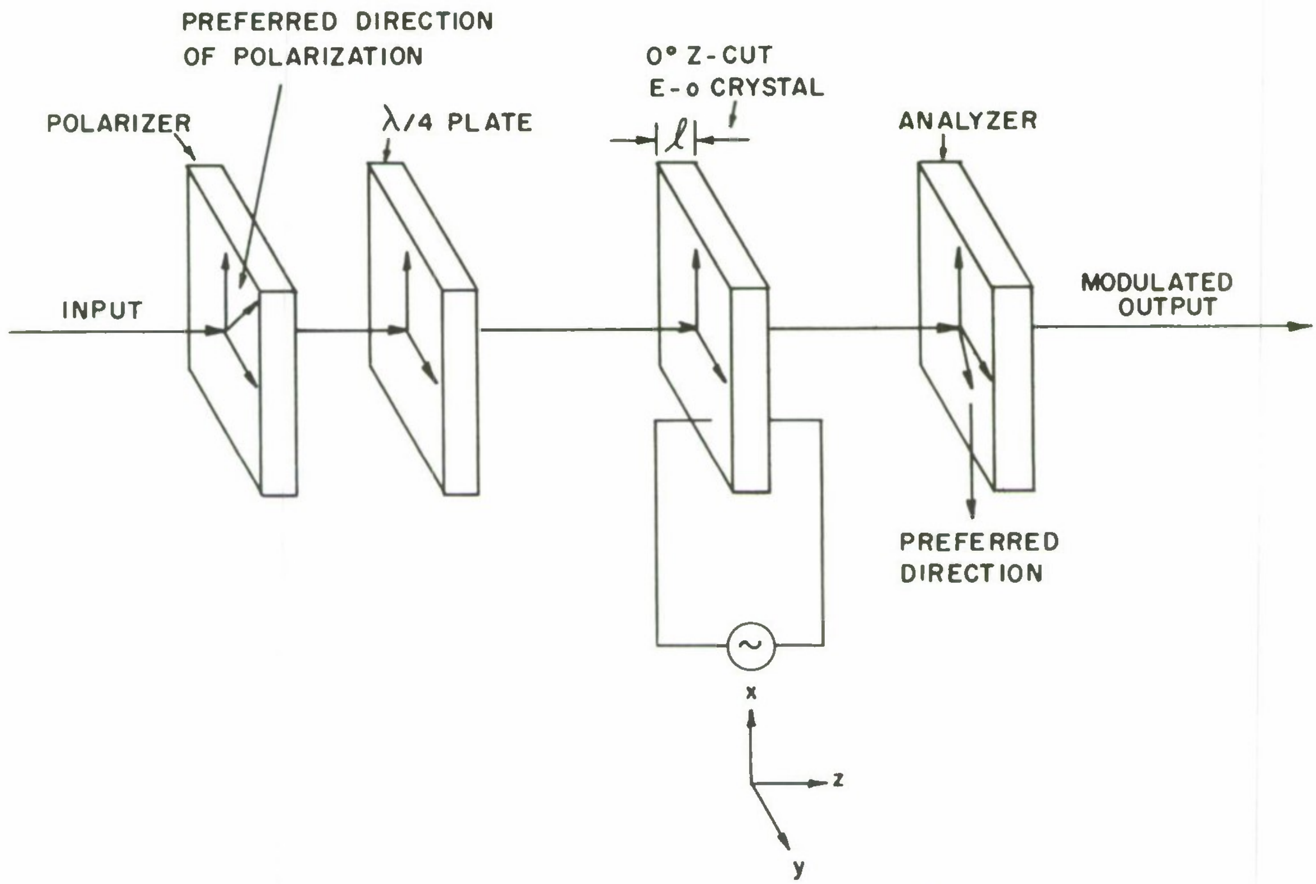


Figure 4—Longitudinal Pockel's effect modulator (electric field parallel to direction of propagation of light)

direction of the e-o crystal, it becomes biaxial. The index of refraction is different along the two axes and the difference is given by:

$$\Delta n = \frac{\lambda \tau}{d} \qquad (7)$$

where

- Δn is the change in the index of refraction
- λ is the wavelength of the incident light
- τ is the retardation expressed as an integer
- d is the crystal thickness in the Z direction

and

$$\tau = \frac{r_{63} V_z n_o^3}{\lambda} \qquad (8)$$

where

r_{63} is electro-optic coefficient in the case where the electric field is applied along the Z direction, meters/volt

V_z is voltage in the Z direction, volts.

When the voltage applied to the crystal is sufficiently large to produce a retardation of $\tau = \frac{1}{2}$, all of the light is transmitted through the analyzer. In the model shown, a nominal voltage required to produce the half-wave retardation is 7,000 volts for KDP.

The primary function of the quarter wave plate is to provide for operation along the linear portion of the modulation transfer curve and is not discussed here.

It is possible to design a linear e-o modulator which takes advantage of a long transmission path compared to aperture dimension (Baker, 1965). A schematic diagram is shown in Figure 5. Here the electric field and direction of propagation of the light beam are orthogonal, and the device is referred to as a transverse modulator. The total phase shift for an incident light

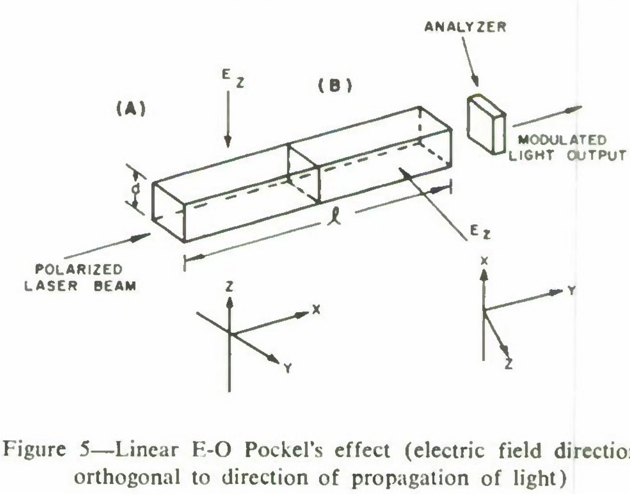


Figure 5—Linear E-O Pockel's effect (electric field direction orthogonal to direction of propagation of light)

beam polarized at 45° to the y axis of crystal A or x axis of crystal B is given as follows (Wenz, 1964):

$$\Delta \phi = \phi_x - \phi_y = \frac{2 \Pi l}{\lambda}[(n_x + n_z) - (n_y + n_z)] \quad (9)$$

or

$$\Delta \phi = \frac{2 \Pi l}{\lambda}(n_x - n_y) = \frac{2 \Pi l}{\lambda} \Delta n \quad (10)$$

For equation 10 to be valid, the lengths of crystals A and B must be held equal to within .0001″. Substituting for Δ n where:

$$\Delta n = \frac{r_{63}\, n_o^3\, V_z}{d} \quad (11)$$

we obtain

$$\Delta \phi = \frac{2 \Pi l,}{\lambda} \frac{r_{63}\, n_o^3\, V_z}{d} \quad (12)$$

which shows the dependence of the retardation on the l/d ratio.

Some typical crystal parameter and geometry values for this type modulator are as follows:

$r_{63} = 5.25 \times 10^{-6}$ microns/volt (KDP)
$n_o = 1.468$
$\lambda = 0.6328$ microns
$l = 7.62$ cm.
$d = .18$ cm.

and to achieve 100 per cent modulation, the half wave voltage required is approximately 440 volts. This compares favorably to the 7 KV for the longitudinal modulator. A photograph of a typical e-o modulator is shown in Figure 6.

2. *Quadratic E-O effect*

Many isotropic materials when placed in an electric field behave like a uniaxial crystal with the optic axis in the direction of the field. In this case, the induced birefringence is a function of the square of the applied electric field and the phenomenon is called the Kerr effect or the quadratic e-o effect.

The Kerr effect in liquids is a well-known phenomenon and is discussed in many textbooks. A commonly used liquid is nitrobenzene and for a cell having a 2.5 mm aperture (electrode spacing) and a 2 cm. length, the nominal voltage required for 100 percent modulation is 6,500 volts. This value is large in comparison to the 440 volts required of the modulator previously described.

However, there exists a unique class of crystalline elements, of the Perovskite family, which in their paraelectric phase exhibit a strong quadratic effect. The dependence of the applied field as a function of the crystal properties and as a function of the number of Π phase retardations has been shown to be (Geusic, 1963, 1964)

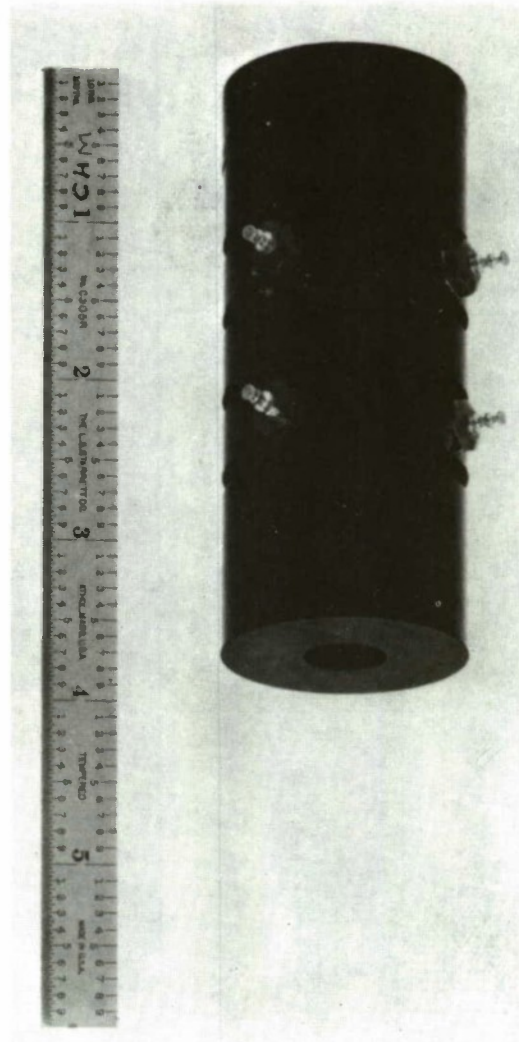


Figure 6—E-O modulator

$$V_{(m \Pi)}^2 = m \left[\frac{\lambda d^2}{n_o^3 (g_{11} - g_{12}) \epsilon^2 l} \right] \quad (13)$$

where

- V is the voltage applied to the crystal (volts)
- m is the integral number of the Π phase retardations
- λ is the vacuum wavelength of light (meters)
- $(g_{11} - g_{12})$ is the quadratic e-o coefficient m^4/c^2 m^4/c^2
- d is electrode spacing (meters)
- l is crystal length in direction of light propagation (meters)
- ϵ is static dielectric constant, farads/meters and
- n_o is index of refraction in absence of electric field.

Equation 13 is valid when the electric field is applied along a (001) direction and the light propagates in the (100) direction.

Reported data for $KTa_{.65}Nb_{.35}O_3$ (KTN) which has a Curie Temperature of ~ 283°K is listed in Table I.

Table I—Properties of KTN for λ = .6328 micron and T - 295° K

Parameter	KTN
$g_{11} - g_{12}$	0.174 m^4/c^2
n_o	2.29
Q	~ 150
ε	10,000
P_s (saturation polarization)	10 μ C/cm^2

Since the effect is quadratic, a dc bias can significantly reduce the differential half-wave voltage required for 100 percent modulation. This can be readily seen by looking at the following example. Let us choose a block of KTN having the dimensions d = 2mm and l = 10 mm. The voltage required to achieve a phase retardation of Π, 2 Π, 3 Π, etc. is shown in Table II. For 100 percent modulation, it is only required to achieve a differential phase shift of Π radians. Thus, if we apply a dc bias of 330 volts to the crystal described above, the required ac peak drive is 22 volts. This will allow us to go from a full "off" to fully "on" condition of the modulator. A comparison of the KDP transverse modulator and the KTN modulator is shown in Table III.

Table II—Voltage vs. phase retardation in KTN

Phase Retardation	Voltage (volts)	Differential Voltage for Π Phase Retardation (volts)
Π	124	
		52
2 Π	176	
		39
3 Π	215	
		33
4 Π	248	
		30
5 Π	278	
		27
6 Π	305	
		25
7 Π	330	
		22
8 Π	352	

Table III—Comparison of KDP transverse modulator and KTN modulator

	KTN	KDP
Geometry	2mm × 10mm	2mm × 77mm
Drive Voltage	22 volts	440 volts
DC Bias	330 volts	———
Dissipated Power	.54 watts	.47 watts
Required Power	40 watts	46 watts

It is possible to achieve a baseband modulation bandwidth of 30 MH_z with either of the transverse modulators described.

C. Laser deflector

There are various known phenomena for deflection of light beams. The unique properties of the laser allow certain techniques to be considered and others to become practical that normally could not be considered or were not practical with ordinary light sources. A categorization of the various techniques applicable to displays and the state of development is shown in Table IV. Two of the more promising techniques for raster displays are the resonant piezoelectric, and magnetostrictive type scanners. Techniques applicable to a random or selective type accessing format are in the research stage and are not described here.*

1. *Resonant piezoelectric scanner*

A unique electro-mechanical scanner has been developed for specific laser display application by the Texas Instruments Co. under RADC contract AF30-(602)-3271. The principle of operation is as follows. A resonant piezoelectric driven fiber and mirror combination produce a circular scan of light at the horizontal repetition frequency for the raster line standard desired. The circular scan is converted to a linear scan by means of a fiber optic assembly. The vertical scan, 60 H_z repetition frequency, is generated by a galvanometric driven oscillating mirror. The primary reason for first generating a circular scan is that this allows the piezoelectric cartridge to be operated at resonance and thus a large angular deflection is achieved, of the order of 12 degrees. With this magnitude of deflection and considering the diffraction limited condition, the scanner has the capability of resolving 1,000 horizontal elements. A schematic diagram of the piezoelectric scanner is shown in Figure 7. The various constituents

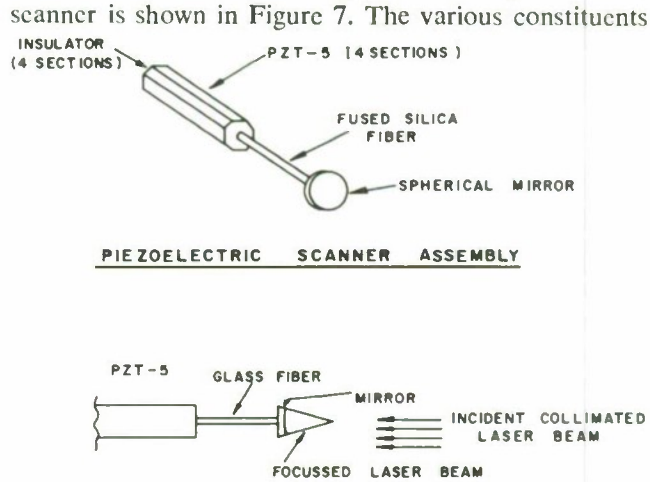


Figure 7a,b—Mirror position—unexcited state

*Selective access laser display beam positioner study with Hughes Aircraft Co., Malibu, Calif. AF30(602)-4097.

Table IV—Laser beam deflection techniques

Category	Specific Type	State of Development (Outstanding Characteristic)
Electro-Mechanical	Rotating Drum Scanner	Operational (limited to raster scan at repetition frequencies of 10 to 15 KHz)
	Resonant iPezoelectric Scanner	Operational (can provide raster scan at repetition frequencies of 15.75 KHz [525 line] and 21.87 KHz [729 line])
	Resonant Magnetostrictive Scanner	Operational (can provide raster scan at repetition frequencies of 28.35 KHz [945 line] and 30.87 KHz [1029 line])
Refraction	Electro-Optic Prism	Research stage (materials limited and resolution limited)
	Ultrasonic Cell	Research stage (resolution limited)
Diffraction	Debye-Sears Effect	Research stage (transmission limited)
	Bragg Angle Incidence	Application stage (characteristics not completely determined, however appears promising for a raster scan)
Coherent Scanner	Coherent Optical Phased Array	Research stage (requires high degree of stability and is resolution limited)
Binary Electro-optic Deflector	Convergent Beam	Research stage (main problems are materials limitation and low switching speed)
	Wollaston Prism Type	Research stage (mainly in idea stage)
Multidirectional Laser	Various types	Idea and early research stage (no prediction)

of the scanner are shown in Figure 7a. Bonding of the interfaces is accomplished with an epoxy cement. The position of the focussed laser beam, with no excitation applied to the scanner is shown in Figure 7b. In Figure 7c, one of the many possible positions of the focussed laser beam with the piezoelectric cartridge excited, is depicted. The electrical connections of the piezoelectric cartridge are shown in Figure 7d.

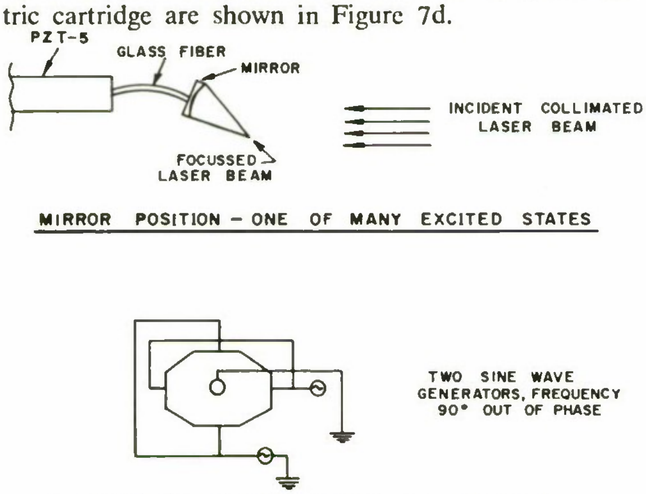


Figure 7c,d—Piezoelectric scanner, conceptual

A photograph of the piezoelectric scanner is shown in Figure 8. A photograph of the fiber optic circle to line converter is shown in Figure 9.

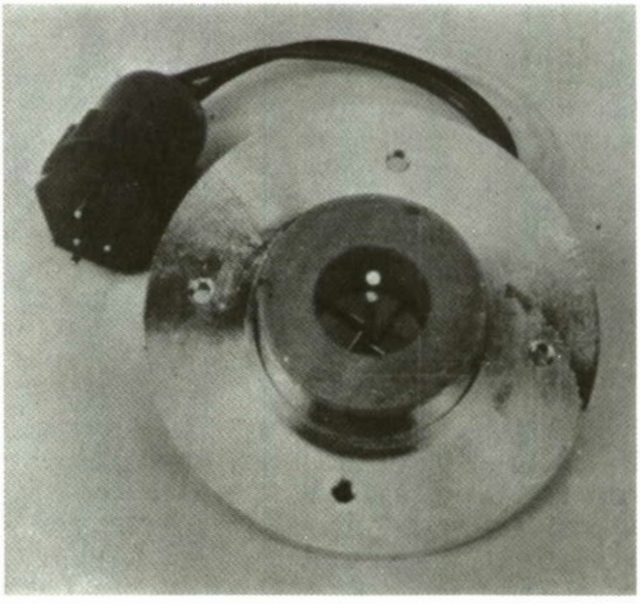


Figure 8—Piezoelectric scanner

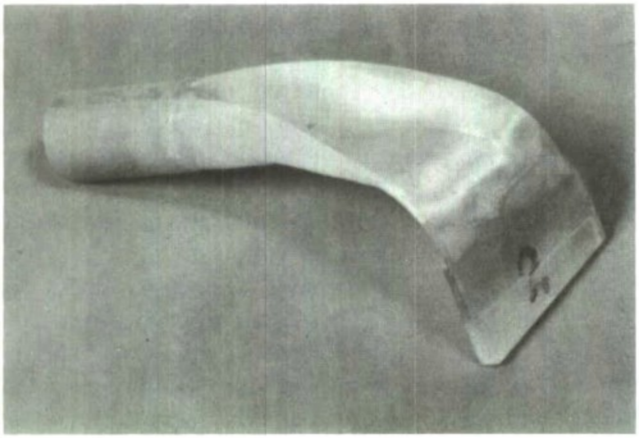


Figure 9—Fiber optic circle to line converter

The motion of the fiber is governed by the equation of motion of a flexural wave in a beam. Detailed discussion of its operation has been previously reported (Baker, C. E., 1965). Some typical dimensions of a scanner built for operation at 15.75 KHz are:

- Clevite PZT-5 Cartridge, 60099
- .005″ diameter fused silica fiber having a nominal length of .150″
- Spherical mirror having a diameter of .050″ and thickness of .003″. Nominal focal length is 1.5″.
- For a 12° deflection, the two sinusoidal signals required are nominally 100 volts peak-to-peak

A photograph of the vertical scanner is shown in Figure 10. The significant achievement is that a linear scan was obtained with a 1 millisecond flyback time. A total deflection of 10 degrees was used from this scanner, although large deflections are possible.

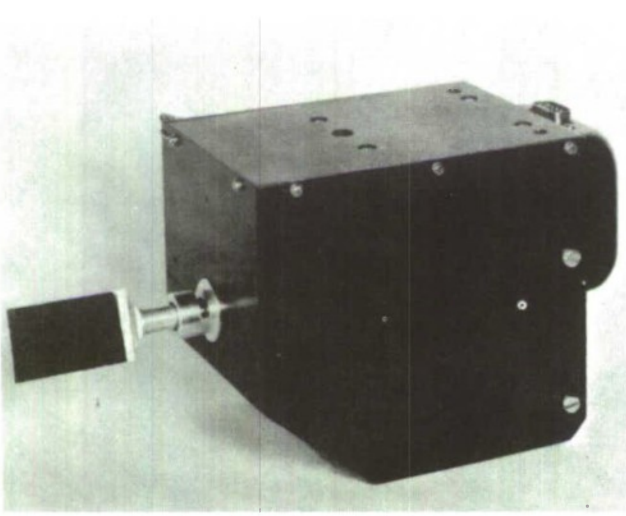


Figure 10—Vertical scanner

2. *Resonant magnetostrictive scanner*

Various problems, the most significant being lifetime, were encountered with the piezoelectric resonant scanner at higher repetition frequencies, *i.e.*, 28.35 KHz. Thus a research program was initiated to explore alternate techniques for providing a high resolution deflector. This work was performed by the Texas Instruments Co., under RADC contract AF30(602)-3731. The magnetostrictive vibrator appears to be highly suited for this application. A photograph of a typical magnetostrictive scanner is shown in Figure 11. The basic operation is similar to that of the piezoelectric scanner in that it provides a resonant circular scan. The mode of operation is the torsional mode, requiring two independent scanners to generate the circular lissajous figures. The magnetostrictive vibration is amplified by an amplitude transformer which can be exponential, fourier or stepped (Crawford, A., 1955). Typical deflections of 18° have been obtained at 30KHz with a reflective mirror having a diameter of .38 cm. This particular scanner appears most promising for obtaining a high resolution laser display of the raster type.

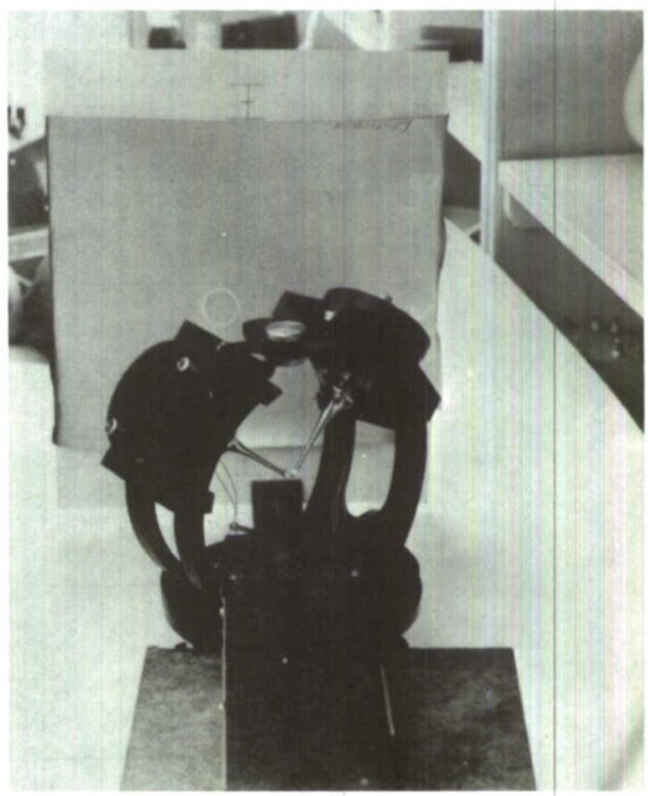


Figure 11—Magnetostrictive scanner

D. Experimental laser display, monochromatic model

An experimental model of a laser display was constructed to demonstrate the feasibility of this type display. This model was designed and fabricated by the Texas Instrument Co., under RADC contract AF-30(602)3271. The laser source used in the model was a 50 milliwatt He-Ne laser operating at a wavelength of .6328 microns. A photograph of the model is shown in Figure 12. The unit was designed to accept either of KTN having the dimensions d = 2mm and $l = 10$ an r-f television signal (off-the-air broadcast) or a 1.5 volt composite video closed circuit signal.

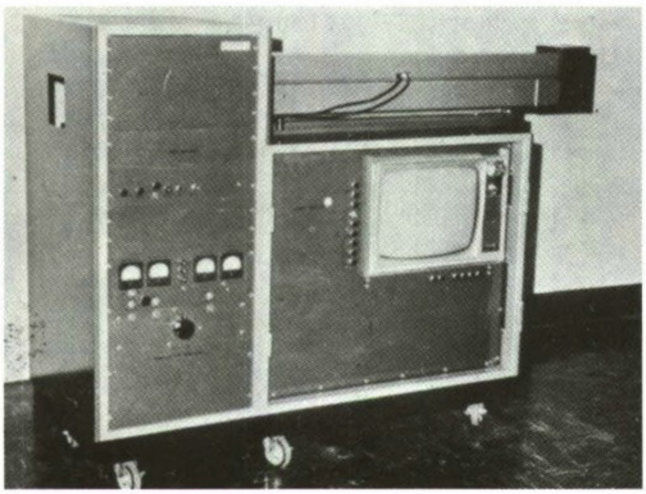


Figure 12—Laser display model, monochromatic

A functional block diagram of the model is shown in Figure 13 and the relationship of the various optical components in Figure 14.

The uniphase wavefront from the laser is amplitude modulated by a potassium dihydrogen phosphate (KDP) electro-optic modulator of the transverse type previously described. The bandwidth of the electro-optic modulator was 5MHz, sufficient to achieve normal broadcast television resolution.

The modulated light beam is scanned into a 525 line television raster by a piezoelectric resonant scanner deflection scheme. A 5½ inch focal length, f/2 aperture projection lens is used to produce a 40 inch square image with a 25 foot projection distance. A photograph of the image thus generated, is shown in Figure 15. The quality of this image is affected by numerous factors, of which alignment of the subcomponents in the optical train is a major factor. The performance which was achieved is described in Table V.

E. Multicolor laser display model

The ultimate objective in the laser display program is to develop the capability for generating multicolor images. In military command and control displays of the alphanumeric variety, the desired colors are red, green, blue, magenta, cyan, yellow and white. The particular attractiveness of the laser display from a color

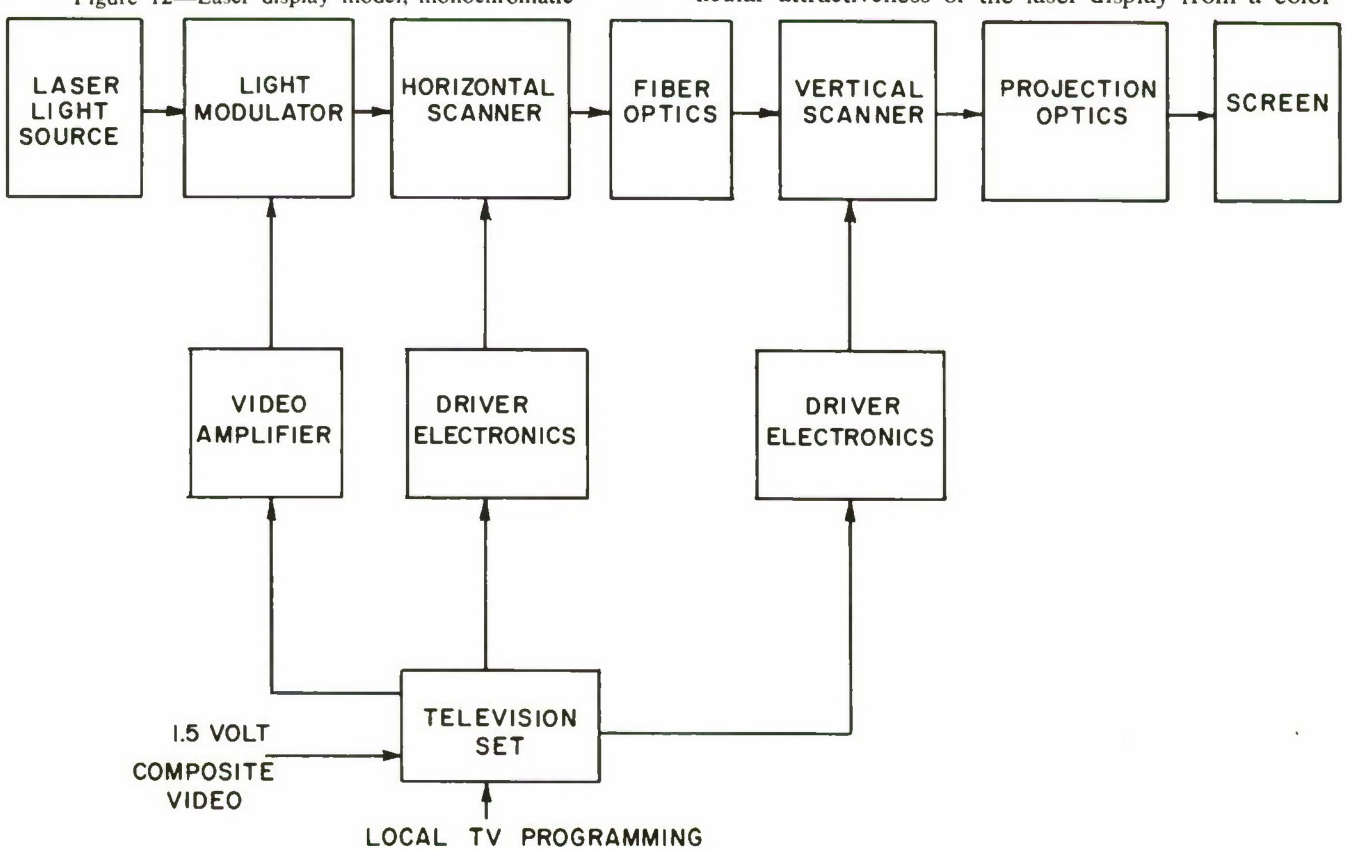


Figure 13—Functional block diagram, laser display

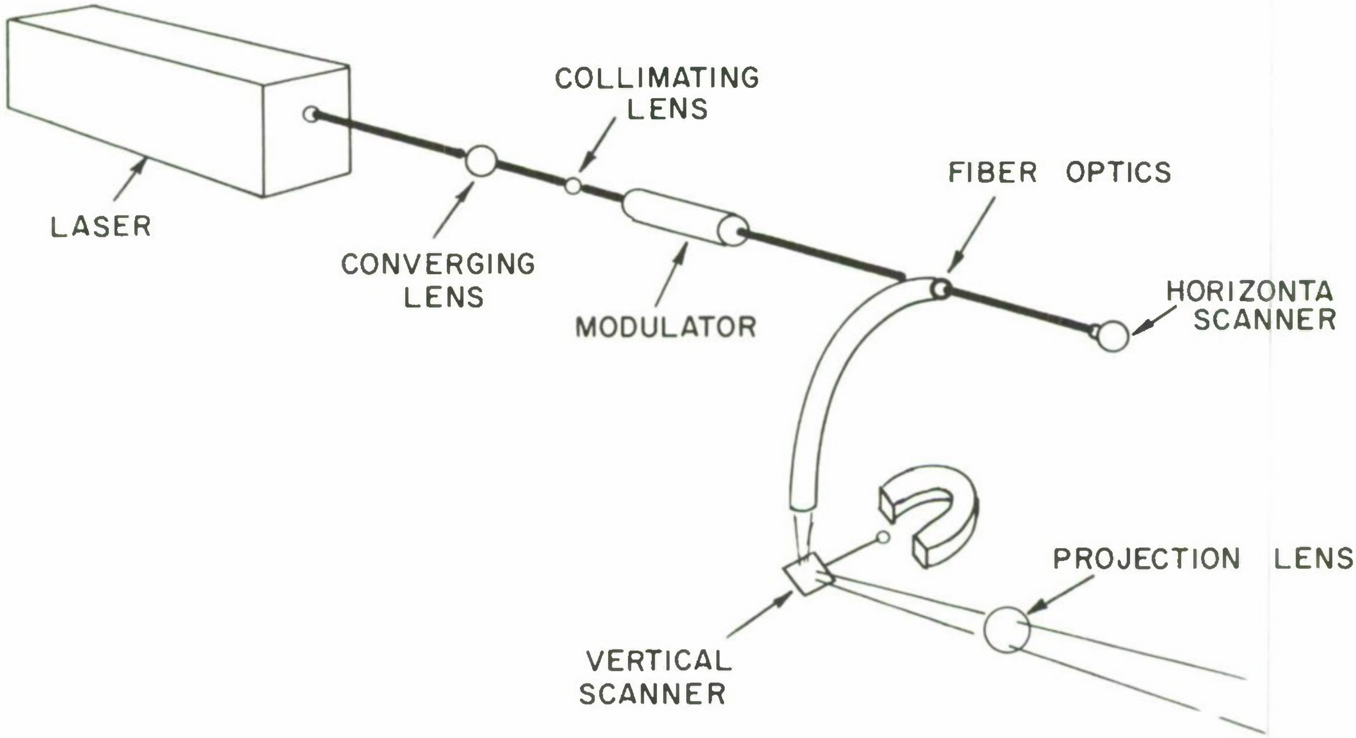


Figure 14—Laser display optical train

Table V—Laser display model (monochromatic) performance

Color	Red and Black
Resolution	Equivalent to 525 line TV system
Contrast	45:1 (measured in dark environment
Image Size	40″ × 40″ at 25 ft. projection distance
Brightness	3 ft-lamberts with a highly directional screen (gain = 32)
Transmission Efficiency	15% (major loss in fiber optic scan converter)
Laser Power	50 mw @ .6328 micron

point of view is that the color mixing can be accomplished prior to the deflector unit, thus precluding misregistration at the screen. Thus an effort was initiated to look at the possibility of combining the outputs of several lasers to generate a multicolor image.*

The optical layout of the multicolor model being fabricated is shown in Figure 16. The three primary spectral components are derived as follows: the red from a He-Ne laser operating at .6328 micron and the blue and green from an Argon laser operating at .4880 micron and .5145 micron respectively. The three beams are then individually modulated and combined by means of a mixing prism. The combined beam then varies in hue, saturation and intensity in correspondence to the modulation input to the three light modulators. The combined beam is then scanned and projected in the same manner as accomplished in the 525 line monochromatic model.

At this time, work is continuing in the fabrication of the deliverable unit. A photograph of the unit is shown in Figure 17. A commercial color television receiver is used as the source of the video signals that modulate the three light beams and can be seen in the photograph. At this early stage of development some preliminary multicolor images have been generated; however, meaningful photos are not available at this time. The completed unit was due for delivery at RADC in October 1966. It is emphasized that this model is developed for the sole purpose of looking at the problems associated with color generation using the laser display concept and is not for field use.

F. Laser display system considerations

Previous discussion has centered upon the principles and characteristics of the components required to generate a coherent light image. It is of interest to determine areas of uniqueness of this image and their effects upon the observer. At this time, the only area of major difference which is seen is that of the granularity associated with a coherent light image.

This phenomenon was first observed in the scattered light from a He-Ne gas laser emitting in the visible

*This work is being performed by the Texas Instruments Co., under RADC contract AF30(602)-3910.

Figure 15—Laser display image, monochromatic

portion of the spectrum at .6328 microns. Various explanations have been given, all in good general agreement (Rigden, J. D., 1962). A scintillation associated with the granular pattern due to relative motion between the reflective surface and the observer has given rise to the term "sparkle" associated with this phenomenon. The basic observed characteristics of this occurrence are:

a. An appearance of bright and dark areas in the laser light which is scattered from a matte surface. The size of the granules (areas) are a function of the viewing distance and angular aperture of the optical system.

b. A motion of the pattern as a function of observer motion, the direction of the motion differing for different observers.

c. A disappearance of the granular pattern by random motion of the screen or use of a colloidal solution, *i.e.*, milk, as the reflecting element.

The explanation of this occurrence lies in the fact that the coherent light scattered by the diffused surface forms a complex diffraction pattern. The granularity arises from interference of the diffraction pattern. The scintillation arises from the random motion of the observer, or his inability to remain perfectly stationary. The motion of the pattern with observer motion is a result of parallax. The differences of the direction of motion observed by different observers is a function of the focus plane of the eye, or for example, the nearsightedness or farsightedness of the observer. The granularity is not observed when the laser light is reflected from a colloidal solution, *i.e.*, milk, due to the large Brownian motion of the molecules in solution. This effectively causes an averaging out of the interference effect as for an ordinary incoherent light source.

This occurrence has also been observed with a dif-

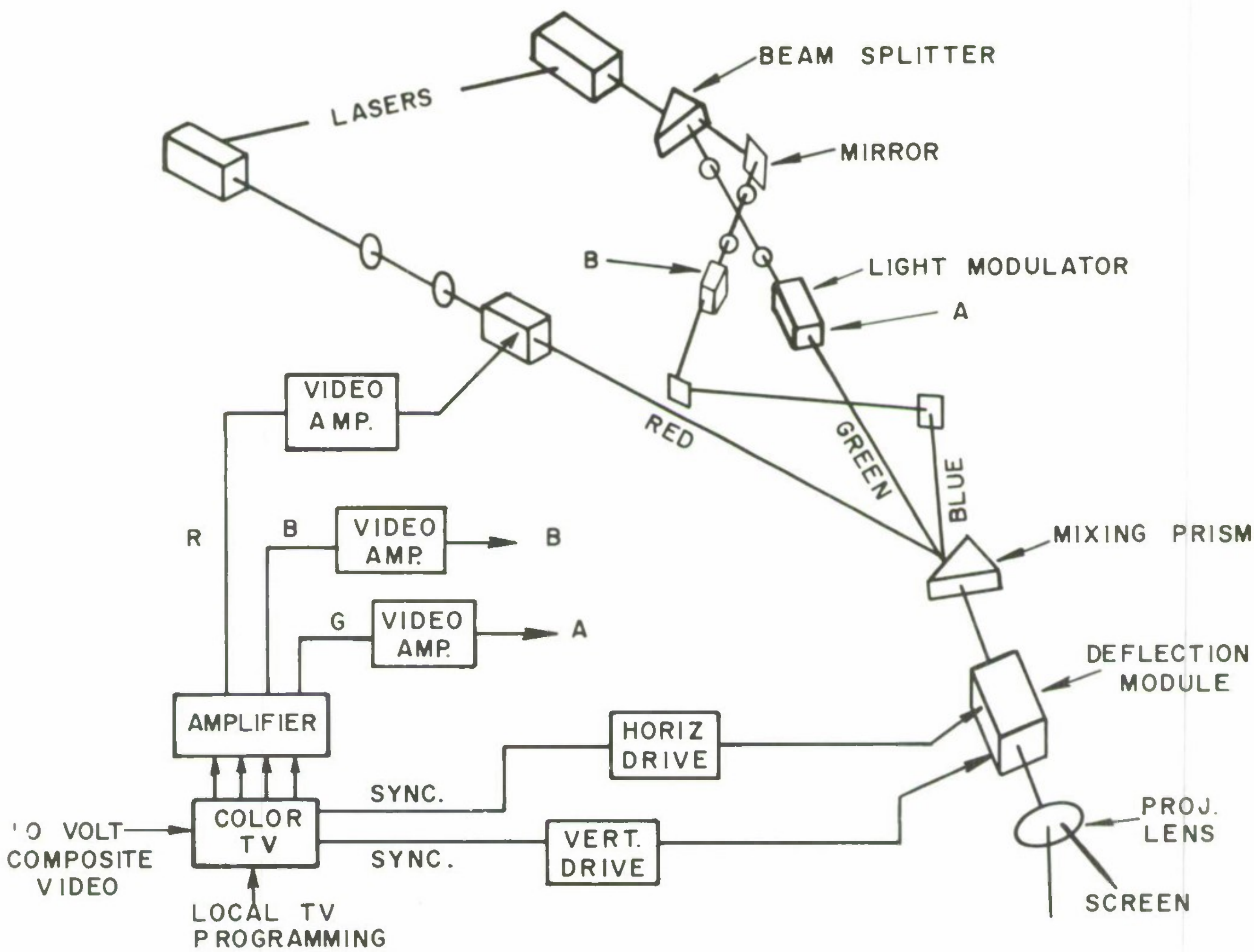


Figure 16—Optical layout, multicolor model

Figure 17—Laser display model, multicolor unit

fraction limited beam of unfiltered sunlight and for light from a high pressure Hg arc, supporting the hypothesis that it is a function of the spatial coherence of the source rather than its spectral purity.

The occurrence or nonoccurrence of this phenomenon in the laser display image is a function of the method by which the image is generated. Observations have been made for several different types of deflection schemes. A two-dimensional image was first generated using two piezoelectric driven mirror scanners to produce a lissajous pattern. A granularity of the image was observed. Also, a laser television system was built (described in the previous section) and a 2-D pictorial image was thus generated. The image did not have the characteristic granularity. In the latter deflection techniques, a fiber optic scan converter is used following the horizontal scanner to convert a circular scan to a linear scan. It is this element which effectively destroys the spatial coherence of the laser beam and in effect the sparkle pattern. A laser television image generated by a deflection scheme consisting of a nonmechanical technique for the horizontal scan and a vertically oscillating mirror to produce the vertical scan was reported to exhibit the granularity. Thus the only conclusions which can be made is that this occurrence will depend upon the extent of the diffraction limit of the beam which is imaged on the reflecting surface, and that methods are available for negating the effect.

A limited subjective impression can be stated for the effect of this granularity on the observer. Many observers have indicated that their ability to concentrate on the image which possesses this "sparkle" is considerably impaired. It is probable that it could have the same effect as a flickering image on the observer. However, qualitative conclusions can only be drawn after extensive experimentation has been performed.

SUMMARY

The present work under way at RADC in the laser display area has been described. It can be seen that the work is experimental and has attempted to obtain a good understanding of the nature of the laser display. Recent developments in the laser area indicate that sufficient power will be available shortly to generate a large screen, high brightness laser display. However, the life and reliability of such a display requires more experience with the major components to be used. Laser efficiency is poor at the present time and has the effect of decreasing the lifetime of the associated plasma tube. The effect of the high power (of the order of 5 watts) on the optical components would have to be determined.

At any rate, it appears that one can predict a set of characteristics which could possibly be realized in the not-too-distant future. These are shown in Table VI. It must be kept in mind that numerous trade-offs are possible amongst the many parameters and values listed.

Table VI—Laser Display Goals

Image Size	Up to 100 sq. ft.
Color	7 color capability: red, green, blue, yellow, cyan, magenta and white.
Brightness	50 fl-lamberts
Screen Gain	Will be governed by specific system constraints. Screen gains higher than 3 compared to a perfect lambertian diffuser probably will not be useful.
Linearity Requirements	Tactical Display—accuracy of 1 part per 1000. Pictorial Display—Deviation from perfect linearity can be as high as 5%.
Update Time	33 milliseconds
Resolution	Raster Display: 945 vertical lines 1000 horizontal lines Random Access Display: 1024 × 1024 resolvable elements.
Symbology	64 symbol capacity with rapid change capability.
Contrast Ratio	100 : 1 as measured in a dark environment.

Light valve displays

A. Eidophor large screen television projector

The only commercially available light valve today is the Eidophor Large Screen Television Projector, a product of Gretag Ltd., Regensdorf, Switzerland. It is available in several line standards, in black and white, sequential or simultaneous color models. The unit described here is the 1,000 line simulataneous color projector. The basic operating principles of the device and certain characteristics will also apply to other Eidophor models.

Figure 18 is a conceptual drawing of the Eidophor Light Valve. The 2,500 watt xenon light source is focussed onto a cold mirror for infrared removal and uniformly illuminates the picture window. The image of the picture window is projected onto the spherical mirror via the Schlieren bar system. For simplicity only three of the actual six horizontal Schlieren bars are shown. The bar system and spherical mirrors are so positioned that light incident on the spherical mirror is reflected to the bar system and back to the light source.

To illuminate the screen, part of the light returning to the bar system from the spherical mirror must be slightly deflected to pass between the bars. The media employed for this purpose is a 0.1 mm thick, transparent, viscous oil film spread evenly over the spherical mirror. Provided that the control layer is smooth, the light from the mirror is undeflected and the screen remains dark. If the oil surface is deformed at a point, a portion of the light illuminating that point is deflected and passes between the bar system producing a corresponding point of light in the dark field at the screen. The amount of light that reaches the screen is proportional to the slope and depth of the oil deformation (see detail upper right, Figure 18).

The means used to deform the oil are electrostatic forces. The metallized surface of the spherical mirror and the surface of the oil layer can be visualized as a capacitor. When electric charges are placed on the oil surface, attractive forces appear between the two surfaces. The oil is basically a dielectric so a charge applied at a point will remain concentrated and exert localized pressure proportional to its charge density. An electron gun directly deposits charge on the oil layer.

The electron beam scans the oil in raster format. When the electron beam spot size is large enough so that adjacent lines on the oil layer touch each other, the charge will be evenly distributed over the entire frame. The oil in the frame area is subjected to constant pressure and remains smooth and the screen appears dark.

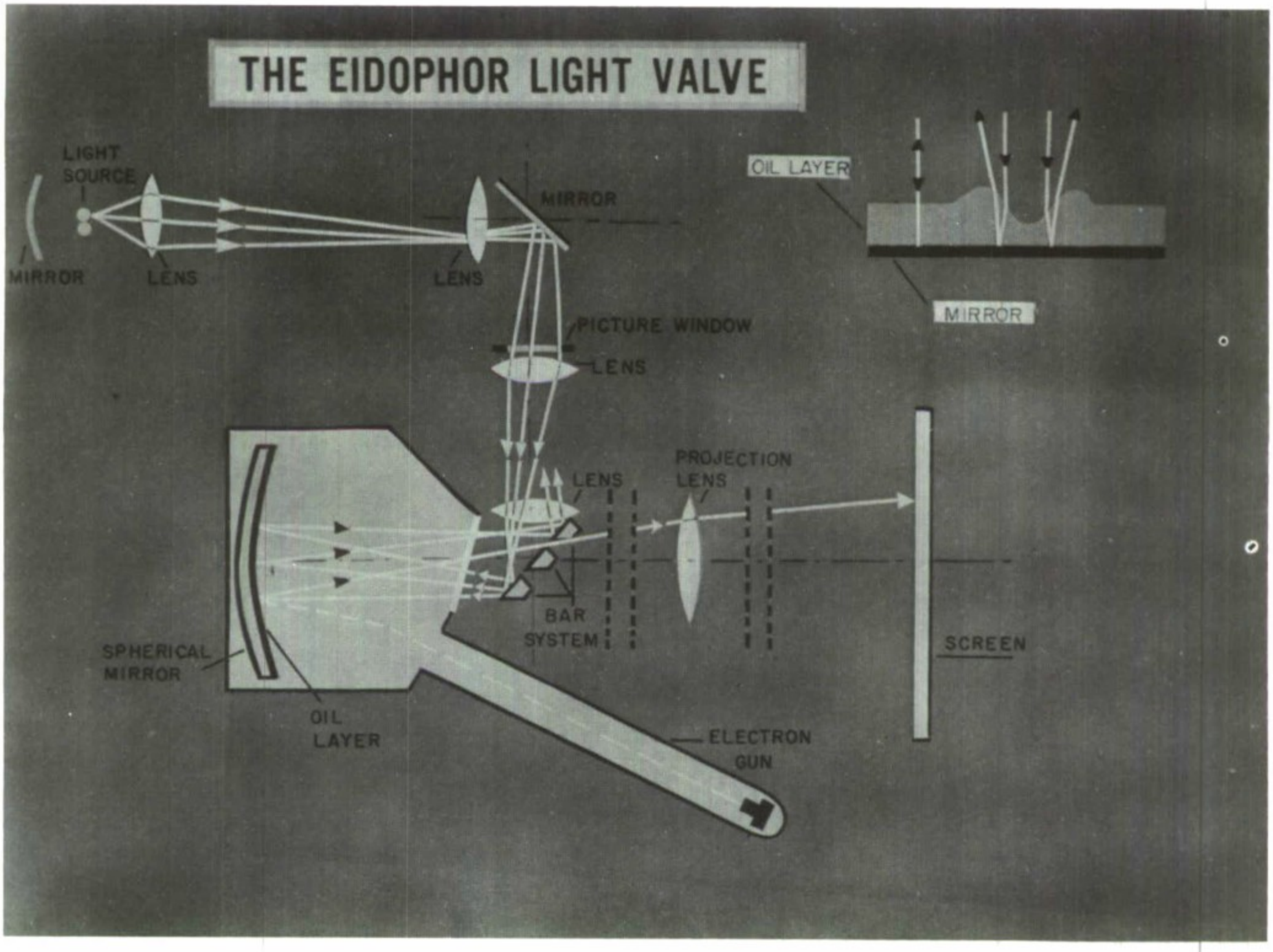


Figure 18—The eidophor light valve concept

Focusing the electron beam to a smaller size, so that adjacent lines do not touch, establishes a non-uniform charge distribution. The pressure distributions will likewise be non-uniform and will cause deformation in the oil layer. By modulating the electron beam focus with a video signal and scanning the electron beam, a video image can be produced at the screen.

A television frame is written every 1/30th second. If one raster is not to impair the succeeding one, the oil layer must be smoothed again before the new frame is begun. Two things are required for this to occur: the deposited charge must be removed and the surface tension must remove the deformation. Conductive additives are included in the oil to facilitate charge removal in 1/50th second. The time required for the surface tension to smooth the deformations depends upon the viscosity of the oil. The proper viscosity is obtained by controlling the temperature of the oil (Gretag Ltd., undated).

To obtain full color, the Simultaneous Color Projector utilizes dichroic filters to break up the light from the light source into the three primary colors: red, green and blue. These individual light beams are then directed to essentially identical cassettes, as described above. Each electron gun is provided with the proper video modulation for its color. The images are projected to the screen where they are super-imposed in a full color picture.

The present capabilities of the Eidophor are tabulated below:

- Resolution: up to 1000 lines
- Luminous Flux: 4000 lumens
- Uniformity of Field: Approximately 30% fall-off center to edge
- Contrast Ratio: 100:1
- Linearity: 2%
- Optical Efficiency: 5%

The Eidophor is a complete projector requiring only power and standard TV signal inputs. All electronics, optics, correction circuitry, etc. are included in the projector.

For command and control military applications the Eidophor as presently manufactured has certain undesirable features. The fact that raster scanning is employed to help smooth the oil layer means that computer generated data must first be converted by complex buffering equipment before the data is in proper form for input to the Eidophor. Random scan is the more desirable mode of operation for computer driven displays. RADC is currently investigating the possibility of using the Eidophor in a random scan mode. Initial experiments indicate that it will be possible to employ the Eidophor in random scan. Skew line segments have been generated at various positions and angles using a standard black and white Eidophor. None of the expected "sag" in the oil film was noticed although the smoothing effect of the raster scan was absent. Modification to the Schlieren system will probably be necessary to make the system equally sensitive to horizontal or vertical lines. If this study is successful the need for complex scan conversion and digital-to-video buffering equipment will be eliminated.

A second problem area of the commercial Eidophor for military application is in the area of reliability. For military field use it is desirable to have minimum maintenance requirements, repair times and maximum operation time between maintenance and failures. Since the Eidophor contains many mechanically moving parts its reliability is theoretically lower than a comparable nonmechanical projector, *e.g.*, one that is fully electronic. Also the Eidophor cassette contains both the electron gun and the oil film control layer. This contributes to the need for continuous vacuum pumping and to short lifetimes of the electron gun cathodes. To obtain minimum down-time a turret of three cathode assemblies is contained in each gun assembly. As a cathode fails a new one can be rotated in place, under vacuum, in a few minutes. This is, at best, a temporary solution to the problem from a military users standpoint. Improvement in cathode life is required if the Eidophor is to be widely employed in field use.

Problems of chemical deterioration of the oil under electron beam bombardment is minimized by renewing the oil film from a filtered supply as the mirror rotates at one revolution every 2½ minutes. Polymerization of the oil does occur, however, and periodic replacement of the oil is required.

B. Pin matrix oil film light valve

While RADC continues to explore ways of improving available display projectors, it also seeks new approaches in display technology. Conceivably these new techniques may provide all the present capabilities of light valve projectors and additionally provide those desired characteristics that current devices lack. The pin matrix oil film light valve is one such technique. The feasibility of this device to perform as a highly reliable random access light valve with expected characteristics similar to the Eidophor, has been demonstrated. The technique utilizes the principle of deformation of a smooth oil film to produce an optical image. However, the Schlieren optical system and the method employed to deform the oil layer differ from the Eidophor.

The pin matrix light valve consists of a permanently sealed cathode ray tube incorporating a special wire mosaic (pin matrix) faceplate. Figure 19 is a conceptual drawing illustrating this technique. Detail of the pin matrix faceplate is shown in Figure 20. The control layer is illuminated through an aperture of a 45° mirror. If the oil film is smooth the dielectric mirror beneath the oil reflects the incident light back through the aperture to the light source. If a deformation is introduced in the layer, the incident light is refracted resulting in a disturbed ray of light which strikes the 45° mirror. An image of the deformation then appears on the screen.

To produce a display, deformations are introduced on a point by point basis and are accomplished by means of electrostatic fields. The fields are established between each of the isolated pins of the faceplate and the external electrode structure. A negative potential is provided to the external electrode by a power supply. The pins are charged positively to establish an electrostatic field around each pin. The flood guns cover the entire mosaic with charge. Depending on the initial pin voltage and the energy of the flood gun electrons, each pin charges to one of two stable potentials. By choosing the proper secondary emitter and using a secondary collector, the write gun can be used to switch each individual pin to the desired stable state. Variable persistence and storage are possible when using the tube in this mode.

Another flood gun energy results in the pins having only one stable state. The write gun charges the pins to some unstable potential and the flood gun returns them to the stable state. This mode would be used for dynamic displays where no storage of the image is desired.

In either case the display can be randomly addressed since the oil film is smoothed by surface tension only. There is no direct deposition of high energy electrons onto the oil layer, hence no chemical deterioration of the oil is expected.

Reliability of a projector utilizing this technique should be high. This is based on the fact that the tube is essentially a standard, permanently sealed cathode

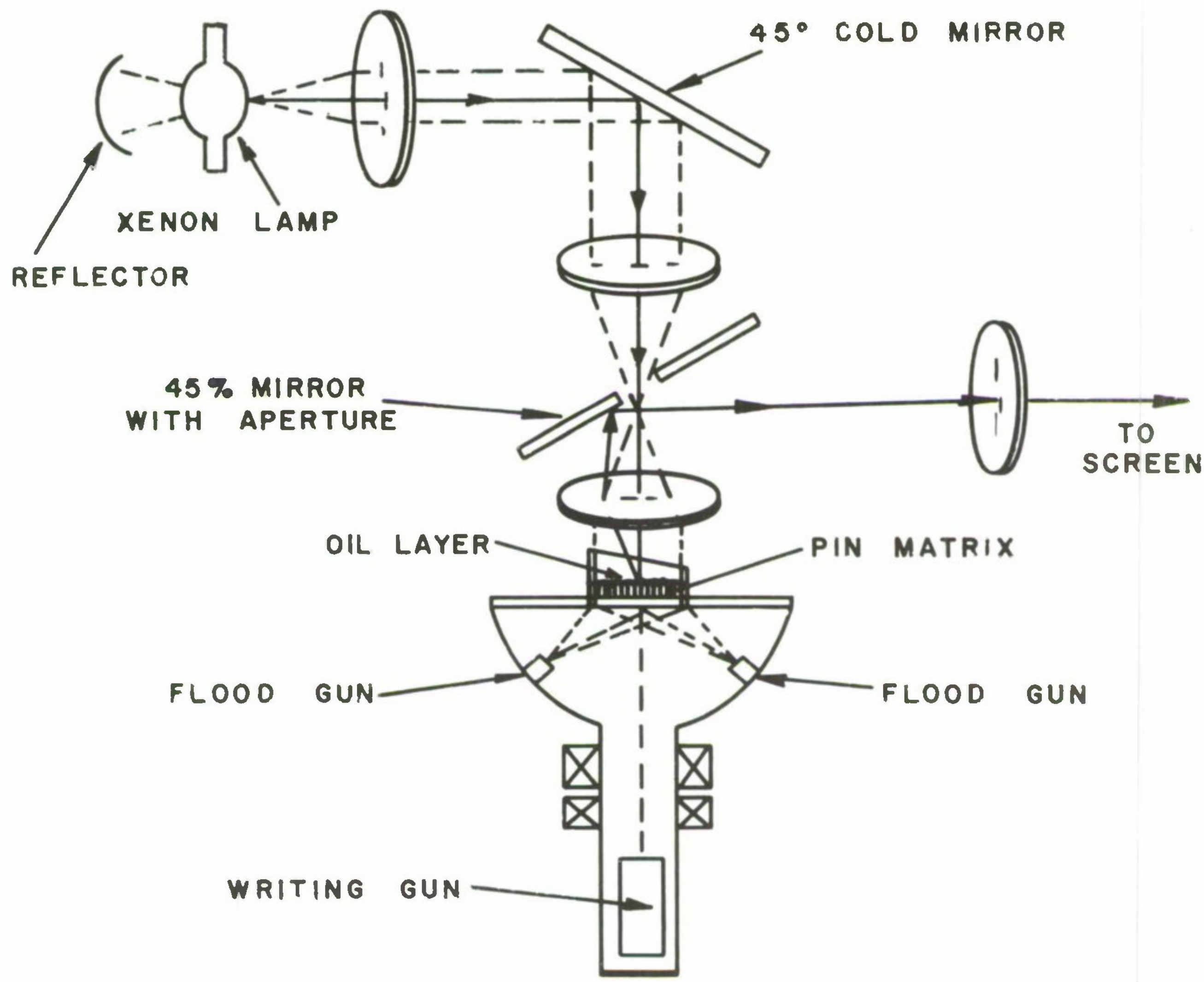


Figure 19—Pin matrix light valve concept

ray tube differing only in the faceplate construction. Since the control layer is external to the vacuum, this source of cathode contamination is eliminated. Also the system is purely electronic which leads to higher reliability since no moving parts are required for operation.

During the first year's study, the contractor* demonstrated feasibility of using this type tube to produce discrete deformations of the oil control layer. The resolution of the write gun used in the breadboard tube was too low to charge an individual pin, however, individual pin images do appear on the screen indicating single pin resolution is possible. Variable persistence has also been experimentally verified. Interpin capacitance is a factor determining how fast the write beam can charge an individual pin. The data rate capability of the tube therefore depends upon the choice of pin size, spacing and the dielectric constant of the insulating glass substrate. Further study is required to determine the optimum matrix geometry and construction.

The optical illumination system used for the breadboard model was adapted from another projector. For this reason contrast and efficiency measurements were not as high as theoretically predicted. Also contributing to degradation of optical quality of the display were scratches in the pin matrix faceplate mirror surface and difficulty in establishing a completely smooth dust free oil layer.

The pin matrix faceplate, specially designed and developed for this technique, was 1 inch in diameter with individual pins of 2 mils diameter spaced 4 mils on center. The difficulty of achieving proper pin size and spacing while maintaining vacuum integrity between the pins and the glass cladding structure is no longer a

*Air Force Contract AF30(602)-3646, Random Access Light Valve Study, Stromberg Carlson Corporation, 1966.

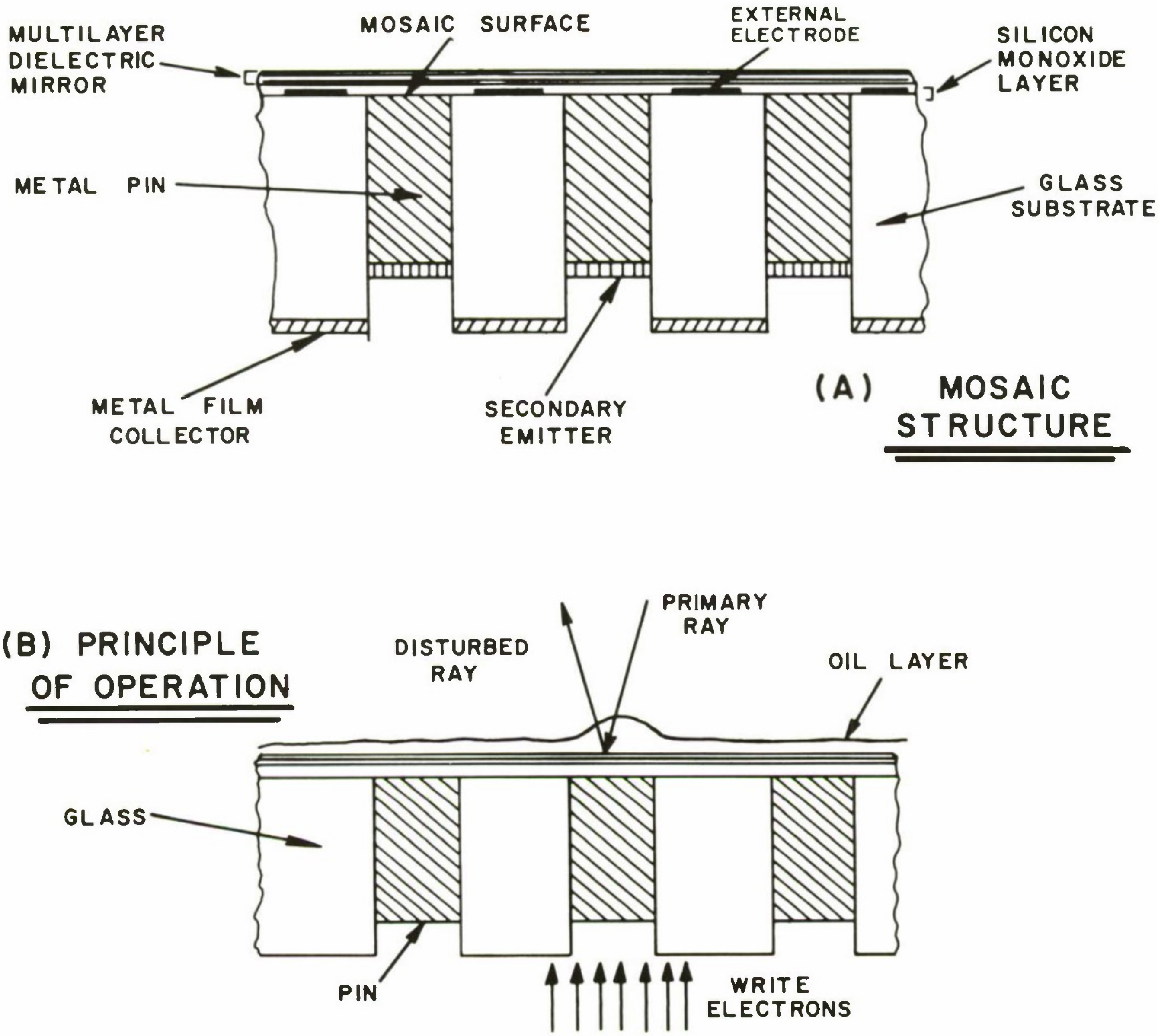


Figure 20—Pin matrix detail

major problem. Present contractual studies** anticipate use of 2 inch matrices.

The contractor has developed techniques for polishing the mixed glass-metal faceplate and for depositing of electrodes, secondary emitters, mirrors and protective coatings.

Areas of concern in the present study are: improvement of the quality of large mosaics, and improvement of mirror and buffer coatings. Optimization of the electron optics to obtain single pin excitation and further study of the secondary emitter requirements are also being pursued (Hamann, O. F. 1966).

**Air Force Contract AF30(602)-4286 Pin Matrix Light Valve Study, Stromberg Carlson, 1966 (in progress).

C. Solid state light valve

The two previously discussed light valves employ oil film control layers and Schlieren optics to produce large screen projection displays. The solid state light valve utilizes an electro-optic crystal and polariscope optics to accomplish this function.

The original concept of using electro-optic crystals in a light valve application dates back to 1934 when M. Von Ardenne invented the Ardenne Tube for the German Post Office. For a variety of reasons, the technique was not extensively explored for display application until recent years. In 1961, RADC took an active interest in the Solid State Light Valve and has since pursued it with in-the-house and contractual efforts (See footnotes 1 - 5).

The goals of these studies are directed toward a sealed electro-optic projector tube capable of achieving a practical real-time display. A sealed tube was fabricated and demonstrated projection of low resolution TV images at low light levels. The results of these studies have indicated the basic feasibility of this technique, however, further improvements are required before a practical device is realized.

In concept, the polariscope optics used in this light valve consists of two plane polarizers whose preferred axes are oriented orthogonally. Light incident on the pair is plane polarized on passing through the first polarizer and is stopped upon reaching the second polarizer (analyzer). If one inserts between the polarizers of the polariscope an optically active element then effectively the plane of the polarized light is rotated and a component of the light passes the analyzer.

In the display scheme envisioned, an electro-optic crystal is placed in a collimated light beam between the polarizer and analyzer (Fig. 21). With no electrical charge applied to the crystal, it is optically inactive and the light is unaffected by its presence. Therefore, no light passes the analyzer. When an electric field is applied across the faces of the crystal in a direction parallel to the light beam, changes occur within the crystal making it optically active. The plane polarized light is changed to elliptical polarization, thus a component of light can pass the analyzer. This effect in electro-optic crystals is known as the Pockels' or Linear Electro-optic Effect. The described concept represents an optical shutter whereby the light that passes the analyzer depends on the bias voltage across the crystal.

By using an electron beam to produce localized

1 Solid State Light Valve Study 1964.[7]
2 Air Force Contract AF30(602)-2645 Solid State Beam Controlled Light Modulator, Motorola Corp., 1963.
3 Solid State Light Valve Study 1965.[6]
4 Air Force Contract AF30 (602)-3263, Autonetics Corp. 1965.[16]
5 Air Force Contract AF30(602-3730, Autonetics Corp. 1966, Final Report not yet published.

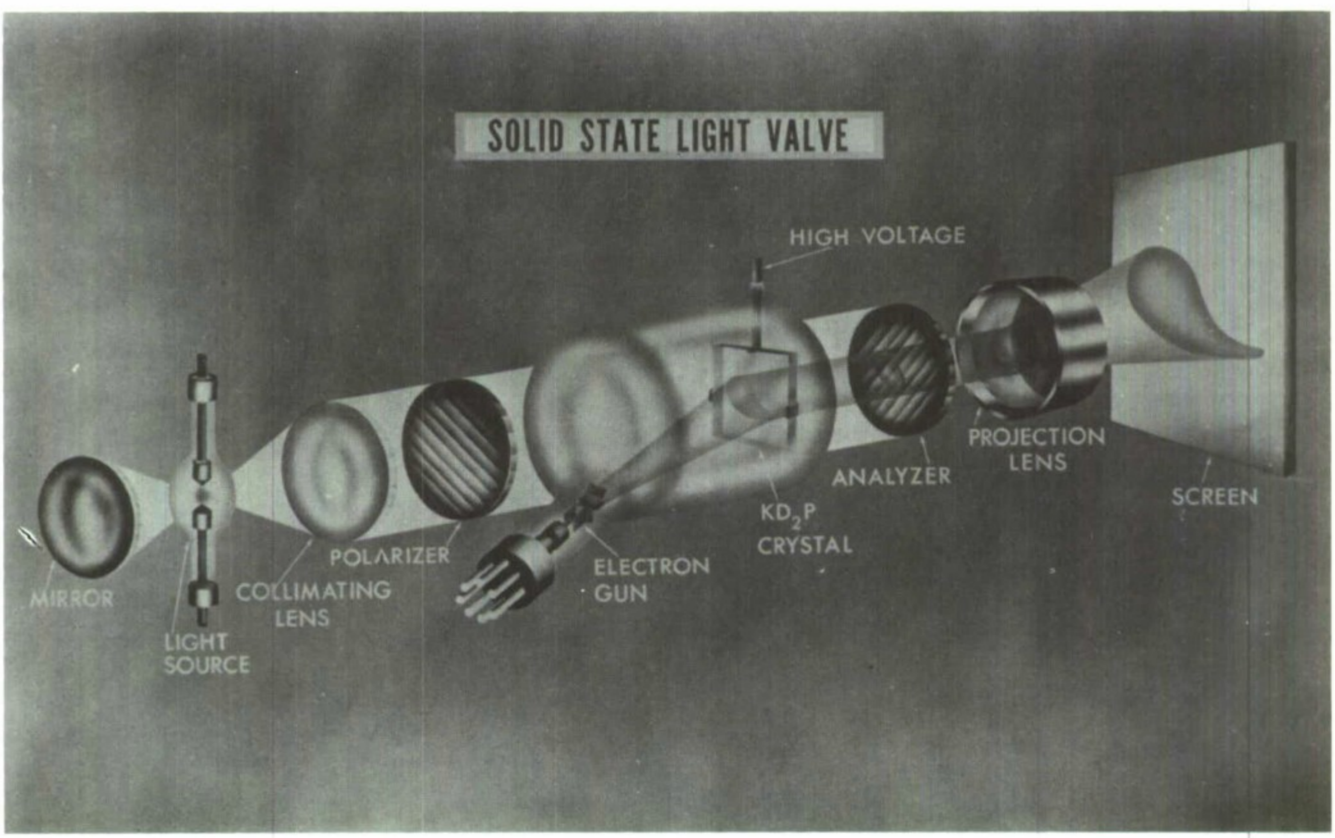


Figure 21—Solid state light valve concept

fields, the effect becomes usable as a display device. The beam is scanned and modulated to deposit an image charge pattern with a one to one correspondence to the desired optical display. This results in a point by point shuttering of the light in proportion to the applied charge and an image is produced at the screen. It can be seen that the technique can be applied in either raster or random scan modes.

Figure 21 depicts a transmissive mode of operation of the solid state light valve. This mode of operation requires a voltage across the KD_2P (Potassium Dideuterium Phosphate) crystal of about 3 KV to produce maximum brightness. In addition, it requires an optically transparent conductive coating on the rear face of the crystal for a ground plane and signal return. This mode of operation is presently being considered in experimentation. However, recently acquired data indicates sensitivity of the electro-optic effect to changes in temperature. This may present a technical problem using this configuration in conjunction with higher light flux illumination. Despite lack of optimization, a tube of this type demonstrated 150 - 200 lines/in. resolution and 100:1 contrast ratio utilizing a standard TV input signal and low intensity light flux.

An alternative mode of operation is the reflex configuration. This tube requires only half the voltage across the electro-optic crystal to produce the same brightness at the screen as the transmissive tube. Here the transparent conductor can be replaced with a metallic mirror and special reflex optics introduced to effect projection of the desired image. Additionally the reflex tube can be designed for ease of temperature control of the crystal if temperature is discovered to be a critical parameter of the device.

Electro-optic crystals used in this technique must be single crystalline, optically clear, strain free and electro-optically uniform. For application these crystals must be cut into Z-0° plates and optically polished to thicknesses of approximately 0.010 inches. Large cross sections (approximately 2″ × 2″) are needed in order to achieve high resolution images. Crystals currently available do not possess all the aforementioned desired qualities. Non-uniformities in electro-optic response, strains and small cross sections are of major concern for practical application. Of the available materials examined, KDP (Potassium Dihydrogen Phosphate) and KD_2P (Potassium Dideuterium Phosphate) are best suited for this technique. RADC plans to study methods of growing better quality KDP and KD_2P crystals of large cross section. Also of interest are new materials which show an improvement over KDP and KD_2P particularly in the voltage required to produce maximum light intensity. (Half-wave retardation voltage.) If this voltage is decreased, the electron-optics and resolution problems become less difficult. One possible crystal improvement is the deuteration of KDA (Potassium Dihydrogen Arsenate). If this follows the pattern set by other deuterated crystals, its half-wave retardation will be less than that of KD_2P.

Before a practical solid state light valve evolves further studies are required. These studies include temperature effects, the seriousness of the crystal outgassing problem, and the display parameters achievable in the optimized tube design. Development of new processing techniques are needed to polish crystals to higher optical tolerances in thicknesses less than 0.010″. Further investigation of sealant-protective coatings is required. Also a means must be found to reliably deposit the conductive electrode on the crystal, since experiments have indicated poor adhesion of the electrodes and changes in their conductivity after a period of tube operation.

CONCLUSION

The primary purpose of this paper has been to describe several mechanisms which have application to the generation of large screen displays. It is obvious that various ancillary equipments are required, in addition to the display generation device, to complete the makeup of a display system. The reason for making this obvious point is to emphasize that in the final analysis of the display system configuration, the most important consideration may very well be governed by availability, cost and performance of this ancillary equipment. For example, some of the work discussed has centered on a sequential (raster scan) mode of operation. Such a system will require digital to video conversion to mate the display device to the computer input. A more practical approach might be to consider selective address (random) mechanisms in the display device, which mate directly with the computer input. Research is presently underway in this area.

Finally, the schemes described and models fabricated are presently categorized as being research devices (with the exception of the Eidophor). A considerable amount of refinement will be required in order that existing reliability and maintainability specifications can be met for field applications.

REFERENCES

1 C. E. BAKER and H. PARKER *Laser display study* U. S. Government Research Report RADC-TDR 65-169 1965

2 B. H. BILLINGS *The electro-optic effect in uniaxial crystals of the type XH_2PO_4* I. Theoretical, JOSA Vol. 39, No. 10 797-801 (a) 1949

3 B. H. BILLINGS *The electro-optic effect in uniaxial crystals of the type XH_2PO_4*. II. Experimental, JOSA Vol. 39, No. 10 802-808 (b) 1949

4 B. H. BILLINGS *The electro-optic effect in uniaxial crystals of the dihydrogen phosphate (XH_2PO_4) type.* IV. Angular Field of the Electro-optic Shutter, JOSA Vol. 42, No. 1 12-20 1952

5 M. BORN and E. WOLF Principles of optics New York, London Pergamon Press pp. 397 1959

6 E. J. CALUCCI *Solid state light valve study* Information Display Mar/Apr 1965

7 E. J. CALUCCI B. C. GANZLER, and A. C. CLOUGH *Solid state light valve study* U. S. Government Research Report RADC-TDR 64-192 1964

8 R. O. CARPENTER *The electro-optic effect in uniaxial crystals of the dihydrogen type.* III. Measurement of Coefficients, JOSA Vol. 40, No. 4 225-229 1950

9 A. CRAWFORD Ultrasonic engineering. London Butterworths Scientific Publications 1955

10 J. E. GEUSIC *Nonlinear dielectric properties of $KTaO_3$ near its curie point.* Applied Physics Letters Vol. 2, No. 10 pp. 185 1963

11 J. E. GEUSIC *Electro-optic properties of some ABO_3 Perovskites in the paraelectric phase* Applied Physics Letters Vol. 4, No. 8 pp. 141 1964

12 Gretag, Ltd. *Eidophor large screen television* Commercial Manual undated.

13 O. F. HAMANN *Random access light valve study* Final Report, AF Contract AF30(602)-3646 RADC-TR 65-451 1966

14 R. E. HOPKINS *Some demonstration experiments in optics using a gas laser* American Journal of Physics Vol. 32 1964

15 J. D. RIGDEN and E. I. GORDON *The granularity of scattered optical maser light.* Proc. of IEEE Vol. 50, No. 111, 2367-8 1962

16 R. S. STITES, W. E. MEYER, C. L. BUDDECKE *Electro-optic projector study* Final Report, Air Force Contract AF30(602)-3263 RADC-TR 65-25 1965

17 J. L. WENZ *Novel laser Q switching mechanism* Proc. of IEEE Vol. 52, No. 6 pp. 716 1964

Military command information systems from the user's point of view

by JOSEPH G. CARLEY, JR., *Lt. Colonel, USAF*
Headquarters United States Air Force
Washington, D.C.

INTRODUCTION

When an automated command and control system reaches operational status, the term "USER" assumes a more specific meaning than the interpretation normally applied by the technical and development community. According to the latter interpretation, any command or agency which acquires and uses a system is referred to as a "USER."

Generally speaking, the Department of Defense concept for a command and control system encompasses the various elements that are required for the commander to control his forces and resources. This includes all of the associated facilities, people and equipment which, when tied together with procedures, form a total integrated system. Of equal importance are the functions necessary to make the system operate effectively. In a headquarters environment, broad participation by the functional staff is obviously essential.

The automated command and control system, of course, is but one element of the total command and control system of any agency, command or service. However, it is rapidly becoming a key element and therefore care must be taken to insure it is appropriately responsive to the needs of the staff and commander. These needs must be translated into facilities, equipment and software design. It is appropriate, then, for the people who are directly responsible for operating the system to look upon the staff and commander as the real "USERS" of the system. This introduces a special category of personnel and organizations which must have a strong voice in command and control matters: The SYSTEM OPERATORS.

Uniformally throughout the Air Force the staff operations officer is assigned the responsibility for managing and operating the command center complex. Since the computer system is an integral part of this complex, the operations officer is placed in the role of SYSTEM OPERATOR.

Although the use of the term "USER" in the title of this paper is generally correct, it is specifically from the perspective of the System Operator that the ideas and opinions are expressed in the following paragraphs.

The headquarters level system

Command control and communications systems can be divided into four major categories: Communications; surveillance; direct control (tactical and/or strategic control of airborne aircraft), and semi-automated fixed facility, headquarters level command center systems. The first three categories of systems are looked upon as special purpose in nature and usually function remotely from the staff officer. It is the final category which currently demands our attention, consumes much of our staff energy and convokes discussion from all quarters. We have, therefore, confined our discussion to this particular area.

The headquarters automated command and control system has been cloaked with a "romance of words" by technical writers, policymakers, system designers, system developers and a myriad of self-styled command and control experts. It is said that systems must be developed from an "incremental, evolutionary approach"; that they must be "compatible" with one another; that standardization must be achieved wherever possible, and that they must be "immediately and directly responsive" to the commander for whom the system is designed.

These words and phrases are found in policy letters, concepts of operations, and professional papers. The policy letters are intended to provide guidance to the commander contemplating the acquisition of an automated system and to the technical agency or contractor having design and developmental responsibilities. However, these particular words and phrases are largely undefined in the specific sense. The reader or listener applies his own interpretation as to their exact meaning depending on his background, parochial interests, organizational loyalties or position. The result is constructive controversy in the approach to command and control policy, management, development, acquisition, operation and utilization.

During recent years the din of command and control discussion has been raised a few decibels by a growing chorus of Air Force command and control system operators. The words of the policymakers have begun to crystalize into specific meanings to some people in this system operator group. As the definitions evolve, so also evolve concepts: Concepts for approaches to design; concepts for development; but primarily concepts for operation. The conceptual points of view of at least one operator are presented herein. The subject matter is examined first from the specific aspects of the development, acquisition, operation and use of a single system and then broadened to provide a perspective and an overview of Air Force automated (headquarters level) command and control systems as a TOTAL RESOURCE. The approach to discussion is made by way of providing a system operator's definition of some of the words and phrases previously referenced.

The incremental, evolutionary approach

There is complete agreement between policymakers, system operators, designers, developers and systems engineers that an automated command and control system, or any computer system, must be well documented in specification form before actual development begins. The most recent policy statement concerning design specification was published on 29 July 1966 by the Secretary of Defense in his defense memorandum, *Management and Use of the Electronic Computer,* Washington, D.C.: Department of Defense, pg. 3, who said:

> *"We must select and acquire new or replacement computers only after systems have been redesigned to make full use of the improved capabilities of later model hardware, and then only where there are proven cost benefits. In these cases, systems redesign and programming will be accomplished prior to delivery of any equipment. Further, computers will not be selected until the performance of the complete hardware/software package required in the systems specification and request for proposal has been clearly demonstrated by either a full-scale or benchmark test."*

Depending on the interpretation of the reader, the policy and sound engineering principle for fully documented systems could be in conflict with the policy of incremental, evolutionary development. Let us explore the possible meanings of the latter policy.

One interpretation could require the development of a series of systems, with each system in the series designed as a complete entity and with each system representing a controlled step upwards in power, capacity and complexity. Theoretically, the command center would eventually reach the panacea in automated capability.

A major advantage to this approach would be the learning curve to which the operators, staff users and technical support agencies would be exposed. Each system in the series would benefit by the design errors or deficiencies of its predecessors. However, such an approach would place the operator in the position of operating, maintaining and improving one system while designing and developing its replacement. His resources would be divided. Further, the period of overall development activity would be long, tedious and expensive.

A second interpretation is based on the idea of definition and specification from the "total system" approach. This approach requires the determination (to the best judgment of the system designers) of maximum future requirements for equipment in such areas as computing power, display, core and mass storage, speed and capacity. In the software area all elements, including both support and functional applications, must be defined and documented, designed and developed. Once completed, the system is turned over to the system operator at which point the incremental growth process begins.

This is considered the classical approach and the one most likely to be recommended by the system engineer. But from the operator's point of view, there are some important problems associated with the total system approach which must be examined.

In the first place, if a command and control system is to be developed in consonance with the Department of Defense command and control concept, the system must, by necessity, result in a complexity of interacting files and programs. These files and programs support multi-users working on problems and solutions which are influenced and even founded on information, decisions, factors and operational requirements derived from the various functional activities.

As a practical example, the support functions such as transportation, logistics, personnel and medical cannot begin the planning process for any contingency operation until the force requirements are specified. In turn, the force requirements cannot be determined until the threat is analyzed and both defensive and offensive weaponry are selected, the length of engagement forecast and probable attrition and expenditure rates are computed.

Once an operation gets under way the means must be provided which will permit the appropriate staffs at all levels to monitor current operations, to determine the validity of planning factors and to analyze and establish future requirements.

Obviously the basic automated system requirement calls for a planning oriented capability designed around an ability to develop accurate operational profiles. These profiles directly influence one another and at several levels of command must be combined into an overall profile of the total operational requirement.

This general description of the problem is "old hat" to the military staff officer. Throughout military history these planning functions have been performed with varying degrees of success and have been a strong influence in the results of war. Nor is it to be considered a profound announcement that "automation of operational profiles is within the technical state-of-the-art today." However, when the design goals of an automated system encompass this total operational spectrum, the designers are faced with an overall problem which may yet be too complex for assurance of a satisfactory product.

One of the major problems of system design is the translation of the ideas and experience of functional staff specialists into an understandable form for the systems designer. As the systems design grows in complexity and size, there is a comparable increase in the numbers of people required to cover the tasks involved. The determination of these individual functional requirements and the accomplishment of associated design activities must be performed under an overall and integrated management scheme. The resultant management task is best described as severe. Top management personnel find it extremely difficult to correlate all of the activities in such a manner that the original concept of the particular system design remains inviolate.

The period of time required to develop studies, prepare detailed specifications and begin actual development normally extends beyond the average tours of duty of the functional staff officers and even beyond the tenure of the management, analysis and programming staffs of the contractors and other technical supporting agencies. With the rotation of personnel new ideas and influences are introduced in the direction of systems development. Hence, continuity in developmental management and direction is periodically jeopardized. Further, the original concept is in constant danger of gradual erosion from many other causes, including: Budget cuts, technical problems, policy changes and, last but most important, changes in military missions and requirements.

The development of the Headquarters USAF Command and Control System was approached in a combination of alternatives one and two. An initial small scale "off-the-shelf" computer was placed in the command post and operated with a software system which served as a small model of the "total system" design. Even though the system was limited in capability, it was considered a success and was accepted as the basic design for the next step. The small scale system shouldered the staff support task and was treated as an operational system, during which time valuable training and operational experience was gained. The same system was also literally exported to five Air Force major commands with surprising success. (More about this later.)

Development of the second system was undertaken on the operational site in the USAF Command Post at Headquarters USAF and was subjected to all of the erosion processes referred to previously. The most significant influence came in the form of US military action in Southeast Asia.

The Headquarters USAF "total" system had been under development for several years when hostilities began and final realization of the design capability was still several years in the future. The smaller system was pressed into the role of supporting staff requirements brought on by Southeast Asia. "Inhouse" technical personnel who could be spared from the larger system development task were concentrated on improving the operational system. Unfortunately, only a small portion of the total "inhouse" force could be spared from the developmental activities without impairing production schedules.

It is important to note that neither the operational nor developmental system designs could accommodate the new requirements generated by Southeast Asia. It would have been difficult to forecast that modern war management would require the examination of such large volumes of raw operational data, or that the evaluation spectrum would venture into an infinity of requirements, or that they would be demanded with such rapidity from the highest levels of national authority. This does not mean, however, that the system designers could not have accommodated these requirements or that the functional specialists involved were shortsighted and failed to recognize the growth potential of an automated system. As a matter of fact, the requirements for detailed information at such high levels are largely disassociated from the general theory that "automation in itself creates a demand for detail." The unanticipated actual requirements, in the opinion of the author, have grown out of necessity because of the world political situation and the requirements for "tight" management and responsiveness in supporting US Forces in Southeast Asia.

Establishing the reporting system for producing the required detailed raw data and the basis from which the necessary computer programs could be developed necessitated lengthy and continuous collective efforts on the part of staff personnel from the Joint Staff, the Services, the Commander-in-Chief Pacific and his service components.

It was felt in the beginning that the task of coordination might prevent early achievement of a reporting system which could satisfy the requirements and interests of the various levels of command and staff and of the National Command Authorities. Considering the magnitude of the task, however, it was completed in record time. The Headquarters USAF operational system analysis and programming contingent embarked on a maximum effort, which continues today, to provide a capability for appropriate Air Force personnel to review and analyze the detailed air operations reports flowing in from Southeast Asia.

Although the accomplishment of the "inhouse" personnel thus far can be looked upon with a measure of pride, the actual requirement has only been partially satisfied. There is continuous pressure on the system operator to produce greater numbers of capabilities within a much shorter time frame than is possible for the numbers of people available.

The above opinions and the Headquarters USAF example leads us to some fairly firm conclusions on the "total system" approach to design and development. For example, there can be no argument that command and control systems must be based on an integrated-function concept to insure correlated staff participation in the command and control function and to make the most effective use of the power of a computer.

On the other hand, to arrive at an integrated system in "one pass at the target" may be attempting to do too much at one time. The calendar time required for system development is excessive which exposes the concept to erosion and vacillation processes and the functional design itself may be overtaken by events. Lengthy on-site development activities disrupt command center operations and strain the patience of command center personnel, operational supervisors and staff users as well.

The system operator must get an operational system as soon as possible so that he can respond to his immediate requirements. His system must incorporate all of the available and worthwhile "state-of-the-art" tools which will provide for flexible and rapid response. The system operator's attitude can be reduced to a simple statement: "Get your head out of the clouds and show me some results—now!"

If we rule out the series of systems and total system approaches, you may ask: "Where do we go from here?" This is a valid question and it must be answered.

There should be five objectives in the design and development of an automated command and control system:

- Build to accommodate the Department of Defense concept.
- Do not accomplish basic system design and development on an operational site.
- Get something operational as soon as possible.
- Provide proven and powerful hardware and software tools to assist in rapid modification of the functional software subsystem.
- Plan from an overall approach to avoid duplicative software developmental effort.

The above objectives can be accomplished in a straightforward approach. First, we must undertake to define the logical design parameters and hardware concept for a system which will someday accommodate the requirements of the participating functional staff agencies. Secondly, we must insure that the equipment to be selected is truly "off-the-shelf" with a proven reliability history and will accommodate the system design. Once equipment selection action has been taken, however, we must insure the overall system design remains within the design capability of the selected equipment. (This will avoid questionable "cloud nine" development.) Next, full budgetary support must be given to the development of the basic control, support and utility software to insure that the required degree of flexibility and responsiveness is an inherent attribute of the system. The resulting basic system should then be intalled in a central Air Force "overhead" facility and subjected to a rigid "shakedown" period. This facility must be assigned to, and under the direct control of, Headquarters USAF.

It is at the "overhead" location also that functional software applications should be added to the basic system, tested and determined to be fully acceptable by the USING command. These applications may be either emulated or simulated versions of the current operational system at the command of primary interest or new programs developed by that command with "inhouse" personnel and/or with contract support.

When a specified operational base-line is reached, the system should be installed at the operational site and turned over to the operator for application and concentrated effort toward rapid growth. This approach eliminates many of the cited deficiencies of both the series of systems and total system approaches. It concentrates initial developmental activity on a solidly based system which can truly grow in an evolutionary manner and in "quantum jumps."

Of course, this approach also has its unique problems among which is the necessity to define the parameters of the integrated software system concept. Obviously, there will be trade-off problems to be overcome between the tendency to go overboard with generalization and modularity in the basic support software package versus the need for a specialized basic software system to support the unique requirements of command and control.

The most promising aspect of the proposal, however, is the opportunity to establish a standard approach

Air Force-wide for the basic tools to be incorporated into a system. This would permit the concentration of developmental funds toward the design of a superior basic system which could be amortized through its use universally by the commands and agencies receiving acquisition approval.

The elimination of problems previously discussed plus an opportunity to introduce reasonable standardization in the development of Air Force command and control systems causes this system operator to support the "see it before you buy it" approach just explored. This leads us to the next area of discussion and the definition of associated popular phrases from the system operator's point of view.

Compatibility and standardization

The word compatibility is normally used in reference to operational relationships between systems. In the discussion to follow it will be shown that there are two valid levels of compatibility: One directly concerning operational relationships and the other related to the economics of Air Force management of operational systems. This latter interpretation is, in turn, closely related to standardization goals which shall also be specified.

Compatibility, as generally applied in policy papers, refers to the ability to transfer data from one system to another "without buffers." This area of consideration constantly presents a problem of communication between the policymakers and the more technically oriented operators. Because of this deficiency in technical understanding at some policy levels, there is the even present danger of the publication of unnecessary restrictive policies or those which contain wording subject to conflicting interpretations. It is extremely difficult to get the point over that flexible file maintenance (data control) and retrieval systems can be designed to respond rapidly to any operational reporting requirements levied by higher authority, regardless of whether the directive originates from an operational command authority or a higher service headquarters. It is also difficult to convince these same personnel that the overall system design, programming language, specific retrieval language attributes or model and design of computer equipment have little to do with the ability to transfer data. Of course, it is mandatory that the data storage and retrieval software elements of the systems be designed to accommodate and be operated according to Department of Defense standards for Data Elements and Related Features (Codes) and Terms of Reference.

However, if the objective is to transfer all operational data machineto-machine in digital form, then a common-user digital communications system is required. The communications system, having a standard language, must serve as a translator between the originating and target machines. In this case, translation occurs even if the interacting machines are of identical make and design. Of course, the common-user communications system requires the use of hardware buffers between communications terminals and the computers being served.

Although the elimination of buffers is a policy goal, its achievement may be easier said than done. In the first place, command and control systems rely heavily on data support from comptroller-type management-oriented systems. Departing from the common-user communications system concept (and the unwanted buffers) would require standardization of all interacting computer systems in use in the Department of Defense and probably certain other Federal agencies and departments. Further, a dedicated, completely compatible, communications network would have to be established between the systems involved. Obviously, this would be unrealistic under current policy and will remain so unless it is proven that the concept of common-user communications systems is inadequate for command and control purposes.

It should be apparent that the buffer problem is primarily associated with hardware and basically unrelated to the software design insofar as compatibility is concerned. Therefore, it can be concluded that the Air Force can fully satisfy compatibility requirements for data transfer by building flexibility into its systems in the manner described.

At the next level of compatibility, the cold facts of economic necessity come into consideration. This orientation of the meaning of compatibility falls within the purview of the service secretaries who are charged with the responsibility for effective management of their respective departmental resources.

The meaning of compatibility from the service aspect, according to the author, should be defined as follows:

> *"Standardization of both hardware and software to the degree necessary to enable the transfer of personnel and computer programs, as well as operational data, between operational systems for immediate utilization, while retaining maximum flexibility for individual commands to add, delete or modify functional software applications to satisfy unique requirements."*

This definition will support the overall command and control system concept, enhance the ability of the Air Force to economically manage its operational systems and will insure the ability of each command to respond to its respective operational command authority and resource management channels as well.

An actual Air Force example of the basic concept on which the above compatibility definition is founded is the AIR FORCE INTEGRATED COMMAND AND CONTROL SYSTEM. As previously mentioned, by 1965 Headquarters USAF had exported the small-scale system, developed in the Headquarters USAF Command Post, to five of its major commands of which four are components of unified commands. This action was an initial step to provide five command centers with a relatively inexpensive automated capability from which expansion could emanate and through which the commands concerned could be introduced to the automated command and control concept.

Under the rules established in the basic approval directive, the EXECUTIVE SOFTWARE SUBSYSTEM (Control, data control, utility and retrieval complex) cannot be modified except on approval of Headquarters USAF. The operational or functional applications included as a part of the program package have been released for modification at the will of the command concerned. However, the commands are required to coordinate with Headquarters USAF on intended modifications to the operational programs and on the development of new programs. The single purpose of this requirement is to prevent duplicative efforts between the six systems (including the Headquarters USAF system).

Maintenance of the EXECUTIVE SOFTWARE SUBSYSTEM is retained as a responsibility of the Headquarters USAF Command Post computer programmers. Responsibility for the maintenance of modified functional programs or of any new programs is assigned to the originating command. This responsibility includes analysis, programming, documentation and maintenance.

System operator conferences are held on a regular basis to discuss operational problems and to exchange ideas. These conferences have been the point of origin for the present managerial approach to system configuration control. There is wide support for the concept of central control over the Executive Program subsystem for it is through this managerial technique that the ability to transfer operational programs between the systems is assured. Further, the commands are able to devote their resources to the development of operational capabilities unique to the command rather than having to expend their limited technical capability on executive subsystem maintenance.

The system operator conferences have also resulted in current emphasis on a detailed standard for computer program documentation. Since various commands are using programs developed by other commands in the system, the need for qualitative control over the standard of documentation is an absolute necessity.

Each command has the responsibility for coordinating with its respective unified command headquarters to insure operational capabilities being considered by that command for development will, in fact, be responsive to operational requirements of the unified commander. Of course, coordination does not occur if the program is designed only to support internal requirements. In the event the unified command has a requirement for operational data formats which impact the executive program, the Air Force component commander must submit a requirement for its revision to Headquarters USAF. It is interesting to note that this form of basic modification has not, as yet, been necessary.

The Air Force Integrated Command and Control System is planned for replacement of its present hardware in the 1970 time frame. This system has introduced a concept embodying a practical and economical approach to system development and management. It treats Air Force Command and Control Systems as a part of a total resource for which the Secretary of the Air Force has the responsibility to manage and for insuring that operational requirements placed on each element of the total resource are fully satisfied in the shortest possible time, regardless of whether the source of the requirements are internally or externally generated.

It is appropriate at this point to divert for a few paragraphs to clarify previous statements concerning the requirement for an "overhead" facility assigned "under the direct control of Headquarters USAF."

The very notion of standardization carries overtones of serious import on the prerogatives of the commander. These prerogatives must be protected at every level of command to the maximum extent possible. With this as a controlling factor, justification for standardization of automated command and control systems is invalid unless founded on operational necessity or when there are overriding economic considerations. Significantly, there are two important objectives in Air Force command and control: (1) There must be continuous effort toward the achievement of ease in operational interaction between command and control systems; and (2) all possible means must be fully exploited to achieve economy in the development and operation of automated command and control systems, particularly in the area of software development. Standardization facilitates both of these objectives.

The Air Force Integrated Command and Control System has proved that it is feasible to exercise central management over standardized elements of systems without interfering with command prerogatives to an excessive degree. It has also validated the concept that software economy can be achieved through inter-command coordination. It is important to note, however, that standardization has been imposed only to the degree necessary to permit mutual support, leaving considerable flexibility for the local commander to satisfy unique requirements. It is also important to emphasize that AFICCS management-supporting procedures are designed to encourage, rather than stifle, individual system operator initiative.

Since standardization enables an environment of mutual support, effective mutual support operations are therefore dependent on the existence of standardization. Once the standardization criteria has been established and systems have been designed accordingly, then procedural machinery must be set in motion to retain that degree of standardization. This means control.

The publication of procedures to define that which is to be controlled and the method by which control is to be exercised is adequate for the purposes of implementation. However, effective control is impossible unless the necessary capability is provided to monitor system activities and for follow-up action to assure compliance with standardization and procedural policies. This requirement is critical in the control of standardized software. Experience has shown that the agency charged with this responsibility must have detailed knowledge of the standardized system elements and, as a matter of fact, the agency performing software control must actually *maintain* the standardized elements of the system.

This brings us to the crux of the requirement for an "overhead" facility and the necessity for it to be properly aligned organizationally. There are several factors which must be considered before determining who should be charged with control responsibility and where it should be performed. In the first place, the dynamic environment in which automated command and control systems must operate demands unfettered action and reaction between system operators. This in turn requires strong leadership to insure such activity is conducted in an organized manner thereby taking full advantage of all opportunities for mutual support. Secondly, control must originate at the source. Each system operator has an individual responsibility to impose strong control over his system in the interest of good management and to insure compliance with the policies of higher headquarters. Effective local control is a basic ingredient for success in the Air Force-wide management function. Therefore, the overall control function is dependent, to a great extent, on the aggressive and willing cooperation of the leaders of the various systems involved. Thirdly, the ground rules for control will occasionally make it necessary to disapprove a command recommendation. Obviously the controlling agency must act with the full support and authority of the staff at the level where control is to be exercised. Fourth, an agency assigned responsibility for supervising standardization aspects of systems should not be directly associated with an operational system where self-inspection would be necessary and where responsibilities would be divided between the overall management requirement and the local system support requirement. Fifth, the organization assigned the control responsibility should be functionally aligned in a parallel manner to the subordinate systems. The purpose of this is to provide for lateral staff coordination within a single functional area for operational command and control matters. Finally, control can only be imposed on the basis of respect for authority. More specifically, control must be placed at a level higher than that of those organizations being controlled. It cannot be conducted under a theory that technical competence will be the primary qualification for the controlling agency. The proper criteria would be to equip the control function with authority (through organizational alignment) and technical competence as well.

The solution as to who should control what and where it should be performed is obvious to this author. As previously stated, systems are organized under and assigned to the Director of Operations at the various organizational levels uniformly throughout the Air Force. Therefore, it would be logical to retain functional responsibility of operational command and control matters under the Director of Operations at Headquarters USAF where it currently resides for the Air Force Integrated Command and Control System. This alignment takes advantage of the excellent communications which exist between Headquarters USAF Command Post and Air Force Major Commands and, more important, keeps lateral continuity in inter-system staff coordination.

Under this concept, the "overhead" facility would be a control facility managed by operators. This operator element would: Have an assigned personnel capability to perform maintenance on standardized system software; provide technically oriented operational advice to the staff of Headquarters USAF; control documentation standards; establish operational procedures; and would perform operational evaluations to determine the effectiveness of the Air Force systems involved. The ideal environment in such a facility would include a technical support contingency, backed up in depth by Air Force Systems Command, to accommodate requirements for major improvements and other developmental tasks which exceed the production capability of the system maintainers. Placing the overhead system directly under and responsive to Headquarters USAF through the Director of Operations would provide the authority necessary for effective control. With operationally oriented personnel in charge of the facility, the cooperation of the Air Force major commands involved would be encouraged.

To further emphasize this latter and most important point, it is certain that operators will be more prone to assist and cooperate in the function of control if the leadership is carried out by operators who are "in the same boat" and who are directly associated with the next higher headquarters. Headquarters USAF is that higher headquarters in the case of Air Force major commands.

When an overhead facility is discussed in some quarters, the impression is given that it would be used primarily for experimental and development activities pertinent to current operational systems. It is recognized that this form of activity is essential to evolutionary growth and improvement and is of such importance that it might even be reasonable to invest in a second overhead facility for the specific purpose of experimental and developmental functions. However, if economical considerations prohibit this course of action, then developmental and experimental operations would have to be conducted on the control facility. Under this situation, the overhead facility would primarily be a maintenance and control facility with other requirements taking second place in order of priority.

Specific Areas for Standardization

Within the overall framework of the conceptual approach of the Air Force in the management of command and control systems and from the system operators point of view presented herein on design and development, a strong case can be made for a realistic approach to standardization. Actually we are proposing a specific plan of attack and specific objectives to be used as a basis for development and management of Air Force headquarters level command and control systems.

From the overall point of view, the subjects pertinent for discussion are the programming language and its associated compiler, the retrieval language, the control, support and utility program subsystem, the hardware configuration, computer program documentation, and general operating procedures and standards.

Action has been taken by Headquarters USAF to establish Jovial J3 as the standard Air Force command and control programming language. This particular language was selected because it is currently in wide use and has been successfully used over a number of years. It is a solid base from which evolution and improvement can occur, but on a controlled basis. In the near future an Air Force Manual on Jovial J3 and an associated general compiler specification will be coordinated for publication. This manual will establish, for the first time, a standard that is expected to facilitate ease in personnel transfer between systems, implementation of an Air Force-wide standard training program which the Air Force Training Command can support, easier transfer of programs between systems with different makes of processors, and easier transition to new processors without total software losses. This standard will be particularly directed toward headquarters level systems but will be generally applied throughout the Air Force with a stipulation that specific approval will be given for diversion from the standard in justifiable cases. The primary objective is to eliminate, as much as possible, unique language forms and compilers.

Although the above is a statement of intent on the part of Headquarters USAF, the primary concern of the system operator is not the name of the language but rather one of assurance from the policymakers and the developmental community that (1) a standard language is identified and associated with a standard compiler, (2) changes to the standards are centrally controlled, and (3) the compiler (and its language) established as a standard is determined to be adequate for the command and control task.

The retrieval language is only beginning to receive attention. It can be envisioned in the future that participating staff officers will require direct access to the command and control computer system through the use of communications consoles. However, a major problem is confronted in training potential staff users to use a retrieval language. The staff officer trainee is normally totally committed to the pressures of his functional duties. He finds little time for training and, even when given detailed training, his use of the system is sporadic in nature and therefore his knowledge of the system software capabilities and of the retrieval language is rapidly lost. There is a general feeling that the concept can only be successful over the long period if the requirement for knowledge and use of the system is made a "way of life" for staff personnel. This means the introduction of the concept at the pre-commission schools such as the Air Force Academy and at the professional schools of the Air University. By exposure to the concept in an educational environment, the possibility for permanent retention will be greatly enhanced. This means, of course, that there must be some basic retrieval language standards used in command and control systems on which the educational and training programs can be founded.

We have covered in some detail the concept of a standardized executive subsystem software complex in previous paragraphs. The advantages of this approach are primarily based on the requirement for economy. However, there are inherent operational advantages which cannot be overlooked. These have also been briefly covered.

It is the opinion of this system operator that hardware standardization can be achieved to a very effective degree. There will be variance in the configuration between individual systems in the numbers of consoles, large panel displays and other peripheral devices. There also will be cases where a command center is provided more computing power than the task at hand is estimated to require. However, these variances and apparent possibilities for "overbuy" may not be significant. The basic software system can be designed to accommodate the varying numbers of displays and other devices without impacting the standardization criteria of the software system. The instances of "overbuy" will become insignificant when amortized over several systems. Further, original estimates of system requirements are consistently underestimated. Finally, it is well known that hardware costs are generally on a downward trend. For example, the cost of core memory was, at one time, an item which was subjected to close scrutiny by the budget monitor. This is no longer the case. The cost of core and other items of hardware is considered in context with overall objectives and costs. Software is the big cost item which is now considered the area where economy is a necessity. Economy is only possible through a concept of system design which will permit mutual and collective support between command system operators.

Computer system documentation is also currently receiving attention at both Department of Defense and Service levels. There is a collective effort under the Organization of the Joint Chiefs of Staff to establish such standards for the World-Wide Military Command and Control System. The Air Force is supplementing these general standards with supporting procedures which will permit effective control over quality, production and use. A concurrent Air Force effort is under way to automate the development of program specifications. The system is planned to provide for automatic production and updating of program documentation tapes. Consideration is being given to having these tapes forwarded to Headquarters USAF where they can be interpreted by a special purpose machine which will produce line-, paragraph-, page- and flow diagram-associated off-set masters. These masters could then be delivered to Defense printing for the production of final documents. The objective, of course, is to economize in the production of documents as well as remove the drudgery of flow diagramming from the computer program specialist.

Air Force operating procedures and standards for headquarters level systems is an area requiring considerable attention and action. In order to provide needed guidance, work has begun on the production of several Air Force manuals and regulations. The directives will establish operating standards and will provide guidance on such subjects as the management of operational systems, general facility requirements, and skill, manning and training requirements. These documents are being developed by the Air Staff with the support of the Electronic Systems Division, Air Force Systems Command.

SUMMARY

The headquarters level command and control system is presently receiving concentrated attention at the policy, operational, staff management and developmental levels throughout the Department of Defense and particularly in the Air Force. It is our opinion that these types of systems will, in the future, evolve from a specified base-line rather than being submitted to a complete developmental program.

The time is past where operational activities in a command center will be subjected to, and disrupted by, lengthy automated system developmental activities. The base-line system will be developed in an "overhead" facility. The voice of the operator is now being heard and is insisting on system development from a "foot on the ground" approach. He wants flexible and responsive tools on which he can respond today and build for tomorrow.

The Air Force must approach its command and control systems from a perspective of a worldwide, total resource. It must provide its component commanders with systems that can rapidly respond to the requirements of their respective unified commands and to Headquarters USAF, but with priority of emphasis to the operational command channels. These systems must also, however, be designed to operate in mutual support of one another in the "air environment." The need for mutual support is based on the fact that there is a worldwide similarity in air operations through Air Force Doctrine for the employment of Air Power and through standard air operational policies and procedures which unavoidably effect command and control operations. The prospect for effective management of this command and control resource embraces, to a great extent, the same resource management principles which have provided the uniforms we wear, the aircraft we fly and policies under which we function as an Air Force.

The Air Force standardization objectives which have been discussed are basic and must be accomplished within the framework of guidance provided by the Joint Chiefs of Staff for the World-Wide Military Command and Control System. Standardization will have for its purposes the attainment of the greatest degree of economy possible without degrading the requirements of Air Force commanders (including those requirements imposed by their respective unified commanders for commands so assigned) and to facilitate ease in mutual operational support, not only between operating commands, but including Air Force support commands as well.

It must be made clear that this approach to the overall management of the Air Force command and control resource must not be confused with the *functions* of OPERATIONAL COMMAND and RESOURCE MANAGEMENT which are respectively performed by the unified commands and the services. The essential point to remember is that, in order for these functions to be performed with continuity and effectiveness, the Air Force command and control resource (or system) must be looked upon and managed as an entity. The mission of Air Force command and control systems, both individually and collectively, is to perform all functions required of them by appropriate authority. Therefore, the entire Air Force command and control management apparatus must be oriented toward this objective.

The Air Force Integrated Command and Control System is considered a tangible example which validates the concepts and opinions expressed in this paper as a system operator's point of view. We believe these concepts and opinions outline the future profile for development, operation and use of Air Force Command and Control Systems and possibly that of the World-Wide Military Command and Control System.

A methodology for command information system analysis

by JOHN A. EVANS
The MITRE Corporation
Bedford, Massachusetts

RATIONALE FOR DEVELOPING A METHODOLOGY

The requirements translation gap

Effective application of computer support at every level, up and down, of the command hierarchy constitutes one of the necessary first steps in evolving toward a more responsive command and control system. This is not a simple assertion; rather, it is an indication of a complex dilemma that exists between two communities: the user, who is operations-oriented, and the designer, who is technologically-oriented. The situation is further complicated by the fact that these communities have tended to grow toward a more inextricable interrelationship with every application of automated aids. Their experience, gained through several iterations of the painful command system acquisition cycle, leads them to generally agree that there is an inadequate contextual understanding of what a command information system is and does, and why. Similarly, a common understanding of the command hierarchy within which an information system is embedded has yet to be developed.

The results of these inadequacies provide the basis for the dilemma. Command information users, for example, tend to characterize and specify the type of computer support needed from a base that is either too gross (we need to plan faster) or too narrow (we need a computer with a 16-K memory core). On the other hand, designers, in spite of an intimate knowledge of existing technology and projected trends, are unable to create and put together relevant design approaches. Often, their system designs incorporate unintentional biases favoring one solution. Thus, detailed hardware/software specifications are not meaningfully related to the command's operational environment, functional activities, and problem areas.

The problem of applying computer technology was precisely characterized by McLucas (1966), who stated:

> *We have been generating technology faster than we have been able to apply it. . . . I believe the emphasis in the next 20 years must go more to applying technology than to creating new technology. . . . The next generation of technological progress has to concentrate on systems analysis and systems engineering, the putting together of complex systems. . . .*

Failure to establish a common contextual basis for user/designer interaction has retarded effective assimilation of computer support for command information systems. The resultant dilemma created can be characterized as a requirements translation gap. Hobbs and Craig (1964) alluded to this gap when they wrote:

> *On the one hand, there is often an explicit description of the functions to be executed by a number of black boxes with performance standards. On the other hand, there is usually a vague, if any, description of the operational concept and requirements for the overall system.*

McCarn (1965), at the Second Information System Sciences Congress, was more explicit:

> *Often some rather arbitrary design criteria have been specified, such as, "the system must be able to respond to a query in three minutes" (or three hours), or "data in the files will include all material received to within the last two hours," or "displays in seven colors are required with a positional accuracy of .05% of full scale." Such design criteria often pass for system requirements. On the other hand, the requirement may be based on "the increasing flow of information and the compression in time of the decision making process."*

The urgent need to resolve the gap

The nuclear age underscores the need to develop a method of selective response (Benington, 1965), tailored to various levels of escalation, and intensifies the importance, and the complexity, of providing a methodology for information systems analysis (Morgenstern, 1962). *More* information* about globally dispersed operational situations which is *more* relevant to the maintenance of context (Thompson 1964a) must be *more* selectively and rapidly accessed, assessed, and distributed within the command hierarchy. At the same time, greater emphasis must be placed on using the entire command information structure to develop meaningful, multiple option plans, as opposed to using it primarily for the maintenance, execution, and control of a strategic integrated operational plan which would be implemented during the course of a thermonuclear spasm response.

This must be done to guard against the danger of uncontrolled escalation and to permit a more fine-grained tailoring and deployment of limited U.S. combat force resources. Implementation of a selective response policy (Taylor, 1959) in the face of the threat of escalation requires the command information system network to be less stress-sensitive to overloads and more rapidly adaptable to the data flow within and between commands. It also implies that military officers directing or involved in the staffing process be allowed to regain the initiative and flexibility of command they have started to lose to computer-centered command and control systems. Such systems hinder initiative because they require the skills of analysts/programmers (middlemen) to translate the generally implied actions desired into specific man-to-computer, computer-to-man operating procedures and assessments. The potential danger created is that the middleman may unintentionally provide a misleading response.

Flexibility also diminishes because the data base is rigidly formatted, and man/machine procedures are rigidly planned. In some instances, the preplanned execution steps triggered by a specific query may not be correctly interpreted by the user. In order to restore flexibility and initiative, the computer's attractive storage and processing capabilities must not only be retained, but further exploited. Third-generation computer technology (Licklider and Clark, 1962; Bennett, Haines and Summers, 1965), because of its on-line, interactive, and distributable features, offers potential for restoring these characteristics to military officers. If key staff members are provided with greater on-line control of easy-to-access storage and processing capabilities, they will be able to effect significant improvements in the management and distribution of data that connects operational forces and sensors with the operational commanders who alert, deploy or commit combat resources.

Effective assimilation of third-generation computer technology emphasizes the need for first performing more command analysis homework in order to establish a sound basis for improvements in command information system design: more explicitly, the requirements translation gap must be closed. This basis can be achieved by working toward a methodology such as the one described in this report. It is believed that the net effect of implementing such a methodology would be the realization of a significant decrease in the probability of misunderstanding between user and designer, and a significant increase in the probability of developing more relevant design approaches. Through application of this methodology, some necessary first-steps in evolving toward a more responsive command and control system could be initiated.

*Information as distinguished from data (See page 255).

The nature of the proposed methodology

Resolution of the user/designer dilemma requires the establishment of an integrated, problem-oriented methodology that encompasses the basic tools of systems analysis. Initially, the methodology implemented would provide a frame of reference for the user to help him delineate more relevantly those operational and functional command procedures that can be improved through incorporation of computer support. As the methodology evolves, it can be extended, integrated, and synthesized with other approaches to achieve its ultimate objective: the creation of an analysis framework that stimulates more effective user/designer interaction through a common contextual understanding of command information systems and the command hierarchy.

Problem formulation

A systems analysis-oriented framework, such as the methodology would provide, is needed by the user/designer communities for reassessment and future redirection of evolutionary command information system development. Many experienced designers in industrial and military organizations now agree (Edwards, 1965; Armed Forces Management, 1963; and Institute for Defense Analyses, 1961) that no problem formulation, the first six phases in an operations research process, has been underrated and needs a great deal more attention. E. S. Quade (1964) characterized the situation when he wrote:

> *A real pitfall is the failure to allocate and to spend a sufficient share of the total time available deciding what the problem really is. The problems faced by the system analyst frequently belong to that class in which the difficulty lies more in deciding what ought to be done than in deciding how to do it. It is a pitfall to give in to the tendency to*

"get started" without a lot of thought about the problem.

(See also Churchman, Ackoff, and Arnoff, 1957)

Inadequate characterization of the problem often leads to a study of the wrong problem or to the preclusion of some design alternatives. This can be avoided if the problem is stated in terms of functional needs, without implying how it is to be solved. The tools of analysis that can be derived through the implementation of the proposed methodology can be applied to problem formulation.

No matter how inadequately a problem is characterized, its meaning to a systems engineer who attempts to design a headquarters command information system is translated into an inability to answer certain specific questions, such as:

- What does the commander and his staff do?
- What are the procedures and data used in creating command products, such as orders and reports?
- What are the throughput rates of the various parts of the system?

Until a common contextual understanding of the command structure is acquired by the user/designer communities, questions such as:

- What belongs in the data base?
- How should it be organized?
- How and in what format will the operator want to recall processed information from the data base?

cannot be rationally answered. What is needed, then, has perhaps been most cogently stated by Major General Gould (1966): "The need for automation in the command and control process has required a new discipline—the science of detailed analysis of the job to be automated and the precise definitization of how the job is to be done."

Van Horn (1964) was one of the first to recognize that current limitations to implementation of computer-augmented command systems stem largely from inadequate analysis of information structures, decision-making, and management organization:

It is interesting to note that the majority of technical people in the command and control field are specialists in hardware design, while the major problems lie in determining information requirements, selecting good decision rules, and developing the system to implement these information structures and decision rules. This discrepancy may well be the most significant problem in the field.

The need to understand much more about the current military information systems structure before installing computers was underscored by Lewis (1965) when he equated the "study of command" to "the study of the distribution and exercise of power." He further hypothesized a law of "Conservation of Power" which dictates that any significant change in a command capability causes a change in the distribution of power, and, consequently, "will be resisted—by some." A primary and particularly important objective of prior analysis is to clearly indicate, to both the designer and user, the subtle impact of various possible computer support alternatives on the user's ability to execute assigned missions.

Purpose of the report

The awareness of the requirements translation gap accompanied by a growing sense of urgency to close it indicates the need for a better understanding of the problem. The purpose of this paper then is to:

- clarify the nature of the gap;
- contribute toward a methodology for closing this gap by characterizing the types of analyses/products needed;
- make an urgent plea for:
  1. synthesizing and extending the existing applicable analysis effort, and
  2. preparing an instructional package, complete with methodology and case study experience.

The urgency of this plea stems from the need to accelerate the systems analysis orientation of military and technical personnel who are currently involved in taking next-steps toward evolutionary development of specific command systems.

Scope

This paper focuses on the initial information systems analysis effort (see Figure 1) which must be undertaken in order to provide systems designers and management with a basic operational and functional understanding of the command, and with a context for evaluating and controlling the direction of subsequent command system development. These initial analysis activities are emphasized as opposed to those which take place "downstream" (shaded areas, Figures 1 and 3).

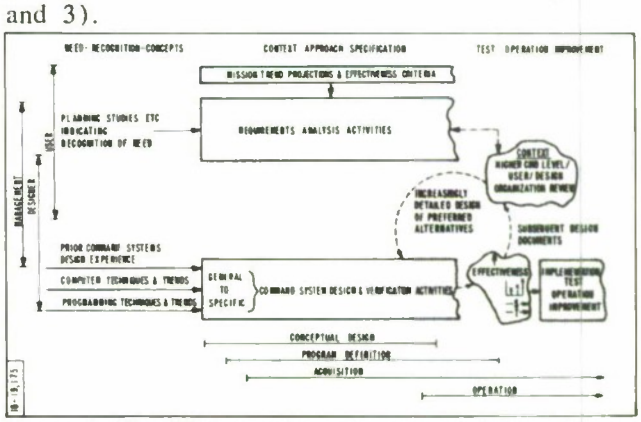


Figure 1—Simplified overview of command information system analysis and design process

Because the report concentrates on a methodology for *command* information system analysis, the discussion and illustrative material focus on the information flow related to plan generation/modification—a function of particular concern to component command levels and above (see Figure 2).

The illustrative material in particular is drawn from analyses of a Unified/Specified command and several component command headquarters because commands at these levels* critically need a capability to help them replan more effectively. Moreover, it is in this area that the potential offered by computer technology appears to have broadest application: namely, to permit a more flexible command response to a wider range of contingencies, tailored to various levels of escalation.

Organization

The remainder of this report is organized into four discussion areas. The first two clarify the nature of the requirements translation gap by defining some of the terminology and by presenting an overall description of the command information system design process. Collectively, these sections provide a framework for subsequent discussion. The various types of recommended analytical efforts are identified Analysis Methodology, while the following section briefly evaluates the advantages and limitations of the approach. The last section is reserved for conclusions and recommendations.

THE COMMAND INFORMATION SYSTEM ANALYSIS PROCESS— SOME BASIC TERMINOLOGY

INTRODUCTION

A logical first step for closing the translation gap is to discuss some basic terminology and characterize those aspects which have a bearing on improved analysis and design.

The command hierarchy

Command information systems, already intricately sensitive, information capacity while maintaining the related, are continuously converging toward a detailed and standardized relationship through the acquisition of computer support subsystems. Thompson (1964b) presents a vivid discussion of how the command hierarchy achieves its large, though increasingly stress-operational context of commanders and their staffs. This aspect will not be further mentioned except to note its relevance in comprehending the information structure of the hierarchy.

Assuming further agreement can be reached on the distinction which has been made between command and control functions, it is useful to comment on the time-phased hierarchical nature of this broad planning (command emphasis) and operations (control emphasis) process (see Figure 2). Since this report focuses on *command* information systems, it is concerned primarily with the analysis and characterization of the information flow related to the generation and modification of operational plans—plans which focus on the deployment and employment of forces, service-directed logistics support to those forces, and transportation movement support. More specifically, this paper deals with the component command levels and above (see Figure 2) because of the emphasis of these commands on longer-range plan generation and

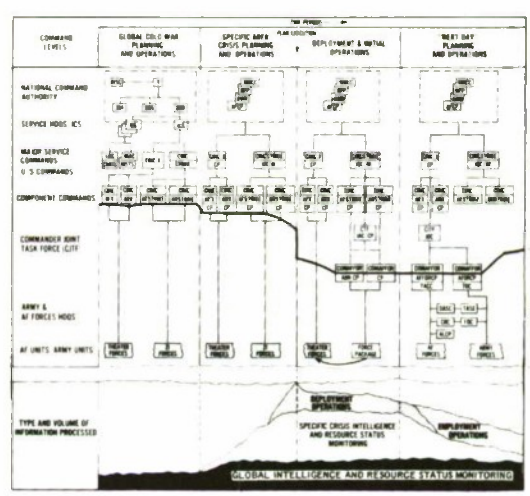


Figure 2—Time-phased hierarchical nature of operational planning and operations

*L. S. Christie and M. G. Kroger (1962) state that: "*Command* functions *involve broad problems of planning*, assessing the capabilities of the command forces and those of the enemy, allocating resources, alerting and committing the command forces. These functions require the gathering of large amounts and many classes of information, aggregating the information and processing it to enable the commander to make knowledgeable, deliberate decisions in the context of changing objectives. *Control* functions, on the other hand, characteristically involve the direct control of weapons in situations where, although the volume is large, the information can be categorized into relatively few classes, and since the objectives are definite and fixed, the problem is one of directing action toward the objectives through error detection and correction."

modification. The emphasis is restricted, however, to the development of the multiple-option type of limited war planning required to implement a *flexible* response (*i.e.*, contingency planning) as opposed to the emphasis given to a thermonuclear spasm response which is characterized by a more centralized, detailed type of preplanning.

Note that both centralized and decentralized information processing are desirable, depending upon the crisis-related time period because, as a crisis develops, massive amounts of dynamic intelligence and resource status information must be acquired, synthesized and forwarded up through the hierarchy (McCarn, 1965). After employment operations begin, additional operations monitoring data must be digested. Plan modification for shorter time periods must be further delegated in order to avoid what Thompson (1964a) has called the "catastrophic fractionization of context," caused by distorted reports and time delays. Once the crisis action operations have stabilized to some degree, a tendency to reassert centralized planning occurs. Viewing the command information structure in this context, it is perhaps easier to see why more attention must be given to understanding the existing hierarchical planning and operations process in order to *effectively* assimilate computer technology in a way which increases its flexibility and responsiveness. If this is not accomplished, the formal and informal staff to staff and intrastaff communication network, which has served well over the years, will become rigidified and less responsive to the command hierarchy.

Command information

The primary reason for developing elaborate data processing schemes is to supply the command organization with the critical facts (*i.e.*, information) needed to plan and control operations. By conceptually distinguishing data from information, systems analysts can concentrate on mapping only the critical aspects of the command information structure and, in addition, designers will have better guidance for determining to what part of the data base computer support can be applied. Throughout this report, data refers to all the facts which could be obtained, while the term information denotes the particular facts commanders and their staffs ought to know in order to address a particular problem. Information is thus derived from raw data, but it has certain qualities, such as accuracy, timeliness and relevance to a particular situation. These qualities permit the analyst, designer or manager to assess its *value* (Gregory and Van Horn, 1964) along with the *cost* of obtaining this information via more sophisticated data processing techniques.

Command information system

In this report, it is assumed that "command is a function of sophisticated people in a complex environment responding to an ambiguous situation."* Thus, a command information system consists of the people, facilities, equipments and procedures which enable the commander and his staff to interact in an attempt to define the ambiguous situation, to respond with appropriate plans, and to assign, alert, commit and monitor forces as required.

The boundaries (*i.e.*, data input and sources, processing capabilities, output products and destinations, etc.) of a command information system network are frequently defined by the limits of authority delegated to the organizations jointly charged with providing automated support to the command. Within a particular command, the following factors have tended to restrict even more severely the definition of the command information system:

a. limited in-house systems analysis resources and/or restricted freedom of action necessary for the identification and analysis of intrastaff information flow;
b. restrictive command system acquisition procedures which have not been modified to allow for early application of funding and technical resources to focus on the technical requirements translation problem;
c. blurring of intrastaff functional interfaces engendered by competitive staff associated with establishment of independent readiness centers;
d. acquisition of computer suport training packages (e.g., 473L operational training capabilities) which have not been tailored to a particular command's set of missions;
e. inappropriate level of responsibility and/or sufficient commander interest in developing a useful computer support capability; and
f. lack of analysis framework.

Brief comments on the last three factors cited appear appropriate.

The restriction imposed by d. presents a rather novel situation in that the arrival of a prepackaged set of capabilities could have a major impact on defining what is perceived to be the information requirements of the command. Particular care must be taken to ensure that the advantages of accelerating the training of an in-house staff do not lead to a warped characterization of the command's vital information requirements. A

*Quotation attributed to J. H. Lewis, Weapons System Evaluation Group, IDA; see article by N. P. Edwards (1965). Moore (1958) says, "Knowledge of people is the core of the improvement process because changes can be brought about only through people."

sense of urgency exists to define intrastaff requirements to guide the training capability toward one which serves and does not warp the command's mission execution capabilities.

In regard to e., note that if there is no top-level support of an intrastaff users group to encourage the identification of command-wide information requirements, the command system tends to become synonymous with those inputs which somehow reach the command and control organizational unit usually responsible for maintaining computer support to the command post. Similarly, the direct interim outputs of the computer (as opposed to the command) are likely to be reviewed as command products. The information "system" tends to be exclusively operations-oriented and is apt to neglect important intelligence, planning, and logistics interface requiremnts.

One purpose of this report is to contribute the framework cited in f. to

- expedite the system analysis orientation of in-house personnel,
- draw attention to the required changes in command systems acquisition procedures, and
- intensify commander interest in assuming a more active role in sponsoring a broader in-house information requirements analysis; this must be done in order to successfully net together the soon-to-be automated readiness centers with the training packages in a way which enhances the command's mission execution effectiveness.

OVERVIEW OF THE COMMAND INFORMATION SYSTEM ANALYSIS AND DESIGN PROCESS

Characterization of the analytical framework

The framework used in this report to characterize the nature of the recommended command information analysis effort (see page 259) is a two-stream requirements analysis and design process flowchart (see Figure 3). The various types of analyses and their respective documentation products are isolated from the design process and are shown as a part of a parallel stream effort. Preferably, this effort should be performed at the user's site, and, to an increasing extent, by those members of the user's staff who have been trained in systems analysis approach. It should be noted, however, that to the extent these analyses/products do not exist in an updated form at any given stage of a command system development, the methodological approach is intended to be flexible enough for application during later stages of the analysis and design process.

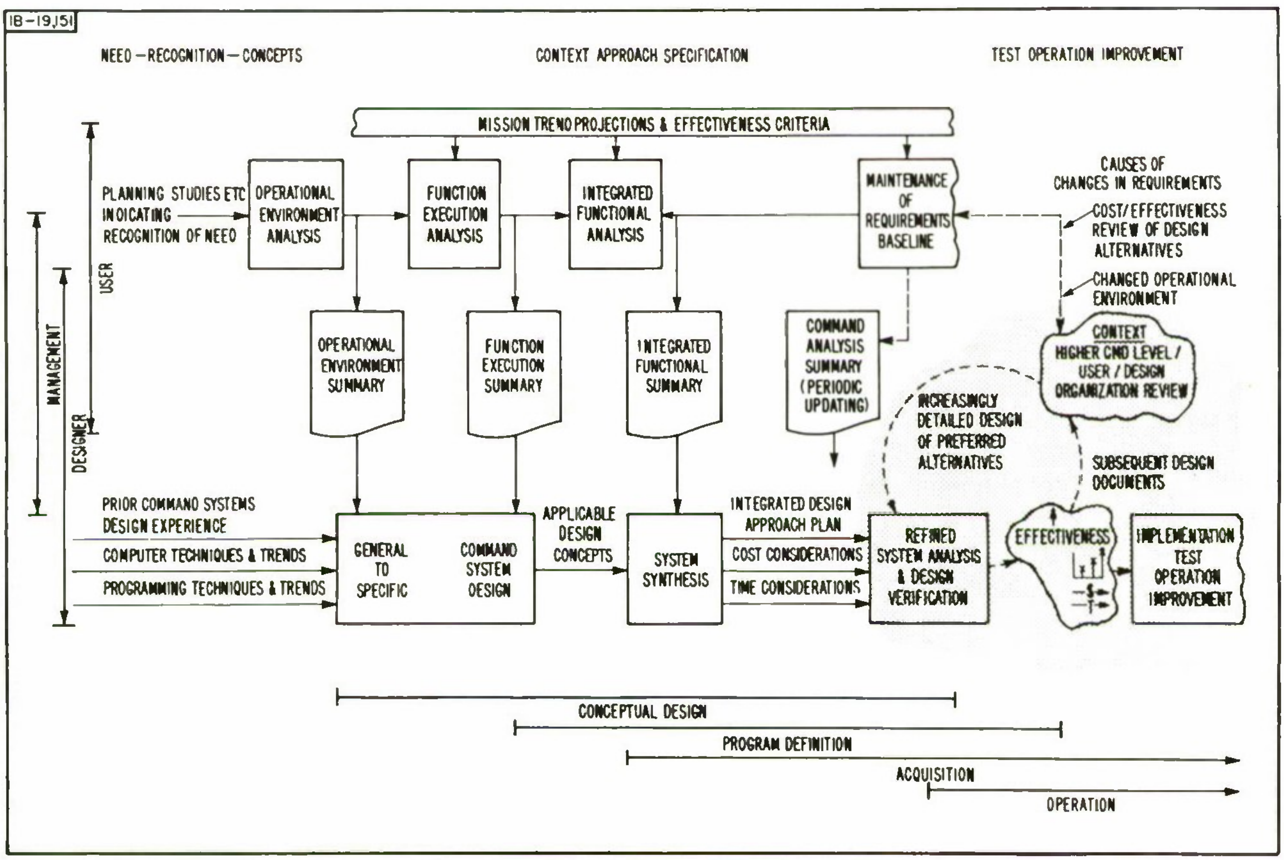


Figure 3—Command information system analysis and design process

The upper stream isolates those analytical activities essential to closing the requirements translation gap, and indicates their relationship to the rest of design process. Analytical efforts associated with the upper stream are usually more efficiently accomplished on-site at the using command, and, as the depth and breadth of the command's technical competence expands, should be conducted, to an increasing extent, with in-house technical resources. Conversely, analytical and design efforts associated with the lower stream can be conducted at off-site locations in order to conserve and to gain access to a broader spectrum of specialized technical resources.

On-site analytical efforts should generate four major products (documents):

- Operational Environment Summary
- Function Execution Summary
- Integrated Functional Summary
- Command Analysis Summary

Their primary purpose is to provide, in an iterative, usefully partitioned fashion, the job-oriented contextual understanding necessary for the development of applicable design concepts for the user, designer and higher level acquisition management.

The major stages in the evolutionary acquisition of an improved set of operational capabilities are shown on the lower part of the diagram to indicate that while this type of analysis may be required throughout the evolution of a command information system, it is emphasized during the preprogram definition stage. The command might recognize a need for increased responsiveness of its information structure or might become aware of a significant advance in the technological state-of-the-art, and, consequently, would initiate

Hardware	Software
• Compatible Computer Families	• New Programming Languages
• Fast-Response Systems, such as: Real-Time Systems, Time-Shared Systems, Data Communications Systems	• New Operating Systems being released, intimately tied to the new computers
• Extension of Optical Scanning Applications to read hand-written numerals (eventual automatic handwriting/voice recognition)	• Multi-Programming and • Multi-Processing Techniques which raise questions of data protection, audit trails, and job accounting
• Cheaper Mass Storage Units having increased capacity	• Generalized File Processing Software for converting file-oriented applications more rapidly
• New Data Communications Products made economical through use of time-shared, broadband channels	• Decision Table Processors for shortening software application development and reprogramming
• Output Display Devices more widely used for more rapid, more flexible man/machine communications	• Job/Function Applications Packages to reduce programming

Table I—Projected trends in computer technology

planning studies leading to the development of these products. The associated design/development/acquisition agency assesses the implications of these products, draws on its prior command systems design experience and on its knowledge of software/hardware trends (such as those shown in Table I), and, finally, creates specific design concepts applicable to that command.

Major analysis products

The first product, called an Operational Environment Summary, provides the design effort wih a basic understanding of the operational environment within which the command system must assist in the execution of command missions (see page 259, Operational Environment Analysis). The second product, called a Function Execution Summary, provides the system design effort with documents containing more flow-oriented data. These data indicate how each major command function identified in the Operational Environment Summary is executed. This type of product could be developed for one or more types of missions and/or scenarios; it should address dynamic impacts on the information structure, caused by the superimposure of new workload priorities during the various crisis-related time periods.

When all of the major command functions have been analyzed in this way, a more detailed, integrated study is made of interactions among these command functions prior to and during the execution of assigned missions. The product of this type of analysis is called the Integrated Functional Summary (see page 262, Function Execution Analysis). In addition, a Command Analysis Summary document should be prepared as required to update and summarize, from an integrated point of view, the results of previous information system analysis efforts.

Mission growth trends and effectiveness criteria

In order to avoid a common pitfall which involves the application of computer technology to an unexamined, *existing* information structure, mission growth trends and effectiveness criteria are also derived concurrently. These criteria, and derived indices (measures) that indicate the degree to which they are satisfied, assist in the assessment of present command function performance, the isolation of problem areas, and the evaluation of design alternatives. The results of mission growth trend assessment and the derivation of appropriate criteria are also generally useful in guiding the systems analyst toward *critical* information flows and the more important problem areas amenable to data-processing support. Appropriate portions of this information are incorporated into the recommendations of other summary documents (see page 265, Derivation of Functional Effectiveness Criteria).

The requirements baseline

Because the command system is evolutionary by nature, a requirements-oriented information system analysis provides a vital and continuous source of inputs for evaluation and control of the design process. In order to ensure proper redirection of the design process, the requirements baseline must be maintained. This involves recognition and immediate assessment of any development that indicates a change in the command's current specification of requirements. In this way, the operational and system requirements can be modified and adapted to changes in the operational environment and/or to the results of increasingly detailed analysis/verification/design activity (see page 266, Maintenance of the Requirements Baseline).

System synthesis

The analysis products are then meshed with applicable design concepts, generated from the preliminary command system design studies, to provide a framework for the System Synthesis effort. Essentially, this effort is used to determine the several ways in which files, information flows, processing rules, and human decision-making activities can be reorganized to enhance the execution of major command functions. A starting point in the System Synthesis effort is the derivation of specific operational capabilities,* which are then ranked according to factors such as: the degree of potential improvement offered; the command's sense of urgency; and the ease of incorporating such capabilities into a quick-fix system. The capabilities selected are assimilated into an integrated, time-phased master plan, tailored for evolutionary growth.

Suggested design approaches consider the particular characteristics of data-processing and display techniques as well as the decision-making and other capabilities of the commander and his staff. Other considerations—the wide choice of hardware and software techniques, the problem of optimal allocation of task responsibilities between man and machine—indicate the need for conceiving design alternatives. At least some of these alternatives, along with projections of their impacts on the time, cost and resources drain, should be incorporated as an essential part of the Integrated Design Approach Plan (see Design Spectrum Approach).

Synopsis of downstream design process

The subsequent system design effort is condensed into its more essential aspects at the risk of conveying an oversimplified impression. At this stage, system analysis concerns the identification and comparision of alternative design approaches in order to select, at least tentatively, preferred conceptual designs. This is done in accordance with the operational and functional system requirements previously specified in a first-pass fashion.

Downstream systems analysis, an iterative process (counterclockwise circle, Figure 3), is needed to eliminate alternatives which are economically inefficient and to select, from those remaining, the one which best meets the objectives of performance, operational availability and cost. Trade-offs among hardware/software systems parameters, and among cost, performance, and availability of operational capabilities are made in order to select a preferred approach. The effectiveness criteria derived earlier are critical to the analyst in distinguishing among design alternatives.

The analysis becomes increasingly detailed as the more promising hardware configuration alternatives and their associated software packages are reviewed by designer/user and higher level command managements. As a consensus develops, more sophisticated systems analysis and design verification techniques are applied.*

Design spectrum approach

A discussion of primarily independent design community efforts which focus on determining the appropriate degree of automated support to command information systems is outside the scope of this report. However, a brief synopsis of a proposed design spectrum approach is provided because of the vital contributions it can make to alleviating the user/designer dilemma and to closing the requirements translation gap.

Good design judgment includes careful consideration of alternative approaches/techniques. A broad base of such alternatives would permit the designer to select and apply the right degree of automation as he converts user requirements into hardware/software configurations. The resultant capability would not only be job-tailored, but job-insensitive as well; that is, the capability would be comparatively insensitive to changes in data input/output formats, loads, analyses, etc., and comparatively adaptable to changing job-specific requirements.

Design approaches/techniques, if ordered according to increasing degrees of sophistication, provide a spectrum of design alternatives. From these, the designer

*Specific operational capabilities are defined as the use of a subset of the data base in conjunction with a set of man/machine procedures to accomplish a specific job, such as flight following, report generation, etc.

*For instance, the use of automatic flowcharting techniques such as AUTOSTATE, decision tables, etc. See Van Horn (1964), Doughty (1966), Evans (1965), and Canning (1962).

can select the one with a level of sophistication sufficient to meet

- the more critical user requirements, and/or
- those requirements that are easiest to address, and/or
- those user requirements that can be satisfied in the shortest time period.

As he eliminates those approaches/techniques whose levels of sophistication are insufficient for fulfilling the system's operational and functional requirements, the designer gains design modification insights not attainable when working from a narrower, one-approach design base. The correlation between planning and operations effectiveness (in the sense that cumulative needs are met) and increasingly sophisticated data processing approaches/techniques are shown in Figure 4.

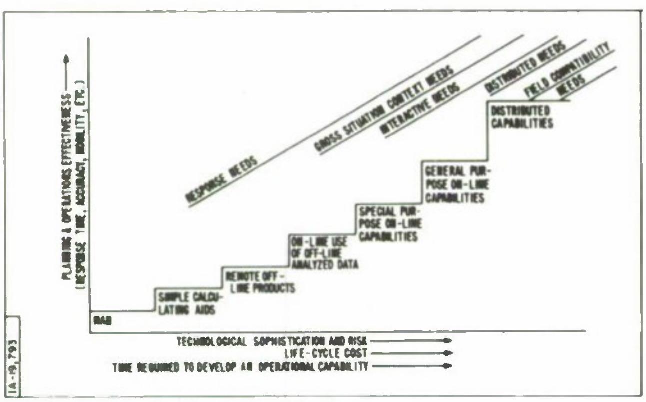


Figure 4—Major factors affecting the correlation of increasingly sophisticated aids with cumulative needs

Several characteristics inherent in the structure of the design spectrum approach increase its attractiveness. First, in applying it, the analyst cannot help but consider the marginal benefits associated with each level of sophistication. This is a natural fall-out of thinking through the added value and associated costs of various degrees of sophistication in deriving a time-phased, cost/effectiveness design approach. Second, the design spectrum approach provides a unique opportunity for gaining some useful and relevant insights for the determination of the most appropriate type of computer support to be implemented ae various command levels for common functional problem areas (McCarn, 1965). Weapon-to-target resource allocation, for example, is a functional problem area common to several levels within the command hierarchy. However, the required response characteristics of the automated aids may vary for each level. Finally, the design spectrum approach, because of its structure, permits consideration of several design solutions to the same functional problem, thereby providing a method for implementation of the quick-fix capability often sought by the user.

The design spectrum approach thus can provide a step in closing the requirements translation gap; moreover, it is a step that can be taken independently by the design community. Use of the design spectrum approach should also stimulate joint efforts for identification of subtle requirements and joint discussions of relevant trade-offs. This collaborative user/designer interaction could lead to the selection of a design approach that embodies the appropriate degree of automation within acceptable performance/cost/availability constraints.

ANALYSIS METHODOLOGY

This section characterizes, in greater detail, the methodological approach reviewed previously. Though the analysis and products are interrelated, they are discussed separately in order to clarify their primary orientation and contribution to overall command information system analysis.

Operational environment analysis

The primary purpose of this analysis is to provide an initial, broad understanding of the operational environment for a particular command's information system. More specifically, the analysis focuses on the command missions and operational environment, major inter- and intra-command interfaces, design constraints, the identification of major command functions, and recommendations for further study.

Mission assessment and operational environment

Initially, a clear-cut understanding of the command's assigned missions should be achieved, along with a broad conception of how the command information system supports the execution of these missions and *why.* An assessment of which command missions will grow in importance and, consequently, which information flows to stress, should also be made. This would require an unsophisticated but nevertheless informed grasp of how factors, such as the following, impact on the command's ability to perform its mission:

a a range of most probable threat scenarios and types of locations (i.e., limited war in Southeast Asia);
b operational concepts and doctrine;
c magnitude and type of force structure/weapons to be employed; and

d allocation of responsibilities among elements of the Unified/Specified command structure.

The type of information available, e.g., planning studies, intelligence estimates, contingency plans, standard operating procedures, and exercise evaluations, assists in providing some basis for the assessment. Data collection and evaluation techniques, including staff interviews, participation in command post, field and testbed simulation exercises, are of value.*

Inter-command organizational environment

A second aspect of the analysis should focus more specifically on identifying and characterizing the functional nature of the major external command relationships and interfaces. The policies and procedures (e.g., JCS Publications) which define the role (e.g., command relationships) of each of the interfacing commands should be understood. In addition, the major input/output documents (e.g., reports, plans, etc.) should be identified, and data relevant to the input and output products exchanged (Kirby, 1964) should be characterized in terms such as quality, quantity, and timeliness.

Intra-command organizational environment

The analysis of major staff activities (i.e., intelligence plans, operations, logistics, and communications) should be assessed (see Figure 5) in terms of type, quality, and actual versus desired timeliness. Simultaneously, a first-pass assessment can be made to determine the type and extent of computer support that should be used to enhance the effectiveness of staff performance as it carries out its assigned responsibilities and tasks. Decriptions of any computer capabilites already in existence within the command (e.g., in support of the command post or other readiness centers) should be included of course, and should stress what operational capabliities exist, or are planned, and why.

Identification of environmental constraints and requirements

While conducting both the inter- and intra-command environmental analyses, certain input/output documents can be identified as design constraints. This identification should be applied to those documents whose content, timeliness, etc., cannot be changed because it is not within the realm of authority of either the user or the designer to do so. However, the inadequacy or tardiness of this information may be critical to a particular staff's effectiveness. If so, steps should be taken to alert appropriate higher level organizations so that problem areas beyond the scope of the current design activity can be resolved.

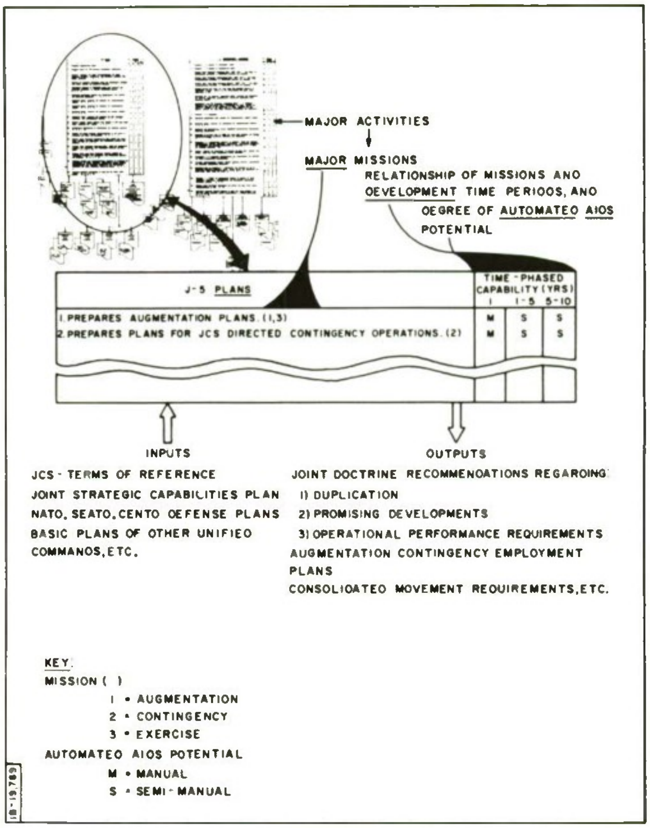


Figure 5—Synopsis of J-staff activity analysis

Political and economic factors should also form part of the constraint and requirements framework. Examples of such considerations include a command's strong desire for a quick-fix capability and/or the immediate need to focus on a particular problem area with a limited amount of available funds and technical staff resources.

Identification and characterization of major command functions and their interrelationships

The key to appropriate follow-on analysis and design resides in a good conceptual identification and characterization of major command functions. A conceptual, functionally oriented model (see Figure 6) should show the various command functions and their interrelationship, and should be derived for various levels. Problem areas can be identified and related to task activities within one command function. Subsequently, an assessment of their impacts can be made on other command

*Sophisticated techniques such as combinatorial analysis and war-games simulation are not much help at this stage in determining the sensitivity of a particular command's ability to execute missions under a variety of threats, using the varied force structure, etc. Communication with various analysts and commanders who have addressed this particular command problem or held similar responsibilities may, however, be of significant value at this time.

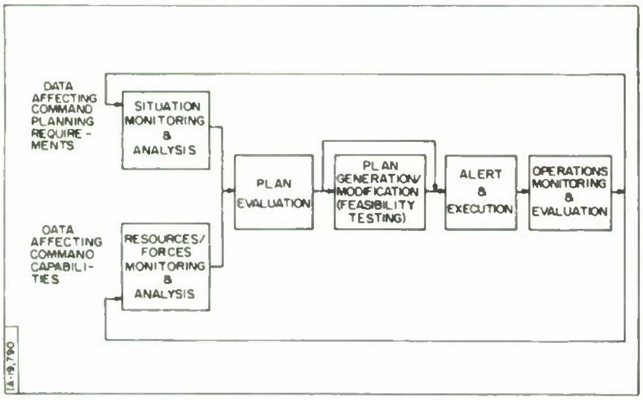


Figure 6—Major command functions and their interrelationships

functions and, in turn, on a particular mission as it proceeds through various stages and time periods. The model also provides a ready grasp of the relative importance of intrastaff activities in support of each command function as the command state-of-alert intensifies over the various crisis-related time periods.

The relative degree of high-priority (computer) support required for mission execution during various time periods is shown in Figure 7, which identifies the major command functions and typical outputs, and grossly indicates their interrelationship and relative importance as the command state-of-alert intensifies. The effectiveness of the command and its information system is extremely sensitive to the efficient execution of a number of major command functions which cut across external and internal lines of staff organization. Furthermore, it is dependent upon the integrated and simultaneous execution of those functions which require high-priority support during various stages of command alert.

These can be best exemplified by comparing (see Figure 7) command functions 4 and 5 with 2 and 3. The first set (4 and 5) does not require any priority support during a cold war or routine state-of-readiness; however, it does have a very high-priority support requirement as a specific contingency approaches and as a subsequent plan is executed. The second set (2 and 3) has a continously high-priority support requirement associated with its timely execution, regardless of the command and state-of-alert.

In regard to operational plan generation/modification, note that, during a routine state-of-alert, a high-priority value is placed on a number of globally dispersed contingency plans. As a specific contingency or a related set of contingencies ("hot plans") appear to be developing, planning attention focuses on the modification of specific crisis plans.

Meaningful priority support requires first that more become known about the type and volume of intrastaff information exchanged in support of the command functions, and next about priority activities within a specific command function. When these requirements are known, data-base size, file structure, and the type

IA-19,194

COMMAND FUNCTIONS (TYPICAL OUTPUTS)	DEGREE OF PRIORITY SUPPORT REQUIRED		
	COLD WAR	SPECIFIC CONTINGENCY APPROACHING	CONTINGENCY RESPONSE BEING EXECUTED
1 a) PLAN GENERATION-ALL THEATRES b) SPECIFIC PLAN MODIFICATION - "HOT PLANS" (OPERATIONS PLANS)	HIGH LOW	LOW HIGH	LOW HIGH
2 SITUATION MONITORING (INTELLIGENCE REPORTS)	HIGH	HIGH	HIGH
3 FORCE / RESOURCE MONITORING (FORCE READINESS REPORTS)	HIGH	HIGH	HIGH
4 ALERT & EXECUTION (OPERATIONS ORDERS)	NONE	HIGH	HIGH
5 OPERATIONS MONITORING/EVALUATION (FORCE STATUS, DAMAGE ASSESSMENT REPORTS)	NONE	HIGH	HIGH

Figure 7—Relation of major command functions to the need for priority support

of work station configuration (hardware/software) can be estimated.

Problem characterization and recommendations

One final aspect applies to the Operational Environment Analysis; namely, that it should characterize those problems identified to date and recommend further areas for analysis.

Functional execution analysis

Function Execution Analysis focuses first and in depth on the dynamic flow of information related to a particular command function, such as the plan generation/modification function throughout, say, the execution of a command's augmentation support mission. Iterative and more refined analyses around those activity sequences (sub-functions) which relate to major problem areas should be subsequently analyzed.

Identification of critical information flows

The partitioning, in a conceptually useful way, of critical information flows associated with each major command function is essential in an overall analysis of command structures (McCarn, 1965; Gotto, etal. 1962). In studying command information systems which emphasize the broad problems of planning, a useful initial step is to achieve an understanding of the hierarchial planning-oriented information flows within the total command structure. With this objective in mind, Figures 8, 9, and 10 were prepared to show, in several levels of iteration, the primary ingredients of a Function Execution Summary document—the main product of a Function Execution Analysis.

A static representation of the family of planning documents, i.e., plan generation/modification inputs and outputs, was first developed (see Figure 8). The sequential interrelationship of the basic planning documents among the commands involved in the operational plan generation process is highlighted. The flow of documents from and to superior and subordinate commands is depicted, and a key is included to explain the presentation of this type of flow diagram.

Next, a simplified (Figure 9) and then a detailed (Figure 10) representation of time-sequenced analyses of the critical information flow throughout the command structure were developed. Based on the initial inter- and intra-analyses of the command structure, basic milestones in the generation of a contingency plan can be established within the command hierarchy (see Figure 9). Timing estimates for various scenarios and force structures can be obtained, and both routine and flap generation times indicated.

The flow process shown in Figure 10 represents a simplified version of the flowchart actually developed. Nevertheless, it should provide the reader with sufficient perspective to see that this type of chart can be used by the system designer to achieve a better understanding of how a computer-augmented command system capability could be developed to serve the needs of crisis-oriented plan generation/modification. The level of detail actually approached is dependent upon

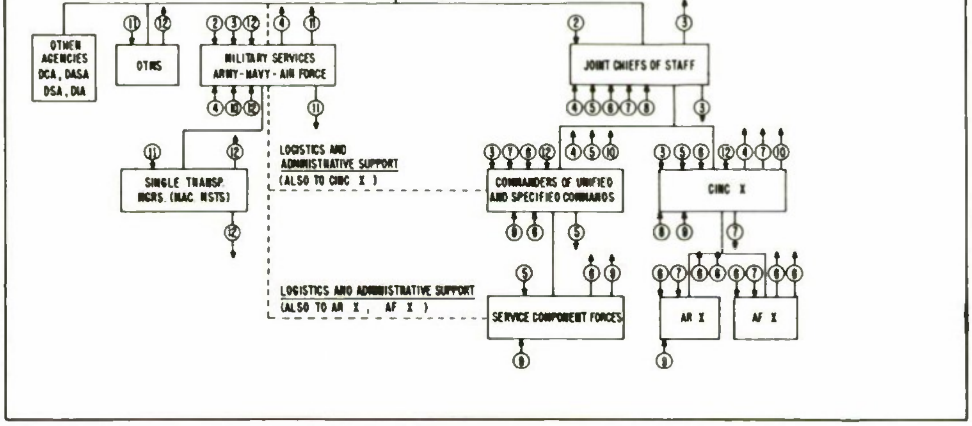


Figure 8—Joint planning organizational structure and basic planning documents flow

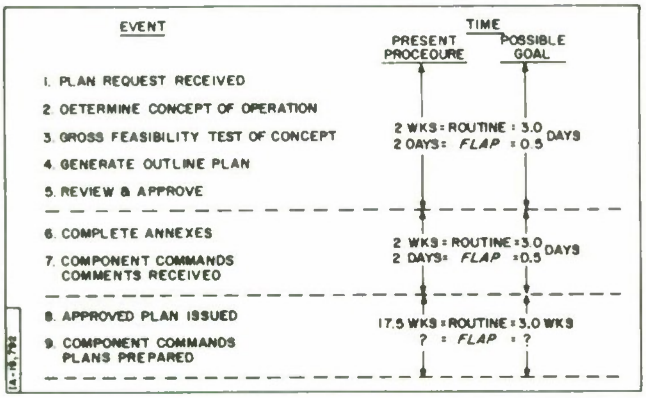


Figure 9—Planning time (illustrative) of plan generation process

such factors as the overall importance of the command function, its growth potential, and time resources available for analysis. Input/output documents of data used by a particular command staff (CINC X staff was emphasized in Figure 10) in performing activities or making decisions related to the development of operational plans must be described in detail.

The matrix approach to mapping the planning process flow

A row is devoted to each critical J-staff shown in Figure 10. Similarly, critical delay times associated with a specific stage in the planning process can be identified in a column bounded by two milestones, and can be related to specific staff/command activities earmarked for reduction. This matrix approach (Figure 10) to mapping the planning process flow enables the analyst to expeditiously isolate those command-wide (between columns) and/or specific staff (individual row) activities which are associated with planning problem areas (e.g., cause of large delay times and inaccuracies).

The same matrix would be useful in pinpointing the assistance a third-generation distributed computer capability could provide in making the staffing process more responsive. The critical flow path (heavy lines in Figure 10) can be studied to determine the nature of the personalized data base, the type of work station configuration, and the type of data management capability the staff must use throughout the plan generation/modification process in order to make it more effective. Each staff element must maintain a sufficient overall context, which requires on-line interaction with a dynamically changing data base, as it assists the commander in defining and reacting the ambiguous situation.

From this emphasis (see Figure 10) on using the contextual representation as a framework for more detailed analyses, time-shared access priorities and

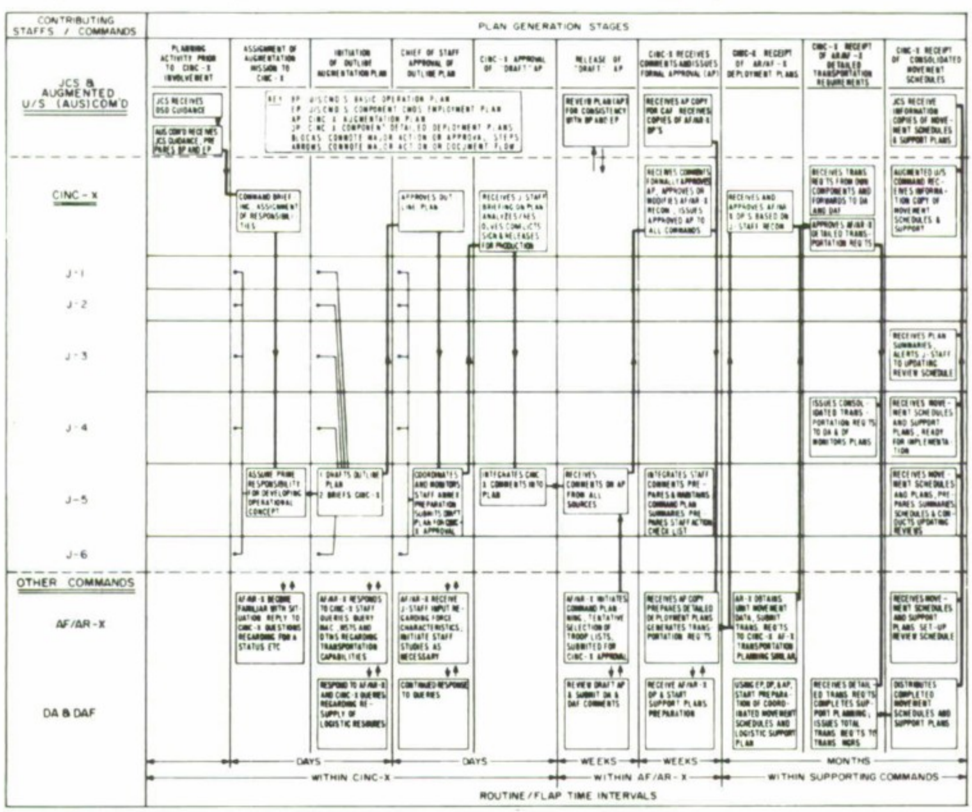


Figure 10—Preparation of an augmentation/contingency plan: CINC X command example

software applications packages can be derived. The computer support which does exist is not readily accessible (usually a single, large computer-complex, man/machine interface) nor sufficiently responsive (primarily off-line batch processing).

Study of a diagram similar to that shown in Figure 10 revealed that, as planned modification time grows increasingly restricted, fewer plan options were able to be considered prior to the selection of a tentative concept of operation, and further feasibility testing became more difficult. The diagram was used to assist in isolating the deployment feasibility problem area and in characterizing the conceptual design of computer-supported operational capability that is generally useful to a number of commands.

Identification of planning problem areas within a specific command

Once a broad appreciation of the hierarchical planning process and its present limitations has been attained, identification/analysis of the information system within a particular command (*i.e.*, CINC X and/or its component commands, AF X or AR X) must be made. First, several levels of staff activity related to each of the major command functions should be sequentially traced and characterized by a simplified representation of the kind shown in Figure 11. The resources available and a growing appreciation of where the major problem areas lie might initially limit the Integrated Functional Analysis. Instead, an in-depth Functional Analysis can be conducted and focus on one of the major planned command functions, such as plan generation/modification.

This kind of information flow analysis will identify a number of planning problem areas (*e.g.*, force tailoring, deployment feasibility analysis, etc.) which could be aided by data-processing techniques.* These were designated as problem areas because it became clear that critical planning effectiveness criteria, such as response time and the accuracy of plan feasibility testing, were not being satisfied to the extent desired by the command. Problem areas of this kind usually have four characteristics in common.

- At present, they require a significant amount of time to accomplish, thus delaying plan generation/modification.
- They cause inaccurate plans to be prepared, resulting in: over/under commitment of force packages containing an inappropriate mix of combat capabilities, inappropriate time-phased

*Once problem areas have been synthesized and man/machine task allocations developed as part of this System Synthesis effort, they are referred to as man/machine-oriented capabilities (see Figure 11).

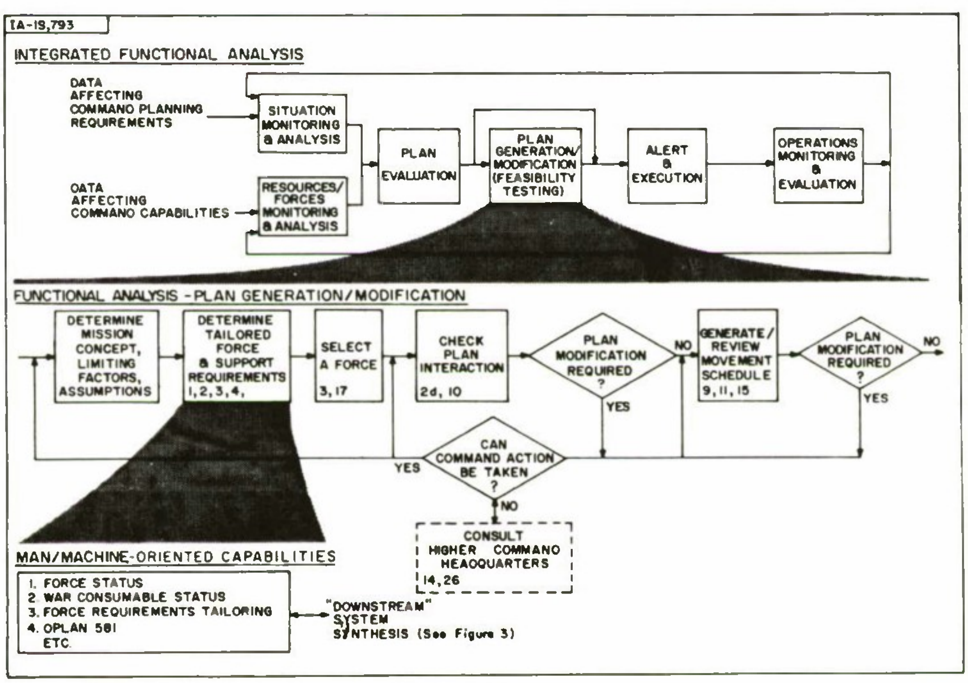


Figure 11—Process relationship between command functions and capabilities

arrival of force packages, and off-load base saturation, etc.

- They can be resolved, to some extent, by automated data retrieval, storage, and processing techniques.
- They can be resolved, or at leased lessened, because CINC X design organizations have sufficient authority to take some corrective action.

Identification of these problem areas provides additional insight and understanding to the systems designer and indicates a range of potential applications (simple to complex) which must be considered in the development of integrated planning capabilities. These problem areas form a key basis for the determination of the specific man/machine operational capabilities, which are subsequently derived (see Figure 11, System Synthesis) and included in an Integrated Design Approach Plan (see Figure 3). An integrated ranking of applicable problem areas (see page 258, Systems Synthesis) derived from the preparation of other command functions Execution Summaries (see Figure 3), should, if possible, be performed prior to the submission of the Integrated Design Approach Plan.

This conceptually useful, iteratively analyzed, partitioning process is essential to the characterization of *functional* command information system requirements. It is especially attractive because it can be tailored to the level of analysis effort available and to the desired degree of improvement that the command foresees as necessary. This methodological approach should contribute to the elimination of unbalanced requirements for command information systems.†

Derivation of functional effectiveness criteria

During the course of performng the operational and functional analyses, which start with a comprehensive understanding of the various missions and growth trends, the systems analyst should concurrently derive functionally oriented effectiveness criteria (Jacobs, 1965). This process is illustrated for the plan generation/modification function (see Figure 12). In essence, the process is one of initially identifying major problem areas which, upon reflection, are indicative of major limitations in the present planning process. Subsequently, each of these basic limitations must be defined in an operationally meaningful sense to the command under study. Simplified and easy-to-observe indices of criteria satisfaction, such as time to produce an outline plan during routine/flap conditions, are identified.*

†Hobbs and Craig (1964) noted this imbalance in discussing the nature of the specification problem: "The best that can be expected from such a specification is optimized subsystem design because of the fairly well defined black box elements of the system. However, some optimization might not even

*The analyst is advised to initially focus on a few criteria, using easy-to-observe indices of the degree to which the present planning process is satisfied. Downstream in the design process (Figure 3, Refined Systems Analysis and Verification), additional criteria can be defined and more sophisticated indices/measures of criteria satisfaction can be included. The additional evaluation comprehensiveness at this stage is warranted during the test of well-defined, specific, potentially high payoff, man/machine capabilities. See Evans (1965).

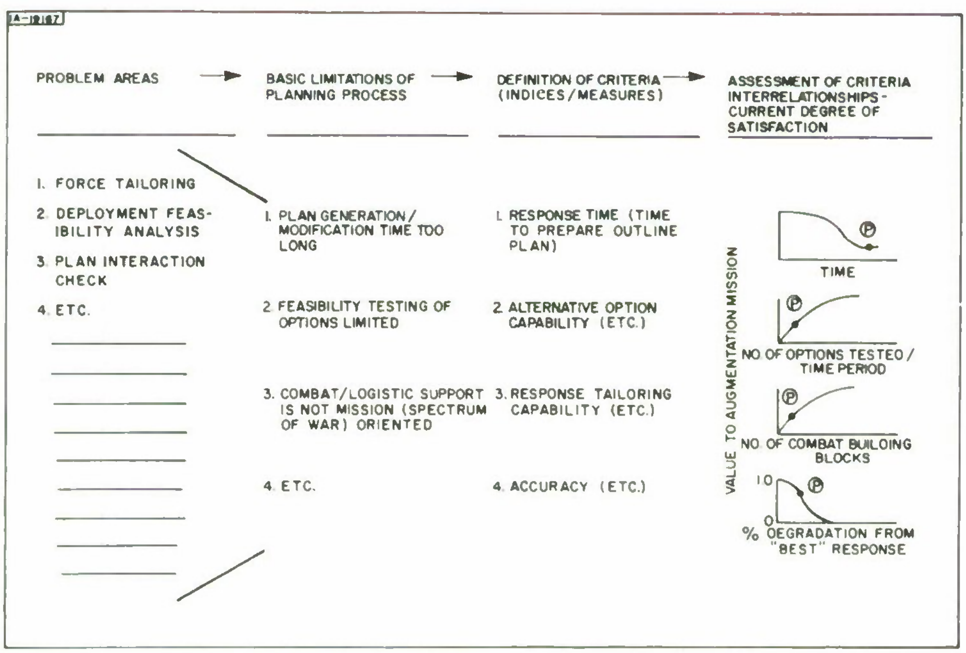


Figure 12—Derivation of functional effectiveness criteria

Finally, the systems analyst must assess the relative importance and the interrelationships of each of the criteria. The P in Figure 12 shows the degree to which a particular command's planning process satisfies several criteria. The understanding of criteria interrelationship is extremely critical in the conception of an improved planning process. For example, increasing the satisfaction of one criterion, such as planning response time, may conflict with the also desirable increased satisfaction of another criterion, such as the importance of considering an increased number of planned options before deciding to feasibility-test a particular option in greater detail. Lewis (1964) touched on this problem when he said:

> *Let us consider a few of the contributing criteria which would, if we knew how, add up to being "right." Let's take time. It is* always *a Good Thing to save time. There are circumstances in which it would appear that being able to do something quickly is of extreme value. Unfortunately, however, the value of time saved cannot be quantified, even though it may mean the difference between one catastrophe and another. Although one has this impression of the value of it, one cannot say how much it is worth, how much should be paid for it.*

Thus, data-processing support must be supplied to the planning process in such a way as to improve, overall *net* planning effectiveness and not just one criterion, such as planning speed. The analyst must make sure that broad mission improvement objectives (which focus on an integrated improvement of all command functions) are satisfied, and not just planning objectives. A broad perspective of how computers can assist in all command functions (*i.e,* the command mission in general) should be determined from similar analyses of other command functions.

Consideration must be given to the additional follow-on problem of communicating extensive plan modifications through subordinate commands in the hierarchy which have to understand and implement this changed guidance in the form of operational orders.

Maintenance of the requirements baseline

The importance of maintaining a requirements baseline is perhaps best stressed by discussing the impact of some unique command system characteristics. Any subsequent indication that the established requirements need to be changed should be immediately assessed to assure appropriate redirection of the design process. Mission growth, obsolescence, and operational constraints as affected, for instance, by shifts in the allocation of contingency and general war responsibilities within the Unified/Specified command structure, or by revised estimates of enemy capability and intentions, can have a dominant influence on the projected information requirements and desired set of operational capabilities.

Periodic undating, resulting in the issuance of a Command Analysis Summary (see Figure 3), should be conducted to reflect the impact of these changes on functional design and on the alternative design approaches selected. In addition, the time, cost, and performance data being generated during the iterative refinement of design alternatives (counterclockwise loop, Figure 3) may also provide good reasons for modifying official statements of system requirements (*e.g.,* design performance estimates were too optimistic).

Impact of unique command information system characteristics on the maintenance of the requirements baseline

Command information systems, unlike the more commonly understood weapons systems requirements analysis, have a number of unique characteristics which make their requirements analysis job not only more complex but of continuous importance to the design job.

The first characteristic emphasizes the fact that man—the decision-maker—is an integral part of the command system and not simply a user, as in the case of a weapons system. The job of analyzing human decision-making requires a knowledge of the social sciences as well as of the physical sciences, and a greater familiarity with the job being aided by the system. Consequently, requirements analysis is more complex.

Second, command information systems cannot be bought off-line and in batches because they are an integral part of the "nerve" center of an operating on-line command. They must be acquired in an evolutionary manner; that is, an increased capability must be phased into the existing system without disrupting its present operational capability. Thus, requirements must be *continually* analyzed and the implications of changed requirements immediately assessed in order to effect efficient control of the design process.

Third, the missions and capabilities of the many other commands and their command information systems, which interface with the prime command being analyzed, are also continually changing. This fact further complicates requirements analysis and necessitates the constant surveillance of integration requirements.

In short, the maintenance of the requirements baseline is complex and of continuous importance throughout the operational life of the command and its command information system.

EVALUATION OF METHODOLOGY

Advantages

An attractive feature of the methodological approach presented in this report is its broad applicability to all phases of the command system development process and to the development of management information systems. The characterization of major command functions and specific characteristics of information flows will, of course, differ for various commands, depending upon mission responsibilities. However, the conceptual framework provided and the characterization of the various types of requirements translation analyses should be useful from the point of view of providing a running start, especially to the systems analyst with limited experience and/or familiarity with the command hierarchy.

A second important feature is the demonstration, through illustrative material, of the iterative, broad-and-general to partitioned-and-detailed analytical approach that can be used in the mapping of command information flows. This approach allows the systems analyst to maintain a broad and balanced perspective (*i.e.*, prevents "catastrophic fractionization" of the *systems analyst's* context). Its application should result in a balanced specification of requirements, an in-increased probability of identifying major problem areas and, simultaneously, improved assessment of their relative importance, as well as a more realistic derivation of effectiveness criteria.

The approach is also flexible in the sense that it can be tailored to the command's improvements objectives, as well as to the funds and qualified technical resources available. Several kinds of analysis products, such as those illustrated, were actually used to provide a context for evaluation of a command (CINC X). They also assisted in providing a context for useful conceptual design approaches that led to a significant improvement in the command's ability to rapidly tailor joint forces and assess the feasibility of various deployment options (*before* assigning and committing a particular force package). Because its value has been demonstrated and proven, the methodological approach merits consideration for development and further application.

Limitations

The major limitation to this approach is that it remains more of a guide than an easy-to-apply analysis manual because intelligent application of the methodology requires broad skills and experienced judgment.* A possible pitfall to the use of this type of approach is that the analyst could become preoccupied with the flowcharting *per se*, leading to user/designer frustration and disinterest, overemphasized concentration on the *existing* system, and to *more* flowcharting, which is very sensitive to alternative "if" flows at a detailed level. This pitfall can be avoided if the analyst concentrates on the following questions:

- *why* does the command structure exist?
- *where* does it need improvement?
- *how* can it be improved?

and on the following objective: Emphasis on critical flows—flows directly related *to identifiable problem areas* and *to the actual/desired outputs*.

Another limiting aspect is that this paper only contains a description of the initial step toward the comprehensive methodology required; integration with other approaches and extensive synthesis must be further studied. The development of a comprehensive, teachable methodology (which could expedite the effective assimilation of computer technology in the command structure) depends upon a broad-to-specific synthesis which uses an historically based perspective in assimilating various updated ingredients (see Figure 13). Without a proper historical perspective, the analyst cannot project changes in the operational environment (which impact on mission trends) and relate to them the technology and tools which *will become available* for problem identification and solution. As the need for methodology is recognized and its comprehensiveness increased anu taught, the probability of exploiting the potential of a third-generation computer technology to improve the command and control structures will be significantly increased.

A quick survey of the literature indicates that if approaches to a methodological framework exist, it is only in the minds of a relatively few elite analysts.

---

*The nature of the skills required and the belief that they can be taught has been given credence by others. Parsons and Perry (1965), in summarizing Davis (1964), feel that "though not a science, the type of skill required involves some combination of *technology* and *art* which would require methods of teaching more closely related to generating synthetic experience than to instilling abstract knowledge." McCarn (1965) alludes to additional skills which are required: "Almost inevitably most of the personnel operating command systems believe they are performing adequately. The definition of what can't be done which ought to be done is usually viewed as unwelcome criticism. It is not surprising that good functional requirements go undefined. The only requirements that really get well defined are those where a command wants to duplicate the efforts of another command or organization from which it has not received responsive service it thinks it could provide itself. . . . The control and coordination of innovation in a structure like the chain-of-command is a complex and difficult problem."

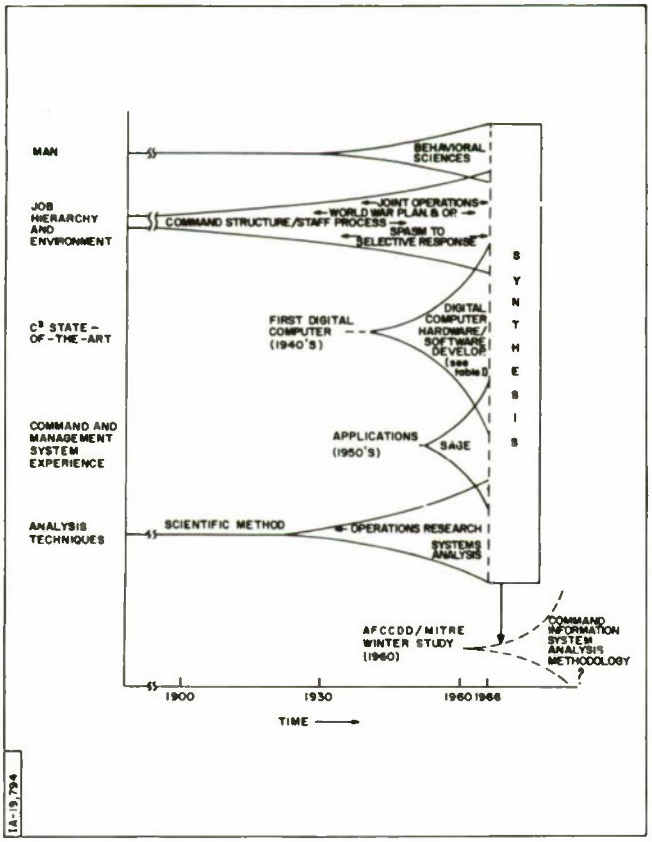


Figure 13—Synthesis ingredients required for development of a comprehensive methodology

Such information must be made available *today* to assist the military/designer community in evaluating how computers can be used *tomorrow* to improve the command information structure.

CONCLUSIONS AND RECOMMENDATIONS

Conclusions

The analysis methodology and the gap

The methodology discussed in this paper can assist in evolving toward a comprehensive understanding of the command and control structure as it relates to the information system network. Development of this methodology will contribute to a better understanding of the "job," hierarchy, and environment, and a better characterization of the operational and functional requirements. This will, in turn, allow more effective assimilation of computer technology in the command and control structure.

In so doing, the methodology can assist in closing the requirements translation gap—a gap which has produced vague operational concepts and resulted in the use of arbitrary design criteria. The contributions toward providing a comprehensive methodology as discussed in this report consist of an overall analysis framework, characterization of the specific types of analyses and products needed, and use of illustrative materials.

Command information system network

Until the hierarchical command information system network is *widely* understood by the user/designer communities, and until the impacts of important and dynamic recent changes on the command structure become an integrated part of the design and evaluation context, costly and time-consuming errors will continue to be compounded.* Some of the problem areas requiring assessment and solution, and examples of their impacts on the command and control structure, are listed below.

- Implementation of a selective response in the nuclear age.
  Impact: *more* information, from *more* sources, *more* carefully and rapidly assessed, in order to support development/evaluation of multiple options, *more* rapid replanning, etc.
- Maintenance of global commitments which will use, to an increasing extent, U. S.-based combat resources, tailored with greater precision and deployed with greater speed.
  Impact: intensified emphasis on accelerated staffing, replanning, etc.
- Recent changes in responsibilities with the Unified/Specified command structure, including the creation of a new Unified command.
  Impact: revised mission trends, new information sources and reporting systems, etc.
- Emphasis on reshaping doctrine and tactics to increase the effectiveness and mobility of joint operations.
  Impact: more complex planning, increased intercommand interface problems, etc.
- Emphasis on centralizing control as well as planning responsibilities, by-passing, in some instances, intermediate command levels.
  Impact: input source distortion and delays, fractionization of context, etc.
- Intensified need to respond swiftly to fleeting targets of opportunity.

*The urgency associated with developing such a methodology was, perhaps, dramatically underscored in a recent article, when, in discussing a specific command and control system, the officer interviewed stated that its value as a "deterrent power is unquestionable, that it will take 10 years and over $400 million to add it to the inventory can very well be questioned." Cited as the main problems: a lack of clear-cut development responsibility and improper definition of user requirements. See Armed Forces Management (1966).

Impact: more rapid data summation and more flexible staffing at field headquarters level and below, etc.

The nature of these problems must be more fully understood if the increased computer support to higher level commands and initial computer support to field headquarters are to be effective.

Comprehensive command system requirements

Specific command operational and functional requirements do not exist, although they have important and subtle impacts on *effective* use of computer technology. As a result, there is a marked tendency to gravitate toward computer-centered command and control systems which have disrupted the age-tested staffing process in unanticipated ways. These factors have contributed to command frustration, to less than effective use of computer support, and to costly and time-consuming redesign effort.* Nevertheless, commands continue to acquire larger, as well as additional, computers to support individual staff-oriented readiness centers. They are also trying to assimilate computer-supported training capabilities for which information requirements have not been completely determined.

Adaptation of third-generation computer technology

Third-generation computer technology utilizes significantly improved display-oriented, man/machine interface techniques. Because these computers can be packaged into smaller, lightweight, more rugged configurations, they permit distribution of digital data management, analysis, and reporting capabilities to key staff elements throughout all levels of the command structure. This type of computer capability holds potential for reversing trend toward "catastrophic fractionization of the commander's context," and for significantly improving the responsiveness of the command and control structure—once the requirements translation gap is closed.

Recommendations

Development of analysis methodology

Concerted efforts must be exerted to provide the user/designer/management communities with a more comprehensive design evaluation and control context. An analysis methodology is required to assist in providing an improved understanding of the command information system network and in deriving meaningful operational and functional requirements for specific commands. This task must be accomplished first if we are to design for an effective assimilation of third-generation technology—effective in the sense that the command information system network will be less sensitive to information volume overloading and more rapidly adaptable to staffing assignments at all command levels.

An instructional package, which synthesizes and extends the existing applicable analysis effort, must be prepared. This package, complete with methodology and case study experience, could be used in training qualified blue-suit and civilian personnel for follow-on, on-site responsibilities.

Design spectrum approach to developing insights for the generation of requirements and design modification

A design spectrum approach to the development of a range of data-processing techniques which address particular functional problem areas (McCarn, 1965) should be undertaken. By presenting users with a variety of increasingly sophisticated techniques for solving common command problems (e.g., force tailoring, report generation, etc.), unanticipated requirements/constraints can be discovered and design modification insights obtained. Furthermore, chances are increased of developing, in a shorter time, an operational capability to address a specific user's requirement.

Since these design spectrum and methodological approaches complement each other, their development should be carried out simultaneously. The primary responsibility for implementing a spectrum approach lies with the designer, while the responsibility for translating system requirements belongs to the user. The implementation of the design spectrum approach, which can be aided by an improved definition of operational and functional requirements, will provide a better basis for user/designer interaction. This joint effort will expedite the conception and application of useful operational capabilities necessary to the achievement of a more responsive command information system.

ACKNOWLEDGMENTS

While the responsibility for the contents as presented rests solely with the author, the insights that led to the synthesis of the methodology could not have been acquired without significant contributions by members of the user/designer communities. The author is especially indebted to the staffs of JCS, USSTRICOM, Hq. USAF, TAC, USAFE, and PACAF for operational insights related to the evolutionary development of

*The article (*Ibid.*) goes on to say, "While there were about 60 . . . liaison officers in the . . . plant . . . where the system was built, assembled, and tested, they could not provide adequate information on what the user really wanted. Time and again scope changes were allowed as . . . 'requirements' changed. . . . And each time this happened," said an officer close to the program, "it cost more money and more time."

their command systems. I am also indebted to a number of my associates at The MITRE Corporation for encouragement, comments and helpful criticism, and to Mrs. Patricia Chatta for assistance in structuring and editing the paper.

REFERENCES

1 E. M. BENNETT, E. C. HAINES, and J. K. SUMMERS *AESOP: A prototype for on-line user control of organization data storage, retrieval and processing.* In AFIPS Conference Proceedings: 1965 Fall Joint Computer Conference Vol. 27, Part 1 Washington, D. C.: Spartan Books, Inc. 1965

2 H. D. BENINGTON *Command and control for selective response.* In K. Knorr and T. Read (Eds.) Limited Strategic War New York: Frederick A. Praeger 1962.

3 R. G. CANNING (in conjunction with Auerbach's Standard EDP Reports Editorial Staff) *Selection procedure —EDP systems analysis techniques.* Auerbach Corp. and BNA, Inc. 1962

4 L. S. CHRISTIE and M. G. KROGER *Information processing for military command.* Datamation 8, 58-61 1962

5 C. W. CHURCHMAN, R. L. ACKOFF and E. L. ARNOFF *Introduction to operations research: part 2, the problem.* New York: John Wiley and Sons 1957

6 R. M. DAVIS *Military information systems design techniques.* In E. M. Bennett, J. Degan and J. Spiegel (Eds.) Military Information Systems New York: Frederick A. Praeger, Inc. 19-28 1964

7 J. M. DOUGHTY *The AESOP testbed.* In D. E. Walker (Ed.) Information System Science and Technology: Papers Prepared for the Third Congress Washington, D.C.: Thompson Book Co. 1967

8 N. P. EDWARDS *What does commander need is still an unanswered question in R & D.* Armed Forces Management 11, 78-82 1963 and 1965

9 J. EVANS *Analysis of a testbed simulation approach to the evaluation of operational planning.* Bedford, Massachusetts: The MITRE Corporation MTP-25 (Prepared under Contract AF 19(628)-5165) December 1965

10 O. T. GATTO, C. B. McGUIRE and R. L. VAN HORN *Research aspects of command and control.* The RAND Corporation RM-3155-PR August 1962

11 G. B. GOULD, JR. *Command control and communications system.* Signal XX, 70-71, 79 1966

12 R. H. GREGORY and R. L. VAN HORN *Automatic data-processing systems.* Belmont, California: Wadsworth Publishing Company, Inc. (second printing) 1964

13 L. J. HOBBS and L. J. CRAIG *The specification and selection of military information processing systems.* Presented at the 1964 National Winter Convention on Military Electronics 5 February 1964

14 *How not to build C&C systems is still an unanswered question in defense.* Armed Forces Management 12, 111-112 1966

15 Institute for Defense Analyses, Res. & Eng. Support Division *Computers in command and control* TR-61-12 2, 20 November 1961

16 J. F. JACOBS *Practical evaluation of command and control systems.* Bedford, Massachusetts: The MITRE Corporation MTP-7 (Prepared under Contract AF 19(628)-5165) November 1965

17 R. L. KIRBY *Integration of military information systems.* In E. M. Bennett, J. Degan and J. Spiegel (Eds.) Military Information Systems New York: Frederick A. Praeger, Inc. 134-146 1964

18 J. H. LEWIS *Evaluation of effectiveness.* Fourteenth in the Information Command and Control Series of Lectures sponsored by the University of California 13 January 1964

19 J. C. R. LICKLIDER and W. E. CLARK *On-line man-computer communication.* In AFIPS Conference Proceedings: 1962 Spring Joint Computer Conference Vol. 21 Washington, D.C.: Spartan Books, Inc. 1962

20 D. B. McCARN *Large-scale system design techniques* In J. Spiegel and D. Walker (Eds.) Information System Sciences: Proceedings of the Second Congress Washington, D.C.: Spartan Books, Inc. 1965

21 J. L. McLUCAS *Excerpts from an address to NEREM members during their annual meeting* Delivered 3 November 1966

22 L. B. MOORE *How to manage improvement.* Harvard Business Review 36, 75-84 1958

23 O. MORGENSTERN *The command and control structure.* Research Memorandum No. 47 Princeton, New Jersey: Princeton University 15 October 1962

24 H. H. PARSONS and W. E. PERRY *Concepts for command and control systems.* Falls Church, Virginia: System Development Corporation TM-WD-227/000/00 (Prepared under Contract Nonr 4923(00), available through DDC.) December 1965

25 *Planning for data processing* EDP Analyzer 4, 3 1966

26 E. S. QUADE *Pitfalls in systems analysis.* In E. S. Quade (Ed.) Analysis for Military Systems Santa Monica, California: The RAND Corporation 1964

27 M. D. TAYLOR *The uncertain trumpet.* New York: Harper & Sons, Inc. 1960

28 F. B. THOMPSON *The fractionization of the military content.* In AFIPS Conference Proceedings: 1964 Spring Joint Computer Conference Vol. 25 Washington, D.C.: Spartan Books, Inc. (a) 1964

29 F. B. THOMPSON *Fundamentals underlying military information system design techniques.* In E. M. Bennett, J. Degan and J. Spiegel (Eds.) Military Information Systems New York: Frederick A. Praeger, Inc. 47-87 (b) 1964

30 R. L. VAN HORN *Formal structures for information system design.* Presented at First Congress on the Information System Sciences ESD-TR-63-474-11 May 1964

31 *Where defense should improve command/control development* Armed Forces Management 9, 32-33 1963

COMMAND CONTROL SYSTEMS FIELD EXPERIMENTATION

RICHARD M. LONGMIRE, *Chairman*

Session Participants:

WALTER LESIW
ERVIN KAPOS
EUGENE P. VISCO

Introductory Remarks

RICHARD M. LONGMIRE
Research Analysis Corporation
McLean, Virginia

Dr. Enthoven* has said,

"*. . . the state of our analysis of the effectiveness of large combat units involving a mix of weapons and mobility is still inadequate. . . . We are quite certain that the available measures of effectiveness such as firepower index do not account properly for several important attributes of such forces . . . if one wanted to make a case that it is not possible to measure the effectiveness of Army divisions in any meaningful way, one doubtless could do so convincingly. But the problem is, whether we like it or not, we have to have such a measure. . . . This measure is essential not only to permit realistic force comparisons and determination of force requirements, but also to allow us to estimate the incremental contributions of new weapons or new organizational concepts.*

"*Can we develop the kind of theory of the effectiveness of Army divisions that is required?*

"*Among the approaches that need to be pursued are field exercises and tests, map exercises, war games, and other simulations, and detailed empirical study of the history of war.*

"*We need studies directed not at the derivation of broad abstractions like the so-called 'principles of war,' but at empirical propositions with precise quantitative content.*"

The Army, Navy, and Air Force appreciate the need for "field exercises and tests" to obtain quantitative empirical data on command control systems which support large operating forces like an Army division.

The major Army tactical command control data system under development is ADSAF (Automatic Data Systems Within the Army in the Field). ADSAF is to be composed of three systems: TOS (Tactical Operations System), CS_3 (Combat Service Support System) and TACFIRE (Tactical Fire Direction System). TOS will support the intelligence, operations, and fire support coordination functions. CS_3 will perform personnel, administrative, logistics and other support functions. TACFIRE is to be used for tactical and technical control of field artillery. ADSAF is being considered for employment at echelons from battalion through Army level.

The Army has undertaken a four-year experiment in the Seventh Army environment in Germany to determine the Army-wide requirements for automation of the Tactical Operations System. In addition, an extensive program of testing will be conducted at Fort Hood, Texas, to determine the automation requirements for the Combat Service Support System. These two major efforts follow many years of Army field tests with tactical data systems experimental hardware and software. Included in the series of tests were: tests, beginning in the early 1950's, to evaluate computer support of field artillery functions leading to the development of the FADAC (Field Artillery Digital Automatic Computer) System, which is now in operational use, and to the requirements for the follow-on TACFIRE System; the First Intelligence Simulative Test (FIST), a division level test in 1962 to evaluate several concepts in intelligence data processing; Exercise Major Domo, the 1964 troop test of a Field Army level automated tactical operations center concept, designated the AN/MSQ-19.

*Dr. A. Enthoven, Assistant Secretary of Defense for Systems Analysis, "Cost-Effectiveness Analysis of Army Divisions," 30 March 1965, Part I Proceedings of US Army Operations Research Symposium.

The present emphasis in Army field tests of tactical data systems (ADSAF) is on the development of user requirements in the operating environment of the user by representatives of both the user (e.g., Seventh Army for TOS) and the developer (Automatic Data Field Systems Command). One of the conclusions of the 1966 Working Conference on Army Tactical Data Systems* was: "Hardware, software, procedures, and people are so interrelated in accomplishing a functional process that it is difficult to determine the user's requirements for a process in any environment where the interrelationship cannot be observed."

Even though the Army has been deeply involved in field tests for some time it is clear that much additional work is required to develop an adequate technology to support the design for, conduct of, analysis of data from, field tests of tactical data systems. It is also difficult to translate such field test data into requirements for tactical data systems which can be supported on a cost-effectiveness basis. The Research Analysis Corporation is providing the Army support in this regard.

*The findings of the Working Conference are being reviewed by the Automatic Data Field Systems Command and have not as yet been endorsed by the Army.

The Working Conference also concluded, "The major unsolved problems in the development of military information systems appear to be in the areas of system design technology and field testing." It is believed that the extensive Army tactical data system field experimentation now underway will provide major contributions to the development of a suitable technology for field experimentation. Perhaps these contributions will be presented at the next Congress.

The major paper in this Chapter discusses Air Force field experiments and tests of command control systems. This paper by Walter Lesiw on the NORAD COC work provides a rather detailed history of testing as a part of the command control system development process. He recommends a representative but separate facility for testing; highlights the importance of documenting plans for experimentation; identifies the need for development of simulation technology early in the program; states the requirement for at least one system engineer to "trouble shoot" during the test to provide corrective action as required, and for a simulation supervisor to insure credibility of the simulation; and supports the phased approach to system evolution, including the development of functional programs in increments which are later integrated.

Field Experiments and system tests in NORAD COC development*

by Walter Lesiw
The MITRE Corporation
Colorado Springs, Colorado

INTRODUCTION

This paper describes developmental activites associated with design and evolution of the Headquarters, North American Air Defense Command Combat Operations Center (NORAD COC). The Center is situated in a hardened site, referred to as the NORAD Cheyenne Mountain Complex (NCMC), southwest of Colorado Springs, Colorado. Of initial concern in this review of developmental events is an examination of the program of experiments, exercises, demonstrations, and tests employed in achieving an operable integrated system. A further objective is to arrive at some conclusions regarding the utility of exploratory experiments and systematic testing in evolving large-scale information systems of the type discussed.

General system characteristics and situational factors affecting development

Evolution of the Cheyenne Mountain system was appropriately preceded by systematic analysis and documentation of organizational functions and information handling observed in an antecedent "manual" system. The analysis provided a basis for preparation of design specifications for a "new" system which would automatically process, retrieve, and selectively display the large volume of data received at the COC.

Early planning for system acquisition included decisions to: establish an interim facility for assembling the system incrementally; initiate operations for experimentation purposes; and, prepare the system elements for installation and test operations at the hard site. The process was conceptualized and documented in the form of discrete developmental steps (phases), including a guide for conducting experiments using the configuration of equipment, programmed logic, and personnel assembled in an experimental facility. For convenience the original manual system was designated Group I (Chart A), the experimental facility Group II, and the COC scheduled for Cheyenne Mountain, Group III. Group II was visualized as an evolutionary link between I and III.

Equipment installation, facility preparation, program shakedown, operator personnel indoctrination, and experiment planning occupied the initial weeks of the experimental phase. During the early months of experimentation a separate study of the proposed Cheyenne Mountain Complex was undertaken by NORAD at the request of the Office of the Secretary of Defense. The study resulted in a recommendation for a more austere system, a compressed development schedule, and the quickest possible transition from Group I to Group III operational capabilities. New program directives called for an early end to experimental activities and the rapid preparation of plans for implementing a program of Category II System Development Test and Evaluation (Dept. A.F., 1963) in Group II, to be completed in Group III following occupancy.

As a consequence of these program changes experimental emphasis gave way to activities which were to be clearly operationally oriented, e.g., operator and maintenance procedures development and simulated operational employment. The Group II experimental facility, still fragmented as a system since elements of hardware, programs, and personnel were yet to be delivered, and after somewhat less than a year of experimental utilization, became a vehicle for operational development and terminal tests. This new phase of "system testing" was carried out under formally prepared Category II Test Plans.

*The effort described in this paper was accomplished under Contract number AF19(628)2390, managed by the Cheyenne Mountain Complex Management Office, Electronic Systems Division, Air Force Systems Command. The views expressed here do not necessarily constitute a position agreed to by the contracting agency.

Chart A. Facilities, Phases, Programs[1]

Group I	Group II				Group III		
NORAD COC-Soft Site (Manual System)	Experimental Facility				NORAD COC-Hard Site (Cheyenne Mountain Complex)		
← Analysis →	← Experimentation & Demonstration →				← Category II System Tests →		
	Phase C	Phase C-1	Phase D	SOSP[2]	Phase E[3]		
			(Cancelled)	(Transition)	1-E	2-E	3-E

Explanatory Notes:

1. Phase designations correspond to program package changes (Phase E in 3 increments)
2. SOSP—Simulated Operational Support Plan; a stop-gap measure during translation to Phase E.
3. Category II system testing of all Phase E program increments occurred in Group III. The final package, 3-E, was employed in Group III.

The developmental system

A system construct involving Group I, Group II and Group III as interacting elements, each having an integrity of its own, is useful for placing the total developmental effort in perspective. Group I, the existing soft site Combat Operations Center provided a point of departure, a history of precedents and procedures and a philosophy of operation of fundamental importance to experimental utilization of Group II. When merged with the new physical and functional capabilities introduced in Group II, these became the basis for establishing concepts of operations and plans for demonstrations and evaluation of Group III. In this larger context it is appropriate to view Group III not as an entirely new system but rather a combination of new design elements and formidable remnants of a previous configuration which had to be functionally integrated.

The experimental facility was assembled to represent the NCOC rather than to provide an operational prototype. Consequently, the experimental command post appeared as a single dais instead of a 3-tiered structure, while divisional areas interacting with the command post were simply clusters of display consoles functionally analogous to divisional design. The command post came closest to duplicating its operational counterpart because it served as a facility mock-up during dais layout design. On the other hand, communications installed in Group II exhibited least fidelity due to delays in establishing network requirements and developing procurement specifications.

Experimental system environment

The experimental facility was arranged to represent the working areas of five functional organizations constituting the Combat Operations Center. These information users were concerned with the performance of command post, surveillance, tactics and weapons, reconstitution planning, and communications and electronics tasks. Each activity was manned by one or more operator positions. A simulation room was established for centralized management and conduct of experimental operations.

Equipment complex

The data processing subsystem for the experimental phase consisted of a Philco Model 211 CDP with a 32K core memory storage unit, an interim low-density magnetic drum system, and 12 magnetic tape units for on-line processing. The data display subsystem included 15 display consoles, each having a keyboard hardcopy unit. A Display Data Controller provided interface between CDP and display consoles.

Communications in the form of 30-button Call Directors were installed at each operator position. Five also were located in the simulation room, and one at the Test Engineers position in the computer

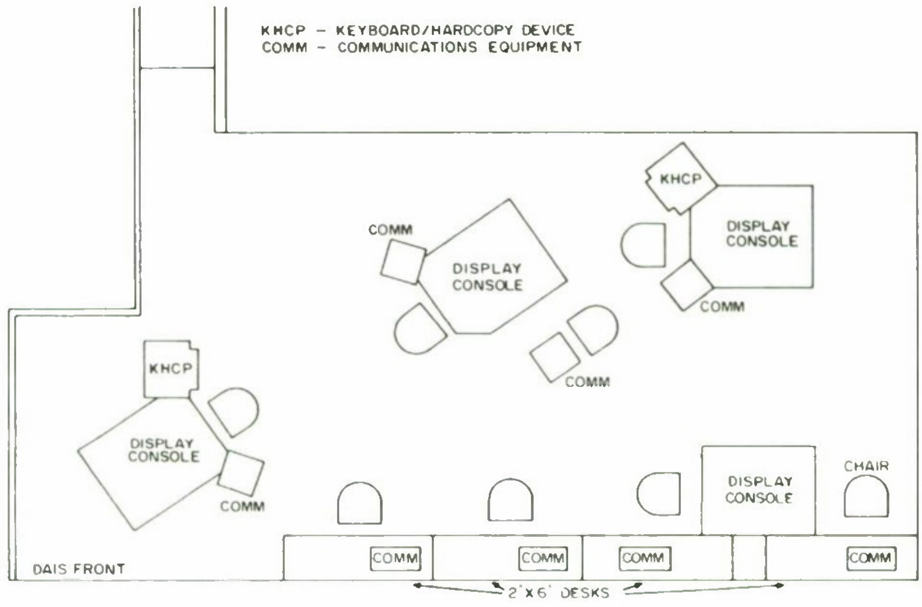


Figure 1—Equipment placement in modified command post, Group II

area. Seven positions were equipped with hands-free speaker phones. Each position had access to the dial system, a public address system, and every other position by push button action. One hundred and ten full-period internal circuits were employed. All external source inputs were simulated.

Midway through the experimental phase, a Philco Model 212 CDP with improved core memory unit reliability was installed to support processing tasks not requiring the drum system. As a consequence, processes for the generation of simulated data and reduction of recorded data ran two to three times faster, faciliting a more rapid rate of system testing. Shortly thereafter, a high density drum was installed to operate with the 212 CDP in support of processing tasks necessitating drums. This change brought about a significant reduction in operator request response times (computer interrogations) and a man/machine interface which better resembled later developmental stages.

During initial experimentation the command post was utilized as a facility for mock-up analysis of dais display consoles, communications components, and general layout of battle staff workspace. Consequently it was not immediately available for routine experimental system operations. However, an interim command post in the form of a three console cluster was substituted briefly. Recommendations for configuration of the Group III Command Post, based upon the mock-up analysis and dais study, were incorporated in Group III facility specifications. Console layout and communications (Figure 1) for the experimental Command Post tests, were also patterned after the mock-up. Divisional placement for experimentation is shown in Figure 2.

Two major display subsystems, the Large Wall Display for the command post and the closed circuit

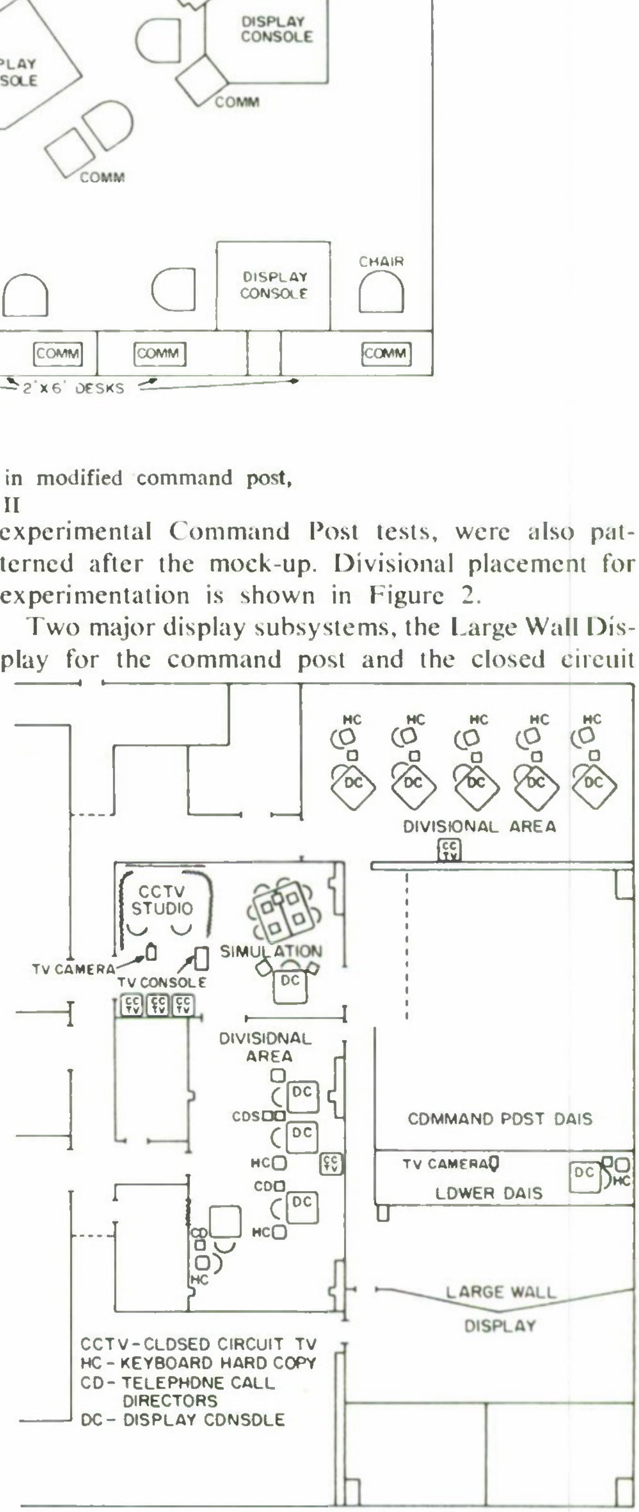


Figure 2—Equipment placement for divisional testing, Group II

television network for the COC, were not available during the initial 12 months of system experimentation. However, the importance of the large display to command post and system tests, and the feasibility of substituting a console for large wall display activity, made it expedient to simulate control and utilization of the wall display by assigning its functions to one of the display surfaces of the 3-console cluster described earlier as the interim command post. Absence of color, reduced screen size, and console presentation of the display were the deviations introduced by the substitution. Controls and functions were unchanged. This adaptability of consoles to changes in functional assignment and physical location made experimental reconfigurations of the facility relatively simple and rapid.

Figure 3 illustrates the Group II equipment complex essentially as prescribed by the NORAD CMC Study Group. Among Study recommendations was the directive that this interim configuration would provide a development agency and the user with a capability to test equipment, programs, and procedures in an "operational environment" prior to installation of the system in the hardened CMC facility; also, that this would be done in increments based upon prescribed delivery of progressive program modules.

Program system

System experimentation in Group II was paced by the development of three computer program systems —utility programs, operational programs, and support programs. Initial experimental plans were largely concerned with determing the operability of the system, particularly in terms of program-computer-operator interactions, and with establishing a stable test system which could support planned experiments. Program packages (Chart A) were identified by corresponding test phases—Phase C, Phase C-1, a stop-gap step for changeover (SOSP), and Phase E.

Phase C programs included the utility system used in the production, testing, and maintenance of the operational and support programs. Utility system capabilities directly supportive of system testing were those of Environment Generation used to pre-load the data base and control tables, and, System Tape Read-In to "start up" a test. The operational program system consisted of the group of programs providing capabilities prescribed by operational specifications for the Phase C system. Support programs consisted of a general NORAD environment simulation subsystem, simulated message analysis and conversion subsystem, and data reduction and analysis subsystem. Of these support subsystems, the first is a group of JOVIAL coded programs esigned to automatically generate simulated input data for the 425L system (NORAD COC) and to operate in conjunction with the second. The second is designed to produce simulation input tapes compatible with the system of operational programs. The third subsystem is a group of programs that provides for data reduction. Phase C programs retained their identity through succeeding phases of program evolution and are represented in the current system.

Phase C-1 programs were the result of cancellation of a Phase D system which would have entailed elab-

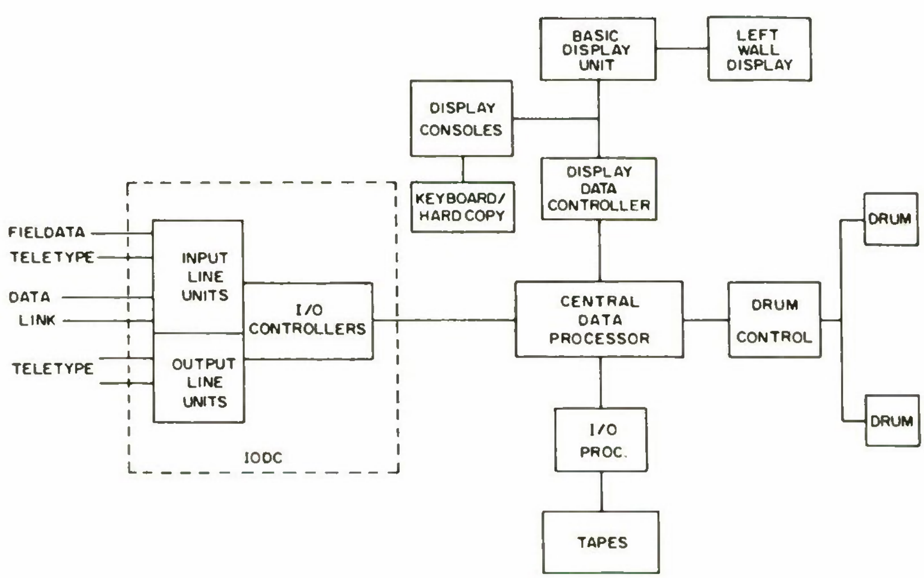


Figure 3—Equipment configuration (Simplex System)

orate changes to C. The alternative was an upgrading of Phase C by the addition of a new function and a new capability plus some organizational changes which together constituted Phase C-1. After three months of testing with the C-1 programs, the experimental effort was interrupted by the NORAD Study Group's decision to initiate development and test of the final Phase E program package in three increments, 1E, 2E and 3E.

The first increment of the Phase E program was characterized by the absence of a live interface between consoles and operators and an incomplete capacity for processing externally generated messages. Basic display capabilities were observable through the use of program-simulated console actions. Increment 2E permitted operator interaction with the computer whereas computer-generated outputs were not available. Input processing of three important data categories were yet to be included. The final increment accounted for the processing of all specified message inputs and the generation of required outputs to recipient agencies. A full complement of man/machine interactions was also provided. System tests were supported by the 3E program through the remainder of Group II operations (less than 5 months).

By way of contrast, the Phase E system of programs, after redirection by the NORAD Study Group, differed in two significant respects from the earlier systems. First, the automated support of a number of COC functions was made relatively austere. Computer processing of several categories of mission-related data was eliminated and plans for sophisticated program interaction with an elaborate COC organizational function were dropped, although these capabilities had been introduced in the Phase C-1 program design. Secondly, the rationale of program sequencing and scheduling was revised to permit more efficient utilization of central data processing capacity, providing a more rapid response to input demands. Each of these differences from C-1 had a significant effect upon the design and execution of system tests that followed.

Manning

System experiments and tests were planned as integrated team operations. Military personnel were assigned the responsibility of manning operator positions while supervision and support of the test system was accomplished by contractors. Programming, test design and test implementation were also contractor supplied. The development and application of simulation technology fitted to continuously changing experimental and test needs were contractor tasks with progressive participation by military personnel.

Management of the system acquisition and test program was carried out by an agency of the Air Force Systems Command* with the close collaboration of NORAD, the ultimate system user.

The personnel configuration defining operator positions and console allocations underwent several changes in the course of system evolution. Some were a function of test experience, others were occasioned by reorganization directives originating within NORAD. These changes affected not only the assignment of consoles to positions but the general composition of the COC-command post, divisional organizations, and affiliated (interfacing) agencies. The number of system positions manned during experimental activities was not limited to the set of consoles employed. An initial experimental plan called for a team of three operators manning a single functional area, whereas the final demonstration of system performance included 24 operator positions distributed among all elements of the COC including battle staff personnel situated on the command dais.

Developmental procedures

The requirement for procedures specification embraces a wide spectrum of activities characterizing the system development process. It is submitted at this point that the task of evolving procedures for operating and maintaining a newly configured system is an much a design problem as the conception and creation of the equipment and programs otherwise defining the system. However, it is recognized that more often than not procedures are developed on-the-spot as an expedient to "get the system to work." Acknowledging the fact that procedures are at least in part derivatives of the functional attributes of the equipment and program mix, an examination of these was made in order to establish initial operator tasks at the divisional level and subsequently at the command post level.

Additional procedures were necessary for conducting system operations. Among these were the experimentation procedures necessary to achieve objectives specified in planning documents, also the simulation procedures for representing the environmental factors and conditions impinging upon the system in operation. No less important were the procedures for collecting, recording and reducing experimental and test data as well as those procedures for integrating interaction between the COC and affiliated centers and agencies collocated in the Cheyenne Mountain Complex. The conduct and management of the system development program, in general,

*The implementing agency was referred to variously as: Detachment No. 10, Field Test Force, and Cheyenne Mountain Complex Management Office, all of the Electronic Systems Division, AFSC.

required establishment of documentation and reporting procedures as did the initial experiments and interim development activities.

Simulation technology

Earliest plans for experimentation included requirements for a Simulation Group and specifications for "realistic simulation of the system environment, including the insertion of impromptu simulated 'events' as a challenge to system operators during an experiment." The system of simulation was expected to possess capabilities for: external inputs from existing NORAD systems; intra-system inputs such as simulated console switch actions; equipment simulation; time compression and expansion, including time-marked magnetic tapes for skipping blocks of tape recorded simulated events; representations of threatening and stressful environments, and a full range of variations within and among these.

Relationships between experiment design and simulation techniques were considered in a preview of general methods for the generation and utilization of simulated data for Phase C experimentation. Simulation prerequisites to effective experiment design were stated in the form of three sequentially related functions:

- Specification of the simulated environment,
- Preparation of the simulation materials, and
- Control of simulation data during the experiment.

The first of these was expected to be a part of detailed experimental plans in a form sufficient to guide personnel responsible for producing the simulation materials of the second. The third function was viewed as a management problem for a simulation supervisor.

These relatively simple conceptions of simulation were superseded after a period of experimentation during which progressively larger segments of the system were integrated and exercised until all functional organizations were participating. The result was a system approach to simulation problem concepts and implementation techniques the scope of which amounted to a technology for the support of various aspects of system development—design, experiment, exercise, test, training, demonstration and evaluation.

With all system functions beginning to operate synchronously, and simulation techniques oriented toward performance generation on a system level, it was appropriate to construct a scenario representing hypothetical war conditions which would stress mission performance capabilities and capacities of the system in the time period it was scheduled to become operational. The product was a system test scenario which contained the necessary attack elements and information handling properties required to initiate preparation of a simulation package for a complete system exercise. These conditions provided the framework for development of the simulation technology which was finally employed.

Steps in the design and production of a simulation problem now included:

- Definition of the rationale of the problem scenario;
- Compilation of the major events—sequential event listing;
- Preparation of briefings—events of relevance preceding the exercise;
- Detailed specifications of the simulation materials—time factors, simulation aids, voice scripts, etc., and
- Problem production and checkout.

The specifications were used to generate punched card decks which in turn were used to produce a simulation tape. Card deck listings were also employed to produce the scripts provided to the voice simulators. Checkout consisted of a time and content comparison of simulation tape, an initial conditions tape, the briefing materials, and the voice script, with final verification accomplished by a dry-run of the exercise.

Problem packages from the simulation library consisted of the input message card decks, the voice script card deck, and problem folders. The folders, organized separately for the Chief Simulator, the simulators, and the observers (discussed under data recording) contained event listings, reference maps, voice scripts, background material, and operational briefings. Fourteen problem packages were eventually on file in the Simulation Library.

An important aspect of the library is a Master Card File which contains, in modular form, sets of message input cards prepunched for variations of at least 10 important message types. The card sets are cross-indexed to identify correlated messages (related in time or by virtue of other common factors). The Master Card File allows rapid and efficient construction of new problems by reducing the requirement for scripting new message cards. Variable fields in the standard messages of the card file are left blank to facilitate application to new problems.

Key positions for the control of simulation operations were those of the Chief Simulator and the Test Engineer. The Chief Simulator position was frequently occupied by the test designer since his familiarity with problem details and questions not covered by the simulation materials contributed to maintenance of exercise continuity. The position of Texas Engineer was established to place an agent, familiar with test objectives and requirements and total system operations, at the computer to assist on questions of system startup, program loops, exercise aborts, recoveries from equipment failures, and compilations of test data.

Simulation techniques were also utilized in special test situations other than those involving complete system operations. In equipment testing and in certain computer program tests, teletype punched-paper tapes were used. These tapes were developed by recording selected live data transmissions or by manual punching of desired simulations of real transmissions on paper tape. The technique permitted effective test activities which lead to compatible interaction among the local equipment/program complex, communications facilities, and distant message processing installations.

Experimental and test data recording

Initial guides for recording of experimental data included provisions for manual recording by assigned observers and automatic recording using a variety of devices. Recording requirements were to be specified in detail by experimental and test plans and would cover data content, recording mode, recording duration, periodicity of records, and any unusual techniques likely to affect scheduling of experiments.

Subsequent plans specified a data recording system involving computer program techniques, photographic methods, magnetic voice tape techniques, and observer commentary. The computer program would provide a capability for recording tables, registers and other content specified by the experiment designer. In addition, recording at some or all of the following points would be obtainable:

- Before or after operation of some specified program or programs;
- At specified intervals during operation of some specified program or programs; and
- At set intervals independent of program operation.

Magnetic voice-tapes were to record, with time marks, the following classes of telephone calls:

- Calls between any two tactical positions (operators);
- Calls between any tactical position and any agency, real or simulated, which is external to the System; and
- Calls by the Experiment Supervisor via the tactical telephone network to any position or station.

Observer commentaries, recorded manually at a station or operating position, would account for significant data impractical or impossible to obtain via alternative techniques. Instructions in the form of observer test procedures were developed which included orientation statements, a review of formatted data collection sheets, and guides for reporting results.

Recording techniques specified for system performance measurements during Category II tests involved on-line data recording of three types:

- Hardcopy—each output to keyboard hardcopy devices constituted a record available for analysis;
- Modular recording—each program module recorded, in fixed format, data regarding the timing of its operator and its communication with its input buffers and other programs modules. The content of external input messages and operator actions was recordable by this method. The selection of modules for recording was under operator control; and
- Assembly test recording—intended primarily as a program checkout tool, assembly test recording provided a flexible method of recording any portion of the computer data base at any desired time. However, it could not be utilized while modular recording was in process and was not intended as a routine recording technique.

In addition, system performance measurement involved recordings of off-line programs concerned with generation of simulated data for use with the operational system since these contained simulated input traffic histories. Also of interest were line unit recordings consisting of data flow at input/output line units recorded via magnetic tape equipment, teletype page printers, and teletype tape reperforators. Routine equipment maintenance records assembled for compiling system availability data also consituted an element of the recording regimen.

Experiment and test plans

The series of experimentation and test plans prepared during the period May 1962-December 1965, clearly reflect the discontinuity of the developmental program. However, the methods of approach proposed in successive plans demonstrate a strong thread of consistency particularly with regard to the structuring and supervision of experiments and the concept of integrating system operations as a prerequisite to conducting meaningful experiments and tests. Initial plans for Phase C, were of an experimental character in that they emphasized a period of functional analysis and system familiarization as well as a time for assembling a body of experimental data to determine design modifications and shape advanced test planning. The system, newly configured in the Group II facility, was viewed as a "breadboard" or working model to support controlled, laboratory-like studies.

Subsequent plans for Phase C-1, proposed a continuation of experimental activities and the beginning of mission-oriented system testing, as well as mock-up and design analysis activities directed at establishing a command post which would be integral with the Phase C configuration. Following a brief period of implementation, Phase C-1 plans were supplanted by a proposal for a transitional period of Group II utilization during which the system would be modified to resemble, in performance, the configuration recommended for operational development by the NORAD Study Group.

Final plans* called for implementation of a program of Category II—System Development Test and Evaluation (Dept. AF, 1963) that would culminate in an operational system which conformed to requirements established by the NORAD Study Group. Thirteen areas of testing were specified and the program spanned Group II and Group III system configurations. The test plan terminated upon satisfactory demonstration of system performance.

The overlapping sequence of plans and phases (Chart A) included:

- Phase C Experimentation;
- Phase C-1 Experimentation, Mission-Oriented Tests, and Mock-up Analysis;
- Simulated Operational Support Tests and Command Post Operations;
- Category II System Development Test and Evaluation:
  Group II Experimental Facility,
  Group III Cheyenne Mountain COC, and
- System Performance Demonstration and Turnover.

Experiment and Test Objectives

Objectives for each of the phases represented in the cited sequence were stated in corresponding plans. Basic objectives of the experimental program were: to revise and augment preliminary positional handbooks and task analyses, on the basis of controlled experimentation, in order to provide the designer with data required in design of the Group III system; and, to evaluate system performance from an operational point of view and determine whether the design met military requirements.

A subsequent statement of Phase C objectives specified main and subordinate elements. The former committed all experiments to the task of evaluating the Phase C "package," i.e., determine how the 425L System (NORAD COC) could be improved to better meet the requirements of the NORAD Mission. This was to have been realized by achieving subordinate objectives of:

- Proving the capability of the test bed (operability and compatibility of elements);
- Familiarizing evaluation personnel with the Phase C system during its assembly and checkout;
- Providing design information for use in Group III design by analysis of the initial Phase C system, also new, and/or modified system designs;
- Developing standard operating procedures in coordination with the operator/user;
- Collecting performance data on the elements of the Phase C system;
- Building a background of simulation capability, operator knowledge and experimental techniques prior to more sophisticated levels of experimentation;
- Developing operator task descriptions, personnel evaluation criteria and training procedures for use in later phases, and
- Developing detailed operational performance criteria for use in Category II and III testing of later phases.

The terminal statement of Phase C objectives included the following priority:

- Operate the Phase C experimental model;
- Accomplish design improvements on the model;
- Establish performance criteria based upon operation of the system;
- Develop operating procedures in conjunction with user/operator personnel;
- Develop techniques for Category Testing;
- Develop training materials for operators and other personnel, and
- Develop exercise and evaluation techniques.

Phase C-1 was conceived as the second stage of developmental testing to be carried out in the Group II facility. It was to involve not only the assessment of pre-specified hardware and programs applied to simulated defense situations but the immediate "on site" design and test of procedures as well. A goal of this phase was the determination of command post layout, equipment needs, personnel procedures, and displays for Group III. Specific test objectives were concerned with development of empirical data describing Group II performance and demonstration of mission handling capabilities of the COC configuration as represented. In decreasing priority, the specifics of interest were:

1. Derivation and documentation of the performance attributes of Group II by conducting tests which permitted observation and description of posi-

*The program of experimentation and tests was considered by the System Program Office (SPO) to fall under the general heading of Category II testing. A plan to this effect was drawn up in February 1963, and superseded by a subsequent plan in January 1965.

tion functions, division operations and command post utilization of 425L information resources and automation capabilities in carrying out NORAD COC responsibilities.

2. Based upon test experience, derivation of a summary profile of system performance variables and, under standardized test conditions, collection of sample measures of performance in order to develop quantitative criteria for performance assessment and mission-oriented demonstrations.

3. Demonstration and assessment of system execution of NORAD mission responsibilities employing the C-1, Group II, configuration under representative conditions of COC operation.

4. Specification and documentation of operational procedures for C-1 Group II.

5. Design detailing: Identification of design problems at any level of system operation and recommendations for remedial measures (hardware and facility modification, procedural and personnel changes, program and document alterations, et.al.).

6. Preparation of a test guide for Group III testing.

Five of the objectives were augmented with matrices blocking out the design approach. Each matrix represented a generalized data collection form containing grouped variables, design conditions, special requirements, and probable observations. Command post tests, on the other hand, were expected to involve more than experimental analysis and test, e.g., mock-ups, command dais rearrangements, and display innovations. Consequently objectives for these tests were separately documented with the prescription to:

1. Establish a basis for command post test operations by specifying techniques and procedures to be employed by each battle staff position in performing mission tasks using Group II capabilities—displays, machine stored data, programmed data handling, simulated problems, and divisional resources.

2. Specify the characteristics of command essential displays and develop procedures for their preparation and presentation under varying conditions of system operations.

3. Demonstrate integrated command post operations involving full command staff participation, interaction with lower echelons, supporting activities and eternal agencies, under varying defense conditions based upon performance data obtained under 1 and 2.

4. Assess the adequacy of Group II communications—network and end instruments—for supporting command post operations during test (requirements and preliminary assessment criteria were supplied).

5. Determine detail design requirements including modification implications based upon command post test observations and analysis.

These objectives represented a revision to a previous list which addressed Phase C-1. The change entailed an initial response to the redirection proposed by the CMC Task Force implying a shorter, intense test effort with emphasis upon operational procedures development and practice in system utilization so as to facilitate transition from a Group II test complex to demonstrated operational capability in the hard site.

The Simulated Operational Support Plan, intended to fill the gap between Phase C-1 experimentation/tests and the initiation of Phase E tests under a formal Category II test plan, proposed "simulation" of the Phase E programs as well as reconfiguration of the Group II test system after the hardware and organizational concepts recommended by the CMC Task Force. Primary objectives were concerned with:

- Providing a means for implementing the command post development program;
- Providing an environment for operator procedure development for application to Phase E;
- Developing a stable system configuration, describing it, and documenting the corresponding operator procedures, and
- Developing a demonstration capability that would provide basic system introductions to user/operator personnel.

Secondary objectives for SOSP were to:

- Continue development of new simulation techniques that would be applicable to Phase E problem design; also, explore various techniques in supplementary simulation;
- Establish observer criteria and data acquisition forms suitable to Phase E testing, and
- Establish a convenient reference source for simulation data.

The Category II Test Plan, prepared under Air Force Regulation 80-14, constituted the most conventional approach to testing taken during the system development process. It accounted for the terminal phase of testing although covering a greater interval of time than all preceding experimentation and test activities combined. Group II and Group III tests were included as well as the three increments of the Phase E program package.

The prime purpose of the plan was to "ascertain that the NORAD COC operational functions and characteristics are substantially in accordance with the requirements specified in the CMC Task Force Study Report and other system control documents." Phase E System Testing is of major interest because it is concerned with the observation and assessment of complete system functions. The goal of this system test plan was to bring the system into being as specified, to operate and improve it under test conditions, and to provide an observational basis for developer/user judgments regarding the acceptability of demonstrated capabilities for performing mission tasks. Objectives for achieving the goal were:

- Verification of the physical characteristics and functional capabilities of the system as these were defined in requirement documents;
- Measurement of system performance of those functions which were susceptible to available measurement techniques, in order to provide baseline performance data at turnover for the information of the using operating commands;
- Identification of system changes which were implementable within authorized resources and schedules of the system test program, and adequate documentation of change recommendations beyond the scope of developer resources;
- Provision of familiarization with the test system and experience in manning the test system under simulated conditions of COC operations for command/operator personnel;
- Development, exercise, and improvement of operating procedures, to include normal operations, system degradation, recovery, and manual backup;
- Demonstration of system operations under variable conditions of readiness for handling live and simulated inputs with the express purpose of briefing personnel and agencies having a need for knowledge of system configuration and test progress, and
- Preparation for system performance demonstration including a description of the "as delivered" configuration.

Detailed test plans, objectives and schedules as well as implementation methods and procedures were stated in Test Order documents. These in turn were reduced to manageable Mission Orders which established specific objectives and test conditions under which implementation occurred. Nine Test Orders and a plan for testing systems interfacing with the NORAD COC were prepared, supplemented by approximately 23 Mission Orders.

The plan for System Performance Demonstration (SPERD) also appeared as an Appendix to the Category II Test Plan. Performance demonstration was to be accomplished through a command post exercise patterned after "problems" of a type employed previously in Group I and Group II systems. Some Category II test controls for managing system tests were to be withdrawn so that the system was "at the disposal" of the user/operator in the performance of mission tasks. Essential SPERD objectives were to:

- Provide NORAD with an opportunity to operate and observe the Basic NORAD COC in an environment free of test constraints so that a determination could be made of the degree to which the system had achieved readiness for air defense operations, and
- Conclude Category II testing and support system turnover plans.

Documentation

The developmental program was supported by a continuous interest in keeping all aspects of developmental activity suitably documented. This held for broad planning documents, system specifications, test orders and subordinate test procedures, test design and methodology, data collection and result reporting, and phase reviews.

Of particular concern in the present context are those documents which established a framework for the conduct of experimental and test activities. The initial documents were primarily concerned with broad guidelines for undertaking studies in a laboratory-like environment. They entailed conceptual estimates of what the system would consist of and how it would operate. Experimentation was visualized as an exploratory and evolving process which contributed significant data to advanced planning and final system design.

Second order documents included phase experimentation plans, special test orders, and the test plans required under Air Force Regulation 80-14.

The bulk of documentation, after plans, consisted of individual test reports, phase summary and evaluation reports, design analyses, and design recommendations at all system levels.

Experiments, tests, demonstrations and exercises

To this point it has been necessary to develop a structure for the review of experiments, tests, demonstrations and exercises conducted in evolving the system. Consequently attention was given to the developmental setting within which these activities occurred; some of the physical and functional characteristics of the experimental system; several methodological considerations, and the plans and objectives developed for conducting experiments, tests, demonstrations and exercises.

The term experimentation has not been used in the classical sense which would imply discretely manipulable variables and laboratory conditions of control.

However, it does pertain to the exploratory activities concerned with establishing and utilizing an experimental tool capable of performing functions and being manipulated so that observations regarding its immediate character, and predictions and recommendations concerning its subsequent character, were possible. Consequently experimentation in this instance refers to the activities carried out during the Phase C and C-1 interval, whereas testing refers to the array of tasks and actions scheduled under the Category II Test Plan. Indoctrination and preparatory activities, typified by SOSP and SPERD, as well as certain test orders and training sessions, fell under the headings of demonstration and exercise.

Experimentation

Each objective specified for Phase C experimentation was addressed during the experimental series. Of 188 experimental change recommendations processed and approved during this phase, 40 were adopted for Phase E design. On the basis of Phase C experimentation, it was judged that 425L as planned for the initial operational system would improve NORAD COC operation. Of those items on which experience had been gained in Group II, many met NORAD requirements, some needed elimination or modification, and some were not clearly assessable. Significant system findings in summary were:

- The display capability of the experimental system was a measurable improvement over the Group I operation. Timely distribution of data and rapid retrieval of information by operators was facilitated.
- Operating procedures needed improvement in order to exploit system capabilities.
- The potential for higher system capacity (over Group I) was clear particularly in the volume and variety of information that could be maintained and displayed for command utilization. More timely and accurate data could be placed at the disposal of the command organization.
- Response time in system handling of certain critical events was not equal to Group I capability but data processing design changes were expected to improve this condition.
- Some processing and correlation aids demonstrated in the experimental system provided capabilities not achievable in the manual system.
- Positional organization and allocation of operator responsibilities were in need of clearer definition.
- Details of display manipulation and data base content would require some changes as system operator-task relationships were completed.

An additional result of Phase C activity was the command post mock-up design, fabrication and review which concluded with NORAD approval of the design and instructions to proceed with specifications for Group III procurement and modification of the Group II command post for experimentation and test.

Of the 188 experimental change recommendations occurring during Phase C, equipment accounted for 30, general program and documentation 36, functional area program and documentation 99, support and utility programs 7, facility-command post 3, operator methods and procedures 6, and system-general 7.

Objectives for Phase C-1 experimentation were not fully realized because test activities under C-1 test plans were interrupted by redirection of the Group II effort. Nevertheless approximately 20 command post test starts were accomplished before the phase was aborted. Of all test starts only one, a preliminary trial, was supported by an operable large wall display. All others simulated large group display utilization by assigning the large command post display to one of the command dais display consoles. Positional procedures prepared in advance were submitted to trial, modified as a consequence of test experience, and applied in updated form to succeeding test runs.

Simulation of the operational program system during the latter part of Phase C-1 tests permitted development and practice of procedures for each command dais position. Test observation and data collection techniques were established, tried and improved. An important step in development of demonstrable integrated system performance was achieved in the first "formal" Group II command post exercise (patterned after certain Group I techniques). The system, which this time included an operating large group display, was staffed entirely with military personnel. Test management and demonstrations were concluded to be procedurally reliable for conducting the balance of the command post tests.

Some interim findings upon conclusion of the C-1 Phase were that the large group display was not yet reliably accessible for test operations; operator skills which had an important bearing upon system performance were in need of a systematic program of development and maintenance and total system reliability—programs, computer, display equipment, personnel, operating procedures—at this point was sufficiently low to hinder conformance to test and demonstration schedules. However, these were all attributed primarily to the Group II development stage and problems of simulating Phase E program capabilities.

Demonstrations and exercises

Requirements for demonstrations and exercises, including training sessions, recurred throughout the development program and frequently appeared as explicit objectives and formally scheduled events in experimentation and test plans. Some of the earliest demonstrations concluded with NORAD concurrence on command port layout design and recommendations for procurement.

Command Post Demonstrations were developed in order to exhibit typical system features and capabilities for observers interested in design and operational aspects of the system. Two demonstrations were developed and employed.

1. Command Post Demonstration 1 (CP Demo 1) was designed as a 15-minute exercise to introduce the primary capabilities of the 425L display consoles and the SOSP representation of the 3-E Computer Program design. Demonstration design included program control of all console switch actions and a 15-minute magnetic tape recording of system inputs. These tape inputs generated a sampling of displays critical to mission performance. Appropriate simulated switch action "inputs" were included to demonstrate operator display manipulation and information retrieval. The standard presentation entailed running CP Demo 1 for approximately 15 minutes followed by 30 to 40 minutes of "live" display console operation by observers.

2. Command Post Demonstration 2 (CP Demo 2) was designed to supplement CP Demo 1 with a 45-minute tape showing a time-compressed sequence of peace-time, attack, and battle situations. Subsequent changes reduced the tape to 30 minutes, with fewer inputs, and incorporated the large wall display. Inputs consisted of a variety of events typifying the situations depicted. Simulated switch action "inputs" were used to demonstrate a number of complex operator actions including control of the large wall display. The routine presentation of CP Demo 2 involved running the CP Demo 1 tape for 15 minutes followed by 25 to 30 minutes of the CP Demo 2 tape. However, during initial demonstrations the wall display was not available due to installation difficulties.

SOSP testing concerned with development and documentation of operating procedures and familiarization of NORAD COC personnel with system performance in a simulated operational environment, was largely a demonstration phase. A total of 55 successful and partly successful runs were accomplished. Command post positions were repeatedly exercised over a wide range of simulated operational situations. CP Demos 1 and 2 were employed frequently during the SOSP interval and new demonstrations were developed.

SOSP experience suggested that demonstrations and training exercises be clearly separated from tests and that test participants be shielded from visiting observers and non-participating "experts." The importance of demonstrations for satisfying a variety of developmental requirements was made evident and repeatedly confirmed. The need for a stable operator complement and systematic positional and team training opportunities in support of system testing and operational system development was firmly established during the SOSP phase.

Objectives in the 2-E test series under the Category II Test Plan included the familiarization of Group I COC personnel and the Group III training organization regarding all aspects of system operations including simulation. Several system runs were allocated primarily for purposes of position training and operational procedures development. Recommendations were made that procedures be designed, documented and a review of test data, showed an apparent shortage of formally specified and documented position-oriented procedures which could be submitted to test and employed as operator training guides. The lack of documented procedures at this stage of system testing and the consequent difficulty in developing operator proficiency capable of exploiting system potential were considered serious barriers to the bchievement of levels of system operation permitting valid and reliable measures of operator dependent system performance (Lesiw, 1965a). A recommendation was made that procedures be designed, documented and verified, and systematic training be undertaken if the objectives of performance measurement and improvement were to be realized in Phase E System Testing.

During final testing of 3-E in Group II four test exercises were explicitly concerned with procedures and skill development problems. However, the combination of all test runs provided over 600 man-hours of positional familiarization with an average of 23 hours per position spent in the performance of command post and divisional operator tasks under a variety of simulated tactical and strategic attack conditions. Accumulated experience during Group II system tests amounted to manning, under simulated battle and task stress conditions, of more than one full week of prime shift operations with a Battle Staff directing the command post and divisional areas under the NORAD recommended organizational concept of COC management. Supplementary familiarization included approximately 9 hours of continuous participation in a Group II command post exercise patterned after Group I exercises involving COC and subordinate NORAD echelons.

Group III Phase E system tests resulted in 1160 man hours of system familiarization. Records of position encumbents indicate that at least 68 persons received operational position experience. Personnel rosters at the beginning of Category III system operations revealed that about 85 per cent of these remained to man positions in this follow-on period.

System Performance Demonstration (SPERD) constituted the primary device for accomplishing system turnover to NORAD upon conclusion of Category II tests. Design of the demonstration which entailed participation of lower NORAD echelons, had its orgin in large-scale command post exercises carried out in Group I.

SPERD was considered to have achieved its objectives particularly in providing NORAD with an opportunity to operate and observe the basic NORAD COC in an environment relatively free of test constraints. The success of the demonstration was borne out additionally by a CMC Technical Approval Demonstration (TAD) Board finding that the system was acceptable and essentially ready for operational use. SPERD also served as a source of data to permit verification of certain equipment functions and interface requirements not possible in earlier testing due to late equipment installation schedules.

SPERD, while not fundamentally concerned with familiarization, did serve to integrate personnel in an environment which netted over 350 man hours of experience in 24 operator positions including many incumbents of the initial operational system.

Category II system testing

System testing under the Category II Test Plan covered a period of approximately 17 months. Test objectives stated in the test plan governed all system testing. A 3-increment program package paced the test effort. The initial increment involved neither live interface with console operators nor the full capability for processing external messages. Consequently tests were confined to program shakedown and demonstration techniques developed previously for simulating console operator actions. Twenty-three test runs were accomplished before the second increment, 2-E, was available for tests.

Thirty tests were completed using the 2-E program by addressing all objectives specified in the system test plan. Of the 30 tests, 16 were concerned with the task of verifying the achievement of physical, functional, procedural and organizational requirements specified for the system. Performance measurement, particularly with regard to the handling and disposition of specialized system data, received considerable test attention. Based upon 2-E System Tests, 37 change recommendations affecting 2-E, 3-E (Group II), and 3-E (Group III) were documented dealing with procedures, training, and manning; communications and equipment; program; and miscellaneous items concerning test implementation and developmental needs. Fifty per cent of the recommendations had to do with procedures, training and manning.

Observation of positional performance across the system during 2-E tests, and a check of test data, showed an apparent shortage of formally specified and documented position-oriented procedures which could be submitted to verification tests and employed as operator orientation and training guides. The lack of documented procedures and the consequent difficulty in developing operator proficiency capable of exploiting system potential, were considered serious deterrents to the achievement of system operating levels which would permit valid and reliable measures of operator dependent system performance (Lesiw, 1965a).

Assuming operable hardware with known limits, a determination of what the system could do was largely a function of the procedures employed and the skill and experience of operators. Performance measurement during the 2-E test series was accomplished by sampling divisional area and command post tasks, and applying relatively inexact measurement techniques to the performance of developmentally primitive procedures by operators comparatively unskilled in system exploitation. Therefore, the obtained data were not indicative of proficient system performance (what the system is potentially capable of doing) but rather of how the system worked. What the system was capable of doing remained to be determined after procedures and skills were fully developed.

The final program increment, 3-E, supported system tests in Group II and Group III. Thirteen tests concluded the test activities scheduled for Group II. All tests contributed to the achievement of five major objectives of system testing. Three of 16 capabilities verified under 2-E system tests were reconfirmed while 7 additional user-oriented requirements were verified under 3-E Group II tests.

Of 39 recommendations made during the 3-E Group II test series, 13 were concerned with changes to the computer programs, 4 with modifications or additions to the then current equipment configuration, and the remaining 22 pertained to the personnel subsystem (procedures, manning, and training). Recommendations regarding procedures were greatest in number for the command post, most having to do with display generation, some with responsibility allocation, and a few with interaction between command post and divisions.

Results of the 2-E and 3-E Group II system tests provided a sound basis for continuation of system tests in Group III, including preparation for the demonstrations of system performance which terminated Category II tests. Simulation techniques and library resources were deemed adequate to test requirements specified in Group III test plans, while those capabilities established in the Group II system were considered ready for installation and operational extension in Group III.

Testing in Group III under the Category II Test Plan, assumed an integrity not realized in Group II. The pressure of firm schedules and an expected end product gave an easily understood and respected purpose to test plans and activities. Rapport among test, operator, simulation and support personnel exceeded levels achieved at any previous phase of experimentation or test. Results of Group III, 3-E, testing were considered in a context of overall Category II System tests and were interpreted in terms of the broad system objectives specified in the Category II Test Plan.

Equipment, program and operator aspects of system performance were examined separately and in combinations. Conclusions regarding capabilities were developed by engaging concepts of system capacity, availability, response time and compatibility.

Functional verification of equipment (Did it conform to specified requirements, and were alterations necessary?) was an integral part of all Category II tests and led to design discrepancy reports and proposals for design changes. Maintainability of the system equipment complex was established under several Test Orders. Equipment availability was determined for major components (MC) during a test run of 9 days. Major components included at least one central data processor, one magnetic drum, a display data controller, one input/output data controller with 16 operable input and 2 operable output lines, and a configuration control switch. Availability, expressed as a per cent, was found to be high in an 8-console configuration, decreasing with additional consoles up to 15. The large wall display, scheduled for access 22 hours daily during this test interval showed an availability comparable to the major components.

Program performance was investigated as an aspect of system preparation for tests. Standardized test vehicles were developed consisting principally of simulation tapes containing a broad sampling of specified inputs, also procedures for inserting operator actions and recording detected discrepancies. In a nine-month period concluding with termination of Category II tests, 193 program discrepancies were identified and documented. The high rate was attributed primarily to insufficient communication among programmers when programs were reorganized or recompiled and reassignment of personnel within programming groups. However, the correction rate for documented program errors was high enough to compensate for the volume of discrepancies permitting the system test effort to proceed according to schedule. Furthermore, only 8 per cent had a significant effect upon system testing in that they seriously degraded scheduled tests. Most of these were of the program loop or halt type. Nineteen per cent of the errors had to do with message input/output processing, 44 per cent were concerned with display generation, and 37 per cent affected internal processing.

Equipment/program performance, measured in terms of response time associated with the interaction of central data processor and operational programs, was determined by varying conditions of input loading. Of several input variables affecting response time, two were selected for control: message input rate, and operator action rate. Using the maximum capability of the source equipment and communication lines, and maximum capability of the simulation generation program, simulated message and switch action inputs were distributed as uniformly in time as conditions permitted. Under these extremes, equipment/program governed responses to switch actions and message inputs were well within specified limits.

The extent to which the computing capacity of the system is utilized was also measured under several conditions of loading. The measure employed was the percentage of total operating time during which the system is "idle" (time in which no inputs were being processed or waiting to be processed). The data reported gave idle time percentage as a function of six pre-selected combinations of simulated message input load and simulated operator action load. The range of idle time percentages were such that currently conceived maximum demands utilized about one-quarter of the systems computing capacity, while under the most extreme demands conceived, three-quarters would be used.

Equipment/program/operator performance actually involved verification of basic NORAD COC operational capabilities. A verification checklist based upon a guide developed during Group II system tests provided a standard for determining compliance. Results were obtained for five organizational functions and a special capability associated with system operation during training. Positive capabilities for performing all operational functions specified were demonstrated by the system. At the detail level, a total of 154 position tasks were observed; 145 were confirmed, 2 partially confirmed, and 7 were unconfirmed in comparison with pre-specified criteria for task performance. Problems noted in verification of the command post function were of a procedural nature with the principal difficulties having to do with techniques for data base monitoring and information exchange (Lesiw, 1965b).

The apparent influence of message input rate and simulated defense environment upon speed of response and accuracy was investigated for selected aspects of performance. The categories of obtained measures included responses to Command Post queries; report of condition changes to the Command Post; elapsed time between key message inputs and their appearance in command post displays; data base update rates, particularly error message correction; and, function maintenance and restoral action time. Specific results are not essential to the present discourse, however, the measurement categories may be of methodological interest.

System design specifications were based in part upon expected characteristics of data inputs to the central data processor (rates, degradation levels, etc.) during crisis situations, as well as upon expected characteristics of remote users of CDP outputs. An examination of some of these manifest characteristics in conjunction with the system design which emerged, provided insight regarding compatibility between the system and the real world. During the course of testing, two classes of input data were available for analysis and assumed to be representative, with respect to time distribution, of real crisis situations as currently conceived. The classes were operator console switch inputs, and certain message inputs via the input/output data controller (IODC). The switch inputs were obtained during realistic simulation tests and the message inputs during exercises which utilized live data from remote NORAD installations. The sample inputs were considered to be typical of periods expected to exhibit overload conditions.

It was concluded earlier that the system's computing capacity was more than adequate to deal with immediately envisioned real world input data. Analysis of the input samples just cited suggested a model to adjust such design parameters as computing speed, program scheduling, and input buffer allocation, in the event processing requirements increase significantly in the future. Operator console inputs were observed to have error rates (inputs unacceptable to the computer program) which had no noticeable effect on overall system performance hence no incompatibility was believed to exist at the console/operator interface. Message error detection and handling (those rejected by the IODC and those accepted by the IODC but rejected by the computer program) involved manual operations in the correction process. The probability of IODC rejects forming queues was extremely low and corrections were accomplished quickly. Computer detected message errors, on the other hand, were more likely to queue and took longer to correct. Appropriate recommendations for resolving this incompatibility with real world demands were made. Computer-generated message outputs could be given only limited study hence results regarding compatibility were inconclusive. Although compatible interactions were established among certain message outputs, transmission circuits, and terminal equipment, problems were identified and submitted to immediate study for improvement recommendations to be implemented during post Category II operations.

Table I summarizes the Category II test effort by relating major test objectives to the numbered Test Orders. It also shows whether objectives were achieved as a direct or indirect result of test plans. Recommendations for improvements as a consequence of the Category II Test Program were documented separately. A total of 92 improvement recommendations were made, of which 62 were proposals for specific system changes. Thirty took the form of study proposals. Of the original 92, 45 were concerned with computer program modifications, 45 with changes to internal system equipment, and 2 with external interfacing systems. Of the 62 proposals for specific changes, 40 were funded and scheduled as Category II testing closed. Thirty-six of the 62 dealt with equipment and 42 with programs. Display consoles, the large wall display, and display generation programs tallied the largest numbers of change proposals. Of the 30 study proposals, 19 were concerned with computer program improvements, 9 with equipment improvements, and 2 with improvements to external interfacing systems.

Developmental program implications

With this survey of the developmental effort involving experiments, tests, demonstrations and exercises essentially complete, it should be possible to review the nature of these accomplishments from a utilitarian point of view. However, it must be clearly recognized that the results of the developmental program, particularly their interpretation, are heavily dependent upon the methodology employed. For it was the rationale contained in the experimental and test plans which gave purpose, direction and systematicity to the developmental activities. And it was the techniques and procedures developed for simulation, data recording and documentation which laid the foundation for conducting developmental system operations and collecting relevant system data. The adaptability of the methodology injected a note of consistency throughout system evolution and made it possible to adjust the developmental effort to major changes in system concepts and program management.

Experimentation

The preparatory and exploratory activities of Phases C and C-I established an experimental system which could be operated, observed and described. It prompted development of simulation techniques which could place realistic demands upon the system, including its operators. The experimental system functioned as a diagnostic instrument in that it exhibited deficiences requiring immediate corrective action along several system dimensions—programs, operator procedures, equipments, and facilities—before development could continue suitably. Changes were instituted and evaluated. Command post mock up and layout design analysis, including studies of peripheral command post displays, were accomplished during this experimental interval which resulted in preparation of specifications for the Group III command post installation. Experimentation with various configurations of the command post, and its interaction with other elements of the Combat Operations Center permitted development of command post operating modes and concepts of integrated system operations necessary to follow-on phases of effort. The general outcome of the experimental phase was a test system with established rules for conducting system operations manned by operators indoctrinated in the philosophy of the test program. Included was the technology for accomplishing a wide assortment of test objectives as well as channels for introducing and implementing system change recommendations.

Demonstrations and exercises

Planned and extemporaneous demonstrations were employed throughout the developmental program in response to a variety of needs. Earliest demonstrations were concerned with illustrating system design characteristics and capability performance particularly for those requiring design or functional knowledge of the system for R&D management or operational planning purposes. Demonstrations were also employed as indoctrination tools for personnel assigned to man the system as well as for technical and management representatives of various agencies concerned

Table I. Category II Test Objectives—Test Order Correlation

Objectives	Test Orders†							
	303	307	308	311	313	314	315	316
1. System Preparation: Program Acceptance			**					
2. System Performance Measurement								
a. Operator Independent	**						**	
b. Operator Dependent					**	**	**	*
3. Equipment Functional Verification		*			*		*	
4. System Operational Capabilities Verification		**			**	**		*
5. User-Operator Personnel Familiarization				*	**	**		*
6. System Performance Demonstration								
a. Preparation					*	*	*	
b. Accomplishment								**
7. System Orientation Demonstration				**		*		
8. Procedures Development			*		**	**	*	*
9. System Improvement Identification	*	*	*		**	**	*	*

NOTE: *Indirect Objective **Direct Objective

†Test Orders 310 and 312 were considered initial increments (2-E, 3-E Group II) of TO 314 for this tabulation.

with developmental progress of the 425L Program. The most notable demonstration from a utility standpoint was the highly organized and extensively planned System Performance Demonstration that concluded Category II Testing and provided a vehicle for assessing the readiness of the Group III system to assume operational tasks.

Exercises for interim estimates of system effectiveness were developed for both Group II and Group III. These had the additional objective of integrating personnel functions and providing opportunities for organizational skill development. Training exercises for both operator procedures and skill development were regularly scheduled features of the test phases. The simulated operational support phase (SOSP) was essentially a period for demonstrating the capacity of the otherwise obsolete Phase C-I system to resemble the functional configuration proposed by the NORAD Study Group.

It is appropriate to conclude that separate demonstrations and exercises satisfied a variety of requirements for knowledge of the system without interrupting or imposing upon ongoing system experiments and tests.

Category II system tests

The period of Category II tests was oriented almost exclusively toward the development of operational capabilities; consequently the terminal product was a system ready to assist in the performance of the NORAD mission. This meant that the test program had provided confirmation of equipment and program design specifications and recommendations for changes as these were evident from test experience. It also implied that performance characteristics had been examined and that performance data were available regarding system response time, capacities for handling inputs and generating outputs, error identification and correction, data retrieval and display accuracy, and system compatibility with the real environment within which it had to perform.

The utility of this phase was especially manifest in the success with which general test objectives were addressed, and the users (NORAD) readiness to assume full control of system resources and capabilities at the conclusion of Category II tests. In addition the system recipient was provided with a package of simulation and exercise technology which made it possible to continue simulated operations during Category III operational testing or during regularly scheduled exercises to sustain and enhance skills for handling the system routinely and on a crisis conditioned basis.

Use of the Group II test facility in support of experimentation, demonstration and test activities preceding activation of Group III, contributed significantly to an abbreviated test effort and early operational employment in the latter. Category II testing in Group III was completed in less than five months. A program of product improvement consistent with the incremental approach was submitted for follow-on implementation.

Assessment of the field experimentation and test experience

In the preceding Section on Implications it is evident that the results of the developmental program of experiments and tests were viewed in a positive light. The conclusion drawn is that the program was a successful one in terms of both process and product. However, this conclusion does not imply that the program was ideal nor that methodological, administrative, and management problems were entirely absent. There were indeed many problems which should be of interest in an assessment of the development program experience.

Experimental system control

The system available in the earliest period of experimentation was incomplete and functionally unreliable, hence it could not be considered a "good" representation of its eventual operational counterpart. However a good representation at this early date would have become a poor one in the light of subsequent changes introduced by the NORAD Study Group. At this stage of experimentation then:

- Additions and changes were continuous in almost all system dimensions—hardware, programs, data handling, personnel, et al.;
- Developmental operating characteristics were unreliable, making performance measurements tenuous, especially for change recommendations and
- Overt design and procedural problems, particularly those concerned with bringing an operable system into being, were resolvable and constituted the prime work of experimentation.

Simulation technology, fundamental to credible exercise conditions, evolved with the system. The experimental phase did not have the benefit of military participation (subject matter experts) during initial simulation design. The result was lack of realism in problem situations which in turn affected operator motivation and hindered procedure development. Extensive user involvement in simulation development at a later date was disruptive due to the magnitude of change introduced in problem design and general simulation technique. Nevertheless the changes were constructive and improved technology. It can be said that:

- Sophistication in application of simulation techniques was partly contingent upon specific experience with the mechanics of utilizing the system under development, and partly dependent upon an understanding of the environment within which the system was to be employed. No less important was an awareness of the objectives to be realized as a result of experimentation, and
- Effectiveness of simulation, both in content and management, increased with time. Efficiency in problem design and production also showed a corresponding improvement.

Operating procedures and personnel skills could not be developed systematically under field experiment conditions; personnel assignments were unreliable and initial procedures, although specified to a useful level of detail, were not effectively extended and documented, hence were not easily verified or passed on to new position incumbents.

Experimentation as a phase of development was not fully understood or accepted as an important step in system evolution; consequently, support requirements were not satisfied as readily as might have been expected. At the same time, the experimental program did not enjoy a high priority in budgetary and developmental schedule considerations.

Achievement of experimental and test objectives

Despite situational factors, a partly installed system, programming problems, and an abruptly cancelled schedule of experiments, the objectives of experimentation were realized in limited form and were palpable as:

- A developmental concept of system operations;
- An established, working system, i.e., an initial increment capable of supporting analyses, tests, demonstrations and training exercises;
- A concept of simulation for the system and initial development of simulation technology;
- A cadre of indoctrinated experimental system operators not fully proficient but capable of employing developmental procedures;
- Estimates of data handling capabilities and selected performance characteristics;
- Program, equipment and facility changes, and
- Resolutions of command post design and a working basis for integration of command post functions with the balance of the system.

The objectives of Category II system testing were achieved essentially as specified but required the development of several supporting documents which served as criterion statements and performance measurement guides. Criterion statements extracted from a family of system requirement documents, were necessary to accomplish the objective dealing with verification of physical and functional capabilities. A performance measurement guide, concerned with measures of system interaction with the external environment and maintenance of internal integrity, addressed the objective of "measurement of system performance . . . in order to provide baseline performance data . . . ." Both documents are methodological curios in that they represent explicit endeavors to deal with the evaluation issues of system performance measurement (What to measure?) and criterion development (What constitutes performance adequacy?).

Utility of the Group II experimental facility

The utility of the experimental facility has been substantiated throughout this review. Only a few additional remarks are necessary to the effect that it:

- Provided a setting for early experimental analysis of system design and operating characteristics;
- Supplied an austere but effective test bed for initiating Category II tests sufficiently in advance to influence the configuration of the Cheyenne Mountain Complex NCOC and to simplify transfer of integrated Group I and II operations to the hard site shortly after occupancy;
- Provided some initial system information of interest to the NORAD Cheyenne Mountain Complex Task Force in its deliberations regarding the direction to be taken in NCOC system development;
- Facilitated a variety of design changes as testing progressed, and supported evaluation of approved changes upon implementation;
- Served as a relatively economical mock-up facility for command dais analysis, design and test, as well as a model for exploring variations in facility design for the 3-tiered command post in advance of Group III construction, and
- Provided a basis for establishing initial support requirements for operation and maintenance of the 425L segment of the Group III NCOC.

CONCLUSION

From the experience with field experimentation and tests associated with NORAD COC (System 425L) development as described in this review, it is judged that the contributions of these developmental activities facilitated an orderly and economical evolution of a system demonstrably capable of performing its specified mission. Within situational and physical constraints, objectives were achieved for each phase of system experimentation and test. The experimental facility (Group II) did in fact provide a relatively economical and effective vehicle for establishing, exploring, assessing and improving initial system concepts, design and operations. Experimental data obtained in early observations of system behavior permitted meaningful forecasts of operational system performance and estimates of personnel subsystem requirements, as well as change recommendations (hardware, programs, facilities) which were incorporable in plans and specifications for the system subsequently installed in the Cheyenne Mountain Complex.

It is also true that the program of experiments and tests, particularly the former, was hindered by developmental deficiencies in several system elements which required inordinate amounts of preparatory and shakedown attention. This was especially evident in early programs and certain equipments such as the camera, processor, projector system which generated large screen displays. Redefinition of the system and redirection of the development program as a result of the Cheyenne Mountain Complex Task Force Study were disruptive influences in that they required a new cycle of planning, a revised concept of system, and a re-orientation of developmental activities from experimentation to operational development testing. The methodological approach employed was adequate to bridge the shift and the result was a well-planned program of Category II testing which facilitated transition from Group II to Group III and rapid turnover of a system acceptable to the recipient user.

Generalizations

Based upon the experiences related in this review, a number of generalizations appear justified regarding field experimentation and testing of large scale information systems of the type represented by the NORAD Combat Operations Center. Order of presentation does not imply priority.

1. It is advisable to establish a separate facility representative but not duplicative of the physical and functional characteristics of the subsequent system as a vehicle for experimentation and exploratory testing. Preferably this experimental facility should be established early enough so that experimental and test findings are influential in setting and refining design specifications for the eventual operational system rather that in correcting defects in a concurrently installed "operational" system.

2. The importance of documented plans for experimentation and tests, particularly methodological considerations—objectives, experimental design, data collection and reduction, and result reporting—cannot be overstressed since these constitute the framework for management and implementation. They also may provide the standards for assessment of results if measurement techniques and criteria for evaluation are integrated with objectives, experimental design, and data collection and recording procedures. An explicit methodology and implementation plan simplifies the task of assessing effects of major changes to a developmental system and its associated experimentation and test program. Short notice reorganization of effort is also facilitated.

3. The development of simulation technology should be accomplished systematically and comprehensively as early as possible in the establishment of an experimental and test system and should receive studied attention for improvement throughout the experimental phase not only because it directly affects test results but also because it will have an important role in the post-test system as a device for system exercises and proficiency evaluation.

4. In conducting experimental system runs and scheduled tests, at least one "system engineer," intimately acquainted with simulation technology, program characteristics, system equipment, and objectives and procedures for the scheduled experiment or test, should be stationed at the central data processor to manage the test system. The role of this test engineer is that of an on-the-spot diagnostician of probable system problems, familiar with routine malfunctions and acceptable conditions of system degradation, who can troubleshoot and correct or make immediate arrangements for corrective action as emergency needs arise. This test controller will make judgments to "abort" or to continue in the light of prevailing conditions. He should function as an "optimizer" of experimental system utilization.

5. The credibility of simulation is enhanced by the presence of a simulation supervisor who is thoroughly versed in the subject matter aspects of combat operations center functions, the characteristics of the simulation problem to be employed, the immediate objectives of the experiment or test scheduled, and the organization and management of the simulation center.

6. Complex programs designed for large scale information systems of the type encountered in the NCOC development effort may be easier to handle in the experimentation and test phase if they are designed and produced in logical or functional increments and integrated with the balance of the

experimental system in successive test phases.

7. The phase approach to system evolution appears to facilitate organization and management of experimentation and tests because it promotes periodic assessments of system status and logical junctures for introducing additions and changes. The assumption is made that phase terminations are complemented by documented statements of achievements, analytical findings, and recommendations for follow-on increments as well as for configuration of the eventual operational system.

REFERENCES

1 DEPARTMENT OF THE AIR FORCE *Research and development testing/evaluation of systems, subsystems and equipment.* Air Force Regulation 80-14 Washington D. C. 14 August 1963

2 W. LESIW *Psychological factors in system development: performance measurement.* J. Colo. Wyo. Acad. Sci. 1965 Vol. V, No. 6, p. 47 (a)

3 W. LESIW *System organization and information management:* NORAD COC. Paper delivered at IEEE Workshop on Implementation of Communications-Based EDP Information Systems Phoenix 3 December 1965. (b)

INFORMATION SYSTEMS FOR INTELLIGENCE

RUTH M. DAVIS, *Chairman*

Session Participants:

MAURICE H. HELLNER
JOSEPH BLUM
HAROLD WOOSTER
MERRILL FLOOD

Introductory Remarks

RUTH M. DAVIS
Department of Defense
Washington, D. C.

Yes, Intelligence information systems do have much in common with other information systems. Most of those who have observed Intelligence information systems recently either by desire or by direction have made it their objective to stress the commonality between Intelligence and all other information systems. Such observations are akin to those which treat similarities among men of different origins and discount the differences. In both cases, it is the differences, which when neglected, create the problems and the frictions. One feels some sympathy towards the professional societies of England who finally were led several centuries ago to pay mathematicians for *not* submitting papers containing proofs of how to trisect the angle. So it is that one looks now with delight to those authors who have the insight to understand and present the important differences which set apart Intelligence information systems and their customers—the Intelligence analysts.

The papers of Dr. Hellner and Dr. Wooster prepared for the Session on Information Systems Within the Intelligence Community are remarkable in the way in which distinctive features of Intelligence and of the Intelligence analyst are isolated and put into perspective. As Dr. Hellner points out, "the heart of the problem of Intelligence analysis is to draw valid inferences from circumstantial evidence." The Intelligence analyst must make use of "hypotheses that invite investigation" in his reasoning and his generation of intelligence products. Intelligence information systems must assist the analyst as he formulates multiple hypotheses and they must be willing, contributing partners. Information systems which merely store and retrieve facts will be "of assistance to the *annalist* but not to the *analyst*."

Dr. Wooster compares, in his always-effective manner, the intelligence analyst to the miner whose job is to take low-grade ore and enrich it ("beneficiation of fairly low grade ore") There is the implication throughout his paper that any information system which does not improve this ore refinement-like task of the Intelligence analyst should be itself relegated to the salt mines. He also states quite unequivocally that mechanization in any of its standard forms "can provide a partial, but only a partial, amelioration" of the troubles besetting the Intelligence analyst.

The long and productive experiences that Dr. Hellner and Dr. Wooster have enjoyed in the Intelligence field and the Information Science field respectively place considerable weight upon their observations.

Perhaps their very apt reasoning can be extended by recalling that information can deal only with four sets of phenomena. Two of these sets relate to *actual events*—those which stand for themselves—and which are studied by means of information on the *functioning* or *behavior* of the organization, the device, the object itself, and by means of information on the *environment* in which the functioning or behavior occurs. The other two phenomena relate to *symbolic events*; namely, those that stand for or are seen in place of another event. In this case, observations must be made on *communications* or what we may call the *symbolic functioning* of organizations, devices, objects, etc., and on the *environment* or the *context* in which the communication occurs.

Most military information systems with the exception of Intelligence systems handle information dealing with actual events. Logistics information systems manipulate data on loading of cargo vehicles on inventories of items, on schedules for cargo delivery and the like. Planning is done against a known environment of cargo vehicles, inventories, ports, airports, requirements and schedules.

Command-control information systems are stocked with data on readiness status of military units and equipment, on locations of known ships and aircraft, on known characteristics of artillery, submarines, missiles, etc. Planning is done against a known background of trajectories, routes, launch times, targets and weapons. Banking information systems carry data on existing status of checking accounts, savings accounts and loans and on completed withdrawals or deposits. Planning is done against a known environment of interest rates, congressional legislation, stock market trends and the like.

In all these just cited cases the actual event (or "thing") can be checked by direct observation and the environment itself is subject to direct observation.

In the world which is the subject of Intelligence, the actual production of a factory in a foreign unfriendly country cannot be directly observed. The color, density and volume of smoke pouring from smokestacks of the factory must be viewed from a photograph (a symbol of the factory communicated via photographic means) and its production estimated. If a foreign newspaper article can be located which describes a new university laboratory concerned with welding techniques for a given metal which is located near the site of the factory, then one might infer the factory output from circumstantial evidence. In reality, the *actual event* of interest is the number of operational missiles in the country in question, the factory output is the *symbolic event* and the smoke visible in the photograph is the *communication*. The Intelligence analyst must somehow, as Dr. Hellner states, employ categorization and classification techniques and draw proper inferences through interpretation of indicators exhibited. In so responding to information, the Intelligence analyst cannot use libraries and information in the customary way and should develop some specialized tools and practices, the lack of which Dr. Wooster deplores.

How might one present this world of symbolic events with which the Intelligence Community deals daily? Consider perhaps a simplified model of such a world. At any given moment in time, events are occurring which are either isolated or members of a sequence. As they are happening, information is being received about events. This information, however, may not have any direct bearing upon those particular events which are occurring about which one desires information. The third element in this model then, is that set of events deemed of importance. Thus, the triad of sets of events which forms the basis of any model of the world as seen by the Intelligence Community is:

1. The set of events which is occurring (symbolic events).
2. The set of events about which information is being received (communication of symbolic or actual events).
3. The set of events about which knowledge is required (actual events).

The implication that these sets of events may be completely or partially discrete is of extreme importance to the design of effective information systems for Intelligence analysts. To bridge the gap between the set of events which is occurring and the set about which information is received requires the application of inferential logic of a type not generally understood. No information system has yet been designed to provide assistance to the Intelligence analyst in this formidable task.

However, Dr. Blum, in his very lucid paper is quite insistent that "computer simulation provides a powerful methodology" for performing experiments rapidly on complex information systems. He rightly points out that in the future a greater emphasis will be placed on including software control features in such simulation experiments and certainly in any information system aiming to aid the Intelligence analyst software will predominate over hardware in importance. Dr. Blum also states that it is advisable to utilize the analytic approach with its mathematical models whenever possible and hence lends support to the sentiments of Dr. Hellner concerning the need for experimenting with analytic system models of inferential logic to support the Intelligence analyst.

The three papers for the Session when viewed as a composite, provide an intriguing picture and an accurate picture of the symbolic world with which the Intelligence Community must deal, of the special characteristics which make for a good Intelligence analyst in his role as observer of this symbolic world and of a powerful tool called simulation to aid in the design of those special and different information systems needed to aid the Intelligence analyst.

Evidence and inference in foreign intelligence

by MAURICE H. HELLNER
Defense Intelligence Agency
Washington, D. C.

INTRODUCTION

As President Roosevelt prepared for the Yalta Conference in early 1945, the Joint Chiefs of Staff brought pressure upon him to persuade Stalin to bring Russia into the war against Japan (Congress, 1951). Their insistence was based primarily on concern over the 700,000-man Japanese Kwantung Army in Manchuria. This army had the reputation of being an extremely efficient, well commanded, and effective fighting force. The Joint Chiefs feared that in a conquest of this army without Russian help U.S. casualties would run extremely high. Roosevelt followed their advice and "persuaded" the Russians to enter the war against Japan by agreeing to Russian territorial acquisitions in the Far East (Toland, 1966).

When the Russians entered the war against Japan in the summer of 1945, the highly-touted Kwantung Army crumbled away before them. After the Japanese surrender, the U.S. learned that by the time of the Yalta Conference the Kwantung Army was depleted and by no means an effective fighting force. Most of its elite units had been transferred piecemeal to the battle zones of the Pacific as the need for them arose. In January 1943 for example, the Japanese had 14 divisions in Manchuria. Two years later, 10 divisions plus 50,000 men in smaller units had been shipped out.

This is a discouraging example of a failure of intelligence to provide required information to high military and civilian officials at critical moments in history. The failure is all the more appalling when it is realized that the information which would have permitted an accurate estimate of the situation existed in bits and pieces within the U.S. Intelligence Community but was never properly collated. As U.S. forces captured island after island in the Pacific war, the Japanese units which were destroyed were noted. No comprehensive effort was made, however, to relate these units to units which were previously known to have existed in Japan and Manchuria. Even in those isolated instances where this was done (as in the case of the Japanese 50th and 135th Infantry Divisions on Saipan when Army G-2 observed that these were units of the Kwantung Army), no particular significance appears to have been attached to the fact (Farago, 1954). Thus, although some of our highest officials opposed our efforts to bring Russia into the Pacific war, the specter of the "powerful" Kwantung Army shaped the thinking of our top decision makers.

This is but one illustration of the sobering truth, learned long ago in the physical sciences, that in the intelligence field the mere collection, storage, and retrieval of "facts" is no guarantee that an accurate assessment of a situation will follow. Except in rare instances, the facts do not "speak for themselves." They must be correlated in such a manner as to provide the basis for valid inferences. The heart of the intelligence problem is to develop techniques and procedures for doing just this. Direct observation, and sometimes intuition, are important pathways to knowledge, but the bulk of human knowledge is obtained by inference.

The foregoing should exercise a restraining influence on those who believe that modern computer technology can provide a major breakthrough in intelligence processing capabilities merely by increasing one's ability to store and retrieve data. The intelligence processing problem consists of considerably more than providing expanded library services for an increasing volume of information to a large number of analysts. It is an utterly superficial view that truth is to be found by merely collecting, storing and retrieving facts.

Although there is considerable literature on intelligence operations down through the ages, including a great deal on methods of collection and observation, the literature of the trade contains virtually nothing on the methodology of analyzing the data that has been collected. It is a strange paradox that in the literature and folklore of intelligence the heroes have invariably been the collectors of information (e.g., the spies), whereas in the literature and folklore of crime the heroes have nearly always been the effective analysts of information (e.g., the detectives). Throughout the years of recorded history there are startling examples of accurate intelligence estimates and predictions, but they are nearly always based on the acquisition of a few items of dramatic information from a single source rather than through the painstaking correlation and analysis of obscure and fragmentary data from many sources. The U.S. victory at the Battle of Midway, for example, was due in major part to an accurate intelligence prediction, but this prediction was based on the acquisition of the Japanese battle plan rather than through the piecing together of fragmentary data (Fuchido, 1955).

One may well ask at this point whether all inferences in intelligence are not destined to remain more artistic than scientific. At best, intelligence today is a discipline in the process of formation. The major obstacle to progress lies, not so much in computer technology (although major improvements are required here), but in developing methodology for the analytic process. Today, systems designers attempting to develop computer systems for processing intelligence data are forced first to attempt to develop the procedures of analysis before they can proceed with their task. It is as though the systems designers attempting to apply computer technology to a mathematical problem had first to develop the formal logic of mathematics before they could proceed.

Nature of the problem

It has been pointed out by scores of writers that the first step in developing any methodology is to define the problem—to isolate the questions and issues involved. We have already suggested that the mere storage and retrieval of data is not the basic problem in intelligence analysis. The basic problem is to develop a methodology which will facilitate the making of valid inferences from the available evidence.

One may object at the outset that this is true of any field of human investigation, and so it is. In the intelligence field, however, the process of drawing valid inferences from available data is a particularly complex one, and one that has not been studied in depth. The complexity arises from two principal causes: (1) the nature of the raw information; and (2) the character of the processing requirements. These may be summarized as follows:

1. *Nature of Raw Intelligence Data:*
   - (a) Wide variation in form, content, timeliness, and reliability.
   - (b) Excessive amount of complicating detail. Nearly every "indicator" can be confronted by other "indicators" which contradict it.
   - (c) Extremely high volume.
   - (d) Large gaps and inconsistencies.
   - (e) Deliberate deception and disguise of vital elements.
   - (f) Non-controllable information sources.
2. *Character of Processing Requirements:*
   - (a) Great importance of random and rare events.
   - (b) Unpredictability of processing load.
   - (c) Tendency toward rapid changes in focus of attention.
   - (d) Wide variety of user requirements.
   - (e) Severe time restrictions on many processing tasks.

Superimposed on all of these factors are a number of complications, many of which are unique to the field of intelligence processing:

1. Physical and social scientists search for indicators of classes of events. Intelligence analysts are more concerned with indicators of specific events. This raises an acute methodological problem. How can we determine the probability of a single case (a coup d'etat, a surprise attack, etc.)? To put it another way, how can the scientific method which has been developed to analyze the universe of recurrent events be adapted to permit valid predictive inferences pertaining to political and military events which are unique and nonrecurrent?

2. Although each political and military event is unique, it is a mental and physical impossibility to deal with each event as completely unique. Classification and categorization are fundamental to human understanding. The entire concept of "normal" and "abnormal" situations implies categorization.

3. "Categorization" in the intelligence field is an extremely delicate process. The rules for defining individual categories are influenced markedly by the estimate of the situation the analyst already has in his mind. The history of intelligence processing reveals quite clearly that important pieces of information are often subsumed in the wrong category for no other reason than that the prevailing estimate of the situation made it logical to do so. This is perhaps the single greatest weakness of intelligence processing in totalitarian countries where the "facts" which are collected are often misinterpreted because the official party line misguides the categorization process. Thus, the USSR was surprised by the German invasion of 1941, in spite of ample evidence that German troops were massing on the frontier, because the official party line emanating from Stalin was that the Germans would not attack (Seth, 1964). It is interesting to note, however, that even British intelligence fell into the same trap on that occasion. The Joint Intelligence Committee as late as one month before the Nazi invasion of Russia, categorized the reports of large-scale Nazi troop concentrations on the Soviet border as part of the pressure Hitler was putting on Stalin to provide additional strategic materials to blockaded Germany. This was done because the offical British estimate of the situation was that there was a large community of interest between Germany

and Russia in overrunning and dividing the British Empire, and that it was therefore illogical for Germany to attack Russia at that time (Churchill, 1950).

4. An individual "indicator" from which an inference may be drawn has only a probability relationship to any given situation. No one indicator is likely to be conclusive. Each indicator, by itself, is susceptible to a number of interpretations. When one attempts to formulate a profile of indicators, it is difficult to determine the confines of the profile. This is particularly true because implication is a non-symmetrical relationship. The reverse of "If 'p' then 'q' " is not necessarily so.

5. The main stream of events and indicators frequently obscures important subordinate events. Thus, the U.S. Intelligence Community predicted quite accurately the time, place, and direction of the main Japanese thrust at the beginning of our involvement in World War II. It warned our decision makers that Japan was likely to go to war either the last week in November or the first week in December 1941 and that the main attack would be directed south toward Indonesia, the Philippines, and Malaya (Wohlstetter, 1962). It became so obsessed in its analysis of all "indicators" of this attack, however, that it did not establish the required the required collection and processing posture for detecting the subordinate attack against Pearl Harbor.

6. Accurate analysis of raw information is sometimes complicated by the fact that two or more major events are developing at the same time. In such cases, the "categorization" of indicators becomes extremely confusing because some indicators may point to one event, some to the other, some to both, and some to neither.

7. International political and military affairs are characterized by action and reaction. It is generally impossible to analyze accurately the likely courses of action of foreign countries except in relation to actions and policies of other countries. Moreover, the "facts" our estimators look at are almost certainly not identical to those of the foreign planners. During 1941, for example, the pattern of action and reaction between the Japanese and American governments became progressively more complex. By and large, U.S. intelligence officers in Hawaii were seriously handicapped in formulating their estimates of impending Japanese actions by their dearth of information on U.S. diplomatic moves.

8. In intelligence, we are dealing primarily with descriptive data. We have very little information pertaining to casual relationships. This considerably complicates the problem of drawing valid inferences.

9. In intelligence, transient relationships and characteristics are often more important than permanent or invariant ones. Thus, the transient relationships between a group of ships in a task force may be far more meaningful than any formal order of battle. There is, however, a great poverty in methodology for structuring data bases to accommodate transient characteristics and relationships.

10. Normally, the term "evidence" refers to an accumulation of data and the term "inference" refers to propositions or conclusions which were not included in the original data but which were logically deduced, inferred, or extracted from it. Such a distinction, however, has a deceptive clarity in most professional disciplines, and especially in intelligence. In "real life," yesterday's inferences are today's basic data. Inferences and facts become integrated in such a manner that it is impossible to separate them neatly. More significant, in many ways subtle and obvious past inferences shape current "facts."

11. In the real world, a solid prediction often goes sour because of the intervention of an accidental event —a heart attack by a key political figure, a plane crash which upsets plans for a coup d'etat, a flood, etc. Because of this, the best one can hope for in intelligence is prediction in terms of probability.

12. To the academic scientist, time may not be of particular consequence. He is after a high degree of confidence in his results, and whatever time it takes to get the degree of confidence he is after, he can usually take. The intelligence analyst, on the other hand, must frequently reach his best possible conclusion in a limited time.

13. Finally, there is the age old problem of understanding behavior that is alien to one's own culture. Any attempt to develop simulations, models, or profiles of indicators must face the fact that our criteria and belief systems may be inaccurate and unrealistic with reference to a particular foreign society. Allen Dulles, the former Director of Central Intelligence, expressed this point very colorfully when he said: "Actions and reactions can no longer be estimated on the basis of what we ourselves might do if we were in Khrushchev's shoes, because, as we have seen at the United Nations, he takes off his shoes (Dulles, 1963).

Approach

One may well ask, in view of this long catalog of obstacles and difficulties, whether intelligence processing is not destined to remain more in the nature of an art than a science. Game theory, model building, statistical techniques, content analysis, simulation, and operations research all appear to have application to the field of intelligence, but only in restricted areas. By and large, it does not appear possible to formulate any "laws of events" in the military and political area in quantitative terms.

In a very real sense the heart of the problem of intelligence analysis is to draw valid inferences from circumstantial evidence. If approached in this manner, it is believed that progress can be made in developing a methodology. As used here, the term "circumstantial evidence" is contrasted with what may be described as evidence bearing directly on the main problem. It denotes evidence concerning certain facts which, so to speak, surround the main event. It is like piecing together the pieces of a picture puzzle, but a picture puzzle of which many parts are missing, while others have been damaged, distorted, or faked.

It is true that in the vast majority of cases in which reasoning is based on circumstantial evidence there is no possibility of generalizing the result. That is so because the majority of the cases dealt with are such as are not likely to recur in exactly the same form. This is precisely the situation we find in intelligence.

The characteristic feature of reasoning from circumstantial evidence consists of the construction of a system for which the various items of evidence form coherent parts. We begin with a suggested explanation or solution of the problem. Such tentative explanations are suggested by something in the subject matter and/or by our previous knowledge. When they are formulated as propositions we normally call them "hypotheses." The function of a hypothesis is to direct our search for order among facts, particularly for order among confusing facts.

It is perhaps strange that although intelligence analysts have borrowed many scientific methods, they have shown an odd reluctance to resort to the formal use of hypotheses. To many, a hypothesis has seemed like a commitment, a pre-judgment, when the mind should be kept open for the "facts." They are afraid that the use of hypotheses will lead them into the danger of forcing the facts to fit the theory. They feel that any given situation should be studied without prejudice or presupposition.

Yet, scientists learned long ago that nature gives no reply to a general inquiry. She must be interrogated by questions. The observer can only observe that which he has been led by some hypothesis to look for. Charles Darwin expressed this concept very well: "About 30 years ago there was much talk that geologists ought to observe and not to theorize; and I well remember some one saying that at this rate a man might as well go into a gravel pit and count the pebbles and describe the colors. How odd it is that anyone should not see that all observation must be for or against some view if it is to be of any service (Darwin, 1903).

It is frequently stressed that the intelligence analyst must be open-minded, but open-mindedness should not be construed to mean mental vacuity. There is an important difference between hypotheses that invite investigation and fixed theories that control investigation. A hypothesis is really a mental attitude, an approach to the categorization of facts. It is a method of assuming certain situations experimentally in order to test the result. The hypothesis may very well prove unsatisfactory and be discarded.

It cannot be denied, of course, that there is a subtle tendency for one to bestow a sort of parental affection upon an intellectual offspring. If this tendency is not resolutely offset, the usefulness of a hypothesis will vanish.

One means of control is the use of multiple hypotheses. By employing this method the analyst becomes the parent of a family of hypotheses, and thereby cannot fasten his affections unduly upon any one. The problem with this approach in the past has been that the human mind has extreme difficulty in pursuing two or more lines of reasoning at the same time. Simultaneous vision from different standpoints appears to be a virtual impossibility for the unaided human mind.

The introduction of computer technology into the processing of intelligence information may well provide the assistance to the human mind that is needed for employing the method of multiple hypotheses. If so, it is evident that this will require a man-machine relationship of an order which has not been achieved thus far.

The requirements for such an approach are rather demanding. Some of these are:

1 An on-line, time-sharing computer system that offers considerably more than the mere retrieval of a specified item of information on demand. It must possess a query capability which permits the testing of hypotheses.

2 Data bases which are structured not only to permit the categorization of data according to common characteristics, but, more important, to permit the discovery of connections whereby one item of data can be related to another. A useful data base in intelligence will not be achieved by categorization alone.

3 Data bases which are structured not only to accommodate permanent characteristics and relationships, but also transient characteristics and relationships. In a fast-moving world, transient characteristics and relationships are frequently much more important than permanent ones.

4 Extensive use of "feedback" so that valid inferences can be fed back into the data base.

5 Research into the kinds of events which are reasonably predictable on a probability basis. A beginning could be made by ranking important kinds of events in terms of their probable predictability.

CONCLUSION

Computer systems which offer little more than the mere storage and retrieval of "facts" are of limited utility in the processing of intelligence information. They can provide a certain amount of assistance to the annalist but not to the analyst. They can provide very little assistance in the formulation of valid inferences, particularly predictive inferences.

Intelligence today is a discipline in the process of formation. The techniques employed are a mixture of that which is unique to intelligence and that which is not. The mixture is certainly unique. Future progress in this field will be heavily dependent upon the development of methodology and technology for drawing valid inferences from raw information which is generally incomplete, contradictory, of varying accuracy, and often faked.

REFERENCES

1 U. S. Congress, Senate, Committee on Armed Services, *Hearings on the Military Situation in the Far East* 82nd Congress, 1st Session On page 3332, two memoranda from the JCS are reproduced verbatim 1951

2 JOHN TOLAND *The Last* 100 *Days.* Random House New York page 107 1966

3 LADISLAS FARAGO *War of Wits.* Funk & Wagnalls Company, New York pp. 106-110 1954

4 MITSUO FUCHIDA *Midway, The Batte That Doomed Japan.* U.S. Naval Institute, Maryland page 131 1955

5 RONALD SETH *Operation Barbarossa.* Anthony Blond, London Chapters 4 and 7 1964

6 WINSTON S. CHURCHILL *The Grand Alliance.* Houghton Mifflin Company, Boston page 355 1950

7 ROBERTA WOHLSTETTER *Pearl Harbor, warning and decision* Stanford University Press page 386 1962

8 ALLEN DULLES *The Craft of Intelligence.* Harper's Magazine page 168 April 1963

9 CHARLES DARWIN *More Letters.* page 195 1903

Modelling, simulation and information system design

by JOSEPH BLUM
Department of Defense
Washington, D. C.

INTRODUCTION

The need for simulation

It has been conceded that very little scientific knowledge exists about information systems although much has been written. The literature in this field contains a preponderance of philosophy and speculation and the overall impression is that progress is painfully slow. What progress we have is derived mainly from experience in the development of actual information systems. Such systems characteristically have a development cycle which consumes as much as five or ten years traversing a sequence of phases which begins with a feasibility study and ends with final implementation. It is not uncommon for knowledge concerning optimal performance to become known so late in the development that changes in many design parameters are to costly to make. A methodology is badly needed which will permit a more rapid performance of scientific experiments on information systems. Such a methodology will not only increase progress in the research laboratory but if applied early in the development cycle will also provide the necessary feedback to permit the changes in design parameters which achieve a closer to optimal performance for the total system. It is the thesis of this paper that computer simulation provides a powerful methodology suitable for attaining these objectives. Up to the present this methodology has been applied mostly to solving hardware design problems. For the future a greater emphasis will be placed on including software control subsystems within the scope of simulation experiments. This is particularly important, for example, in storage and retrieval systems which provide on-line service to a collection of users for the logical organization of system control is substantially imbedded in software.

Our experience has indicated that, to a greater degree than is provided by alternative approaches, simulation

- provides a vivid and understandable picture of how a system works,
- makes explicit those assumptions which are often implicit,
- encourages the use of explicit measures of performance,
- permits decision-makers to play an active role earlier in the design process,
- provides a convincing demonstration of logical completeness and consistency,
- provides decision makers with greater insight into how to make trade-offs in system parameters,
- inspires confidence in management and results in a better engineering of consent among those who will be required to implement, operate and use the resulting system,
- permits an early recognition, formulation and solution of difficult and subtle interface problems, and
- provides an early discovery of explicit aspects of the environment which require further study.

Basic concepts in computer simulation

Simulation is a problem solving technique. Essentially it is a particular technique of the experimental method and is especially useful in situations where experimentation with the physical system is inconvenient, costly or impossible. In such situations an abstract model of the proposed or physical system is created and experiments are performed upon the abstract model. Experiments with models of information systems generally involve dynamical relationships. For such systems simulation may be defined briefly as a dynamic representation achieved by constructing a model and driving it through time. In principle the clerical, arithmetic and decision-making operations required to drive a model can be performed by hand. In practice large or complex models require the assistance of electronic computers to carry out the enormous number of operations. The logical structure of the model and the control mechanisms for driving it are incorporated within a computer program. The computer program generally will also include suitable controls over the course of the experiment and a complete record of the internal operation of the model. The program not only generates a variety of data but also processes data, analyzes data and prepares reports to integrate the results of the experiment. The general procedure for implementing a computer simulation experiment is outlined in Figure 1.

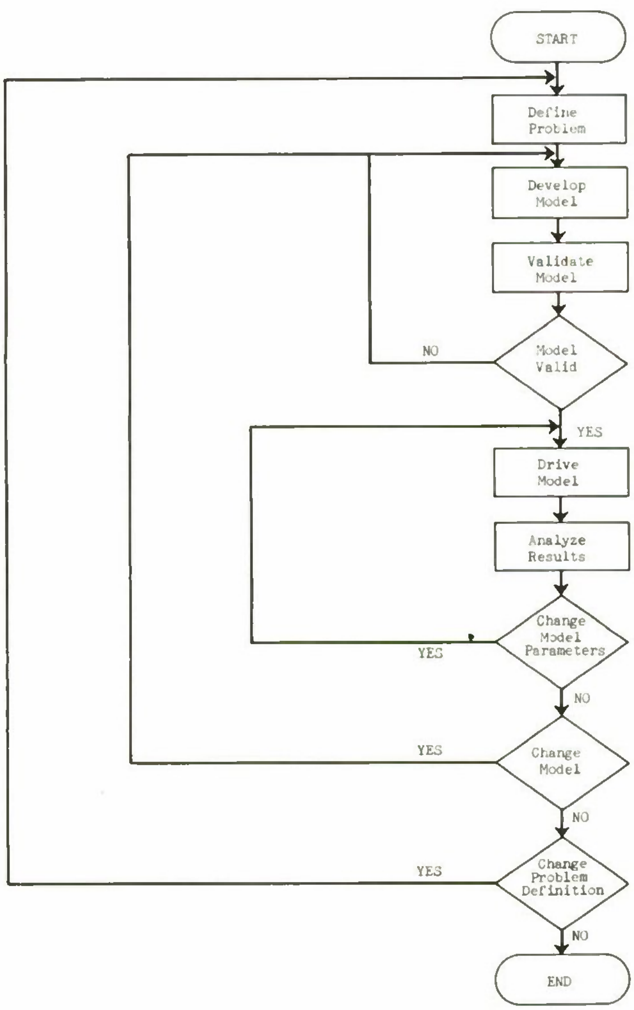


Figure 1—Procedure for simulation experiment

Implementation of simulation experiments

The design of a simulation experiment, as in other forms of experimentation, should depend on the motivation and objectives of the experimenter. The following situations may occur singly or in combination:

- The experimenter has a hypothesis concerning a system. The experiment is designed to confirm or reject the hypothesis.
- The experimenter has alternative system designs. The experiment is designed to select the "best" alternative.
- The experimenter has a proposed system design. The experiment is designed to study the behavior and performance of the system.
- The experimenter has a description of a system. The experiment consists in modelling and simulating the system to determine whether the description is logically consistent and complete.

The last situation, though not in the classical pattern for an experiment, has been found very useful in testing preliminary design concepts. With a modest outlay of effort the designer can quickly spot basic flaws or holes in the design. Frequently these are found in the modeling phases even before the computer program is written. In the design of complex systems a single experiment is generally inadequate to solve the collection of problems facing the designer. It then becomes necessary to lay out a program of experimentation and to modify the program as required. The experimenter's skill and experience in the use of the scientific method play an important role in the successful organization, implementation and execution of such a program. Finally, if the experimenter intends to make a contribution to knowledge he has the obligation to document the experiment with care and thoroughness so that other independent investigators can duplicate the experiment and confirm the results and conclusions.

Defining the problem

The problem definition phase is crucial to the design of a simulation experiment. A poorly defined problem will lead to an inadequate model and inconclusive results. A number of important and basic questions are posed below to which the experimenter should provide satisfactory answers before he proceeds.

- What problem am I trying to solve? If the problem is solved what new knowledge will be available? How important is this knowledge? To whom?
- Toward what objectives should the solution of the problem be oriented?
- Am I solving this problem in behalf of a decision-maker? If so, does he agree with my statement of the problem and the objectives to be attained?
- What form of solution is acceptable to me? To the decision-maker?

In system design research problems and objectives have a tendency to change. It is therefore wise for the experimenter to periodically return to the above questions and probe for desirable alterations to previous answers. During the process of providing a complete definition of the problem the experimenter must consider the following factors:

- Information known or assumed about the system, its parts, functions, properties and relations.
- System variables which affect performance.
- Variables in the environment which interact with the system.
- Variables which can be controlled.
- Variables which are not controlled and the mechanisms by which their values are set.
- Measures of attainment of desired objectives.

It is usually the case in complex information systems to present several measures of performance so that performance may be represented by a vector $P = (M_1, \ldots, M_N)$ where the M_i are the various measures. It is useful whenever possible to derive an objective function $V = f(M_1, \ldots, M_N)$ which yields an overall measure of system performance. Such an objective function may be justified mathematically or offered to the decision maker as a reasonable criterion to accept. Since the M_i are presumed to be functions of relevant system variables V_i, it is therefore reasonable to represent the objective function by V

$$V = F(V_1, \ldots, V_k)$$

The following represents some of the ways in which an optimization problem may be formulated for an experiment.

- It may be known or tentatively assumed that there exists a unique vector $(V_1^*, \ldots, V_k^*)$ which yields an optimal value V^* for the objective function F.
- The objective function V may be accompanied by constraints. For example we may require that $M_i \geqslant A_i$ where A_i are prescribed values.
- It may be known or tentatively assumed that the optimal value V^* is not attained for a unique vector and that the equation $F(V_1, \ldots, V_k) = V^*$ defines an "optimal surface."

The experimenter may not know initially how to best define his policy of optimization. He may begin with the first situation above and find that the optimum solution occurs at a place where one of the M_i is disappointingly low. Thus he finds that the objective function above is not a completely satisfactory criterion and he is led to formulate the next situation. However, he discovers that the optimum is no longer unique and he is really in the third situation.

Another difficulty which frequently arises is that the experimenter does not in fact know how the M_i depend on the system variables V_i. He must therefore design one or more experiments dedicated to the task of establishing the true causal relationships and determining the functions which relate the M_i to the relevant V_i. These considerations help to show why the experimenter must frequently conduct preliminary experiments and why a sequence of experiments does not always follow a predetermined path.

Modeling considerations

However powerful a technique computer simulation may be it must be remembered that it is not the only approach to solving problems in the design of information systems. The experimenter should consider all possible alternative approaches before deciding to embark on simulation experiments. The analytic approach makes use of mathematical models and mathematical methods of solution. This approach, when feasible, is frequently more economical than other approaches and provides powerful tools for predicting system response as a function of relevant system parameters. Mathematical analysis has been difficult to apply to systems which involve multiple interacting queues and complicated scheduling rules. However, progress is being made in developing mathematical tools in the areas of queuing theory, mathematical programming and network flows which will be applicable to the solution of more difficult problems. Significant progress will be made when researchers learn how to effectively combine simulation and analytic techniques. One method which has been successfully employed used mathematical analysis in one or more of the subsystems and applied the results to simplify the simulation of the total system.

In construction of a model the experimenter is attempting to achieve a representation which at one and the same time is less complicated than the real system and easier to use for research purposes. He achieves simplicity by representing only those properties which are relevant to the solution of the problem. In very large and complicated systems the experimenter has to find a good balance between the accuracy of representation and the manageability of the model. Practical considerations which enter here include time, cost, personnel, the computer to be used and the nature of the programming system. A variety of simplification techniques exists for this purpose and the experimenter selects from among them. It is important for the experimenter to make these techniques explicit and record them so that any unsatisfactory results from the experiment can be investigated in the light of the simplification techniques employed. In achieving the desired degree of approximation the model builder may use two different approaches:

- Start with the simplest version possible and "sneak up" on the final model by adding appropriate refinements until satisfactory results are achieved.

- Start with a model which incorporates all the details and carve away the least relevant features one at a time checking to see that satisfactory results can still be achieved.

The author favors the first approach for the following reasons:

- It is easier to begin with simple situations and permit the model builder to "warm up."
- The experimenter receives earlier results and may use them to sharpen his understanding of the system's behavior. The feedback is useful in further modeling.
- It is easier to implement because computer programs are more easily debugged in stages.
- It is easier to analyze results and evaluate the experiment with a simpler model.
- It appears to be a more economical way to arrive at the goal.
- It simplifies project management problems.

Finally, we must acknowledge that there exists no firm procedure by means of which the experimenter can construct a satisfactory model. Model-building remains a highly-sophisticated engineering art and its successful practice depends on the skill, judgment and experience of the experimenter.

Validating the model

Validation procedures are employed by the experimenter to answer the following questions:

- Does the model function as it was designed to function?
- Does the model yield simulation histories which are valid representations of the system under study?

The first question, perhaps more properly referred to as model testing, is relatively easier than the second. The distinction between the two types of questions can be illustrated with a simple example. Suppose the model is designed to represent a computer system in which jobs to be executed by the computer have different priorities. Suppose further that the model is designed so that 50 per cent of the jobs which arrive are priority 1 jobs. To answer the first question the experimenter checks to see if the model generates job arrivals in such a way that 50 per cent of them are priority 1 jobs. To answer the next question the experimenter must show that 50 per cent of the jobs which arrive at the real system are in fact priority 1 jobs. In this example model testing and model validation can employ similar statistical techniques. However, the application of these techniques to the two cases will confront the experimenter with very different problems, particularly with respect to the problem of data acquisition.

In model testing the data required is obtained from simulation runs and specifications of the model. The experimenter will check to see that the results from the simulation runs reflect suitably on the assumptions and properites in the design of the model. The relevant variables are checked to see if they lie in the prescribed or predictable ranges. The statistical variables are checked for bias and extremes in variation from the mean. Further statistical tests may be required to verify that the random variables are distributed in accordance with prescribed statistical distributions. The degree of reliability which is associated with such statistical testing will usually have implications with respect to the sampling technique and size of sample. In turn, the size of sample will have implications on the duration of the simulation run. In such matters the experimenter should be competent in the area of statistical testing of hypotheses or be prepared to seek such advice. In gathering data from the model the experimenter should avoid using data generated during initial transient conditions of the model. Sometimes the transient conditions can be eliminated by suitably initiating the model. Other techniques exist for reducing the duration of the transient state of the model. It is generally a good idea to make a number of runs with different values of the model parameters to insure the stability of the model over the parameter space which will be of interest to the experimenter.

In validating their models experimenters face two kinds of situations:

- The model was built to represent a real system which can be observed in operation.
- The model was built to represent a hypothetical system which cannot be observed in operation.

For the first situation the task of validation is relatively simple. The basic approach is to compare the behavior of the model with the behavior of the real system. Historical data collected from the real system will be very useful in making such a comparative analysis. Difficulties may be encountered if the historical data is incomplete or unreliable. A more crucial test is to use the predictive capabilities of the model to forecast the behavior of the system. Here the experimenter can expect difficulties in gathering data, especially if the process of acquiring the data tends to interfere with the normal operation of the system.

In the second situation, which is the one in which the system designer generally finds himself, the problem of validation is most difficult. The model's behavior cannot be compared with an actual system but perhaps some of its subsystems can be so compared. And perhaps there are similar systems with which it may be partially compared. In many instances the designer has to settle for the reasonableness of the model and the degree to which its behavior conforms to his own intuitive understanding of such a system. Even so the model may be efficient for the purpose of making relative comparisons of system behavior with respect to a selection of alternatives such as decision rules. In obtaining relative results simulation offers a special advantage in that all the variables can be controlled so that the same sequence of system inputs can be repeated in each run. Even the random events can be made to repeat so that the variations in the system output cannot be attributed to random fluctuations between runs.

Preparation of the computer program

The design of a computer simulation experiment must interweave into a harmonious pattern the design of a scientific experiment and the design of a computer program. In order to obtain maximum service from the computer the experimenter will incorporate into the program:

- The structure of the model,
- the logical machinery to drive the model,
- analytic and decision making procedures,
- the procedures which generate and record system status information, and
- the routines which assemble, process and arrange the data into final output reports.

The experimenter frequently encounters two kinds of difficulties:

- Exceeding core storage.
- Excessive run time.

The causes of these difficulties are various and the experienced designer learns to anticipate these problems in the earlier stages of designing the experiment. In considering these problems the designer will often judge the desirability of:

- Reducing the size of the experiment,
- simplifying the model,
- segmenting the computer program, and
- different computers and programming systems.

EXCEEDING CORE STORAGE

Large and complicated models will naturally occupy a large amount of storage. When such a model is driven the data generated by the model will sometimes overflow memory. When simplification of the model cannot solve the problem the experimenter may consider running the program in distinct sequential phases, where each phase occupies available core during its execution. The following breakdown has often worked successfully:

- *Input phase*—Initial system status is read in, tables to generate random variates are constructed and exogenous events are generated and stored on magnetic tapes.
- *Simulation phase*—The model is driven using the exogenous event tape as input. System status information is generated and stored on tape.
- *Output phase*—The system status information is input, processed, and analyzed to produce output reports.

Another way to avoid memory overflow is to reduce the duration of simulation. For example instead of simulating the operation of a computer system 24 hours it may be possible to simulate three 8-hour shifts in three separate computer runs. With this approach provision must be made to preserve continuity, *i.e.*, the system status at the beginning of a next shift must be the same as the system status at the termination of the preceding shift.

EXCESSIVE RUN TIME

A computer simulation experiment may sometimes make excessive demands for computer time. The most frequent causes for this are:

- Large decision space.
- Unfavorable time-compression ratio.
- Slow stochastic convergence.

Large decision space

Let the function

$$V = F\,(U_1, \ldots, U_m)\, V_1, \ldots, V_n)$$

represent an objective function in which the U_i represent uncontrolled system variables and the V_i represent the controlled or decision variables. If the designer has an optimization problem to solve, *i.e.*, to find values for the V_i such that V attains its optimum value an obvious approach may be to replicate the simulation experiment as many times as the cardinal number of the decision space, *i.e.*, the number of choices for the decision vector (V_1,..., V_n). With even a small number of variables V_i and a range of 10 different values for each variable the number of replications may become prohibitive. When this happens the experimenter may have to arbitrarily reduce the size of the decision space or use a more sophisticated experimental design such as a fractional factorial design. Other possibilities involve simplification of the model in one way or another.

Unfavorable time-compression ratio

If the computer program is executed in t units of real time and drives the model through T units of simulated time then the ratio T/t is called the time-compression ratio for the program. A computer program which simulates an information processing system and which has a time-compression ratio of 100:1 can provide a simulation of 24 hours of processing in about 15 minutes on the computer which is quite reasonable in cost. Consider now a model in which certain events occur every few milliseconds and other events occur say every few minutes. There is contained in the same model a micro-world of events and a macro-world of events and the time-compression ratio for the model will be governed by the time-compression ratio for the micro-world which in such situations is likely to be quite small, say 1:1 or smaller. To simulate an 8-hour shift with such a time-compression ratio will be very costly. One approach to conquering this difficulty lies in simulating the micro-world of events separately and to characterize the behavior of the micro system so that its effect can be imbedded in the macro system at the macro-level. Other approaches have been used which avoid dealing explicitly with the micro-level (Katz, 1966).

Slow stochastic convergence

Information systems which operate as service systems generally contain queuing subsystems. Models which contain stochastic subsystems generally pass through transient phases before reaching a "steady state." When a model converges to the "steady state" very slowly an excessive amount of computer time is used. Various techniques have been adopted to avoid the long transient period during simulation. When approximate information can be obtained about the steady state the model is initialized so that it starts with an initial state relatively near to the steady state. For example, when queues are involved these are pre-loaded with customers when simulation starts.

Run times are frequently governed by the statistical sampling techniques employed to obtain estimates of random variables. Excessive run time may sometimes be reduced by using more efficient sampling techniques and more sophisticated statistical analysis.

Progress in computer simulation

The last few years have seen a rapid growth in the application of computer simulation to the design and development of new systems. The accumulated experience and successes in these projects have inspired the development of a number of simulation languages for programming simulation applications. Among these GPSS and SIMSCRIPT have attracted a considerable number of users. Improvements have been made in both of these languages and plans are under way to develop new language systems which will provide even more powerful tools for the experimenter. Research effort at IBM and at Lockheed, for example, has been expended to develop languages which are "process oriented" rather than "event oriented." One such language system developed at Lockheed has recently been implemented and is now ready for use in the field.

The system designer and the system analyst are both concerned with two important questions about a particular system:

- How can I accurately describe the system?
- How can I evaluate its performance?

It is clear that progress in systems theory should have important consequences for applications like system design and system analysis. A general theory of systems will undoubtedly imply a general theory of models and this should lead to important consequences for the model builder. System theory is young and it lies in the province of basic research. For the present its importance relates more to future promise than to past accomplishment. The applied scientist should follow its developments and be prepared to test theoretical concepts and evaluate them for their potential utility in system applications.

REFERENCES

1 R. L. ACKOFF *Scientific method, optimizing applied research decisions.* John Wiley, N.Y. 1962

2 WALTER M. CARLSON *Scientists' requirements.* A. KENT and O. E. TALBEE (eds) *Electronic information handling.* Spartan Books, Washington, D. C. pp. 85-91 1965

3 JOHN J. CLANCY and MARK S. FINBERG *Digital simulation languages: a critique and a guide* AFIPS Conference Proceedings Vol. 27, Part 1 1965 Fall Joint Computer Conference Spartan Books, Washington, D.C. pp. 23-36 1965

4 RUTH M. DAVIS *Artificial intelligence applications to military problems* A. KENT and O. E. TAULBEE (eds.) *Electronics information handling* Spartan Books, Washington, D.C. pp. 225-265 1965

5 LOUIS FEIN *The structure and character of useful information processing simulations* AFIPS Conference Proceedings Vol. 27, Part 1 1965 Fall Joint Computer Conference Spartan Books, Washington D.C. pp. 277-283 1965

6 W. H. KATZ *Simulation of a multiprocessor computer system* AFIPS Conference Proceedings Vol. 28 1966 Spring Joint Computer Conference Spartan Books, Washington, D.C. pp 127-139 1966

7 C. McMILLAN and R. F. GONZALEZ *Systems analasis, a computer approach to decision models.* Richard D. Irwin, Incorporated, Homewood, Ill. 1965

8 T. H. NAYLOR, J. L. BALINTFY, D. S. BURDICK and K. CHU *Computer simulation techniques.* John Wiley, N.Y. 1966

9 ALLEN NEWELL *Limitations of the current stock of ideas about problem solving* A. KENT and O. E. TAULBEE (eds.) *Electronic information handling* Spartan Books pp 23-36 1965

The zoo and the jungle – a comparison of the information practices of intelligence analysts and of scientists

by HAROLD WOOSTER
Air Force Office of Scientific Research/Office of Aerospace Research
Arlington, Virginia

"Anyway, the actual workings of the Secret Service, like those of criminal investigation, hold a limited emotional appeal for most people."

KINGSLEY AMIS in *The James Bond Dossier*

PROLOGUE

The zoo and the jungle

The naturalist must always be careful to distinguish between zoo behavior—the behavior of animals in captivity—and the behavior of the same species in the wild state. He should also be continually aware of the effect of the observation upon the phenomenon observed.

In many ways—surprisingly enough, most of them good—the relationship of the intelligence analyst to his supporting information systems may be regarded as zoo behavior. He lives on the same premises; simple operant conditioning of both the analyst and the information system can bring the two into harmony. If he withers from lack, or bloats from excess, of adequate information the zoo-keeper has certain embarrassing immediate problems to face.

The scientist in the state of nature is another matter entirely. He has to be lured to the information system with salt blocks, strange scents and even more peculiar calls. The observer in his camouflaged blind sees only healthy, active specimens. By definition, the inactive are not observed, but can only be inferred.

INTRODUCTION

This paper is an attempt to explore and contrast two related areas—the information requirements of the intelligence analyst and the practicing scientist. "Intelligence" is used in this paper in a highly restricted sense. It is limited almost entirely to that information gleaned from open scientific and technical literature—difficultly accessible, perhaps, but at least in its country of origin, available to the nationals of that country.

Sherman Kent (Kent, 1965) has pointed out that there are areas of the world in which the phrase "overt intelligence" would be regarded as an oxymoron. For, as he writes:

> *"If in fact the Soviets engage in what we of the West call 'intelligence research and analysis' they have another name for it and a name bereft of the cachet of 'intelligence.' It is seemingly inconceivable to them that large numbers of people will be quite overtly engaged in something known as intelligence work, able to inform all and sundry that this is in fact their calling, and obliged to guard with secrecy only those matters having to do with their sources, methods, the foci of their attention and the content of their findings."*

Or, to quote a Russian source (Orlov, 1963):

> *"According to the views of Russian officers, it takes a* man *to do the creative and highly dangerous work of underground intelligence on foreign soil; as to the digging up of research data in the safety of the home office or library, this can be left to women or young lieutenants who have just begun intelligence careers."*

Or again, to quote Gen. William Donovan (cited in Orlov, *ibid.*):

> *"Intelligence is not the mysterious, even sinister thing people think it is, but is more a pattern of pulling together myriad facts, making a pattern of them, and drawing inferences from that pattern."*

This is the intelligence pattern (*razvedka* or no) to be discussed in this paper.

The information practices of scientists

Study of the information habits of scientists has almost become the dermatology (the patient usually does not die, and he never really gets well)* of the information sciences. One does occasionally hear of the experiments which fail, as in the letter in *Science* (Dray, 1966) recommending that the Information Exchange Group on Immunopathology be discontinued for nine apparently valid reasons. Failure to meet user needs is at least one of the many reasons why information systems totter and fall (see the author's *Post-Mortems Can Be Fun*) (Wooster, 1965a), but studies of user requirements seem to go on forever.

The International Conference on Scientific Information, held in Washington in 1958, provides a convenient starting point to survey the literature in the field. Törnudd reviewed most of the information use studies published before 1958. Her bibliography cites 69 articles. (Törnudd, 1958)

That same year Mortimer Taube (Taube, 1958) attempted an evaluation of the then total existing literature of use studies, first pointing out that of the 69 studies cited by Törnudd only 15, dating from 1955, had not been listed in previous compilations by Shaw (Shaw, 1956), Henkle (Henkle, 1956) and Stevens (Stevens, 1953) dating back to 1953. Taube added to the 69 Törnudd citations the 12 papers on the subject given at the Conference itself, bringing the total to 81.

Menzel, in 1960 (Menzel, 1960) attempted to collate tables of findings from more or less comparable studies. His bibliography, admittedly more exclusive than those cited above, lists 26 articles.

In 1964, Davis and Bailey (Davis and Bailey, 1964) compiled an annotated bibliography of 438 "use studies" containing virtually every study of significance up to 1963. One reviewer (Paisley, 1965) has pointed out that "This is a difficult source to use because of the high proportion of chaff; about 350 of the citations are commercial periodical readership studies and library school research exercises," which would seem to leave a total of 188 valid papers.

This same reviewer (Paisley, *Idem*) cited some 75 relevant papers in his review of the literature.

Herner and Company in 1966 (Herner and Co., 1966) broadening the scope to include physicians, were able to locate "several hundred" papers dealing with information patterns in science, with 110 dealing specifically with the problem in biomedical sciences.

Some idea of the flavor of the field may be obtained from two papers. The first is the "classical" 1958 paper (the author is tempted to define classical as having been around long enough to pass into the Russian literature and then back out again via an English journal, the original source being lost in the process) of Russell Ackoff, then heading the Operations Research Group at Case Institute of Technology (Ackoff, 1958).

Ackoff et al. made 25,000 observations on more than 1,500 chemists chosen as a representative sample of the U.S. in 1957-1958. Out of a 90-hour week (including week-ends and evenings) the average chemist spends 16.5 hours in scientific communication, 10.4 hours with equipment, 6.7 hours in business communication, three hours in data treatment and all of two and one-half hours in thinking and planning. The average chemist spends more time in communicating (23.2 hours a week) than in all the rest of his professional activities combined (15.9 hours per week).

More recently the "Auerbach study" (Auerbach Corp., 1965) which lasted 16 months and cost almost $300,000 attempted to discover just how the 36,000 scientists and engineers in the Department of Defense actually got the information they needed to do their jobs. They found that half of these either consulted each other or went to individuals' personal files of information to meet their primary needs. In 39 per cent of these cases the particular information requirements were completely satisfied.

Equally discouraging, only half of the individuals interviewed knew of their own information services, which were rarely ever used as a first source of information. About one-fifth of them had never heard of the Defense Documentation Center (or its predecessor, ASTIA); nineteen per cent didn't know of the existence of any of 33 specialized Department of Defense information centers!

The job of the intelligence analyst

In contrast to the almost embarrassing wealth of information about the information requirements, uses and practices of scientists, the author has been able to find only one unclassified paper relating specifically to the information needs of intelligence analysts.

In fairly precise terms, much of the work of the intelligence analyst consists of the beneficiation of fairly low grade ore, or even tailings. For those of you who did not grow up in a mining community, "beneficiation" means to make richer; "tailings" are the residue from previous inefficient refining operations.

*Since the disease of Venus are, at the moment, of more interest to the dermatologist than, one would hope, to either the information specialist or the astronaut, the author's biomedical background compels him to warn of:

". . . the young man from Bombay
Who thought lues just went away
As a result he has tabes,
And bandy-legged babies
And thinks he is Queen of the May."

An inaccurate version of this medical mnemonic may be found in (Gordon, 1957).

To continue the mineralogical metaphor a little further, the sort of intelligence operation which makes the plot for best-sellers (either as paperbacks or proposals) could be defined as "high-grading"—"to steal rich ore from a mine, especially very rich gold ore." High-grading was very much a matter of individual enterprise—an individual with a lenient foreman could stagger out of a mine with several weeks' pay in his pants cuffs. Reworking tailings is a much duller factory process, calling for efficient high volume machinery, figuring profits in pennies per ton, instead of dollars per ounce.

So volume is the first item to keep in mind. One information installation alone, CIRC—Centralized Information Reference and Control, described by Ray Barrett (Barrett, 1965) is designed to process about 8,000 to 10,000 input reports per month, provide a retrieval capability for the total system holdings, and carry out weekly dissemination to a large number of user groups. By way of contrast, the Clearinghouse for Federal Scientific and Technical Information, responsible for the sale of U.S. Government sponsored R&D reports to the general scientific public, adds about 2,300 reports per month, plus an equal number of translations (Fry, 1966).

These reports are then indexed, classified, abstracted, translated, stored and retrieved. These may or may not be treated as conventional library processes, with one major exception—the assignment of a subject classification according to one of several allowable hierarchical coding systems which have become more or less standard throughout the intelligence community. Amusingly enough, at least for one subject area in one agency, the European Universal Decimal Classification, an expatriated form of Dewey Decimal Classification, is used. Not for its merits which, with a few isolated exceptions such as the Engineering Societies Library, have been insufficient to overcome the chauvinism of American librarians but because Russia, in common with most European countries, has standardized on UDC to the extent of printing UDC classification numbers on journal articles. It turns out to be simpler to use the Russian classification system in this particular instance than to reclassify the articles.

Presumably a substantial portion of an agency's manpower resources are devoted to these more or less routine library operations. TDCK, the Netherlands Armed Forces Technical Document Center in The Hague, which resembles an intelligence agency at least to the extent that its preferred output is evaluated engineering reports rather than bibliographies, devotes 35 of its 65 manpower spaces to maintaining the library (Schüller, 1965). The Defense Documentation Center, which performs solely library operations in the sense of this paragraph, uses some 400 manpower spaces to process and retrieve perhaps 3,000 to 4,000 reper month.

Those of us concerned with the processing of scientific information would certainly contend that money spent in improving these operations is "leverage" money, that saving money on library operations is false economy, but I can certainly visualize some jim-dandy arguments on the score of analysts vs. librarians.

All of this has gone on, mind you, before we or the documents ever meet the intelligence analyst. The analyst sits at a desk and reads and writes. He has two major sorts of jobs; keeping *au courant* in a particular subject/geographical/language field—the breadth of the field varying directly with the grade level and the depth inversely—and preparing reports, known variously as "finished intelligence" and "evaluated intelligence."

These reports fall into three main classes:

Class	Synonyms
PAST	
Basic descriptive	Basic research
	Fundamental research
	Basic data
	Monographic data
	Encyclopedic data
PRESENT	
Current reportorial	Current intelligence
	Current evaluations
	Current appreciations
	Cable material
	Hot intelligence
FUTURE	
Speculative-evaluative	Estimates
	Strategic estimates
	Evaluations
	Staff intelligence
	Capabilities intelligence

(Kent, 1965)

The time allowed for preparation of these reports ranges, as one would expect, from the near-academic to the nearer-frantic.

The information from which these reports are prepared usually comes from three sources: the personal memory of the analyst (a practice facilitated by the tendency to minimize the usual scholarly apparatus of citations and foot-notes in intelligence reports); whatever personal files the analyst has looted from the input flowing across his desk and squirreled away against a time of need or obtained by "unsystematically canvassing outside sources of information" (Kent, 1965), and by subject searches of a central information file.

These subject searches range from the highly specific —"What was the rainfall in such and such a place on such and such a date?" to perhaps such general areas as metals *and* stress *and* cryogenics. The analyst differs slightly from the normal user in a greater tendency to use such qualifying terms as "I don't want anything older than 1958" or "I don't want documents concerning American research" or he may even say "Documents must all be unclassified."

I have at times had the impression that analysts tend to ask very broad general questions, preferring to run barefoot through the windrows of paper in search of the needle of truth rather than trusting the information/system library to use a magnet. Tastes differ, of course. Some analysts say that they never use a subject search, since they know perfectly well what data are available in their areas! Others prefer to frame their searches in terms of authors, geographical locations, facilities and personalities. There are even those who have learned to use the indexing system itself as a rather sensitive indicator, spotting trends by unexpected increases or decreases in the use of certain indexing terms.

At last-Confrontation!

It is plausible, if rash, to attempt to compare the information practices of intelligence analysts and scientists in the following table, remembering that the separation is nowhere near as clean as the device of two columns would indicate and that in fact one class of scientist—there is no convenient American term for this class, but the English one, "information scientist" should suffice—is almost indistinguishable in his work habits and information needs from the intelligence analyst.

Table I

Designatum	*Intelligence Analyst*	*Scientist*
Use of oral/informal information	Seldom, outside his own agency	Preferred mode of obtaining current information. Auerbach, (1965)
Attendance at open scientific meetings	Usually vicarious	Avidly (Korchin & Clarke, 1959)
Use of library	Heavy	Light
Preferred place of reading	Desk	Office, laboratory or home—very little in library
Percentage of time spent in information processing	>75%	<50% (e.g., Ackoff 1958)
Languages	English + (1-4)	English ± 0.5
Prefer translations	Usually	Always
Use of foreign sources of information	Up to 100%	Less than 2% (Syracuse U., 1966)
Maintain personal file	Usually, in office	17-100%—in home or office (Jahoda, 1966)
Good Housekeeping Seal of Approval for literature	TOP SECRET	Publication in referred journal (Pasternack, 1966)
Use of microfilm or microfiche	Readily	Only in desperation
Leads to information	Reference services	Gossip, hot tips from friends, scanning (not reading) 5-10 current journals. Little use of abstracting services (Gerard, 1958)
Bibliographic sophistication, e.g., ability to use standard reference tools	High	Negligible to fair
Use of mechanized current awareness services, e.g., KWIC and SDI	Good, if available	Good if chemists, biologists, ACM, and/or NASA contractors. Otherwise, fair to poor. (Sprague, 1965)
Use of extra-mural information sources, e.g., specialized information centers	Good	Good if in in-group; otherwise only if led gently by the hand or touted by a buddy.

Table I, *continued*

Designatum	*Intelligence Analyst*	*Scientist*
Number of full-scale, retrospective literature searches per year	1-12	0-2
Availability of mechanized, automated, computer-based, etc. information systems:		
Past	Poor	Zero
Present	Fair	Poor
Future	Good	Fair
Principal product	Reports	"New scientific knowledge": gossip; journal articles to maintain status; reports to maintain sponsor.
Attitude toward writing	Evil necessity	Necessary evil
Ambience—feed back on products	Anechoic	Resonant
Output processing methods	Early Bronze Age	Late Stone Age
Barriers between author and ultimate consumer	3 to 10	1 to 2; journal referees and/or editor.
Average cost of information services per user	$200 to $2,000 and	$10 to $200. In certain rare instances, e.g., pharmaceutical firms, as high as $1,000.

Discussion of Table I

I would hope that the high face validity—which is a fancy word meaning superficial plausibility—of the statements in this table require only exegesis to convince the doubter that, no matter how wrong I may be about his group (analyst or scientist), I have fairly anatomized the group to which he does not belong. I can provide documentation for most of the statements in the scientist column—the volume of literature is large enough to prove almost any point. Contrarywise, I can document none of the statements in the intelligence analyst column, and welcome any documented disproof.

Use of oral/informal information

This would seem to be the scientist's favorite mode of communication—in his own corridors, by telephone, or at meetings held in pleasant, distant and preferably overseas, locations. It is quite clear from the literature that corridor gossip is the best part of such meetings—that papers are attended only in desperation or inclement weather.

It is my impression that in intelligence processing such oral/informal communications are reduced to writing at the earliest possible stage, and that the strict compartmentalization mandatory because of security tends to discourage casual subject-oriented conversation in-house.

Use of library and place of reading

The frequency of a user's contact with his library/information center tends to be inversely proportional to his distance from it. The curve is not, however, monotonic—as those who have listened to arguments about, say, departmental libraries versus centralized libraries know, there seems to be a forbidden zone, with the lower edge dimensioned in hundreds, or at best thousands, of feet within which a scientist will not travel to get information from a library. The upper edge, at least in days of grantsmanship and easy travel funds, is probably in the order of thousands of miles. The intelligence analyst with a library on the premises has a considerable advantage over the scientist, on, say, a campus like UCLA where the library is too far to walk and parking is almost impossible if he drives. It might even be possible to distinguish between information-oriented and lab-oriented scientists by releasing suitably marked specimens at varying distances from a library.

A good university library can, either from its own holdings or through the informal but highly effective library community, provide the user seeking an identified item with a copy, a reasonably legible surrogate or, in the case of incunabula and similar rarities, identification of the physical location of almost every book and scientific journal ever published. It can also provide a descriptive catalog of every bibliographic item that should be on its shelves. But its subject cataloging of its own holdings, let alone those of other libraries, lags far behind. (Perhaps because books are more fun to buy and hold than catalogs, and especially catalogers.) In effect, the bargain that a library makes with its more sophisticated users is, "If you can tell me what you want, I'll get it for you (if you're prepared to wait) but it's up to you to find out what you want."

The precise converse of this situation may occur in intelligence installations which have concentrated on mechanizing their bibliographic reference system, perhaps even to the point of providing abstracts of documents of interest, without making equally good provisions for providing full-text copies of the documents on which the bibliographic apparatus is presumably based. Analysts become very unhappy when the computer tells them about documents they can't get; many analysts say that 90 per cent of their time is spent in getting their hands on the document they want after they have found out they want it.* Perhaps the implicit promise of delivery should be made more explicit.

Or, if the traditional library resembles the traditional Hollywood librarian, hiding all sorts of interesting physical resources behind a dowdy facade till the last reel, the unwisely mechanized information system may resemble a fan-dancer—all index and no delivery.

Maintain personal files

Jahoda (Jahoda, 1966) in work done for AFOSR has found that 46 of a sample of 75 research workers maintained personal files. This seems to be in accord with values reported in the literature of from 45 to 100 per cent. Wallace (Wallace, 1964) has described the preparation of personal indexes, printed with the aid of computers for professional and administrative personnel at Systems Development Corporation. To the best of my nowledge, most discussions of mechanization of intelligence information processing have concentrated on mass processing at the central facility. It would seem entirely possible that perhaps more computer support could be given in maintenance of individual analysts' files. There must be something better than the present untidy hoards.

Good Housekeeping Seal of Approval

Pasternack (Pasternack, 1966) editor of *The Physical Review,* is the latest journal Brahmin to point out that the primary purpose of the journal is to, and I am paraphrasing wickedly, substitute the value judgments of the referees and editor for that of the individual scientist. "If you read it in *Physical Review,* you can be almost sure it's true—but if it's in a report, you'd better be ready to snort." (Anonymous, 1966)

The scientist given a paper to evaluate would be quite unhappy if all descriptive information were deleted. Before he ever reads it he wants to know who wrote it, where he worked (both laboratory and country), where and when it was published, perhaps even what agency supported the work. There are certain internal tests he can and does apply: Are the curves given without any experimental points or, even worse, do all the points fall precisely on the curves? Does the author recognize and explain any inconsistencies? Are there a reasonable number of references, and is the purpose of including each reference clear, or are they just the window dressing of a pseudo-erudition? (Perhaps the same pseudo-erudition which drives me to mention Goedel's theorem on formally undecidable propositions—that it is impossible to demonstrate the non-contradictoriness of complex systems without going outside those systems.)

Apparently there are times when intelligence finds it necessary to deprive the analysts of these useful clues, and interpose a middleman between the collectors and the analyst:

"The middleman grades the data for reliability of source and accuracy and reliability of content. . . . (He) according to standard practice, is restricted to a very narrow language in making his evaluations. He is permitted to grade the reliability of the source according to the letters A, B, C, D, and the content according to the numbers 1, 2, 3, 4. Thus A-1 would designate a report of unvarnished truth that was straight from the horse's mouth. . . . If the data happen to have come from a document, a newspaper or press release or some such, one school of evaluators simply designates their value with the single word 'documentary.' . . . Often middlemen have no independent line on the reliability of the source, and instead of admitting as much, will proceed to grade the source on the apparent reliability of the content. This movement in vicious circles is neither helpful nor valid." (Kent, 1965)

I am told that, all other factors being equal (or unavailable), the value an analyst places on an item tends to vary directly with the classification of that item.

Use of mechanized current awareness services

Descriptive cataloging, that phase of the process of

*By way of contrast, a worker in a fairly specialized information center in the nuclear field tells me that his time is spent as follows:

Finding and retrieving references	10-15%
Finding and retrieving documents	10%
Retrieving data from the above sources	50%
Comparison, evaluation and report writing	30%

cataloging which concerns itself with the identification and description of books, is a deceptively simple process. Most non-library oriented beginners in the field of information tend to discover descriptive cataloging like Napoleon discovered Russia—the first steppes are easy, but it gets tougher the further you go. The title of a publication is part of its descriptive cataloging. If this title is key-punched, it can be manipulated in various ways to give indexes known as KWIC (Key Word In Context), KWAC (Key Word And Context) and KWOC (Key Word Out (of) Context).

A review by Stevens (Stevens, 1965) found more than 40 examples of KWIC and its congeners as of February 1964. Perhaps 8 of these, especially those produced by Chemical Abstracts Service, Biological Abstracts and, of course, that of the ACM, have passed the test of the marketplace even if they are not, perhaps "the miracle of the decade." (Baker, 1960)

As the co-designer of one such system, WADEX (Ripperberger, Wooster and Juhasz, 1964), I am keenly aware of its limitations—an almost inescapable bulkiness, caused by the necessity of replicating titles as many times as there are significant words, and the miserable inadequacies of titles as written by the average author. Later forms of WADEX, especially WADEX III, and presumably its rivals, have learned to handle enriched titles—primitive subject terms added to the author's original title—and class numbers. I would imagine that any installation, intelligence or no, that key-strokes descriptive cataloging information and adds a classification number might find some form of KWIC index, sorted by class numbers, a cheap and not too nasty method of setting up a current awareness system.

A full Selective Dissemination of Information system is something else again. Luhn's original concept (Luhn, 1958, 1961) assumed that both documents and users would be indexed by some form of coordinate indexing, and that the abstracts chosen by machine matching of terms would be distributed to individual, named users. It is my impression that there is a growing tendency to drop this personalized service and substitute the distribution of sets of abstracts to classes of users. Barrett, for example (Barrett, 1965), in describing CIRC, says that: "A user profile is a list of topic tags, or descriptors, which describe the scope of a user group's interest. Note that I said *group*. All of our dissemination is based upon unit profiles. We discovered early on that individual profiles had a very high degree of duplication, and that it appeared more economical to talk in terms of a profile serving a unit rather than an individual. Such a unit might have two, three, five, or even 10 people in it working on closely associated subject areas."

I see no reason why a perfectly good SDI system could not be made to work based completely on an hierarchical subject classification system, assigning classification numbers to documents and to user groups.

Ambience—feed-back on products

Perhaps the principal difference between the intelligence analyst and the scientist lies in their ambience. A scientist lives in a highly resonant environment. He talks to his peers informally, he presents papers at meetings, he publishes papers and receives reprint requests, people write him letters of praise or otherwise. If he is as much of a schnook as most of us are, he judges other people's papers and bibliographies by whether or not they have cited his own papers. (Advanced cases read other's papers as they read the newspapers, only instead of turning to the comics they turn to the bibliography.) Certainly in the village community that science was, if not in the concatenation of conurbations that it has become, the scientist continually received feed-back on his work.

Not so the analyst. He finishes writing a report, turns it in to his supervisor, has it bounced once or twice on general principles, OKs the final copy and turns it loose. And that's that. Period, paragraph. Time to start a new report. If he has been allowed to retain a strong prose style and a gift for felicitous phrasing, he may recognize sentences or whole paragraphs of his own in reports prepared by others, but will search in vain for quotation marks or proper bibliographic citation. He knows he wrote it, but so what?

Once upon a time the old Air Research and Development Command had a problem like this. Scientists and engineers working on classified projects envied their unclassified colleagues who got to present papers at meetings. ARDC's solution was the invention of the ARDC Science Seminar. Held under proper security safeguards, it provided a classified forum for classified papers, with perhaps a trace, but only a trace, of Air Force hoop-la in panels of judges, and awards for the best papers and best presentations.

I'm not quite sure that it takes a Herman Kahn to think of the unthinkable, and visualize intra- and perhaps even (shudder) inter-agency competitions for the best intelligence reports. We had amazing luck with a similar problem in our workshop on "Working with Semi-Automatic Document System" by inviting only those people who never gave papers at meetings, but stayed home and did the work while their bosses took the credit. To the best of our ability it was an all-Indian conference. (Those chiefs who muscled their way in usually wound up out of harm's way in a session on system design and evaluation.)

I understand that there are two classified scientific-technical journals in the intelligence field with the usual editorial paraphernalia and that one of them in fact does award yearly prizes for the best papers. This is fine as far as it goes, but publication without reader feed-back (the average author of the average article in a scientific journal usually receives 10-15 reprint requests; that same article is unlikely to be read by more than 7 per cent) tends to be a fairly harmless form of solitary vice. In presenting scientific results, as in several other matters, there really is no substitute for a live audience.

Output processing methods

As I have pointed out elsewhere (Wooster, 1965b, c) the basic problem in technical writing is to put it off till the last possible minute, meanwhile building up vast untidy heaps of reference material and the requisite nervous tension, and then converting sources and tension into finished manuscript as quickly as possible.

Whatever the potentialities of reactive typewriters, little is actually being done to ease this private ordeal. I tend to regard those who do not know how to use a typewriter as hopeless. I am sure that anyone who really wanted to curb the "information explosion" could cut the production of papers in half by outlawing lined pads, ball-point pens and paleographic secretaries. I am indebted to Project INTREX (Overhage, 1965) for pointing out that the average scientist does not know how to type well enough to use a computer console adequately for information retrieval. I am not overly fond of solution which presupposes dictating—such as the long anticipated voice-operated typewriter. If *cacoethes scribendi* is bad, *cacoethes loquendi* is 10 times worse.

Given a candidate who is willing to learn how to type, though, it should not be impossible to devise an inexpensive writer's work station, complete with keyboard, display screen, and my own invention, the "plagiarist's pencil," which hangs alongside the light pen but is actually a print reader, with the capability of tracing passages in text and transferring them to my manuscript.

And while you're at it, be sure to put wheels on it. I want something that can be wheeled into my cubbyhole when I need it, and rolled back to the storeroom when I don't. And the last thing in the world I need is a large, expensive, monocular conscience glaring at me when I'm not ready to start writing. And, since creation should be at least as solitary as procreation, the last thing in the world I want to do is to load my bookshelves onto a wheelbarrow and trundle them off to a brightly-lit central facility with one-way glass windows to show me off to visitors whenever I write a paper.

SUMMARY

Both the scientist and the intelligence analyst are concerned with converting the information from scientific and technical documents into written manuscripts. Although the scientist devotes a large percentage of his time to information processing this is only incidental to his overt goal, gaining new knowledge and insight, and his covert goal, enhancing his stature in the scientific community by peer group recognition. The scientist, *qua* information processor, is an untrained, part-time amateur. The intelligence analyst is a trained, full-time professional. The scientist deals largely with unwritten informal information sources; the analyst is usually confined to formal, written sources. The analyst labors under certain handicaps which are inescapable concomitants of the information with which he deals and the uses to which it is put. Mechanization can provide a partial, but only a partial, amelioration of some of these handicaps.

REFERENCES

1 R. ACKOFF *An operations research study of the scientific activity of chemists.* Un-numbered NSF report, Office of Science Information Services November 1958

2 Auerbach Corporation *DOD user needs study.* Philadelphia, Pennsylvania Vol. I, AD 615-501; Vol. II, AD 615-502

3 D. R. BAKER *Statement in U.S. Congress, Senate Committee on Government Operations.* p. 169 1960

4 R. BARRETT CIRC—*Centralized Information Reference and Control.* In Proceedings of the workshop on working with semi-automatic documentation systems. AFOSR/SDC report prepared under contract AF 19 (628)-3418) AD 620-360 May 1964

5 R. A. DAVIS and C. A. BAILEY *Bibliography of use studies.* Drexel Institute of Technology Graduate School of Library Science Philadelphia, Pennsylvania 1964

6 S. DRAY *Information exchange group no. 5.* Science pp. 153, 694-695 1966

7 B. FRY *Personal communication.* 1966

8 R. W. GERARD *Mirror to Physiology;* A Self-Survey of the Physiological Sciences. American Physiological Society, Washington 1958

9 R. GORDON (pseud.) *Doctor in love.* Doubleday, New York p. 46 1957 The author, who should know better, has substituted "rabies," a highly improbable sign of advanced Casanova's infection, for "tabes."

10 H. H. HENKLE and M. EGAN *Ways and means in which research workers, executives and others use information.* In Documentation in Action Reinhold, New York pp. 137-159 1956

11 Herner and Company *A recommended design for the United States medical library and information system. Vol. II, Background studies.* Prepared for the National Science Foundation under NSF Contract NSF-C-442, for the Office of Science and Technology January 1966

12 G. JAHODA, R. D. HUTCHINS and R. R. GALFORD *Characteristics and use of personal indexes maintained by scientists and engineers in one university.* American Documentation Vol. 17, pp. 71-75 1966

13 S. KENT *Strategic Intelligence for American World Policy*. Archon Books, Hamden, Conn. 1965 Essentially a reprinting of the first edition, copyrighted 1949 by Princeton University Press.)

14 S. J. KORCHIN and K. S. CLARKE *An evaluation of a physician education program of the Chicago Heart Association*. July 1959

15 H. P. LUHN *A business intelligence system*. IBM Journal of Research and Development pp. 314-319 October 1958

16 H. P. LUHN *Selective dissemination of new scientific information with the aid of electronic processing equipment*. American Documentation Vol. 12, pp. 131-138 April 1961

17 H. MENZEL *Review of studies in the flow of information among scientists*. Columbia University Bureau of Applied Social Research AD 400-686 1960

18 A. ORLOV *Handbook of Intelligence and Guerilla Warefare*. University of Michigan Press, Ann Arbon 1963

19 K. OVERHAGE . . . *INTREX: Report of a Planning Conference on Information Transfer Experiments*. MIT Press, Cambridge, Mass. 1965

20 W. J. PAISLEY *The flow of (behavioral) science information—a review of the research literature*. Institute for Communication Research, Stanford University November 1965

21 S. PASTERNACK *Is journal publication obsolescent?* Physics Today Vol. 19, pp. 38-43 1966

22 E. A. RIPPERBERGER, H. WOOSTER and S. JUHASZ *WADEX. Word and author index, a new tool in literature retrieving*. Mechanical Engineering pp. 45-50 1964

23 J. SCHULLER *Personal communication*. 1965

24 R. R. SHAW *Studies on the use of literature in science and technology*. In Pilot study on the use of scientific literature by scientists. National Science Foundation, Washington

25 R. E. STEVENS *Characteristics of subject literatures*. ACRL Monographs No. 5-7 pp.10-21 January 1953

26 R. H. SPRAGUE, JR. *A comparison of systems for selectively disseminating information*. Bureau of Business Research, Graduate School of Business, Indiana University Indiana Business Report No. 38 Foundation for Business and Economic Studies, Indiana University, Bloomington 1965

27 M. E. STEVENS *Automatic indexing: A state-of-the-art report*. National Bureau of Standards Monograph 91 1965

28 Syracuse University Research Institute *A study of the frequency with which Russian, French, and German scientific articles are cited in selected American scientific journals*. Prepared for NSF under Grant NSF-G-19275.1 and Grant GN 501 June 1966

29 M. TAUBE *An evaluation of use studies of scientific information*. Documentation, Inc. AFOSR TN 58-50, AD 206-987 1958

30 E. TÖRNUDD *Study on the use of scientific literature and reference services by Scandinavian scientists and engineers engaged in research and development*. Proceedings of the International Conference on Scientific Information pp. 19-76 1958

31 E. M. WALLACE *Experience with EDP support of individuals' file maintenance*. In Proc. of the Am. Doc. Inst., Parameters of Information Science American Documentation Institute, Washington, D.C. 1: 259-261 1964

32 H. WOOSTER *Post-mortems can be fun—the cost analysis of information systems*. Library Journal pp. 2968-2973 July 1965 (a)

37 H. WOOSTER *Possible effects of automatic information handling on technical writing and publishing*. In Papers from the 5th Seminar "Practical Publications," Huntsville Chapter, Society of Technical Writers and Publishers April 1965 (b)

38 H. WOOSTER *Writers 3, Robots 2—The future of technical writing*. Paper given at the 5th Seminar and Technical Art Show, Society of Technical Writers and Publishers Huntsville, Alabama April 1965 (c)

TEXT PROCESSING SYSTEMS

SAMUEL N. ALEXANDER, *Chairman*

Session Participants:

DAVID G. HAYS
HAROLD BORKO
WILLIAM B. KEHL

Acquisition, archiving and interchange

by David G. Hays
The RAND Corporation
Santa Monica, California

INTRODUCTION

In the field of computing, the word "input" is used ambiguously. It can mean the work done by keypunch operators and others who prepare programs and data for input to a computing system, but it can also mean transfer of information from cards or magnetic tape into a storage device. In the latter case, input is part of a cyclic process that can go on indefinitely with little or no human intervention; processing yields results that are stored outside the computer temporarily, then brought back. "Acquisition" is our term for conversion of information from a form designed for human consumption to one designed for machine input. It is a difficult area of automatic language processing, and the difficulties experienced with acquisition schemes are sometimes resolved at the cost of making problems for those who would archive or interchange files of text. (Each element of information in a bibliography, a subject heading list, etc., is a small unit of text.)

Keyboards and conventions

Given a block of text in the form of typescript or print, acquiring it for automatic processing requires some device. The two main classes of acquisition devices are those actuated by keyboards and those that automatically decode the signal from a photoelectric scanner. Let us examine some problems encountered in working with keyboard actuated devices; with a scanner, they arise in more tedious versions.

If the designer of an acquisition system could begin by writing specifications for the device he wanted, without regard to cost, most of the difficulties inherent in the process would disappear. To begin with, he would specify production of paper copy, to be used for proofreading, simultaneous with production of tape or some other recording for computer input. He would specify a large set of character shapes, each shape to be avilable in several sizes, weights, and so on. He would specify that the machine operate line by line, ing and superscripting would be provided, and for the construction of complex formulas the operator would ator could produce when necessary. Thus, subscripting and superscripting would be provided, and for the construction of complex formulas the operator would be able to place characters anywhere on the page.

No device with all these characteristics is on the market now, although some of them have been obtained by modification of standard devices. Sammet (1966) mentions several devices modified to permit easier input of mathematical formulas. (Martin, 1965; Minshey, 1963; Davis and Ellis, 1963). Urdang (1966) presents a keyboard adopted for typing dictionary copy; the uppercase letters are replaced with special characters, so that capitals must be indicated by typing in red. With two cases and two colors, this device has four characters per key, but the proofreader must remember that a red per cent mark is a lowercase Greek sigma. Kuney, Lazorchak, and Walcavich (1966 a, b) also modified their input devices in some respects. Still, it is hard to imagine a way of obtaining the variety of the printed page with a device that produces paper copy by means of engraved characters and mechanical action. If the two uses of paper copy—immediate feedback to the typist, and subsequent proofreading—are separated, the first can be served by a cathode ray tube display, on which more varied characters and arrangements can be created, and the second can be served by proof copy generated somewhere other than the typist's device.

Even if the designer could specify such a device, he would have to specify the training of his typists in addition. As the keyboard and its ancillary controls become more complex, the training problem increases, and brings in two new difficulties. If the necessary

---

*Any views expressed in this paper are those of the author. They should not be interpreted as reflecting the views of The RAND Corporation or the official opinion or policy of any of its governmental or private research sponsors.

training is too rigorous, many candidates will be unable to master it. And few, if any, will be able to attain the keyboard speed of the ordinary typist working at the ordinary keyboard.

The practical designer, therefore, adopts a different approach, doing the best he can with commercially available devices (perhaps slightly modified) and the skills he can hire (perhaps slightly enriched). If he must deal with the Cyrillic alphabet for one job, and with the Greek and Roman alphabets, using various sets of diacritics, for other jobs, he adapts his machine to each job instead of trying to adapt it to the whole flow of jobs once and for all. Thus, he can purchase a device equipped with Cyrillic type, and hire trained typists for the flow of Russian text; he can buy other devices, and hire other typists, for different flows. If he purchases a device with interchangeable type, then only a part of the device has to be replaced in going from one job to another. If his machine does not produce paper copy, changing from Roman to Cyrillic is only a matter of changing conventions—no hardware alterations are made, but the interpretations of the encoded characters transmitted to the computer are altered.

One set of conventions is not even appropriate to the full range of jobs to be done in one alphabet. Thus, for example, both French and English are typed in the Roman alphabet, but with a limited character set it would be unwise to establish the French diacritics on the keyboard on a permanent basis; rather, by convention, they should occupy certain positions when French is typed, but those same positions should be used for other characters when English is in process. If a given manuscript contains special marks, such as those used in poetic scansion, conventions must be adopted for that one manuscript; it would be most foolish indeed to keep those marks on the keyboard after the one manuscript had been completed. If some of the text to be treated contains bold face, a conventional way of indicating a bold-face segment must be adopted, but the character used for that purpose can have a different use when text without bold face is being acquired.

Normalization

If we now consider the case of a laboratory in which the conventions for use of keyboard actuated devices vary as freely as this, we can see that archiving will pose grave difficulties. In a sense, each batch of text has been acquired in a distinctive encoding scheme. There is not even a closed set of encoding schemes, since a user may appear at any time who requires a set of conventions different from any set used in the past. If a library of all text acquired by the laboratory is maintained permanently, the encoding conventions for each unit stored must be retained because without them text is unreadable. All processors working on text must be constructed in variant forms for the different encoding schemes, or else adapted by means of parameters to the many possibilities.

The current situation with regard to interchange of textual materials among research centers or among operating entities such as libraries is about like the situation in our hypothetical laboratory. The university librarian will discover, not too long from now, that many flows of bibliographic information are available on magnetic tape. Each is encoded in a way that was considered appropriate to his purpose by some designer. the university librarian—and the library's programmer—may find that all the encoding schemes are about equally acceptable, but no one is sufficient. The library requires all of the flow of information, and for many purposes and in many libraries it may be considered necessary to merge the several flows.

It should be self-evident that universal adoption of the American Standard Code for Information Interchange would be no more to the point than universal adoption of a single brand and model of paper-tape perforator. It is not at the level of the ASCII, or the perforator as a physical device, that the designer solves his major difficulties. He must deal with a far greater range of characters than ASCII code, or a paper-tape perforator, is designed to handle. The large set of characters is essential for satisfactory representation of the content of texts. What is required is a uniform way of dealing with indefinitely increasing variety of characters and arrangements of characters.

To return once more to a hypothetical laboratory in which language processing is performed on the grand scale, the laboratory could have a single encoding scheme encompassing all characters known to exist in text of interest, and expansible to include any characters discovered in the future. The encoding scheme would take into account variations of size and weight, features of arrangement, etc. In order not to reimpose the same old difficulties on the acquisition system, the laboratory could have a program with which to mediate between acquisition encoding and archival encoding. Various conventions would be adopted, as described in earlier paragraphs; but the output of the keyboard actuated device would not be considered the archival form. An acquisition program would work on the recording produced by the typist, taking into account all temporary alterations of hardware and all special conventions adopted for the given manuscript. This program would yield an archival tape in which every character, and every feature of arrangement, was immediately distinguishable from all other characters and features of arrangement found in all of the manuscripts ever acquired. Thus, any program written for processing text in this laboratory could be written in a single version, taking the archival form as input, and therefore capable of operating on the entire archive with no special adjustments.

As for the problem of interchange, it is clear that those who use a common archival format can exchange material without much difficulty. Laboratories that have an acquisition program of the kind suggested can accept information in nonstandard encoding by adjusting the acquisition program to the characteristics of the outside source. Thus, the acquisition program can facilitate collection of material from the publishing industry, and from all other uncontrollable sources.

The RAND acquisition program

To be of practical use, the acquisition program must be adaptable to a wide variety of input devices and conventions for using them, since the whole intention is to vary the conventions as often and as much as necessary to make the keyboard job easy. A program of this kind has been written at The RAND Corporation (Graves, et al., 1966). It need not serve forever, although it is in practical use now; it should be redesigned and rewritten as soon as ways of making it more convenient become clear.

The user of the RAND program designates certain characters on his input device as shifts. Naturally, the shifts defined by the hardware of the device in use are ordinarily among these. For each shift, the user provides a complete table of interpretations of the punchable characters. Thus, many changes of character sets, by alternation of hardware or by change of convention, amount to nothing more than changes in one of these tables.

Sometimes it is desirable to expand the character set by allowing for strikeovers. Thus, overstriking a lower case *c* with a slanted line makes the conventional symbol for "cents." The RAND program allows for such conventions by moving a pointer backward as well as forward through an array of character positions, and allowing for combination of a character representation already stored with a new one struck at the same position. Of course, underscoring and diacritics are readily treated in this way.

Urdang's keyboard provides overstriking for diacritics and underscoring, but his conventions call for some words to be treated as underscored (and ultimately set in a special type face) when the typist underscores only the initial letter. This convention makes the character a shift for the RAND program; succeeding characters would be coded as in the appropriate type face. This convention also makes the spacebar a shift, putting the system back into normal shift.

In core storage, the RAND acquisition program sets aside a region as a page array. Like a sheet of paper being fed through a typewriter, this region is divided into rows and columns. In a machine with 36-bit cells, one full cell is reserved for each character position. As each row of characters is filled, a test can be performed to determine whether an independent multiline segment of input—a "page"—has been completed. The input is therefore processed line by line until a page is filled, whereupon analysis and output processing of the page is performed.

In many input streams, functionally distinct elements are differentiated by position and typographic features. Thus, a chapter title may be distinguished by size and face of type, by blank space above and below, by centering horizontally on the page, and so on; or a chapter title might be distinguished rather by inclusion of the word "chapter" flush left at the top of the page, with a number following and a string of underscored characters completing the line. These format features can be interpreted by two additional parts of the RAND acquisition program. The formatter makes straight-line cuts vertically or horizontally on the page, dividing the page into boxes. If the input is simple enough, the formatter alone can isolate all functionally distinct elements. For example, a bibliorgaphy format might consist of double-spaced elements each separated from those before and after by extra double spaces. In some interesting streams of material, at least certain kinds of boxes will have to be analyzed by a parser. A parser can use features of individual characters, for example whether a character position is blank, contains a letter or figure, contains underscoring or certain punctuation marks, and isolate—for example—author, title, and place of publication in a citation. Our parser produces a description of a specific string of text by referring to a grammar. The identity of a subsegment is determined only when identification of the subsegment will lead to a proper interpretation of the structure of the entire stretch.

After the formatter has established box outlines and the parser has analyzed whatever boxes have complex internal structure, a selector brings out elements with specified function and a recoder converts the internal representation into that required for output. Output may be to a printer, another processor, or an archiving program that writes tapes for permanent storage.

The usefulness of programs like the formatter and parser has been questioned, for example by Buckland. (1963 b, 1965). The information provided by punctuation, type face, and arrangement is sometimes ambiguous and often difficult to disentangle; if the original typist puts in explicit signals to identify elements of the text, the decoding job is easier. Formatting becomes an output problem For example, Kuney, Lazorchak, and Walcavich (1966 a, b), use code symbols like α-title; these symbols are typed in red, and interpreted as macros calling for certain type faces and sizes and certain positioning. Functional coding has the advantages propounded for it, but if a large corpus in which functional coding was not used is offered, the formatter and parser simplify the work of making it usable. Furthermore, once functional symbols replace ordinary typography in the typing of drafts, the copy must be put through a computer before it can be distributed; in some operations, the original material must be presentable, even though a machine-readable version is to be retained for some secondary purpose.

If a stream of input does contain functional encoding, it can be handled in one of several ways. The RAND acquisition program could be modified to accept macros that would store a title, for example, not as punched in the input tape but as wanted in the output. Although this would be possible, it seems exactly contrary to the spirit of functional encoding. The other two possibilities begin by treating functional encoding as an extra set of characters. A user of the RAND program often allows one bit to signify that the character is a letter, another to identify it as a numeric digit; he can use a different bit to mark a character as a macro symbol. Then, in writing an archival version of the text on magnetic tape, the user's program can put the functional encoding either in the text stream itself, or in a label record associated with the text.

Each user of the RAND acquisition program must describe his input device and his conventions for using it in detail. The acquisition program that exists today provides several formats for writing different parts of the description. To some extent, the formats provided are different because the program allows more complexity at some points than at others. Other differences are not so easy to explain in rational terms.

The program which develops line arrays in storage by reading input characters and consulting tables is, conceptually, a finite-state machine. Its state at any moment can be given by naming a shift and a character position. It will put an image of the next input character in that position, choosing the image by consulting the table for that shift. The user specifies, for each character, its image in each shift and the state to which it tells the machine to pass. Any input character can cause the machine to go next to any shift, but position movements are of a few kinds only. No change, one space forward, one space back, to the next tab stop, to the left margin, and to the next line are the only ones used at present. In view of the power of subsequent processors, an application calling for greater power at this stage would be hard to imagine.

The program for testing ends of pages allows the user to write regular expressions over line descriptions. Each line description in turn is written as a regular expression over characters. When a line is finished, the test routine examines the last line constructed and the first line description. If the line satisfies the description, the routine moves to the preceding line and the following description.

The lowest level reached in the description of an input device is below that of single characters. Each character position in storage occupies a full cell in which each single bit position can be used for a significan feature, or a combination of bit positions can be used for multivalued features. The lowest level of description is that of the features. On the next level, characters are described by reference to the features present or absent in them. The user may need to refer to a class of characters defined by shared features, or to an individual character for which he specifies just the features needed to obtain a unique description. Next, the user can refer to a string of characters, or a class of strings, and the strings can be placed arbitrarily on the page, or can be specified as a row of characters, extending from left margin to right; a column of characters, occupying corresponding positions on consecutive rows from top to bottom of the page, may have to be described also. The next level of description is a box, made up of all character positions between limits that can be conceived of as straight, horizontal or vertical lines on the page. Finally, there is the level of the page itself.

Revised description format

The RAND acquisition system might serve its users better if a single format were available to them for descriptions on all levels. Let us consider what such a format might be like. The major element will be a procedure, consisting of stipulations, grammar, and command.

One stipulation will identify the class of objects to be defined by the grammar: pages, boxes, columns, strings, or characters. The user must also stipulate the context in which his procedure is to be applied. A page can only be defined relative to a page, but narrower contexts can be specified. A box, row, etc., can be defined relative to another box. Thus, for example, a user might be seeking a row of blank characters, but only require that the row of blanks extend across one box; outside that box, the same row might contain nonblank characters within the limits of the page.

The user must stipulate whether the object to be defined is taken as consisting of rows, columns, or characters. The user must stipulate where the first element of his grammar is to be applied. For example, it might apply to the first row standing in the page array, or to the last one. Thereafter, progress through storage is determined by the grammar itself.

Finally, the user must stipulate whether or not the procedure is recursive. This stipulation has two consequences, which are described below; one has to do with allowable forms of grammar, the other with types of command.

A grammar is a sequence of statements, each with three parts. The first is a description of some unit or class of units. The second and third are action clauses, determining respectively what is done when the unit description does or does not apply to a specified unit in storage.

The description of a unit can be (i) the name of a procedure, (ii) a specification of features, or (iii) an ordinal number. If the procedure is recursive, a statement can contain the name of another procedure on the same level. In a nonrecursive grammar, a description can be given by naming a procedure, but only one of lower level. For example, a nonrecursive procedure for defining a class of boxes can call another procedure that defines a class of rows; only a recursive box-definition procedure can call another procedure that also defines a class of boxes.

If the description of a unit is given by a full specification of features, it uniquely identifies a type. Thus the user can specify, character by character, the entire content of a complete box, row, or column. But a description of a single character might include only one feature, specifying that whatever character occupies the position tested must include, for example, the feature "underscored." Other operators (currently available only in the string-parsing segment of the RAND acquisition system) would permit the user to describe a specific concatenation of named characters, or any arbitrary string over a specified alphabet, or any of a given list of characters, or any character not on a given list, or any string of characters not in a specified list.

Some of the stipulations of a calling procedure can be carried forward to those it calls. If a procedure is stipulated to define boxes in terms of rows, any procedure it calls must be a row-definition procedure. If procedure B is stipulated to define a box relative to box A, then any procedure called by B must be relative to box A also. Hence it should be unnecessary to make all of the stipulations listed above explicitly; some of them should be filled out when the procedure is called.

The action clauses in statements of a nonrecursive grammar are concerned only with movement of pointers. One pointer is in the grammar itself and is transferred from statement to statement. Other pointers are in the storage area; let there be one for each level. A storage pointer moves by characters, rows, or columns. An action clause must therefore specify whether the storage pointer is to move forward or backward by one unit or not to move at all; it also specifies where the pointer in the grammar is to go, but it would be convenient to assume in general that if no movement is stated it goes always to the next statement in order. (If the unit description is an ordinal number, the pointer moves that many units ahead or backward in storage, if possible.)

When a statement is executed, if the unit description it gives is the name of a procedure of lower level, the statements of that lower level grammar are immediately executed. Thus a grammar operating on rows might call a procedure operating on characters; the lower level procedure would take the row pointer at its current position, and start its character pointer at the stipulated place in that row, relative to any limits stipulated. The character pointer moves back and forth under control of the lower level procedure until it is determined that the row does or does not satisfy the procedure; the lower level procedure then returns control to the calling procedure, announcing the outcome "true" or "false."

When a nonrecursive procedure called by the user's program reaches the end of its grammar, it has positioned its own pointer and reached a condition of acceptance or rejection. That is, the grammar is satisfied with the pointer in a certain place or there is no place in the array submitted where the pointer can be placed to satisfy the grammar. In case of failure, the procedure should inform the user's program of the circumstances. In case of success, the nonrecursive procedure makes a cut, dividing the unit to which it is applied into two units of the same level. The user must supply labels for these two units; the command at the end of the procedure is therefore "apply these labels to the units obtained." In the present RAND acquisition system, the user can specify whether the sub unit on which the pointer lies belongs with the first or second of the two units obtained, or is to be excluded from both.

A recursive procedure must be capable of naming many subunits simultaneously. The user calls for application of a procedure to an array; that procedure is satisfied if it can find a way of subdividing the array in such a manner that certain other procedures are satisfied for subarrays; and so on. Thus a recursive segmentation structure develops, and when the whole array is found acceptable to the main procedure all of the subarrays are labeled. The appropriate command to place at the end of a recursive procedure is therefore "apply permanent labels to wanted subarrays." In the current RAND system, the parser is followed by a selector that does just this.

Online operation

If an acquisition system of this kind is to be operated on line, that is, with a direct link between the typewriter and the computer, it should be capable of following the typist line by line or character by character, accepting corrections of errors on the spot and interrupting to point out errors where possible. The typist should be able to delete an entire line rather than let it go into storage. The typist should also be able to replace one character with another, but if overstriking is permitted for the sake of enlarging the character set, the scheme for correction by overstriking must be designed so as not to interfere. If the display device used at the console is a cathode ray tube, and if format analysis is performed during input typing, it would be possible to display the results of formatting to the typist immediately after a page of input is completed. The boxes formed by the formatting program could be outlined on the display tube, so that the typist could make corrections immediately if necessary. The on-line station is.exceedingly convenient for editing, where input is obtained from an uncontrollable source. Errors in the input can be identified if the result of acquisition processing is put on a screen in an appropriate format, and an editor with a suitable cursor can rapidly make the minor changes that are likely to be necessary on account of input errors (Buckland, 1963a).

The complexities of chemical and mathematical formulas can be handled more conveniently with a console having a powerful cursor mechanism than with a typewriter (Cossum and Hardenbrook, 1964; Martin, 1965; Minsky, 1963). If a text contains many equations or diagrams, the typist-s job is inordinately difficult. But let us imagine a system with a typewriter keyboard, RAND Tablet (Davis and Ellis, 1964), and cathode ray tube.

For ordinary text, the ordinary typewriter keyboard is used. The text is displayed on the screen as it is typed. When an equation is to be inserted, the operator moves to a tablet and adjusts the cursor so as to put the material in its proper place relative to the previous text. The typist then draws in the symbols of the equation or diagram, putting them in proper relative orientation. Such an operation of course requires character recognition programs, but character recognition is apparently somewhat less difficult when the cursor can be followed as it moves. Furthermore, recognition errors can be corrected immediately.

Presumably the computer is operating programs for conversion to archival encoding at the same time. What encoding scheme should be used for chemical formulas and mathematical equations is not obvious yet. One scheme would be to encode the graphical shape of the formula, using the standard symbols for superscripting, subscripting, line return, and so on. The other possibility is functional encoding. For this purpose, a table of symbols for summation, integration, and other operations would be stored. With each would be given information about its arguments. The two-dimensional display prepared on the Tablet would be parsed, and what would be stored—presumably under a special flag—would be an abstract representation, perhaps in parenthesis-free notation. Then, as appropriate for information stored in the catalog format, programs suitable to the display devices at hand reconvert from abstract represensation into one more familiar to the reader. Several proposals have been made for functional or abstract representation of chemical formulae. One of them might well be adopted for use here.

Archives are worth establishing only insofar as use can be predicted for them. Some reasonable purpose must be foreseen, but that alone is not enough. The information stored will serve a good purpose only if properly selected and suitably encoded. Encoding suitability depends on standardization, and completeness. This paper has discussed problems of a low intellectual order, but they must be dealt with. A common encoding and formatting scheme is required, and it must be rich enough so that extracts from an archive can be published in a version of very high quality when necessary.

REFERENCES

1 L. F. BUCKLAND *Recording text information in machine form at the time of primary publication* In Automation and Scientific Communication Washington: American Documentation Institute pp. 309-310 1963 (b)

2 L. F. BUCKLAND *Machine recording of textual information during the publication of scientific journals* Inforonics, Inc., Maynard Massachusetts 1965

3 L. F. BUCKLAND *The use of a cathode ray tube display console for editing textual information* In Automation and Scientific Communication Washington: American Documentation Institute pp. 179-180 1963 (a)

*Any views expressed in this paper are those of the author. They should not be interpreted as reflecting the views of The RAND Corporation or the official opinion or policy of any of its governmental or private research sponsors.

4 W. E. COSSUM and M. E. HARDENBROOK *Computer generation of atom-bond connection tables from hand-drawn chemical structures* Proceedings Washington: American Documentation Institute pp. 269-275 1964

5 M. R. DAVIS and T. O. ELLIS *The RAND tablet: a man-machine graphical communication device* RM-4122-ARPA, The RAND Corporation, Santa Monica, California 1964

6 H. J. GAWLIK *MIRFAC—a compiler based on standard mathematical notation and plain English* Communications of the ACM Vol 6, No. 9 pp. 545-547 September 1963

7 P. A. GRAVES, D. G. HAYS, M. KAY and T. W. ZIEHE *Computer routines to read natural text with complex formats* RM-4920-PR The RAND Corporation, Santa Monica, California 1966

8 M. KLERER and J. MAY *An experiment in a user-oriented computer system* Communications of the ACM Vol. 7, No. 5 pp. 290-294 May 1964

9 J. H. KUNEY, B. G. LAZORCHAK and S. W. WALCAVICH *Computer-aided typesetting for the Journal of Chemical Documentation* Journal of Chemical Documentation Vol. 6, No. 1 pp. 1-2 February 1966 (a)

10 J. H. KUNEY, B. G. LAZORCHAK, and S. W. WALCAVICH *Encoding manuscripts as primary input to chemical information handling systems* Presented at the 151st meeting of the ACS, Pittsburgh 1966 (b)

11 W. A. MARTIN *Syntax and display of mathematical epressions* MAC-M-257 Project MAC Massachusetts Institute of Technology, Cambridge, Massachusetts 1965

12 M. MINSKY *MATHSCOPE, part I—a proposal for a mathematical manipulation-display system* MAC-M-118 Project MAC Massachusetts Institute of Technology, Cambridge, Massachusetts 1963

13 J. E. SAMMET *Survey of formula manipulation* Communicators of the ACM Vol. 9, No. 8 pp. 555-569 August 1966

14 L. URDANG *The use of typographic coding in information retrieval in* Progress in Information Science and Technology Washington: American Documentation Institute pp. 193-200 1966

15 M. B. WELLS *MADCAP—a scientific compiler for a displayed formula textbook language* Communications of the ACM Vol. 4, No. 1 pp. 31-36 January 1961

Utilization of on-line interactive displays

by HAROLD BORKO
System Development Corporation
Santa Monica, California

INTRODUCTION

Text processing on a computer is not yet fully automatic. The present state-of-the-art in data and language analysis is such that significant results can best be achieved by a man-machine system in which the computer programs are designed to facilitate on-line interaction by the human decision maker.

Three such interactive programming systems are described in this paper. One is the General Purpose Display System (GPDS), which provides the user with a convenient tool for constructing and manipulating a variety of visual displays. The second is a sentence analysis system (PLP II) that computes an ddisplays the syntactical structure of sentences while the user, viewing the results, makes corrections or selects those analyses that are most satisfactory. The third is an interactive document retrieval system called BOLD which helps the user formulate a search request and displays abstracts of the retrieved documents on the scope so that he can make an immediate decision regarding their relevance.

All three systems operate under time sharing on the AN/FSQ-32 computer at System Development Corporation and use remote stations.

Equipment configuration

To enhance flexibility and man-machine interaction, the input equipment provides a functional overlap so that messages can be transmitted by more than one device. The standard inquiry station consists of a teletypewriter and a cathode-ray-tube display console. For GPDS, this equipment is augmented with an auxiliary keyboard, the RAND Graphic Input Tablet, and the control function selection panel. The equipment is arranged as illustrated in Figure 1.

The teletype

The basic communication device is a standard Model 33 teletype. This teletype is the only means of communicating with the Time-Sharing System and is also used to load the program into the Q-32 computer.

The cathode-ray-tube (CRT) display console and light pen

The CRT is the principal output device. It displays both textual and graphic material with a high degree of resolution, and has controls for intensity, focusing, and display positioning. A light pen is attached and may be used as an other input device to inform the executive program of the selection of a display option or to identify some portion of a display to be manipulated.

The auxiliary keyboard

The auxiliary keyboard has the same key configuration as the teletype and is used in a similar fashion. However, as an input device it is used only with GPDS and it differs from the teletype in two main respects: it produces no printed copy and therefore has no output capability; and it transmits one character at a time to the scope as the keys are depressed, whereas, the teletype transmits a line of characters when the carriage is returned.

The RAND Graphic Input Tablet

The RAND Graphic Input Tablet is an electronic input device consisting of a copper writing surface and a writing stylus. The writing surface is analogous to the display seen at the CRT on a point-by-point basis. Touching the point of the stylus lightly to the writing surface causes the analogous point on the CRT to be illuminated. Pressing the point of the stylus firmly against the writing surface causes the location of the analogous point on the CRT to be transmitted to the computer. The Graphic Input Tablet can be used for the same purposes as the light pen, but it does not have the limitation of being able to respond only to light emitted by the CRT. It is especially useful for producing hand-drawn displays.

The control function selection panel

The control function selection panel, also called the button box, contains 60 pushbutton switches, of which only 49 are currently being used. When a switch is de-

Figure 1—The augmented user station (GPDS)

pressed, an operating command controlling certain aspects of the program is put into operation.

*General purpose display system**

Of the three systems that will be described, GPDS is, as the name implies, the most general-purpose display system. It is used for generating and manipulating CRT displays and is designed to be used by persons with varying degrees of sophistication in data processing and computer technology. Built into the system is an extensive explanatory text as well as error-detecting messages. (Vorhaus, 1965; Guillebeaux, 1966). The object is to help the nonprogrammer user, such as a military commander, a business manager, or a scientist, operate the system.

The versatility of this equipment and the GPDS programs can be illustrated through an actual example of usage by the salary administration section at SDC. To insure salary consistency for similarly qualified individuals in different divisions within the company and to insure compatibility of one company's salary schedule with the industry in general, salary statistics are accumulated and placed in a data bank for analysis by GPDS.

A GPDS process has been written to analyze a subset of a data base, calculate the second order curvilinear regression equations for five specified percentiles (10, 25, 50, 75, and 90), and display the regression curves in graphical format on the scope (Figure 2). This process also has the option of displaying and printing a table of the actual values of the points on the plot as illustrated in Figure 3.

*GPDS was developed by SDC in performance of Contract AF 19(628)-5166 with the Electronic Systems Division, Air Force Systems Command, in performance of ARPA Order 773 for the Advanced Research Projects Agency Information Processing Techniques Office, with partial support from SDC's independent research program. The principal investigators are Alfred H. Vorhaus and Sally C. Bowman.

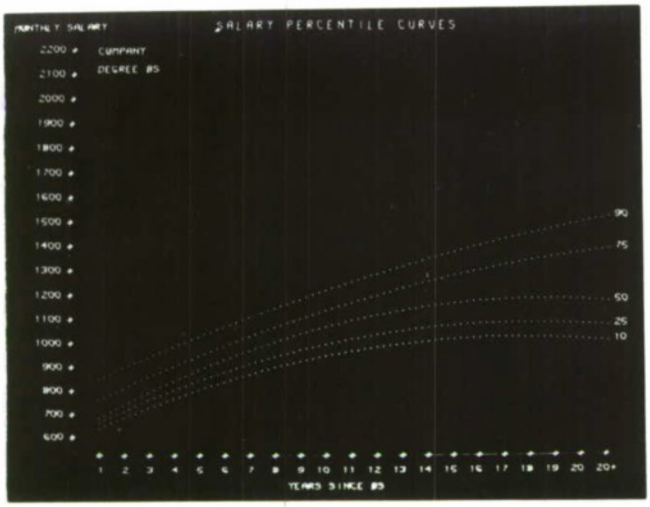


Figure 2—Salary percentile curve

COMPANY
DEGREE BS

YEARS SINCE BS	10%	25%	50%	75%	90%
1	642	669	686	763	851
2	686	716	742	810	898
3	730	763	795	860	942
4	769	804	842	904	986
5	807	845	889	948	1027
6	839	880	933	989	1069
7	872	916	972	1030	1110
8	901	945	1007	1069	1148
9	925	975	1042	1104	1183
10	948	998	1072	1139	1222
11	969	1022	1098	1172	1254
12	986	1039	1122	1201	1286
13	998	1054	1139	1230	1319
14	1010	1069	1157	1257	1351
15	1019	1077	1169	1283	1377
16	1025	1086	1180	1304	1407
17	1027	1089	1186	1327	1433
18	1027	1089	1189	1345	1457
19	1025	1089	1189	1363	1480
20	1022	1083	1186	1377	1504
20+	1013	1077	1177	1392	1525

Figure 3—Tabular display of salary data

The data so far shown have been for a single company. Similar data exist for the industry as a whole. The analyst working on-line with GPDS may be interested in comparing the salary curves of a single company with the composite for the industry as a whole. He can construct and display the composite salary curve, or build a more complex display in which both the individual company and the composite curves are presented simultaneously.

It is important to note that this particular application of GPDS is being used by salary administration people, who, working with a user's manual and some programming guidance, manipulate a large data bank and construct displays quickly and conveniently to answer a variety of questions.

*Sentence analysis**

Information retrieval and automatic question-answering systems require a capability for analyzing statements and questions in natural language. During the past years a number of automatic sentence parsers have been developed but none provide only, nor do they provide all, of the correct analyses. As a result, the Language Processing and Retrieval Staff at SDC has concentrated its efforts on building an interactive system that derives a grammar from manually parsed sentences. The interactive features include the capability for users to change the grammar, to select one of several presented parsings, and to correct errors in the machine parsing.

The system is programmed in LISP 1.5 and operates from an inquiry station consisting of a teletypewriter and a CRT display unit. The program system is called PLP II, since it bears many resemblances to the Pattern Learning Parser previously developed and described by McConlogue and Simmons (1965). This new version, however, contains several unique and interesting features:

- First, the input is in the form of sentences that have already been dependency analyzed. From these sentences, the system derives vocabulary and grammar rules that it applies to new sentences of similar structure, the notion being that it is easier to develop a consistent grammar by having the computer derive its own grammar rules from correctly parsed sentences than to develop the grammar manually by making a linguistic analysis of a large corpus of English text.
- As a second feature, a dependency analysis and a labelled phrase structure tree are produced and the tree structure displayed for each sentence that the program parses.
- In addition to the tree structure, the program produces kernel sentences—one for each sentence string that may be presumed to underlie the surface structure of the sentence (see Chomsky, 1965).

*This research was sponsored by the Advanced Research Projects Agency Information Processing Techniques Office and was monitored by the Electronic Systems Division, Air Force Systems Command under Contract AF 19(628)-5166 with SDC. The principal investigator is Robert F. Simmons.

- Finally, and most importantly, it is an on-line interactive system. The users have the freedom to change the grammar, to correct the analyses the system makes, and to select from the several parsings the one that is intuitively best suited for his needs.

It is our belief that, for some years to come, such a machine-aided approach will be most effective in obtaining correct analyses of text.

The following example will help explain the operation of the program

Sentences are input to the system in the following fashion:

THE	OLD	MAN	SAT	ON	THE	BEACH	Sentence
ART	ADJ	N	V	PREP	ART	N	Parts of Speech
N	N	V	*	*V	N	*PREP	Dependencies

This input is in the form of three strings, where the first is the list of English words in the sentence, the second is the corresponding list of their parts of speech or word classes, and the third is the list showing the word class on which each word in the sentence is dependent. Using the information contained in these three strings, the system augments its existing dictionary with the new vocabulary, word-class items, and dependency rules. A dictionary entry is constructed for each word in the form of a set of 4-tuples containing (1) the word class for the preceding word, (2) the word class of the word itself, (3) the word class of the following word, and (4) the word class on which it is dependent. For the word MAN in the previous example, the dictionary entry would be:

MAN: ADJ-N-V, V

After many such sentences and their analyses have been input to the system and many such dictionary entries have been stored, the system can attempt to parse sentences that have not been analyzed previously. For example, the following sentence was analyzed automatically:

THE BOOK THAT YOU READ IS ON THE TABLE IN THE HALL.

PLP II looks up each word in its dictionary and obtains for each the set of 4-tuple frames that it has thus far accumulated. Generally, this set consists of 3 to 10 such frames for each word. Using the information provided by preceding and following word classes, the system is able to discard most of the frames as being inconsistent with the present context. It is also able to use context clues within the sentence to calculate word classes for words that were not in the dictionary. It does this by predicting, from the word-class contexts of the preceding and following word, the class of the word in question. (A detailed description of the operation of the system is available in Burger *et al.,* 1966.)

The result of this phase of the program is a dependency analysis of the sentence. By means of a display, the user may examine each string in the analysis and correct any errors. He may also request a display of the phrase-structure tree for the sentence (Figure 4). This tree is automatically constructed from the dependency analysis information with the aid of a brief phrase-structure grammar. As is always the case, additions, deletions, and modifications can be made on-line.

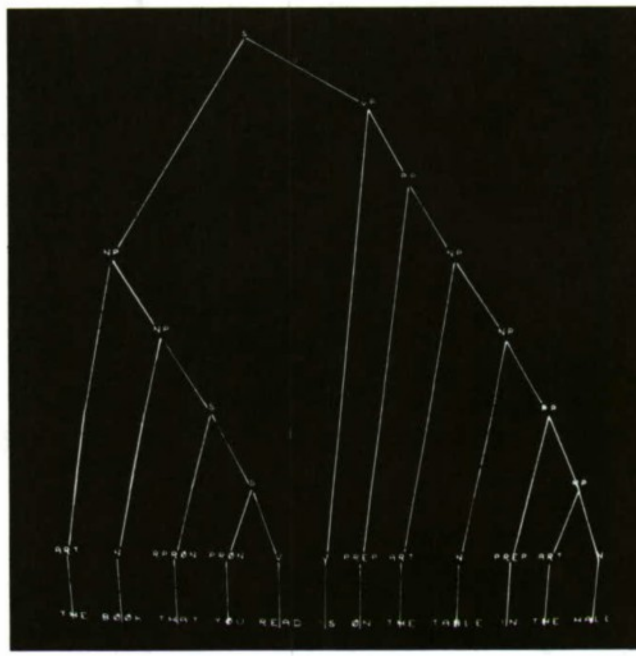


Figure 4—A phrase structure analysis

Although the PLP II system is still in an early stage of development, it has proven to be a valuable research tool. It is particularly useful because the researcher can interact with the parsing system in an on-line mode. He can select one of the parsings offered; he can correct errors; he can augment the grammar; and he can modify sentences to gain new insights into the grammar of the language. Furthermore, he can perform all of these operations rapidly, while his interest with the problem is current.

*BOLD (Bibliographic On-Line Display)**

The third interactive data-base processing program is a document storage and retrieval system called BOLD (Borko and Burnaugh, 1966). The system was designed to allow the user to search for information in a file of magnetically coded and stored document abstracts in much the same manner that he would search through a library. He has the capability of browsing through the collection, examining the documents filed

*This program is supported by SDC's independent research funds. The principal investigator is Harold Borko.

under each subject category, and he is also able to search for documents containing very specific information. If he is not sure of the correct procedures to use, he can receive help and instructions. Most importantly, he is able to state his requests in natural English, for the system would surely fail if the user first had to learn programming before he could retrieve information.

Like GPDS and PLP II, BOLD is programmed for use with SDC's Time-Sharing System. The inquiry station consists of a teletypewriter and display scope with light pen. The programming system has two major modules: (1) the data-base generator program, which builds tables of structured information from a Hollerith prestored magnetic tape, and (2) the display and retrieval program, which retrieves the requested information from the structured data base and displays it on the scope. A technical description of the data base generator and of the display and retrieval subsystems has been prepared by Howard Burnaugh (1966) who wrote the programs.

The data base that is presently being used was obtained from the Defense Documentation Center and consists of abstracts of approximately 6000 documents. For experimental purposes, a subset of these documents is used. The particular tape from which the illustrative examples were derived consists of the first 1745 abstracts and 6883 retrieval terms. The documents are grouped into subject categories organized according to the DDC classification system. However, the program is flexible, and various classification systems and indexing systems can be used.

BOLD is an interactive system, which means that a dialog is established between the user and the system to enable the user to request and obtain relevant documents from the collection. The requests and the system's responses are stated in as close an approximation to natural language as is possible. Ideally, the user with only a knowledge of the English language and a skill in typing should be able to establish a rapport with the machine. Although this ideal may never be fully achieved, a great deal of human engineering skill has gone into the project to approximate it.

After a user has logged in and the data base and program tapes are loaded, the system reports this fact by typing

THIS STATION IS NOW UNDER THE CONTROL OF THE BOLD SYSTEM
OPERATION INSTRUCTIONS R OBTAINED BY THE REQUEST:
INSTRUCTIONS/

Simultaneously, a tutoring display (Figure 5) will appear on the scope. This display defines the 10 light-pen actions that are available to the user.

Figure 5—Initial tutoring display

The user begins operation by flashing the "B" character with the light pen or typing BEGIN/ on the teletypewriter. Commands such as BEGIN, SEARCH, BROWSE, CONTINUE, etc., must be followed by a slash. The user types a question mark to ask for help, and for all other interactions no punctuation marks are used.

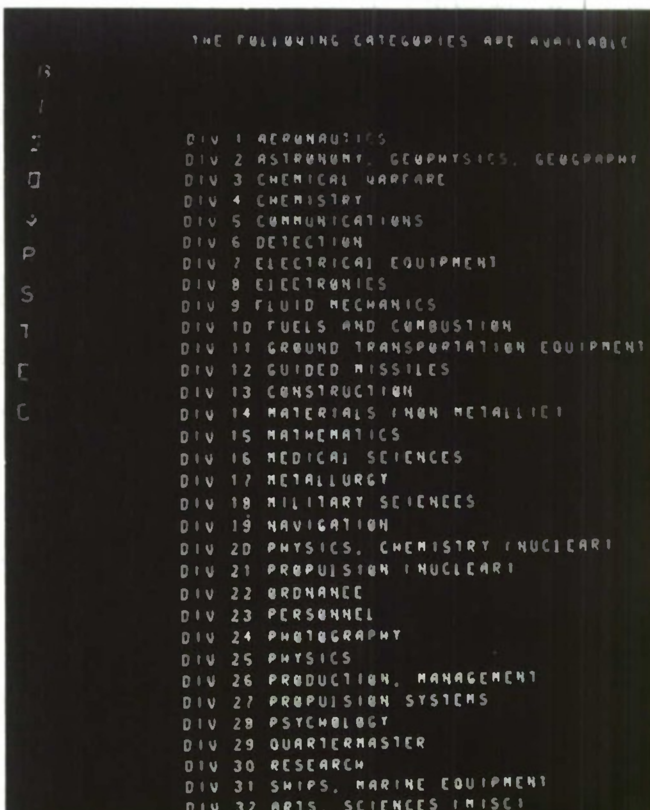


Figure 6—Classification categories

When the BEGIN command is accepted (signified by //), a new display appears (Figure 6) that indicates the 32 divisions or main subject categories into which the data are divided. If the user wishes a further breakdown, he may use his light pen to flash a division. By doing so, he is requesting more information about that category and receives a display of the subdivisions and the number of entries in the category. If he chooses to browse through the items in this category, he does so by either flashing the □ character with his light pen or typing BROWSE/ on the teletypewriter. He then receives a display consisting of the first abstract in the selected category (see Figure 7). If this abstract is not complete, because of the limited number of characters that could be displayed on the scope, he may obtain the remainder by light-penning the "C" or "continue" character. In a similar manner, he can view all the abstracts in the selected category.

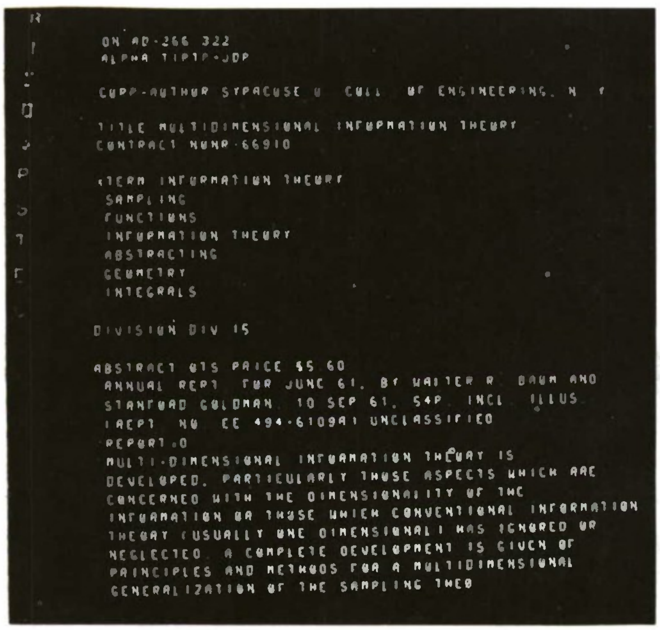


Figure 7—Viewing the retrieved abstract

Although browsing through an organized collection of documents is one way of searching for information, a more commonly used method is to request documents by subject headings or index terms. Many information centers use a form of coordinate indexing, and retrieve information by combining a number of index terms to form a specific request. Usually a trained information specialist must help the user formulate his request for information into a search request made up of approved index terms. In an interactive system, the user requests help by interrogating the dictionary.

By way of illustration, let us suppose the user is doing research in the field of space travel. He is preparing a report on this subject and he wishes to search the collection for relevant articles. He sits at the inquiry station, and first interrogates the dictionary to determine which words can be used as index terms for retrieval purposes.

The following dialog takes place:

SPACESHIPS?
THESE MAY BE RELATED TO SPACESHIPS
SPACESHIP CABINS
SPACESHIPS
SPACESHIPS - POWER SUPPLIES
SPACESHIPS - STABILITY
*END
SPACE?
THESE MAY BE RELATED TO SPACE
SPACE CAPSULES
SPACE CHARGES
SPACE ENVIRONMENTAL CONDITIONS
SPACE FLIGHT
SPACE FLIGHT - CONTROL
SPACE FLIGHT - SURVIVAL
SPACE MEDICINE
*CONTINUE? YES
SPACE MEDICINE - EFFECTIVENESS
SPACE NAVIGATION
SPACE PERCEPTION
SPACE PROBES
SPACE RECOVERY SYSTEMS, INC., EL SEGUNDO, CALIF.
SPACE SCIENCES LAB., GENERAL ELECTRIC CO., PHILADELPHIA, PA.
SPACE SHIPS
*CONTINUE?NO
LUNAR FLIGHTS?
*NOT FOUND
MOON FLIGHTS
*NOT FOUND
MARS FLIGHTS?
*NOT FOUND
MOON?
THESE MAY BE RELATED TO MOON
MOON
MOON - ATMOSPHERE
*END
LUNAR?
THESE MAY BE RELATED TO LUNAR
LUNAR PROBES
*END
MARS?
THESE MAY BE RELATED TO MARS
MARS
MARSH CHARLES A.
MARSHALL JOHN M.
*END

He begins by asking whether SPACESHIPS is an index term by typing the word followed by a question mark. The system responds that, in addition to SPACESHIPS, there are a number of other similar terms that are also usable index words. The system finds these related terms by dividing the query word in half and locating all index terms that start with the same combination of letters.

The user, now recognizing that the term SPACESHIPS might be too specific, asks for information about the more general term SPACE. Again the system responds with a set of related terms. Note that the word SPACE by itself is not an index term, for it is always used in combination with another word. In response to a dictionary inquiry, the system types seven index terms and then asks the user whether he wishes it to continue. After two such inquiries, the user feels he has enough information on this subject and tries some other terms. Some of the words he tries are not index terms, but in his interaction he finds enough that are.

As a result of this dialog, and with the information he has obtained, he is now in a position to formulate a search request. He selects six terms and formulates these into a search request by indicating that he would like to have displayed the list of document numbers that contains any one of these six terms; that is, he combines these terms by means of an OR rather than an AND logic, although both the AND and NOT logic are also available.

He makes his request as follows:

SPACESHIPS OR LUNAR PROBES OR MOON OR MARS
25 ENTRIES ARE REF'D BY SPACESHIPS

6 ENTRIES ARE REF'D BY LUNAR PROBES

13 ENTRIES ARE REF'D BY MOON

3 ENTRIES ARE REF'D BY MARS

*END
SPACE FLIGHT OR SPACE PROBES
15 ENTRIES ARE REF'D BY SPACE FLIGHT

8 ENTRIES ARE REF'D BY SPACE PROBES

*END
SEARCH/
51 ENTRIES

Note that when the user types a request, as distinct from interrogating the dictionary, he does not use a question mark. The system tells him how many entries in this data base (1745 abstracts) are referenced by each term.

He now orders the system to

SEARCH/

and the system responds that there are

51 ENTRIES

Since there is a total of 70 documents that have been indexed by these six terms, it is clear that some documents were indexed by more than one.

The system locates these 51 documents and displays the list by identification number and index term.

The display appears on the scope (Figure 8). Note that not all the documents can be displayed at one time. Of the 51 entries only 37 have been searched. The user may now remove references to the documents that are of less interest by light-penning "R" and the document number. By light-penning the "C," or "continue" character, he allows new document references to be displayed. He may also reorder the arrangement of the display by light-penning the "E" character and two document numbers that he wishes to exchange.

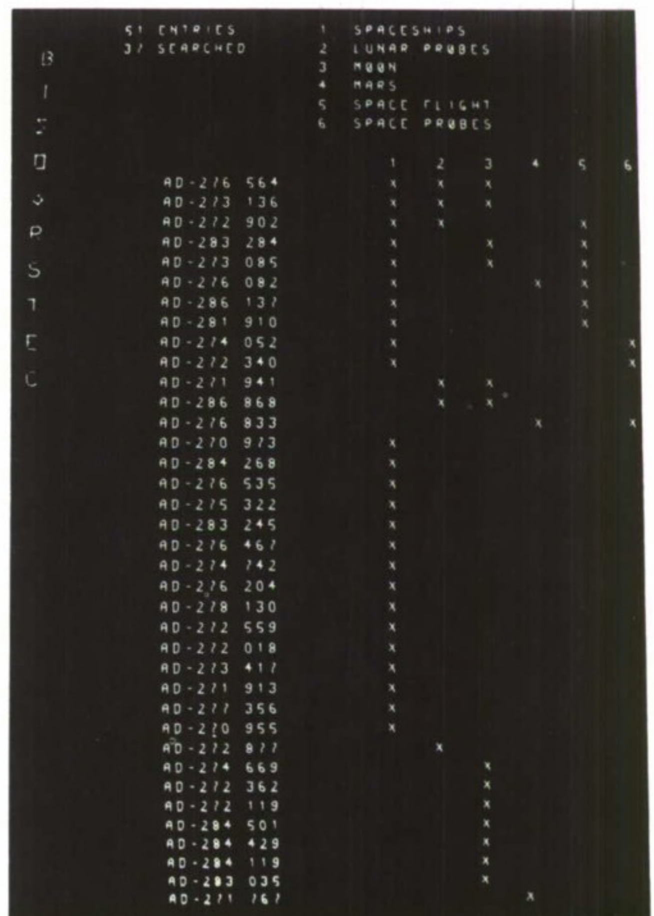


Figure 8—Search matrix of retrieved documents

Before requesting copies of the 51 documents that have been indexed by one or more of the six retrieval terms, the user would like to have more information about their contents. He may obtain this information by simply typing BROWSE/ or by light-penning the appropriate BROWSE symbol.

The system responds to the BROWSE command with

INPUT ATTRIBUTES WANTED

In this instance, let us assume that the user wishes to see just the author and title of the retrieved articles, so he lists these as the attributes wanted. He could also have requested the index terms, the contract number, the date of publication, or the complete abstract.

The first set of authors and titles is displayed on the scope (Figure 9) and the rest can be obtained by the "continue" action.

Figure 9—Document titles and authors displayed on scope BOLD

Should an immediate permanent record be wanted, it can be obtained by the command.

TYPE DISPLAY/

In this manner, the user can browse through the entire set of 51 entries that have been retrieved in response to his request, or that subset of documents that he has not removed from the display. He may save any information that appears for future reference. He interacts with the system, and when he leaves the inquiry station he leaves with the feeling that he has obtained most of the relevant material that the system has in store. The response has been rapid, and the experience has been a satisfying one.

CONCLUSION

Three of the SDC interactive programming systems have been described. These are: (1) the General Purpose Display System (GPDS), (2) the Pattern Learning Parser (PLP II), and (3) the Bibliographic On-Line Display System (BOLD). The versatility of these systems is illustrated by the range of problems that they are capable of handling. By using on-line interactive displays, the man and the machine are able to engage in a dialog as both work together to solve problems. The computer processes data rapidly and displays the results. The human decision maker interprets the displays and determines the accuracy and relevance of the results. The information provided in the displays enables him to steer and control the step-by-step progress of the program. As a result of his involvement, problems are solved more efficiently and in a more satisfying manner.

REFERENCES

1 H. BORKO and H. P. BURNAUGH Interactive displays for document retrieval. *Information Display,* in press.

2 H. P. BURNAUGH *Data base generator for the BOLD system.* Santa Monica, California: System Development Corporation document TM-2306/001/02, 1966

3 H. P. BURNAUGH *The BOLD User's Manual* Santa Monica, California: System Development Corporation document TM-2306/004/01, 1966

4 J. F. BURGER, R. E. LONG and R. F. SIMMONS *An interactive system for computing dependencies, phrase structures and kernels.* Santa Monica, California: System Development Corporation document SP-2454, 1966

5 N. CHOMSKY *Aspects of the theory of syntax.* Cambridge, Massachusetts: The M.I.T. Press, 1965

6 C. GUILLEBEAUX, and E. E. GRANT *The general purpose display system (GPDS) user's manual—Parts I, II, and III.* Santa Monica, California: System Development Corporation document TM-2722, 1966

7 K. L. McCONLOGUE and R. F. SIMMONS *Analyzing English syntax with a pattern learning parser.* Communications of the ACM. Volume 8, Number 11, 1965

8 A. H. VORHAUS, A. H. *General purpose display system.* SDC Magazine, August 1965

LABORATORY SIMULATION OF TACTICAL SYSTEMS AND THE QUEST FOR CRITERIA

IRVING K. COHEN, *Chairman*

Session Participants:

DAVID A. SCHUM
JOSEPH M. DOUGHTY
LOUIS W. MILLER
RICHARD J. KAPLAN
WARD EDWARDS
RICHARD L. VAN HORN

Concerning the evaluation and aggregation of probabilistic evidence by man-machine systems*

by DAVID A. SCHUM
Rice University
Houston, Texas

INTRODUCTION

Two years ago at this same gathering George Briggs and I described some of the details and early results of the man-machine system simulation program at The Ohio State University Human Performance Center (formerly called the Laboratory of Aviation Psychology). This simulation program was then and still is concerned with human performance in information-processing systems which perform diagnostic functions. The initial impetus for examining man's role in diagnostic systems came from papers by Ward Edwards (1962, 1963). It became apparent at the time the first of these papers appeared that the man-machine system simulation facilities being developed at OSU would be rather ideally suited for research on some of the issues raised in Edwards' paper. Since 1962 a fairly large number of experiments have been performed using these system simulation facilities. Detailed descriptions of the simulation facilities are available elsewhere (Briggs & Schum, 1965; Schum, Goldstein & Southard, 1965a). Some of the research has been described in reports issued through our sponsoring agency (Southard, Schum, & Briggs, 1964a, b; Schum, 1966b; Schum, Goldstein, & Southard, 1965a, b). Other research has been reported in various professional journals (Schum, 1966a, b; Schum, Goldstein, & Southard, 1966; Schum, Goldstein, Howell, & Southard, 1966). Both the development of the system simulation facility and the subsequent research program have been sponsored by the Behavioral Sciences Laboratory, Aerospace Medical Research Laboratories, United States Air Force Systems Command.

There have been two major objectives in our research program. The first objective has been to provide information about human capabilities and limitations in the tasks of evaluating and aggregating probabilistic evidence in complex circumstances. The second objective has been to uncover and investigate some of the problems which might be encountered if computer-implemented Bayesian evidence-aggregation procedures were used in actual situations. Several changes have been made in our simulation methodology in the two years since our last report. Much of the detail in simulating a real-time stimulus environment was found to be unnecessary for current purposes. In addition, several lower-level information-processing functions performed by men in our earlier experiments were not incorporated in the current series of experiments. In the present paper I will describe some of the features of our present research on man's role in diagnostic systems and will discuss the manner in which system performance criterion problems have affected our efforts.

Man's role in diagnostic systems

The output of a diagnostic system is in the form of an opinion about the likelihood of each of the various hypothesized causes or states of the world which are assumed to be underlying the emergence of data from some environment of interest to the system. Revisions of these opinions are required as new data emerge. Conceptually, at least two basic information-processing steps are required. The first step involves evaluating the diagnostic impact of items of evidence as they occur. This is roughly equivalent to determining the extent to which each datum allows discriminations to be made among the hypotheses. In some instances, historical records in the form of relative frequencies linking data and hypotheses are available as aids in this process. More often, perhaps, the assessment of evidence impact is possible only by means of the judgments of some expert whose intuition has been educated through experience with similar or related phenomena. Evaluation of unique one-of-a-kind military intelligence data is an

*The research reported in this paper was sponsored by the Aerospace Medical Research Laboratories, Aerospace Medical Division, Air Force Systems Command, Wright-Patterson Air Force Base, Ohio, under Contract No. AF 33(615)-2248 with the Ohio State University Research Foundation. Further reproduction is authorized to satisfy the needs of the United States Government.

example of this sort of impact assessment. The second basic information-processing task involves aggregation of the impact of each new evidence item with the combined impact of all items previously evaluated. In other words, current opinions about the likelihood of hypotheses result from an aggregation of new data with old data. The state of the system's current opinions can be indicated by posterior probabilities, posterior odds, or any other similar expression of confidence in the truth of each of the hypotheses.

By now, nearly everyone with an interest in man-machine information-processing systems is probably aware of Edwards' proposals for the allocation of the above tasks among the man and machine components of diagnostic systems. If men can make evidence impact assessments in the form of $P(D|H)$, likelihood ratio, or some other analogous quantity, then computers programmed according to Bayes' theorem could perform the requisite aggregations. Bayes' theorem prescribes how the aggregation should be performed. Several recent experiments have demonstrated that, in diagnostic or inferential tasks, men extract a significantly decreasing proportion of total amount of diagnosticity in data as the amount of data to be evaluated are increased (Peterson, Schneider, & Miller, 1965; Schum, 1966a; Schum et al., 1966). Bayesian evidence-aggregation procedures ought to curtail the loss in diagnosticity which results from limitations in human ability to aggregate large amounts of probabilistic data. These procedures should, therefore, permit utilization of large amounts of data and should allow for the incorporation of more expert opinion in the actual diagnostic process. Presumably, several or many experts, each knowledgeable about certain classes of data, would provide the evidence impact assessments. In most current diagnostic situations an individual assesses the impact of evidence and performs the required aggregation mentally. The evidence he receives may have been altered through summarizing and editing operations by members of his staff. If there exist more data than the individual can incorporate in his judgments, he must ignore or discard some of them. This is a wasteful procedure (some or all of the discarded data may have been obtained at great expense), and it may be a disastrous procedure if the individual ignores or discards data which happen to conflict with prior expectations he might have of the truth of certain hypotheses. There are thus at least two methods of producing output in systems which perform diagnostic functions. Posterior probabilities or odds can be estimated by men directly from the data or they can be calculated by Bayesian aggregation of men's estimates of $P(D|H)$ or some related quantity. There may, of course, be other methods of arriving at expressions of confidence in the truth of hypotheses. Moreover, there may be diagnostic situations in which basic information-processing functions, other than those described above, are required.

Concerning performance criteria in diagnostic system simulation

Diagnoses form the basis for selection of a course of action. Rational selection of a course of action demands thorough consideration of the cost and payoff effects that the alternative actions may produce. It is possible, as Edwards (1966) has recently explained, to evaluate the diagnostic process independently from the action-selection process. There are, apparently, some unique problems in evaluating inferential or diagnostic behavior regardless of the context in which such behavior occurs and regardless of the method by which the diagnoses are produced. The most obvious criterion for evaluating inference accuracy involves the question of whether or not the hypothesis identified as most probable by some inference method (just prior to the selection of some action) was in fact the "true" hypothesis, i.e., the hypothesis which actually generated the observed evidence. This criterion, although conceptually simple, presents several difficulties. First, there may be no way to verify which hypothesis actually generated the evidence. Such verification may depend upon additional data which the involved system may not be able to collect. One can always argue that subsequent evidence, even if obtainable, can only strengthen or diminish one's confidence in previously selected hypotheses but cannot absolutely verify them. External sources from which one might seek verification of hypotheses may be difficult to find.

Another problem associated with the above-mentioned criterion arises because of the equivocal nature of evidence. Suppose that the true hypothesis (hereafter symolized as H_t) is subsequently known or verified in some manner to everyone's satisfaction. A diagnostician may have assigned entirely appropriate impact to the evidence he has observed and he may have aggregated this evidence in a near-optimal manner. Yet he could be led by the evidence toward acceptance of the wrong hypothesis. In a word, an improbable series of events may have happened and, although behaving in a highly rational manner, the diagnostician was led by the evidence toward the wrong hypothesis. (I am aware of the fact that the state of this diagnostician's prior expectations about the truth of each of the hypotheses also influences which hypothesis he might ultimately have accepted.) The point is that a performance criterion based solely upon H_t may, on occasion, be insensitive to ideal behavior. Because all evidence is, to some degree, consistent with the truth of more than one hypothesis, it is necessary to emphasize that even Bayesian evidence-aggregation procedures guarantee only logical consistency in combining evidence and not perfect accuracy in the selection of H_t.

There is, finally, uncertainty associated with the specification of an exhaustive set of mutually exclusive hypotheses. One can allow for this uncertainty but not reduce it by defining a single diffuse hypothesis which represents the union of those unknown or obscure causes or states of the world one might not have considered. This single diffuse conglomerate of unknown causes is euphemistically termed the "catch-all hypothesis." The problem relates to the level of description one desires in formulating hypotheses. One can always dichotomize states of the world into some gross state (such as "war") and its complement ("not war" or, perhaps, "peace"). Selection of an effective course of action, however, requires a more subtle description of the state of the world. If war is imminent, we should at least want to specify who our adversaries are. In a more subtle formulation of hypotheses one may, through ignorance, leave out valid potential explanations for the evidence he will collect. Some protection is afforded by a catch-all hypothesis. In most diagnostic situations one should be prepared to reformulate one's hypothesis set as evidence appears. This presents further difficulties for performance criteria based upon H_t. The true explanation for some pattern of evidence may never have been explicitly included in one's set of explanatory hypotheses.

Having all but damned criterion measures based upon H_t as probably unobtainable and frequently insensitive in actual inference situations, I shall proceed to explain why and how we have used such measures in our simulation research over the past four years. We sought to evaluate human commerce with the conditional probabilities implied by the Bayesian paradigm in an environment which would be complex but *controllable*, one in which a wide range of variables of interest to diagnostic systems could be manipulated. It was the issue of controllability which led us to the type of data-generation model used (with some variation) in all of our studies. The data-generation model is the set of underlying contingency rules which governed the appearance of and relationships between the events upon which subjects based their probability estimates. All of the models used in our various experiments were matrices of experimenter-defined *objective* probabilities of the form $P(D|H)$. These $P(D|H)$ values define the probability of occurrence of each possible event under each of a specified exhaustive set of mutually exclusive hypotheses. The hypotheses and data described causes and effects in a fictitious hostile environment.

Data existed in major classes, each class having an exhaustive set of mutually exclusive subclasses. Distributions of $P(D|H)$ were assigned by the experimenters across the subclasses of each data class under each hypothesis. Particular values of $P(D|H)$ assigned in this manner depended upon the objectives of the experiment in question. Samples of evidence of "scenarios" were generated by first choosing some hypothesis and then generating a single subclass from each data class according to the multinominal $P(D|H)$ distributions appropriate to each class and to the hypothesis in question (H_t). The essential fact is that we, as experimenters, were in the position of knowing what H_t was for each sample of evidence thus generated.

True $P(D|H)$ values were defined according to the (sometimes bizarre) imagination of the experimenters. Subjects, therefore, could not be expected to make reasonable estimates of $P(D|H)$ except through repeated exposure to events in the environment. The basic *frequentistic* character of our quasi-realistic data base is thus apparent. Subjects could learn evidence impact in the form of $P(D|H)$ only by accumulating relative frequencies linking data and hypotheses. The manner in which subjects could determine evidence impact forms the basic methodological difference between our systems experiments and those of Ward Edwards and his associate (1966) and those of Kaplan and Newman (1966). Edwards and his associates have developed a facility for presenting unique, one-of-a-kind or non-frequentistic intelligence data to subjects. Evidence impact was specified by Edwards' subjects on the basis of plausible linkages of data and hypotheses. Subjects could form these linkages because they had extensive training in the description and past history of a quasi-realistic environment. Kaplan and Newman used a data-generation model based upon a circular normal probability distribution. There is a good reason for questioning the suitability of a frequentistic environment for studying human ability to make assessments of evidence impact in the form of $P(D|H)$ estimates. One can argue that, if relative frequencies linking data and hypotheses were available in some situation, further estimation of $P(D|H)$ by men would be unnecessary. I hope, however, to be able to demonstrate, in the next section, that meaningful experiments can be performed regarding data impact assessments when a frequentistic data base is used.

In our simulation experiments we have been able to specify the exact hypothesis under which each evidence sample was generated. In addition, we could easily specify a prior probability distribution by means of which we could regulate the number of times a given hypothesis would be true during the course of an experiment. In other words, specified prior probability distributions regulate the number of scenarios generated under each hypothesis. All of the several dependent variables used in our experiments to indicate diagnostic accuracy have been based upon our knowledge of H_t for each scenario. In addition, all of these dependent variables have involved posterior probabilities or operations performed upon posterior probabilities. In what follows, $P(H_t|D)$ will indicate the value of the posterior probability (estimated or calculated) which was placed under H_t. In addition, I will be referring to *terminal* values of $P(H_t|D)$ or $P(H_t|D)$ estimated or calculated on the basis of *all* data in some scenario (D thus indicates the evidence in an entire scenario). There are several methods of producing $P(H_t|D)$ in our experiments. The objectives of the particular experiment being performed determine which methods can be compared. Following is a list of the basic methods for determining $P(H_t|D)$ in our experiments. The $P(H_t|D)$ values resulting from all of these methods rest upon assumption of the prior probability distributions specified by the experimenter.

1. $P(H_t|D)$ can result from estimates made directly from data by subjects when (a) the evidence impact has been specified for the subjects by the experimenter, or (b) the evidence impact is learned by subjects through repetitive instances in which data are known to have occurred in the presence of the various hypotheses (this learning requires some subsequent knowledge of H_t by subjects). If evidence impact is specified by the experimenter, the only task for subjects is evidence aggregation.

2. $P(H_t|D)$ can be calculated from Bayes' theorem using subjects' estimates of $P(D|H)$ or some related quantity (the precise forms that these "related quantities" can take are specified in the next section of this paper). Evidence impact in the form of $P(D|H)$ is learned through accumulation of relative frequencies. More complicated impact-estimation tasks, however, require some aggregation of probabilities. A discussion of these more complicated tasks is also included in the next section. This method of posterior probability determination is our analog of Edwards' PIP (probabilistic information processing) system (Edwards, 1962).

3. $P(H_t|D)$ can be calculated from the data in each scenario using Bayes' theorem and the "true" experimenter-specified $P(D|H)$ values. This calculation of $P(H_t/D)$ results from the best possible aggregate of items of evidence whose impacts are perfectly known. It is thus *the* theoretically ideal $P(H_t|D)$. This quantity is highly useful in scaling levels of such variables as degree of conditional nonindependence among evidence items, scenario diagnosticity, and degree of evidence conflict. This calculation also serves as a model for ideal inference behavior when subjects themselves have access to true $P(D|H)$ as in (1a) above. This calculation can be performed under the assumption that the data in a scenario are mutually independent under every hypothesis. It can also be performed upon conditionally nonindependent data. Such calculation requires that the experimenter "tell" Bayes' theorem where the nonindependence exists. This is done by incorporating joint conditional probabilities of the form $P(D_x,D_y|H)$ in the Bayesian solution.

4. $P(H_t|D)$ can be calculated from the current relative frequency *estimates* of $P(D|H|D)$ [or $P(D_x,D_y|H)$] at the time the scenario was presented. A requirement is that the computer be programmed so that the relative frequency estimates of these quantities can be updated in the computer as each new event occurs during some experiment. This calculation represents a model for optimal inference behavior when the impact of evidence must be learned by the subjects, as in (1b, or (2) above. It incorporates the same information available to the subjects at every point in time during an experiment.

The Bayesian calculations described in (3) and (4) above are, of course, performance criteria in themselves. In one sense, H_t merely provides a reference point for comparisons of subjects' responses in methods (1a,b) and (2) with calculations in (3) and (4). One problem encountered in these comparisons arises when some scenario contains a large proportion of data most highly consistent with the truth of some hypothesis *other than* the one under which the scenario was generated. When this happens, calculated $P(H_t|D)$ may not be the largest value in the terminal posterior probability distribution. Such a problem is generally encountered in our work when scenario size is small.

A characteristic of our method of generating scenarios is that, as the amount of evidence in a scenario is increased, the more likely it is that Bayesian terminal $P(H_t|D)$ will be the largest value in the distribution of terminal posterior probabilities. In other words, scenario size and scenario diagnosticity are confounded; evidence typically mounts in favor of the hypothesis which is actually generating the scenario. Suppose that for some scenario a subject's estimate of $P(H_t|D)$ is .75 and some Bayesian calculation of $P(H_t|D)$ is .15. Suppose further that there are five hypotheses. This implies that there is a value of calculated $P(H|D)$ larger than .15 under some hypothesis other than H_t. Suppose that thlis other value is .70. In this case there exists a disparity between the criterion of experimenter-known "truth" and a Bayesian criterion representing the outcome of formally optimal evidence aggregation. In other words, the scenario evidence, in the aggregate, is more highly consistent with the truth of some hypothesis other than H_t. When such a situation occurs (and it sometimes does during the course of an experiment), the experimenter is placed in a logical cleft stick in evaluating the subjects' inference behavior. According to the "truth" criterion by itself, the subjects' performance is commendable. By the Bayesian criterion, the subject has "backed into" the truth by means of inappropriate interpretation or aggregation of the evidence.

In our experiments we have used an assortment of dependent variables. One measure of inference performance used in our early experiments was simply mean $P(H_t|D)$. Ward Edwards then convinced us of the superiority of log odds and log likelihood ratio. In most of our systems experiments, either four or six mutually exclusive and exhaustive hypotheses have generated evidence. In this multihypothesis situation, log likelihood ratio (LLR) has been calculated by means of the following equation:

$$\log \Omega_{t_1} - \log \Omega_{t_0} = LLR \qquad (1)$$

where:

Ω_{t_1} = the posterior odds favoring H_t, i.e., $\frac{P(H_t|D)}{1-P(H_t|D)}$

[$1-P(H_t|D)$ is the sum of the terminal posterior probabilities assigned under all hypotheses other than H_t.]

Ω_{t_0} = the prior odds of H_t, i.e., $\frac{P(H_t)}{1-P(H_t)}$.

[$1-P(H_t)$ is the sum of the prior probabilities assigned under all hypotheses other than H_t.]

If LLR is calculated from subjects' estimates of $P(H_t|D)$, the resulting quantities are called subjects' LLR (SLLR). When calculated from the $P(H_t|D)$ resulting from Bayesian aggregation of subjects' $P(D|H)$ estimates, the LLR values are called PIPLLR. When calculated from Bayes' theorem as in methods (3) or (4) above, the resulting values are called Bayes' LLR (BLLR). Two performance indices relating SLLR (or PIPLLR) to BLLR have been used in several recent experiments. One index is called "accuracy ratio" (Peterson, et al., 1965) where:

$$\text{Accuracy Ratio (AR)} = \frac{SLLR}{BLLR} \qquad (2)$$

SLLR and BLLR can be either positive or negative quantities which means that AR can be positive or negative. A positive value of BLLR or SLLR occurs when $\Omega_{t_1} > \Omega_{t_0}$, i.e., when some increment of certainty in the truth of H_t has been added to that existing before the occurrence of the evidence. Negative BLLR or SLLR occurs when $\Omega_{t_1} < \Omega_{t_0}$ i.e., when the evi-evidence occasions an opinion change in the direction of some hypothesis other than H_t with a concomitant reduction of certainty in the truth of H_t from that certainty existing before the evidence was presented. As the method of calculation of Ω_{t_1} and Ω_{t_0} shown immediately below Equation 1 indicates, we are actually considering only two hypotheses, H_t and its complement $\bar{H}_t$. This dichotomization of the hypothesis set, the fact that a few scenarios contain a high proportion of evidence not favoring H_t, and the fact that subjects inappropriately interpret and aggregate evidence diagnosticity in our complex situation all combine in such a fashion that AR, when calculated, is often difficult to interpret. In addition, when ARs are combined across scenarios and subjects, they are often misleading.

Table I illustrates the various interpretations AR can have in different regions of its domain. AR is actually indicative of the *direction* and the *amount* of subjects' opinion revisions relative to those of Bayes' theorem. Positive ARs result when subject and Bayes' theorem move opinions in the same direction (either toward H_t or toward its complement $\bar{H}_t$). Negative AR occurs when a subject moves his opinion in one direction and Bayes' theorem moves its "opinions" in the other. In Table I, Δ_s refers to the amount of opinion change by a subject toward H_t or $\bar{H}_t$, and Δ_B refers to the amount Bayes' theorem opinions change toward H_t or $\bar{H}_t$.

Table I

Interpretations of accuracy ratio (AR) throughout its domain

	Interpretation		
AR Range or Value	*Agreement as to Direction (Toward $\bar{H}_t$ or H_t)*	*Relationship between Δ_S and Δ_B*	
AR > 1.0	Yes	$\Delta_S > \Delta_B$	
AR = 1.0	Yes	$\Delta_S = \Delta_B$	(Optimality)
0<AR<1.0	Yes	$\Delta_S < \Delta_B$	(The conservatism effect)
AR = 0	—	$\Delta_B > 0$ $\Delta_S = 0$	Subject made no revision of prior probabilities as a result of the evidence.
−1<AR<0	No	$\Delta_S < \Delta_B$	e.g., subject moved opinions more slowly toward H_t than Bayes' theorem moved its opinions toward $\bar{H}_t$.
AR = −1.0	No	$\Delta_S = \Delta_B$	Same extent of opinion revision, different direction.
AR<−1.0	No	$\Delta_S > \Delta_B$	e.g., subject moved opinions faster toward $\bar{H}_t$ than Bayes' theorem moved its opinions toward H_t.

Another dependent variable has been termed the "difference measure" (Schum, Goldstein, Howell, & Southard, 1966), where:

$$\text{Difference Measure (D)} = SLLR - BLLR \quad (3)$$

In all of our experiments, subjects estimating $P(H|D)$ are required to record, before evaluating data from any scenario, the prior probability distribution they are actually assuming. In rare cases do these overt expressions of prior probability differ from the prior distributions specified for them by the experimenter and used by him in calculations of $P(H|D)$. If these reported priors can be believed, Equation 3 can be re-written as:

$$D = \log_{10} \Omega_{t_1} \text{ (subject)} - \log_{10} \Omega_{t_0} \text{ (Bayes)} \quad (4)$$

or as:

$$D = \log_{10} \left[\frac{\psi(H_t|D)}{1 - \psi(H_t|D)} \right] - \log_{10} \left[\frac{P(H_t|D)}{1 - P(H_t|D)} \right] \quad (5)$$

where:

$\psi(H_t|D)$ = a subject's reported terminal posterior probability under H_t.

$P(H_t|D)$ = the calculated value of the terminal posterior probability of H_t.

Under the assumption that a subject's prior distribution is the same as the prior distribution used in Bayesian calculations, D is simply a measure of the degree of similarity between subjects' and Bayes' theorem posterior odds. A D value of zero occurs when Ω_{t_1} (subject) = Ω_{t_1} (Bayes), or equivalently, when $\psi(H_t|D) = P(H_t|D)$. Negative D values result when Ω_{t_1} (subject) < Ω_{t_1} (Bayes). This implies that $\psi(H_t|D) < P(H_t|D)$. Positive D values occur when Ω_{t_1} (subject) > Ω_{t_1} (Bayes), i.e., when $\psi(H_t|D) > P(H_t|D)$.

In Table II are two examples of AR and D values along with the prior and posterior probability distributions from which they were calculated. The subject's estimates of posterior probabilities and the Bayesian distributions are actual examples taken from one of our inference experiments. The most typical or most frequently occurring values of AR in an experiment are those in the range $0 < AR < 1.0$. In nearly every one of our experiments, however, several aberrant values of AR, such as those in Table II, have occurred. In Example A large positive AR (73.49) reflects the fact that both subjects and Bayes' theorem opinions moved away from H_t (Hypothesis D) on the basis of scenario evidence. The Bayesian movement, however, was small relative to the subject's movement. The negative value of D in Example A indicates that the subject's posterior odds under H_t (5/95) were small compared with Bayesian posterior odds (49/51). Antilog D is the ratio of Ω_{t_1} (subject) to Ω_{t_1} (Bayes).

In this example

$$\frac{\Omega_{t_1} \text{ (subject)}}{\Omega_{t_1} \text{ (Bayes)}} = \frac{1}{18.3}.$$

In example B Bayes' theorem is led away from H_t (Hypothesis B) by a small amount, yet the subject has moved his opinions in the direction of H_t by a substantial amount. The disparity in direction of movement produces the negative AR value. The large size of the AR value (—54.91) is due to the large difference in the amount by which the subject and Bayes' theorem moved their opinions. The positive D value (.5121) results from Ω_{t_1} (subject) being greater than Ω_{t_1} (Bayes). Antilog $D = \frac{\Omega_{t_1} \text{ (subject)}}{\Omega_{t_1} \text{ (Bayes)}} = 3.25$.

A fourth dependent variable, called a "dichotomous measure," ignores the absolute size of $\psi(H_t|D)$ or $P(H_t|D)$. Using this dichotomous measure, one asks only whether or not $\psi(H_t|D)$ [or $P(H_t|D)$] was the largest value in some posterior probability distribution. There are some instances in which it is desirable to assess subjects' ability to choose the true state of the world without taking into account the size of their judgments of confidence. When subjects attempt to maximize the size of reported $\psi(H_t|D)$, measures of their performance based upon aggregations of $\psi(H_t|D)$ can be misleading (Schum, Goldstein, Southard, 1966). The dichotomous measure provides additional information which is often useful in evaluating subjects' inference behavior.

In evaluating inference accuracy we have thus made free use of our ability to specify H_t. Since the laws of probability apply in our research as they do elsewhere,

Table II
Accuracy ratio (AR) and difference measure (D) examples

EXAMPLE A

				H_t		
Hypotheses:	A	B	C	D	E	F
Priors:	.100	.200	.100	.500	.050	.050
Subject's Terminal $\psi(H|D)$:	.520	.400	.010	.050	.010	.010
Bayesian Terminal (P(H|D):	.103	.385	.014	.490	.004	.005

$\psi(H_t|D) = .050$; $1 - \psi(H_t|D) = .950$
$P(H_t|D) = .490$; $1 - P(H_t|D) = .510$
SLLR = —1.2787; BLLR = —.0174
AR = 73.49; D = —1.2613

EXAMPLE B

		H_t				
Hypotheses:	A	B	C	D	E	F
Priors:	.167	.167	.167	.167	.166	.166
Subject's Terminal $\psi(H|D)$:	.050	.400	.100	.200	.200	.050
Bayesian Terminal P(H|D):	.024	.164	.477	.118	.168	.050

$\psi(H_t|D) = .400$; $1 - \psi(H_t|D) = .600$
$P(H_t|D) = .164$; $1 = P(H_t|D) = .836$
SLLR = .521; BLLR = —.0095
AR = = 54.91; D = .5121

we suffer when results of the best aggregate of the evidence do not coincide with our knowledge of H_t. However, this happens only infrequently. In some of our current experiments we have found it necessary to discard scenarios for which Bayesian $P(H_t|D)$ is not the largest value in the calculated terminal posterior probability distribution. In entirely abstract or in quasi-realistic experimental situations such as ours, it is difficult to see how there could be accuracy criteria other than those involving either experimenter-known H_t or the hypothesis emerging as most probable from Bayes' theorem calculations.

Speed, in addition to accuracy, is a criterion. In a diagnostic systems context one experimental question relates to the speed at which competing alternative diagnostic systems can arrive at conclusions from the same evidence. Two competing diagnostic systems (i.e., two different procedures for producing output) may arrive ultimately at the same conclusion. One basis for a choice between these alternative systems concerns the speed with which the conclusion has been reached. This question is basic to the major performance comparison in our systems experiments. This comparison, of course, is between the PIP method for producing $P(H|D)$ (method 2 above) and the method involving direct estimation of $P(H|D)$ (method 1 above). The reasons why PIP should be superior have been discussed earlier in this paper. Another research question involving speed concerns the effects of time stress on the ability to make estimates of $P(D|H)$ and $P(H|D)$. An experiment on this issue will be described briefly in the next section.

Some current methodological considerations

A simulated diagnostic task environment

Following is a discussion of the key methodological features of our current series of (three) experiments. The basic issue in these experiments concerns a comparison between posterior probabilities produced by the PIP method and those estimated by subjects directly from data. Subjects in these experiments performed in a simulated military threat-diagnosis context. Only a few details of the data base need to be mentioned. In a fictitious "world" there were four countries: Aggressor I, Aggressor II, a neutral country, and a country called US. There were four states of the world (hypotheses) of interest: Aggressor I is planning to attack US; Aggressor II is planning to attack Aggressor I, Aggressor II is planning to attack the neutral country; and peace will prevail. The hypotheses in this set were defined by the experimenters to be mutually exclusive and exhaustive. Although there are other logically possible states of the world, the subjects were instructed to disregard them as possible causes for the evidence they would observe. Each of the four hypotheses generated data from as many as 18 data classes. Each data class defined a class of activity in the fictitious world such as "proportion of strategic bomber force on alert status in the homeland of Aggressor I." Further, each data class had five subclasses each representing a possible state or condition of the data class in question. The five subclasses in the above example referred directly to the proportions of strategic aircraft on alert 10, 30, 50, 70, and 90 per cent).

Objective probabilities of the form $P(D_{jk}|H_i)$ were initially specified by the experimenter under each of the four hypotheses, where $\underline{j}$ refers to the j^{th} data class, $\underline{k}$ to the k^{th} subclass of data class $\underline{j}$, and $\underline{i}$ to one of the four hypotheses. In addition, joint conditional probability values of the form $P(D_{gx},D_{hy}|H_i)$ were defined since, as discussed below, conditional nonindependence among data items was a feature of the present data base (in the above expression $\underline{g}$ and $\underline{h}$ are two data classes and $\underline{x}$ and $\underline{y}$ are subclasses within $\underline{g}$ and $\underline{h}$). These objective probabilities formed the model from which samples of data (scenarios) were generated under each of the four hypotheses. Each scenario contained single subclasses selected at random, one each according to the $P(D_{jk}|H_i)$ or $P(D_{gx},D_{hy}|H_i)$ distributions appropriate to the hypothesis in question. Subclasses were generated from as many as 18 data classes. Scenario size, an experimental variable in two of the experiments, was manipulated by regulating the number of subclasses generated in each scenario.

An IBM 7094-1401 system, with interface provided by four digital display consoles on line to the 7094, was utilized in five ways: (a) to generate scenarios from the objective probability matrices specified by the experimenters (the number of scenarios representative of each hypothesis conformed to prior probability distributions also specified by the experimenters), (b) to arrange these scenarios in predetermined sequences according to experimental objectives, (c) to provide a means for presenting items of evidence from these scenarios to subjects at specified time intervals, (d) to provide the means for subjects to accumulate and retrieve the past-history information necessary in their specification of evidence impact, and (e) to provide the means for automated collection and analysis of subjects' $P(D|H)$ and $P(H|D)$ responses.* The computer function allowing for past-history accumulation and retrieval is of some importance and bears further explanation. Subjects could call up matrix presentations

*Computer programs for scenario generation and presentation, past-history accumulation, and response collection were developed by Jack F. Southard and Mrs. Louise Wombolt of the Human Performance Center. Mrs. Carol Estep developed programs necessary for statistical analysis of subjects' responses.

on their digital displays. These matrices contained the current relative frequency of each subclass of any single data class or the current joint relative frequency of subclasses from any *pair* of data classes under each of the four hypotheses. The 7094 was programmed to update these relative frequencies as the experiment progressed. From these matrices subjects could obtain estimates of conditional probabilities having the following forms: $P(D_{jk}|H_i)$, $P(D_{gx},D_{hy}|H_i)$, $P(D_{hy}|H_i,D_{gx})$. These probabilities were used by subjects in their estimates of evidence impact. After an experimenter-controlled data-processing time limit, a subject entered his response directly into the digital display by means of a light pen.

Eight highly experienced subjects served in the three experiments. "Highly experienced" means that each subject had served in a number of previous experiments involving the estimation of conditional probabilities in complex circumstances. In addition, each subject was given intensive training relative to his particular task in the current experiments. Four subjects served as $P(D|H)$ estimators in a semi-PIP condition. The semi-PIP condition is discussed below in detail. These subjects specified the impact of various-sized subsets of data from each scenario. The 7094 performed the aggregation of these impact statements according to Bayes' theorem. Each subject had his own console and worked independently of the other three subjects. In addition, all subjects were presented with identical data. For each scenario, therefore, there were four separate Bayesian calculations of terminal $P(H|D)$, each calculated from the impact specification of a single subject. The other four subjects each estimated the quantity $P(H|D)$ directly from the scenarios. That is, they assessed evidence impact and performed *all* of the necessary aggregation of this evidence. They also responded directly into the digital display consoles and they observed the same scenarios as subjects in the semi-PIP condition.

A semi-PIP condition

In a PIP system several or many experts, each knowledgeable about a certain class of evidence, would provide probabilistic expressions describing the diagnostic impact of evidence items as they occur. The data-processing capability of such a system will depend upon both the number of experts available to evaluate the various classes of evidence and upon the amount of evidence each expert must process. If the rate of accumulation of certain classes of evidence is high, $P(D|H)$ experts may not have time to process and respond to each individual datum. Indeed, they may have to process clusters or subsets of data and provide a single $P(D|H)$-type response (under each hypothesis) which will specify the diagnostic impact of the entire subset. This assumes, of course, that no data are to be discarded. The point is that, on occasion, some evidence aggregation may have to be performed by $P(D|H)$ experts. The PIP system notion cannot guarantee that the *entire* task of evidence aggregation will be done by computers. It does guarantee, however, that the *total* amount of necessary aggregation could be broken down into smaller and more manageable aggregations. The success of the PIP idea depends to a great extent upon how well men can estimate $P(D|H)$ or some related quantity. The difficulty of this estimation task depends at least upon the availability of historical records, upon processing time, upon how equivocal and conflicting the data are, upon the size of the evidence set being evaluated, and upon whether or not there are nonindependencies among data. It is of interest, therefore, to evaluate $P(D|H)$ [and $P(H|D)$] estimation by men under conditions which simulate some of these problems.

In the current series of experiments subjects in the PIP group specified the diagnostic impact of clusters or subsets of data. The size of the clusters or subsets of data was a variable in one of the experiments. A single response was provided under each hypothesis which specified the aggregate impact of a data subset. The task was made quite complex because conditional nonindependencies existed between data within a subset and between data from one subset and data from some preceding subset. In order to prescribe the *aggregate* impact of a subset of data, a PIP subject frequently had to aggregate three types of conditional probabilities: $P(D_{jk}|H_i)$, $P(D_{gx},D_{hy}|H_i)$, and $P(D_{hy}|H_iD_{gx})$. In prescribing the precise form which PIP subjects' responses would take, we exploited the likelihood principle. According to this principle, of considerable importance in applications of Bayes' theorem, $P(D|H)$ need only be defined up to a multiplicative constant. In prescribing the impact of a subset of evidence, a PIP subject assigned a simple ratio across the set of hypotheses. The number 1 was assigned to the hypothesis under which the occurrence of a subset of data seemed least likely. Numbers assigned under the other three hypotheses indicated the strength of a subject's opinion regarding the relative likelihood of the evidence subset under each of these hypotheses. Numbers were allowed in the range 1 to 999999. An assumption which this method makes is that, if the three possible *independent* likelihood ratio estimates were made by pairing each of the three hypotheses with the one judged least likely, the same ratio as that described above would be recovered.

There were always three and as many as six subsets of evidence presented in each scenario. The computer system aggregated the impact specifications made by

PIP group subjects for each data subset within a scenario. The resulting $P(H|D)$ distributions were never displayed to them. Subjects who estimated $P(H|D)$ were required to make the same complex impact assessments but they were also required to perform the other aggregation necessary for opinion revision. Their response in two of the three experiments was in the form of a normalized $P(H|D)$ distribution. In the third experiment the subjects responded by assigning a ratio, similar to that described above, which indicated posterior opinion about the truth of hypotheses.

Conditional nonindependence of evidence

Part of the task of evidence evaluation consists of appraising the influence various evidence items have upon one another. The diagnostic impact of some current event may be considerably different if it is known that some other event has occurred. Two events are said to be *nonindependent* if the occurrence of one event somehow influences the probability of occurrence of some other event. In a diagnostic context, such nonindependence may be implied or mediated by the truth of some hypothesis being considered. That is, the underlying process causing the nonindependence is represented by the truth of some hypothesis being considered. Nonindependence mediated by the truth of some hypothesis is called *conditional nonindependence*. Another class of event nonindependence includes instances in which the probabilistic relationship between events is implied or mediated by some process not represented by the truth of any hypothesis being considered. Further discussions of these two classes of nonindependence including examples of each type are to be found in reports by Edwards (1963) and Schum (1966b).

Formally, two items of evidence (D_1, D_2) are *independent* in the light of some hypothesis (H_i) if:

$$P(D_2|H_i, D_1 = P(D_2|H_i) \qquad (6)$$

Equation 6 expresses the condition in which the probability of D_2, when the truth of H_i is assumed, is the same whether or not D_1 has been observed.
Equation 6 implies that:

$$P(D_1, D_2|H_i) = P(D_1|H_i)\ P(D_2|H_i) \qquad (7)$$

If, however, $P(D_2|H_i, D_1) \neq P(D_2|H_i)$ for some H_i, then D_1 and D_2 are said to be *nonindependent* when H_i is true, i.e., there is nonindependence between D_1 and D_2 conditional upon the truth of H_i.

Conditional nonindependence between certain data classes was incorporated in the data-generation models of all of our current experiments. The actual procedure for instituting conditional nonindependence in data-generation models is discussed elsewhere (Schum, 1965b). In the present experiments nonindependence existed only among *pairs* of data classes, i.e., there were no higher-order nonindependencies. The reason for incorporating this feature is that there is often additional diagnosticity in conditionally nonindependent evidence. Recognition and exploitation of conditional nonindependence can allow one to move more quickly toward the acceptance of some hypothesis. In other cases, independent consideration of two or more data may lead toward acceptance of one hypothesis while consideration of suspected conditional nonindependence of these data may lead toward acceptance of some other hypothesis. In any case, it is of interest to investigate how well men can recognize and exploit conditional nonindependencies in the task of specifying evidence impact.

Figure 1 illustrates the formal requirements of an impact-specification task when (a) samples of evidence of various sizes arrive, (b) some of the evidence items are nonindependent, and (c) when a single impact-specifying response is required for each sample as it arrives. In the example, four samples of evidence have come into existence at successive times, T_1, T_2, T_3, and T_4. The samples at T_1 and T_2 contain two items each, the sample at T_3 contains three items, and the sample at T_4 contains four items. A single response, such as a probability of the form $P(D|H)$ assigned under each hypothesis or a simple ratio like that described above, is required for each sample. At T_3, for example, this response must characterize the aggregate impact of D_5, D_6, D_7. Conditional nonindependence among data items is indicated in Figure 1 by connecting lines such as those between D_1 and D_3 or between D_5 and D_6. Nonindependence of items *within* a sample is illustrated by the pairs D_5,D_6 and D_8,D_9. This means, for example, that the joint impact of D_5 and D_6, when one of the hypotheses is true, is different from the total impact arising from independent consideration of these data, i.e., that $P(D_5,D_6|H_i) \neq P(D_5|H_i)\ P(D_6|_i)$ for some H_i. Nonindependence between an item from one sample and an item from some preceding sample is illustrated by the pairs D_3,D_1 and D_{11},D_4. This means, for example, that the impact of D_3 has been altered by knowledge that D_1 has occurred under the assumption of the truth of some H_i.

In Figure 1 there is an equation for each of the four temporally separated evidence samples. The left-hand side of each equation is an expression for the probability of the *entire* sample given one of the hypotheses. Such a probability (or an analogous quantity) is required under each hypothesis being considered in order to specify the aggregate impact of the sample (a single response means one number under each hypothesis). The precise form of each of the four left-hand expressions in Figure 1 depends upon the number of

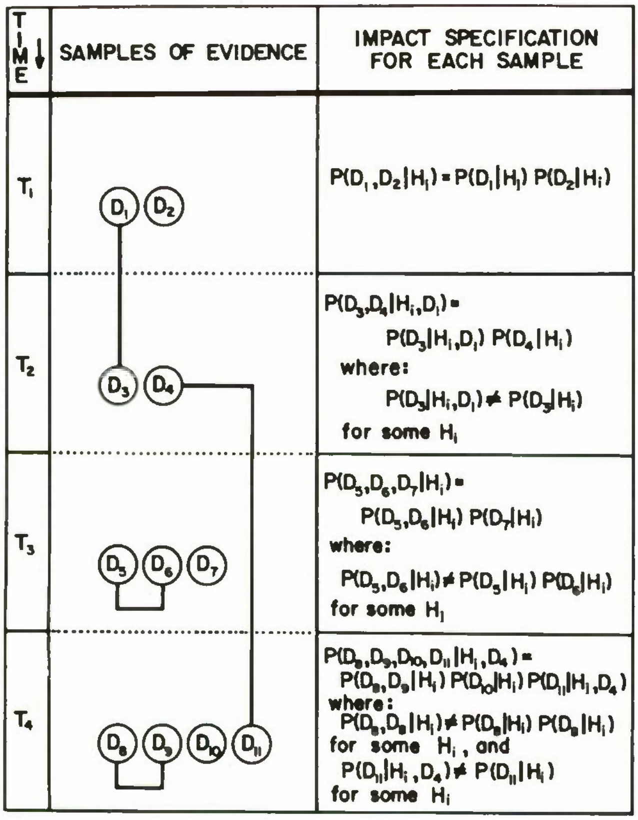


Figure 1—Formal requirements of a sample diagnostic impact-specification task when evidence is conditionally nonindependent

items in the sample and upon whether or not there is conditional nonindependence involving evidence from this sample. As an example, consider the data at T_2. The expression for the aggregate impact of these data is actually a probability of the form $P(D_3,D_4|H_i,D_1,D_2)$. But, since both D_3 and D_4 are independent of D_2 (under every hypothesis), the above expression can be written as $P(D_3,D_4|H_i,D_1)$. D_1 is included in this expression since D_3 and D_1 are nonindependent in the light of one of the hypotheses. Similarly, $P(D_5,D_6,D_7|H_i,D_1,D_2,D_3,D_4) = P(D_5,D_6,D_7|H_i)$ since, in the example, D_5, D_6, and D_7 are all completely independent of D_1, D_2, D_3, and D_4. The right-hand side of each equation contains the "lower-order" conditionaly probabilities involved in the specification of total impact for each sample. The exact form of each of the conditional probabilities in the right-hand side of each equation depends upon the presence or absence of conditional nonindependence and upon what has already been estimated and incorporated into Bayes' theorem. At T_2, for example, there is some H_i under which the impact of D_3 is changed because D_1 was observed. Now the joint effect of D_1,D_3 is specifiable by a probability of the form: $P(D_1,D_3|H_i) = P(D_3|H_i,D_1)\,P(D_1|H_i)$. Observe, however, that $P(D_1|H_i)$ has already been incorporated at T_1 (it is assumed in the present example that the evidence impact estimates are being immediately aggregated by Bayes' theorem). At T_2, therefore, the appropriate probability estimate, when D_3 is nonindependent of D_1 given some H_i, is of the form $P(D_3|H_i,D_1)$. Also observe at T_2 that D_3 and D_4 are independent under every hypothesis. The right-hand side of the equation at T_2 could be written as $P(D_3|H_i,D_1)\,P(D_4|H_i,D_1)$, but $P(D_4|H_i,D_1) = P(D_4|H_i)$. The probabilities shown in Figure 1 are those which will reflect nonindependence when it occurs. For those hypotheses under which data are independent the form could be changed. Suppose at T_2, under some hypothesis (H_k), D_1, D_2, D_3, and D_4 are all independent. The equation at T_2 would then simply be $P(D_3,D_4|H_k) = P(D_3|H_k)\,P(D_4|H_k)$.

The example in Figure 1 can now be related to the task of the subjects who served in the PIP group in our current experiments. Their basic task was to estimate conditional probabilities similar to the left-hand expressions in the equations of Figure 1. Their actual response, however, was in simple ratio form (the number 1 being assigned to the hypothesis under which the data sample was least likely, etc.). They had access to lower-order probabilities similar to those in the right-hand side of the equations in Figure 1. These probabilities came from the simulated historical records (relative frequency matrices) which subjects could call up on their digital display consoles. PIP subjects were, therefore, required to aggregate the lower-order conditional probabilities in order to formulate their aggregate impact assessments. Subjects were given extensive training in all but one of the formal aspects of this task. They were never told how to combine the lower-order conditional probabilities. Their responses were, in fact, complex judgments based upon mental aggregation of the lower-order conditional probabilities.

The posterior probabilities resulting from Bayesian aggregation of the PIP subjects' responses were compared, in all experiments, with posterior probabilities estimated directly from the same evidence by four other subjects. These four non-PIP subjects had the same task of impact assessment as PIP subjects but were required to do the additional aggregation tasks required in posterior probability estimation.

Some experimental variables

Analysis of the data from our current series of three experiments is not yet completed as this paper is being written. I can at least mention the variables which were manipulated in these experiments. In the first experiment we varied the rate of accumulation of scenario evidence. Scenarios always contained six items.

In one condition the six items were presented one at a time. Subjects (in both groups) had three minutes to evaluate each item. In the second condition the six items were presented in successive subsets of three items each. Subjects had three minutes to process each subset. In the third condition all six items were presented simultaneously. Subjects had three minutes to process the entire scenario. In the second experiment we held the rate of accumulation constant (at three items per three-minute processing-time period) and varied the size of scenarios, i.e., the total amount of evidence. Scenarios contained either 6, 12, or 18 items. The third experiment was similar to the second except that we attempted to unscramble the confounding relationship between scenario size and scenario diagnosticity. This confounding problem was discussed in an earlier section of this paper. Briefly, scenarios of increased size become typically more diagnostic of H_t. One is never sure whether the decrease in inference accuracy (relative to Bayes' theorem) as amount of evidence increases is due to the increased *amount* or due to the increased *diagnosticity* of the evidence. PIP and non-PIP subjects' performance was compared under nine experimental treatment combinations resulting from three levels of scenario diagnosticity and three levels of scenario size (6 items, 12 items, or 18 itcfs).

REFERENCES

1 G. E. BRIGGS and D. A. SCHUM *Automated Bayesian hypothesis selection in a simulated threat-diagnosis system.* In J. Spiegel and D. E. Walker (Eds.) Information Systems Sciences: Proceedings of the Second Congress Spartan Books, Washington, D.C. 1965

2 W. EDWARDS *Dynamic decision theory and probabilistic information processing.* Hum. Factors 4, 59-73 1962

3 W. EDWARDS *Probabilistic information processing in command and control systems.* University of Michigan, Institute of Science and Technology Ann Arbor, Michigan ESD Tech. Docu. Rep. No. 62-345 February 1963

4 W. EDWARDS *Non-conservative probabilistic information processing systems.* University of Michigan, Institute of Science and Technology Ann Arbor, Michigan Contract AF 19(628-2823, Final Report 1966

5 R. J. KAPLAN and J. R. NEWMAN *Studies in probabilistic information processing.* IEEE Transcations on Human Factors in Electronics HFE-7, 49-63 1966

6 C. R. PETERSON, R. J. SCHNEIDER and A. J. MILLER *Sample size and the revision of subjective probabilities.* .J Exp. Psychol. 69, 522-527 1965

7 J. F. SOUTHARD, D. A. SCHUM, and G. E. BRIGGS *An application of Bayes' theorem as a hypothesis-selection aid in a complex information-processing system.* Aerospace Medical Research Laboratories Wright-Patterson Air Force Base, Ohio AMRL Tech. Docu. Rep. No. 64-51 August 1964(a)

8 J. F. SOUTHARD, D. A. SCHUM and G. E. BRIGGS *Subject control over a Bayesian hypothesis-selection aid in a complex information-processing system.* Aerospace Medical Research Laboratories Wright-Patterson Air Force Base, Ohio AMRL Tech. Docu. Rep. 64-95 September 1964(b)

9 D. A. SCHUM *Prior uncertainty and amount of diagnostic evidence as variables in a probabilistic inference task.* Organ. Behav. Hum. Perform. Vol. 1, No. 1, August 1966(a)

10 D. A. SCHUM *Inferences on the basis of conditionally nonindependent data.* J. Exp. Psychol. in press (b) 1966 An expanded version is available as AMRL-TDR-65-161 December 1965

11 D. A. SCHUM, I. L. GOLDSTEIN, and J. F. SOUTHARD *The influence of experience and information fidelity upon posterior probabilities estimation in a simulated threat-diagnosis system.* Aerospace Medical Research Laboratories Wright-Pataterson Air Force Base, Ohio AMRL Tech. Rep. No. 65-25 (a) April 1965

12 D. A. SCHUM, I. L. GOLDSTEIN and J. F. SOUTHARD *Further investigation of the effects of reduced input data fidelity upon the determination of posterior probabilities in a simulated threat-diagnosis system.* Aerospace Medical Research Laboratories Wright-Patterson Air Force Base, Ohio AMRL Tech. Rep. No. 65-233 (b) December 1965

13 D. A. SCHUM, I. L. GOLDSTEIN, and J. F. SOUTHARD *Research on a simulated Bayesian information-processing system.* IEEE Transactions on Human Factors in Electronics HFE-7 37-48 1966

14 D. A. SCHUM, I. L. GOLDSTEIN, W. C. HOWELL, and J. F. SOUTHARD *Subjective probability revisions under several cost-payoff arrangements.* Organ. Behav. Hum. Perform. in press 1966

The AESOP testbed: test series I/2*

by JOSEPH M. DOUGHTY
The MITRE Corporation
Bedford, Massachusetts

INTRODUCTION

This paper describes a test and evaluation of on-line digital computer planning aids having possible application to the planning of tactical air combat operations. The development and evaluation of these on-line aids was accomplished by simulating the planning activities characteristic of the Fighter Section of a Current Plans Division of a Tactical Air Control Center. The simulation, currently in operation in the laboratory, is exercised in two ways—manually and with computer aids. Both are carried out in the context of a Joint Task Force operating in a limited war situation.

The manual simulation is concerned with the activities related to the generation of plans for the allocation and scheduling of ordnance and aircraft against ground targets. The experimental subjects ("the planners") in this simulation operated in accordance with established Joint Task Force doctrine and procedures, but without benefit of any type of computer aid. This manual simulation supplied baseline data that was used to establish standards of performance for the force employment planning processes being replicated. This manual simulation will be referred to as the TACC-MANUAL I/2 System.

The second simulation differed from the first only to the extent that the planners used on-line digital computer capabilities in the accomplishment of their planning tasks. These capabilities will be referred to as the TACC-AESOP I/2 System.

Testbed based design

The TACC-AESOP I/2 System was developed in the ESD/MITRE System Design Laboratory.

A development program** of this type requires some means for determining directly and rapidly what possible types of on-line planning aids are useful, which are inept, and which may be actually detracting from the objective of fast and effective planning. One must have a model or models of the digital planning aids to be developed. The AESOP† digital computer prototype is such a model. It can be used to simulate various techniques for on-line information retrieval, file updating, and report generation; as well as the generation and utilization of logical and mathematical models in order to investigate the utility of different forms of computer aids to planning. Comparative data obtained from this simulation are used to guide the development and refinement of on-line computer aids that show promise for planning use. In this regard, the digital model building and execution capability provided by the simulation is one of the prime tools of the experimenters in this investigation. One must also have a model of the type of organizational planning activity to which the digital planning aid is to be applied. In this investigation, this takes the form of a laboratory replica of the Fighter Section, Current Plans Division, of a Tactical Air Control Center. In addition, one must have some evaluative model through which to obtain a better understanding of how the aids might and will be used, where they can help, and how their design should evolve. This evaluative model is discussed in the section on *evaluation procedure*.

AESOP design guidance and verification concepts are based to a considerable extent on a testbed philosophy. The physical testbed in this instance is a laboratory facility that utilizes an IBM 7030 computer, with five on-line input/output stations each consisting of a cathode ray tube display, a light pencil, and on-line typewriter and high speed printer. Using this facility, the AESOP prototype planning aids can be embedded in replications of real time planning operations that are driven by simulated environments characteristic of the operations in question. Thus, experimental applications of on-line digital aids to planning can be exercised in order to systematically evaluate their utility. Such a procedure helps to insure the operational relevance and utility of the proto-

*The work reported on in this document was sponsored by the Electronic Systems Division, Air Force Systems Command, under Contract AF19 (628)-5165.

**For a more complete description of this program, see (Bennett, et al., 1965).

†AESOP - An Evolutionary System for On-line Planning.

types. Successful development and test then lead directly to field application and/or to the establishment of specific new requirements for computer assistance.

As in most system testbeds, the real time operations and their environment are only selectively represented in the AESOP testbed. That is, the replicated planning activity is only a part of the total operation, the rest being simulated in some abstracted form. The outside environment which drives the operation is also simulated in abstracted form. The criteria guiding the structuring of the testbed replica are credibility and control. The credibility criterion is, to a considerable degree, judgmental; the judgment rests on the basis of an analysis of the real world process and a validation of the ensuing model, with its abstractions, by experts, preferably, the real world practitioner. Control refers to test control, the requirement that critical test conditions are known and controlled at all times. The testbed must allow the important variables affecting the process to be represented in precisely identifiable form; over a range of values within which the process may be stressed.

The testbed

This section describes the command structure, environmental interface, and the Fighter Section of the TACC Current Plans Division being simulated in the laboratory. An appropriate scenario provides historical events, geographical setting, and associated environmental events as a vehicle for the external system.

The external system: Command structure and information flow

Figure 1 indicates the command structure which defines the external system. The internal system (Fighter Section) is bounded by heavy lines. Command elements, marked as simulated, interface with the Fighter Section, directly or through higher command elements.

The Joint Task Force (JTF) headquarters insures the necessary coordination of mission requirements. The Army component of the joint operation (COMARFO) provides the TACC Current Plans Division, through the Direct Air Support Center (DASC), with requests for preplanned (next day) close air support missions (CAS). The DASC handles all immediate (on-call) close air support mission requests but levies requests for fighter-bomber sortie allocation on the TACC Current Plans Division.

The Air Force Commander (COMAFFOR) receives preplanned interdiction and counterair mission requests from JTF headquarters. These requests are input to the Current Plans Division from the Deputy for Intelligence (COMAFFOR), who also provides the Current Plans Division with situation briefings and target intelligence. The Deputy for Operations (COMAFFOR) provides the Current Plans Division with weather information.

Requests for fighter-bomber escort for reconnaissance and airlift missions are input to the Fighter Section by the Airlift and Recce Sections within the Current Plans Division.

IA 15,979-1

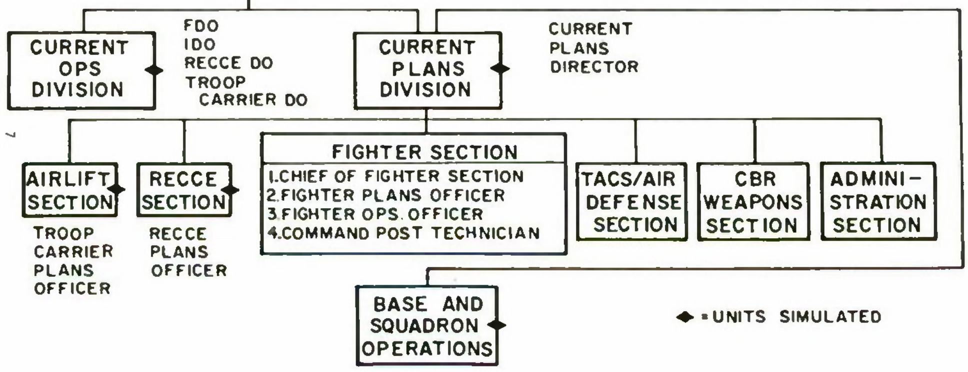


Figure 1.

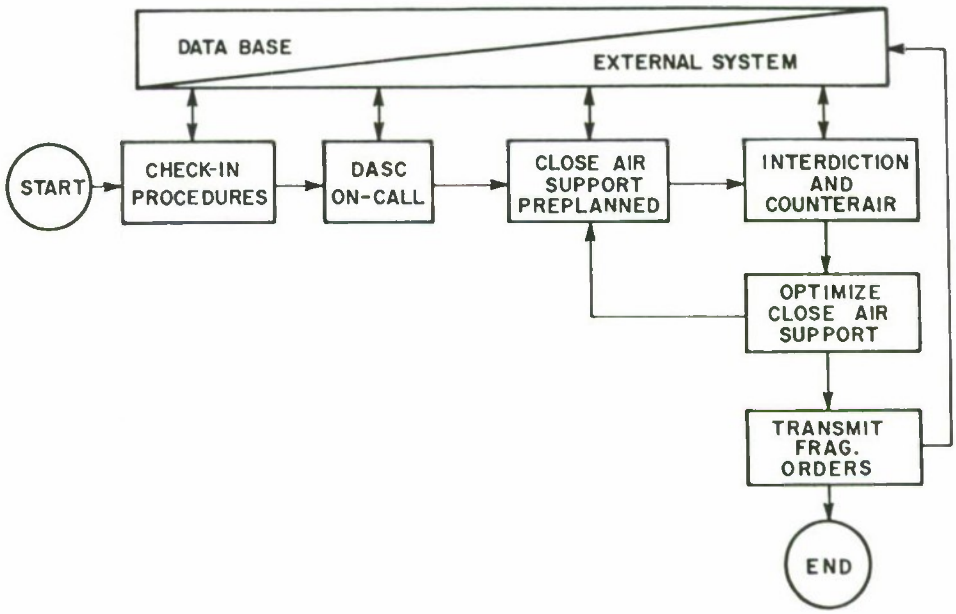


Figure 2

In the Current Operations Division, the Intelligence Duty Officer (IDO) provides the Current Plans Division with information on the current ground situation. Aircraft and crew status is input to the Fighter Section from the Fighter Duty Office (FDO), Current Operations Division. Base and ordnance status information is provided by the several Base Operations.

The basic output of the Fighter Section is a Fragmentary Order reflecting the requirements for meeting DASC on-call and preplanned mission requests. This Fragmentary Order is sent, when appropriate, to the Airlift and Recce Sections in the Current Plans Division and to the Fighter Duty Officer, Recce Duty Officer, and Troop Carrier Duty Officer in the Current Operations Division. All Fragmentary Orders are first approved at the daily planning conference by the Current Plans Director before being sent to the various Squadron Operations and/or to the Fighter Duty Officer in the DASC.

The internal system: Process overview

Figure 2 illustrates the internal system planning process and its articulation with the external system and the data base. At the beginning of each shift, the planners check-in, receive current intelligence briefings, review status information and active Frag Orders, calculate the number and types of sorties available for next day's missions, and obtain estimates of close air support requests and requirements for fighter-bomber escorts on recce and airlift missions.

The Fighter Section of the Current Plans Division is concerned with planning for three mission categories: (1) DASC on-call close air support requirements, (2) preplanned close air support targets, and (3) preplanned interdiction and counterair targets. Planning tasks in each of these three mission categories differ in several ways.

For DASC on-call planning, the task involves only the allocation of specific squadrons to be on ground alert at the required time. Further, W/E* solutions have been completed by the DASC and are not done by the Fighter Section for the DASC portion of the planning process.

On the other hand, allocating for preplanned close air support sorties requires W/E planning as well as scheduling take-off, time-over target (TOT), landing, turn-around, and recycling times.

Interdiction and counterair planning does not ordinarily require W/E solutions but does require scheduling of take-off, TOT, landing, turn-around and recycling times. The W/E solutions for interdiction and counter-air missions are normally supplied to the Fighter Section by the external system.

The internal systems: The test systems

There follows in this section a description of the Manual and Computer-based versions of the internal system that were competitive systems in AESOP Test Series I/2.

TACC-MANUAL I/2

This system is a one-man version of the procedures, manuals, working forms, wall-mounted displays, and

*W/E planning involves determining the number and type of ordnance required to achieve specified results on various types of targets.

work environment of the Fighter Section, Current Plans Division, Tactical Air Control Center, of a Joint Task Force operation. The model is based on Joint Task Force doctrine and procedures.

In the test situation, three of these systems were operated simultaneously but independently.

Figure 3

Figure 3 shows the layout of the room containing the three systems in operation. Each cubicle contains a working table on which can be seen files for manuals and work forms, a wall-mounted map, a large, wall-mounted mission board and a small, wall-mounted status board.

The test observer sits in the foreground.

TACC-AESOP 1/2

This system is an IBM 7030 Computer-based version of the above manual system. The system is an experimental prototype and evolutionary application of generalized AESOP* capabilities to a particular Air Force tactical planning task.

The system is a display-oriented, on-line system which allows the single operator to retrieve, display, print, change, and store planning information; to call up routines for performing arithmetic functions or to execute routine data manipulations; and to generate system outputs—all on-line through the media of a CRT Display, on-line typewriter, light-pen, and high-speed printer.

Figure 4 shows the layout of the room containing the three identical versions of this system that were operated simultaneously but independently during the test. The three planners are shown seated at their work-stations in front of the display with typewriters and high-speed printer to their right. The two test observers are shown standing.

These three TACC-AESOP 1/2 systems time-shared the IBM 7030, with these systems having

Figure 4

priority as foreground jobs over the background batch processing.

System procedures

There follows a brief, simplified description of the procedures followed in each of the systems during the test sessions. These are discussed in the three principal steps of the planning process common to both systems.

Step 1: Initialization. The planning task began with the input into both systems of (1) the set of targets against which they were to plan missions and (2) the resources available for this purpose. Since, during the test sessions, this information was preloaded into each system, the planners simply checked it for completeness and accuracy, and then proceeded to Initialize their systems. Initialization consisted of calculating and storing basic planning information such as (1) the number of aircraft sorties available by squadrons, by aircraft type, and by mission type; (2) the number and types of ordnance required to achieve the level of target destruction requested for each target element, and (3) the distance and time to each target from each of the squadron bases.

MANUAL planners used tables and nomograms to calculate ordnance requirements, and measured distances and time to targets on the wall-mounted maps. This information was then copied onto a work form (the Plan Generation Worksheet) which, when completed, consisted of many pages containing the basic planning information needed to work out an allocation of resources to missions (the next step of planning, to be discussed below).

AESOP planners used computational routines to compute ordnance requirements, distances, and times. For example, the planner simply typed the input message WEAPONS ( ) and the program computed the number and type of ordnance for each target element and the number of sorties of each aircraft type needed to carry this ordnance. This information was automatically stored in appropriate files for later retrieval by display or hard-copy print. Similarly, by typing the input message DISTANCE( ), distance and flying time to each target was computed and stored. As a final step in this initialization the planner executed a routine which copied into a so-called notebook file selected information from the prior computations which allowed him to work out a gross allocation of resources in the next step of planning.

Step 2: Gross planning and assignment. For both systems this step was the time consuming and difficult part of the process. The planner had the task of assigning and scheduling his available resources to missions such that the following criteria (listed in order of importance) were met as closely as possible for all targets:

- The requested level of damage (DP_k).
- The requested time over target (TOT).
- The minimum use of recycled aircraft (i.e., sortie rate, SR).
- The minimum total flying time (FT).

MANUAL planners used the Plan Generation Worksheet to develop a gross plan first. This gross plan, when completed, was a solution to the problem of resource allocation but lacked the details of each mission; e.g., take-off time, time over target, time of return, next ready time of recycled aircraft, flight call-sign. These details were next calculated and entered on another form, the Aircraft-Target Worksheet. With this, the assignment of resources to missions was completed.

AESOP planners were provided with no computerized decision aids to this allocation problem. At the stage of evolution of the system at the time of testing, such aids as planning algorithms or Linear Programming techniques had not been adapted for use in this TACC planning task, although work was proceeding in this direction. Consequently, the planners' task in the allocation of resources was basically the same for both systems, although differing in detail. The AESOP system did, however, provide such aids as the automatic generation of the mission details (take-off time, time over target, P_k, time of return etc.), automated bookkeeping on remaining resources (number and time of availability of aircraft and sorties by squadrons, aircraft type, and mission type), and automated posting of all mission information in a central file called MISSION. The planners' decisions regarding the assignment of resources to missions were entered into the system through a typewriter input message, e.g.,

ASSIGN (T115 27TFS 3)

which meant, assign to target number T115 three aircraft from the 27th Tactical Fighter Squadron. When this instruction had been processed, the assignment was stored and displayed in the MISSION file together with all the automatically computed mission details. This also updated those bookkeeping files in which running accounts of resource utilization and remaining availability were maintained. Changes in decisions regarding assignment could be implemented by first unassigning, e.g., typing UNASSIGN (T115 27TFS), and reassigning.

Step 3: Frag order and mission posting. The output of the two systems was the Frag Order reflecting the plan developed in Step 2. The Frag Order contained all the information necessary for the operating squadrons to carry out the mission planned by the Fighter Section. In addition, the plan was posted for briefing and monitoring purposes.

In the MANUAL system, the Frag Order was written by preparing a page for each mission assigned to each squadron. This involved copying into a form all the details on each mission worked out in the preceding steps and adding call-sign information. The posting of the mission was done by copying from the Flag Order onto a wall-mounted mission board the details of each mission.

In the AESOP system, the Frag Order was printed on-line by simply typing for each mission the input message DO A FRAG and adding the target number, the page number, and the assigned call-sign. Mission posting was automatically completed with the Frag Order generation since all mission information was stored in the MISSION file where it was available for on-line display or printing, locally or remotely, for briefing and monitoring purposes.

Evaluation procedure

The evaluative model used in this test asserts that man-machine systems should be evaluated on four aspects: (1) Formality, (2) Utility, (3) Excellence, (4) Desirability. This classification scheme follows Jacobs (1965). Evaluations of systems on these four criteria should answer the following questions about the system:

- **Formality** — Does the system meet the requirements and expectations of knowledgeable TACC planning personnel insofar as credibility is concerned?
- **Utility** — Does the system do the job it is supposed to?
- **Excellence** — How well does the system accomplish its goal?
- **Desirability** — How efficient is the system in utilizing resources to accomplish its goal?

The criteria outlined in the succeeding section deal primarily with the excellence and desirability of the output and of the process in producing an output. The evaluation procedures, however, should assure that the formality and utility criteria are met for the sake of completeness. System formality was satisfied prior to the test series by establishing realistic and noncontradictory planning ground rules and procedures, reflecting Air Force doctrine as published and interpreted by Air University personnel. Subjects were trained according to these specifications, and test sessions were governed by these "formality" rules.

The utility criteria was met for any plan output by correcting it, after the test session in which it was generated, so that all errors (e.g., violation of resource availabilities) were removed. The correcting was done by *deleting* from the final plan those parts which were in error. In other words, the output that was scored via the method outlined in the next section was a utile output. Obviously, the extent to which the original plan did not satisfy the utility requirements was reflected in lower excellence and desirability evaluations, since the erroneous parts of the plan made no contribution whatsoever to the score.

The excellence of output, or how well the plan satisfied the goal of potential target destruction, was measured as a function of the P_k (probability of target destruction), and TOT (time over target) measures. The desirability of output, or the efficiency of committing resources to the planning goal, was measured as a function of aircraft utilization as reflected in the Sortie Rate and Flying Time measures.

Process excellence, or how well the system operated in producing its output, was stated in terms of operator opinion and attitude. Process desirability, or how efficiently the system utilized resources in producing the output, was measured in terms of total time to plan.

Details on these criteria and measures are given in the following paragraphs.

Criteria and measures of excellence and desirability

In the following discussion *plan output* and *planning process* are distinguished as separate categories for evaluation purposes.

Output

The approach was to combine output measures of excellence and desirability in such a way that the systems could be discriminated as better or worse according to a single criterion score. The following function yielded such a score:

$$W = 10 \cdot S_{pk} + 8 \cdot S_{TOT} + 5 \cdot S_{SR} + 4 \cdot S_{FT}$$

where W is the overall worth score of a plan and S_{Pk}, S_{TOT}, S_{SR}, and S_{FT} are the scores from the output measures of P_k, TOT, Sortie Rate, and Flying Time, respectively, computed according to the specification given below. The task of the planner was to maximize the above criterion function.

Obviously, the worth function can be modified if more realistic weights are found—or its entire form can be changed. At present, though, it is a reasonable first pass at a credible criterion function, based as it is on the opinion of people experienced in this kind of planning.

Following are detailed discussions of the scoring of the above four key output variables which subjects controlled and traded off in generating a TACC plan.

Planned $P_k(\overline{P}_k)$ levels for each target. The penalty for not meeting a desired P_k is a function of both the priority-precedence and the planned P_k. The farther a planned P_k falls below a desired P_k for any target, the lower the P_k score for that target. The relative penalty increased in order of priority-precedence; i.e., the higher the priority-precedence, the higher the penalty for P_k degradation. Except for preplanned CAS targets, exceeding the desired P_k does not increase the criterion score. Increasing CAS P_k's beyond the desired P_k will increase the score only if the desired P_k's on all CAS targets have been met. P_k should only be degraded after lower order criteria have been degraded.

Let N = number of targets in a problem package, and R = the rank of a target in the package, $1 \leq R \leq N$. Ranking will be done on the basis of priority-precedence.

$$\text{Let } Q = \sum_{i=1}^{R} \frac{N+1}{N+i} \tag{1}$$

(a) For each target where desired P_k (DP_k) is less than or equal to $\overline{P}_k$, score

$$\frac{100}{Q} \cdot \left[\frac{N+1}{N+R}\right] \tag{2}$$

(b) For each target where $\overline{P}_k$ is less than DP_k, score

$$\frac{50}{Q} \cdot \left[\frac{N+1}{N+R}\right] \cdot \frac{\overline{P}_k}{DP_k} \tag{3}$$

(c) For CAS targets, in the situation where all CAS targets have $\overline{P}_k$— DP_k, and no CAS targets have been delayed, score

$$\frac{10}{Q} \cdot \left[\frac{N+1}{N+R}\right] \cdot \frac{P_k - DP_k}{DP_k} \tag{4}$$

as a bonus.

These scores are designed to give generally higher value to higher ranking targets, while assuring that the value of meeting the DP_k on *any* target is more than the value of *not* meeting the DP_k on any other target regardless of rank. Furthermore, no improvement in score is achieved by planning a P_k greater than DP_k, except in the case of CAS targets, where any available apportioned sorties remaining after all DP_k's are met should not be left unassigned.

Planned time-on-target ($\overline{TOT}$) for each target. Targets must be hit within the range of DTOT through LTOT. If the STOT (scheduled time over target) falls outside of this range, the target is in effect uncovered, and its P_k criterion score is set to zero. However, within this range, the closer the STOT is to the DTOT, the higher the criterion score.

This score is computed as follows:

(a) For each target where TOT is equal to the desired (DTOT), score $\frac{100}{N}$ where N is the total number of targets. (The fraction, 0/0, shall be interpreted as 1.)

(b) For each target where TOT is later than DTOT, but before LTOT (latest time over target), score

$$\frac{10}{N}\left\{1+9\cdot\left[\frac{LTOT-TOT}{LTOT-DTOT}\right]\right\}$$

$$\left\{\frac{N+1}{N+(N-R+1)}\right\} \qquad (5)$$

where N is the total number of targets and R is the rank of the target.

(c) For each target where $\overline{TOT}$ falls outside the DTOT:LTOT range, score 0.

DTOT:LTOT range, score 0.

These scores give higher value to striking targets closer to the DTOT. In the event of mission segmentation, the planned TOT for the first strike will be taken as the planned TOT for the mission.

Sortie rate of aircraft as used in the plan. Although some aircraft may be recycled and flown against more than one target on mission day, planners should keep the used sortie rate as low as possible. In short, aircraft should be recycled only when resources or apportionments are stressed.

This score is computed by multiplying the number of aircraft used in the plan by 100 and dividing by the number of sorties.

Total flying time. When a choice of aircraft types and/or squadrons exists, assignments should be based on the minimum mission flying time associated with the assignment. Mission flying time = (one-way flying time by aircraft type) multiplied by (number of aircraft type sorties assigned).

To compute this score, multiply the total number of each aircraft type by its sortie rate, by its combat radius in nautical miles and divide by its combat speed in knots. Add these values for all aircraft types. The result is the maximum available flying time (FT_{max}). The used flying time (FT_{used}) will be the sum of each sortie times its one-way flying time divided by 60. The score is:

$$100\times\left[\frac{F_{max}-FT_{used}}{FT_{max}}\right] \qquad (6)$$

All of the above scores should range roughly between 10 and 100 for an average plan with fifteen target elements and stressed resources. Also, the scoring method has been constructed to give higher scores to greater worth as determined by the criteria outlines.

Process

In order to assess the excellence of the process of generating the system output, the operators of the systems in test were asked to critique briefly the systems after each run and more comprehensively at the end of the test series. The objective was to discover opinions and attitudes concerning experienced advantages, disadvantages, difficulties, problems, needed improvements, confidence in, and the like, in the use of the systems to achieve the goal of a plan output.

Process desirability is best expressed in terms of man-hours or man-minutes spent in completing a plan. Since one-man teams were used to operate the test systems, the measure of process desirability became simply the number of minutes from start of planning to completion of the plan. Since planning progressed through similar stages in each system, time measures for each stage were also obtained.

Test procedures

AESOP Test Series I/2 was conducted in the ESD/MITRE System Design Laboratory during the period October 13 through November 2, 1966. In all, 10 four-hour test sessions were held during which 60 plans were generated, 30 on each system.

Subjects and training

Preceding the test series, an intensive four week period of subject training was conducted on the manual system (TACC-MANUAL I/2) and computer-aided system (TACC-AESOP I/2). Training on the latter was combined with final system shakedown. The six subjects were trained an equal amount on both systems.

Employees of the MITRE Corporation were used as test subjects. Original plans had been to use Air

Force Officers as subjects but last minute pressures to complete the test as soon as possible did not allow sufficient time for their selection, assignment, and training.

All subjects were trained to the point of producing acceptable plans in each system within the four hour time period allowed. However, at the time the test sessions were begun no subject had reached his maximum performance level. The test design, therefore, controlled for continuing gains in skill by alternating the subjects on the systems in successive test sessions.

Test sessions

Test sessions were conducted three times a week, with the two competing systems being operated simultaneously on the same problem. Since multiple copies of each system were available, three TACC-MANUAL I/2 and three TACC-AESOP I/2 systems were operated concurrently; the manual systems in one room and the computer-based systems in another. Within each room precautions were taken to isolate the three planners. A maximum of four hours were permitted to complete the plan.

Before each test session all subjects were briefed simultaneously. This was a simulation of an Intelligence Briefing on the ground and air situation in the context of which they were planning. A 10-day scenario of a hypothetical Joint Task Force operation had been written to provide operational-like inputs into the "next day" planning shop. (Fighter Section).

Both systems were pre-loaded* with the information necessary for the planning task: targeting requests and aircraft resources available to the planner. Thus, in the TACC-MANUAL I/2 system, this information had been typed onto the work forms used by the planners and posted on the appropriate wall-mounted status board and map. In the TACC-AESOP I/2 system, this information had been inserted into the planning files to be used by the operator.

All planners began work on the same planning problem at the same time. Planning continued without interruption (except for occasional "downs" due to computer or equipment failure on TACC-AESOP I/2) until a plan had been completed, posted, and frag order cut.

One test observer for the manual system and two for the computer-based system recorded data and insured that all test procedures were correctly followed. All system actions and the time of such actions were automatically recorded for TACC-AESOP I/2. These records were supplemented by records of "waiting time" (the time between the initiation of an action and the system response) made by the planners and observer using stop-watches. In TACC-MANUAL I/2, planners time-stamped each work form as it was used. Such records on the two systems gave a detailed, time ordered record of the entire planning process.

The output of the manual system was available in the completed work forms and a photographic record of the wall-mounted mission board on which the plan was posted. The output of the computer-based system was recorded by printing the files containing the completed plan and the frag order.

Over the course of the test series, planners were required to complete three debriefing forms. One was completed by each planner at the conclusion of each test session. The purpose of this form was to record the users' opinions on the systems' performance during each session and to provide explanatory information for the evaluation of specific aspects of planning problem solution. The second debriefing form was completed at the end of the test series and was designed to elicit user preferences among systems and sub-systems and the reasons for these preferences. The third debriefing form was also completed at the end of the test. It was a planners' critique of the TACC-AESOP I/2 system in which the planner was encouraged to discuss at length the strengths and weaknesses of the computer-based system and to make specific design modification recommendations.

Test design

The six test subjects were divided into two groups, A and B. The two groups were then alternated on the two systems from session to session as follows:

	——SYSTEM——	
Test Session	TACC-MANUAL I/2	TACC-AESOP I/2
	Group A	Group B
2	Group B	Group A
3	Group A	Group B
4	Group B	Group A
5	Group A	Group B
6	Group B	Group A
7	Group A	Group B
8	Group B	Group A
9	Group A	Group B
10	Group B	Group A

The test problem for each session was the same for all planners. Equivalent but different problems were used throughout the test, each problem consisting of 15 target elements and with available aircraft

*An earlier test was conducted to determine the time and accuracy of this loading process.

sortie resources within ± 2 sorties of the number required to meet the requested P_k and DTOT (desired time over target).

Test results

Analysis of the test data is currently in progress. Results will be published as soon as possible.

Preliminary analysis does indicate, however, a superiority in performance of the computer-based system (TACC-AESOP 1/2) over the manual system (TACC-MANUAL 1/2) in all measures of the evaluative criteria of excellence and desirability: Total time to prepare a plan (process desirability); plan quality (a combination of output excellence and desirability), and planner preference (process excellence).

The extent and significance of the differences, however, remain to be determined.

REFERENCES

1 E BENNET et al *Simulation of computer and operator procedures for tactical force employment planning* MTP-14 The MITRE Corporation 1965

2 J F JACOBS *Practical evaluation of command and control systems* MTP-7 The MITRE Corporation 1965

3 J SPIEGEL et al AESOP: a general purpose approach to real-time direct access management information systems MTP-33 The MITRE Corporation 1966

JUDGE: A value-judgment-based TAC command system*

by Louis W. Miller and Richard J. Kaplan
The RAND Corporation
Santa Monica, California
and
Ward Edwards
University of Michigan
Ann Arbor, Michigan

I. INTRODUCTION

The purpose of this Memorandum is to review work completed on a value-judgment-based Tactical Air Command system. The problem of interest is this: a TAC commander must decide whether to grant a particular request for immediate close air support, or to deny it on the grounds that he must conserve resources to fulfill a possible later request of greater importance. We have assumed that resources (missions available for close air support) are seriously limited relative to demands and that there is a wide range of importance in the requests. Whether or not these assumptions characterize current TAC operations, they do describe circumstances in which a TAC command system should be prepared to function. Since our proposed system is based on judged values or utilities, we have called it a Judged Utility Decision Generator (JUDGE).

This Memorandum begins by summarizing the aspects of the TAC control problem that are relevant to our proposed system, and the present solution to that problem. Section III presents a set of formal rules for making mission-dispatching decisions. These rules assume that numerical measures are available for evaluating possible mission outcomes. They first translate values of mission outcomes into values of the missions themselves; a solution to the problem of predicting the effectiveness of missions is crucial to that translation. Next, the rules specify a procedure for making dispatching decisions based on mission values; the procedure responds to fluctuations in arrival rates of requests having various values.

*This research is sponsored by the United States Air Force under Project RAND—Contract No. AF 49(638)-1700—monitored by the Directorate of Operational Requirements and Development Plans, Deputy Chief of Staff, Research and Development, Hq USAF. Views or conclusions contained in this Memorandum should not be interpreted as representing the official opinion or policy of the United States Air Force.

How the numbers representing outcome values are obtained makes no difference to the formal rules for using them, so long as they have the appropriate cardinal-utility properties. But we assume that expert judges should ordinarily supply such numbers. Section IV presents a small experiment comparing numbers obtained by several judgmental procedures; the data indicate that one procedure is clearly preferable to the others, and that the numbers so obtained compare well with "correct" numbers (available in this experiment but not in TAC control system settings). Thereafter we discuss a field study which incorporated this procedure and was the first test of a JUDGE system. Used in the study were experienced TAC officer subjects who made value judgments about mission outcomes in a verbally-described, realistic, limited-war setting.

The designs both of JUDGE and of procedures for evaluating and refining it illustrate some general principles concerning the design and evaluation of judgment-based command systems. Since our views on this topic seem radical to us, and since we know of no published presentation of similar views, we conclude this Memorandum with a rather extensive, abstract discussion of the problem and our opinions about its solutions. Our main conclusions are that all command systems must be based on explicit or implicit value judgments, that it is better if these judgments are explicit and if formal decision-theoretical algorithms are used to translate them into decisions, and that systems based on such judgments and algorithms are self-validating.

II. THE TAC CONTROL PROBLEM

The Tactical Air Command (TAC) has numerous operational missions to perform including air defense, counter air, interdiction, assault, airlift, reconnaissance, and close air support. Close air support (CAS) missions use tactical aircraft in direct and immediate response to requests for support from ground forces. A special agency, the Direct Air Support Center (DASC), controls strike aircraft on CAS missions. The DASC controls aircraft assigned to it by the Fragmentary Operations Order (frag order) written each day by the command organization of the Tactical Air Commander. The DASC is collocated with the Tactical Operations Center (TOC) of the Army, to facilitate TAC-Army coordination.

Requests are made from the field by the Forward Air Controller (FAC) over the Tactical Air Request Net. The FAC is an Air Force fighter pilot stationed with the ground troops. He knows the capability of the aircraft and of the weapons they carry, and can advise the ground commander about feasibilities of missions or can suggest the use of air in cases where the ground commander might not otherwise consider it. The FAC transmits Army requests for CAS to the DASC; these requests are approved or disapproved by each echelon of Army command between the requester and the TOC. Each of these intermediate Army commands has an Air Liaison Officer to help with the approval decisions.

The DASC has the authority to communicate with air bases and send strike aircraft on missions specified by an Army-approved request. Whether the final decision to fulfill a request is made primarily by the Air Force or by the Army depends mostly on the personalities of the individuals involved. Doctrine says that when the request gets to the DASC, it has become a requirement and must be satisfied if it is within that organization's capability. If all requests for CAS could be satisfied immediately, the DASC's main function would be status recording and communications; it would be necessary only to send the aircraft off and follow them to make sure that they go to the target and did what they were supposed to do. But TAC must be prepared to function in situations where many more demands are placed on its resources than can be satisfied. In that case, a TAC control system for CAS must consider every request as it arrives and decide whether it will be fulfilled or whether planes will be withheld for use in response to a possibly more important future request.

This sketch of the TAC control problem highlights two difficult tasks that a TAC control system should perform. The first is evaluating the relative importance of each request; the second is predicting the arrival of future, more important requests. If all requests are equal or nearly equal in importance, then a first-come first-served policy will be as good as any other, and no control problem exists. If requests vary in importance, but nothing whatever can be said about the likelihood of future, more important requests, then although first-come first-served is not a good policy, no better one seems feasible. Note, however, that the value problem has logical priority over the prediction problem. What must be predicted is not merely the arrival of future requests, but the arrival of future requests more valuable than the one now being considered; such a prediction is impossible unless the worth of each request can be measured at least roughly.

In current TAC field operations, the importance of each request is, in principle, evaluated by means of a priority number assigned to the request by the requester. Requests transmitted to the DASC are seldom assigned less than priority "1" by their originators. The informal rules specify that you should not ask for air support unless you really need it; if you do need it, you want to be sure that your request is attended to. This procedure, of course, makes the priority assignment system meaningless.

A closely related problem is the self-adaptation of demand for CAS to the supply of airplanes. Requesters all listen to the network on which requests are made; if no planes are available at a given time, no requests are made. Apparently, informal social pressures discourage unfulfillable requests. This phenomenon would not occur, we believe, in any environment in which TAC's resources were less overwhelmingly abundant relative to the need for them than is the case in typical exercises.

Even in the present abundant environment, this self-adaptive mechanism has three regrettable features. A larger supply of requests would provide more intelligence information to higher headquarters. It would permit higher level Air Force and Army commanders, rather than requesters in the field, to evaluate relative importance of potential requests. And it would reduce the chances that the man whose urgent need develops relatively late in the day, after almost all of the day's missions have been flown or assigned, will have to do without needed CAS. So the essence of our proposal is that requests should be encouraged and that procedures based on value judgments should be used to determine which requests should be fulfilled and which should not.

A plane and pilot may fly several CAS missions in a day. How many such missions they can fly depends on the nature of each mission and on the turnaround time for the plane, which may vary from 30 minutes to many days. At present, the frag order specifies that the DASC will have a certain number of sorties available at the beginning of the day and additional sorties available at specified times thereafter. TAC bases are responsible for turning their planes and pilots around sufficiently fast to provide the missions specified in the frag order. This rigid procedure does not allow unplanned variations in the number and nature of missions assigned early in the day to influence the number of missions available later—though in practice the system operates more flexibly than the formal description implies. A TAC control system should be designed to consider explicitly how the number and nature of early missions, whether anticipated or not, influence later availability of aircraft when DASC is deciding on early mission assignments.

III. THE LOGICAL DESIGN OF JUDGE

This section presents a set of formal rules that accepts as inputs statements about the values of various possible mission outcomes, past experience with incidences of requests having various values, and plane availability. Outputs are dispatching decisions and future plane availability. Empirical questions about the availability of inputs is not a concern here. We assume that values will in general be made available by exploiting expert human judgment, that past experience with requests will be available, and that information of a specified nature about plane availability will be available. Section IV examines procedures for obtaining value judgments.

In the following discussion, we assume that value should be maximized. For us, this assumption has somewhat the status of an axiom; the philosophical discussion at the end of this Memorandum examines it and its implications to some extent. We also assume that under conditions of uncertainty, expected value should be maximized. The notion of expected value maximization as a criterion of optimal risky decision-making has far too long a history to summarize here. For an elementary discussion, see Edwards (1954). For situations in which the so-called gambler's ruin problem does not arise, we know of no alternative rule for risky decision-making that deserves consideration.

JUDGE's inputs include values of targets that might be attacked, probabilities that attacks will destroy their targets, and number of planes avilable. To make dispatching decisions, it considers the value of sending one or more planes and thus of perhaps destroying the target, and the cost resulting from the fact that if the planes are sent against this target, they will not be available to send against other targets later. In order to evaluate this cost of later unavailability, JUDGE must consider future requests and future plane availability. JUDGE's outputs are forecasts concerning future plane availability and dispatching decisions. The computations that transform its inputs into its outputs are complicated. A technical but nonsymbolic summary of their nature follows.

To make computation possible, JUDGE divides time into a sequence of discrete periods called horizons. (It makes no difference to the formalities how long a horizon is; we tend to think of it as one or two hours.) The crucial point about horizons is that new planes, or planes turned around after earlier missions, become available for use only at the beginning of each horizon; the supply of planes within a horizon is taken as fixed. This oversimplification permits a computationally and intellectually convenient division of the problem into two parts: the dispatching problem and the planning problem. The dispatching problem is exclusively concerned with the decisions made within one horizon regarding aircraft assignment; the planning problem is concerned with both the process of planning a day's activities (important, for example, for maintenance planning purposes) and the task of supplying to the dispatching algorithm some numbers that tie together the several horizons that make up a day.

Within one horizon, the only decision to be made is how many planes, if any, to send in response to each request. Each request comes with two kinds of information attached: a value for destroying the target, and a number, or perhaps a function, that represents the effectiveness of planes against the target. The dispatching rule knows how many planes it has available for use in the remainder of the horizon, and it has a basis for estimating the arrival rates of requests with various values.

To use this information, the dispatching rule calculates an expected gain or loss for each possible number of planes that might be sent. The expected gain of sending no planes is zero. The expected gain of sending one or more depends on the target's value, the probability that the planes being sent will kill the target, and the cost of the lost opportunity to use those planes against some later target. Calculating this opportunity loss is complicated, and depends on the forecasted arrival rates for requests of various values. Among the various available dispatching decisions, the one with the highest expected value is selected. Increases in mission value, in supply of planes, and approach of the end of the horizon all increase the willingness of the system to dispatch a plane.

The dispatching rule operates within a horizon: the planning problem is concerned with the interrelations among horizons. The output of the planning algorithm is a forecast of the number of sorties to be available in each horizon. This forecast permits the dispatching rule to find out the value of planes left over at the end of a horizon. They are obviously valuable, since they can be used during the next horizon. But they obviously lose value at the horizon boundary, since immediately before the boundary they are all the resources the system has, while after it the system has been supplied with a specified additional number of planes. Their value in the new horizon is simply the expected value of the additional missions they make possible in the next horizon, taking into account the fact that the end of the next horizon might conceivably also find planes sitting on the ground unused. The expected number of planes to be resupplied at the beginning of each horizon must be calculated, and this depends on the dispatching decisions made in previous horizons. The computations begin with a list of numbers which in essence predicts the number of sorties to be flown during each horizon. This prediction reflects actual past dispatching decisions and forecasts future ones. This prediction is updated periodically by a linear programming procedure to reflect changes based on actual experience with requests and actual previous dispatching decisions. The updating will probably be done once per horizon, depending on how much conditions turn out to differ from those planned for, how much computer time is available for a rather demanding calculation, and so on.

The procedure taken as a whole is suboptimal in a number of ways, and depends on a number of special assumptions. However, it should be a good first approximation. As it becomes desirable to improve on it, it will be possible to change parts of the procedure piecemeal, or to add other features not now included.

The dispatching rule

The dispatching portion of JUDGE is developed under a set of assumptions concerning the resources, admissible decisions, request process, target values, and mission effectiveness measures. These assumptions are described in the following paragraphs.

Resources. The aircraft to be used over a horizon are all of the same type and loading and are completely interchangeable. Situations involving collections of nonhomogeneous aircraft, in which the substitutability of one type for another depends on the target, would require the state variable representing inventory of remaining sorties to be multi-dimensional.

Admissible Decisions. The requests JUDGE handles are of the "as soon as possible" variety, and it is assumed that a dispatching decision is made immediately upon receipt of each request. The only admissible decisions consist of assigning to the mission a number of sorties, between zero and the number of remaining sorties. Collecting requests or delaying action on marginal requests is not permitted.

The Request Process. The arrival of requests over time is assumed governed by a Poisson process with a known rate of λ requests per hour. The substance of this assumption is that over a very small time interval of length h, the probability is $(1 - \lambda h) + o(h)$ that there will be no requests, the probability is $\lambda h + o(h)$ that there will be one request, while the probability of two or more requests is negligible. (The symbol o(h) means a quantity very small relative to h.) These probabilities are independent of any events that may occur outside of the h-interval under consideration. The request rate will be taken to be a constant over any particular horizon, although allowing λ to vary over the horizon could easily be incorporated into the calculation if dependence of λ on time could be estimated.

In the absence of evidence, this Poisson hypothesis provides a plausible assumption regarding the arrival of requests. Certain mathematical results indicate that the Poisson process is an appropriate model if there are many potential requestors acting independently. (See, for example, Cox and Smith, 1954.)

Mission Value Functions. Associated with each request is a mission value function that specifies an immediate reward for each act that could be taken in response to the request. Let v be the judged value of destroying the target, and let $\eta(x)$ be the extent to which a mission of x sorties can be expected to achieve the desired effect. The mission value function, u(x), is taken to be the product:

$$u(x) = v\eta(x) \quad \text{for} \quad (x = 0, 1, 2, \ldots).$$

For a target occupying a small area that could be destroyed by a single attacking aircraft, it is reasonable to propose that

$$\eta(x) = 1 - (1 - p)^x,$$

where p is the probability that one aircraft will be successful. Then the formula above represents the probability that not all x airplanes miss the target. The parameter, p, depends on many factors relating to the target, aircraft, weapons, and tactics, but current technology is sufficiently well developed to provide reasonable values. In an implementation of JUDGE, it is possible that p might also be obtained through the judgment of qualified personnel.

As a result of this form for $\eta(x)$, the mission value functions have the desirable properties that the marginal utility of sending an additional plane decreases for increasing x, and that the reward for not responding to a request is zero.

With targets of significant area, say troops dispersed over a region the size of a football field, the effectiveness function should reflect the degree to which the target is covered by lethal ordnance. By making some coarse assumptions, the same simple form for $\eta(x)$ can be rationalized. Let T denote the target area, let L represent the lethal area of the weapons a single aircraft deposits on the target, and let p_1 be the probability that a plane successfully hits the target. If we subdivide the target into areas of size L and assume that a plane hitting the target hits any particular subdivision with probability L/T, then the probability that any subdivision is killed is $(P_1L)/T$. The probability that any particular subdivision is killed by an attack of x aircraft is then given by

$$\eta(x) = 1 - (1 - p)^x,$$

where

$$p = \frac{P_1L}{T},$$

and this is also the total expected portion of the target destroyed.

Sensitivity to the number of sorties can be varied by introducing a second parameter and writing the mission effectiveness formulas as

$$\eta(x) = 1 - (1 - p)^{x^\alpha}$$

We have explored hypothetical effectiveness functions generated this way with α in the range of 0.8 to 1.4. Although we offer no interpretation for the extra parameter, a function of this form should be capable of fitting a wide variety of effectiveness functions that might be derived from very detailed analyses. For simplicity we shall assume that the mission effectiveness functions are specified by the choice of a single parameter, p.

In order to keep in mind that the mission value function depends on both the estimated target value and the parameter of the effectiveness function, a mission value function will be indicated by u(x:v,p).

In developing the dispatching rule, the target values and effectiveness parameters will be treated as random variables. Since these variables may very well be correlated, we shall indicate a marginal distribution for target values and a conditional distribution for effectiveness parameters. Denoting the two random variables by V and P, respectively, define:

$$F(v) = \mathrm{Prob}(V \leq v),$$

and

$$G(p|v) = \mathrm{Prob}(P \leq p|V = v).$$

These distributions are assumed known; the forecasting problem will be discussed in the following subsection.

Derivation of the Dispatching Rule. Within a horizon, the state of the system may be described by specifying the number, n, of remaining sorties available for dispatching and the amount of time, t, left until the horizon is over. The cost associated with dispatching sorties is achieved by attributing a value, $W_n(t)$, to being in the state (n,t). This value is a measure of the total expected utility associated with the dispatching decisions to be made over the remaining t hours of the horizon when there are n sorties that can be used. Then the price of x sorties in the state (n,t) is $W_n(t) - W_{n-x}(t)$.

If a request having attributes v and p is received when the state is (n,t), the best decision is to choose the value of x, (x = 0, 1, . . . ,n), that yields the maximum difference between reward and cost, the value of x that maximizes

$$u(x:v,p) + W_{n-x}(t) - W_n(t).$$

Since the decision does not affect the last term in the above expression, it may be dropped. The quantity

$$\max_{0 \leq x \leq n} \{u(x:v,p) + W_{n-x}(t)\}$$

represents the sum of the immediate reward and the value of the state resulting from the best decision in the event that a request with characteristics v and p is received when the system is in state (n,t).

Under the notion that the target value and effectiveness functions associated with a request are selected randomly, the expected value of the reward, plus the value of the state resulting from the action taken when a request arrives at state (n,t), is

$$\int_v \int_p \max_{0 \leq x \leq n} \{u(x:v,p) + W_{n-x}(t)\}\, dF(v)\, dG(p|v).$$

Taking h to be a very small number, suppose that there are n sorties available in a horizon that ends in t+h hours. Given that a request arrives in the next h hours, the expression above represents the value of state (n,t + h). On the other hand, if there are no requests in the small time interval, the sum of the expected rewards to be earned over the t + h hours is the same as that for the slightly shorter horizon of length t hours, and then $W_n(t + h) = W_n(t)$. By assumption, the probabilities of a request and no request in the small interval are, respeetively, λh and $(1 - \lambda h)$. Weighting the two expressions for $W_n(t + h)$ by the eorresponding probabilities gives

$$W_n(t + h) = (1 - \lambda h)\, W_n(t) + \lambda h \int_v \int_p \max_{0 \leq x \leq n} \{u(x:v,p) + W_{n-x}(t)\}\, dF(v)\, dG(p|v).$$

Following the standard proeedure of subtracting $W_n(t)$ from both sides of the equation, dividing by h, and then taking limits as h beeomes small, we obtain the differential equation:

$$\frac{dW_n(t)}{dt} + \lambda W_n(t) = \lambda \int_v \int_p \max_{0 \leq x \leq n} \{u(x:v,p) + W_{n-x}(t)\}\, dF(v)\, dG(p|v),$$

which will be referred to as the "value equation." For any horizon, a series of these equations are solved reeursively starting with n = 1 and going up to the number of sorties allocated for use in the horizon. The reeursive solution is neeessary beeause for any value of n, the solution depends on the $W_m(t)$ functions for $m < n$.

The partieular solutions depend on a set of boundary eonditions as follows: the notion that the utility of no sorties available at any time is zero is embodied in the boundary eondition

$$W_0(t) = 0 \quad \text{for} \quad (t \geq 0).$$

In addition, a boundary condition is needed for each value of n eonsidered to speeify the value of having n sorties left over when the horizon ends. For the last horizon, leftover sorties are useless so that in this speeial ease it would be reasonable to impose the conditions:

$$W_n(0) = 0 \quad \text{for} \quad (n = 1, 2, \ldots).$$

For other horizons, boundary values should be related to the marginal value of having additional sorties available in the following horizon. The use of these boundary values to eonneet one horizon with the following one will be diseussed under planning.

The dispatehing rule itself is a byproduct obtained in solving the value equations, and eonsists of a table of the optimizing x's as funetions of the parameters v, p, n, and t. However, in an implementation of JUDGE, it would be more practical to retain just the solutions, $W_n(t)$, as a two-dimensional table with some suitable grid for the time parameter. Then the optimizing decisions for any v and p eombination eould be readily eomputed as required.

A fruitful way of eharacterizing the dispatehing rule is to define $R_x(n,t)$ as the set of points (v,p) such that the optimal ehoiee is to dispateh x sorties for a request with attributes v and p reeeived when the system state is (n,t).

A boundary point between $R_{x-1}(n,t)$ and $R_x(n,t)$ may be determined by fixing p and solving for the value of v that yields the same total of immediate expected reward plus expected value of the resulting state when either x−1 or x sorties are dispatched. That is, for a particular mission effeetiveness funetion $\eta(x)$ (which is determined by the choiee of p) solve for the value of v that satisfies

$$v\eta(x-1) + W_{n-x+1}(t) = v\eta(x) + W_{n-x}(t).$$

The solution, denoted by v_x is

$$v_x = \frac{W_{n-x+1}(t) - W_{n-x}(t)}{\eta(x) - \eta(x-1)}.$$

For $v < v_x$, it is better to send x−1 sorties; for $v > v_x$, sending x sorties is the preferred action. That is, v_x is the minimum value of v for whieh x sorties would be dispatehed for a request whose mission success parameter is p and the state is (n,t).

Some eare must be taken beeause the sequenee of v_x's obtained in this way may not be inereasing. It eould happen that $v_{x+1} < v_x$. Such a reversal would indieate that there is no value of v sueh that dispatching x sorties is the optimal aetion. Then the ehoiee is between x−1 and x+1, and a new value of v_{x+1} must be determined as

$$v_{x+1} = \frac{W_{n-x+1}(t) - W_{n-x-1}(t)}{\eta(x+1) - \eta(x-1)} .$$

Now it must be checked that $v_{x+1} > v_{x-1}$. If not, x−1 sorties would never be dispatched, and v_{x+1} must be recomputed by comparing the results of dispatching x−2 and x+1 sorties, provided that $x \geq 2$. This checking and recomputing process continues until either there are no more reversals or we have found the point on the boundary of $R_0(n,t)$ and $R_{x+1}(n,t)$.

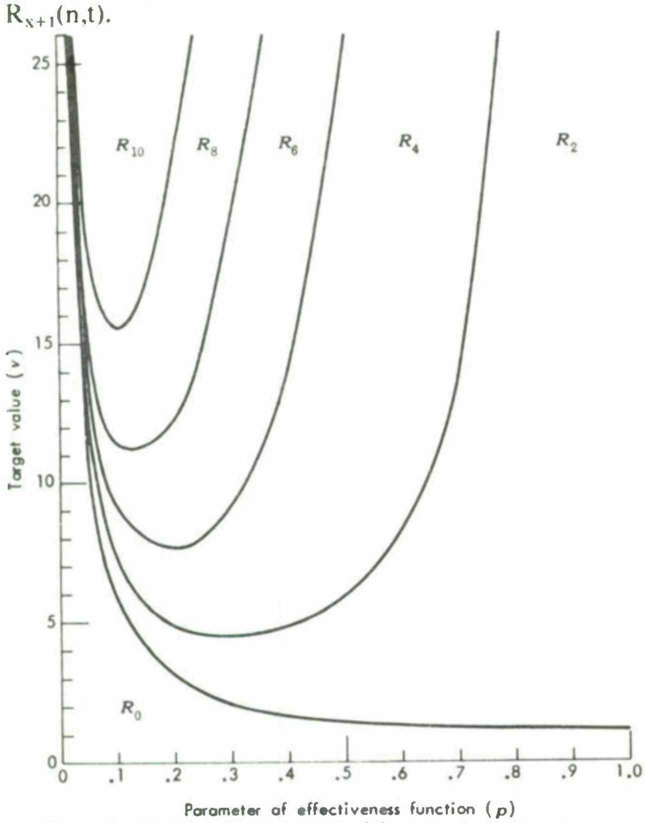


Figure 1—Typical dispatching decisions for fixed state (n,t) for a combination of $(v,p) \in R_x$, dispatch x aircraft

To illustrate how the sets $R_x(n,t)$ might look on the (v,p) plane, Figure 1 was constructed for the mission value function $u(x{:}v,p) = v(1 - (1-p)^x)$. The values of $W_n(t)$ were taken from a numerical example whose solutions to the value equations are shown in Figure 2. To generate the curves of Figure 1, n was chosen to be 20, while t was set at 2.0 at the end of horizon 1. It was assumed that only even numbers of sorties would be sent on any mission, so that for any p, v_x was calculated from

$$v_x = \frac{W_{22-x}(t) - W_{20-x}(t)}{(1-p)^{x-2} - (1-p)^x} \quad \text{for } x = 2, 4, \ldots, 10.$$

These curves reflect the notion that the higher the value of the request, for a fixed parameter, p, the more sorties we are willing to expend in an effort to achieve success. The general U-shape of the regions in

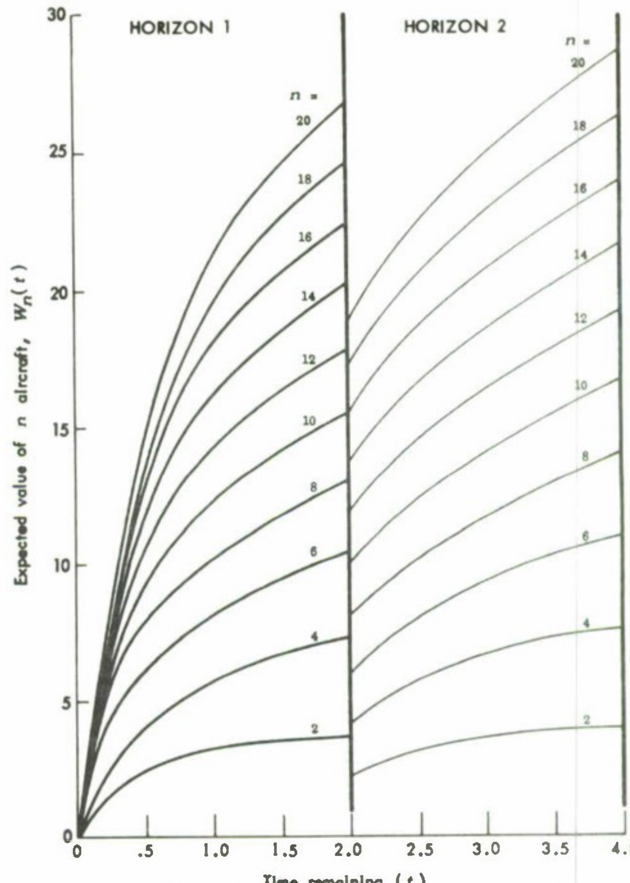


Figure 2—Example solutions to the value equation for two horizons at end of day

Figure 1 are attributable to the behavior of $\eta(x)$ as a function of p. For missions of constant v, the additional expected reward for dispatching an additional aircraft is proportional to $\eta(x+1) - \eta(x)$. As a function of p, this difference increases for small values of p, reaches a maximum, depending on x, and then decreases. Suppose that the optimum action were to send x sorties on a mission with value v and some small success parameter, p. If p were a bit larger, the additional reward for sending another aircraft might be great enough to justify another, according to our expected value criterion. But if p is large, the improvement in reward to be gained by another aircraft is insensitive to changes in p. In the limiting case where p = 1, there would never be any reason for dispatching more than one aircraft.

If n were smaller or t were larger, the region $R_x(n,t)$ would be moved upward, because under these the price of sorties would be higher.

The value equation may be written in terms of the regions, $R_x(n,t)$, as

$$\frac{dW_n(t)}{dt} + \lambda\left[1 - \underset{\substack{(v,p)\,\in \\ R_0(n,j)}}{\int\int} dF(v)\, dG(p|v)\right] W_n(t)$$

$$= \lambda \sum_{x=1}^{n} \iint_{\substack{(v,p) \in \\ R_x(n,t)}} [u(x:v,p) + W_{n-x}(t)]\, dF(v)\, dG(p|v).$$

Approximate Solutions to the Value Equation. Even with the simplest assumptions about the forms of u, F, and G, it would be difficult to obtain exact solutions to the value equation because the coefficient of $W_n(t)$ and the right-hand side are very complicated functions of t. A simple and satisfactory solution to this problem is available by approximating the true solutions with step functions.

To compute approximate solutions, the time horizon is divided into increments of length Δ, and approximate values of $W_n(t)$ are determined at the points $\Delta, 2\Delta, 3\Delta, \ldots,$ under the assumption that $W_n(t)$ is constant over the half-open interval $[j\Delta, (j+1)\Delta)$. Let the approximating value for $W_n(t)$ for $t \in [j\Delta, (j+1)\Delta)$ be denoted by $\overline{W}_n(j\Delta)$.

In carrying out the computations, one would start with j = 1 and calculate $\overline{W}_m(\Delta)$ for each m from 1 up to the maximum value desired. Then j would be stepped and the process repeated, calculating the $\overline{W}_m(2\Delta)$ values, etc.

Consider the computation of $\overline{W}_n[(j+1)\Delta]$. The quantities, $\overline{W}_m(j\Delta)$, $(m = 0, 1, \ldots, n)$ have already been obtained and are stored. Since the $\overline{W}_m(t)$'s are taken to be constant over the Δ interval, the decision rule is constant over the interval also. The sets, $\overline{R}_x(n,j)$, of points, (v,p), such that x sorties would be dispatched may be determined. For notational convenience define:

$$\alpha = \lambda\left[1 - \iint_{\substack{(v,p)\in \\ \overline{R}_0(n,j)}} dF(v)\, dG(p|v)\right],$$

$$\beta = \lambda \sum_{x=1}^{n} \iint_{\substack{(v,p)\in \\ \overline{R}_x(n,j)}} [u(x:v,p) + \overline{W}_{n-x}(j\Delta)]$$

$dF(v)\, dG(p|v)$.

It is understood that α and β are specific for particular n and j. They may be evaluated numerically by placing grids over the domains of v and p. According to the approximation scheme, they too are constant over the Δ interval, and the approximate form for the value equation is

$$\frac{d\overline{W}_n(t)}{dt} + \alpha \overline{W}_n(t) = \beta.$$

The solution to this equation is

$$\overline{W}_n(t) = ce^{-\alpha t} + \frac{\beta}{\alpha},$$

where c is an arbitrary constant. The constant is evaluated from the boundary condition supplied by the known value of $\overline{W}_n(j\Delta)$. Thus,

$$c = \left[\overline{W}_n(j\Delta) - \frac{\beta}{\alpha}\right] e^{\alpha j \Delta},$$

and $\overline{W}_n[(j+1)\Delta]$ is given by the recursive relationship:

$$\overline{W}_n[(j+1)\Delta] = \left[\overline{W}_n(j\Delta) - \frac{\beta}{\alpha}\right] e^{-\alpha\Delta} + \frac{\beta}{\alpha}.$$

The estimates of $W_n(t)$ obtained by this method will be smaller than the true values. Upper bounds can also be obtained by using $\overline{W}_m$'s at $(j + 1)\Delta$ rather than $j\Delta$ in the computation of α and β. In developing the $\overline{R}_x(n,j)$ sets, a preliminary estimate of $\overline{W}_n[(j + 1)\Delta]$ is required (this is what we are trying to compute), but this can be done by extrapolating from $\overline{W}$ values associated with surrounding combinations of j and m. It is not difficult to develop compromise schemes that give estimates between the upper and lower bounds. All such variations, including the upper and lower bounds, are asymptotically correct (for large t) so that there is good control over error. Accuracy is improved by making Δ small, but the amount of computation varies directly with the number of points on the time grid. Computational experience has shown that some refinement in the method of choosing replacements for the $\overline{W}_m(j\Delta)$'s for use in computing α and β can make up for a rather coarse grid.

Figure 2 displays approximate solutions to the value equation for two horizons covering the last four hours of a day. The time scale is in hours remaining, and the boundary between the two horizons is at 2.0 hours. In both horizons, the number of available aircraft was set at 20 (curves for odd numbers of remaining sorties have been omitted). Since time 0 represents the end of the flying day, the boundary conditions were set to reflect zero value for any unused sorties at time 0. The boundary values for the next carlier horizon (horizon 2) were set in accordance with the conclusions reached in the next subsection. Additional details about the specific example used to generate the data in Figure 2 are given at the end of Sec. III.

Models similar to the one presented in this section have been published by Kincaid and Darling (1963) and Kaufman (1963).

The planning technique

The purpose of the planning portion of JUDGE is to set values for the number of sorties available at the beginning of each horizon. The planning computation will probably be done once per horizon. For specificity, the following paragraphs describe procedures appropriate for the first such calculation of the day; minor modifications are required for subsequent updatings.

In our notational scheme, the horizons will be numbered from the last occurring (horizon number 1) to the earliest. Such "backward numbering" is slightly more convenient than numbering horizons in their natural sequence and is analogous to our use of "t" in the previous discussion to represent the time remaining in a horizon. If there are r horizons in the flying day, the result of the planning stage is an r-dimensional vector $\underline{y} = (y_1, y_2, \ldots, y_r)$, where the i^{th} component represents the number of sorties planned to be available when the i^{th} horizon begins. These numbers will not be exactly realized because the results of the dispatching procedure and the aircraft recovery are subject to uncertainty, and because the planning technique is based on certain approximations.

Since we must now consider a number of horizons, the $W_n(t)$ functions will be superscripted with the appropriate horizon indices. Thus $W_n^i(t)$ will be the expected value of the state (n,t) in the i^{th} horizon. Also let the length of the i^{th} horizon be denoted by t_i.

In developing the plan, we assume that there is a fixed number, s, of aircraft that are to be used for close air support during the day. The object is to determine $\underline{y}$, subject to restrictions about the number of sorties that can be flown, that maximizes the quantity:

$$\sum_{i=1}^{r} W_{y_i}^{i}(t_i).$$

This objective function is appropriate since one term in the sum represents the total expected value of the sorties available within the associated horizon. Then the sum over horizons is a measure of the expected value to be gained from the allocation $\underline{y}$.

It is clear that the planning technique should depend on the dispatching procedure, because it is impossible to plan without taking into account the effects of the dispatching method. This dependence is evident in our choice of objective function. On the other hand, the dispatching rule depends on the planning technique in the computation of the $W_n^i(t)$ functions. As discussed previously, the dispatching rule for the i^{th} horizon depends on the boundary conditions $W_n^i(O)$, for $(n = 1, 2, \ldots, y_i)$.

The issue is: what should these boundary conditions be? Taking them to be zero is unsatisfactory because this would place an unrealistically low price on missions at the end of the horizon. Another alternative is to make the value of n sorties at the end of a horizon the same as the value of n sorties at the beginning of the next horizon. That is, to set $W_n^{i+1}(O) = W_n^i(t_i)$. But this would place too high a price on missions at the end of the $(i+1)^{st}$ horizon in view of the imminent resupply.

Since unused sorties can be carried over to the next horizon, the boundary values should reflect their expected utility in the next horizon. Suppose that the planned number of available sorties for the i^{th} horizon is y_i. Then the boundary conditions for the next earlier horizon should be

$$W_n^{i+1}(O) = W_{y_i+n}^{i}(t_i) - W_{y_i}^{i}(t_i)\ .$$

The right-hand side of this equation is a measure of the additional utility of n sorties brought from the $(i+1)^{st}$ horizon to the next.

Suppose that there are n sorties still left h hours before the end of the $(i+1)^{st}$ horizon, where h is a very small number. The cost of dispatching x sorties under these conditions would be

$$W_n^{i+1}(h) - W_{n-x}^{i+1}(h) = \left[W_{y_i+n}^{i}(t_i) - W_{y_i}^{i}(t_i)\right] -$$

$$\left[W_{y_i+n-x}^{i}(t_i) - W_{y_i}^{i}(t_i)\right] = W_{y_i+n}^{i}(t_i) - W_{y_i+n-x}^{i}(t_i)\ .$$

Thus, this way of setting the boundary conditions satisfies our intuitive feeling that the cost of x sorties should be continuous across horizon boundaries.

The interdependence between planning and dispatching suggests that the plan be arrived at by an iterative procedure such as the following. The computation is begun with an initial guess at the allocation vector $\underline{y}$. Based on this starting point, the values $W_m^i(t_i)$, associated with various numbers (say from 1 to s) of available sorties at the start of each horizon are calculated by the dispatching rule computation for use in the objective function of the planning algorithm (still to be described). The planning algorithm gives a new $\underline{y}$ vector that satisfies the sortie availability constraints and maximizes the objective function.

However, the boundary conditions used to calculate the $W_m^i(t_i)$ values depend upon the original $\underline{y}$ vector. If the new allocation is very different from the original, the $W_m^i(t_i)$ values corresponding to the new allocation might differ considerably from the original set of values, so that the new allocation would not really be optimal. The solution is to calculate the new set of end-of-horizon values based on the new allocation and use them in the planning algorithm to obtain a third allocation. This iterative procedure could be carried out until two successive allocations are fairly close.

Although there has not yet been any computational experience, the convergence should be rapid and it is doubtful that the process of recalculation would be (or should be) carried out more than once. This conjecture is supported by numerical results indicating that end-of-horizon values are not very sensitive to the boundary conditions from which they were obtained. (Compare the sets of end-of-horizon values for the two horizons in Figure 2. The only difference between the horizons is in the boundary conditions.)

One difficulty arises in deriving the boundary conditions for a horizon from the end-of-horizon values of the next later horizon. To allow a full range of possibilities for the x vector, the planning algorithm should be supplied with values $W_m^i(t_i)$ running from 1 to s for each of the r horizons. It is natural to take all s boundary values for the last horizon to be zero. The boundary values for the next to last horizon (horizon 2) depend on the current y_1 since we take $W_m^2(O) = W_{m+y_1}^1(t_1) - W_{y_1}^1(t)$. This provides only $s-y_1$ boundary values for horizon 2. Continuing in this way, we would be able to calculate only $s-y_1-y_2$ boundary values for horizon 3, and so on. A satisfactory way to avoid running out of boundary conditions is to set the missing boundary values for a horizon equal to the highest legitimate value available from the previous horizon. For example, in horizon 2 take

$$W_m^2(O) = W_{s-y_1}^2(O) = W_s^1(t_1) - W_{y_1}^1(t_1)$$

$$\text{for } (m = s-y_1+1, \ldots, s).$$

A Planning Algorithm. The vector, x, which is chosen to maximize

$$\sum_{i=1}^{r} W_{y_i}^i(t_i)\ ,$$

is subject to constraints that reflect the limited number, s, of aircraft and the distribution of takeoff-to-ready times. The planning algorithm could be formulated as a stochastic programming problem, but such a formulation would be hopeless from the computational standpoint. Therefore, we shall retreat by writing the constraints in terms of expected values.

Since we are operating as though resupply occurs only at discrete points in time; a discrete representation of the takeoff-to-ready time distribution will be used. The most useful form in which to express the distribution is achieved by defining:

q_{ij} = Prob (an a/c launched in horizon j will not become ready by horizon i)
for $(i < j, j = 2, \ldots, r)$,
$q_{jj} = 1$ for $(j = 1, \ldots, r)$.

The double subscript notation allows us to specify a different distribution for each launching horizon (j) if necessary. This will be necessary if the horizons are not all of the same length. For each j, there is a distribution of takeoff-to-ready times. Then q_{ij} is the sum of the tail of the distribution over horizons occurring later than i (horizons with indices less than i). Setting $q_{jj} = 1$ is equivalent to assuming that an aircraft cannot be used twice within one horizon.

For planning purposes, assume that all sorties made available within a horizon will actually be used during that horizon. Since JUDGE is intended for use under conditions of resources severely limited relative to expected demands, this assumption should not be far from the truth.

The total expected number of sorties that are unavailable (in the air or being recovered) by the end of horizon i is the sum over each horizon from r down to i of the product of sorties launched during the horizon times the proportion of those sorties that have not yet returned to the ready state. This number of sorties must be less than the total number of aircraft, s. Then the set of r constraints (one for each horizon) have the form:

$$\sum_{j=1}^{r} q_{ij}y_j \leq s \qquad \text{for } (i = 1, \ldots, r) \, .$$

Although only integer solutions to this programming problem are desired, in view of the other approximations being made, one should not object to rounding fractions if s is not a small number.

An inconvenience arises because the objective function is non-linear in the components of x, but this can be remedied by reformulating the problem in terms of another set of variables. Suppose that x is an allocation vector. For each j, (j = 1, . . . ,r), define

$$z_{jk} = \begin{cases} 1 & \text{if} \quad k \geq y_j \\ y_j - k - 1 & \text{if} \quad y_j < k < y_j + 1 \\ 0 & \text{if} \quad y_j + 1 \leq k \end{cases}$$

Going the other way, given z_{jk} (j = 1, . . . ,r; k = 1, . . . ,s), then

$$y_j = \sum_{k=1}^{s} z_{jk} \qquad \text{for } (j = 1, \ldots, r) \, .$$

Also define

$$c_{jk} = W_k^j(t_j) - W_{k-1}^j(t_j) \qquad \text{for } (j = 1, \ldots, r \, ; \; k = 1, \ldots, s) \, .$$

The variable z_{jk} has a value of 1 if the k^{th} sortie should be assigned to the j^{th} horizon and 0 if not. The coefficient c_{jk} is the incremental value of assigning the k^{th} sortie in the j^{th} horizon.

Consider the linear programming problem:

$$\text{maximize} \quad Z = \sum_{j=1}^{r} \sum_{k=1}^{s} c_{jk}z_{jk} \, ,$$

$$\text{subject to} \qquad \sum_{j=1}^{r} q_{ij} \sum_{k=1}^{s} z_{jk} \leq s \quad (i = 1, \ldots, r)$$

$$0 \leq z_{jk} \leq 1 \quad (j = 1, \ldots, r; \; k = 1, \ldots, s) \, .$$

In any reasonable situation, one would expect that $c_{jk} \leq c_{j(k-1)}$ (decreasing marginal utility of additional sorties), which would imply that $z_{j(k-1)} = 1$ if $z_{jk} > 0$. This assures us that a solution to the above problem would not, for example, refuse to assign the twelfth sortie and simultaneously indicate that there should be a thirteenth one, and that there would be at most one fractional z_{jk} for any j. Then the two programming problems are equivalent. Very efficient computer codes for solving quite large linear programming problems exist, and good methods for solving the "zero-one" problem are becoming available.

Forecasts of future mission requests are incorporated into JUDGE through the specification of request rates and the joint distributions of the mission characteristics v and p for each horizon. These are used directly in computing the dispatching rule and indirectly in planning, since the planning technique uses the dispatching calculation.

In an implementation of JUDGE, it is likely that the planning computation would be carried out several times in the course of a single day, perhaps at the beginning of each horizon. This would make it possible to incorporate forecast modifications if they were indicated after the beginning of a day. It would also permit the system to adapt to the recovery of aircraft that have been dispatched in earlier horizons. When redoing the planning calculation, the current fleet status would be incorporated into the constraint set of the linear program, and the new forecasts would be used in the dispatching computations that provide the objective function.

Although a formal forecasting system for JUDGE has not yet been developed, experience in previous days, modified by knowledge of daily plans, should provide reasonable estimates of future demands.

A Numerical Example of the Dispatching Computation. A computer program capable of calculating the dispatching rule for simple examples has been written in order to examine certain aspects of the dispatching rule. This program was used to generate the data for Figure 2. In the interest of simplicity, the program was designed to operate with a simpler mission value and probability structure than that described previously. Instead of working with an analytical formulation of u(x:v,p) and a joint distribution of v and p, the program can accommodate up to eight arbitrary mission value functions in the form of a table. These functions are indexed by a single parameter. The distribution functions F and G are replaced by a set of up to eight probabilities representing a distribution over the indices of the mission value functions.

For the numerical results to be discussed, eight mission value functions were used. None gave an improvement in immediate reward for more than eight sorties or offered a reward greater than 8.0 when the maximum of eight sorties was dispatched. All results are based on boundary conditions of $W_n(O) = 0$; n was carried up to 48 aircraft, and t was taken to 5.0 hours. The data used to plot the left side of Figure 2 represent a portion of the results for the particular case of $\lambda = 6.0$ requests per hour. (The time units are rather arbitrary; by relabeling the time scale this would be equivalent to a two-hour horizon with a request rate of 15 per hour, or any other combination for which the expected number of requests in the horizon is 30.)

The primary outputs of the program are the solutions to the value equations and the dispatching rule in the form of a three-dimensional table. This table specifies the number of aircraft to be sent as a function of: the number of remaining sorties, the time left in the horizon, and the index number of the mission value function appropriate to the request. In addition, the program has the ability to test the dispatching rule with sequences of requests generated by the Monte Carlo method.

Table 1 illustrates how the dispatching rule behaves with respect to a particular type of mission request (i.e., a fixed mission value function) as the request rate, number of sorties available, and time remaining are varied. The sequence of appropriate decisions as a function of time is given for three values of λ and six values of n. As one would expect, the generosity with aircraft increases with the number of sorties left and as the end of the horizon approaches, but decreases with higher request rates.

The data of Table 2 are the result of a Monte Carlo experiment designed to examine the sensitivity of the dispatching rule to errors in forecasting the request rate. Five values of λ were used. The column labeled "Rule Value" contains the values of $W_{48}(5.0)$ obtained from the value equation in the process of calculating the rules. When the forecast is accurate, these numbers represent the expected total value of all requests satisfied. The remaining column headings indicate the actual rates that governed the arrival of mission requests in the Monte Carlo tests. The entries in these columns are the averages of values actually attained for missions dispatched for the indicated combinations of forecasted and actual rates. Within each column, the highest score was obtained when the forecast was correct, but it also appears that the dispatching rule can tolerate fairly high forecast errors without a great deal of degradation in the performance measure.

Table 2—Values attained for various forecasted and actual request rates

λ	Rule Value	Actual Request Rate				
		2.0	4.0	6.0	8.0	12.0
2.0	46.6	45.7	60.4	62.4	64.2	66.0
4.0	61.7	42.9	61.2	66.9	71.1	73.9
6.0	68.5	39.0	59.2	68.7	73.0	78.9
8.0	73.7	36.3	56.6	68.0	73.3	81.9
12.0	83.1	30.2	51.1	62.5	70.6	83.4

Table 1

NUMBER OF AIRCRAFT TO DISPATCH FOR A SELECTED REQUEST TYPE

Time Left		n = 7			n = 15			n = 23			n = 31			n = 39			n = 47		
In Horizon	λ	4.0	6.0	8.0	4.0	6.0	8.0	4.0	6.0	8.0	4.0	6.0	8.0	4.0	6.0	8.0	4.0	6.0	8.0
0.2		7	7	7	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8
0.4		7	5	5	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8
0.6		5	5	0	8	7	7	8	8	8	8	8	8	8	8	8	8	8	8
0.8		5	0		8	7	5	8	8	8	8	8	8	8	8	8	8	8	8
1.0		4			7	5	1	8	8	7	8	8	8	8	8	8	8	8	8
1.2		0			7	3	0	8	8	5	8	8	8	8	8	8	8	8	8
1.4					5	0		8	7	1	8	8	8	8	8	8	8	8	8
1.6					5			8	4	0	8	8	4	8	8	8	8	8	8
1.8					3			8	1		8	8	1	8	8	8	8	8	8
2.0					0			7	0		8	6	0	8	8	4	8	8	8
2.2								6			8	3		8	8	0	8	8	8
2.4								4			8	0		8	8		8	8	3
2.6								2			8			8	5		8	8	0
2.8								0			8			8	1		8	8	
3.0											6			8	0		8	6	
3.2											4			8			8	3	
3.4											1			7			8	0	
3.6											0			5			8		
3.8														3			8		
4.0														1			8		
4.2														0			7		
4.4																	5		
4.6																	2		
4.8																	0		

An interesting feature of the dispatching rule is suggested by Figure 3, obtained from the Monte Carlo experiments. On this diagram are plotted the inventories of remaining sorties as functions of time for various actual request rates when the rule used was calculated from a forecasted request rate of 6.0. The rule seems to have a built-in ability to correct itself. When demands are very high, the number of sorties available dropped rapidly, causing the criteria for dispatching a given number of sorties to become more stringent. Thus, even when the request rate is badly underestimated, some resources are still conserved throughout most of the horizon.

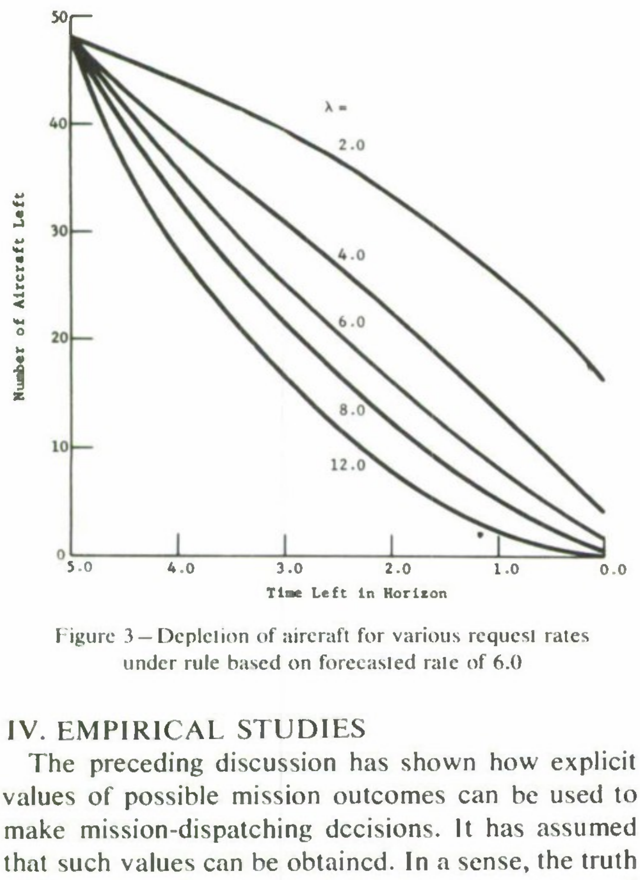


Figure 3 – Depletion of aircraft for various request rates under rule based on forecasted rate of 6.0

IV. EMPIRICAL STUDIES

The preceding discussion has shown how explicit values of possible mission outcomes can be used to make mission-dispatching decisions. It has assumed that such values can be obtained. In a sense, the truth of the assumption is self-evident. A man, presented with a table in which a set of such values is to be entered, can easily be persuaded to enter numbers. The question is not whether such numbers can be obtained, but whether they are appropriate bases for decision. A general and abstract discussion of this and some related questions is reserved for the conclusion of this Memorandum.

Situations exist in which an appropriate external standard of value can be defined. One necessary, though far from sufficient, property that judged values should have if they are to be appropriate bases for decision is that they should in such situations agree reasonably well with the external standard. A study exhibiting that property is described below.

Another possible criterion for appropriateness of judged values is that judges should agree. But this criterion is tricky. Taken literally, it denies the obvious truth that men may disagree about values even when they know the same facts. Still, a system whose performance depends entirely on the identity of its value judge is uncomfortably subjective—though perhaps realistically so.

One way out of the dilemma is to exclude the man from the definition of the system by saying that the system exists for the purpose of implementing its value judge's values as effectively as possible. If human values in fact differ in irreconcilable ways, no other position is possible, since all decision systems must be based on human values in one way or another. But the facts may permit a less subjective resolution of the problem. Presumably training should serve to create a communality of values. If so, men should agree to some extent on their value judgments if they have been trained at all, and the extent of that agreement should be an increasing function of the amount and communality of their training and experience. If this kind of agreement is desired, then an appropriate performance criterion for JUDGE (or perhaps for the larger system, including selection and training procedures for its value judges, in which JUDGE is embedded) would be that it enhances this kind of agreement, as compared with its competitors. This is, of course, a property that can be studied in experiments and simulations.

This section presents the results of two studies: an experiment on methods for collecting value judgments, and a field study permitting some comparison of JUDGE with current procedures not based on explicit value judgments.

The basic idea of JUDGE leaves unanswered many questions about best system design. To provide a vehicle for answering such questions (especially those about the effect of operator training and experience on JUDGE's performance) and to permit more extensive comparison of JUDGE with current procedures than was possible in the field study, a laboratory simulation of a TAC control system environment is necessary.

The methodology experiment

An experiment was run to determine the best procedure for asking human subjects for value judgments. Three procedures were used to estimate the prices of used automobiles.

1. **Ratio.** The subject said how many times more or less valuable a car was than a carefully defined

standard car, by placing a mark on a scale having the digits 1 through 5 logarithmically spaced on it, and checking beside the word MORE or LESS to indicate the direction of the judgment.

2. **Difference.** The subject said how much more or less valuable a car was than the standard car, by placing a mark on a scale the same size as the one for ratio judgments, but with figures of $500 to $2000 linearly spaced on it, and marking MORE or LESS to indicate the direction of the judgment. Both this scale and the ratio one were open ended at the top. The standard car was the same one used in the ratio procedure.
3. **Direct.** The subject simply estimated the value of the car he was judging.

The stimulus material consisted of four equivalent sets of 10 automobiles, ranging in retail price from about $500 to about $4500 in the Kelly *Blue Book*, and described as they would be in an advertisement for used cars. One car, priced at $1800, was included on all four lists, and was used as the standard in the two procedures that required one. A fourth task, that of paired comparisons, was administered to the subjects, but the results from this task are not reported here. (The laboriousness of paired comparisons precludes their use in systems requiring numerous human judgments.)

The subjects, 40 volunteer college students who were paid for their two hours of participation in the experiment, were randomly divided into four groups. Each group received a different pairing of the four procedures with the four groups of cars. Thus each subject used all procedures and judged all cars, making a total of 40 judgments. The combination of groups of subjects, procedures, list of cars, and order of presentation was specified by a Greco-Latin square design so that all order effects and effects of combinations of procedures and lists of cars would be completely counterbalanced.

The subjects' responses were transformed into dollar values. For the ratio procedure, this was done by multiplying the *Blue Book* value of the standard by the subject's judgment for the other car. For the difference procedure, the subject's judgment for the other car was added to or subtracted from the *Blue Book* value of the standard. For the direct estimation procedure, the ratio of the subject's estimates for the other car and for the standard car was calculated, and then processed by the same procedure used for the ratio scale estimates. (This served to adjust the direct estimates for a form of individual bias.) The performance measure was the mean square error for each subject on the nine cars he judged by each procedure, the error being the difference between the estimated dollar value and the "true" *Blue Book* value.

An analysis of variance was performed on these error scores. The Greco-Latin square design of course precludes calculation of variance estimates for interactions. None of the methodological variables (groups of subjects, lists of cars, and order of presentation) made any systematic difference to error. The effect of scaling procedure was highly significant; inspection of mean confirms this conclusion. The mean squared error (times 10^{-5}) for the ratio procedure was 31.86; for the difference procedure, 6.62; and for the direct estimation procedure, 8.37.

On the basis of these results, the difference procedure was chosen for use in the field study described below. Although it and the direct procedure are equally attractive on the basis of these data, the availability of a natural scale of value (that is, dollars) for used cars makes the direct procedure more appropriate for them than it would be in general.

To estimate the overall reliability of the groups of subjects, we calculated the product-moment correlation coefficients between the values obtained by each procedure and the *Blue Book* values. Table 3 shows the means over subjects within procedures and the standard deviations for the three procedures. Note that the correlations are gratifyingly high. This means not only that the subjects agreed well with the *Blue Book*, but also that they agreed well with one another.

As a further check on intersubject agreement, we calculated product-moment correlations between all possible pairs of subjects within each group of ten subjects. The means and standard deviations of these correlations are shown in Table 4. The size of these numbers is evidence of the high reliability of the estimation process.

Table 3 – Mean product-moment correlation coefficients between estimated and actual values over subjects within procedures

Ratio		Direct		Difference	
M	σ	M	σ	M	σ
.894	.066	.810	.074	.819	.084
.787	.116	.900	.046	.729	.150
.822	.105	.886	.132	.919	.051
.887	.078	.927	.135	.926	.051

Table 4—Means and standard deviations for product moment correlation coefficients between all subject pairs within each group of ten subjects

Ratio		Direct		Difference	
M	σ	M	σ	M	σ
.779	.114	.800	.113	.789	.115
.709	.157	.849	.086	.712	.130
.665	.192	.769	.166	.887	.063
.824	.090	.925	.043	.876	.062

The field study

The field study was performed to explore techniques for obtaining value judgments in relatively realistic situations, and to examine JUDGE's performance based on those value judgments. The exercise, carried out in a classroom-like situation, used scenario materials based on an unclassified lesson plan developed at the U.S. Army Command and General Staff College at Fort Levenworth, Kansas. Seventeen Air Force officers familiar with tactical air warfare served as subjects. These were a group of Forward Air Controllers and Air Liaison Officers stationed at Cannon Air Force Base in Clovis, New Mexico.

The subjects were given a brief description of the political situation leading to the hypothetical conflict, and a detailed description of the battle situation was given using a large scale map. The battle was set in Southeast Asia, and involved three divisions of allied forces operating against a roughly comparable enemy force. The subjects completed the entire experiment described in the remainder of this paragraph in a single three-hour session. The experiment was concerned with two successive days of a battle, and two hours of each of these days were simulated. Each two-hour period is referred to as a situation in the experimental design. Additional narrative material provided the transition from the first to the second day.

During each simulated two-hour period, a sequence of targets was presented, one at a time. A target report was a printed form giving a description of the target, its location, the source of the report; for the DASC* system, a mission effectiveness function and the time of the report were also included. A sample target report for the DASC system appears as Figure 4. For the DASC portions of the experiment, subjects were told that the mission effectiveness function associated with a particular target embodied all relevant information concerning aircraft and ordnance performance. The graph of a mission effectiveness function indicates the probability that the target will be destroyed or, in the case of a distributed target, the portion of the target expected to be destroyed as a function of the number of aircraft dispatched.

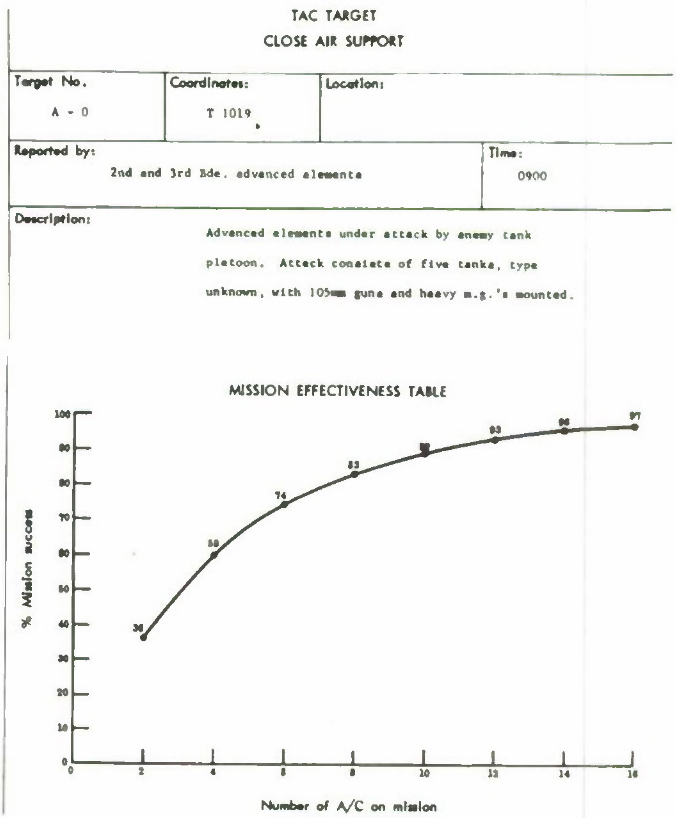


Figure 4—Target report for DASC

To compare JUDGE with sortie allocations made in the usual way (the DASC system), the subjects were exposed to each battle situation twice. Acting as decision makers in the DASC system, the subjects allocated specific numbers of sorties against the targets as they appeared; acting as value estimators in the JUDGE system, they assigned values to the targets as they appeared. The subjects were instructed to try not to let their responses under one operating mode affect their decisions under the other. Two groups of subjects were formed and the ordering of systems was counterbalanced as shown below.

	Situation 1		Situation 2	
Order	1	2	3	4
Group 1	DASC	DASC	DASC	DASC
Group 2	JUDGE	JUDGE	JUDGE	JUDGE

*DASC is the authors' abstraction of the current operating system.

Before operating in the DASC mode, the subjects were told that they had 40 sorties to dispatch in the two-hour situation and that they should expect to receive requests at the rate of about ten per simulated hour. (The actual times on the target reports were selected by a random process which yielded 18 targets for the first situation and 22 targets for the second.) Six different mission effectiveness functions of the form $\eta(x) = 1 - (1 - p)^{x^{\alpha}}$ were used and were displayed to the subjects during the instructional period. The subjects were told that the distribution of effectiveness functions over targets would be uniform.

For the JUDGE task, the target described in Figure 4 was used as a standard. The subjects were to associate a value of 100 with this target and to consider an utterly worthless target as having a value of 0. All other targets were to be valued relative to these fixed numbers. The value to be judged was the importance of destroying the target.

To convert the subjects' value responses into dispatch decisions, a computer program was written to calculate the JUDGE dispatching rule. This program treats the prior distributions of value judgments and mission effectiveness functions as independent. In testing the JUDGE system, two different value distributions have been employed. Distribution U is a uniform (rectangular) distribution with a range of values between 0 and 150; distribution T is triangular with a mode of 0 and gives nonzero probabilities for values up to 225. Both of these one-parameter distributions have means of 75, their major characteristics are shown in Table 5. Distribution U was chosen because of its simplicity, while distribution T was suggested by observing a histogram of all value responses obtained in the field study. This histogram was constructed with very broad intervals to smooth over subject preferences for certain round numbers.

Table 5 — Prior distributions of values used for JUDGE

Distribution	U (Uniform)	T (Triangular with mode 0)
Density Function	$\frac{1}{c}$ for $(0 \leq v \leq c)$	$\frac{2}{c}(1 - \frac{v}{c})$ for $(0 \leq v \leq c)$
Mean	$\frac{c}{2} = 75$	$\frac{c}{3} = 75$
Standard Deviation	$\frac{c}{2\sqrt{3}} = 43.3$	$\frac{c}{3\sqrt{2}} = 53.0$

As a prior estimate for the distribution of mission effectiveness functions over targets in each situation, we used the actual distribution taken over both situations. This distribution assigned nearly equal probabilities to the six functions and was not exactly correct for either situation 1 or 2. In addition, the program was supplied with the forecasted rate of 10 requests per hour, the horizon length of two hours, and the initial availability of 40 sorties.

For each simulated time point, t, at which a target was presented during the DASC phases of experiment, a vector of values, $\overline{W}_m(t)$, (m = 0, 2, . . . ,40) was stored. Given a subject's sequence of value judgments, the corresponding JUDGE dispatches can then be easily calculated. In both systems, missions were required to have multiples of two aircraft.

The means and standard deviations of the subjects' value responses are tabulated in Table 6. Comparing these data with the information in Table 5 indicates that both of our prior value distributions were in considerable error for many of the subjects. To be useful, however, JUDGE must be robust against forecast errors.

Table 6—
Means and standard deviations of value judgments

Group	Subj	Situation 1		Situation 2	
		Mean	Std-Dev	Mean	Std-Dev
1	1	55.8	47.7	58.3	45.5
	2	120.0	51.6	122.9	58.4
	3	68.1	58.8	92.5	68.8
	4	67.8	63.1	106.9	82.2
	5	85.0	41.8	84.6	59.4
	13	45.6	45.0	53.8	59.9
	14	47.5	49.2	58.1	48.8
	15	82.8	52.6	96.0	66.1
	16	75.2	36.0	90.4	23.5
	17	71.7	58.5	72.5	55.7
	Mean	72.0	50.4	83.6	56.8
	Std-Dev	20.5	7.9	21.6	14.8
2	6	124.2	54.1	120.2	40.6
	7	106.9	44.7	95.0	30.0
	8	65.6	27.7	69.8	36.3
	9	96.7	54.4	60.6	29.4
	10	68.6	23.3	80.6	35.1
	11	91.1	32.3	76.2	41.9
	12	57.5	51.9	53.1	39.6
	Mean	87.2	41.2	79.3	36.1
	Std-Dev	22.5	12.2	20.9	4.6
	Grand				
	Mean	78.2	46.6	81.8	48.3
	Std-Dev	22.6	10.9	21.4	15.5

Comparison of JUDGE and DASC

To provide a basis for a meaningful comparison of JUDGE and DASC, we examined two other dispatching rules. One was a theoretical optimum, the best that could have been done with perfect foreknowledge of the incidence and value of all requests during the two-hour period. In the absence of such perfect foreknowledge, this optimum is of course unattainable. To calculate the perfect sequence of dispatching decisions, pairs of planes were assigned to targets in order of decreasing marginal utility, ignoring the sequence in which the targets arrived, until all planes were used up. This optimum differs from one subject to another, since it is based on value judgments.

The other dispatching rule was first-come, first-served (FCFS). We recognized that any dispatching rule, no matter how absurd, would be certain to ob-

Table 7—
SYSTEM PERFORMANCE: SITUATION 1

Subj	Expected Utility					Effectiveness		
	FCFS	DASC	JUDGE(U)	JUDGE(T)	Optimum	DASC	JUDGE(U)	JUDGE(T)
1	238	395	450	447	480	0.65	0.88	0.86
2	351	419	532	548	619	0.26	0.68	0.74
3	251	333	449	445	459	0.40	0.95	0.94
4	193	297	398	394	417	0.46	0.92	0.90
5	341	435	531	530	530	0.50	1.00	1.00
6	329	730	772	779	850	0.77	0.85	0.86
7	470	651	754	774	777	0.59	0.92	0.99
8	246	352	439	439	441	0.54	0.99	0.99
9	374	545	662	652	674	0.57	0.96	0.93
10	224	325	404	395	433	0.48	0.86	0.82
11	342	471	578	579	594	0.52	0.94	0.94
12	177	249	404	408	420	0.30	0.94	0.95
13	171	159	346	342	375	0.06	0.86	0.97
14	310	472	465	462	497	0.86	0.83	0.81
15	249	329	466	451	496	0.33	0.88	0.82
16	290	377	467	464	499	0.42	0.85	0.83
17	368	376	509	512	543	0.05	0.81	0.82
Mean						0.448	0.888	0.893

Table 8—
SYSTEM PERFORMANCE: SITUATION 2

Subj	Expected Utility					Effectiveness		
	FCFS	DASC	JUDGE(U)	JUDGE(T)	Optimum	DASC	JUDGE(U)	JUDGE(T)
1	328	410	436	461	467	0.59	0.78	0.96
2	514	629	666	708	769	0.45	0.60	0.76
3	362	493	539	550	563	0.65	0.88	0.94
4	410	536	592	605	629	0.58	0.83	0.89
5	374	442	563	562	603	0.30	0.82	0.82
6	492	658	769	783	841	0.48	0.80	0.84
7	389	581	683	694	695	0.63	0.96	1.00
8	310	456	510	524	536	0.64	0.86	0.95
9	275	331	411	408	424	0.38	0.91	0.89
10	313	510	552	575	575	0.75	0.91	1.00
11	282	323	488	479	488	0.20	1.00	0.96
12	176	250	296	282	296	0.62	1.00	0.88
13	292	325	427	461	461	0.20	0.80	1.00
14	241	496	499	489	536	0.86	0.88	0.84
15	383	454	551	580	626	0.29	0.69	0.81
16	432	488	603	599	616	0.30	0.93	0.90
17	247	494	555	559	567	0.77	0.96	1.00
Mean						0.511	0.861	0.908

tain some expected utility if it dispatched aircraft at all. Only improvements over some minimal performance level should be given credit. So to specify such a minimal system, we chose to dispatch planes four at a time to all targets until planes were exhausted.

Table 7 shows the results for situation 1, and Table 8 for situation 2. The first five columns contain the expected utilities earned by FCFS, DASC, JUDGE (U), JUDGE(T), and the optimum system for each subject, based on each individual's value judgments. The column headings JUDGE(U) and JUDGE(T) are, respectively, JUDGE computed with the U and T value distributions described in Table 5. The last three columns contain relative effectiveness numbers for DASC and the two versions of JUDGE. For each subject, they are calculated by treating the expected utility obtained by FCFS as 0 and the expected utility obtained by the perfect system as 1. With this definition of the origin and a unit of measurement of the utility function, DASC performs about 50 per cent as well as the unattainable optimum, while JUDGE performs about 90 per cent as well as the perfect system.

The large discrepancy between JUDGE and DASC indicates, as expected, that JUDGE is much more efficient in implementing a subject's values than the subject is himself. JUDGE separates the evaluation portion of the dispatching task (the portion that depends on human expertness) from the decision making portion, which, given the value judgments, is a difficult computational task that a computer can perform more effectively than a man.

It is important that DASC, while much inferior to JUDGE, is more superior to FCFS. Since the scores justifying this assertion are based on the value judgments, this finding means that the DASC decisions are by no means unrelated to the JUDGE value judgments. It is reasonable to believe that each subject's JUDGE dispatches and his DASC dispatches are attempts to implement the same set of values; the JUDGE dispatches are simply more effective at doing so.

An incidental observation makes the same point. One of the authors, familiar with the stimuli ahead of time, and with complete knowledge of the scoring rules, performed the role of a subject twice. Though he attempted to make DASC dispatches that would produce high scores, his data closely resembled that reported in Tables 7 and 8. It is simply difficult to translate a value system into dispatching decisions, and JUDGE does it much better than men can.

That JUDGE(U) and JUDGE(T) are so nearly alike further indicates that JUDGE is rather robust under variation in the distribution of value judgments from its prior expectation.

We attempted to scale the subjects' situation 2 responses based on their situation 1 value judgments to see if the effectiveness of JUDGE could be improved by making a subject's distribution of value judgments correspond more closely to the forecasted distribution. Since there was a considerable degree of correlation between the subject's means and standard deviations between the two situations, it is reasonable to suppose that it would be profitable to modify a subject's responses using statistics obtained from observing him at an earlier time. The mean and standard deviation of distribution U are 75 and 43.3 respectively. Suppose that a particular subject's mean and standard deviation in situation 1 were m and s. Letting x represent a response in situation 2, an adjusted response for that target was computed by the formula

$$x_{adj} = \frac{43.3}{s}(x - m) + 75.$$

This simple adjustment is just a stretching (or compressing) and a movement of the origin of the subject's response scale based on his previous responses.

The application of this adjustment technique resulted in very modest effectiveness increases for JUDGE(U). That the improvement was only slight was to be expected from the already high effectiveness of JUDGE and from the large amount of robustness already evident.

Intercorrelations among subject's scores

So far, evidence indicates that JUDGE implements each subject's values better than his own decisions can. Ideally, we would like to show the relation between an individual subject's value judgments and some ultimate criterion of value, such as "winning the war." Unfortunately, we do not know how to derive target values from any such ultimate criterion. Indeed, if we could do so, such calculated values rather than human judgments should be the values that JUDGE translates into decisions.

Thus, no ultimate validation of JUDGE, or of any command system, is possible. The philosophical basis for this conclusion is the subject of the last section of this Memorandum. However, relevant questions can be examined. For example, intra-subject and inter-subject reliability are both relevant. If a subject's value judgments collected at one time systematically differ from his value judgments for the same targets in the same situation collected at a different time, there would be some doubt about the appropriateness of implementing either set of values. Unfortunately, the field study provides no information about intra-subject reliability since there was no replication. However, the use of many subjects permits us to examine the extent to which one subject agrees with another.

Table 9 presents the mean intercorrelations among subjects within groups and situations for three sets of numbers: DASC dispatching decisions, value judgments, and JUDGE dispatching decisions based on those judgments. Three points can be noted about these numbers. First, they are all low. Clearly these subjects disagreed with one another both about how valuable the targets were and about how many planes to send against each. Second, in every case, the mean intercorrelation between JUDGE dispatching decisions is higher than that for the DASC decisions and higher than that for the value judgments on which they are based. These decisions reflect not only those values, which differ from subject to subject, but also mission effectiveness functions and times at which the requests arrived, both constant across subjects. It is gratifying that the JUDGE intercorrelations are higher than those for DASC. Perhaps the most suggestive feature of Table 9, however, is that the mean intercorrelations for situation 2 are higher in all cases than those for situation 1, and JUDGE gains more than DASC. Clearly learning is going on, and it seems unlikely that it has reached its limits.

Table 9—

Average correlations of subjects with all other subjects

Group	Situation 1			Situation 2		
	DASC Dispatches	JUDGE Values	JUDGE Dispatches	DASC Dispatches	JUDGE Values	JUDGE Dispatches
1	0.346	0.413	0.461	0.396	0.539	0.550
2	0.252	0.238	0.358	0.396	0.530	0.566

Further experimentation in a laboratory where greater demands on a subject's time can be made is necessary to establish the upper limits of learning and both kinds of reliability. The field study has established that JUDGE works; it is much superior to DASC, and it is robust against various kinds of deviations from prior expectation.

V. GENERAL COMMENTS ON VALUE-JUDGMENT BASED COMMAND SYSTEMS

As we worked on JUDGE, we found ourselves strongly influenced by a set of philosophical ideas about judgment-based systems in general. These ideas have to do with the purpose, design, and evaluation of such systems. Most of our ideas about the TAC problem follow from these more general considerations—though many of them could be justified from other, less radical, points of view than the one we present here. Since we have seen no other presentation of the position we have come to, and indeed very little discussion of the issues underlying validation of command systems exists on paper, we have chosen to end this Memorandum with a fairly extended discussion of these philosophical questions.

We believe that all command systems are and must always be judgment-based. A command system exists to make decisions. We find it useful to distinguish between the decision, or selection of an action, and the decision process. A decision process describes a complex sequence of events beginning with recognition that an action will have to be selected, and ending with implementation of the selected action. We believe that human judgments play an absolutely necessary role in all decision processes, at least for decisions important and interesting enough to concern command systems. Our goal is to analyze the decision process into functions, to ascertain how and by what means each function should be performed, and to allocate those functions best performed by men to men, and those best performed by machines to machines. The assertion that all command systems are and must always be judgment-based means, then, that the set of functions best performed by men will never turn out to be empty. One reason for that assertion is that command systems exist to serve human purposes, and require men to specify what those purposes are. We believe also that, at least for a long time to come, men will be indispensable for a number of other functions in the decision process, as the following discussion exhibits.

The fact that all command systems are judgment-based is often not explicitly recognized because of a faulty definition of system boundaries. The statement is sometimes made, for instance, that "the function of a command system is to help the commander do his job." This is too narrow a definition of the system. In the definition that we consider appropriate, the commander is a part of the system, and the function of the entire system, commander and all, is to make "the right decisions."

When is a decision right? Everyday evaluation of decisions is usually done by comparing the choice actually made with the one that would have been made either by the man doing the comparing or by some appropriate authority. If the person making the comparison is the one who made the decision, then most decisions are right by definition, and this definition of rightness is uninteresting. If someone else judges whether a decision is right or wrong, then when he says it is wrong we have two experts who disagree, and usually no satisfactory means of resolving the disagreement.

The search for a way out of this impasse has characterized almost all research on command systems. We feel that a departure is possible only after radical redefinition of what is meant by "right" in this context. The remainder of this discussion presents our redefinition and shows how to use it.

Validity and reliability

The notion that there is a right action, answer, diagnosis, or other system output, and that the problem of system design is to devise a system that produces this right output reminds us of the problem of validating the design of intelligence tests or other personnel selection instruments. The test designer may start out with the rather simple idea that some abstract quantity like, say, intelligence exists and that his task is to measure that quantity in the individuals he tests. He develops a possible method for measurement, and then must consider whether or not it really measures what he hopes it measures. How can he find out?

Two main approaches have been taken to the problem of validating tests. One, the criterion-oriented approach, depends on comparing the test under study with some other, already available measure of the quantity to be measured. If test A is "known" to measure intelligence, then test B can be considered valid if it correlates highly with test A. In the case of intelligence tests, Binet's original idea was to use success in school as his validating criterion.

Criterion-oriented validity has two major difficulties. One is that most criteria are themselves suspect, either because they are of doubtful relevance to the abstract entity to be measured (is success in school really primarily a function of intelligence?) or because as measures the criteria themselves have unattractive properties, such as unreliability, or both. The other is that for most abstract entities we might want to test (e.g. propensity to take risks) no appropriate criterion is available.

Because of these difficulties, the testers have developed a second and quite different approach to validity, which they call construct validity. (There is yet another concept called content validity, having to do with validating tests of mastery of a subject matter, but it can be neglected here.) The basic idea of construct validity is that a test should make sense and data obtained by means of it should make sense. One form of making sense is that different procedures that purport to measure the same abstract quantity should co-vary. If one procedure is taken as a criterion for the other, this is simply criterion-oriented validity. But even if neither procedure is taken as valid, a priori, the fact that they correlate highly indicates that to some extent they measure the same thing or closely related things. If, in addition, the kind of underlying quantity that each might tap seems on some a priori basis to be the same, then observation of covariation is an instance of what has been called converging operations, and contributes to validity. (The concept of construct validity is broader than this description indicates; this discussion serves only to give it flavor.)

A procedure (test, system, etc.), whether or not it is valid, should certainly be reliable. This word, taken from test theory, is a bit too specific for our purpose. We will say that a procedure should be intellectually coherent. One requirement of intellectual coherence is that repeated attempts to measure the same thing, or equivalent things, by means of the procedure should produce the same measurements. Another requirement of coherence is that variables that seem irrelevant to the procedure should not affect it.

Other requirements of coherence exist. For decision-making procedures, logical consistency is one of them. Thus if a decision-making system prefers act A to act B and act B to act C, it should not prefer act C to act A; it should be transitive. Similarly, it should exhibit the properties known in decision theory as avoidance of dominated strategies, independence from irrelevant alternatives, and a few others.

A final class of coherence requirements is harder to describe. There are obvious expectations about the behaviors of certain systems. An information processing system, for example, should not act as though its odds for hypothesis A have been increased as a result of evidence that clearly makes A less likely than it was before. Information that an act has become more valuable than it was before should not cause a decision-making system to become less likely than it was before to choose that act.

Now that the concept of coherence has been introduced, it becomes easier to talk about validity. Validation is simply establishing the coherence of a procedure, or several procedures. Thus no sharp line separates the concept of reliability from that of validity; both concepts refer to agreements among measures, and a continuum exists from cases in which the measures essentially repeat the same procedure (reliability) to cases in which rather different procedures seem to measure the same thing (validity).

The validation-type reliability of judgmental systems

We assert that no external measure of the performance of a judgment-based decision-making system is possible. Any such measure would have to compare the decisions the system made with decisions made some other way, and there would have to be some good reason to suppose that the decisions made the other way were the right ones. But if we reject the idea that the business of a decision-making system is to imitate some individual's decisions (in which case the only point of building the system would be to save the individual the trouble of making those decisions himself), then no basis remains for asserting that the decisions made by one procedure (e.g., by the commander) are inherently appropriate simply because they were made by that procedure, regardless of their content. An examination of the merit of decisions in terms of their content is a matter of intellectual coherence or reliability, not validity.

We assert also that intellectual coherence or reliability is very measurable and is in fact what we want the output of a decision-making system to have. The rest of this philosophical section concerns some thoughts about how to obtain intellectual coherence in judgment-based command systems, and how to establish that it has (or has not) been obtained.

Before going on, however, we pause to answer a possible objection. It is possible to think of command situations and systems within which the quality of system performance is easily defined and easily measured. From a too-superficial viewpoint one could argue, for example, that a business management should maximize dollar return and that dollar return is easily measured. Actually, both of these statements are incorrect, since most businesses have many goals other than maximum dollar return (e.g. market position, stability of employment, maintenance of stock value, image, etc.) and the accounting fictions underlying the definition of profit are the output of an elaborate and basically subjective judgmental process. But lower-level command systems may have rather acceptable external standards of quality of performance. The goal of a cab dispatching system, for example, might be to minimize total customer waiting time—an easily measured quantity.

But why is that goal appropriate, rather than some other, possibly inconsistent goal, such as minimizing number of miles traveled by empty cabs? We can think of no example in which the choice of goal, or of weighting function by means of which to combine goals, is not a judgmental matter. So even command systems in which performance measures are easily identified are inherently and essentially based on value judgments—the value judgments that specify the system's goals.

A task analysis of command systems

As we envision it, the basic function of any command system is to make decisions. The formal analysis of decision-making is extensive and the basic principles are well understood. We will not review that analysis here; such references as Luce and Raiffa (1957) present it. Implementing that analysis in the system design is, we believe, necessary to attain intellectual coherence, but far from easy. Table 10 presents a detailed breakdown of functional steps in implementing the analysis, steps that must be performed in one way or another by any command system. Table 10 also contains our opinions (very much subject to modification) about whether each function should be performed by men, machines, or both, and our opinions (firmer) about whether each function should be performed at the moment of decision or in advance.

Functions 2 through 6 should be performed ahead of time if possible, but it will not always be possible. We are particularly interested in functions 5 and 10, which along with function 2 (about which we know nothing abstract) constitute the basic functions that men should perform in command systems. Functions 5 and 10 correspond to the two basic variables of decision theory: utility and probability. (Functions 10 and 11, as stated, imply a point of view about how relevant probabilities should be obtained in decision-making systems; systems that work this way are called Probabilistic Information Processing or PIP systems, and are discussed by Edwards and others (see Edwards, 1965; Edwards, Lindman, and Phillips, 1964; Edwards and Phillips, 1964; Schum, Goldstein, and Southard, 1966; Kaplan and Newman, 1966). Both of these quantities, we assert, are inherently judgmental; the basic roles of men in command systems are to make these judgments.

Two principles for judgment-based decision system design

Table 10 implies two principles for the design of judgment-based decision systems. These principles have somewhat the status of axioms; they are fundamental to our argument, but not directly demonstrable. We will state and argue for them, but firm establishment of their appropriateness for guidance of system design must come from success of the resulting systems.

Principle 1: The judgments that must be made by men in decision systems should be fragmented into relatively small, elementary parts when possible.

Table 10—

Functions of Command Systems

Function	Performed By	When Performed
1. Recognize that a decision problem exists	men	ahead of time
2. Identify available acts	men	ahead of time if possible
3. Identify relevant states that determine payoff for acts	men	ahead of time if possible
4. Identify the value dimensions to be aggregated into the payoff matrix	men	ahead of time if possible
5. Judge the value of each outcome on each dimension	men	ahead of time if possible
6. Aggregate value judgments into a composite payoff matrix	machines	ahead of time if possible
7. Identify information sources relevant to discrimination among states	men	ahead of time
8. Collect data from information sources	both	at moment of decision
9. Filter data, put into standard format, and display to likelihood estimator	both	at moment of decision
10. Estimate likelihood ratios (or some other quantity indicating the impact of the datum on the hypotheses)	men	at moment of decision
11. Aggregate impact estimates into posterior distributions	machines	at moment of decision
12. Decide among acts by using principle of maximizing expected value	machines	at moment of decision
13. Implement the decision	both	at moment of decision

We see five advantages to using this principle.

1. It permits the automation of significant elements of the decision-making task.
2. It greatly simplifies the task of the human beings working in the system, by reducing the difficulty of each judgment.
3. Because of 2, it reduces the difficulty of training the system operators.
4. It permits allocation of the judgment task to several men rather than one; there is no requirement that all information must ultimately be evaluated by one man. In systems with a high information load, this consideration alone would be enough to justify the principle.
5. Because of 1, it permits machine portions of the system to monitor and insure intellectual coherences of various kinds, either by checking judgments to insure coherence or by performing operations according to rules that guarantee coherence.

Since the fifth point is crucial, it is appropriate to add that examination of actual human decisions, in laboratory (see Edwards, 1954) and other contexts, indicates that incoherences frequently occur in them. In fact, the known rules for intellectual coherence of decisions are sufficiently demanding so that it is extremely difficult by unaided intuition, to make a reasonably large set of decisions without violating some of them. So principle 1, properly applied, can be expected to result in major gains in coherence.

Clearly the functional analysis of Table 10 and the use of principle 1 are appropriate only if men can in fact perform fragmented judgments effectively. In an important sense, it is self-evident that they can, since men can and do generally make reasonably appropriate decisions, and some version of each of the functions listed in Table 5 must be performed, implicitly or explicitly, before any decision can be made. But it is not self-evident that men can perform such functions explicitly, as is necessary to implement the philosophy of system design implied by Table 10. Principle 2 asserts that they can.

Principle 2: Men can make explicit probability and value judgments, using appropriate response mechanisms and after appropriate training. Where external standards of correctness of such judgments are available, human judgments will usually be incorrect, but not severely so, and appropriate system design can minimize such errors. Whether or not external standards of correctness are available, appropriately obtained judgments will be relatively coherent.

Principle 2 is an empirically testable assertion, and a variety of experimental evidence bears on it. Psychophysics is the branch of psychology devoted to the extraction of human judgments about reasonably simple sensory events; a basic conclusion is that men make such judgments very well indeed. (See Stevens and Galanter, 1957.) Recent research on probability estimation indicates that men can judge relative frequencies with great accuracy (Robinson, 1964; Shuford, 1961). More complex probability judgments suffer from an inherent deficiency that has been named conservatism; men are unable to extract from data as much certainty as the data justify. (See Peterson and Miller, 1965; Phillips, Hays, and Edwards, 1966.) But human conservatism in probability estimation can be overcome by appropriate system design; that is the point of the PIP system mentioned above. (See Edwards, Lindman, and Phillips, 1964.) Evidence on human value judgments is sketchy and fragmentary; such as it is, it indicates that men do rather well at translating even rather complicated value systems into numbers. (See, for example, Yntema and Torgerson, 1961.)

But final justification of the use of Principles 1 and 2 for system design can come only from success of the resulting systems.

Judgmental systems are self-validating

The arguments presented above have led us to a set of ideas about how to design judgmental systems. One crucial feature of these ideas is that explicit value or utility judgments lie at the core of such systems. This fact has implications for the validation (in the sense already defined) of such systems.

As we have pointed out, validation in a classical sense consists of demonstrating coherence between the output of a system, in this case decisions, and the comparable output of some other system considered to be effective or valid. With decision systems, however, such acceptable external criteria are nonexistent. The usual expedient is to use some wise and experienced decision-maker's judgments as the criterion. However, unless the goal of the system is merely to reproduce that man's decisions, this procedure is unsatisfactory—especially since any man's decisions are likely to be incoherent to some extent.

As we have already argued, intellectual coherence, taken over as large a domain of thought as possible, is an alternative approach to validation, and the only approach available for the command systems of interest here. And a variety of kinds of coherence are built into the sort of judgmental system implied by Table 10.

Still other kinds of coherence can and should be examined by means of research on any proposed command system; such research can range from highly informal studies of system elements to major formal simulations of the system as a whole.

In the particular case of decision systems based on value judgments, a natural requirement of the system is that the decisions should cause as much value as possible to accrue to the entity in whose service the decisions are being made. Procedures internal to the system will guarantee this, given that the value judgments made by system operators are taken as the "true" values. The question of whether this criterion is met therefore reduces to two other questions. Are value judgments reliable, from time to time within one judge or from one judge to another? If not, can the unreliability be accounted for as true differences in values, from time to time or from one judge to another? This second question is bound to be a matter of opinion, since "true" values are inaccessible and perhaps undefined. Still, the opinion need not be entirely unguided by data.

Obviously such questions of reliability will be extensively studied in the course of system design. So will other forms of intellectual coherence. By the time any command system of the kind implied by Table 10 has been fully designed, formal and empirical information about the various relevant kinds of intellectual coherence will have been built into the design details at almost every point. Thus the design of such a system is self-validating, in a very important sense.

Of course no validation is ultimate; each conclusion that a system is valid for its purpose means no more than that a decision has been reached to proceed to the next step in its design and use. Conclusions reached during system design concerning the level of quality to be expected in system performance will be modified or replaced by conclusions based on the result of simulations; conclusions based on simulations will be modified or replaced by conclusions based on actual system use. And even actual use is not an ultimate criterion; each new use under new conditions may require a new judgment of system validity. Because, in the last analysis, all we can ever mean by stating a procedure is valid is that, on the basis of what we know, it makes sense.

REFERENCES

1 D. R. COX and W. L. SMITH *On the superposition of renewal processes* Biometrika 1954, 41, pp. 91-99.

2 W. EDWARDS *The theory of decision* Psychol. Bull. 1954, 51, pp. 380-417.

3 W. EDWARDS *Probabilistic information processing systems for diagnosis and action selection* In J. Spiegel and D. E. Walker (Eds.) Information System Sciences: Proceedings of the Second Congress Spartan Books, Washington, D. C. 1965.

4 W. EDWARDS, H. LINDMAN and L. D. PHILLIPS *Emerging technologies for making decisions* in T. M. Newcomb (ed.) New Directions in Psychology II Holt, Rinehart, and Winston, New York, 1964

5 W. EDWARDS and L. D. PHILLIPS, *Man as a transducer for probabilities in bayesian command and control systems* in M. W. Shelley and G. L. Bryan (eds.) Human Judgments and Optimality John Wiley and Sons, New York, 1964

6 R. J. KAPLAN and J. R. NEWMAN *Studies in probabilistic information processing* IEEE Trans. Hum. Fact. Electron. HFE-7, 1966, pp. 49-63.

7 G. M. KAUFMAN Statistical Decision and Related Techniques in Oil and Gas Exploration Prentice-Hall, Englewood Cliffs, New Jersey, 1963.

8 W. M. KINCAID and D. A. DARLING *An inventory pricing problem* J. Math. Anal. Applic. 7, 1963, 183-208.

9 R. D. LUCE and H. RAIFFA Games and Decisions John Wiley and Sons, New York, 1957.

10 C. R. PETERSON and A. J. MILLER *Sample size and the revision of subjective probabilities* J. Exp. Psychol. 70, 1965, pp. 117-121.

11 L. D. PHILLIPS, W. L. HAYS and W. EDWARDS *Conservatism in complex probabilistic inference* IEEE Trans. Hum. Fact. Electron. HFE-7, 1966, pp. 7-18.

12 G. H. ROBINSON *Continuous estimation of a time-varying probability* Ergonomics 7, 1964, pp. 7-21.

13 D. A. SCHUM, I. L. GOLDSTEIN and J. F. SOUTHARD *Research on a simulated Bayesian information-processing system* IEEE Trans. Hum. Fact. Electron. HFE-7, 1966, pp. 37-48.

14 E. H. SHUFORD *Percentage estimation of proportion as a function of element type, exposure time, and task* J. Exp. Psychol. 61, 1961, pp. 430-436.

15 S. S. STEVENS and E. H. GALANTER *Ratio scales and category scales for a dozen perceptual continua* J. Exp. Psychol. 54, 1957, pp. 377-409.

16 D. B. YNTEMA and W. S. TORGERSON *Man-computer cooperation in decisions requiring common sense* IRE Trans. Hum. Fact. Electron. HFE-2, 1961, pp. 20-26.

NEW DIRECTIONS FOR AUTOMATED INFORMATION SYSTEMS

PAUL G. GALENTINE, JR., Colonel, USAF, *Chairman*

Session Participants:

JAMES P. ANDERSON
FRANK E. DIAZ
THOMAS L. CONNORS
G. M. NORTHROP

New directions for information systems through advances in machine organization

by James P. Anderson
Fort Washington, Pennsylvania

INTRODUCTION

While automated information systems design has made considerable progress over the past five or six years, it seems apparent that it has fallen far short of the promises and the concepts advanced during the era of the large scale command and control systems that fostered its development. Part of the failure to realize the lofty ambitions of that time appear to be related to our concepts of computers and the ways we are forced to use them.

Presently, we have a new opportunity to affect the structure of a new generation of machines. This opportunity will be occasioned principally because of the very rapid reduction in logic costs being brought about by the developments in microcircuit technology.

It is not the intent of this paper to offer pat solutions to the problems we all agree are quite difficult; rather, it is to suggest several approaches that are feasible in the context of the new hardware technology, and from this perhaps stimulate contributions to machine organization from information systems designers. In this way, we may arrive at machines that will assist them in solving basic problems rather than overcoming the inadequacies of the tools that are given to work with.

Direct execution machines

One of the continuing and pervading problems in the design and development of information systems has been what has come to be known as the programming problem. While not all of the "problem" can be attributed to the nature of the computer systems we work with, sufficient difficulty has been encountered to force the adoption of one of the various higher level programming languages to increase the intelligibility and decrease the time required for preparation of programs.

Even with the present programming languages, much time is spent in compiling and debugging, the latter frequently involving peculiar quirks of a particular machine, for example, how it encodes data. From the point of view of resources used, one must also consider the large amount of effort expended by the manufacturers and various systems development groups in preparing compilers, diagnostic programs, utilities and the like to create an environment for programming information systems. The costs which are ultimately passed on to the users become almost incalculable.

In addition, in order to permit user independence from particular hardware there has been a major effort at standardizing the various languages as well as adoption by the various services and government agencies of one of the languages as a required standard for their problems. The Air Force, for example, has adopted JOVIAL as the standard for command and control systems while the DOD has placed increased emphasis on use of COBOL for many applications.

As a result of these language standardization efforts, one might consider establishing requirements for machines that directly execute programs in one of the standard languages as a means of simplifying some of the programming problems. Earlier work has shown the feasibility of this approach (Anderson, 1961; Bashkow, 1966). With current software costs exceeding the development costs of hardware, the economic pressure for adoption of this kind of approach will become even greater for the next round of machines.

With the objection to direct execution machines based on the additional cost of logic being rapidly eroded, the only remaining objection centers mainly on the single language capability of such a machine. This objection would not be relevant for those situations where *de facto* standards are insisted upon for various reasons, and could be overcome altogether by structuring such machines to have the language analysis logic separable (perhaps by plug-in units) from the main part of the machine. The current emulator technology suggests that a basic machine structure can be adapted to a variety of purposes by varying the logic itself.

It is not suggested that the direct execution machines will be applicable in all information system situations, but for many applications, particularly those involving dedicated systems, this approach could be quite powerful. When combined with on-line techniques, one might even anticipate simpler control programs by reducing the interface with compilers, and providing direct language debugging.

Parallel organizations

The present generation of computer systems has been different from its predecessors in one major respect, that is, nearly every manufacturer offers a multiprocessor version of one or more machines in his line. The initial emphasis on this kind of organization and the various multicomputer organizations that were both contemporary and preceded it, were for reliability and systems balance. However, due to the R&D on executive systems for multiprogramming control for both time-sharing applications and multiprocessor management, it became apparent that it would be possible to exploit the multi-processor structure to give better performance through parallel operation on a single problem. To this end, some of the control languages for present machines have been enriched to permit programmer specification of opportunities for parallel execution of segments of the same program. While this is an important step, it is insufficient to achieve the desired levels of performance in information systems.

Under different assumptions from those in the preceding section of this paper, it is clear that it would be possible to achieve quite spectacular improvements in performance by systematically exploiting all parallelism inherent in every program. In order to be most effective, such systematic exploitation must be based on automatic detection of opportunities for parallel execution. Reliance on programmer specification of such opportunities is both unnecessary and burdensome.

The programming technology is sufficiently advanced to handle parallelisms in the fine. A method for detecting parallelism in algebraic languages is outlined in the appendix. Another method, and a system structure for exploiting parallelisms in algebraic expressions is outlined in an article in the February 1966 issue of the *IEEE Transactions on Electronic Computers* (Hellerman, 1966).

While exploiting parallelism in computational languages is certainly feasible, one area that would be more fruitful to examine for information systems application is the possibility of organizing highly parallel systems for data processing. The basic techniques are apparent: Data partitioning, exploiting repetitive operations on files (such as extraction of a subset of a file for subsequent processing) and exploiting parallelism at the statement level in a manner similar to that found in the proposals for purely algebraic languages.

In addition to multiple processor structures already in existence, the design of highly parallel data processing oriented systems requires many more channels into and out of disc files or other mass storage devices than are presently available. It is possible that virtual channels patterned after the multiplexed drum technique used on the GE 645 will provide a necessary increase in performance when applied to head per track disc files.

The reason highly parallel data processing systems are believed to be important is the improvement in overall performance that would accrue to almost any kind of information system through the reduction in sorting time alone, using data partitioning techniques.

The present lack of a generalized multi-file data retrieval and reporting program on any systems known to the author seems to be related at least in part to the sorting that would be required.

With very few exceptions there has been little innovation to the basic structure of general computing systems since Mauchly and Eckert designed EDVAC and UNIVAC. The reasons for using highly stylized encoded instructions for commands are being rapidly dissipated, as are the reasons for the treatment of programs as sequential processes. The many problems in information systems that are tied to programming and performance may be unnecessary and exist only because we choose to live with information systems that are not suited to our requirements. The means for changing this situation is at hand, and it seems fairly certain that new machines, both direct execution and highly parallel, will be developed.

New developments will not be limited to the two suggested above, nor is it intended to suggest that direct execution machines and highly parallel organizations will solve all problems in automated information systems. What is intended is to suggest that insufficient attention is paid to the design of proper hardware tools for this kind of problem, compounding the software problems that are enormous enough in themselves.

Appendix A: an approach to explicating parallelism in algebraic languages

Nature of sequentiality and parallelism

The study of parallelism in programs is in fact a study of required sequentiality within a program. There are two sources of sequentiality (and parallelism) arising from the precedence of the arithmetic operator, and the sequentiality enforced by the availability of the operands.

As an example of the first form, consider the expression a·b-c/d+e. Given the machine capability one could compute the partial results a·b and c/d in parallel before proceeding with the sequential operations of subtracting the partial results and adding e. A trivial example of the second form is shown below:

(1) A = B
(2) C = D
(3) E = A + B

Here the first two assignment statements could be executed in parallel, while the third would have to be deferred until the first statement was completed.

The simple examples shown above illustrate two of the basic concepts involved in the detection of implicit parallelism within programs. There is one additional concept of importance, that of "regions." The region is the span over which implied parallelisms can be logically sought. The nature of regions is closely related to the notion of scope found in the constructs of ALGOL. Examples of regions are the scope of a For Statement, the true clause of a Conditional Statement, the false clause of the same kind of statement, Compound Statements, etc. In general, the regions are bounded by statements that interrupt the implied control flow.

In addition to the natural parallelism of programs, there are a number of special cases that occur with sufficient frequency to warrant inclusion in any consideration of automatic analysis of implied parallelism. One construct that can be exploited is repetitive operations on vectors and arrays. Thus, the inner product of two vectors could be recoded as a set of individual multiplications and a tree of additions, or as a set of interactions over segments of the vectors. Similarly, the multiplication of two conformable matrices can be expanded into independent computation of each element of the resultant product matrix. These special cases will not be discussed further here.

Techniques for analyzing intra-expression and inter-statement parallelism

The basic technique for analyzing and detecting intra-expression parallelism is to convert the expression into a tree form using techniques of algebraic expression analysis found in compilers. The execution order for each binary operator in the expression can be derived directly by associating with each operand and partial result entering into the compilation, a level number derived according to the following rules:

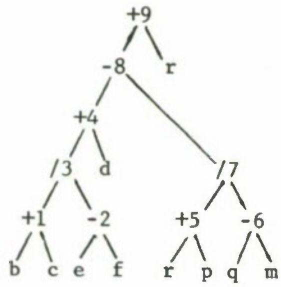
(1) All variables (including constants) are of level 0.

(2) For each partial compilation, find:
$L_T = MAX(L_1,L_2)+1$, where L_T is the level of the result
L_1 is the level of operand 1
L_2 is the level of operand 2

The level numbers are the execution order for each of the operators in the expression. As an example, the expression

(b+c)/(e–f)+d–(r+p)/(q–m)+r

would result in the following tree:



The numbers indicate the order in which the nodes are generated by the scan. The scan would be able to generate pseudo three address code with the execution order found in accordance with the rules given above. The code would be:

Line Nr.	Operand 1	L_1	Operator	Operand 2	L_2	Result	L_r(execution order)
1.	b	0	+	c	0	t1	1
2.	e	0	-	f	0	t2	1
3.	t1	1	/	t2	1	t1	2
4.	t1	2	+	d	0	t1	3
5.	r	0	+	p	0	+2	1
6.	q	0	-	m	0	t3	1
7.	t2	1	/	t3	1	t2	2
8.	t1	3	-	t2	2	t1	4
9.	t1	4	+	r	0	t1	5

Thus, the expression would permit the operations on lines 1, 2, 5, and 6 to be executed in parallel first. Following this, the operations on lines 3 and 7 could be executed in parallel, after which the operations on lines 4, 8 and 9 would have to be executed in sequence.

The technique described above depends only on the precedence (hierarchy) of the operators in the language being analyzed.

The treatment of inter-statement parallelism depends on recognition of the regions, and an analysis of the availability of variables. This is merely another way of saying that one cannot logically use the result of a computation before it is available. As an example, consider the following statements:

(1) A = B + C

(2) D = F + 1

(3) E = B + A/D

Statements 1 and 2 could be executed in parallel. However, statement 3 would have to be deferred until the results of the previous operations were available. In this case, the results of the previous operations are available simultaneously with a level value of 1 which would permit initiation of the third statement at the very next instant. If, however, the first statement were the result of the expression used to illustrate intra-expression, the third statement of the region would have to be deferred until its completion.

By noting that the level value of the final result defines the earliest available time for that variable, we are able to extend the techniques used for intra-expression analysis to cover the analysis of regions by affixing in the symbol table the level number achieved by each variable as assignments are made to it. Now, rather than using zero for the level value of variables, we can use the level value stored with the variable in the symbol table, and apply the algorithm noted above.

The techniques noted above can be applied during compilation to explicate all of the natural implied parallelism within a program.

1 J. P. ANDERSON *A computer for direct execution of algorithmic languages.* In AFIPS Conference Proceedings: 1961 Eastern Joint Computer Conference. 184-193 1961

2 T. R. BASHKOW, A. SASSON and A. KRONSFELD *System Design of a FORTRAN Machine.* July 1966

3 H. HELLERMAN *Parallel processing of algebraic expressions.* IEEE Transactions on Electronic Computers Vol. EC-15 No. 1 82-91 February 1966

Definition problems of command control systems

by Frank E. Diaz
System Development Corporation
Lexington, Massachusetts

INTRODUCTION

Policy statements are for the most part expressed in general terms. This generality provides the broad coverage usually desired of such statements and at the same time allows for flexibility in application. The generality of policy statements, moreover, serves still another purpose. Such statements tend to be political in nature where political can be taken to mean "in the context of a people and organizational environment of diverse and at times conflicting backgrounds, purposes, and goals." In such a context useful statements must be of such generality and, at times, ambiguity to allow for agreement among the differing people and organizational elements. Many of the current concepts being applied to command control systems are essentially policy; that is, they exhibit a political character including generality and ambiguity. Application of these policies or concepts to command and control systems requires specific interpretation.

In essence, command control systems are defined by policies and concepts appropriate to a political environment. In applying these policies and concepts to the technical development of command control systems, the generality and ambiguity inherent in our definitions create problems. For purposes of development, we need definitions which are scientific rather than political in nature.

This paper explores some of the problem areas which are brought about by the application of essentially politically-defined concepts to a decision-making and developmental effort which is essentially technical in nature.

Command control systems

There is no established exact definition of "command control systems." The term seems to be one that when used means what one wants it to mean, to paraphrase Humpty Dumpty. An example of this is the high probability of finding in most papers that discuss command control systems a phrase such as "For purposes of this article, command control systems are defined as. . . ."

Nominal definitions

One document (Parsons and Perry, 1965) cites some 10 different nominal definitions of command control systems (pp. 73-77) in addition to their own notion that "a descriptive analysis constitutes the 'definition'" (p. 7). Most of the nominal definitions cited have in common an assertion such as "a command and control system consists of all the equipments, people, procedures, etc., needed to determine and direct courses of action for assigned forces." Such definition may be useful as an encompassing term for general discussion. At an application level it is too encompassing and that is its weakness. It encompasses systems of decidedly different characteristics. It includes not only those systems at superordinate levels primarily for planning and management actions but also those systems at subordinate levels for immediate direction of forces, e.g., aircraft control.

Categorization

Attempts to reduce this very general definition by categorizing (or classifying) have been made. For example, "strategic versus tactical," or "command versus control" or (in another vein) "real time versus nonreal time" have all been used. These classification schemes give the appearance of narrowing the definition since they seem to subdivide the systems into smaller groupings. This help is, for the most part, illusory as the problem is right where it was before; namely, using often ill-defined words to define an ill-defined concept.

In his paper for the Second Congress, Benington distinguished between strategic and tactical command control systems, where "strategic" and "tactical" were used in their classical sense (Benington, 1965). That paper carried the distinction one step further. Benington listed (at least by example) those systems he wished to be considered as "strategic." Thus was added to his definition the most important factor of "pointing to" or providing "observability" to the definition.

In most cases (the above is cited as an example) the technique has been employed on an after-the-fact basis. In other words, existing systems are listed as belonging in one category or another without necessarily explicitly establishing the criteria of classification. The utility of this approach, if used to classify a new or proposed system, depends heavily on familiarity with the systems as categorized for comparison. Nevertheless the notion of "observability" is important in clarifying the definition of command and control systems.

Explicit criteria for classification

As implied above, the addition of explicit criteria to an observable classification scheme would reduce ambiguity of definition and thus improve communication. This in turn suggests that criteria themselves be observable characteristics; examples of this approach follow.

Sensing—first, one may consider the means by which the systems sense the environment whether largely automatic, e.g., radar, or largely manual, e.g., messages prepared by people. Such a dimension seems readily reducible to quantification, e.g., the percentage of inputs of each kind.

Effect—a similar dimension has been previously suggested as "... the degree to which the system is automatically coupled to the environment that it must sense and effect" (Shaw, 1966).

The latter distinction in fact provides two dimensions, the sensing factor discussed and the effect or action dimension. This latter dimension relates to the degree of interposition of people between the computer output and the action point. Again, quantification in terms of percentages might be useful.

System response time—as was noted earlier, the "real time versus nonreal time" classification has at times been used. In general, however, we do not have adequate definition of real time as applied to command control systems. One definition of real time is that in any simulation in which the time dimension was modeled on a ratio of 1:1 to real time the system was tautologically called "real time" (Gill, 1966). This definition rules out most command control systems as real time. Generally, however, when we speak of command control systems as real time we mean a system which must perform some function (or calculation) with sufficient speed to ensure satisfactory performance. A good example is intercept direction via data link in which the vectoring calculations are done in a time scale faster than real time; that is, the relative aircraft positions are projected; i.e., predicted in less time than the aircraft can actually fly the projected distance. In this sense, then, we mean more precisely response time rather than real time. Real time in this sense connotes a time restriction imposed on the system by some environmental condition.

Unfortunately, the term "response time" is often taken to mean the elapsed time between request by a user and receipt of an output (display) from the computer. Usage of the term "response time" in this sense seems to be quite common in discussions of time-sharing systems.

Since the term is quite applicable in both cases, it is proposed that the former be called "system response time-- and the latter, "display response time."

There are at least four possible ways to quantify on the dimension of system response time: (1) The "tightness" of the response time requirements; (2) the number of functions having a response time requirement; (3) the portion of computer time devoted to meeting response time requirements, and (4) the predictability of the response time requirement.

User interaction—another dimension to be called "user on-line versus off-line" would deal with the means by which users interacted with the computer and program. Consideration and quantification of display response time, user input time, ratio of people intervention and nonintervention to user inputs and outputs would be the means of measuring such a dimension.

Further consideration for definition

It is obvious that the kinds of criteria discussed above may be applied equally well to any automated information system. There is nothing about them which inherently enables one to distinguish a command control system from any other system. Of course, if the work of obtaining measurements were done with care, one might be able to distinguish among command control systems.

It must, therefore, be the case that, if system characteristics in the physical sense do not separate one kind of system from another, something transcends the usual physical characteristics which does enable such a distinction. It would seem that the *intent* of the system in this instance is the distinguishing factor; that is, a common control system is intended to *provide* command control. Since, as seems to be the case, it is relatively easy to define the physical dimensions; that is, the system part of the term, then the difficulty must be with the "command control" part of the term. It would be consistent with the basic notion of this paper if "command control" were the *political* part and "system" the *technical* part of the phrase "command control system."

Current situation summarized

Current definitions of command control systems seem to rest on the assumption of inherent difference between command control systems and other systems. For that class of systems to which the label "command control" is applied, no definitions useful to distinguishing at the level of application exist. In this sense, application includes interpretation of policy to a particular case. Little, if any, work has been done to enable distinction among command control systems by measurable criteria.

Current concepts

Continuing with the theme that many problems are brought about by attempting to apply essentially political definitions to technical decisions and/or development, three important concepts are examined.

Standardization

A most important trend today in the development of automated command control systems is the emphasis on standardization. This has some varying interpretations but in general seems to mean standardized computer equipment packages as well as standardization on what some people have called "nonfunctional software." The term "nonfunctional software" generally seems to refer to such things as the compiler systems (implying a standard language), the utility or program production systems, certain kinds of generalized data management capabilities, and in some cases includes an executive program capability of a type which is found most often in operating systems.

Incorporation of data management capabilities seems to be consistent with the major conclusion of a review of seven command control systems which was: "The major trend in computer programming for military command and control systems seems to be the trend toward the development of generalized, user-oriented information systems" (Shaw, 1966).

These are often referred to as data management systems wherein a generalized capability to structure, enter, update, retrieve, and format outputs on some set of information files is provided by means of a user-oriented language which enables the user to manipulate the data at his disposal. Essentially, a language for nonprogrammers is created which allows them to do things which in other systems only programmers could do.

Inclusion of data management and an executive capability begins to stretch, however, the adequacy of the term "nonfunctional." Certainly the "user-oriented information system" is intended to be quite functional. It would seem that, whether intentional or not, we have a definition which, while perhaps adequate in a political sense, is inadequate in conveying the technical connotations.

In distinction to the definition of standardization as given above is the realization that it is no longer appropriate to consider the computer or computer program as just an isolated processing tool. Rather, we have grown in our understanding to the extent that we tend to consider the computer and programs as one element or subsystem of a larger system doing information processing. Among these other elements or subsystems within the broader system context are, for example, the man-machine interface, the communication system, and the information subsystem. More recently, there seems to be a slowly growing recognition that any one given system (within a major command, for example) deals only with a portion of the total information base with which the command must deal.

Standardization as defined tends to be in conflict with the growing recognition of the computer and programs as one subsystem of several making up the automated information system. If one accepts the latter view, then it seems quite clear that some, perhaps substantial, standardization of the other major subsystems is necessary to solve the problem within the computer and program area.

The degree of standardization of the other subsystems which one thinks is required may well depend on the kind of command control system one is talking about. For example, consider the two dimensions of automatic versus manual sensing and user on-line versus off-line discussed above (Explicit criteria for classification).

For a system with high manual and low automatic sensing only minimal standardization in the sensing and communications would be required. On the other hand, if the same system had a big user on-line loading, then substantial standardization to the display subsystem would be required.

Whether one views the need for standardization (to some degree) of the other subsystems in order to achieve standardization in the computer and program area as a serious problem or not is not important to the viewpoint of this paper; what is important is that it be recognized as a concomitant of computer and program standardization. It is, of course, obvious that the given definition of standardization does not necessarily convey either this or perhaps other technical concomitants of standardization. These kinds of failures of definition may result in substantial misunderstanding of the price one must pay in order to achieve, for example, standardization as defined.

It has been suggested that the data management capabilities approach to command control systems development reduces the problem of concomitant standardization to a manageable size. The problem here is that this prescription does not inherently delimit the "kind" of command control system.

The suggestion, however, does seem to be true within limits: (1) the kind of system; (2) the inputs are generally a result of human sensing of the environment; (3) the outputs are generally directed to people for further consideration; and (4) the system as a whole tends to be nonreal time (e.g., few predictable system response times with respect to its environment).

Evolution

Certainly one of the most important concepts today in developing automated information systems for command control is that of evolution. This word has been used to describe the way in which such systems are going to be placed into the operating inventory of the Air Force. Evolution seems to mean beginning with some established system base line and altering portions of that base line through time so as to provide needed improvements to the command control system. Examples of what this seems to mean are that there may be changes in computers but in such a manner that does not require radical alterations to the communication system; the upgrading of the communication system in a way which will be commensurate with the computer system; adding to or modifying the man-machine interface in a way which permits continuity in the staff operation.

Parenthetically, with regard to this concept, two comments should be made: (1) Most evolution to date has been confined within the computer/program subsystem; (2) it has been in the semi-automated air defense environment where some evolution as defined above has taken place. Certainly the original system was developed in a way which was not consistent with evolution as defined above.

Evolution defined in this fashion recognizes, but not explicitly, system boundaries broader than just the computer. On the other hand, acceptance of these broader boundaries raises a number of problems, a most important question being evolution from what; that is, to ask what is a minimal base line from which one begins to evolve, and, for a given base line, what is a reasonable growth expectancy?

Evolution was defined, more or less, as a process to be applied to command control systems. Once again, the operational characteristics of the process have not been made explicit and the cases (systems) to which the concept is to be applied are not well defined. It is not clear whether adequate definition of the systems would be sufficient or whether the dimensions of evolution need also be established.

That some sort of minimal base line capability is required for evolution seems obvious. The minimum requirements for the various subsystems have not been determined. One would suspect the requirements would vary according to the "kind" of system; that is, just where on the various dimensions of measurement a given system might fall. Certainly at an absolute minimum, all of the various subsystems ought to be considered. This is another way of saying more explicit definition of command control system is required if the concept of evolution is even to be discussed in a meaningful way.

An example of the kind of definition and/or consideration required is that of the Air Force Interim Command Control System (AFICCS). The installation of a computer in a command control environment at a major air command raises immediate and significant problems of the relationship of that computer to the existing communication network. When a communication network is largely manual, largely manual means of adapting the computer to that environment are used as the solution. This tends to force update and maintenance of the data base into off-line batch processing mode where this means substantial human intervention between the communication subsystem and the computer program subsystem. An operating command which is attempting to use such a computer in support of command post activities such as monitoring the execution of operations is faced with the serious problem of trying to use a computer operating in an off-line batch processing mode in a environment in which rapid response on-line processing is generally more appropriate.

Adequate consideration of these two subsystems, computer processing and communications, is critically important. The basic design of the computer program system will be radically affected by the assumptions one makes concerning the communications subsystem.

In addition to the definition problem, in considering evolution in any system or for any system, it seems to be very important that we begin to develop some kinds of understanding of what are the concomitants of any given change; that is, if we change some subsystem, what are the impacts within that subsystem and what are the impacts between that and other subsystems?

In order to better understand the dynamics of change, it is necessary to have better definition so that concepts, assumptions, inferences, etc., about change can be tested and modified against the reality of experience.

Organic operation

One of the strongest trends in the recent past has been the requirement of organic software capability for command control systems; that is, the "blue-suit" or "in-house" computer program capability that is required for these kinds of systems. This capability is generally thought of as being composed of some portion each of Air Force and civil service personnel, although no ratios have been established. At the present time the military personnel generally have some operational Air Force Speciality Code with a C or D prefix indicating programming or systems analysis capability.

The tasks to be performed by the in-house capability have not been well defined; consequently, the types and skill levels have not been adequately identified. The requirement for an in-house programmer and analysis capability has resulted in a number of problems in the personnel area. For example, the installation of AFICCS has created personnel problems at each of the major commands possessing the system. Simply stated, there are not enough experienced programmers and analysts to provide adequate timely manning. For the future, moreover, rotation and reassignment policies for Air Force personnel tend to mean that for each major air command at any given time one third of their personnel will be recent arrivals on the scene. Thus, the need to provide for improvement of individual and group performance via training is continual. This need re-emphasizes the need for adequate task and skill level definition.

One suggestion to alleviate this problem is to enable transfer of experienced individuals amongst the major air commands possessing this kind of system. On the other hand, one of the reasons for using operationally experienced personnel with the C or D prefix was to provide the computer programming and design activity (the organic capability) with the operational experience that was felt critical to appropriate maintenance and development of any command control system. Therefore, if it were to become a practice to transfer command control specialists amongst commands in order to retain system continuity and competency, there is an immediate question about degradation to the goal of having an operationally competent, command-knowledgeable person dealing with the command control system problem. It would seem that a practice of transferring command control specialists from place to place would merely alter the fracture point between the command control information processing specialists and the operating user, not necessarily resolve that problem.

It has been argued that this problem is reduced by the use of data management systems. The concept of data management systems, however, raises totally different problems in the personnel area. It will be recalled that data management systems provide a technique by which nonprogrammers can do those things which in other systems are done by programmers. As these kinds of systems come into Air Force use, impact on personnel must be considered. To date nearly all of the work on these systems has been done by people from a population substantially different from that made up by Air Force staff officers. It is not obvious that all of the successes reported in the experimental R&D environment are directly translatable to the operational environment. Assume, however, that such systems are introduced either as the result of additional experimentation or because the expected gains are estimated to outweigh any disadvantages. It would appear that it would be quite important to place emphasis on training the typical staff officer to use such systems. This course of action would provide for greater utility and wider acceptance than emphasis on training for in-house command control system programming, the more usual definition of organic capability.

The foregoing discussion indicates that the behavioral requirements of an organic capability may well be subject to change as a result of new technological development. It would be desirable under these conditions to be able to examine in detail the impact on task requirements likely to be brought about by introduction of data management systems. As has been pointed out, this is not possible; thus, it is difficult to identify the changes to a concept which might be necessary as a result of changes to the environment.

REFERENCES

1 E. BENNETT, J. DEGAN and J. SPIEGEL *Military information systems, the design of computer aided systems for command.* Fredrick A. Praeger New York 1964

2 H. D. BENINGTON *Some needs of strategic command* In Spiegel, J. and Walker, D. E. (Eds.), Information System Sciences; Proceedings of the Second Congress Spartan Books Washington, D.C. pp. 251-257 1965

3 J. FERGUSON *Military electronics system for command and control* IEEE Transactions on Military Electronics Vol. MIL-9, No. 2 pp. 80-87 April 1965

4 S. GILL *Why real time computing is different* The Computer Bulletin Vol. 10, Number 1 pp. 18-22 June 1966

5 C. L. LASTRUCCI *The scientific approach* Schenkman Publishing Co., Inc., Cambridge, Massachusetts 1963

6 A. R. MARSHALL *An approach to evolution* In Spiegel, J., and Walker, D. E. (Eds.) Information System Sciences: Proceedings of the Second Congress Spartan Books Washington, D. C. pp. 485-494 1965

7 R. C. MENDENHALL *Some considerations relative to Blue Suit organic software capability for command control systems* TM-(L)-2137 System Development Corporation November 1964

8 H. M. PARSONS and W. E. PERRY *Concepts for command and control systems* TM-WD-227 System Development Corporation December 1965

9 C. S. SHAW *Computer programming and command and control* TM-2857 System Development Corporation April 1966

Software concerns in advanced information systems

by THOMAS L. CONNORS
The MITRE Corporation
Bedford, Massachusetts

INTRODUCTION

Some preliminary results and opinions are now available based on experiences in building and using the ADAM system, an experimental generalized information processing system. Although not comprehensive, these conclusions point out some areas of concern to developers of software for large generalized systems.

In particular, experiences with ADAM and some of its users indicate that:

Generalized software techniques compare favorably with more conventional methods of building large program systems in that their use reduces the time required to develop and implement application programs. A corresponding cost accrues in memory-space or operating-time efficiency, and unless the data base structures and procedure definitions are optimized with respect to the system and the job this cost may be unacceptable.

For effective use of generalized software on large problems, the user must control the optimization of his data base structure and procedure definition. He must understand enough about how the system works to exercise this control, and the software must allow him to do it. In other words, the software must provide the user simple ways to do simple things and powerful ways to do complex things—a principle we might call "graded simplicity."

Some individual software features can be recommended as feasible and desirable in promoting generalization or helping the user exercise control. Among theses are:

- Extensive capabilities for accepting problem "hard-code," *i.e.*, application programs written in the same language as, and to the same interfaces as, system programs.
- A provision for specifying processes in a procedure-oriented language as opposed to and in addition to a command or query-oriented language.
- A provision for saving individual data structures and procedures on some external medium (say magnetic tape) for later restoration into the same or another version of the system.
- The capability for accepting data from and generating data for other systems.

Generalized information processing

The types of information systems to which these remarks pertain directly are those concerned with the generation and maintenance of data bases; with processing these data to reorganize, query, and generate output from it; and with performing specified calculations on the data. These information systems are generalized in that they isolate the programs from the data through separate data description mechanisms and they include programs which perform common functions not specialized to any single application. A number of such systems classified by terms such as "data management systems" or "generalized information systems" and recently proposed, designed, or implemented have been described (ESD, 1965; IBM, 1965).

The ADAM system (Connors, 1966; ADAM Project, 1966) is an experimental large-scale generalized information system, designed as a laboratory tool to perform the functions described above (along with many others such as display handling and remote access). ADAM has been applied on an experimental basis to implement portions of a number of different applications, including:

- a tactical airborne radar system characterized by high volume, real-time inputs;
- a prescheduling tool for satellite missions which emphasizes man-machine interaction through display consoles;
- a personnel assignment problem with a specialized algorithm for matching men to jobs and
- an inventory requirements analysis problem with an extremely large data base.

The inventory problem was by far the largest and longest. It was conducted in conjunction with the Air Force Logistics Command (AFLC), and involved generating files, performing calculations, and producing reports which duplicated the AFLC Consumption Item

Requirements Computation (D041), using subsets of their 170-million character data base.

These applications provide the foundation for the opinions presented. No formal quantitative evaluation is available at this time; however, a report of the AFLC experiment is forthcoming (Char and Foreman, 1966) and an evaluation of the ADAM system is planned.

Generalized software effectiveness

The very diversity of application areas to which ADAM has been applied supports the viewpoint that generalized capabilities *can* be provided. An experience with one part of the AFLC experiment demonstrates the effectiveness of generalized techniques as compared with conventional programming. The segment in question requires 15,000 Autocoder instructions in the original operational system. The corresponding nine major procedures comprise approximately 55 statements in the ADAM file manipulation language. No measure of the time and effort required to program the original is available, but it would be very surprising if it were less than the five man-months it took to replicate it in ADAM—complete with a problem re-analysis, since the problem-specification documents were not available to us.

The file manipulation statements were not simple—the process itself was quite complex. In some cases, it took a long time to write an appropriate query to retrieve from the data. But a query *could* be written. A user reports, "... only simple queries were answered in a time span representative of on-line operation but even those that took longer, several days to a week, were faster than any other method that could have been used to get the data, if it could be obtained *at all* by other means" (emphasis supplied).

On the other hand, users react to the complexity of the system and the ease with which it is possible to cause gross inefficiencies in computer use with apparently simple inputs. More about that in the following section.

User involvement

Perhaps the most important conclusion reached through the ADAM experience is that the user of a generalized system must become involved in it—to the degree that he understands it well enough to specify, in the necessary detail, the processes he initiates. The requirements then are three: the user must be aware of the implications of how his data is structured; he must understand and control the optimization of his procedures; and the system must allow control of optimization.

For a trivial but real example, in one application the user actually needed to perform the supposedly simple process described in almost every elementary computer programming text: multiply price by quantity to get total cost. Due to the way his data were organized when received, price and quantity were in different files, with corresponding entries linked by a symbolic identifier. The ADAM file generation statements generated the file in the order of the original data, for simplicity. The single statement which accessed the data, performed the multiplication, and stored the result was simple to write, even on-line. The resulting processing involved serially accessing the file which contained price, and, for each entry, directly accessing the entry in the other file which contained quantity. In the ADAM system, direct access is slower. Although the simplicity of the query suggested a fast response, the processing actually went on interminably until it was prematurely stopped after one hour's running time. Later analysis estimated that it would take two hours to complete.

By moving price and quantity to the same file, the user reduced the running time required by a factor of eight. He preferred a faster calculation at the cost of a somewhat more cumbersome file generation. But only *he* could tell if changing the structure in this way would meet his requirements for *other* uses of price and quantity. Different processes he specified could turn out to be much slower because price was no longer in its original file. Even if he decided to keep price in both files, he would not solve his optimizing problem. Two copies in two files implies a more complicated and time-consuming update process when prices change.

This experience and a number of similar experiences serve as convincing evidence that a system can help a user optimize by giving him facilities to restructure his data, but it cannot optimize for him. And he cannot optimize unless he knows how the system will perform the processes he specifies.

Consider another example. In the ADAM system, the statement which begins

FOR CITY BOSTON ...

causes a direct access to the BOSTON entry in the CITY file. The apparently equivalent statement:

FOR CITY . IF NAME EQ BOSTON . . .

causes a serial search through entries in the CITY file until BOSTON is found. The user must know and appreciate the difference in order to query the system effectively.

One way to avoid requiring a user to understand his software is to devise a system with only one way to do an operation—a form of uniform simplicity. Some systems, for example, provide direct access through an index to all data items. This forces simplicity for the user, but costs him effectiveness. For example, a fully indexed data structure can provide direct access to every data item. If a system always requires this, it may spend a prohibitive amount of computer time making indices for data items which a user always accesses serially.

Uniform simplicity hides a pitfall. The easy availability of uniformly simple ways to initiate processes deceives a user of information processing systems when it convinces him that his problem automatically becomes simple.

Graded simplicity is required. For truly simple tasks, simple specifications are sufficient and should be available. For tasks with more complex elements, ways to make complex but efficient specifications are essential. Our experience shows most real jobs contain both kinds of tasks.

Specific software features

The following recommendations for specific software features are not intended to be comprehensive, but represent some suggestions derived from ADAM experience.

Procedure language—In the ADAM file handling language, the actual operations performed as the result of a message are implicit in query-type statements—for example, the user cannot explicitly control iterations within several files. On this account, he loses an amount of control over optimization of complex processes. A language that allows him to describe processing procedures (in addition to commands and simple queries) is required.

Hard-code interface—In the final analysis, a system is truly generalized only if it provides for the inclusion of specialized user-coded programs treated similarly or equivalently to system programs. The mechanism must provide not only for problem programs to be run as separate tasks of their own, but for them to be run in place of, or along with, system functions.

For example, ADAM provides for a user to specify conversion routines to be applied automatically to every input and output of a given item of data. Some system routines are available, but provision is made for a user to include his own and to prespecify where they are to be applied. Similarly, in the ADAM report- and output-formatting definition mechanism, an operation is available (in addition to those like "space page," "print value," etc.) which specifies that a routine is to be executed at this point in the output formatting process. These provisions were liberally used in certain applications; more of them would have been desirable, for example a provision for user hard-code at message input time.

An effective capability for accepting hard-code requires a well-defined interface with system routines which allows access not only to the data base, but to volatile data such as lines of report just before they are output and parts of messages just after they are input. And in addition to explicit calls on these routines, the provision must exist to have them called implicitly at predefined points during the operation of a system function.

Saving and restoring—Many data base systems, including ADAM, treat the system programs and data as part of the data base and go through a more or less elaborate process of allocation and transformation in absorbing new data or programs. During ADAM use, a requirement that became obvious was to allow for saving individual data structures and routines outside the system (ADAM does it by writing them on a magnetic tape) so that parts of the system may be changed or replaced without requiring regeneration of all the rest; instead, the saved structures may be restored to the same system or to a different version.

From this saving and restoring capability, three immediate benefits accrue:

- The data and programs for a single application may be transferred from a current version of a system to a different, presumably improved version thereby separating the maintenance of system from that of application;
- two applications which may not be able to exist together in the system (perhaps together they exceed the system capacity) can each run with exactly the same system software; and
- relatively infrequently used system data and programs need not be kept as a permanent part of the system.

Accept and generate outside data—Experience shows that data more frequently come from and go to other computer programs or systems; less frequently are prepared especially for a single system. At one time the ADAM file generation process made the mistake of expecting all variable length input fields to be terminated by a (user-specified) character. Data do not necessarily come in that way, and in several applications a separate preprocessing step was required to insert the terminators. A generalized system must expect to accept and generate data not specifically prepared for it.

CONCLUSION

Generalized software techniques for advanced information processing systems are with us at least experimentally, and point the way to making large problems somewhat more tractable. Nevertheless, there is no magic to them and it is clear that the user of a system cannot be decoupled enough from the way his problem is solved to avoid understanding and controlling effective optimization. As results of experiences with experimental software techniques become available, they will point the way to acceptable balances between simplicity and effectiveness.

REFERENCES

1 B. F. CHAR and A. FOREMAN *Final report—joint AFLC/ESD/MITRE advanced data management (ADAM) experiment* MTR-285 The MITRE Corporation, Bedford, Mass. 1966

2 T. L. CONNORS *ADAM, a generalized data management system* MTP-29 The MITRE Corporation, Bedford, Mass. 1966

3 Electronics Systems Division and Computer Associates Incorporated *Advanced programming developments: a survey* ESD-TR-65-171 L. G. Hanscom Field, Bedford, Mass. 1965

4 International Business Machines Corporation *Generalized information system application description* E20-0179-0 New York 1965

5 ADAM Project *A user's guide to the ADAM system* MTR-268 The MITRE Corporation, Bedford, Mass. 1966

Some factors in planning for future military data automation systems*

by G. M. NORTHROP
The RAND Corporation
Santa Monica, California

INTRODUCTION

The year 1966 may well go down in the annals of the military data automation community as the Year of the Master Plans. Comforting as this thought may be—for it is helpful to have a clear blueprint of the tasks the future holds—it is not clear that 1966 will also be known as the Year of the Reconciliation of the Master Plans. And until such coordination of master plans is effected within the military services, throughout the Department of Defense, and, yes, possibly across the entire structure of the U.S. Government, it appears unlikely that the master plans of 1966 can hope to serve as blueprints for more than a year or two of the future. Coordination of master plans is not easily effected, and because of the potential controversy involved, the subject is not to be lightly broached. But it has been demonstrated many times that once government spending for a single identifiable function or item or utility becomes a substantial percentage of the budget, more cooperation and less duplication of effort is demanded on the part of all agencies involved. As a typical example, communications services—first within the DOD and now within the government *in toto*—are coming under increasingly greater scrutiny to insure economy and efficiency of service and operation. Government interest in close supervision of all of its data automation facilities and services is either entering or will soon enter this same phase. The Bureau of the Budget's yearly publication, *Inventory of Automatic Data Processing Equipment in the Federal Government,* is a first step in that direction.

Each organization finds a master plan for the future a necessary and useful tool for internal planning purposes, but coordination of the master plans of several different organizations—each plan separately conceived —may be a painful and trying experience for all concerned. However, successful coordination of planning efforts can provide much greater stability and effectiveness of the composite of efforts. This paper will not proffer suggestions that will necessarily ease the burden of data automation master plan coordination, but, because it is intended not as a presentation, but as a vehicle to stimulate discussion, the paper will attempt to raise some of the broad issues that at all levels face the designers of master plans, or plans that coordinate master plans. The treatment of these issues in the paper will be kept as objective as possible; it will be left to the discussants at the Third Congress to take sides and do battle.

To achieve the goal of delineating some future trends of military data processing and to indicate some of the potential effects that masted planning and the coordination of master plans might have on these trends, in the remainder of this paper I will first undertake to categorize the functional uses of data processing by the military; next, the relative scope of data automation activities by the services will be outlined; third, a few of the dichotomies (or paradoxes?) facing master planners will be discussed; then some of the potential bottlenecks that may slow the desired progress of data automation improvement will be listed; and finally, some summarized suggestions of future planning efforts will be presented.

Major military uses of data automation

In the military, as elsewhere, the user of data automation sees its greatest utility and future as an aid to him in *his* work. Understandably, the user frequently thinks that *he* is making the very best use of his machine and that improvements and advances stem foremost from *his* efforts. And the user's data processing equipment often is sufficiently flexible to allow the user to

*Any views expressed in this paper are those of the author. They should not be interpreted as reflecting the views of The RAND Corporation or the official opinion or policy of any of its government or private research sponsors.

branch into activities similar to those of other organizations or agencies; this has a tendency to engender competition and rivalry. Thus, in categorizing the uses to which the military applies data automation, it is recognized that some overlap will exist and that these overlaps may fall in areas where today competition and rivalry may exist. For the purposes of this paper, five major military uses of data automation are considered, although it is quite likely that others could be cited:

- Research and planning systems
- Management systems
- Support systems
- Command systems
- Tactical systems*

Research and planning data automation systems include those at military laboratories and, typically, the system recently proposed for use by the Air Staff for preparation of plans and the formulation of staff positions. Ultimately, the research systems at the laboratories may be widely internetted; a plans system for the Air Staff might have, in addition to an information display center, remote connections to the deputy chiefs of staff and appropriate directorates, as well as connections that make available information from the data bases of other systems.

Management systems are taken to be those that recurringly perform essentially the same tasks, whether in peace or crisis, that are necessary for the normal functioning of a large organization, viz., finance and accounting, personnel records, logistics, etc. Generally, elements of branches of such systems are to be found at every military installation.

Support systems are typified by those in use by the Air Weather Service, the photocomposing system for document publication soon to be installed at Wright-Patterson Air Force Base, etc. Support systems characteristically have one or a very few large data processing facilities and a very large number of users of their output.

Command systems have sometimes been described as "capping" systems, for they are apt to make use of the outputs of planning systems, management systems, and support systems. In the future, as tactical computer systems become more prominent, upper echelon command systems will likely make real-time use of summarized data from them. In a sense command systems either are, or should be, capable of mustering, allocating, and directing resources to meet military commitments—both potential and actual—throughout all levels of crisis and war. Of course, this broad charter places the command system in the position of overlapping the functional areas of planning, management, and (sometimes) support. And, with the present worldwide politico-military environment directed toward controlled escalatory warfare, it is inevitable that even upper echelon command systems will occasionally encroach on some control functions of tactical systems.

*Special purposes computers, such as those used in missiles, are not included.

It is, of course, impossible to avoid functional overlap in establishing a category of tactical data automation systems. It may be desirable to categorize tactical systems as those in use by field forces (whether in the field or in garrison training), or to say that tactical systems are mobile, ruggedized, and, hopefully, small and lightweight. But probably the best differentiation possible is to say that tactical systems are those used by field forces, and not already covered by one of the other four categories. Thus, elements of SAGE and BUIC, shipborne and airborne systems, as well as units that may be carried into combat, such as the Field Artillery Digital Automatic Computer (FADAC), are included. In general, data processors that are integral parts of weapon systems and are essentially special purpose (e.g., inertial navigation computers) are not included here.

Is it necessary to have this categorization of functional uses of data processing by the military? The answer is "Yes," and for at least two reasons. The first reason is that the attempt at categorization points up the extreme difficulty of looking in a meaningful fashion at just one part of the military data automation picture. With rare exceptions it is simply not possible (or, at least, not effective or efficient) to consider one category of military application of data automation (e.g., command systems) without being cognizant of, and coordinating with, activities in most of the other areas. Certainly this is true for many tactical applications, for often they either are now, or in the future will be, primarily duplications of efforts that are already handled by data processing systems that are not capable of field deployment, e.g., the possible future use of data processors in backup airborne command posts.

A second reason for some form of categorization stems from the needs of higher echelon organizations to delineate similar characteristics for comparison of the efficient use of data processing systems. For example, the DOD must at some point view the use of data automation by the three services in some comparative fashion, asking the question, "Are systems of comparable capability being utilized with equal effectiveness?" The measures of utilization may be difficult to establish, but, once established, should the answer in any instance be negative, remedial action would be needed. In the same manner, an appropriate organization looking broadly across all U.S. Government uses of data automation for the purposes of coordination leading to efficient employment of data automation throughout all government agencies must also seek some means of categorization for comparison and, hopefully coordination.

The relative scope of military uses of data automation

What is the relative effort that is invested in providing military data automation? Some statistics on this question may help to provide some insight into future trends of military data processing. The estimated total number of installed computers operating under the aegis of the U.S. Government at the end of FY 66 was about 2500 (BOB 1965). This represents about 9 per cent of the total number of computer installations in the country at that time. Extrapolating available data to the present, the DOD makes use of about 1900 computers, of which about 53 per cent are under control of the Air Force, with the Army and Navy controlling 22 per cent and 21 per cent, respectively.* The direct cost of data automation (with support) to the U.S. Government has been over $1 billion for each of the last three fiscal years. Thus, about one per cent of the national budget is directly attributable to government data automation. The DOD accounts for about two-thirds of the U.S. Government's installed computers and about 61 per cent of its computer costs. Some of these relationships are illustrated in Figure 1, which shows the recent growth of installed and on-order computers in the U.S., installed computers in foreign countries, and installed computers under the control of the U.S. Government, the Air Force, Army, and Navy.

It is noteworthy that while this nation has been installing *additional* computers at the rate of 7,250 per year for the past two years, the U.S. Government rate has been only about 300 additional computers per year, with the Air Force accounting for slightly more than one-third of the new additions each year and the Army and Navy each accounting for about one-sixth or less of the total government rate. This should probably not be surprising, for the military services had priority in filling their needs during the early years of data automation growth, and while computer replacements continue apace in military installations, the rate of additional computer acquisition is not in keeping with that of the nation.

It has often been said recently that the impact of

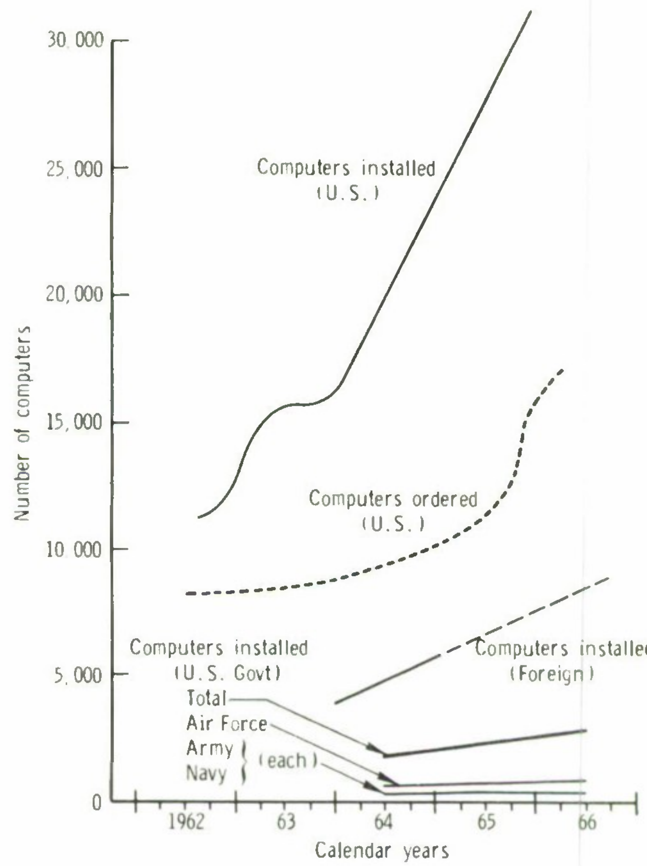


Figure 1—Growth of data automation

government—and especially military—spending on the computer industry is continually dwindling. Figure 1 might be construed to substantiate that claim, for at end of FY 66 the U.S. Government will operate less than 7 per cent of the nation's installed computers; two years previously it operated over 9 per cent. Should present trends continue for the next two years, the U.S. Government's share will drop to about 5.5 per cent.

In broad geographical terms, where do the military services make use of data automation? Table I shows the number of major geographical locations of military data automation locations (BOB 1965).

Table I
Distribution of major locations of U.S. military data processing

Military Service	Major U.S. Urban Areas	Foreign Countries	Total
Air Force	126	17	143
Army	75	4	79
Navy	46	4	50

Within the U.S., the regions with the highest density of military computers are Washington, D.C., San Francisco, San Antonio, Philadelphia, Norfolk, Virginia, and Dayton, Ohio. It follows that each of these regions would be a prime candidate for the on-line service that may someday be provided by a data automation "utility"

*The data presented here include some tactical computer installations such as SAGE and BUIC sites. They do not include airborne or shipborne computers or field mobile units such as FADAC.

system, i.e., a system that might supply on-line computing power from a central facility to all military customers within a specified geographical area. As an aid to visualizing the potential scope of military data automation utility systems, Table II indicates the degree of gross collocation of military computers within the U.S. It is stressed that collocation is taken here to mean computer installations within a general urban area, i.e., within at least a few tens of miles of each other.

Table II
Distribution of collocated military data processing in the U.S.

Military Service Combinations	Major U.S. Urban Areas
Air Force-Army-Navy	9
Air Force-Army (only)	21
Air Force-Navy (only)	11
Army-Navy (only)	4

Table II indicates that possibly one-third of the Air Force's data automation U.S. locations might be able to share a local area military data automation utility system with one or more of the other services. Should it be possible to serve a large geographical area with the utility, obviously even more installations could be supported. Of course, data automation utility systems present certain problems and drawbacks when applied to military tasks, but indications are that commercial applications of these systems will become more prolific in the future and it is likely that the military will find it necessary to evaluate at least the potential use of utility systems, both for individual service applications and for joint service use. More will be said about this later.

The discussion on scope of military data automation thus far has centered on fixed computer installations, thus excluding most tactical applications. While Figure 1 makes evident that the growth of fixed military computer installations is moving ahead at a relatively moderate pace, it gives no indication of the future growth of the use of data automation for tactical purposes.*

While it is doubtless true that no accurate estimates can be made of the degree to which military data automation may ultimately be applied in tactical units, it is perhaps of some value to make a gross reckoning of the number of identifiable military units to which data automation may be applied in the future. An estimate of this kind is given in Table III, which also indicates a range in the number of computers that might actually be involved. To avoid security difficulties, U.S. force size has been based on an unclassified source. (ISS 64)

Table III
Potential future use of tactical military computers

Military Service or Branch	Unit	No. of Units	Tactical Computers per Unit	Total
Air Force	A/C Sqdn	200-250	1-2	200-500
Army	Division	16-19	20*	320-380
Navy	Ship/Sub	400-500	1-2	400-900
	A/C Sqdn	60-70	1-2	60-140
Marine Corps	Division	3-4	20	60-80
	A/C Sqdn	15-20	1-2	15-40
Grand Total				1055-2040

*Already acquired FADAC computers not included.

How reasonable are the totals shown in Table III? That question is essentially impossible to answer and the totals can be considered only as opinion, but some of the assumptions underlying the totals and some of the implications of them can be discussed. The use of military data processing in tactical environments has long been stated as a military requirement by the services. However, only in recent years has the state of technology permitted the production of data processors sufficiently small and reliable to be seriously considered for field use. Acceptable peripheral equipment to work with the central processing unit and appropriate software are today probably the pacing items that delay system applications. Of course, not all so-called tactical applications represent the worst of all possible operational environments; for example, many shipboard aplications and ground applications associated with aircraft squadrons may present much more benign environments than can be expected for equipment taken by Army and Marine units into active ground combat zones. Implicit in some of these comments is the assumption that it will be desirable (and, hence, required) to make identifiable military fighting units down to at least the division, ship, and squadron level essentially independent of higher echelons for at least a major portion of their tactical data automation capability. It is possible that the successful demonstration of a field service tactical data automation utility system could reverse this assumption and, rather than organic computers, the computing power of large central processors would be used remotely by lower echelon units. The mobile communications netting task, while not technically infeasible, would be formidable for a tactical utility system. In general, the need to maintain autonomous capability in the operation of field units will likely keep the pressure high for separate computers.

In terms of the earlier categorization of the use of military data processing, it would appear that tactical uses could easily duplicate all of the other categories

*The computers considered for tactical purposes are assumed to be of the micro-miniaturized general purpose variety, costing possibly $30,000 to $150,000 today. Although no one of these computers would satisfy all tactical needs, it is assumed that a computer of this type could be of great use in many tactical applications, if appropriate peripheral equipment and software could be made available.

listed above. (Should enough computer power be available in the field, it is to be anticipated that some of it would be relegated to certain operations research functions.) Not only that, but it is likely that in some instances command and control, planning, management, and support functions will all be carried out in the same data automation installation, e.g., aboard ship, or organic to the tasks of the aircraft squadron.

To some degree, the data processor organic to the lower echelon tactical unit may itself be the central element of a small remote I/O tactical system, for the success of application of data automation to tactical units may well hinge on the ability to acquire information remotely in digital form, process the information, and send processed results or further queries to lower or higher echelon units. The successful completion of recent tests employing a digital message entry and acknowledgement device at lower echelons, and buffered and printed output at higher echelons, has indicated the advantages and feasibility of some elements of a small tactical system with remote input/output.* It remains to be demonstrated that a central processor unit and appropriate software can be added to complete this application of data automation to one part of a tactical air control system. If a complete system can be made reliable and successfully demonstrated in the field, similar applications in other areas and by other services will likely depend more on the generation of software than on equipment.

Pitfalls and dichotomies in planning for military data automation systems

The road to be followed by planners of future military data automation systems promises to be a rocky one. The requirements for future military data automation systems will likely continue to include, to some degree, the long-standing ones that sometimes seem almost paradoxical: efficient and economical, evolutionary, flexible, and survivable. These requirements, coupled with the great geographical coverage—often worldwide —required of many systems, pose systems planning tasks that are indeed formidable. The recently acquired third-generation hardware capabilities have created further problems for the planner by affording various alternatives that may be employed to satisfy military systems requirements; one pair of alternatives was implicitly mentioned above in the discussion of tactical data automation and is discussed in more detail below.

Utility systems: Yes, no, or how much?

Probably the biggest single question facing the planner of future systems concerns the choice of using a utility system with a large central data processor and remote input/output or of using several smaller (but not necessarily "small") data processing installations at each of the major I/O locations. At the moment there are no *good* guidelines available to help the planner in making this choice, for even special purpose utility systems, such as the Keydata Corporation system in the Boston area, are just beginning to provide operational information (C&A, January 1966). A corollary question facing the planner concerns the size of geographical coverage to be provided by a utility system; this is closely associated with the further question: should the utility system provide capability to more than one military service within a given geographical area? This question has been broached earlier in this paper, when it was pointed out that there are certain areas in the U.S. where a heavy concentration of military data automation is to be found. Under selected circumstances—notably, those associated with universities and colleges—early models of utility systems appear to be working well and the users seem to be more satisfied than not. In installations at institutions of higher learning, the utility system is often used partially in a time-shared mode by several instructors with remote I/O stations in the classrooms, partially by research teams (sometimes to monitor physical experiments in real time), partially for graduate thesis work, and partially by the institution for accounting, payroll, etc. As single utility systems come to provide more and better services than many small machines, perhaps one of the most obvious locations for a military prototype system would be in the Pentagon. A Pentagon prototype utility system that would serve all the services would call for a new level of cooperation among the users and might serve as a useful guide for military utility systems elsewhere. Of course, a Pentagon utility system may be seriously constrained should it be necessary to provide for secure transfer of information throughout the building. For military applications, the task of providing high security for the transfer of high data rate digital information is one demanding early, economical solution. The technology to handle this task is in hand; but, as yet, it is not as economical as it must become.

In at least one case, preliminary results concerning a utility system indicate that it *is* efficient for the user in providing low cost real-time service comparable to that of an on-site installation (C&A, January 1966). Will utility systems afford the military user the flexibility needed? The answer to this question is closely intertwined with military missions, especially during crises and/or wartime. In commercial or educational

*Tests consisted of sending and acknowledging forward air controller messages and were conducted at the Tactical Air Warfare Center, Eglin Air Force Base, Florida.

utility installations, it is conceivable that a crisis affecting one or several users of the system would have little effect on many of the other users, or under certain circumstances various users could reduce their activities. Almost the converse is true for the military users. A crisis that affects one military service is apt to have a similar effect on other military users sharing the system, thus bringing demands by all users to a peak. Difficult though it may be to demonstrate by means of cost-effectiveness, the military need for flexibility in time of stress may surmount what appear to be the obvious advantages to be gained in efficiency of operation that may seem to be afforded by use of utility systems. Of course, the utility system might be designed to accommodate any expected peak loads. However, often it is possible to generate meaningful system design tests only by actual crisis operation, where a design failure may be found too late and prove to be catastrophic.

Another potential application for the utility system concept is at the base level. Today almost all major military bases are apt to have two or more separate computer installations. As more and more functional military areas—maintenance, medicine, communications, etc.—turn to data automation for improved operational performance, the number of computers at each base may greatly increase—unless the additional capability can be provided by a base data automation utility system. Of course, here again loom the twin specters of system flexibility and survivability. As the nation's military policy continues to swing toward developing a capability for mobility and responsiveness of more and more units in order to make credible the concept of controlled, escalatory warfare, even should that mean prolonged, controlled nuclear conflict, then it may develop that the computers used for many functions at base level will be required to be mobile and many functions normally performed at the base would move on short notice to remote locations during time of crisis or conflict.

Thus far, the comments on utility systems have centered primarily on systems that for the most part might be used in relatively benign environments, e.g., in the ZI. As indicated earlier, there appears to be a place for utility systems in the hands of tactical forces in the field and these systems would likely apply to the complete spectrum of possible uses. Aboard ships, warning and control aircraft, and command post aircraft the local environment seems to favor the local utility concept to a considerable degree, primarily because of the ease with which the communications and security problems can be handled. For ground forces in the field, communications, security, and (perhaps more important) the vulnerability of one or a few central data processing locations tends to auger against the use of a utility system for even moderately large geographic areas. Within Army tactical operation centers and like points of management and control, the utility concept will likely be limited to a large central data processor and I/O stations in the immediate vicinity. In general military operations in the field—by individual unit and often by function—are apt to demand organic data processing equipment at many echelons to achieve the flexibility, security, and survivability needed.

Development of future systems

The statements above can be construed to imply that much of the important structure of future military data automation systems is fuzzy at best. Also, one of the most important questions for all systems would seem to be: "Who gets the central processing units?" Some insight to this part of planning might be provided by well-coordinated tri-service experimental tests. It is the writer's belief that the test should be carried out in the field (which may mean aboard ship, at a specified headquarters, a military base, in a test aircraft, etc.) and in conjunction with the potential users of the systems. In some cases it may be desirable to establish parallel development efforts, one at a field installation and one at a research or development center, with coordination insured. Coordination is also required among the services so that hardware and software advances by any service can be exploited as soon as possible wherever they are applicable.

Listed below are some features that might contribute to a workable, productive development and test program. Doubtless, other items could be added.

- *Top-level support, direction, and guidance.* A tri-service program needs support from a cognizant DOD organization, from headquarters staff level in each service, and from the commander of the military organization providing facilities for the tests performed by user and development personnel.
- *A user group to support the test.* Test success depends greatly on experienced, qualified personnel, and for data automation in military tasks, a *user* organization is essential to test the utility of the system.
- *Expert help from contractors.* To insure optimal testing, hardware and software know-how provided by contractual support should be incorporated, but with control in the hands of the user, with appropriate guidance and assistance from development agencies.

- *Off-the-shelf equipment.* Third-generation data automation equipment should be used when possible to provide operational experience that could lead to a better basis for determining more meaningful military requirements. It also would contribute to useful feedback and cross-fertilization among the services, the development agencies, and the planning structure in the DOD.
- *Test to serve at least one need thoroughly.* A test need not attempt to serve all identifiable needs, but it should be directed toward the adequate solution of at least one outstanding task.
- *Data automation experiment as an aid, not an end.* Each test and each development effort should reflect the fact that data automation should be applied to aiding man in his military duties, rather than attempting to replace him.
- *Means for the exchange of experience and information.* A series of on-site field tests and experiments involving user personnel can be most useful if all parties concerned are continuously kept informed of all parts of the test program. For example, an airborne data processing system applied to strategic command and control could have application also for ground and shipboard command and control systems and, more broadly, for other management and support systems where mobility and weight are of prime consideration. Furthermore, software techniques useful in command and control systems may find at least partial application in medical systems, and vice versa. Channels for publishing interim results and meetings to exchange information and experience should be frequent.

The points above represent, of course, but a beginning in the outline of a comprehensive field test and development program. Central to the entire theme, obviously, is the need for top-level coordination, whether the program is undertaken by a single service and its own agencies and commands, or whether it is handled on a DOD-wide basis. Properly coordinated at either level, the results of such a program would be of inestimable benefit to the planners of future systems.

Potential pitfalls and bottlenecks in planning for future systems

While many of the points to be made below have been brought out elsewhere in this paper, a few of the more important pitfalls and bottlenecks in master planning are explicitly noted here for convenience. Foremost among these is the possibility of lack of upper echelon guidance and coordination of master plans developed by lower levels of command. Planners at lower levels may find many months of effort turned aside by decisions made at higher levels concerning, for example, the manner in which utility systems may be used in the future. As another example, decisions concerning the use of data processing aboard most of the Navy's first-line ships or in the majority of the Army's ground units should be made in close conjunction between using commands and higher headquarters and coordinated throughout the DOD, as has been reiterated throughout this paper.

Lack of field experimentation may prove to be a bottleneck in developing meaningful master plans. At the moment this comment applies equally to the question of the general military applicability of the utility system, as well as to the application of data automation to tactical functions.

In spite of the fact that communications technical feasibility is not in question, it may develop that certain communications systems will for a time be inadequate for the many data automation systems that may be scheduled for their use in master plans. Often the master planners at lower echelons look upon a communication system such as might be found at base level or at higher level, such as AUTODIN, as the expected means for transmission of information. As was the case with the early users of AUTODIN, the data automation master planner may find, to his surprise and shock, that not only is much of the communication system's capacity being used by others but also that the system itself exhibits certain characteristics that seriously curtail effective data rate.

It may develop in newly applied data automation systems that the operational unit is inadequately organized and/or staffed to bring the system into full operational capability in the expected period of time. A pitfall of this kind can be expected as data automation is more widely applied to new functional areas such as those encompassed by the surgeon general, the inspector general, the judge advocate, etc. And it is likely to be even more prevalent in the application of data automation in the tactical area. Limited-scope field tests and experiments might tend to ease this potential bottleneck.

Two well-known workhorses conclude this list of pitfalls: One is standardization of data processing languages, and little more will be said except that it is needed, for it is getting attention today and it will doubtless require additional attention throughout the foreseeable future. The second well-known bottleneck is training: Data automation training continues to be needed in all functional areas of application and at all levels of command. All master plans should include an adequate treatment of an accompanying training program. Fortunately, advances in computer software

are leading to programming languages that are increasingly easier to master. And there is always the hope that new data automation systems will come equipped with a programmed learning feature, so that the operation, programming, and maintenance of the equipment can be learned in programmed fashion from the machine itself. A feature such as this would be of great benefit, if it could be made a part of tactical data automation systems, where the turnover of personnel in the unit may be high.

Suggestions for future planning

The central themes of this paper are few and simple. In summary form, reworded as suggestions for future planning efforts, they might be as follows:

- In all master planning efforts, insist on receiving guidance and coordination from above and insure that it is given to echelons below.
- Where experience for master planning is lacking, outline on-site, user-performed tests and experiments to provide the experience and insight required. In field tests, make use of existing hardware (and software, where possible) and contractor support, but keep the user in control.
- In planning for military data automation systems, do not lose sight of the need to keep military systems flexible and survivable; in the future this may impose requirements of redundancy and mobility on systems that today are considered to be of the fixed installation type.
- Keep in clear view the pitfalls and bottlenecks that may plague the data automation systems proposed in master plans and recognize that some of the difficulties may be alleviated by adequate organization and staffing of the operational units, by adequate training at all levels of command, and by communication systems that are compatible with the data processing system and adequate to serve all demands.

REFERENCES

1 *Inventory of Automatic Data Processing Equipment in the Federal Government* Bureau of the Budget Executive Office of the President Pashington, D.C. June 1965

2 *The Military Balance*: 1964-65 The Institute of Strategic Studies London November 1964

3 C&A Market Report, Computers and Automation 9 January 1966

Subject Index