

UNCLASSIFIED

AD NUMBER
AD464899
NEW LIMITATION CHANGE
TO Approved for public release, distribution unlimited
FROM Distribution authorized to DoD only; Administrative/Operational Use; 15 JAN 1965. Other requests shall be referred to Office of Naval Research, Arlington, VA 22203-1995.
AUTHORITY
ONR ltr, 4 May 1977

THIS PAGE IS UNCLASSIFIED

Final Report of the First Phase of the

ADVANCED NAVAL TACTICAL COMMAND
AND CONTROL STUDY (U)

VOLUME IV - METHODOLOGY

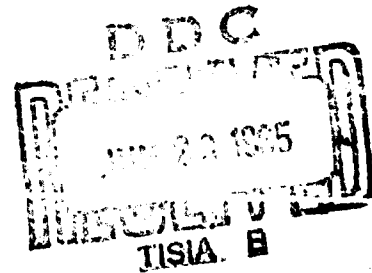
15 January 1965

Prepared for
Advanced Warfare Systems Division
Naval Analysis Group
Office of Naval Research

under
Contract Nonr-4388(00)

by
Informatics Inc.

U.S. military agencies may obtain copies of this report directly from DDC. Other qualified DDC users shall request through the Advanced Warfare Systems Division of the Office of Naval Research.



AD-464 899

Reproduction in whole or in part is permitted for any purpose of the United States Government.

Best Available Copy

ABSTRACT

Reviews a study to set forth improved methods and procedures for Navy planners to make decisions in development, design, and implementation of improvements to tactical command and control systems. This volume reports on the first year's study to analyze planning tools for system design and evaluation, and interprets their use in planning tactical command and control systems. The report discusses in detail planning for system management and the procedures to be followed in system planning. It discusses the role of cost effectiveness and how effectiveness can be measured. Methodology for system planners is treated, covering the role of simulation in system design, development, checkout, and test and evaluation. Simulation languages, mathematical modelling and queuing models are discussed. A new and improved method of determining figures of merit for digital computers is given. The volume recommends a management system for naval tactical command and control systems and concludes with a bibliography of management methodology and planning methodology.

GENERAL PREFACE TO ALL VOLUMES OF THE FINAL REPORT OF THE FIRST PHASE OF ANTACCS

The first phase of the Advanced Naval Tactical Command and Control Study (ANTACCS) is complete. A final report of the first year's work is presented in five volumes of which this is Volume I. These volumes are:

- Volume I Summary Report; a review of the total study to date, summarizing study findings and giving principal conclusions and recommendations. Provides an introduction to all other volumes.
- Volume II General System Requirements; develops for system planners, details of command and control needed to meet the anticipated threat with the anticipated Naval force posture of the 1970-1980 period.
- Volume III Integration; uses system concepts developed in Volume II to give a planning example by analyzing command and control needs of a Task Force Commander, showing how technology (Volume V) and methodology (Volume IV) can be applied to meet his needs.
- Volume IV Methodology; analyzes planning tools for system design and evaluation and interprets their use in planning tactical command and control systems.
- Volume V Technology; collects for system planners basic information on current and projected electronic data processing and display technology of importance to the improvement of tactical command and control.

ANTACCS is a continuing study to assist planners of the Navy's tactical command and control system of 1970-1980. It is sponsored and directed by the Office of Naval Research and is supported by the Bureau of Ships and the U.S. Marine Corps.

The overall program is directed by Mr. Ralph G. Tuttle, the ONR Scientific Officer. The program benefitted from the assistance of a Study Monitor Panel consisting of representatives from:

Bureau of Ships
Bureau of Weapons
Naval Command System Support Activity
Office of the Chief of Naval Operations
Office of Naval Research, and
United States Marine Corps

The first phase of the study was carried out by Booz Allen Applied Research, Inc. and Informatics Inc. from January 1964 through January 1965. Booz Allen Applied Research Inc. prepared Volume II and supplied parts of Volume I. Informatics Inc. prepared Volumes III, IV, and V, and the rest of Volume I.

TABLE OF CONTENTS

		<u>Page</u>
	Abstract	ii
	General Preface to All Volumes of the Final Report of the First Phase of ANTACCS	iii
	Table of Contents	v
	List of Illustrations	vii
	List of Tables	ix
SECTION 1.	INTRODUCTION	IV-1-1
1.1	Perspective	IV-1-1
1.2	Observation and Recommendations	IV-1-4
SECTION 2.	METHODOLOGY FOR SYSTEM MANAGEMENT	IV-2-1
2.1	Introduction	IV-2-1
2.2	Evolutionary System Implementation	IV-2-3
2.3	Management of Evolutionary Implementation	IV-2-19
2.4	Operational Analysis and System Design	IV-2-42
2.5	Hardware Design and Production	IV-2-59
2.6	Software Design and Production	IV-2-80
2.7	Naval Procedures in System Planning	IV-2-105
2.8	Costing, Effectiveness, and System Planning	IV-2-117
2.9	Cost Estimation	IV-2-128
2.10	Effectiveness Measurement	IV-2-148
SECTION 3.	METHODOLOGY FOR SYSTEM PLANNERS	IV-3-1
3.1	Introduction	IV-3-1
3.2	Simulation in System Design	IV-3-3
3.3	Simulation in System Development	IV-3-20
3.4	Simulation in System Checkout	IV-3-25
3.5	Simulation in Test and Evaluation	IV-3-29
3.6	Simulation Languages	IV-3-46
3.7	Mathematical Modeling	IV-3-52
3.8	Queuing Models for ACDS	IV-3-56
3.9	Data Base Usage and Update Model	IV-3-67
3.10	Figures of Merit for Digital Computers	IV-3-88

TABLE OF CONTENTS (Continued)

		<u>Page</u>
SECTION 4.	BIBLIOGRAPHY	IV-4-1
4.1	Management Methodology	IV-4-1
4.2	Planning Methodology	IV-4-2

LIST OF ILLUSTRATIONS

<u>Figure</u>	<u>Title</u>	<u>Page</u>
1-1	Approach to Methodology Studies	IV-1-2
2-1	The Evolutionary Implementation Cycle	IV-2-14
2-2	An ACDS System Management Office (An Example)	IV-2-30
2-3	General System Requirements Analysis Step; Operational Analysis and System Design	IV-2-44
2-4	Task Definition Step; Operational Analysis of System Design	IV-2-49
2-5	Procedure Definition Step; Operational Analysis and System Design	IV-2-52
2-6	Analysis and Design Check Step; Operational Analysis and System Design	IV-2-55
2-7	Time Spans for Hardware Design Phases	IV-2-60
2-8	Hardware Design Phases	IV-2-61
2-9	Processing Computer Program Improvements	IV-2-87
2-10	Major Elements of Program System Design Phase	IV-2-89
2-11	Major Elements of Program Design Phase	IV-2-93
2-12	Major Elements of Program Production and Test Phase	IV-2-96
2-13	Major Elements of System Test Phase	IV-2-100
2-14	Major Elements of System Operation Phase	IV-2-103
2-15	The Six Categories of "Program 6" (Research and Development) of the Five Year Force Structure Program	IV-2-106
2-16	Naval System Planning from RDT&E Policies to ADO/SOR	IV-2-108
2-17	Naval System Planning from ADO/SOR to Engineering Development	IV-2-109
2-18	Major Procedures Governing Naval System Planning	IV-2-110
2-19	Centralized Configuration	IV-2-126
2-20	Distributed Configuration	IV-2-126
2-21	Mixed Configuration	IV-2-126
2-22	The Five Most Predictive Cost Variables in Computer Programming, Shown in Their Prediction Formula	IV-2-144
2-23	Example Arrangement of Scalar Values for an Irreducible Characteristic	IV-2-158
2-24	The Evolution of Two Hypothetical Systems	IV-2-160
2-25	Characteristics of Two Hypothetical ACDS Components	IV-2-164
2-26	Comparison of Two Hypothetical ACDS Components	IV-2-165
2-27	A Non-Quadrille Evaluation of Consoles C and D	IV-2-166

LIST OF ILLUSTRATIONS (Continued)

<u>Figure</u>	<u>Title</u>	<u>Page</u>
3-1	Command Data System	IV-3-7
3-2	Hypothetical System Design	IV-3-13
3-3	Interceptor Model	IV-3-16
3-4	Block Diagram of Design Optimization Problem	IV-3-21
3-5	Carrier Transmission System	IV-3-22
3-6	Functional Block Diagram of F-48 Intercept Simulator	IV-3-35
3-7	Radar Data Input Simulation	IV-3-42
3-8	Using the Cockpit Simulator	IV-3-43
3-9	Interactions Between ATDS, NTDS, and MTDS	IV-3-44
3-10	Two Dimensional Air-to-Air Missile Simulation Block Diagram	IV-3-54
3-11	Simplified Block Diagram of a Queuing Model Simulation for the RW-60	IV-3-61
3-12	Typical Results of a Queuing Study	IV-3-63
3-13	Relations of Computing Systems	IV-3-68
3-14	Effects of Change of Data Base Items	IV-3-73
3-15	Model for Distribution of Data Base Change Rates	IV-3-74
3-16	Model for Data Base Usage in Central Processing Non-Equal Usage Bases are Expressed by $(f_u + a)(F_D + a) = c$	IV-3-78
3-17	Data Transmission Rate as a Function of Fraction of Data Base Centrally Maintained ($K = 0.0822N$, $R = 0.73$)	IV-3-79
3-18	Optimum Values of the Centrally Maintained Fraction as a Function of the Ratio of Overall Usage Rate to Change Rate	IV-3-87

LIST OF TABLES

<u>Table</u>	<u>Title</u>	<u>Page</u>
3-1	Selected Possible Design Points	IV-3-86
3-2	Basic Data from Adams (Nov. 1963)	IV-3-93
3-3	Values of the Multiplier "K"	IV-3-104
3-4	Highland Method Figure of Merit (With K for ~0.8 Value)	IV-3-107

Section 1 INTRODUCTION

There will be some kind of system (or systems) performing tactical command and control tasks in the 1970-1980 time period. No assumption is made a priori in this report as to the type of system (or systems) which will be operational. It is necessary throughout this report to refer to the undefined system (or systems). For convenience, and to avoid having to repeat a long descriptive phrase each time reference is made to this generic system, the term ACDS (Advanced Command Data System) is used throughout this report. It is NOT INTENDED that this term be identified with any system (or systems) currently under development.

1.1 PERSPECTIVE

In his continuing role, the planner of tactical data systems for the Navy must be concerned with the requirements for system improvements. That is, on the basis of increasing threat or changes in operational doctrine he must determine the need for improvements. The planner must also be concerned with the technology which is available to him so that he can continually evaluate hardware and software techniques as to their role in the development of improvements to command and control systems. However, he must also give continuing attention to selecting and developing techniques for the implementation of these improvements. It is with the area of technique selection and development that this volume on Methodology is concerned.

The increased threat and improved technology tend to impel the planner to make changes. Questions of cost and compatibility of these changes constrain him. Methodology is concerned with the methods and procedures for making changes. In other words, Methodology is the study of the tools and techniques for examining these impelling and restraining forces and for the continuing management of the implementation process once decisions are made on system changes. The rapidly increasing complexities of tactical command and control systems, from the standpoint of operations and systems technology, implies an ever-increasing need for improved methodology, and an ever-increasing challenge in the development of methodological techniques.

The general approach taken for methodology studies in ANTACCS is illustrated in Figure 1-1. Since there has been a sizable development of management and technical

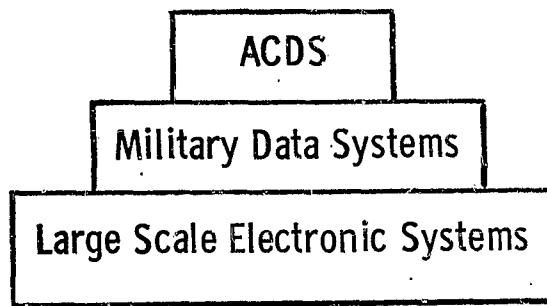


Figure 1-1, Approach to Methodology Studies

methodology for the development of large scale electronic systems, the study team considers this as a point of departure and a foundation on which further methodology studies should be based. However, military data systems have characteristics which differ from general electronic systems. Methodology for military data systems is studied as it exists in practice, or methodology is studied and developed by extrapolating from the methodology for general electronic systems. However, to be more specific and more useful, it is desirable to cast the methodology considerations in terms of the particular problems of ACDS and the particular Navy management and technical environment of ACDS. As a result, the general approach can be considered as the development of a structure based on considerations of general electronic systems, and building on this to the specific problems of ACDS. However, it is noted that methodology techniques and principles for large scale electronic systems and military data systems not specifically oriented to ACDS are, nevertheless, still important to the ACDS planner since they provide him with background and, in many cases, allow him quite rapidly to apply the techniques to ACDS problems.

To be more specific about the general approach taken in ANTACCS the various methodological techniques and principles deemed applicable are first identified. Following this, they are analyzed and evaluated as to their applicability to ACDS. The methodology is broken down into two major areas: management methodology and technical methodology. Management methodology deals with the administrative and management problems of improving system capability; technical methodology deals with techniques for developing answers to design questions. In this volume, Section 2, Methodology for System Management, deals with the former, and Section 3, Methodology of Systems Planning, deals with the latter.

Methodology for systems planners is a challenging subject from many points of view. It is also rather abstract, since there is an inherent non-numerical nature of the subject. In fact, one of the challenges of modern methodology is to develop quantitative approaches to many of the problems. The subject touches on every aspect of activity in systems planning, from decisions on circuit development to decisions regarding the task force commander's use of the system. Finally, the subject is relatively new and poorly understood, especially in connection with large scale systems, and it must be developed to be of use to many different kinds of planners with widely differing requirements.

However, the payoffs for improved methodology are great. Calendar time and costs can be saved by improved management, technical methods, and procedures for system implementation. The study of methodology is essentially a process of introspection and self improvement for the body of naval systems planners. It is quite apparent that, in view of the challenges and the possible payoffs, the Navy should give far more effort to improving methodological tools and understanding methodological principles.

1.2 OBSERVATIONS AND RECOMMENDATIONS

In the following paragraphs the principal points arising from ANTACCS methodology studies are presented.

Evolutionary Approach to System Design. This approach, frequently referred to as "evolutionary implementation", means that as the requirements, environment and technology change, increments of system capability are developed. This approach to system design is in great favor in the Department of Defense. An important aspect of the evolutionary concept as applied to ACDS is that this system will evolve from the present command posts, CIC's, and from NTDS, MTDS, and ATDS.

There are many benefits which accrue from employing the evolutionary approach to system design. These benefits include: shorter lead times, improved and more orderly development of evolutionary doctrine, better scheduling and distribution of costs, and more efficient utilization of Navy resources. However, evolutionary implementation generates a number of challenges or problems such as:

- 1) It creates additional management interface problems since system designers and system implementers must coordinate their activities in a more detailed way with operational units.
- 2) It is necessary that the hardware and software of systems be expandable. That is, it must be possible and convenient to add new memory, processor or display units to an already existing system. Also, it must be possible and convenient to add portions of computer programming to an existing program system.
- 3) Hardware and software should have a general purpose capability (without a cost/effectiveness compromise). This implies, for example, that a display console should be of such a design that it is useful for many types of applications.

The technical problems incurred by evolutionary implementation are especially significant. In the past, except for the computers themselves, data handling equipment has been very much of a special purpose nature. Some important changes in thought must take place in this connection by system planners to overcome these obstacles to successful and orderly implementation.

The System Management Function. In planning new systems and improvements to existing systems, there is a need for one coordinating point or office. This point of coordination might be referred to as a "system management office" in much the same way as the project offices within CNO and CNM. There are many functions which an office of this type should perform such as the following: liaison and coordination, developmental support, implementation planning, program management, operation analysis and system design, and technical support. It is believed that such an office could be established within the framework of the traditional roles of CNO and CNM. The office could be set up in a fashion similar to offices which have been established for Fleet Ballistic Missile and Anti-submarine Warfare. However, one difficulty is that ACDS is not yet regarded as a system. It is noted that the size and charter of such an office would depend greatly upon the purview and size of ACDS as it develops.

Navy Procedures. One of the areas of effort for ANTACCS methodology is to examine procedures for obtaining the required approvals within the Navy Department and the Department of Defense to implement ACDS. The following observation arises from this part of the methodology study. A literal interpretation of the instructions covering the preparation of the SOR, TDP, PDP, etc. are that each incremental improvement to ACDS would have to proceed through unnecessarily tortuous procedural paths. The procedures seem to be oriented toward large scale systems and revolutionary changes rather than evolutionary implementation. If this interpretation is correct, it appears that efforts should be directed toward modification of these procedures to accommodate the evolutionary changes to be made in ACDS (and in other systems as well).

Another observation made as a result of the methodology studies is in regard to the preparation of the TDP in response to the Advanced Development Objective 31-05X. The work of this phase of ANTACCS provides an excellent point of departure for the technical work which needs to be done to write a high quality TDP. However, much work must be done along the following lines before a TDP can be written: definition of the scope of ACDS, functional and technical description of interface systems and the nature of their interface with ACDS, and definition of functions to be automated by data processing to make appropriate dollar and schedule forecasts.

Cost and Effectiveness: Cost and effectiveness techniques are seeing increasing use in systems evaluation in the Department of Defense. However, there has not been much activity in cost and effectiveness studies for military data systems. There are few good techniques for estimating programming costs, for example, and as a result they are very often under-estimated. A formulative technique has been developed for estimating programming costs and is described in this volume. Effectiveness is difficult to measure because of the problem of quantizing effectiveness, the very great pervasiveness of the system, and because of the great scope of the system. Note that before cost and effectiveness can be studied satisfactorily, a system has to be defined accurately as to the functions it is to perform. The study recommends that cost and effectiveness studies be further supported by the Navy, especially as they apply to systems such as ACDS.

Simulation. Simulation is a useful tool for development of large scale data handling systems. Although the study team found that the Navy has successfully used simulation techniques it is noted that most of that simulation involved operational and training matters rather than detailed design or the development of specialized techniques. Simulation can also be used to provide answers to detailed design questions. It is in the latter type of investigations that simulation should receive more emphasis. Tools for improved simulation, such as simulation languages, should likewise receive support.

Formulative Techniques for System Design. This refers to quantitative techniques to provide answers to design questions. The formulative techniques referred to here involve the development of quantitative relations describing system components or procedures. For example, the use of a queuing theory model, to examine the real time operation of parts of ACDS, is a technique which merits further development. A number of other techniques are discussed in this volume and are typical of activities which merit continuing support.

Section 2

METHODOLOGY FOR SYSTEM MANAGEMENT

2.1 INTRODUCTION

As stated in the introduction, the Methodology studies are considered in two parts, Management Methodology, and Technical Methodology. The former is covered in this section. The latter is covered in Section 3, Methodology for System Planning.

This section treats the selection and development of tools for the system planner from the standpoint of the manager or administrator. It also covers topics which are appropriate for consideration by top Navy management personnel. It covers such points as the philosophy of system implementation, approaches to the management of systems, Navy procedures in system planning, and the measurement of cost and effectiveness of data systems.

In Section 2.2 evolutionary system implementation is treated; evolutionary implementation is defined and its benefits and problems are discussed. Following this, the management aspect of evolutionary implementation is presented in Section 2.3. The potential role of a system management office is presented as well as the process of implementation management for a naval tactical data system. The structure of a possible organization within the Navy Department is presented.

The next three sections present a further analysis of the process of system design. The various major steps in operational analysis and system design are covered in Section 2.4. Hardware design and production topics of Section 2.5 include the various steps taken in the development of hardware systems. Since software is so important to tactical data systems with their great dependence on the computer, it is treated in some detail in Section 2.6. The products, inputs, and the steps for software system design and production are presented. Also, system test and operation phases are presented.

The Department of Defense, and the Navy Department within it, are becoming much more concerned with the procedures for implementing the system. There are a number of formal steps to be taken along the decision and approval route in implementing systems. These procedures are analyzed and presented in Section 2.7.

Management methodology concludes with a discussion of cost and effectiveness. The subject is treated in two parts. Elements of cost and techniques of estimating cost are presented in Section 2.8. In Section 2.9, and 2.10, techniques for measuring effectiveness are analyzed and presented.

2.2 EVOLUTIONARY SYSTEM IMPLEMENTATION

2.2.1 General

Many would say, with justification, that NTDS and MTDS systems now in operation have developed in an evolutionary way. Even today, new functions and capabilities are being added to the basic AAW mission. These functions are being implemented by use of increased computer memories, changes to stored program computers, and added display capability. There is increased interest in the evolutionary process of system implementation by the Department of Defense for several important reasons. Therefore, it is important to analyze and discuss system evolution as it relates to ACDS.

This section develops the definition and concept of evolutionary implementation. This definition is important for uniform understanding of subject matter discussed in some following sections. After evolutionary implementation is defined, the reasons for evolutionary process for ACDS are given, and the benefits and problems in evolution are presented. An important aspect of the evolutionary process is its relation to modern technology. This is presented in Section 2.2.5. Section 2.2.6 discusses factors to be considered in deciding size and technical content of an increment of system improvement. Section 2.2.7 discusses steps in the evolutionary process.

2.2.2 The Definition of Evolutionary Implementation

Evolutionary implementation means that as requirements, environment, and technology change, increments of system capability are developed. Each new increment provides some increased capability to meet changing threats and to supply better support to commanders by using advances in technology. Each increment is costed and evaluated before it is added to the system. Each increment is designed to be compatible with the existing system to the highest possible degree.

Occasionally, these evolutionary increments are large. But even the largest does not disturb the operations and capabilities of the Navy to the same extent as development and implementation of a completely new system. Evolutionary increments are much more smoothly integrated into naval operations than are the massive changes of complete new systems, and they produce smaller perturbations to relatively constant budgets.

The four fundamental parts of the evolutionary concept are:

- 1) The system is evolved beginning with what is now at hand, hardware, software, doctrine, etc. For instance, ACDS will be evolved from the present command posts, CIC's, and from NTDS, MTDS and ATDS. These current capabilities will be expanded and enhanced. Seldom is anything "wiped out" to start from scratch.
- 2) Modest improvements are continually added to the system as changes in the mission, technology, or environment require. Changes such as improved inter-ship data links for NTDS AAW, a new program to compute air strike route data, or adding one more USQ-20B to support a command post, are typical examples of the evolutionary increment.
- 3) Each increment of improvement is specifically designed to be compatible with the system now in being. This compatibility is limited only by the requirement to take full advantage of advances in technology, and changes in mission and doctrine. A fine example of this integrated design concept is the CP-667 naval computer which is compatible with the CP-642B in all important respects and can run CP-642B programs but at the expense of decreasing its own efficiency. This computer, running at maximum capacity, provides a tremendous increase in computing capability over the CP-642B.
- 4) Each increment proposed for the system is carefully configured and evaluated to provide:
 - a) Highest military usefulness,
 - b) Least operational disruption,
 - c) Fiscal impact appropriate to budget limitations and the amount of operational capability being added.

2.2.3 Evolution and ACDS

There is no such thing as an unchanging system if it is to remain useful to the national defense. One of the important lessons learned from the Air Force "L-Systems" is that systems must evolve to meet new environments and to use new technology. If this must be done eventually, it should be originally provided for in the system.

Increasing emphasis upon a constant state of high operational readiness in all line units precludes tying up many vessels for a long time to install large "totally new" systems. Improvements must be made with as little interruption to readiness as practical.

Budgetary restrictions make it desirable to spread costs of procurement, installation, and training over many years to husband resources for those large expenditures that cannot be postponed. To meet continuing changes in the threat, technology, and doctrine, evolution is the most efficient way to invest in ACDS.

Modular computing machinery and modular general purpose display equipment now make it technologically feasible to add increments of capability to satisfy new requirements. (See Section 2.2.5)

Perhaps the most important reason why the evolutionary implementation concept is recommended for ACDS is that evolutionary implementation lets the designers and implementers of ACDS remain responsive to the changing needs of the line commander.

2.2.4 Benefits and Problems in Evolution

The principal benefits to the Navy brought about by using the evolutionary implementation for ACDS are:

- 1) Eliminates the vexing "all or nothing" decision when the Navy faces needs for new system capability.
- 2) Permits the addition of operational capability to current systems without needing the long lead times of completely new systems, and reduces the impact of these changes upon operational units.
- 3) Permits the gradual development of operational doctrine in parallel with system evolution instead of requiring a complete new doctrine first.
- 4) Permits better scheduling and distribution of system costs to comply with fiscal requirements and to meet fiscal goals.

- 5) Provides better capability to meet rapid changes in threat, operational doctrine and command requirements, and to take early advantage of changes in technology.
- 6) Permits the more efficient utilization of scarce Naval resources, such as shipyards, ranges and training establishments.

These benefits bring with them some system management problems. These problems are not peculiar to evolutionary implementation, but continuing evolution enhances the impact of each problem. Important examples are:

- 1) Continuing System Management. The primary characteristic of the evolutionary implementation is the time scattering of various system improvements throughout the implementation process. This means that implementation management and technical support tasks continue almost indefinitely, or until ACDS is abruptly and completely replaced. These continuing functions let the Navy use the characteristics of evolution and they require continuing expenditure of Operation and Maintenance funds to do this.
- 2) Timely Support and Line Liaison. It is important to provide timely and adequate support to the line commander. When new and massive systems are being designed, lead times can be so great that timely support techniques are overlooked. With evolution, much support can be given the commander despite short lead times. Techniques must be set up and liaison maintained so that such innovations as radically-advanced Interceptor or ASW tactics may be applied in the field with little or no delay. Fast response must be planned for and maintained to support an evolutionary ACDS.
- 3) Doctrine. Since capabilities of ACDS expand gradually, as a rule there is always some part of each task force which does not yet have all of the latest system changes installed. There will probably be greater differences than this between ships of each class, or between fleets. The doctrine which covers operations with variously-configured ships must be updated and quickly disseminated to line

commanders. This is similar to the NTDS - Non-NTDS ship problem, but it becomes more important as ACDS grows in capability and power, and as it reduces tactical response time.

- 4) Training. Each new increment to ACDS requires an increment of training. Some increments of training may take the form of a few pier-side lectures and one dry run. At the other end of the spectrum may be the requirement to set up a lengthy training program for the CP staff. After installation training, exercising and drills are required. However, increments may arrive on board an AGC two to six times per year when computer program changes are counted. This increased training and indoctrination load must be provided for.
- 5) Integration. All the variously-configured command posts and ships must remain compatible with each other, and increments of improvement must also be compatible. A substantial effort is required to ensure this continuing integration of all aspects of the system. It is a challenging task to design and schedule worthwhile improvements while maintaining maximum compatibility.
- 6) Technology. To make prompt and full use of expanding technology, program management must continue to monitor and evaluate technological progress in several fields. While this effort yields substantial benefits, it requires that talented technical and managerial personnel are applied to the task for the life of the system.

2.2.5 System Implementation and Technology

The current state of technology is far enough advanced to support the evolutionary implementation of large command data systems. Recent and current hardware and software developments simplify the system planner's tasks.

Evolutionary system planning could have been undertaken 5-10 years ago with the hardware and software then available. However, execution of this planning would have been very difficult and extremely expensive with that technology. Since that time, developments in hardware and software technology have increased the ability of system planners to implement large evolutionary systems.

Many facets of technology contribute to this ability. The most important are now discussed.

2.2.5.1 Hardware Technology and Evaluation

General purpose displays and display consoles provide a common hardware interface and common software requirements, which allow for the planned evolution of systems design. Additional commonality of displays and display consoles provides more advanced capability at a lower cost, reduces spares requirements, and requires less training of operator and maintenance personnel.

Multi-computers allow evolution of computing power from modest to large capability by common software and hardware interfaces. Additional modules of computing and input/output capability can be added at these interfaces as system functional requirements grow. This capability may be planned so that there is little interference to the operating system and only a few requirements for other software and hardware changes.

General hardware characteristics which further enhance the system planner's capability are reduced power requirements, physical size, and heat dissipation. These let the planner work with more general system concepts without encountering many hardware constraints. Other hardware trends which contribute to this capability are increased speed and reliability.

2.2.5.2 Software Technology and Evolution

Master control program techniques have developed to a sophistication which supports evolutionary system planning in permitting modules of system improvement. The concepts of a centralized data base, centralized input/output control, and separable units of independent operational programs, all contribute to the ability of software to support evolution. These concepts allow the system planner to make maximum use of the modular hardware now available for command data systems.

Most software producers and users have learned the expensive nature of documentation. The operational expenses of having too little documentation can be balanced, with careful planning, against the production expense of too exhaustive documentation.

The system user has always been able to modify his operational programs by using conventional program-changing techniques. These techniques require the services of skilled programmers. One programming technique under development lets the system user re-program parts of his software system, within certain limitations, using operational personnel. This technique is referred to as "user programming". It is particularly applicable to display and input/output format programming. It greatly enhances the process of software evolution.

2.2.6 The Size of the Incremental Step

Each evolutionary increment or step must be correctly sized to satisfy the urgency of the requirements which generate it, the schedule required for its operational deployment, the amount of technical production it requires, and the funds available to produce and deploy it.

There are no fixed rules for determining the best size of increment. Finding out the technical contents and scheduling of each evolutionary increment is critical to system planning. Each increment must be planned only after carefully considering several important factors, discussed in Sections 2.2.6.1 through 2.2.6.4.

2.2.6.1 Urgency

The principal factor in planning the size and schedule for a system evolutionary increment is the urgency with which that increment must be deployed. The key to the evolutionary concept is to provide increments of evolution which respond to rapid changes in commanders' requirements, to changes in both friendly and foreign technology, and to changes in the environment of threat and doctrine.

For scheduling purposes, increments and changes should be assigned to one of the four following categories of priority. Within each category, each increment or change should have its own specific urgency based entirely upon the requirements of line commanders, technology, and environment. General categories for evolutionary system increments are:

- 1) Emergency Field Changes. These are changes of such critical nature that the line commander implements them with the maintenance and operational personnel he has within his command or with

the assistance of hardware and software specialists in the field for the specific purpose of assisting in installation of the change.

- 2) Expedited Production Changes. These changes are urgent or important changes, the size or complexity of which precludes their being made by the line commander at the field installation. These changes have varying degrees of urgency or priority, but take precedence over normal scheduled production increments.
- 3) Normal Production Increments. These are the scheduled evolutionary improvements to the system. They are designed to take maximum advantage of planned and predictable changes in commander's requirements, operational doctrine, technology, and environment. These increments follow the normal production pattern, each change passing only through those steps required for its production and installation.
- 4) Preferred Changes. These are required improvements to system capability, but do not have enough priority to warrant their being produced and installed by themselves. A backlog of this type of change grows, and, as normally produced increments are planned and scheduled, preferred changes are added according to the degree of preference of the line commanders. Their inclusion in normal production increments is also limited by the availability of production capacity and funding required to implement the change.

2.2.6.2 Availability of Production and Installation Resources

The size and technical content of a system increment is limited by the availability of production and installation resources during the time when this increment is of interest to system management and the line commander. Critical and urgent changes may be forced through to protect the operational readiness of the system. Most evolutionary improvements, however, are generated through some "normal" production process. It is the residual capability of this production process at any point in time which limits the capability of the line commander and the system planner to add individual improvements to the next planned system increment.

The primary limitations are the availability of facilities and personnel, of hardware and software production, and of test and evaluation agencies. For large hardware changes, the availability of shipyard or pierside space and personnel is of importance.

2.2.6.3 Perturbation of Line Units

It is difficult to set a precise numerical limit on the number of times each year a line unit should be interrupted by the installation of a major system increment. It is clear that between each major increment of system improvement, each line unit must have sufficient operating and training time to regain and maintain its tactical efficiency. In weapons systems, field changes may be made on almost any interesting schedule, since most of these changes do not affect the way in which personnel operate their weapon system. In command data systems, almost the opposite is true. Nearly every system change or improvement affects how the staff officer or enlisted operator performs his task or interprets the system outputs as they are shown to him. The effect of the proposed change upon line unit training requirements and tactical efficiency must be considered by system planners.

2.2.6.4 Costs and Available Funds

There are three general cost considerations to be taken into account by system planners when designing and scheduling increments of improvements to ACDS.

First, consideration must be given to keeping the available ACDS production facilities intact and producing. Some modest resources must be devoted to designing and producing evolutionary increments to ACDS. System planners should consider the cost (however small) of not using these resources to produce increments once they are established.

Second, the costs of management and administration for each increment to ACDS will remain rather inflexible regardless of the size or technical complexity of the increment. Therefore, system planners should consider the technique of including as many changes, as are otherwise feasible, in each increment which is produced for ACDS.

Third, and most important, each hardware and software increment has production costs. Funds must be available in the current budget to produce and install the proposed changes and increments.

The four major factors of urgency, production capability, perturbation of line units, and cost have to be considered for every proposed ACDS increment and for every proposed change to be included within every increment. Since each increment normally consists of more than one improvement or change, a numerical designation is convenient for reference and administrative control.

2.2.7 The Evolutionary Cycle

The process of implementing ACDS in an evolutionary manner superficially resembles the classical implementation process of engineering texts. There are two fundamental differences, however.

- 1) Many different increments to the system are in various stages of the cycle concurrently. For instance, Model 3 of a data link may be installed on some ships and operational for others, while Model 4 is in a design phase. At the same time Model 6 of an AGC command post display is at the Naval Electronics Laboratory Development Center, while Model 8 of a CDS computer is in test and evaluation. This large mass of separate activities is difficult to integrate and control.
- 2) The classical implementation cycle provides for a feedback loop between the planner and the live commander. The long lead times required for massive systems atrophies this loop. Changes normally cannot be made in any acceptable period of time. The nature of evolutionary systems allows "quick-fixes" to meet priority command requirements in days or weeks. To provide the necessary responsiveness, special channels must be set up free from routine administrative delay.

System implementors know from experience with aircraft, ships, tanks, or computer systems that their planning, production and installation does not just happen. It must be provided for very carefully. The evolutionary process for ACDS requires a seasoned management activity. It also requires much technical support from naval staff organizations, naval line units, manufacturing contractors, and technical contractors.

The many details of the implementation process may be summarized and discussed as follows:

- 1) Generation of requirements
- 2) Operational analysis and system design
- 3) Sub-system design and production
- 4) Training plans
- 5) Test and evaluation
- 6) Personnel training
- 7) Installation and checkout
- 8) System operation
- 9) Feedback to system management
- 10) Correction and updating

The following sections (2.2.7.1 through 2.2.7.8) discuss the key phases in the evolutionary implementation cycle as shown in Figure 2-1.

2.2.7.1 Generation of Requirements

The evolutionary concept considers that the current capability deployed to the field is a system. New capabilities are evolved from this system. Under this concept, data from current line commanders now at sea or in the field becomes very important.

The requirements for the generation of capability increments come as a result of:

- 1) Suggestions and requests from line commanders
- 2) Studies conducted by developmental activities
- 3) Monitoring the advancing technology
- 4) Monitoring changes in threat, mission and other environment
- 5) Command requirements from senior naval headquarters
- 6) Studies of operations techniques

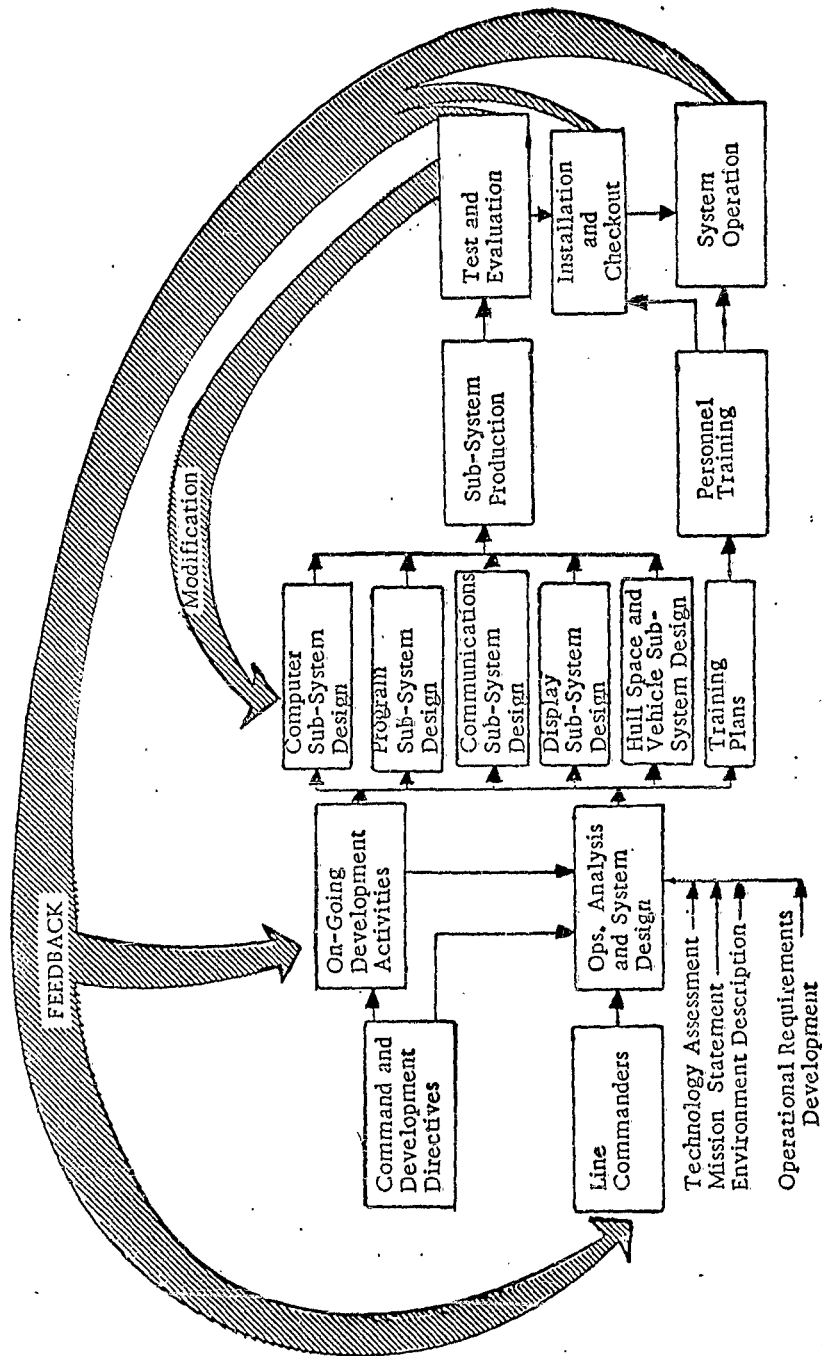


Figure 2-1. The Evolutionary Implementation Cycle

2.2.7.2 Operational Analysis and System Design

In this initial phase of system implementation the system planners decide what sets of requirements, available technology and scarce resources match to provide an increment to current or project capability. Increments are normally planned to accommodate a change in threat, environment or doctrine.

They analyze formal requirements and operational procedures; producing functional requirements and supporting documentation. These requirements are checked against equipment availability, manual capabilities, and operational doctrine to provide sets of tasks specified for operators and machines. These tasks, their definitions and the rules for performing them are the basic system design.

Operational analysis and system design culminates in the preparation of a preliminary operational system description. When this is agreed upon, the operational system description and its supporting documentation are sent to the agencies responsible for the design of the required subsystems.

In evolutionary systems an improvement increment can be small or large. But careful operations analysis and system design is needed to ensure compatibility and operational usefulness. Several increments of quite different purpose and scope are likely to be under consideration at the same time. This shows the need for analysis and design teams assembled for specific tasks such as AAW, Amphibious Warfare, Strike, and ASW.

With so many possible increments under consideration at once, particular attention must be paid to evaluation and testing of each new increment of system design before it is released for subsystem design. Computer simulation is an ideal tool for reducing this workload and for obtaining more complete conceptual testing than can be done manually in the available time.

2.2.7.3 Subsystem Design and Production

This phase is much the same as in the classical or massive system. The primary difference is that the various contractors and naval agencies often are processing modifications to subsystems rather than entire new subsystems. Certain ACDS increments require large and complete subsystems. For instance, providing automated assistance

to the TFC command post in an AGC requires installation of complete computer, program, and display subsystems. In other circumstances, such as changing interceptor tactics to accommodate new missiles, the entire increment can be represented by a change to only the computer program subsystem.

After the subsystems are designed, preliminary technical specifications are exchanged between the contractors involved with that specific increment, then sent to system management. When these specifications are concurred upon, an absolute control on design changes begins and the subsystems are produced.

2.2.8 Training Plans

As soon as the operational system description and supporting documentation emerge from system design, training specialists begin to plan for personnel training. This planning cannot be completed until technical specifications are agreed upon. Even then it must be changed as engineering change proposals are accepted.

Training plans should be made in parallel so that trained personnel, training aids, or both, can be deployed concurrently with the hardware and software subsystems for a particular increment.

2.2.8.1 Test and Evaluation

It is often advisable to hurry one complete set of hardware and software into test and evaluation. All design work is compromise and, occasionally, the unforeseen results of these compromises are not operationally desirable. Rapid feedback from test and evaluation allows production fixes to be made or the problem to be solved in the next increment to be designed. An evolutionary system is almost self-healing.

2.2.8.2 Personnel Training

This activity begins when newly-trained personnel are available in the field during the installation and checkout of a particular increment. Not only are they of assistance during the installation, but also they can enhance their training by assisting.

It is important not to have the training completed on-site or in the field too far in advance of increment installation. The trainees grow stale if not able to practise with the new capability.

During this phase training aids are designed and produced for classroom and field use. They must also be timed backward from installation and checkout.

2.2.8.3 Installation and Checkout

In shipboard systems the installation of other than small increments can pose a severe scheduling problem. Although this problem cannot be eliminated, evolution mitigates it somewhat, since more capability will likely be added through a series of small changes.

In this step also it is vital that rapid feedback is transmitted to system management so that corrections to design or operational procedure can be formulated and installed quickly. It is desirable to accelerate the first installation as much as possible to provide this feedback.

2.2.8.4 System Operation

Once the new system increment has been checked out and is being employed operationally, increased field liaison is called for to capture the new ideas it stimulates in the operational crews. When new increments are first used in the field, operating personnel are full of questions and suggestions. As the newness wears off these ideas become less frequent.

For this reason it is often desirable for designers to go to sea, or to go on maneuvers with the first units to receive these increments. Most designers can improve their future products by a better understanding of the problems of noisy communications, poor ventilation, cramped command posts, and dim displays. A regular protocol should be established to make this post-installation liaison an expected practice on the part of the senior designers and members of system management.

Field operation brings problems of maintenance, and here, also, rapid feedback is required to make the best use of the evolutionary system implementation concept.

2.2.8.5 Feedback

The feedback path leads from test, installation and operation upward through the line command and laterally to system management and development centers. This two-way reporting keeps line commanders informed at all echelons concerning the readiness of their system. It also allows timely and accurate reporting of suggestions and difficulties directly to the system manager and his technical support.

For a widespread system such as ACDS with its many equipments, procedures, and programs in the field, some special reporting technique such as the red-bordered aircraft "Unsatisfactory Report" should be instituted. A green-bordered "System Report" with its own expedited channels would be very effective.

2.2.8.6 Correction and Updating

As soon as feedback information from the field reaches system management, corrections are developed for field installation, and the newer commander's requirements are entered into the planning system. At this point the evolutionary development cycle starts again.

2.3 MANAGEMENT OF EVOLUTIONARY IMPLEMENTATION

In the previous section, evolutionary system development was described and analyzed. The question arises as to how evolutionary system development should be managed. Questions of the method of management of the implementation of ACDS involve a wide range of factors such as: operational requirements, technical system concepts, Navy organization, and present management techniques. In this section, many of these factors are considered in analyzing system management.

2.3.1 The Potential Role of a System Management Office

Section 2.3 describes the functions and organization of a system management office for ACDS. This study recommends that consideration be given to establishing an office of a size and scope commensurate with the size and scope of ACDS. In other words, if ACDS is indeed considered a system in the sense of the traditional weapon systems of the Navy, and if the decision is made for it to be of a considerable size and complexity, then a sizeable management office seems justified. The existence of such an office is probably justified even if ACDS is of very modest size, consisting of very minor improvements to the present NTDS. In this instance, benefits would still accrue through having one coordination and liaison point.

In Section 2.3.3, a system management office is described. The functions of such an office are presented. The reader should not assume that the office need be staffed by a large group simply because many functions are identified and discussed. Rather, as stated above, the size depends upon future developing viewpoints regarding ACDS. The intention here, is to describe the functions of developing ACDS, whether that development is very modest or very complex. The recommendation that is made concerning the establishment of such an office is secondary.

There are a number of reasons in favor of the establishment of a system management office:

- 1) Due to the growing availability of hardware and operational techniques for handling data tactically, ever-increasing attention is being devoted to command and control, and tactical data systems.

- 2) The need for more centralized handling of tactical data being recognized more frequently. Hence command data systems for tactical use are more frequently considered in the same way as weapons systems are considered, and weapons systems have benefited from a systems management approach.
- 3) System complexities; the pervasiveness of command data systems, and the operational and technical problems within the Navy raise important question of Navy-wide coordination and liaison.

In the following paragraphs, each one of these reasons is discussed briefly.

Throughout the Department of Defense, increasing attention is being given to tactical command data systems. There is steadily increasing interest in the Army's CCIS-70 Project. New developments are under study to handle fire support, logistics, intelligence, and other activities of the Field Army. Similarly, Air Force commands have recognized the need for tactical command systems. These have been under development for some time. Increasing attention has been given to the automatic handling of aerial reconnaissance intelligence for tactical uses. The multi-service STRICOM System has an active project under way for automatic operational data handling. It is logical to assume that the growing interest in naval tactical command data systems will continue, and that a greater percentage of the dollars spent for tactical capability will be represented by data handling equipment responsive to commander's needs. Growing costs, and the attendant requirement for efficiency, motivate increased thinking about a system management office.

Up to this point in time, command data systems for Navy tactical operation have not been considered as systems in the same way that weapons systems have been considered. Project offices exist for most weapon systems projects but none exists per se for ACDS, or for that matter, NTDS. As the role of ACDS becomes more clearly defined and more thoroughly understood, the need for centralized coordination will become more apparent.

From a technical point of view also, some aspects of centralization for data handling give rise to greater management needs. Without regard to whether the future ACDS is centralized from a system point of view or decentralized, it will evolve

as a network of data handling and data communications equipment. This network will allow the commander to obtain up-to-date information on all aspects of his fighting force. Therefore, ACDS will be integrated no matter how the detailed design of the system develops. Because of this integration and because ACDS will, with increasing frequency, be regarded as a system, there will be technical complexities which require across-the-board coordination. ACDS will tend to become a system with capabilities for AAW, ASW, STRIKE Warfare, Intelligence, and perhaps even personnel and navigation considerations. Again, while the degree of elaborateness or the cost or size of ACDS may remain relatively modest, the pervasiveness of the system to all operational and technical aspects of tactical forces will become steadily greater. Again, the technical coordination of such a system on a Navy-wide basis appears as an ever-increasing management need.

At the present time, the functions of a system management office such as is described in Section 2.3.3 are divided among a number of organizations: Bureau of Ships, CNO, NAVCOSSACT, and the Fleet Programming Centers are examples. Whether or not many of the functions being performed by these groups should be taken over by a system management office is uncertain. However, there is much advantage in creating a point of coordination for the activities of these varied groups.

A very important question is where such a system management office should appear organizationally in the Department of the Navy. Hopefully, the office would be of sufficient stature to have established for it the special arrangement for special project offices such as that for the Fleet Ballistic Missile which cut across CNO/CNM lines. However, it is doubtful whether during the next few years the importance of tactical command and control will be judged to be sufficiently high by top Navy Department officials to warrant such an organizational arrangement. Certainly from the standpoint of the dollars spent, tactical command and control systems cannot rank with Fleet Ballistic Missile or Anti-Submarine Warfare activities with their large hardware needs. However, an organizational arrangement which would cut across CNO/CNM lines would be highly desirable to accomplish the coordination desired and will probably come about in years to come. Meanwhile, a coordination point in BuShips or in CNO should be established. Perhaps the responsibilities could be principally vested in BuShips Code 607 with elements of CNO/CMC having a continuing coordinating responsibility, or perhaps it

should be with the principal point of coordination in OPNAV Code 353, with a coordinating responsibility vested with BuShips.

It is important to emphasize that the contribution of this section with regard to a system management office is the understanding of the various functions which are necessary. The organizational location and the exact constitution of such a group is of great importance, but is not the main point of the remaining portions of this section.

2.3.2 The Process of Implementation Management

The implementation of ACDS requires the coordination and cooperation of many vital naval activities, such as ONR, BuShips, BuWeps, CNO, CMC, CNM, BuSandA, and Yards and Docks. At some point in time, inputs from and outputs to these agencies must come together and be coordinated. The system management office is the type of organization which would provide the required representation and control.

Each of the interested naval activities would provide suitable personnel to a system management office on a long-term basis so that the interests and technical competence of each activity would be appropriately considered. This type of organization is required because of the pervasive impact of evolution over the entire life of ACDS.

The ACDS system management process consists of six closely related functions:

- 1) Liaison and Coordination
- 2) Developmental Support
- 3) Implementation Planning
- 4) Program Management
- 5) Operations Analysis and System Design
- 6) Technical Support

The first three of these functions are general in nature and are performed in part, or supported by, all persons and offices in a system management activity. Those first three functions are discussed in the remainder of the section (Section 2.3.2), and are independent of the structure of the organization which would perform them.

The last three of these functions are also supported in some degree by all parts of a system management activity. However, they are very closely related to the structure of the organization which performs them. For this reason, they are discussed in Section 2.3.3, which describes one possible form of an ACDS system management organization.

There is some difference of opinion as to how centralized and authoritative a system management office should be. Without regard to this question, a number of specific critical tasks must be accomplished. An ongoing competent technical responsibility and unimpeachable source for system technical detail must be maintained. There must be a coordination mechanism for the various schedules, problems, requirements and organizations involved with the system.

The discussions which follow are based upon two concepts:

- 1) The stated functions must be performed in some organization or set of organizations.
- 2) The functions must be performed by an activity which is senior, or is respected, to the extent that the results will not be consistently challenged nor countermanded.

2.3.2.1 Liaison and Coordination

One of the important functions to be pursued by the system management office is to develop planning and analysis techniques and to interchange this information with similar agencies in the other services and at DOD level. This interchange of information will insure that the Navy remains abreast of new system planning and estimating techniques as they are developed.

The system management office must maintain close liaison with other Offices, Bureaus and Divisions within the Navy and Marine Corps so that it may obtain timely and accurate information to support ACDS technical and operational system decisions. Information must be maintained and updated concerning such items as: delivery schedules of electronic systems, changes in shipyard facility availability, changes in the availability of training facilities, and even the availability of the results of war gaming and naval exercises.

In addition to the system management office providing a funnel for inputs, it also provides the authoritative source from which other naval agencies may obtain managerial and technical information concerning the system, its current and projected configurations, its technological progress and its managerial schedules.

2.3.2.2 Developmental Support for Evolution

The second important general function of the system management office is planning and coordinating the three-stage development process which is required to support evolutionary implementation. This process is not created to support evolution; it exists already. However, the recognition of the three-stage nature of development and the proper coordination of its stages are of great importance to the proper support of evolutionary implementation.

In the first stage, experimental operations, short range improvements are made to current operational capability and to exercising and evaluation capability. The lead time from identification of a needed improvement to its incorporation in current capabilities is less than six months. (By incorporation in current capabilities is meant that the indicated improvement has at least reached the stage of development and testing that it can be run in parallel with current operational capabilities.)

In the second stage, medium range improvements are developed and evaluated where these improvements are expected to need a three month to two year lead time before they become operational.

Experimental exercise and evaluation capabilities are maintained to stimulate ideas for medium range improvements and to provide a test-bed for evaluating these improvements. This stage would evaluate such ACDS capabilities as: improved group display devices, user-programmed displays, or an improved strike route planning program.

In the third stage, an analytic center is operated whose concerns and tools are at a much more abstract level than those used in the centers in the first two stages. The outputs of this third center assist all agencies in planning and analyzing requirements and designs. Certain major EDP and hardware techniques may be shown to be tentatively feasible and ready for further development and experimentation in the second stage. Also,

a development program in EDP technical tools is conducted as a part of this stage. The third stage looks as much as five years into the future, and none of its developments would likely be operational in less than a year (and then only if they were expedited with highest priority through the second and first stages). In support of these three stages, system management activities specify and develop the short and medium range improvements, and the experimental models, perhaps through the assistance of a technical support contractor. This stage would evaluate such improvements as a new problem oriented language or a new computer module.

In planning the allocation of resources to these various activities, it is essential to remember that this organization is intended to provide an almost continuous flow of products and data. If resources are not properly allocated among the various stages and activities, serious bottlenecks or gaps can occur. Fortunately, such a multistage development process is partially self-adapting so that a balanced flow of products and design data is normally achieved. A major role of system management is to monitor the flow of development products through these diverse activities, and to adjust the allocation of resources and the interrelationship between the activities so that efficient and appropriate ACDS development projects are pursued.

An initial plan for the organization of development would have to consider such questions as:

- 1) What resources should be allocated to each stage?
- 2) What relative emphasis should be placed on design and development versus exercising and evaluation?
- 3) Can some of the same facilities be used for both current operations and experimental operations?
- 4) What types of experience are required to perform each of the activities: user, user representatives, analyst, data processing designers, etc.? In managing them? In planning for them? In monitoring them?

- 5) How can operational needs be applied to guide the development of technical tools? To what extent are these tools operationally substantive (e.g., planning models) versus general (e.g., executive systems), versus operational (e.g., artillery fire support systems).
- 6) What documents are required to describe plans, needs, products, evaluations and tools?

Although these questions have been posed with respect to the three stage development mechanism discussed above, they will have to be addressed in the implementation plan. The plan must also consider these additional (and possibly more difficult) questions:

- 1) How many stages does the user need in the development process?
- 2) What is the lead time for the various stages?
- 3) What is the role of present agencies in the proposed mechanism?

2.3.2.3 Implementation Planning

The planning of an evolutionary process for introducing command data systems into a command organization is unique. For, by identifying the process as evolutionary, we emphasize that ACDS development will be dominated by some uncertainty. We cannot anticipate with high accuracy exactly how operational requirements will change, how technological advances will proceed, how commanders and their staffs will profit from automated assistance, or how various command organizations will be restructured or their scope modified. These are a few of the unknowns.

An evolutionary implementation plan handles different problems in different ways. It may establish an organization for attacking the problems without anticipating what the specific solution may be. It may use the planning process to recognize long lead time implementation choices. Although the plan attempts to delay as much as possible the time when these decisions are made, excessive delay will impede future progress; accordingly, in selecting a time for making these decisions, the plan must consider the tradeoffs between uncertainty and delay. Finally, the plan must anticipate the continual need for replanning. It can only do this if it provides for the most thorough technical

and operational monitoring and managerial or project control. Over time, original assumptions prove valid or invalid, schedules are bettered or missed, managerial and technological progress is greater or less. A good plan will suggest when replanning is called for and, possibly, the nature of the corrective action needed.

2.3.2.4 Contents of the Implementation Plan

The Implementation Plan should address the following:

- 1) Goals and phasing objectives for EDP.
- 2) Organization and activities for ACDS development.
- 3) Measures for change, allocation and planning.
- 4) Current and imminent progress.
- 5) Software development.
- 6) Hardware planning and procurement.
- 7) Problem areas.
- 8) Proposed activities.
- 9) Plan modification.

A brief discussion of each follows:

Goals and Phasing for EDP - To what extent will EDP support be required in ACDS to serve operations, intelligence, logistics, communications, gaming, and planning? To what extent can the data bases and processing routines in support of these functions be integrated? What other developments will be taking place during the coming five or so years which will have a major effect on the role of EDP support? What functional needs should guide early development activities? Given significant alternate long range configurations, what intermediate milestones must be achieved to attain each long range goal? What critical decision points exist in selecting between alternate configurations?

Organization and Activities for ACDS Development - How many stages should be planned for developing ACDS? What is the relationship between these various stages? What documents and other products must be generated in performing each of these functions? What agencies are responsible for originating, reviewing, coordinating and approving the various documents?

Measures for Change, Allocation and Planning - What quantitative measures can be applied in planning or reviewing the growth or change of ACDS? What are present planning factors for supporting resources (including various types of personnel) needed to achieve the above measures? What guidelines exist for allocating resources devoted to current operations, current exercises and evaluation, analyses of potential improvements, operational specification of ACDS functions, computer program design and implementation, development of exercise and evaluation support and tools, maintenance of systems (including minor modification)?

Current and Imminent Progress - What is the current manning, experience and history of the various units using tactical EDP in the Navy? What EDP capabilities are currently operational? What EDP developments are scheduled for early operation? What are the current relationships between the various services using and developing tactical EDP? How do present accomplishments compare with past plans and why?

Software Development - How much and what research and development in software tools should be sponsored by the Navy? How would these research and development activities be related to non-Navy R&D? What developments can be undertaken which are not operationally specific; for example, executive programs, time sharing systems, query languages, data base management systems, modeling ideas, etc.? What user or operational guidance is required to initiate such efforts and subsequently to monitor their development? When might significant new developments be ready for incorporation in experimental or operational EDP systems? What steps must be undertaken to ensure that such new capabilities can be introduced into experimental or operational systems with minimum disruption?

Hardware Planning and Procurement - How should the procurement of improved data processing, display, communications and input devices be programmed? What constraints does the normal programming cycle impose on procurement of these improved capabilities? Should the programming cycle be somewhat modified to facilitate the timely procurement of both major and minor hardware improvements? At the time of initial installation, how much processing capability should be reserved to facilitate growth over time?

Problem Areas - In preparing any plan, the planning process generally illuminates problem areas or uncertainties which fall outside the scope of the planning group or which cannot be resolved during the planning cycle. What are these areas? What specific issues and alternatives are involved? How does the plan cope with these problems? (How soon does it assume they will be resolved?) Can the EDP planning activity propose a means of resolving some of these problems?

Proposed Activities - In the light of the above, what changes are recommended to present plans, including changes in organizational relationships, procurement specifications and schedules, and level of supporting resources?

Plan Modification - How should the initial plan be revised? By whom? With what coordination and concurrence procedures? How often?

A number of these planning questions are within the scope of the current ANTACCS and MTACCS efforts. Others remain to be answered as the Navy develops more information about its future operations, the threat and the technology. Of course, the answers to these questions must be regularly updated to maintain the validity of the plan. This updating is one of the most important functions of ACDS system management.

2.3.3 An Organization for Evolutionary Implementation Management

The three system management functions which are independent of organizational structure are discussed in Section 2.3.2. In this section, the remaining three functions of ACDS system management are discussed. These three functions must be performed by any ACDS system management organization, but they are specific and technical, and are best explained by reference to an organizational chart.

The organizational chart referred to in this discussion (Figure 2-2) is specifically and carefully constructed to show an organization which could support the ACDS system management functions as described.

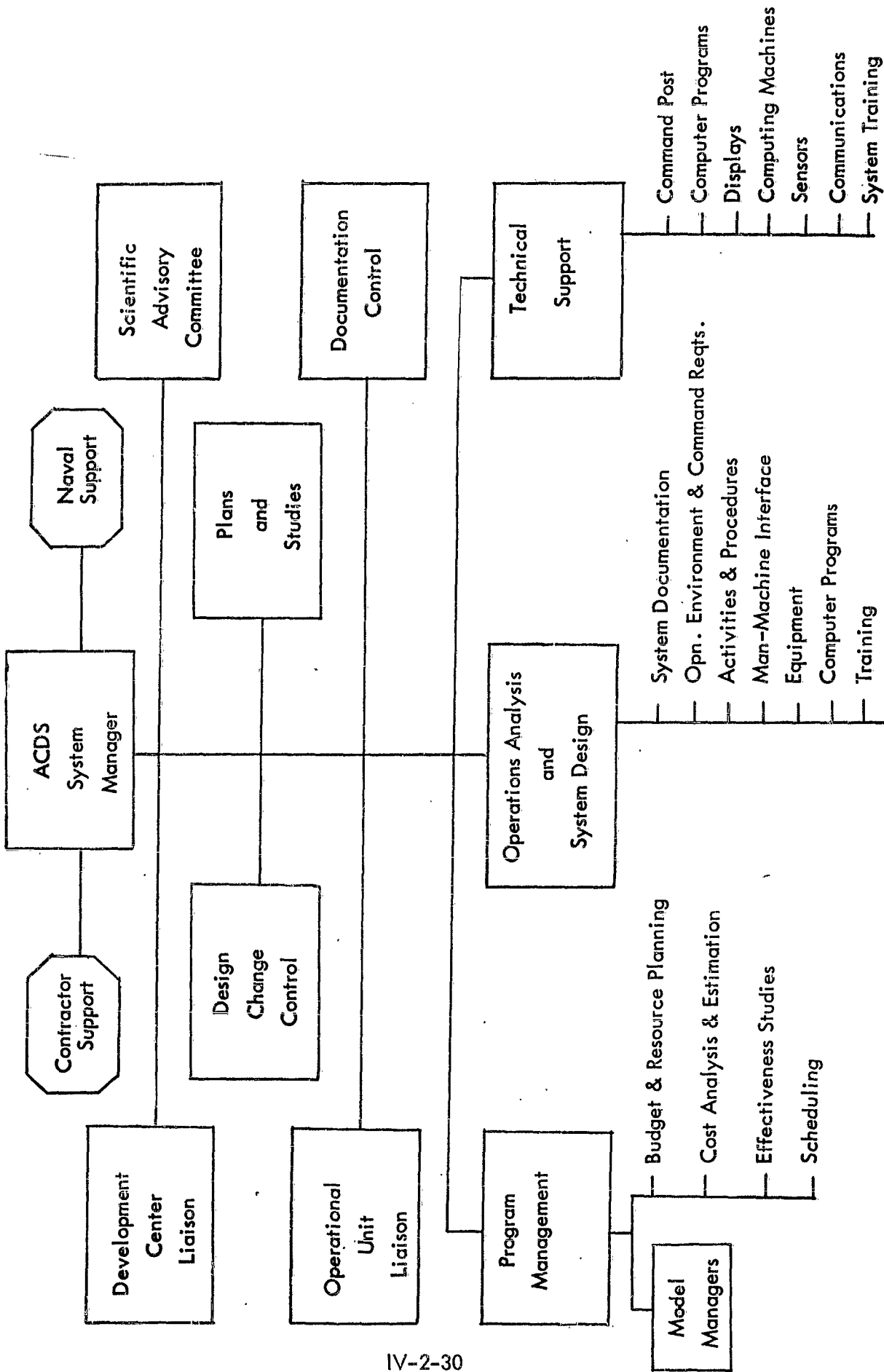


Figure 2-2. An ACDS System Management Office (An Example)

The system management functions discussed in this section are:

- 1) Program management
- 2) Operations analysis and system design
- 3) Technical support

In addition, Section 2.3.3.4 discusses several ancillary functions required to support ACDS system management adequately, but which do not appropriately fall in one of the three functional areas listed above.

2.3.3.1 Program Management

Program management refers to the function which supports the system manager in the execution of his managerial tasks. It contains:

- 1) Budget and resource planning
- 2) Cost analysis and estimation
- 3) Effectiveness studies
- 4) Scheduling
- 5) Model management

Budget and Resource Planning - This activity maintains liaison within the Navy and external to the Navy on all matters pertaining to system, subsystem, and R&D budgets.

It is necessary to maintain an integrated knowledge of the various budgets which are affected by and which affect the availability of funds for the implementation of various improving increments proposed for addition to the evolving ACDS.

One of the outstanding managerial problems in the evolutionary implementation of a large system arises from the fact that instead of one budget for the entire system and a cutoff date for the system and the budget, the system continues to be evolved over many years. The budgets involved are not for one large system, but are small budgets for small improvements. These improvements represent the evolutionary increments to various types of systems. Therefore, the project management activity must maintain cognizance not of one budget, but of perhaps as many as one hundred. To do this requires a separate budget and resource planning activity.

Cost Analysis and Estimation - This maintains technical knowledge and liaison in cost analysis techniques, not only liaison within the Navy but also with the other services and with DOD. It provides cost analysis support to the project management office as required. The evolutionary process, necessary though it is, provides an additional managerial burden, since numerous cost analyses and estimates must be made to consider the impact of all proposed improvements on existing command data systems.

Effectiveness Studies - These specialize in the development and application of effectiveness measurement tools appropriate to command data systems, and to ACDS specifically. In addition, they maintain liaison with like groups in the other services and at DOD level.

The expertise assembled by these personnel is particularly valuable during the adjudication of roles and missions conflicts within the Navy, and with other services, when these conflicts involve measuring the effectiveness of command data systems.

Scheduling - This activity monitors all naval schedules which affect the implementation of ACDS. These include such things as delivery schedules of contractors, class graduation dates of naval training facilities, and availability schedules and production schedules of various scarce resources, such as shipyards, firing ranges and computer time. Many of these schedules are developed originally in other facilities and controlled by them. However, the necessity to make rapid and binding managerial decisions demands that these schedules also be maintained and updated in a central location where they are available to the system manager.

Model Management - Model management is that part of the system manager's authority responsible for the implementation of the various increments of improvement to ACDS. This authority and responsibility contains two important functions; configuration determination, and model implementation.

The installation of evolutionary improvements in system capability can be represented by a series of small steps in improvement rather than a long smooth curve showing a gradual improvement over a period of time. Each of these steps represents an instant in time in which some numbered fleet or some set of ships is provided with a significant increase in command system capability.

Each of these improvements in system capability may be thought of as a new "model" of the system. This concept is required to simplify recordkeeping, the transmission of technological ideas, and the managerial control and monitoring of the implementation process. Various improvements due to reach the field at approximately the same time are grouped together conceptually, operationally, and from an equipment standpoint. They form a "model". This model may then be discussed, monitored, and implemented, and provide military management with an improved capability to control the evolution of ACDS.

The first function of model management is to determine the configuration of each model. For instance, some data links and computing modules, together with display consoles might represent Model X. This capability is carefully analyzed to establish that, as an increment of capability, it will remain compatible with the balance of the system.* It is also carefully examined for operational usefulness and financial feasibility.

The second function of model management is monitoring and controlling the implementation process. The model management activity may have, at a given time, individual model managers for as many as three or four models, with still another set of model management personnel studying and planning the configuration of future models of system improvement.

The evolutionary implementation process requires the use of the model concept to make it feasible to apply operational and system analysis to some tangible and fixed increment of system capability. It also allows appropriate managerial control over the implementation of that increment. Most important, it permits the design and control of a specific increment to meet a specific threat or requirement.

2.3.3.2 Operations Analysis and System Design

This function maintains a continuing knowledge of all ACDS analysis and design studies being performed by naval activities and support contractors. This is its minimum role. The maximum role of this function is to perform, within the system management

* Compatibility is a relative thing. Some planned incompatibilities are occasionally introduced to accommodate advanced hardware, software, operational doctrine, etc. Compatibility really means "as compatible as possible, for a given set of circumstances and objectives."

activity, all the ACDS operational analysis and system design tasks on a continuing basis. In its minimum role, this function represents ACDS system management to the naval agency or contractor performing the tasks.

The organizational chart (Figure 2-2) shows similar names for some areas under operations analysis and system design, and under technical support. In those areas, most of the manning is in technical support. The similar area in analysis and design consists of a senior analyst experienced in that area supported by a few junior analysts. In some instances, this team is responsible for more than one analysis and design area. Technical personnel are borrowed from the appropriate area of the technical support function.

The areas which make up operations analysis and system design are:

- 1) System documentation
- 2) Operating environment and command requirements
- 3) Activities and procedures
- 4) Man-machine interface
- 5) Equipment
- 6) Computer programs
- 7) Training
- 8) Simulation and modeling

System Documentation - Obtains, controls and distributes the system planning documentation such as GOR, SOR, and Command Directives. It also maintains all preliminary documentation such as proposed hardware specifications, etc. for reference by all internal and external analysts and designers.

Operating Environment and Command Requirements - Obtains and distributes all data on the changing threat, new doctrine, new tactics and techniques, and the latest requirements of line commanders. It translates this data into functional requirements to initiate the analysis and design cycle.

Activities and Procedures - Specializes in translating functional requirements into activities and procedures, in defining exactly how the various functions are performed by the system or system increment.

Man-Machine Interface - Specializes in the human engineering aspects of system design. Coordinates and controls such things as display make-up, switch layout and operator task loading. Assists in deciding how much of each task is performed by operating personnel.

Equipment - Specializes in determining what and how much equipment of any type is used for each new task or increment. Receives strong technical support from the technical support function in matters of detailed hardware capability.

Computer Programs - Specializes in determining the amounts and natures of the tasks performed by various ACDS computer programs.

Training - Monitors all system design activities to coordinate training requirements and information into the original design considerations. They cooperate with the man-machine interface group to examine the problems of operator selection and training.

Simulation and Modeling - Provides support to the other areas of the ACDS system management activity in matters of simulation and modeling. Also coordinates ACDS modeling and simulation studies throughout the naval establishment and among supporting contractors to disseminate results and help prevent duplication.

In summary, the operations analysis and system design activity provides analyses, design evaluations and designs to support system management, by drawing from each function, the specialists required to execute this work. The offices also monitor similar work in other organizations which is chargeable to ACDS.

2.3.3.3 Technical Support

The cornerstone of the evolutionary implementation of ACDS is the forecasting, evaluation, and operational deployment of increases in technological and operational capability. Supervising the implementation of a system such as ACDS is a difficult task. The managerial team requires technical support of the highest caliber to make appropriate decisions.

Figure 2- 2 shows a technical support function reporting to the ACDS project management office. The seven areas of technical support have several common managerial functions which are discussed in subsequent paragraphs. Their areas of technical interest are:

Command Post - Is concerned with the location of command posts within ships, the coordination of information regarding personnel required at the different echelons of command posts, space limitations in types of hulls, and the types of display material required for each of the operational and command positions within each of the command posts.

Computer Programming - Maintains cognizance of the computer programming within the system and communication with research and development activities in the computer programming field.

Displays - Maintains current technical information on all types of displays; both individual and group displays.

Computing Machinery - Maintains cognizance of all computing machines in the system as well as their input/output and specialized storage devices.

Sensors - Maintains technical cognizance of the work being done on those sensors which are of direct interest to ACDS.

Communication - Maintains cognizance over technical matters concerned with communication techniques and equipment.

System Training - Monitors the equipment and techniques available for ACDS system and subsystem training.

It is not suggested that these areas should be little project offices or control points in their own right, but simply that they maintain complete competence in their individual technical areas. Then they may advise the ACDS system manager about solutions to problems which fall within their technical interests.

The type of support which each gives to the system manager are presented below.

Prevention of Technical Surprise - One of the most important functions of technical support is to monitor domestic and foreign activities in specific technical areas to prevent ACDS from being "surprised" technically or managerially. These specialists monitor the operation of the system in being, its current and projected operational environment, and the state-of-the-art in research and development. They assist the system manager in making decisions on whether or not new technological capabilities should be included. These support areas allow suitable countermeasures and tactics to be included in future revisions or models either in being or in a planning stage.

A number of critical technical areas such as high-speed crypto machinery, new data links, improved computer storage, group display devices, etc. may produce technological breakthrough. The continuous monitoring of R&D progress in these technical areas would allow ACDS system management to provide for integration of these new capabilities with the least disturbance to the existing system.

Projection and Analysis of Possible Technical Difficulties - The continuous monitoring of important technological areas allows technical support to project and analyze possible difficulties within ACDS and at the interface between ACDS and adjacent systems or subsystems. It is important that ACDS implementation management be advised of possible conflicts between pieces of equipment or between operating concepts and equipment, etc. While the technical support function may not be called upon to solve these potential conflicts, this function must advise system management of the possible existence of conflicts with as much lead time as possible.

Monitor and Develop Technical and Operational Concepts - This function maintains cognizance and performs occasional simulated testing of the various operational concepts being developed for the employment of ACDS or its various components. This function insures continuous smooth development of the system through the proper technical and operational employment of its new increments as they are added. Increments of capability may be added to the system through modest changes in operational techniques, and new operational techniques must be developed in advance of field deployment of new equipment and computer program capabilities.

Approve and Recommend System and Equipment Changes - This function evaluates the impact of proposed system changes. Small changes in operational equipment can be made to increase component efficiency. At the same time, they may actually interfere with the system or with other equipment within the system. Since engineering change proposals are constantly being prepared for all types of naval equipment, there must be some function which screens these proposals for possible detrimental impact on ACDS. By maintaining constant technical cognizance of their various specialized areas and detailed knowledge of ACDS, the technical support staff can screen and evaluate change proposals for their possible impact on the operation and function of ACDS.

Technical Support for Effectiveness and Cost Studies - The technical specialists provide support for evaluation and cost studies frequently made by the program management activity. Since these specialists maintain up-to-date information about their own technical areas and about the operational system, they provide the most readily available support for management studies in costing and effectiveness.

2.3.3.4 Other Implementation Management Functions

Reference to Figure 2-2 shows several additional system management functions. These functions must be performed in addition to those just described to provide high efficiency of operation in the implementation management of ACDS.

Development Center Liaison - There are several naval development testing activities involved with subsystems of ACDS. System management personnel must maintain close liaison with these test and development centers. This activity is shown separately from that of technical support and program management since its primary activity is to communicate with external agencies rather than perform work internal to the project management office.

Design Change Control - This is a small activity concerned with scheduling and coordinating engineering change and system change proposals. It is solely an administrative function but it is required so that technical support functions are not overcome by the administration and scheduling of the inevitable large numbers of design change proposals.

Documentation Control - An important part of any system is the supporting documentation which allows using commands to employ the full capability of the system. With large numbers of subsystems and component equipments likely to be part of ACDS, an administrative function ensures that documentation produced to support the system is of adequate quality and is distributed on time to the using commands. This administrative function does not produce the documents, but controls and coordinates production by other naval agencies and by civilian contractors. This function is particularly important due to the incrementally changing nature of ACDS.

Plans and Studies - This activity produces the long range planning necessary to support the project management office and it provides special studies and briefings on various problem areas. Separating this function from program management and technical support frees those two activities of the high-speed, high-priority management perturbations which are usual in management of large scale system implementation.

This activity provides model management as well as system management with the ability to answer involved technical and managerial questions on a short term basis during implementation of the system. Questions such as "What would happen if we changed "blank"?" must be answered accurately and rapidly to provide management with worthwhile support information. This activity is also necessary to provide briefing materials and special studies for presentation at higher echelons within the Navy and DOD.

Liaison to Operational Units - This activity maintains field liaison directly with line commanders of all echelons to supplement information flow regarding command problems and requirements as well as to assist in the installation of new increments of capability.

Naval Support - This support is provided to the ACDS system manager by all Navy and Marine Corps organizations concerned with ACDS, ACDS components, or ACDS field operation. In return, the ACDS system manager provides these organizations with timely advice and management information.

Scientific Advisory Committee - This committee allows ACDS project management to tap the intellectual and scientific resources of the Navy and industry to question and test advanced proposals and concepts. This committee meets infrequently but on an ad hoc basis to ensure that the latest and most advanced scientific and technical information are available to the ACDS system manager.

Contractor Support - This support is partially provided by those contractors producing or installing ACDS components. It is advisable to provide ongoing contractor support in the areas of systems analysis, system design, hardware and software design and general system management during the initial phases of ACDS implementation. During this time system management is getting oriented and building its capability. Concurrently, it is being asked to make long-range plans vital to the program. An ongoing technical support contractor operating under a hardware exclusion clause can provide important assistance to system management.

2.3.4 Discussion

The widespread impact and the evolutionary nature of ACDS development present difficulties to the organization which manages its implementation. Many of these difficulties are technological and are concerned with the different kinds of equipment and the operational implications of changes to this equipment. It is a time-consuming and tedious task to remain aware of the technological developments which are of future benefit to ACDS. This capability must be available on a day-to-day basis to the ACDS project management.

As complex and challenging as the technological problems are, the managerial and command problems are still more so. The function of an implementation management activity such as has been outlined here, is to ensure that maximum operational capability and combat readiness can be maintained by those using commands which have parts of ACDS deployed to them. This means that there is a continuing flow of important managerial and command decisions to be made on a daily and weekly basis over the entire implementation life of the system.

The system management office, through its direction and coordination, must ensure that the evolution of ACDS is scheduled to provide added capability at the right time to meet the threat, in view of the technology and projected operational doctrine. It must also make certain that procurement and O&M costs are properly scheduled and charged over the life of the system. This office must also be concerned with making best use of such scarce naval facilities as shipyards and training centers. As important as any other management consideration is that of making certain that the installation and testing of new subsystems and equipment does not cause unacceptable interruption to operational capabilities of existing systems.

A management task of the highest priority is to evaluate proposed changes and additions to ACDS from a cost and effectiveness point of view. ACDS system management deals with costing and effectiveness studies within its own house, and it must also be ready to support Navy and DOD discussions and studies involving cost and effectiveness comparisons among ACDS alternatives. This support must continue across the entire life of the system so that ACDS remains in effective competition for its share of the defense dollar.

ACDS implementation should be managed by an organization made up along the lines discussed in this section. Irrespective of where this organization is located within the Navy, it should have the following characteristics:

- 1) Cooperative - Every Navy and Marine Corps agency involved in the design, development, procurement, installation, implementation, and operation of ACDS is represented by skilled technical or managerial personnel assigned for substantial tours of duty to the system management office.
- 2) Authority - Sufficient for adequate managerial and technical decisions to be made.
- 3) Liaison - Maintained with all appropriate naval agencies, civilian contractors, with the other services, and with organizations of the Department of Defense.
- 4) Technological Capability - Maintained at a high level with regard to all of the component subsystems equipment and techniques to be employed within ACDS, by extensive liaison with industry, with research agencies, with using commands, and with contracting authorities within the Navy and the Department of Defense.

2.4 OPERATIONAL ANALYSIS AND SYSTEM DESIGN

2.4.1 General

The first phase of system implementation is that of operations analysis and system design. During this phase, the user of the system (in the case of ACDS, elements of OPNAV) and a team of analysis and design specialists build the foundation upon which the entire system implementation effort is based.

The output of operational analysis and system design is a set of formal and informal documentation which completely describes the system, its mission, its methods of operation, etc. After this phase is completed, detailed hardware and software design and production may begin. The process of operational analysis and system design is described in detail in the following sections, and the process is shown in Figures 2-3 through 2-6. These figures emphasize the high degree of interaction involved in such an effort between the study team and the user.

2.4.2 Inputs to Operational Analysis and System Design

The process of evolutionary implementation is described in Section 2.2. Figure 2-1 in that section depicts the cycle and its phases, and shows the importance of operational analysis and system design to evolutionary implementation. The inputs to operational analysis and system design are many and come from many sources. In general, they are information and formal documentation concerning:

- 1) Mission - Of both the system being planned, and the user of the system.
- 2) Technology - Of the U.S. and foreign powers which can aid or hinder the operation of the proposed system.
- 3) Threat - The threat or threats that the system must face.
- 4) Environment - The environment within which the system must operate. Friendly interfacing systems, foreign countermeasures, etc., as well as the physical and tactical environment.

- 5) Command Requirements - The specific requirements laid upon the system by its user.
- 6) Doctrine - Such formal operational doctrine as describes the activities proposed for the system or similar activities.
- 7) Liaison - Formal and informal liaison with field units of the user. For ACDS, this begins at CNO and extends through CINCPACFLT and CINCLANTFLT to the Commanding Officers of DLGs, DDs and DEs.

During operation analysis and system design these inputs are all integrated to produce the detailed description of the system or improvement best meeting the stated requirements and constraints.

2.4.3 General System Requirements Analysis

This step reviews and integrates the documentation of threat, requirements, environment, mission, doctrine and technology, as well as liaison, and produces a document called the System Operating Concept (SOR). This document completely describes the system at a gross level, and is the basis for detailed system design which follows. The System Operating Concept must be concurred upon by the user, and must have the complete confidence of all parties responsible for the system. Questions on tactical and strategic doctrine cannot be postponed beyond this point without substantial risk. Figure 2-3 shows the general flow of information and activity in the functional requirements analysis step.

2.4.3.1 Review Existing Documentation

The first activity of the operational analysis and system design phase is to review the official documentation which defines the operating requirements of the proposed system. This normally includes an SOR, a detailed statement of system mission, and some supporting documentation. These documents describe the parameters and the specific operational or performance characteristics of a system needed to fulfill a near-term operational requirement.

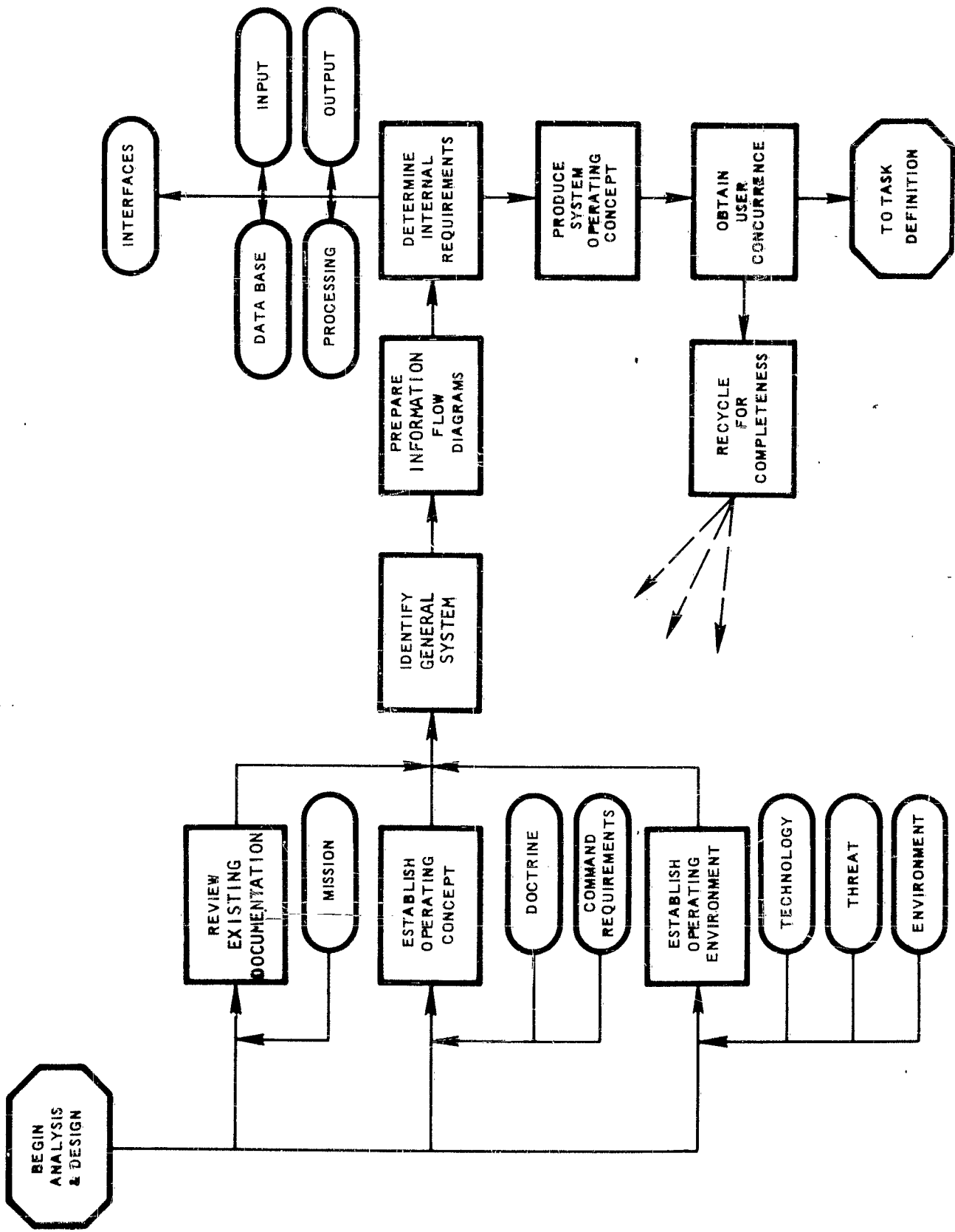


Figure 2-3. General System Requirements Analysis Step, Operational Analysis and System Design

After a period of intensive study and liaison, the study team reviews the SOR and other information with the contracting agency (NAVMAT) and with the operational units (OPNAV) for whom the system is being designed to:

- 1) Make sure that the study team is properly oriented to the scope and objectives of the system.
- 2) Assure that there are no barriers to effective communication, e.g., that all parties agree on definitions of key terms.
- 3) To establish for the study team's benefit the exact current status of all technical developments proposed for the system (or increment).

2.4.3.2 Establish Operating Concept

The second step of the functional requirements analysis is to establish in some detail an operating concept. The operating concept describes the manner in which the operational organization utilizes the proposed system in its field environment. The SOR reflects the formal requirements of the contracting agency. The operating concept reflects what it is that the user (and the line commander) really believes he wants. Establishing the operating concept is not a fixed or formal process but is one which is designed to bring to light the finer detail of the requirements of the user. For example, to establish the operating concept of the Navy's Integrated Operational Intelligence System, the study team examines current naval photographic interpretation doctrine, discusses PI requirements with working interpreters, obtains clear insight into combined future Navy and Marine Corps needs and intentions, and obtains a firm working knowledge of the immediate intelligence environment within which the system is to be used.

2.4.3.3 Establish Operating Environment

At the same time the study team establishes the operating concept, it also works with the using command to establish the operating environment. This process involves understanding the total tactical and/or strategic environment, its relation to the missions and objectives of the using force, and the potential interrelations with other

agencies. It also requires a detailed appreciation of the types of personnel to be assigned, their specialty areas and level of training, and the anticipated workloads under various modes of operation. The requirements for operation with a partial complement of hardware must also be determined.

At the end of this step, the study team has a thorough knowledge of the formal requirements of the system, how the system is to be used in operation, and the environment within which this operation must take place. While these first three steps of functional requirements analysis may take place serially or in parallel, they must all be complete before the next step is taken.

2.4.3.4 Identify General System Requirements

The General System Requirements of a system are identified as a result of integrating the operating requirements, the operating concept, and the operating environment with continuous input from both the contracting agency and the operational command. Identifying these requirements is the first part of breaking out into tangible and comprehensible pieces, those things which must be done for the system to perform according to its particular requirements and concepts.

For instance, a command data system provides command support and control for naval unit commanders in the performance of their operational tasks. To support this requirement, the Command Data System must perform a number of technical functions. One technical function is that the system must maintain and update various types of files. Another technical function is that the system must provide operators with a capability to retrieve various sorts of information upon request from a console. These data must be displayed to the commander and his staff, etc.

The first step in identifying the general system requirements is to isolate and define the operational tasks which must be supported by the system or increment of improvement in question. The second step is to designate the technical functions which are required to perform these tasks. A good example of these two steps is shown in Section 6 of Volume II of this report, although more detail is normally required when electronic data processing support is used extensively.

2.4.3.5 Prepare Information Flow Diagrams

As the study team is identifying the general system requirements, it also begins to prepare information flow diagrams. These are flow charts which show the logical relationship of each of the technical functions of the system to the system as a whole. There may be one set of charts for the entire system, or there may be one set of charts for each operator in the system. The precise manner in which the material is presented is much less important than the logical accuracy and completeness of the material. Flow charts are the usual means of presentation because of the effective manner in which such charts can convey the complex interactions in a system.

2.4.3.6 Determine Internal Requirements

The next part of general system requirements analysis is to determine the internal requirements of the system; that is, to determine the input, output, data processing and data base requirements which follow from the information flow diagram, system operating concept, the operating concept and the operating environment previously established.

In ACDS, for instance, the amount of data inserted into the system by keyboard action and that received by data link must be estimated. In addition, the team must establish the number of files to be moved into and out of bulk storage, the number and type of displays to be generated, the upper bounds on data base size, etc. Initial approximations are made of the amount of hard copy to be in the active index, and first approximations are made of the number and types of file entries along with the frequency of their updating and recall.

Internal processing requirements and data base requirements cannot be quite as accurately fixed at this point as can the input/output requirements. Processing requirements may be thought of as a detailed explanation of the logical relationships shown in the functional flow diagrams. Note that at this point, no assignment of individual processing task has been made to either man or machine. The statement of internal requirements simply indicates in some detail, what processing must take place within the system, without regard to how it is done.

Data base requirements are of about the same level of abstraction as processing requirements at this time. Files, records, and items are not normally designed at this time, but the requirements which they fulfill must be shown in substantial detail.

2.4.3.7 Produce System Operating Concept

The primary output of general system requirements analysis is the system operating concept. It represents the system as it is now understood by the study team, the contracting agency, and the using command. It includes an understanding of the technical and operational environment within which the system operates, the system interfaces with that environment and the data processing functions to be executed. It includes an explanation of the role of naval command data systems, and an explanation of closely related naval doctrine. The system operating concept is the basis upon which all system design work is based; hence, it must be completely agreed upon by all parties concerned.

2.4.4 Processing Task Definition (Figure 2-4)

This step of operational analysis and system design has as inputs the concurred upon system operating concept and the other documents generated during general system requirements analysis. It then fractions these functions into processing tasks. Following this the processing tasks are assigned to men and machines, and divided into steps.

This step requires some additional inputs. They are:

- 1) Equipment capabilities description.
- 2) Manual capabilities criteria.
- 3) Operator authority criteria.

These are each described in Section 2.4.4.2.

2.4.4.1 Divide Functions Into Processing Tasks

The purpose of this part of operational analysis and system design is to further subdivide the system technical functions into processing tasks and steps so that the

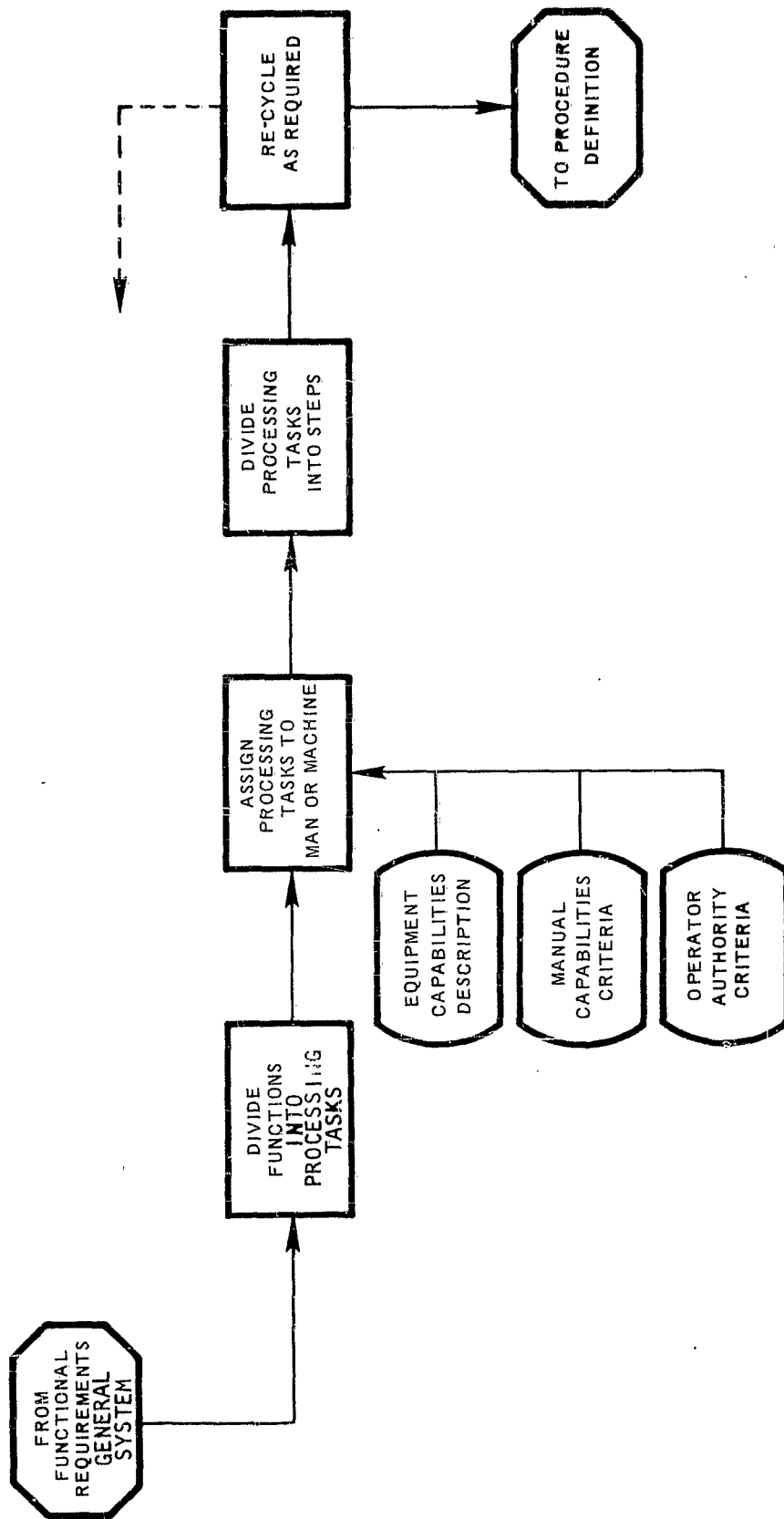


Figure 2-4. Task Definition Step; Operational Analysis of System Design

various portions of the work may be appropriately assigned to men or equipment or to combinations of the two. For instance, in performing the operational task of "strike planning", a technical function of "select penetration routes" contains a processing task "search previous reconnaissance data". This, in turn, contains a step of "display known air defense installations".

All of the various missions of a CDS must be subdivided into processing tasks of such a size that they can be appropriately assigned to either men or machines for their execution in the proposed system.

2.4.4.2 Assign Processing Tasks to Men or Machines

In this part of the processing task definition, the previously defined tasks are divided between men and machines for their performance in the operational system. Required inputs for this phase are a detailed set of capabilities of the equipment available for the system, and a detailed set of criteria concerning the capabilities of the operators postulated for the various positions in the system.

For systems which are developed in an evolutionary fashion, some equipment is already in the field and must be used. It is particularly important that this equipment be accurately described inasmuch as the processing and operational limitations of this hardware controls many of the assignments of tasks to either the men or the equipment.

An additional input required at this particular point is a set of criteria which is developed by the study team in cooperation with the using command. These criteria represent the user's specification as to the authority of the operators who man the various stations in the proposed system. This is the place in which the user specifies which decisions must be operator decisions due to their sensitivity or to the human judgment required.

Using the equipment capabilities description, the manual capabilities criteria and operator authority criteria, the system planner assigns tasks to men or machines. Often this cannot be done without further subdivision of tasks to take account of a need for close man-machine interaction. Thus, certain selected processing tasks must be further subdivided into steps. For example, "request file location of information X" or "compute present remaining combat range of the vessel specified in

vessel switches". When these tasks are further subdivided into steps, and the individual steps assigned to man or machine, the design may progress to the next step.

This close relationship of man and machine is called "symbiotic" by analogy to the biological phenomenon of symbiosis.

2.4.4.3 Divide Processing Tasks Into Steps

When the symbiotic tasks have been assigned to operators and machines, the balance of the processing tasks previously assigned to man or machine may be factored into their component steps. At the end of this activity, all the processing tasks of the system, whether they are performed by the operators or by equipment, are factored into their component steps. This is the finest grain of detail which is required for the definition of the various tasks prior to the generation of technical specifications and SOP's.

At the end of the task definition step, the design results are checked to insure that the data developed thus far meets the requirements shown in the system operating concept and defined logically by the information flow diagrams. It is occasionally necessary to reassign the tasks or to redivide them into steps as a result of having made this logical check. The next step in operational analysis and system design is the procedure definition step.

2.4.5 Procedure Definition (Figure 2-5)

In this step, the study team concentrates upon how the various processing tasks are linked together to perform the operational tasks as required by the SOR and more detailed statements of mission. Up to this point, the analysis and design effort has concentrated upon how each small processing task and step is to be performed. Now, the concentration shifts to how these small steps are combined and controlled to solve operational problems by performing technical functions.

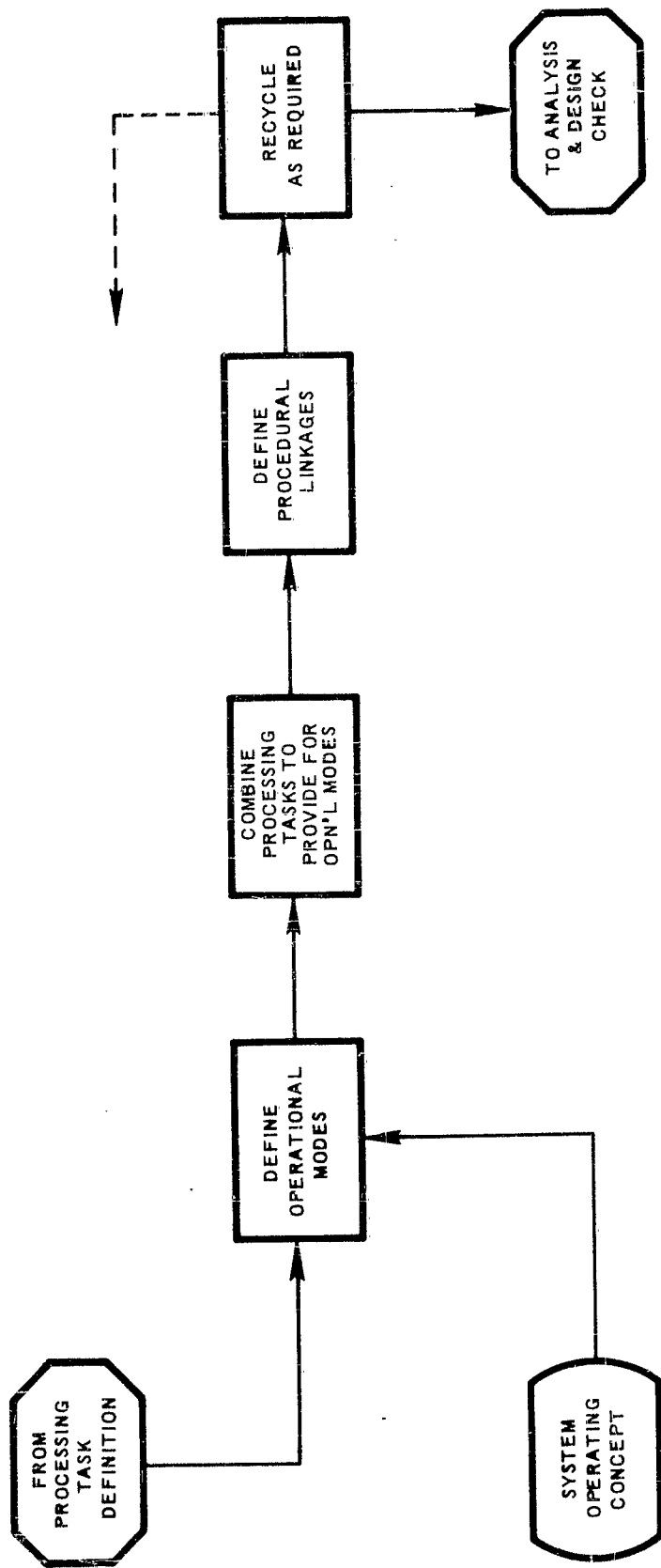


Figure 2-5. Procedure Definition Step; Operational Analysis and System Design

2.4.5.1 Define Operational Modes

The purpose of this part of operational analysis is to establish the rules by which the processing tasks and steps are employed to accomplish the mission of the system.

It is necessary that the study team consider all of the possible operational modes of the system. For instance, in ACDS, we can envision that there are several different "normal" modes of operation: operation with the full complement of consoles (during periods of high alert or operation), operation with a few less than the full complement of consoles (during lower alerts), and operation with so few consoles that various operators must alternate using a single console to perform their operational functions (for instance, during painting, maintenance of retrofitting).

In addition to specifying in detail the various normal operational modes of the proposed system, the user and the design personnel must specify all the predictable abnormal modes of operation such as operation with severe communications failure, operation with the loss of one or more computing module, etc.

2.4.5.2 Combine Processing Tasks for Operational Modes.

The next step after defining the operational modes is to select the processing tasks required to perform the duties necessary in each of the various normal and abnormal operational modes. Various operational modes require various combinations of tasks to meet the functional requirements for each of the possible modes. It is necessary in this step to show in detail how many times each of the various tasks is performed or cycled, and to show the conceptual or information flow between these various tasks. During this step, and the following step, some sort of a flow diagram is normally generated as an informal design tool.

2.4.5.3 Define Procedural Linkages

In this step, the study team defines the logical and procedural connections which link the various processing tasks together to perform each of the functional requirements for each one of the operational modes. These logical linkages must be defined since they are the source material for those system designers who specify standard operating procedures, the contents of operational handbooks, and the

contents of computer program operational specifications. The processing tasks themselves are, of course, of great importance to system design, but it is commonly overlooked that the definition of how these tasks are used to accomplish the mission of the system is of no less importance.

2.4.6 Analysis and Design Check Step (Figure 2-6)

The purpose of the analysis and design check step is to insure that the analytical and design work done to this point defines a system which meets the requirements at hand. The inputs to this step are all of the documentation produced so far in operational analysis and system design, and the principal product is the concurred-upon preliminary operational system description.

2.4.6.1 Synthesize Operations and Test System

The first activity in the checking of operational analysis and system design is to bring together all the analysis and design information developed in the phase to this point. This includes all the inputs mentioned in Section 2.4.2.

- 1) Mission data.
- 2) Technology data.
- 3) Threat.
- 4) Environment.
- 5) Command requirements.
- 6) Doctrine data.
- 7) Operator authority criteria.
- 8) Manual capability criteria.
- 9) Equipment capabilities descriptions

In addition, all the material generated during the analysis and design is included, namely:

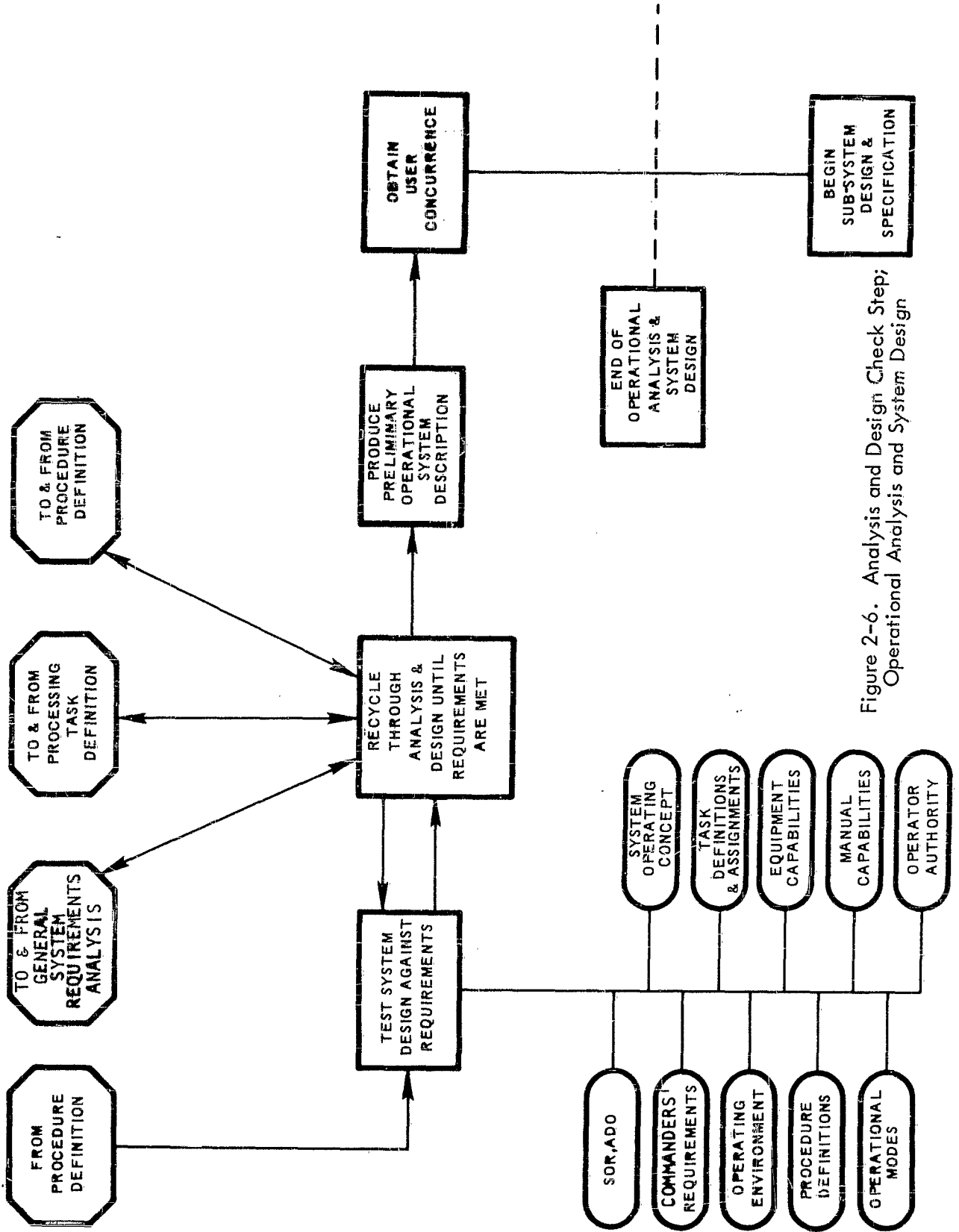


Figure 2-6. Analysis and Design Check Step; Operational Analysis and System Design

- 1) System operating concept.
- 2) General system requirements.
- 3) Information flow diagram.
- 4) Internal requirements.
- 5) Processing task definitions.
- 6) Processing step definitions.
- 7) Operational mode definitions.
- 8) Procedural linkage definitions.

All this material is assembled, and senior personnel from the user join with senior personnel of the operations analysis and design team to follow the information flow diagrams with the task definitions and procedure definitions at hand. Following only the rules and routes thus specifically defined, they attempt to execute simple technical functions using dummies of simplified real data bases. They thus attempt to solve simulated operational problems which are realistically part of each of the various possible operational modes.

It is, of course, impossible to test by hand all the possible routes through the proposed new system, but it is possible to test the most difficult routes, the most routes, and the routes that may most often be followed. It is often desirable to utilize computer simulation for this checking. The results of this checking are used to evaluate whether or not the system as it is now designated actually does meet the operational requirements of the user.

2.4.6.2 Adjustment and Concurrence

It is nearly always found that there are logical errors in the way that the system has been defined, or errors in the manner in which the tasks have been divided between equipment and operators. The solution of these problems requires some recycling through the previous steps of analysis and design. As soon as the appropriate recycling has taken place and the user, as well as the analysis and design team, is satisfied that the system performs as required, then an informal sign-off is made by the user and the contracting agency, and the analysis and design team prepares the preliminary operational system description based upon the material which has been generated up to this point and adjusted during the operational check step.

2.4.6.3 Preliminary Operational System Description

When the representatives of NAVMAT and CNO or CMC agree with the design and analysis team that the system design produced does meet the requirements of the SOR and its further elaboration, the preliminary operational system description is begun.

This is the final product of operational analysis and system design. It is a single comprehensive document stating authoritatively all the data gathered and developed. Each technical function is detailed, along with the processing tasks and steps, and procedure definitions which fulfill the requirements. All man-machine task assignments are detailed, and all hardware and software requirements are presented in rigorous detail. There is normally one section devoted to each of the input data areas mentioned in Sections 2.4.2 and 2.4.4.2. This single document is now the basic source for all system information since it represents the interpretive, analytic, design, and liaison efforts of the analysis and design team.

The title contains "preliminary" at this stage since hardware and software subsystem design have not been completed. This, coupled with the passage of time, leads to modifications of the preliminary operational system description. The operational system description describes the system as it stands completed. The preliminary operational system description is the data source until then.

2.4.7 Preparation of Hardware and Software Subsystem Designs

The preliminary operational system description is used by the various hardware and software contractors to prepare their bids. After the contracting tasks have been awarded, the preliminary operational system description is used to prepare the hardware and software specifications for all of the components and subsystems.

Normally, changes will be detected in hardware and software states-of-the-art, as well as in system mission, environment and operational doctrine. It is desirable to maintain a continuing operational analysis and system design activity, and also to monitor the development of technology state-of-the-art.

2.4.8 Discussion

The impact of evolutionary implementation upon the processes of operations analysis and system design is simple to describe, but pervasive in its effect. When working with an evolutionary system, operation analysis and system design are constantly being pursued at two levels.

The first level is that devoted to the system. Since an evolutionary system is really never completed, it is always being subjected to some analytic and design effort. This effort is devoted to finding those weaknesses which the advancing state-of-the-art may now be profitably used to remedy, and to the incorporation of new missions, doctrine, tactics, and commanders' requirements.

The second level of continuous effort is that devoted to the component or subsystem. This activity analyzes the possible applicability of newly developed equipment to the evolution of the existing system.

While the change in analysis and design activity brought about by the evolutionary development of some large system from existing capability is easy to describe, it is quite different from that required for the large "one-shot" systems which have been implemented in the past. It is essential to note, however, that the procedures followed in operational analysis and system design are the same, regardless of which approach to system implementation is used.

The analysis and design team must continue to examine hardware and software state-of-the-art and make projections as to what may reasonably be included in the next evolutionary changes added to the system. This is a continuous process and one which involves some risk. Some projected improvements never materialize--others never planned for show great short-range promise. One of the difficult tasks of system management is to monitor hardware and software progress and the changing environment. Making the best match between requirements and resources is never easy. In some instances severe risks are justified. In other instances, the delivery date and performance are so highly critical that only "sure-fire" approaches are warranted. In any event, system management cannot make these evaluations alone. They must make these decisions and recommendations in conjunction with CNO and CMC who have the ultimate responsibility for operational readiness.

2.5 HARDWARE DESIGN AND PRODUCTION

2.5.1 Introduction

The design and production of hardware for command data systems is relatively unimportant in some situations. In others, it is the essence of the total system design. Generally, hardware design is the dominant consideration in control systems; whereas software is the quintessence of high level command systems. Since tactical command data systems may be concerned with the control of weapons as well as the command of forces, hardware design is a very important consideration.

Hardware design like software design, is highly dependent upon the system being developed, and there is no such thing as a "typical" system. There are elements of system design, however, that are recurrent, and these are singled out and discussed in some detail. Hardware production may go on concurrently with the hardware design, or it may follow system design and specification. It may consist of a prototype system or a limited production item. It may include the ultimate production of hundreds or thousands of end items.

Subsequent sections center around the design of a system that results in prototype hardware and touches upon some of the necessary phases in preparing for further production. These phases are not all required for each piece of hardware developed, nor are they an exhaustive list of all possible design considerations. However, they are representative of the major hardware design considerations required for ACDS.

Section 2.5.2 discusses design considerations, and Section 2.5.3 discusses production considerations.

2.5.2 Phases of the Hardware Design Cycle

This section presents the hardware design cycle in six phases. In the actual design of ACDS hardware, the precise discussions presented here vary with the pieces of equipment involved. The six phases of the hardware design cycle are:

- 1) Initiation,
- 2) Organization,
- 3) Preliminary design,

- 4) Principal design,
- 5) Prototype construction,
- 6) Test, train, and evaluate.

Each phase is discussed in one of the following sub-sections. Figure 2-7 shows the time spans to be expected for each hardware design phase. Figure 2-8 shows the general information flow of hardware design.

<u>Phase</u>	<u>Name</u>	<u>Time Involved</u>
1	Initiation	1 day to 1 month
2	Organization	2 weeks to 3 months
3	Preliminary Design	2 months to 2 years
4	Principal Design	1 year to 10 years
5	Prototype Construction	6 months to 2 years
6	Test, Train, Evaluate	6 months to 2 years

Figure 2-7 Time Spans For Hardware Design Phases

2.5.2.1 Phase 1 - Initiation

Military systems are designed to fulfill an operational requirement as stated by one or more military organizations. Within the U.S. Navy, this is generally a General Operational Requirement (GOR) or a Tentative Specific Operational Requirement (TSOR) issued by the Chief of Naval Operations, or the Commandant of the Marine Corps. This, in time, may result in a Proposed Technical Approach (PTA) supplied by one of the Bureaus, Laboratories, or other technical agencies of the Navy.

A PTA may be generated internally by one of the Navy's "in house" organizations, or may result from a study effort such as ANTACCS. After a PTA has been accepted and approved, a Specific Operational Requirement (SOR) may be issued by CNO or CMC which leads into the preparation of a Technical Development Plan (TDP). Like the PTA, the TDP may be generated internally by a naval "in house" organization, or outside help may be required. It is at this point that many of the hardware possibilities stated in the PTA are firmed up, and the method of system development is presented. A number of decisions are then made such as, the method of contracting

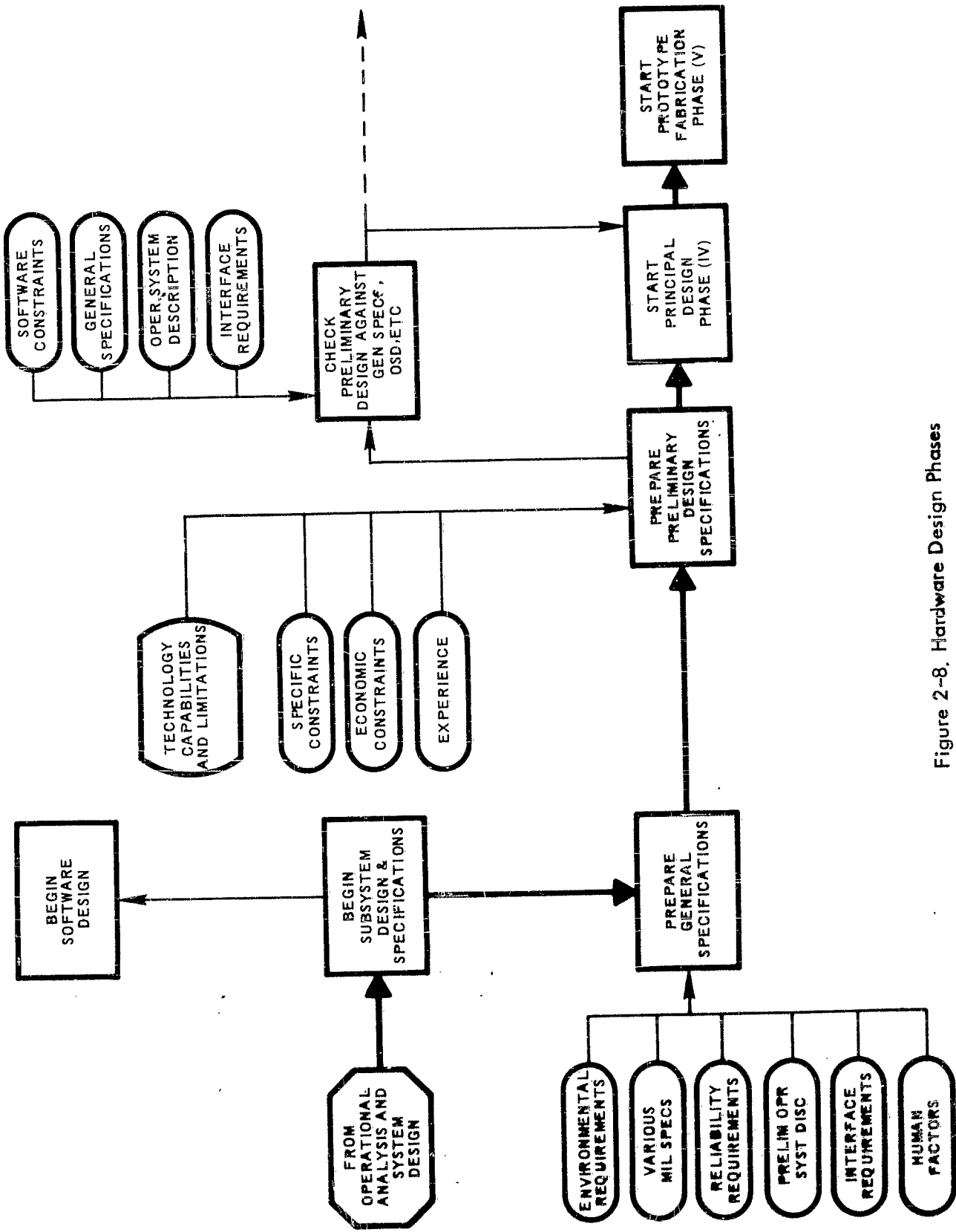


Figure 2-8. Hardware Design Phases

(CPFF, Fixed, Price, or CPIF). The result of the TDP may have a great bearing upon the hardware development and it must include reasonable estimates of technical feasibility.

After the TDP is approved by DDR&E the project can be released for Engineering Development.* Part of this release is likely to take the form of a set of specifications that is issued to industry to bid on the system either as a whole or in parts. After selection of the successful bidder and award of the contract, the serious hardware design commences. Up to this point, the details have been under the complete control of the Navy. From here on, most of the details of design are up to the system designer who is more or less free to do as he pleases at the detail level as long as he stays within the gross constraints of the specifications.

System design is not like a jigsaw puzzle with only one unique solution. It is rather like devising the "best" way to go from Washington to Los Angeles which can have many solutions depending upon the definition of "best". The fastest route may not be the cheapest, and neither is likely to be the most scenic. Making comparisons of this sort, weighing the advantages, and making the trade-offs and arriving at the "best" solution to the problem is the essence of engineering. In system design, the overall parameters are often difficult if not impossible to determine. Before hardware can be designed and produced, however, the parameters characterizing the particular piece of equipment must be known and explicitly stated.

The hardware design is thus initiated by a series of rather formal steps which lead to the award of a contract for the development of a system, or for the development of certain pieces of equipment. In either event, the product must be designed according to a rather general specification.

If the specification is for a system, further design and more detailed specification is required before hardware is designed and produced. The steps of a system design leading to the creation of a detailed hardware specification were discussed in the previous section.

*This process is the subject of Section 2.7 of this volume.

Hardware design and production differ from general system analysis and design in several important respects. A discussion of the organization required for hardware design and production highlights these differences.

2.5.2.2 Phase II-Organization

Though often overlooked, the organization of the "team" is a very important consideration in developing command data system hardware. The organization is important for the reason that it is the nucleus of command and control system for the hardware developer. It establishes the lines of communication between the various key elements, establishes reporting procedures, and provides for necessary checks and balances. Like a military command and control system, it must provide for positive and effective, yet timely control of all key elements, while remaining flexible and responsive to both external and internal pressures and changes.

During the period when the proposals are being prepared, sales engineering, the general manager and his staff, and others in the organization work together closely, and certain relationships exist that cease when the actual work on the program begins. The effort on the part of marketing decreases significantly, while the participation of the general manager and engineering shows a definite increase. Using the award of a contract as the point of departure, a crucial question presents itself to management: How will we manage this program? Even though the question may have been brought up and perhaps even resolved before, the award of the hardware contract requires immediate resolution of this question and implementation of a management plan.

The most common approach is to establish a program office headed up by a person designated as the program manager. Where, in the organization, this program office should be located however, is not so readily determined. Three commonly accepted spots for such program offices are:

- 1) In a staff capacity advising and acting for the general manager (Box A or B)
- 2) In a line position on an equivalent level with engineering, manufacturing, etc. (Box C or D)
- 3) As part of engineering

Regardless of location, the program office is responsible for coordinating the in-house effort, interfacing with the customer and upper management (often representing the customer's viewpoint rather than the company's) and exercising close scrutiny over expenditure of funds.

Smaller, less complex hardware developments tend to be managed under engineering, while larger more sophisticated programs are likely to be managed at a high level where the program manager has direct access to the general manager and may or may not have direct control over elements of engineering, manufacturing and other parts of the organization. Occasionally with major hardware development programs a separate division of the company will be set up with engineering, manufacturing, and other necessary functions as part of the division.

Before discussing some of the functions of the project office, a few words about quality control seem appropriate. The location of quality control responsibility and personnel seems to vary widely from company to company. The ultimate responsibility for the quality of a product rests with the general manager of the organization. Quite often, however, responsibility for this function is delegated either to engineering or manufacturing. Even though this delegation is a common practice, it is not always the best one. Engineering should be charged with providing a well-designed and engineered piece of hardware of high quality at minimum cost. Manufacturing is responsible for taking the engineering design and converting it into hardware meeting the engineering specifications at minimum cost. Each of these organizations needs to concern itself with the quality of the end product, but the final stamp of approval should come from outside both organizations.

A preferred approach is to place the responsibility for quality control where it really should be; responsible directly to the general manager. This can be either at a staff level or in a line position along with engineering and manufacturing. The latter has many advantages over the staff level organization, but either can provide a very workable solution. The interaction of quality control and other elements of the organization is covered in more detail in the later phases of hardware design and production.

Another point about the organization should be made before discussing equipment development. An industrial organization must have people to do a good job. The key men of any group undertaking an important job should be on hand before the job is started. It is often impossible to hire a crew after contract award. It is poor policy to fire everyone on contract completion. Management people have a great interest in maintaining a constant or gradually-changing work force. Such work force planning can avoid many training and indoctrination problems, and lessen the effect on general morale imposed by lay-off.

A stable organization is needed to support important programs. The building of such an organization is an important management responsibility.

2.5.2.3 Phase III Preliminary Design

The preliminary design phase is based upon the general specifications prepared after the completion of the operational analysis and system design. This is discussed in Section 2-4. The general specifications generally cover the gross hardware considerations such as environmental and reliability requirements as well as incorporating as many of the details of the preliminary operational description as is necessary.

General constraints affecting all the equipment are incorporated into the specification at this point. A general constraint might, for instance, require that all equipment be so fabricated that it initially can fit through a submarine hatch either as a whole or in pieces.

The technological state-of-the-art, coupled with the cost of implementing a technical approach is the major factor shaping the output of the preliminary design. Specific constraints such as maximum allowable voltage or time limitations on computations to be performed by the equipment also may play an important role here. One important input into the preliminary design phase is the matter of experience and judgment of the people involved in this phase. Too little experience being brought to bear is likely to result in a less than optimum design, while too much "narrow" or highly specialized experience may result in an overly complex solution to a simple problem or result in a very fine piece of equipment for doing many things that may not really be required.

Although there is no sharply defined point at which Phase III ends and Phase IV begins, the preliminary design becomes principal design when:

- 1) The general internal configuration for the equipment has been completed and specified.
- 2) Specific performance specifications have been prepared.
- 3) Interfaces with external equipments have been specified.
- 4) A schedule for the principal design effort has been prepared.
- 5) The proposed design approach has been checked against requirements and specifications to insure adequate compliance.
- 6) A quality control program has been generated.

At some point in time when most of these items have been covered, the principal design commences. Parts of the principal design can start before the preliminary design is complete. Interface information, for instance, may be very late in being specified.

Note also that as the preliminary design specifications are going over into the principal design process, they are being checked against the general specification, the operational system description, all interface requirements, and those constraints that may result from the software design effort. This is a continuing effort through the principal design and subsequent phases.

2.5.2.4 Phase IV - Principal Design

The principal design period is generally the longest of any of the phases in most hardware or system developments. It is the period when concepts are finalized and converted into detailed specifications. General specifications from the preliminary design phase are used as the basis for detailed and definitive subunit and component specifications. Unproved techniques are checked out with breadboards; unworkable ones are rejected. At the end of this phase, complete, definitive and workable specifications for the fabrication of prototype hardware are complete and construction can be started.

Breadboarding is an important and useful tool during this phase. Breadboarding is a useful adjunct to, but not a replacement for, simulation. A breadboard of one or of a few distributed logic elements, for example, is sufficient to demonstrate the feasibility of the device and the adequacy of the design, but simulation is necessary to determine how thousands of these would work together and to determine the best ways of tying them together logically. After this determination a breadboard can be used to test the method of interconnection, and modification to the basic design can be made if necessary. For example, simulation might indicate that the optimum number of mutual interconnections for each element is ten, while the initial breadboard design might be capable of driving no more than eight without modification. If considered desirable, based upon the simulation results, the breadboard might be modified if this is feasible; if not, a new simulation might be run using an upper limit of eight interconnections.

Other types of simulation also may play an important part during this phase. Equipment external to the hardware may have to be simulated due to nonavailability of the external equipment or impracticability of its use. Some typical examples are:

- 1) Simulating radar video for a radar data processor or display.
- 2) Simulating the output of a computer in the design of a computer-driven display or a computer peripheral device.
- 3) Simulation of peripheral equipment characteristics in the design of computers.
- 4) Simulation of RF interference in the design of communication equipment.
- 5) Simulation of environmental conditions in the development of all types of equipment.

Depending primarily upon the complexity of the hardware, parts of the prototype equipment may go into the construction phase before all the principal design is completed. Equipment component completion should be scheduled in a fashion that ensures completion at about the same time of all necessary subunits that go together to form a unit. Long lead time items, therefore, should start before shorter lead time items. Unfortunately, long lead time items are often the most difficult to design, and a definite effort on the part of the program manager to complete these designs first is

necessary. Care must be exercised to minimize changes that may be necessary to equipment that has been released to manufacturing and result from unknowns in the designs of later specified equipment.

It is during this phase that the key inputs to such management tasks as PERT are generated. This information concerning the key elements, bottlenecks, milestones and completion times, if not the most important output at this phase, is certainly one of the most useful to the manager and the user who is anxiously awaiting the completion of the program. A rough schedule probably exists at the beginning of this phase which will be refined and polished as the principal design progresses: Scheduling, PERT, critical path analysis, and other related techniques are covered elsewhere and are not discussed here. Most of these considerations are equally valid for small hardware developments or large and complex system developments, differing only in degree.

Even though the principal design phase may last longer than all other phases combined, it is relatively straightforward. It is almost exclusively engineering in the true sense of the word. There is a smattering of research in those areas touching upon new and unproved techniques, and also a hint of manufacturing as prototype devices are fabricated, but the principal design phase centers around good old-fashioned engineering. Toward the end of the principal design phase, interaction with manufacturing and quality control at this point must take the necessary steps to ensure that the engineering design satisfactorily meets the overall quality required of externally and internally generated specifications, and that the manufacture of the equipment does not degrade this design to an unsatisfactory level. The important thing is that a team effort is now necessary even though all members of the team do not appear to be working towards the same goal. The team captain is the program manager, quality control acting as referee, and close decisions being made by "top management".

When adequate specifications have been prepared by engineering, manufacturing has accepted them and agreed to fabricate the necessary hardware, and quality control is satisfied with the proposed approach and has approved the engineering acceptance test procedure, prototype construction starts.

2.5.2.5 Phase V - Prototype Construction

Depending upon many factors, prototype construction may be almost exclusively a manufacturing phase or a phase with a great deal of engineering and quality control monitoring. Choosing the fabrication of a small militarized general purpose computer as illustrative if not typical, of such a fabrication process, several of the salient features are discussed subsequently.

Up to this point, much of the design has been a process of breaking the system down into smaller and smaller elements, and defining these in some detail. What may have started as a large and complex tactical command and control system has been divided and subdivided into smaller and smaller bits and pieces until, at this point, specific components such as resistors, capacitors, and transistors must be considered. From this point, the gradual building back into the complete system must start. Much of this initial build up goes on in parallel and, as two or more elements that go together to form a unit are completed, they are joined and these, in turn, are joined with others, and on and on, until an equipment or subsystem is completed.

Even though it is impossible to break down the fabrication process into several distinct steps that are universally true for all situations, a gross breakdown can be made as follows:

- 1) Component assembly into "modules"
- 2) Module assembly into subunits
- 3) Subunit assembly into complete units.

The list could have another step for tying units together to form a complete system, but that aspect of system design is covered elsewhere. In each of the above steps the assembly may be a one of a kind fabrication, small (2 to 10) quantity fabrication or large quantity fabrication. Each step in the above process can easily be the subject of a very long and detailed dissertation. Consequently, the subsequent discussion is a digested and encapsulated coverage.

For the design and production of a militarized general purpose digital computer prototype, much of the component assembly stage is in the category of large quantity

production. Several of the modules making up the computer are identical. A basic flip-flop module, for example, is likely to be used in several subunits. The preliminary design indicates what the major subunits are, and the detailed design indicates how these subunits are made. During the detailed design phase, even though different groups may be designing different subunits, they should be required to use common circuits and modules in the design of their respective modules unless the task cannot be accomplished with the "standard" modules. There is, of necessity, much interplay between the various design groups during this period, and many of the parameters of the "one megacycle flip-flop" may change as the detailed design progresses. By the time the prototype construction gets underway, details of such modules are specified, and, in almost all cases, a breadboard of the module has been built and tested.

In a computer of the type we are discussing here, there may be well over a hundred flip-flop modules and a similar number of logic modules. Fabrication of such large quantities of identical modules warrants setting up assembly line techniques for fabrication and checkout of these modules. Other modules such as those making up parts of the power supplies and the clock are "one-of-a-kind" or very small quantity devices, and are treated as "custom built" elements. In fact, an entire subunit such as a clock may be handled this way with only limited testing until the subunit is completed.

The modules themselves vary considerably in complexity and may contain as few as a dozen components in some simple logic modules. In other modules, such as input/output amplifiers, there may be as many as 500 components. The number of components is highly dependent upon the overall design approach to the computer. Often, extra components are used to increase redundancy and improve reliability. In other cases, the addition of components may reduce the overall reliability significantly.

Testing is a continuing process. Components must be tested prior to being used in building up modules and subunits. These may be tested individually in the case of very critical items or as a batch using random sampling techniques. After the completion of each module, each must be tested. Then as subunits and groups of these are tied together, more tests are necessary. Finally, when the unit is completed, it is tested as an entity. At each stage of this process, a failure results in going back to a lower level for retesting to isolate failures. Although this is not always done, records of

failures and difficulties encountered should be maintained, compiled and analyzed to pinpoint recurring difficulties requiring redesign or modification of the unit.

Depending upon the organization and other factors, testing may be performed by engineering, manufacturing or quality control. Test specifications also may be generated by any of these three. There are too many factors that must be considered to state that one approach is significantly better or worse than another. In any event, however, final approval of the test procedures, and certification of satisfactory accomplishment of test requirements, must rest with quality control. This is not to say that tests set up or approved by quality control are irrevocably the final word. Quite often, these specifications are modified because of economic or scheduling considerations that outweigh the quality control requirements. For example, a requirement to operate the equipment for one hour at 50° C may be impossible to meet without costly and time consuming redesigning of parts of the equipment although the equipment works adequately at 45° C. Management confronted with the figures concerning the cost and time required to make the changes necessary may decide to modify the requirement to 45° C. Quality control would then be so directed. Many factors must, of course, be considered before making such a decision and often the decision may have to be referred to the ultimate customer.

In addition to in-house quality control measures, normally there is concurrent checking by government inspectors. Although there may be some duplication of effort here, in general, the government and the in-house inspections are complementary.

Returning to the module fabrication briefly before going on to the process of tying these together into subunits, we can now examine some of the details of the fabrication and testing of these units. For the purpose of this discussion, it is assumed that plug-in type printed circuit cards are used for the basic module. Even though it is recognized that such modules are not universally used, the trend in the past few years has been to use this type of module whenever possible in militarized electronics equipment. There are several advantages to this type of construction, the most important being the simplification of the maintenance of the equipment. Another important consideration is that of standardization.

Unfortunately, there is no standard printed circuit card configuration used throughout the electronics industry. Even within a given company, there is likely to be a large number of printed circuit board types. For a given hardware development, however, the normal practice is to have one standard printed circuit board type that is used for all plug-in modules. This brings up another matter, that of the meaning of module. When used here, it refers to any circuit assembly that is fabricated as a single unit. It is interesting to note the wide variation in the size, complexity and cost of modules used in the NTDS family of equipments.

One early criterion for selecting a module size was to keep the cost of each at a level where if it failed, it could be thrown away rather than repaired. The upper price for fulfilling this criterion was set at somewhere between \$25 and \$100. Although not all equipments developed in recent years have had modules so modestly priced, many have. The AN/USQ-20 computer has been quite successful in this respect. Even though this is the case, most cards that fail are still returned for repair and/or examination. Such repair and examination provides an exceptionally good basis for affecting a posteriori quantity control and pinpoints potential weaknesses.

Even though the use of printed circuit board makes the design of many circuits difficult to optimize because of layout and total available connector constraints, for the most part it is a definite asset to the design process since the circuit board provides a basic element to build from. The physical size, number of connector pins per board, maximum number of components that can be mounted on the board, and related considerations provide the circuit designer with a set of parameters within which his design must exist. These constraints sometimes force the designer to place part of a circuit on another board even though it might be desirable to have the complete circuit in a single unit. This is unfortunate but often necessary. Where possible, two or more of these circuit spillovers should be combined on a single board to reduce the total number of required boards.

The actual fabrication of a module from a printed circuit board and discrete components is usually performed by a single person following a step-by-step procedure, or by a series of persons each responsible for a specific set of steps. For breadboards and some prototype fabrication, the procedure may be to work directly from the circuit

diagram. Components may be soldered as they are put into the board, or they may be automatically soldered by a dip soldering machine after being manually inserted and trimmed.

After the board is completed and soldered, it is normally given a visual inspection to locate poor connections, cold solder joints, and other discrepancies. This inspection is mandatory for dip-soldered boards but may not be necessary if the board has been hand soldered.

After fabrication and inspection, the board is tested. The testing can take a variety of forms ranging from a fully automatic test where the card is plugged into a tester and the card type inserted into the tester and a good or bad indication is given, to a completely manual test. Due to the cost of setting up such a tester initially, limited production cards are likely to be tested either manually or semimanually with an engineer or technician performing the tests and determining the acceptability of the module.

Except for production of large quantities of equipment, the process of connecting groups of modules together to form subunits is predominately manually accomplished. Tests at this level are difficult because everything external to the subunit that interfaces with the subunit must be simulated for effective tests to be performed.

The wiring that interconnects the various modules making up a subunit may be contained in a rack that houses the subunit, or it may be contained in the cabinet housing all of the subunits. The most common practice in computer fabrication is to have all the wiring for all the cards contained in a single enclosure that has the racks for accepting all the plug-in boards. In fabricating a prototype, this wiring may proceed as the subunits are installed and checked, or it may be completely rewired. Breadboard and prototype wiring have a tendency to take on a rat's nest appearance, while production equipment is normally more orderly.

When dealing with the high speed circuits that are contained in present-day computers, a great deal of care must be exercised in the wiring due to the capacitance that the wiring may contribute to the circuit. A long run of wire connecting two circuits may cause them to perform radically different than when connected with a shorter wire. A good design takes this capacitance into consideration. In the transition from breadboard to prototype and from prototype to production, this may

become acutely important. As we progress into the integrated circuit era, high frequency effects, interconnection and subunit assembly take on even greater significance. Many of these effects are covered in the Technology Volume of this Report.

Most of the significant features of the unit assembly from subunits are the same as those of the subunit assembly, except that fewer simulations of external interaction are required. In the case of the unit assembly of a computer, this simulation consists only of peripheral devices which may exist and, therefore, do not require simulation. Testing of the completed unit is essentially that of demonstrating that all specified requirements on performance are met satisfactorily for the unit as a whole. Size, weight and environmental specification fulfillment must also be demonstrated. Many inadequacies may not show up until the entire unit is assembled and operated as a unit. To remedy some of these, a redesign of some modules or subunits may be necessary. Power supply problems often do not become obvious until this stage in the construction cycle.

A characteristic of this stage is the difficulty in pinpointing the source of difficulties. This is especially true for intermittent failures. For example, an occasional lost bit may be caused by a faulty memory, difficulties in the read-write amplifiers, a bad logic card, power supply noise, a defective component in any part of the computer, a cold solder joint, mutual interference in some of the wiring, or even a subtle intricacy in the program being used for checking out the computer. Since the difficulty is intermittent and cannot be repeated at will, trouble shooting is a nightmarish undertaking.

After these bugs are eliminated, the prototype is ready for final acceptance testing and evaluation. This aspect of a hardware development is covered in the next section along with the problem of training personnel to use and maintain the equipment.

2.5.2.6 Phase VI - Testing, Training and Evaluation

Testing has been covered in previous sections up to the "Moment of Truth", the final government acceptance test. Prior to this time, in-house and government-supervised inspections have been detailed, and are exhaustive and often more severe than those required for the final acceptance. One exception to this would be the case of long continuously operating tests which can be very expensive and in some cases have

derogatory effects upon the equipment. Another is that which involves destructive testing which is obviously not the thing to do with prototype equipment.

Specifications for final acceptance testing may be drawn up by the procuring agency or may be drafted by the contractor and approved by the procuring agency. In very complex developments an intermediate agency is often used to provide the acceptance test specification.

Waivers against specification requirements may be generated at any stage in development and, if approved, become part of the overall acceptance criteria. Quite often conditional waivers are granted on specific items in the specification if the overall equipment meets the general requirements. For example, some subunits may not meet environmental or RFI specifications individually but, when enclosed in the final cabinet, the equipment as a whole meets the requirements. Other conditional waivers may be such that deficiencies are corrected after delivery to the customer. This allows timely delivery of equipment that might have to be delayed if the deficiencies were corrected before delivery. Similarly, acceptance may be conditioned by an agreement to correct deficiencies after delivery.

In the case of large complex equipment, parts of the system may be accepted prior to final delivery. This allows government acceptance of sub-system equipment as it is completed and tested in-house. The completed system must still be accepted as a whole, but detailed tests on individual equipments need not be repeated. Another technique to minimize the amount of total final acceptance testing in larger system developments is to pick one group of equipment as representative and conduct extensive tests on this equipment.

Carefully controlled tests such as acceptance tests, unless they are carried on for prolonged periods of time, cannot be expected to fully check out the equipment. Many deficiencies do not, therefore, show up until the equipment is accepted and goes into the evaluation phase.

Hardware evaluation is the process of checking to see just how well the equipment performs the job it was originally intended to do. That is, it is checked out against the original operational requirements to see if it fulfills all or part of the fundamental objective. This evaluation should be conducted in an environment as close as possible

to that in which the equipment is ultimately to operate. Operating and maintenance personnel also should be "typical" of the ultimate users. A common mistake during the evaluation period for new electronic hardware is to select the best possible people available to do the testing and evaluation. This frequently results in equipment being given a good evaluation in areas where this is not warranted. This results in technically competent personnel making operational evaluations when they may not be qualified to do so. It also results in equipment being considered satisfactorily maintainable when in reality, this is true only if superior maintenance personnel are available. In a medium sized general purpose computer, effective trouble shooting requires a good working knowledge of the computer unless clear, concise, and detailed trouble shooting procedures are established. These must be coupled with a good diagnostic program for the computer. The adequacy of these procedures and programs must be evaluated by the calibre of personnel who will ultimately be maintaining the computer. A highly-skilled technician, with a great deal of experience with electronic equipment maintenance, and a knowledge of programming, who knows the computer organization quite well, can obviously do a satisfactory job of maintaining the equipment with minimum backup in the way of procedures and diagnostics, whereas a less competent, less skilled technician needs more assistance. Unfortunately, in the situations where most military equipment is used operationally, the technicians are most likely to be the latter than the former.

The evaluation period can last from a few weeks to several years. Since future production depends heavily upon the results of the evaluation, the period should be kept as short as possible consistent with performing a completely adequate evaluation. There is no substitute for a complete evaluation, but often late delivery coupled with other scheduling difficulties causes this period to be shortened. Only the user can weigh all the factors and determine how long and how detailed the evaluation must be.

During the evaluation period, the contractor who built the equipment may have field service engineering assistance available to assist in the maintenance and upkeep of the equipment. Where it is at all possible, such contractor personnel should be kept on call for technical assistance. They should always work under the close supervision of the assigned evaluation personnel; otherwise, the evaluation can develop into a contractor evaluation of his own equipment.

Other than ensuring that the delivered equipment is technically sound, maintainable and capable of being operated satisfactorily, this phase in the overall development must check the validity of many assumptions built into the equipment. For example, interceptor control programs contain many assumptions regarding the performance of various aircraft. These must be checked against actual aircraft performance. Interface assumptions (including the interface with humans) have to be checked against the outside world. Assumed noise characteristics must be checked against the noise that actually exists.

Training has been included in this discussion because it is a key ingredient in the successful testing and evaluation of the equipment of interest. Training should precede the evaluation phase and ideally is completed just as the evaluation phase commences. Much can be said for having both technical and operator personnel involved in the final acceptance tests. If they do nothing more than observe the tests, this facet of the program is very useful for the potential users for they can observe both operator techniques and technical procedures.

The level and length of training depend upon many factors but are generally greater for prototype equipment going into evaluation than for production equipment. If those trained for the first equipment are to be used for future instructors, the level of training is different also. If modules are to be repaired in the field, special techniques and test equipment may have to be developed.

Training is a seldom overlooked but often slighted aspect of system development. It is often scheduled too closely and for too short a period of time, and is hardly ever properly budgeted for. The difficulty here arises from the fact that the equipment development funds are completely divorced from training funds. Even if the funds have a common node, the responsibility for the two aspects of the program is divided. This is not an unsurmountable problem, it does require that cognizant and responsible personnel start planning for training early in the program, and remain flexible as the program evolves.

Except to say that training in the use of the equipment must not stop upon the completion of the initial training courses, this study does not discuss continuing training further. The importance of both types of training cannot be overemphasized. Proper planning for and conduct of training is essential.

2.5.3 Production Consideration

Most of the processes discussed in the previous section, which was slanted more toward the development of prototype hardware, apply to equipment in production and differ more in degree than in substance. No attempt is made here to discuss large quantity hardware production in any detail. The subject is too large and too complex. Some of the production considerations that relate to the prototype fabrication are covered in earlier sections.

There are two essential differences between prototype and production fabrication of equipment. The first of these is that the quantity is larger. The second is that this is the final version and changes cannot be allowed unless absolutely essential. A relatively simple change can cause major perturbations due to the large amount of equipment that may be affected as was mentioned in the discussion of module fabrication. When large quantities of similar or identical items are to be constructed, certain techniques for fabrication can be used that are not economically feasible with smaller quantities. Fully automated production lines seldom pay off unless the quantity of equipment produced is extremely large. Because of the initial implementation costs, the break-even point may be at 100,000 items or more.

The production approach adopted on any production item varies with the type of equipment, quantity of equipment, and the company. One element of production (as opposed to prototype construction) is the rigidity of the process and the close control over the changes and modifications. Testing is generally conducted frequently as an item progresses through the production lines with set routines for processing failures. Tests are routine but explicit throughout the progression; schedules are much more detailed and more closely adhered to although subject to changes. More people are involved with the equipment, but there is a larger percentage of specialists with few knowing and understanding the entire equipment. Testing, inspection and supervision by government representatives is quite similar to that for prototype equipment but tends to be more regimented. Costs are carefully monitored and reported. The overall status is carefully monitored, updated and reported.

In general, then, the atmosphere is one of intense activity, rigidity and of progress. As compared to prototype development, the progress of production equipment can be seen as it goes through the various stages leading to the final product. There is a great

deal of inertia at all stages in the production cycle which resists change in any form. Without this inertia, a production line would be little more than a custom workshop where every item is given special care and treatment.

In preparing production specifications for command and control hardware then it is necessary that these specifications be well thought out, specific, concise, and as detailed as possible. If production specifications cannot be provided in this detail, it may be too early for the item to be going into production. In cases where there are many unresolved questions, a pre-production model may be generated. This model, while not being specified in detail, provides the detail necessary for further production equipment.

Such a pre-production model may be highly R&D in nature, but is normally an engineering model that can be used for a variety of purposes after completion. Quite often, this model provides a pattern, or standard, against which production versions are compared. It can also provide a test base for testing portions of production equipment. It also provides an early piece of equipment for use in training.

2.5.4 Discussion

Hardware design and production is a complex and involved process. As in software generation, there are many ways to go about developing the required output, and no single approach is unique, nor can any one be singled out as being the best approach. There are many tools available for use by the hardware designer that have developed over the years. Despite the tools available and the extensive literature available on different aspects of this art, they are not substitutes for experience and creativity in developing advanced command and control system hardware.

The techniques for fabrication of electronic hardware seems likely to change drastically over the next few years due to integrated circuits technology expanding and becoming commonplace. Even though technology will be changing, the basic design procedures and techniques are not likely to change significantly since they are more or less independent of technology.

The evolutionary implementation of ACDS requires that many smaller hardware units and subsystems co-exist in varying stages of design and production. This imposes some slight additional strain upon the managements of the various ACDS hardware

contractors, and upon hardware procurement and test channels within the Navy. However, the fundamental nature of hardware design and production, as discussed here, integrates perfectly with the concept of ACDS evolutionary implementation.

2.6 SOFTWARE DESIGN AND PRODUCTION

2.6.1 General

The inherent flexibility of a system controlled through the use of a general purpose digital computer gives ACDS a distinct general purpose characteristic. The key to the general purpose capability of the digital computer lies in the computer program and its supporting documentation. The program and its documentation are often referred to as software.

Proper employment of software design and production techniques permits the ACDS system planner and system manager to remain more responsive to the line commander, and also permits more effective employment of the advantages provided by an evolutionary implementation.

Experienced software personnel (i.e., computer programmers and training specialists) must participate in the original design of increments of improvement to existing systems. They must also participate in the evaluation of proposed design changes. This is provided for in Section 2.4 (Operational Analysis and Design) and also in Section 2.3 (Management of Evolutionary Implementation).

Small changes in programming can give rise to large modifications in system performance and, in many instances, provide for the accommodation of substantial changes in tactical doctrine. At the other end of the same spectrum, modest changes in system hardware (particularly communications equipment) can generate large changes in system computer programs. Naval system planners must maintain a continuing appreciation of the problems of software design and production in order to make proper use of the power and flexibility of the general purpose digital computer which is the heart of any command data system.

2.6.2 The Products of Software Production

The design and production of different increments of improvement to ACDS capability have differing effects upon the activities of the software contractor. In some instances large changes require basic redesign of parts of the computer program system. In other instances, substantial improvements in performance are obtained by the changing of a few subroutines or numerical constants.

For these reasons, increments of software improvement for ACDS do not always follow the same paths through the software design cycle. Larger changes have an impact on program system design while smaller changes only have an impact upon the design of some particular operational program. Small increments of improvement can be installed in the field at the request of the commander if the proper field activity is provided.

For these reasons, not every increment of system improvement is accompanied by the same products from software design and production activities. The major products are described briefly below.

2.6.2.1 Program System Description

This is the basic computer programming document. It describes the technical features of design of the data base, of the executive program, of program timing, and of such other details as the handling of switch impulses within the program system. Although the program system description is the fundamental document for the computer programming activity, it normally is not affected by improvement increments to the operational programs.

2.6.2.2 Computer Program Operational Specifications

These specifications describe how each of the various computer programs operate within the programming system. These specifications are used by the software contractor to control the design and production of the programs. They do not specify how the programs provide the specified output - only what operations the programs perform, with what frequency and accuracy, and what the inputs and environment are.

Computer program operational specifications for operational and support programs are normally distributed in small numbers to using commands where they may be used as technical reference material, where they must be available for computer program trouble shooting.

2.6.2.3 Program Coding Specifications

These describe in data processing terms exactly how the programs operate upon their inputs to produce the required output. These are of interest only to the data processing specialist and are normally distributed to a rather restricted audience. They are

required by software specialists in the field in order to install and check-out programs and their changes. Coding specifications must be available in the field for computer program trouble shooting, but are not of wide interest to the user.

2.6.2.4 Computer Programs

The computer programs generated during the software design and production cycle connect the various hardware elements of the system through the medium of the general purpose digital computer. It is only through the operational programs that the commander and his staff have access to the system data base and can command the system in order to obtain their results.

There are four general families of computer programs which must be generated for ACDS:

- 1) Operational Programs.
These programs execute the operational tasks of the command data system.
- 2) Utility Programs.
These are the programming tools with which the computer programmers write and check out the operational programs.
- 3) Support Programs.
These programs are deployed to the field along with the operational programs, but are not used in the conduct of the operational tasks of the system. These programs support the line commander in the performance of such system tasks as recording and analyzing system performance during exercises, or reducing and reporting daily operational recording for the production of routine command reports. They also are used to exercise and test the system.
- 4) Facility Programs.
These programs are special testing tools which the programmers and coders of operational programs use to test those programs. Facility programs allow them to simulate an operational environment, run their programs and record the results.

2.6.2.5 Computer Program Support Documentation

These documents are the miscellaneous technical documents produced by the agency or agencies generating the computer programs. Their purpose is to provide technical reference material for other computer programmers involved in the process of production, testing, installation or error correction. When the computer program production activity is small, much of this information can be passed on an informal basis; as the production activity and as the system itself grows larger, more formal documentation is required. Reference to Section 2.9 shows that as these internal documents increase in number, computer programming costs increase substantially.

These documents are so technical in nature that they are seldom distributed to the using organization which, in turn, has no use for them.

2.6.2.6 Operational Handbooks

These handbooks describe the procedures by which each system position operator executes his various operational tasks. The handbooks are normally produced one per position so that they may be distributed at line unit level to each individual operator for the position which he normally mans. The volumes contain schematics of his possible switch actions, and his possible displays, along with explanations of the operational circumstances under which these various displays and actions would be available to him.

In the production of ACDS software, it may be that positional handbooks are written and published by an agency other than the computer programming producer. If this is the case, extensive liaison is required between the computer programming agency and the publisher, since it is through the medium of the computer program that all system operators have access to the system.

2.6.2.7

The production of training materials is normally tied directly to the production of the computer program, since changes in computer programming nearly always force some change in an operator procedure, in the value of data normally presented, or in the time required to obtain certain information from the system. Close liaison is required for the accurate and inexpensive production of operator orientation and training materials.

System exercise materials are those items which allow the line commander to exercise and train his part of the command data system. Since ACDS has general purpose digital computers available to many of its nodes, certain types of system exercises may be conducted by allowing the system computer to present the exercise situation to the operational personnel. This can be done through the reading of a previously generated magnetic tape which contains all the synthetic exercise inputs.

These magnetic tapes must be generated by an agency intimately familiar with both the operational and technical details of the entire system. For this reason, exercise tapes may often be produced by the system computer programming agency. Another example of a support program would be the one used to generate the system exercise tapes.

2.6.2.8 Operational Analysis and System Design

A discussion of operational analysis and system design is provided in Section 2.4. In most discussions of command data systems, operational analysis and design is included in the category of software. In this particular study, it is shown as a separate and distinct process, since it is not only possible but desirable to perform the necessary operational analysis and system design in the ACDS system management activity, while the production of computer programs, operator handbooks, training materials and system exercise materials are most feasibly allocated to other agencies (some of which may be civilian contracting organizations).

2.6.2.9 Summary

The products of software production as shown above are intimately related to the technical details of the computer program and the overall system design. For this reason, those agencies which produce handbooks, and training and exercise materials must maintain constant and complete liaison with the system management activity and with the computer program design and production activity.

2.6.3 The Inputs to Software Design and Production

There are two important inputs to software design and production. First is the formal and informal documentation produced by the operational analysis and system design activity and described in Section 2.4. The second important type of input is

continuing liaison with the system management activity and with operational units of the line commander. The second type of liaison may be provided by the liaison officer assigned to temporary duty with the computer programming activity. It is of extreme importance to provide the computer programming agency with free and direct access to members of the operational analysis and system design team, to authorized representatives of the using command, and to the manufacturers of all of the system hardware.

There is some temptation to assume that the well documented results of operational analysis and system design are the only required inputs for software production (particularly computer programming). This is not the case. Computer programmers must, by the nature of their task, receive detailed answers to operational and technical questions which are not usually foreseen and are not normally contained within system design documentation.

Certain material provided as input to operational analysis and system design must also be provided as basic reference material for the software production process. Specifically, this information should include material describing the predicted operating environment including threat and operational doctrine.

The formal and informal documentation resulting from operational analysis and system design and required by the software production agency is:

- 1) System operating concept (2.4.2)
- 2) Functional requirements and definitions (2.4.2)
- 3) Functional flow diagrams (2.4.2)
- 4) Function, tasks and step descriptions (2.4.3)
- 5) Equipment capabilities descriptions (2.4.3)
- 6) Manual capabilities criteria (2.4.3)
- 7) Operational mode descriptions (2.4.4)
- 8) Procedural linkage descriptions (2.4.4)
- 9) Preliminary operational system description (2.4.5)

The number in parentheses following each of the above items shows the section of this report in which it is discussed.

2.6.4 Steps in Software Design and Production

The process of software design and production in support of an evolutionary system implementation is not a highly formal process. Each change goes through only those channels which are appropriate for the implementation of that particular program change or improvement. Larger changes, which have wider ramifications, necessarily require more steps in their handling. Figure 2-9 shows the process by which computer program improvements are produced in response to new requirements from line commanders. The larger the scope of the program improvement, the more activity is involved in each of the various design and production steps.

The first increment of computer capability to support ACDS requires the creation of an ACDS computer program system. This original program system can be of very modest size (depending entirely upon the system requirements). Subsequent improvements to the program system can be designed to take maximum advantage of the program system capability already present.

The various steps required to establish an original ACDS software and computer program system are discussed in Sections 2.6.4.1 through 2.6.4.5 and are represented schematically in Figure 2-10 through 2-14.

Before beginning a detailed discussion of the contents of each of the steps in software design and production, one point should be emphasized. This discussion will show what steps are necessary to create a software system "from the ground up". If certain physical facilities or certain programming facilities already exist, they do not have to be recreated simply because a block exists in these diagrams. For instance, if a utility system already exists for the naval computer(s) which may be used in ACDS, no new utility system has to be created, although some additions may be required.

For purposes of discussion, software producers tend to think of the steps in software design and production as belonging to a certain phase of the process. This technique is used in the following sections for purposes of simplicity.

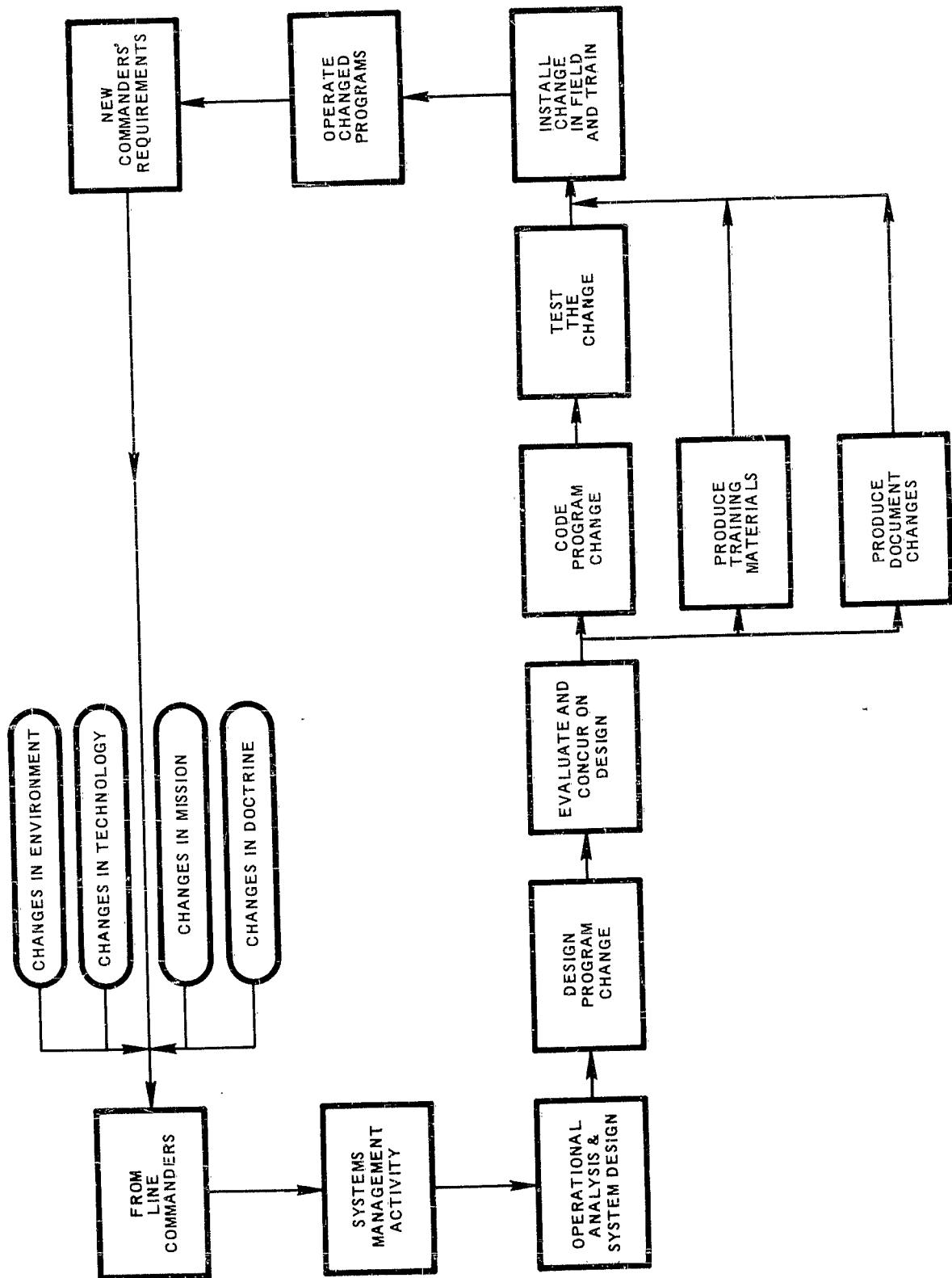


Figure 2-9, Processing Computer Program Improvements

2.6.4.1 Program System Design Phase

Program system design (shown in Figure 2-10) is that phase of the programming agency's activity which designs and specifies the fundamental computer programming concepts, conventions, and standards upon which all subsequent programming activity is based. Once the computer program system is designed and specified, it is seldom necessary to pass through the phase again. The following paragraphs describe the various parts of the program system design phase.

Plan and Begin Computer Facility. This step, and the subsequent step of install EAM facility, is necessary only if the ACDS computer programming agency does not already have access to the full family of ACDS computers as well as a supporting EAM facility. It is possible to write small computer programs well and economically when the computer programming contractor must use the customer's machine. However, for the kinds of programming tasks which we envision for ACDS evolutionary implementation, it is necessary for the computer programming agency to have first priority access to the full set of ACDS computers. This does not require the creation of an "overhead" facility of computers, although this would be the ideal solution from the point of view of the computer programming agency.

There is a critical system management decision which must be made in this area. System management must decide whether it is better to reduce computer programming costs at the expense of computer costs, or vice-versa. By far the most beneficial arrangement in the eyes of the programming agency is the one in which they have installed in their physical facility all the required EAM support, as well as at least one computer of each different type. This, of course, would be modified in the instance of the CP667 and the Q-20B. Q-20B programs could be checked out on a 667 machine operating in the Q-20B mode. From the point of view of minimizing computer purchase or rental costs, the computer programming agency could be directed to travel to some naval facility having the desired equipment. Having arrived, programmers would then wait their appropriate turn to use the desired equipment. Of course, there are many intermediate ways in which computer access may be provided for the programming agency.

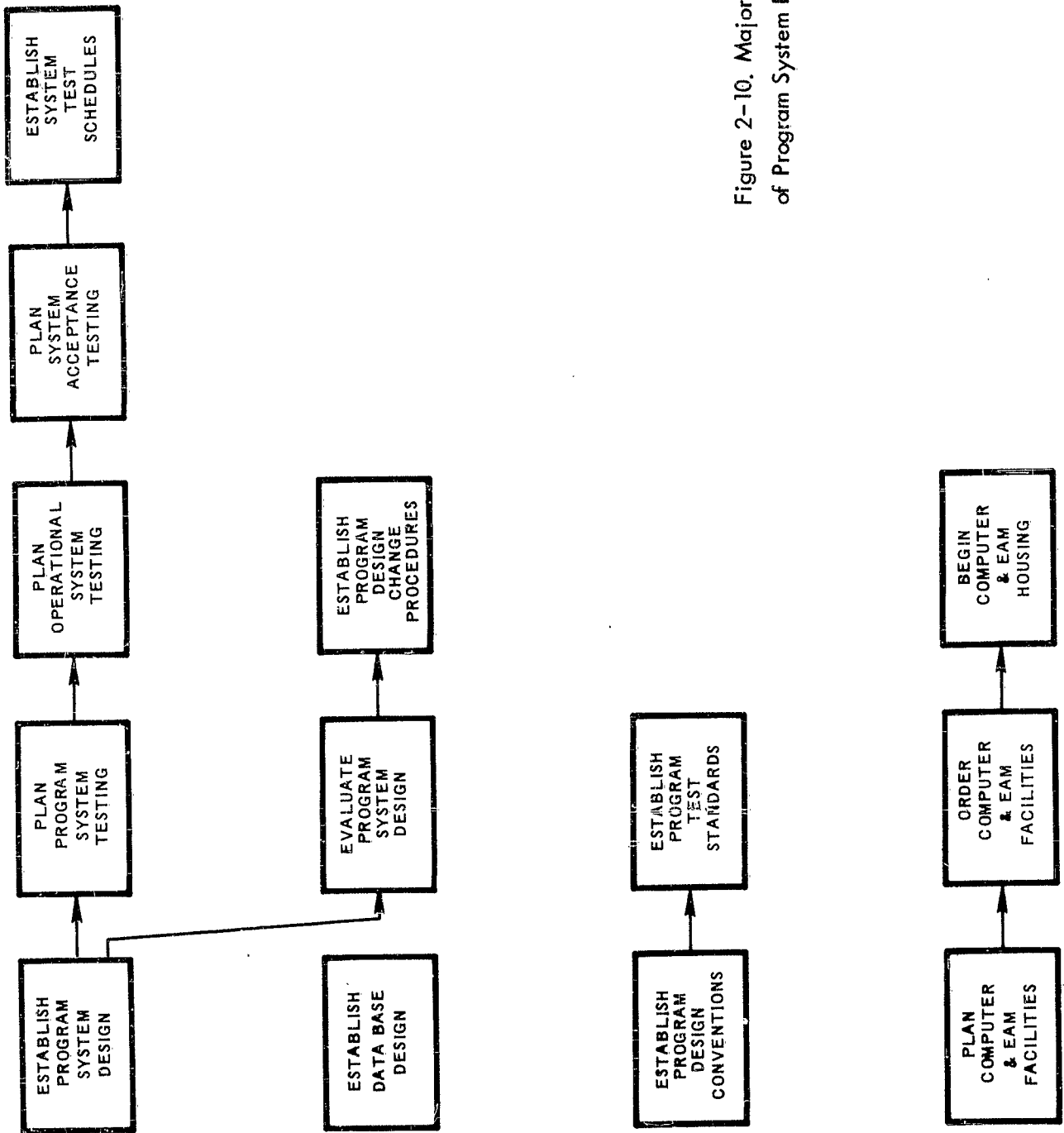


Figure 2-10. Major Elements of Program System Design Phase

Establish Program System Design. In this step, the computer programming agency designs and documents the central concepts of the computer programming system. In a programming system where a large number of tasks are performed intermittently, while others are performed cyclically, careful attention must be given to the task of program system design. Program system design is primarily concerned with the design of the operational programs which direct, coordinate and time the performance of the balance of the programs in the operational system. These programs are normally considered to be small parts of the executive program. The executive program provides the proper input and output messages, receives and forwards the information resulting from switch actions, calls programs in to be operated when they are required, and may be thought of as the CIC through which the system operators control the performance of the operational program system.

The manner in which the executive program is designed determines the ease with which subsequent modifications to the executive program may be made, and also the ease with which changes may be made to other operational programs as evolutionary steps are required in the future.

Plan for System Testing. At the same time as the program system design is being established, a group of software specialists begin the planning concerned with system testing. In this step, they are examining not only the requirements to test the computer program system, but they examine the way in which computer programming affects the testing of the entire increment under consideration. The activity in this step is conducted using inputs from the program system design step as well as the basic system documentation available from operational analysis and system design.

Establish System Tests and Schedules. As the program system design phase comes to a close, program system designers elaborate on the plans for system testing, and they develop tentative schedules for the tests to be conducted in the future. During the design of the original computer program system capability, the details of these system tests and schedules are constantly developed and modified as additional information becomes available concerning the correlated hardware system. During this step, program system designers maintain very close liaison with the system management activity to ensure that appropriate program system support is available for overall system tests.

Set Program Design Conventions and Standards. Once the program system design has been established (and in some instances concurrent with that step) it is necessary to establish program design conventions and standards. These conventions are such things as the manner in which individual pieces of data are referred to, the accuracy with which various types of data are stored, the manner in which various operational programs transmit information to each other, etc. This step is normally concurrent with the step, "establish data base design."

The care with which program design standards are set has a great deal to do with the ease with which increments of computer programming capability may be added to the system in the future.

Establish Data Base Design. Program system designers are concerned with determining the name, nature, accuracy, and official source for each piece of numerical data which must be handled by the operational programs. During the period when the first programming system capability is being created, this step cannot be considered finished until system tests are satisfactorily completed. Additional requirements for new types of information and new accuracies continue to arise and require decisions during the establishment of the original program system design.

In this step, program system designers also concern themselves with the manner in which these large volumes of data are stored and updated within the operational system. The consideration of the updating of these data often requires the design of a small operational program to perform this function. There is substantial liaison required between this step and the previous one inasmuch as they are both concerned with the standards and storage techniques for handling the large amounts of base data which may be required in the system. As soon as the data base design has been established, the collection of the data itself begins.

Evaluate Program System Design. In this step, program system designers analyze the activities and products produced thus far in program system design. An evaluation is made of the degree to which the design products produced thus far meet the requirements as specified in the preliminary operational system description, as well as the internal program system design requirements that have been generated during the design process itself. This step has two parts. During the first part, the evaluation is conducted primarily internally to the program system design group. In the second part

the evaluation is coordinated with operational analysis and system design personnel as well as with ACDS system planners.

Establish Program Design Change Procedures. This step normally follows immediately after the evaluation of the program system design and the agreement that the design is a proper one. In computer programming it is difficult, if not impossible, to "freeze" a design such that it may never be changed until some specified time in the future. The nature of computer programming makes it difficult, if not impossible, to foresee all the possible ramifications and interconnections of processing tasks yet to be designed and coded. In addition, programming design must be changed at various times in the future to accommodate the various increments of improvement to the basic ACDS capability. For these two reasons, a regular channel through which program design changes are proposed, evaluated, and processed is established. It is normally established immediately following the agreement upon the first part of the computer program system design.

2.6.4.2 Program Design Phase

In this phase of the programming agency's activity (shown in Figure 2-11), the operational programs are designed, the designs are evaluated and concurred upon, and data base preparation is begun. The next paragraphs describe the parts of the program design phase.

Design Programs. The first four steps in the program design phase are undertaken concurrently. The design of exercise programs lags slightly to receive appropriate design information concerning operation, utility and data base programs.

Small program changes for the implementation of the basic programming system enter the software design and production process through the program design change channel at the point indicated on Figure 2-11.

The input for the design of these various programs comes from the program system design phase in the form of internal technical documentation. Program designers must also make extensive use of the analysis and design documentation coming from the operational analysis and system design step.

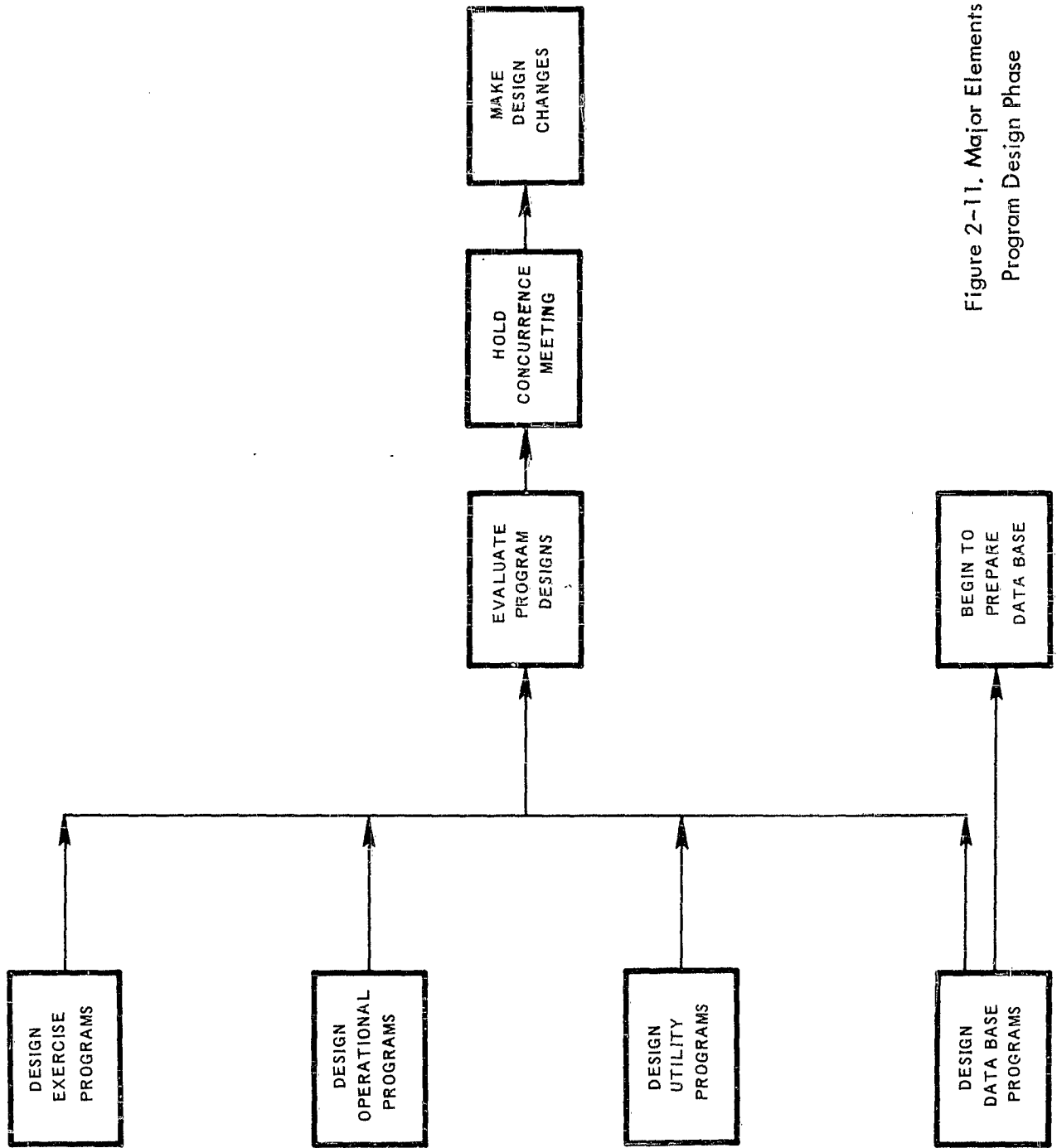


Figure 2-11. Major Elements of Program Design Phase

Evaluate Program Designs. The designs of the various types of computer programs are reflected in documents called computer program operational specifications. That is, these documents specify, in detail, and in computer programming terminology, the precise performance required from each one of the various computer programs in the program system. The word operational refers to the operations of a computer program and not tactical nor strategic operations. Therefore, exercise and test programs, utility programs, and data base programs as well as operational programs all have operational specifications.

The operational specifications for the entire program system are evaluated at a joint meeting of program design personnel, program system design personnel, and personnel from operational analysis and system design as well as technical representatives of the system management activity. The concurrence stemming from this meeting indicates that the computer program designers and the computer program system designers have accurately and appropriately interpreted the operational requirements for the computer programming system.

Make Program Design Changes. For large increments of improvement in computer program capability, a concurrence meeting may last as long as ten working days. During this time, it is to be expected that the number of small errors and inconsistencies will be discovered in the computer program operational specifications. During the period of the concurrence meeting, the remedies for these errors are designed and are, in turn, concurred upon so that, by the end of the concurrence meeting, there is unanimity of opinion as to precisely what is contained in the program specifications and what these specifications will provide in terms of an operating program system.

Begin Preparation of Data Base. The collection of data base information proceeds from the time of the program system design phase. Up to this point, little effort is applied to prepare the data base itself, since the precise configuration of the data base depends upon the final configuration of the computer program operational specifications. Once these specifications and their changes are concurred upon, the data base information may be refined and the construction of the data base is begun.

2.6.4.3 Program Production and Test Phase

During this phase the programming agency codes the various types of complete programs required for the command data system and tests them for performance. Figure 2-12 shows the major elements of this phase. Two distinct types of program testing are employed during this period. They are explained here. The balance of this section explains the production and test phase.

Parameter Testing. Parameter testing refers to the testing by the individual programmer of his particular program or subprogram. A parameter test is one in which the operation of the program is checked in connection with the processing of certain outside or limiting values and certain most popular or most likely values. The output of the program is compared with the previously hand calculated results by the programmer involved. Parameter testing gradually becomes more thorough and more complex until the programmer is relatively confident that his program does perform as he intended it to do.

Assembly Testing. Assembly testing is that testing which examines the operation of an individual computer program as it operates in the environment of its neighbor programs and under constraints which begin to resemble operational conditions. For assembly testing, more complex and realistic inputs are required, larger numbers of different conditions and values are processed, until finally the programmer and his supervisors are satisfied that this area or neighborhood of programs operates as required.

Program Testing is almost completely intermingled with program production. That is, as soon as utility programs are coded, they are tested. As soon as facility programs are coded, they are tested. As soon as data base programs are coded, they are tested. This is necessary so that complete utility, facility and data base support is available prior to the 25% point of the operational program coding step. The 25% mark is arbitrary, but approximately that much coding effort can be expended upon operational programs without having complete computer and support program capability available. If computer and support program capability is not available by this time, high inefficiency results in the process of coding and testing operational programs.

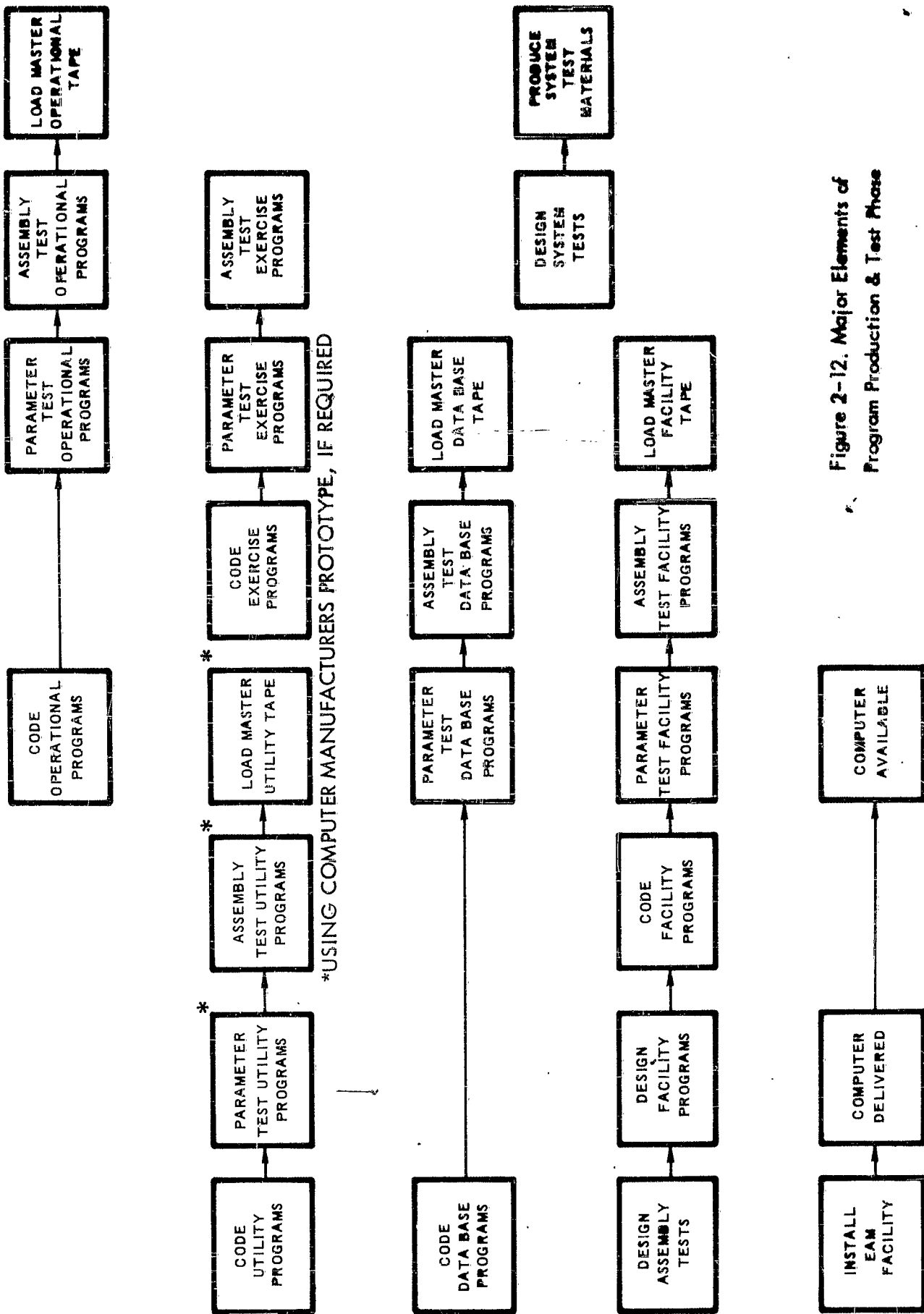


Figure 2-12. Major Elements of Program Production & Test Phase

Code Utility Programs. These are the programs which the computer programmer uses as tools to assist him in the construction of other computer programs. Utility programs normally consist of a compiler and a checker. The compiler assists the programmer in the construction of his program, while the checker assists the programmer in finding the errors which prevent the satisfactory operation of that program. Occasionally, in more expensive and sophisticated utility systems, additional programming tools are provided. In some very modest systems, no checker is provided.

The utility programs must be coded first to provide other programmers with the means of coding their programs.

Code Data Base Programs. In some types of command data systems, the amount of data base information required for system operations is so great that specific operational force support programs are necessary to load, manipulate, or update the data base. If these programs are required for ACDS, they must be coded at this point in the production phase so that programmers coding operational programs may use data base information as well as test programs and utility programs to test the coding of the operational programs.

Design Assembly Tests. These are the standardized tests which demonstrate that each operational program or set of operational programs performs its data processing functions as required by the computer operational specifications.

Install EAM Facility. This facility must be available to the utility system programmers at the beginning of the utility system coding effort.

Design and Code Facility Programs. These programs are designed and coded after the general scope and concept of the assembly testing is known. They are required for both parameter and assembly testing of the operational programs.

Computer Delivered. This step is concurrent with or shortly follows the coding of utilities and facility programs. The completion of these programs indicates that computer programmers must have convenient access to the family of computers which will be used for ACDS. Facility, utility and data base programs can be coded without convenient access to the computers. If this is the course followed, then additional time must be allowed for computer programmers to travel to a computer installation and there check out their programs.

Parameter Test and Assembly Test Utility Programs. While the last previous steps were being performed, the utility system programs were tested. After final adjustment and retesting, they are loaded on a master utility tape.

Load Master Utility Tape. The next step for utility programs is to load them on a master tape arranged in such a way that they are available to support the computer programmer coding and testing operational programs. The first few days of operation in support of the operational programming activity serves to finish the testing of utility programs.

Code Operational Programs. This step may begin at almost any time after the completion of the utility programs. This is the step which generates all the, operational computer program capability for the system.

Computer Facility Available. Without regard to the computer facility provided for the testing of utility data base and test programs, a convenient computer facility must be available on a high priority basis not much later than one quarter of the way through the time allocated for the coding and checkout of operational programs.

Parameter and Assembly Testing of Facility and Data Base Programs. As soon as the computer facility becomes available, parameter testing is begun for facility and data base programs which have been previously coded. This is followed as soon as possible by their assembly testing. The object is to provide a complete computer, utility program, facility program, and data base program subsystem as early as possible during the coding of operational programs.

Load Master Facility and Data Base Tape. When assembly testing of these programs has been completed, they are loaded onto a master tape. Assembly testing of these programs may be reduced somewhat below that required for operational programs for two reasons. First, they are not normally delivered to the Navy. On the few occasions when they may be, they rarely are deployed to line units. The payoff to the Navy of highly documented testing is, therefore, problematical. Second, a very thorough informal testing is given these programs as they support the programmers producing the operational programs. There is some modest advantage in loading the three master tapes (utility, facility, and data base) as early as practicable, at least in provisional form.

Test Operational Programs. As soon as the utility, facility and data base support is available to the operational programmer, he may begin the parameter testing and assembly testing of his programs as previously described.

Code and Test Exercise Programs. As soon as a rudimentary utility capability exists, effort begins on the coding of exercise programs. Exercising programs generate the materials which the line commander uses as synthetic inputs when he wishes to exercise and test his system. In some systems certain exercise programs are used in the field by the commander to record the results of exercises or to keep logs of daily system performance.

Load Master Operational Tape. As soon as a significant number of operational programs are thoroughly tested, a master operational tape is loaded. This tape is continuously reloaded and updated until such time as all operational programs are thoroughly assembly tested and felt to be ready for the program system testing.

Design System Tests. During this step, the tests of the entire computer program system are planned and designed. In addition, the same design team plans the computer program portions of the total system testing to be performed. In this operation, they must maintain close liaison with the system management activity, hardware manufacturers, line commanders, and operational analysis and system design personnel.

Produce System Test Materials. During this step, exercise and testing programs are combined with system test plans to produce the simulated input tape, the console operator scripts, and the pre-calculated results necessary to test the operational computer programming system.

During this step, any additional materials required for the system testing of the entire hardware/software system are also prepared.

2.6.4.4 System Test Phase

This phase is relatively separable from that of program production and program test since it cannot begin until the operational master tape has been loaded the the operational programs thoroughly assembly tested. The major elements of the system test phase are shown in Figure 2-13.

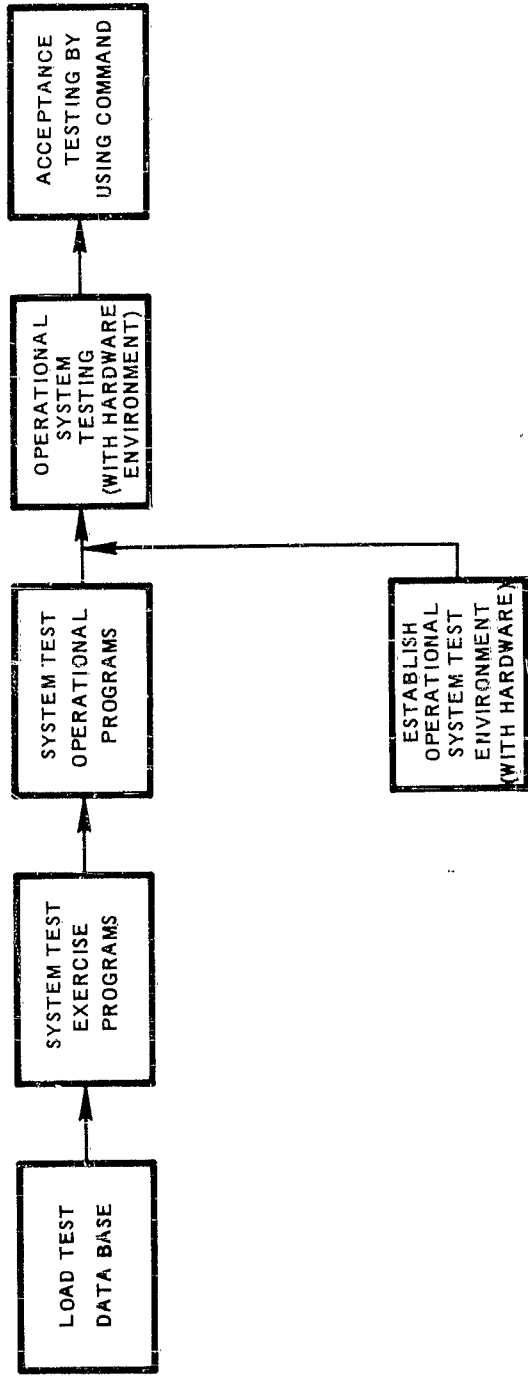


Figure 2-13. Major Elements of System Test Phase

Load Test Data Base. Normally, a standard and synthetic data base is loaded for the purpose of performing program system testing. By providing a stylized and simplified form of data base for program system testing, the reduction and interpretation of the results of testing are made considerably easier.

System Test Operational and Exercise Programs. Since the exercise programs are used to provide the test environment for the operational program system, the program system testing actually uses one set of programs to test the other.

When program system testing is completed, the exercise programs and the operational programs are ready for deployment to the field.

Operational System Testing. This phase in testing is the first one which involves the combined testing of both the hardware and the software*. When the increment of capability being added to the system involves large changes in hardware, operational system testing takes place in a testing and evaluation environment. This environment can be provided either by operational forces or by specialized environments created in a test and evaluation center.

Just as all previous testing uncovered errors and weaknesses in design, the first interfacing of hardware and software which takes place in operational system testing uncovers incompatibilities in design as well as flaws in the execution of designs.

As soon as design and production errors are located, diagnosed and rectified, the completed system is ready to pass to the next step, System Acceptance Test.

System Acceptance Test. By this point in the software design cycle, considerable confidence can be placed in the accurate and continued operation of the software system. For a complete discussion of system acceptance testing, see Section 2.5.2.6.

* Except for that which is the result of running the programs on the computer during their production and test. Substantial hardware testing is accomplished during this period on an informal basis.

2.6.4.5 System Operation Phase

When system acceptance testing is completed, the system is deployed to the field to be installed at the using unit. Some configurations of system improvement will have system acceptance testing conducted after field installation. Major elements of the system operation phase are shown in Figure 2-14.

As soon as system installation and training is completed for the new increment of capability, it is subjected to the most rigorous of all testing--operation in the field by line personnel. With design and test personnel no longer unconsciously babying the system, new shortcomings and flaws appear. During the elimination of these new-found difficulties, the line operational personnel become fully cognizant of the technical and operational details of their new capability.

The processes of daily operation maintenance and training, as well as the analysis and evaluation normally provided by the using command, give rise to new commanders' requirements. These requirements are forwarded through command channels to the system management activity where they are merged with changes in technology, changes in environment, and changes in mission.

This new information and its accompanying set of new requirements is forwarded to operational analysis and system design personnel where the evolutionary system change cycle begins again.

2.6.5 Discussion

In this section we have shown how software is produced for a command data system begin developed in an evolutionary manner.

Two distinct types of effort are required. The first effort establishes the program system and the various utility, facility and support programs required to deploy the first set of operational programs to the using commands. This is shown in Figures 2-10 through 2-14. The second type of effort is that required to provide evolutionary increments to existing command data system capability. This is shown in Figure 2-9.

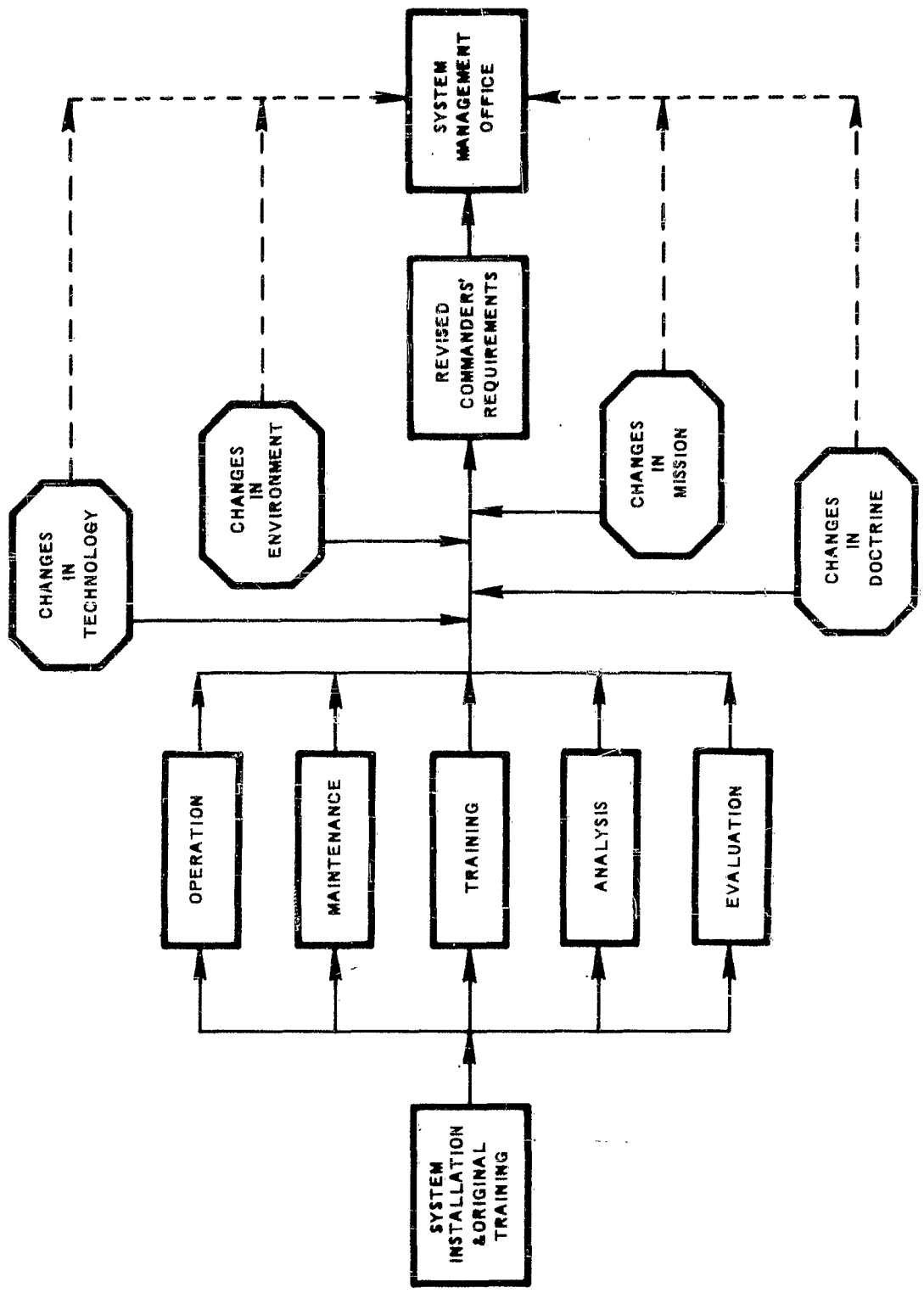


Figure 2-14. Major Elements of System Operation Phase

These two types of software design and production activity are understood by a number of the more sophisticated software contractors, and are ideal for the evolutionary development of command data systems such as ACDS.

2.7 NAVAL PROCEDURES IN SYSTEM PLANNING

2.7.1 General

In earlier sections of this report, system development is discussed from the points of view of the steps and processes to be undertaken and the management aspects related to them. ACDS planners, however, have still other factors to consider. These factors are the project management and control regulations within the DOD and the Navy Department.

This section summarizes the major regulations and policies for initiating and responding to requirements, and for developing implementation plans for Navy systems. ACDS, though still undefined, is likely to be very broad in scope and involved with many of these DOD and Navy procedures. Hence this section summarizes the many aspects of implementing systems from the point of view of regulations and policies.

All Navy system improvements begin in the Navy Department's Research, Development, Test and Evaluation Program; through which all future operational capability is generated. This RDT&E Program is connected by regulation to the Department of Defense (DOD) Five-Year Force Structure and Financial Program (FYFS&FP).

This section presents a brief summary of the major Navy and DOD procedures pertaining to the manner in which systems and projects may be authorized and begun. These regulations reflect the Navy response to certain DOD and OSD directives. Since regulations are subject to change or amplification, some procedural details will doubtless change with time. In the light of a current trend toward managerial and fiscal control being exercised at OSD-DOD level over large expenditures, it is reasonable to suppose that the spirit of these regulations will remain in effect for some time to come.

2.7.2 RDT&E Command Structure

Responsibility for determining Navy operational requirements rests with the Chief of Naval Operations (CNO). Responsibility for conducting RDT&E to meet those requirements rests with the Deputy Chief of Naval Operations for Development (DCNO(D)) through the various Bureaus and Offices, and with the Chief of Naval Research (CNR) through the Office of Naval Research (ONR). Within the overall RDT&E Program, the

CNR is primarily responsible for research, while the DCNO(D) is responsible for development, testing and evaluation. The CNR reports to the Assistant Secretary of the Navy for Research and Development (ASN (R&D)), while the DCNO(D) reports to the CNO. However, the DCNO(D) also has the responsibility for coordinating the entire Navy RDT&E Program with the DOD Five-Year Force Structure and Financial Program using the R&D Management mechanisms described in Section 2.7.4.*

2.7.3 The Five-Year Force Structure and Financial Program

The FYFS&FP is an ordered plan of force structure and price-out projections for coordinating the efforts of DOD and ensuring that these efforts are in accordance with the Basic National Security Plan. All expenditures by all armed services are allocated to one of the seven "numbered programs" in the FYFS&FP according to what portion of the Force Structure they support. For example, all R&D for all services is included in the FYFS&FP as part of Program 6 (R&D). Program 6 is administered at DOD level by Department of Defense Research and Engineering (DDR&E). Each numbered program is divided into major categories. The categories in Program 6 are shown in Figure 2-15.

1	RESEARCH
2	EXPLORATORY DEVELOPMENT
3	ADVANCED DEVELOPMENT

(PROJECT DEFINITION PHASE Required for Projects over \$25,000,000)	

4	ENGINEERING DEVELOPMENT
5	MANAGEMENT AND SUPPORT
6	OPERATIONAL SYSTEMS DEVELOPMENT

Figure 2-15. The Six Categories of "Program 6" (Research and Development) of The Five Year Force Structure and Fiscal Program

* The role of the Chief of Naval Material in RDT&E is not evident from the regulations available for this analysis.

All R&D funding for any project for any service must appear as a line item in one of these categories. Which category a project appears in depends upon the current state of the project. In general, any project's development cycle begins in Category 1, 2, or 3 of Program 6 and progresses to higher categories, requiring more extensive justification at each step. The justification required to progress to a higher category is detailed in the management procedures published by DDR&E.

A substantial number of DOD and Navy procedures govern the procedures, documentation, and approvals which are required for various Navy RDT&D projects in support of the FYFS&FP. These are shown schematically in Figure 2-16 and 2-17. Figure 2-18 shows the principal regulations which describe each step. Sections 2.7.4 and 2.7.5 summarize the documents and procedures, and Section 2.7.6 presents a synopsis of the various planning tools involved.

2.7.4 General Navy R&D Management Mechanisms and Requirements

General R&D projects within the Navy flow from operational requirements established by CNO or CMC through DCNO(D). There are three kinds of standing requirements documents.

- 1) Naval Research Requirements
- 2) Exploratory Development Requirements
- 3) General Operational Requirements

Naval Research Requirements (NRR's) comprise a list of 11 areas in the sciences, numbered from R001 through R011 (for example, R001 is chemical sciences). The NRR's form a standing authorization for ONR and other developing agencies to initiate projects in those areas which provide information related to the solution of specific practical problems or to better understanding of the subject under study. Such projects belong in Category 1 of Program 6 of the FYFS&FP.

Exploratory Development Requirements (EDR's) comprise a list of 19 Navy functional areas numbered from F001 through F019, (for example, F001 is target surveillance). As with NRR's the EDR's form a standing authorization for ONR and all developing agencies to initiate projects in the areas of their competency. For EDR-based

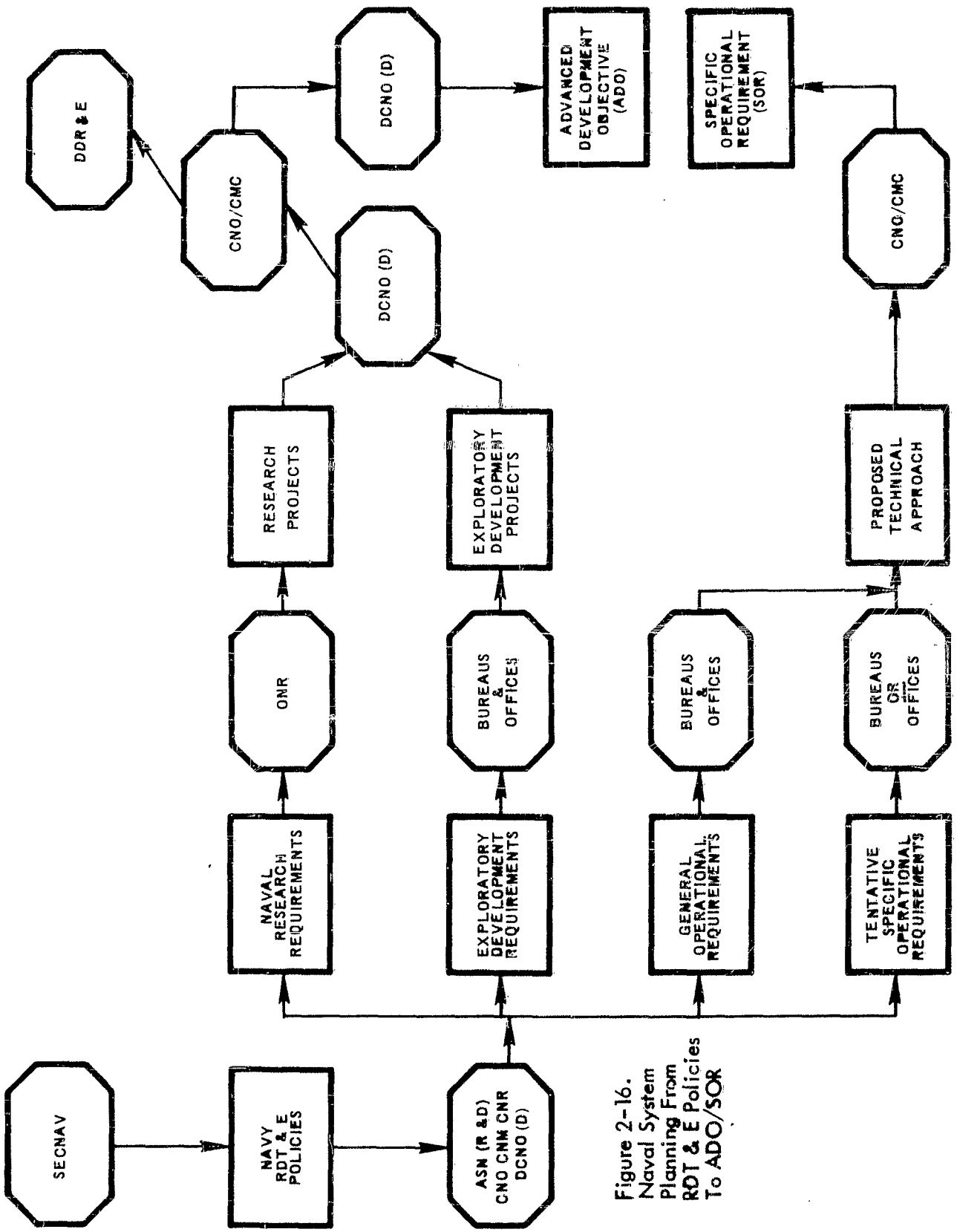


Figure 2-16.
Naval System
Planning From
RDT & E Policies
To ADO/SOR

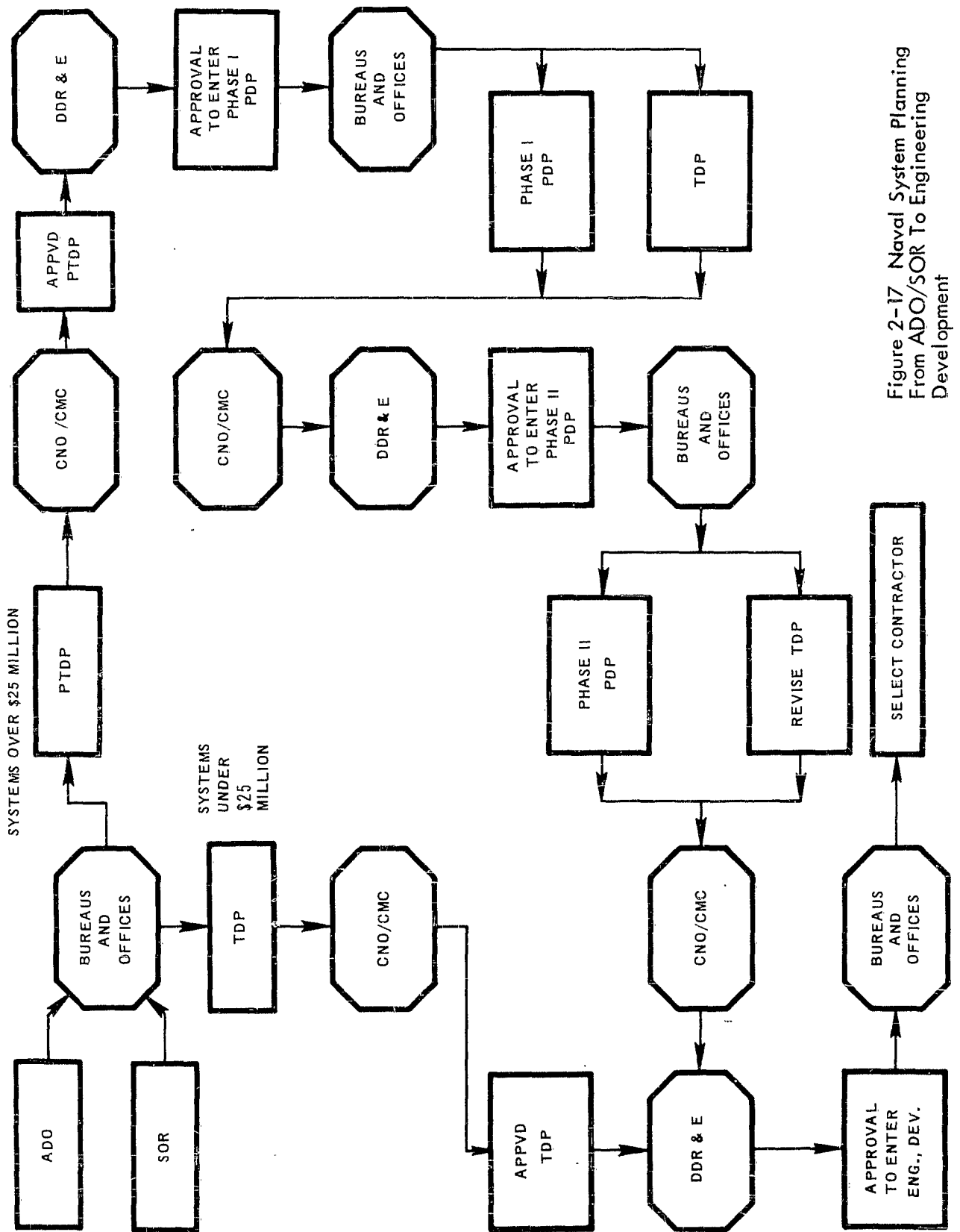


Figure 2-17 Naval System Planning From ADO/SOR To Engineering Development

projects, however, the purpose is to demonstrate new techniques or establish the feasibility of a system, subsystem or component. EDR-based projects belong in Category 2 of Program 6. Advances in knowledge and technology resulting from NRR or EDR-based projects may result in proposals for specific development via procedures discussed in subsequent sections.

<u>RDT&E Policies</u>	<u>General Operational Requirements</u>
SECNAVINST 3900.7A 5000.16A 5000.17 5430.67 7110.10 7133.3	OPNAVINST 3910.9
MCO 3900.3A	<u>Tentative Specific Operational Requirements</u>
OPNAVINST 3900.8B 5430.20 5430.21	OPNAVINST 3910.6B
NAVCOMPT 7000.9B	<u>Proposed Technical Approach</u>
<u>RDT&E Requirements</u>	OPNAVINST 3910.8
OPNAVINST 3910.13 3910.14A 3910.16	<u>Research & Exploratory Development</u>
MCO 3900.4	OPNAVINST 3910.11 3910.13 3910.14A 3910.16
<u>Naval Research Requirements</u>	<u>Specific Operational Requirements</u>
SECNAVINST 5430.20A	OPNAVINST 3910.6B
ONRINST 5910.2A	<u>Advanced Development Objectives</u>
OPNAVINST 3910.2A	OPNAVINST 3910.7A
<u>Exploratory Development Requirements</u>	<u>Technical Development Plan</u>
OPNAVINST 3910.3A	OPNAVINST 3910.4B 3910.12
	<u>Project Definition Phase</u>
	DOD DIR 3200.9
	SECNAVINST 3900.28
	NAVMATINST 3900.2

Figure 2-18. Major Procedures Governing Naval System Planning

General Operational Requirements (GOR's) comprise a long list of general requirements, one for each Navy functional warfare and support area. (For example, GOR 14, in the STRIKE warfare grouping of functional warfare areas, is amphibious assault). For GOR's, however, the procedure of project initiation is different from that under NRR's and EDR's. NRR's and EDR's authorize project initiation with no further approval necessary. GOR's only authorize the developing agencies to submit to CNO development proposals which may or may not be approved for project initiation. GOR-based projects are directed toward meeting definite operational requirements in a particular area, and are initiated under Category 3 of Program 6.

2.7.5 Specific Navy R&D Requirements

The CNO and CMC may also (through the DONO(D)) direct the initiation of specific projects. There are two means for doing this. The first is the creation of a Tentative Specific Operational Requirement (TSOR), and the second is the creation of an Advanced Development Objective (ADO). These are discussed below.

Tentative Specific Operational Requirements (TSOR's) are addressed to an appropriate Bureau or Office, and provide amplification of a particular operational capability need already stated in general terms in a GOR. The TSOR does not establish a firm requirement nor authorize the initiation of a development project. It does require the address agency to respond with a Proposed Technical Approach (PTA). The PTA presents, for CNO or CMC consideration, one or more methods for achieving the desired capability, and provides three general classes of information:

- 1) Provides technical analysis of the possible development.
- 2) Assesses technical risks and costs involved.
- 3) Recommends methods for providing the capability after consideration of cost-time and cost-performance trade-offs.

The PTA can also be voluntarily submitted by an agency directly in response to a GOR without the receipt of a TSOR. In either case, CNO considers the information provided, and either approves or rejects the PTA. If it is approved, CNO then issues a Specific Operational Requirement (SOR). The SOR is a more detailed elaboration of the guideline data provided in the TSOR, and is the document authorizing and directing

initiation of the specific development project. The first step in such initiation is the generation by the developing agency of a Technical Development Plan (TDP) as the output product of Category 3 (Advanced Development) under Program 6. Subsequent approval of the TDP, first by CNO and then by DDR&E, is required before the project can move on to Category 4 (Engineering Development). For larger projects, (those larger than \$25,000,000) an extra phase is required prior to entry into Engineering Development. This phase is the Project Definition Phase (PDP), and is entered by submitting a Proposed Technical Development Plan to DDR&E.

The PDP is, in effect, an additional two stages of elaboration between preliminary and final versions of the TDP. Thus, for large projects, three successive approvals by DDR&E are required before Category 4 can be entered, one for the PTDP and one each for the two phases of PDP.

Advanced Development Objectives (ADO's) outline an experimental system or major component not yet assured as to military usefulness, technical feasibility, or financial acceptability. The ADO (as does the SOR) directs a developing agency to respond with a TDP to accomplish the stated objective. However, TDP's responding to ADO's need not furnish data in some areas required of those responding to SOR's.

2.7.6 Synopsis of Naval System Planning Tools*

The regulation from which this information is extracted describes the policy and procedures for coordination and integration of RDT&E within the Office of CNO, and provides guidance in RDT&E matters for other Bureaus and offices. The regulation also describes the planning documents and administrative devices presented here.

Planning Objectives (PO)

This document separates the common objectives of Navy functional warfare and support into four major groupings:

- 1) STRIKE Warfare,
- 2) ASW,

* Extracted from OPNAVINST 3900.8B, 9/16/63, Planning Procedures for the Navy RDT&E Program.

- 3) Command Support,
- 4) Operational Support.

General Operational Requirements (GOR)

One of these is prepared for each functional warfare and support area. It states in broad terms the capability required in that area. It includes estimated threat and operational requirements needed to meet that threat. Based on GOR's, technical bureaus are encouraged to submit to CNO development proposals in the form of Proposed Technical Approaches (PTA's). GOR's are prepared in accordance with OPNAVINST 3910.0.

Tentative Specific Operational Requirements (TSOR)

The TSOR is generated by the CNO and states the need for a particular operational capability. It outlines system characteristics required to fulfill that operational capability and defines desired performance. It directs the technical bureau to which it is addressed to submit a PTA containing one or more recommended methods for prosecuting the development of the system. TSOR's are prepared in accordance with OPNAVINST 3910.6B.

Specific Operational Requirements (SOR's)

The SOR is the response of the CNO to a previously submitted PTA. It states the requirement for a particular operational capability. It is essentially the same as the TSOR, except that it extends performance definitions throughout the operational environment, and it adds a numerical statement of goals for reliability, maintainability, and personnel requirements. The SOR directs the technical bureau of procedure at TDP. The SOR is prepared in accordance with OPNAVINST 3910.6B.

Advanced Development Objective (ADO)

This outlines an experimental system or major component not yet assured as regard to military usefulness, technical feasibility and financial acceptability. An ADO directs a specific bureau to prepare a TDP to accomplish the objective stated. The objective

may be to conduct a feasibility study, to develop an experimental warfare system, or to develop R&D test and evaluation equipment. ADO's are prepared in accordance with OPNAVINST 3910.7A.

Exploratory Development Requirement (EDR)

This states the need for investigations and studies to demonstrate new techniques and Naval functional areas, or the feasibility of a system, subsystem, or component. This comprises the effort directed toward improvement and expansion of Naval capabilities through application of advances in technology. EDR's are published by the CNO in accordance with OPNAVINST 3910.3A. EDR's direct all developing agencies to plan for and initiate appropriate projects in their areas of competency.

Naval Research Requirements (NRR)

These are statements, in general terms, of the need for studies and investigation in the 11 physical and life sciences to provide information related to the solution of specific practical problems, or to better understand the subject under study. NRR's are published by the CNO in accordance with OPNAVINST 3910.2A and ON RINST 5910.2A. NRR's direct all developing agencies to initiate appropriate projects.

Marine Corps Requirements

These are generated by the Commandant Marine Corps (CMC). If the capabilities described are intended for joint Navy and Marine Corps use, Marine Corps requirements must be prepared in accordance with OPNAVINST.

Proposed Technical Approach (PTA)

This presents for CNO consideration, one or more methods for achieving a required capability. It may arise as a response to a TSOR, or it may be voluntarily submitted by a Bureau or Office in response to a GOR. It may have three purposes:

- 1) To provide technical analysis of proposed developments.
- 2) To assess technical risks and costs involved.
- 3) To recommend methods for accomplishing the task at hand.

The PTA must emphasize the trade-off options involved in cost versus time and cost versus performance. New system concepts, generated within bureaus or field activities, may be documented and forwarded to the CNO by a PTA. PTA's are prepared in accordance with OPNAVINST 3910.8.

Technical Development Plan (TDP)

The TDP comprises the plan for the fulfillment of an ADO or an SOR. It is a detailed description of the effort necessary to accomplish the development, and it includes a recommended funding schedule. Its approval by CNO gives authority to commence a development project to the extent of the funds provided by separate actions. When funded, a TDP becomes the primary management control and reporting document for the life of the project. For major developments whose cost will exceed \$25,000,000, a Program Definition Phase (PDP) must be added to conform with OSD procedures. In this case a preliminary TDP is required. TDP's are written in accordance with OPNAVINST 3910.4B and 3910.12.

Project Reports

These are required for analysis and review of RDT&E projects in the categories identified in Enclosure 2 to this regulation (OPNAVINST 3900.8B). Project reports are submitted on DD Form 613 in accordance with OPNAVINST 3900.14A.

Monthly Project Evaluation (MPE)

This provides the monthly updating of information in the TDP summary. It is composed in accordance with OPNAVINST 3910.12.

Research and Exploratory Development Program Highlights

This is used to inform RDT&E managers as to the progress and problems of projects in Categories 1, 2 and 3 (Research, Exploratory Development and Advanced Development).

Hot-Line Report

This is a technique for the rapid reporting of potential and actual trouble spots in RDT&E projects. It is submitted when needed in accordance with OPNAVINST 3910.13. Telecommunication means are authorized.

2.7.7 Discussion

Considerable technical systems effort needs to be accomplished before a TDP can be written in response to ADO 31-05X.* The effort would begin with analysis leading to the decisions on what ACDS is and what its technical relationships are with other systems which generate or require data. The work of this first phase of ANTACCS provides an excellent point of departure, especially in consideration of the technical functions of the system in support of the Task Force Commander.

An examination of the Navy regulations concerning system development, and an analysis of this in terms of command and control and evolutionary systems implementation, indicates that those regulations may not be appropriate. The literal interpretation of them shows that small incremental improvements to the system would need to go through unnecessarily torturous procedures. The procedures seem oriented toward large-scale systems and revolutionary changes rather than the small incremental capabilities which are likely to be added to systems where increased capability can be achieved through adding computer capability and performing the required additional programming.

* The ADO issued for ACDS.

2.8 COSTING, EFFECTIVENESS, AND SYSTEM PLANNING

2.8.1 General

In the future planning of ACDS questions will arise with increasing frequency concerning the subject of "cost-effectiveness." Although it is uncertain at this point exactly how large ACDS will be; the larger and more complex it is, the more important this subject will become.

Cost-effectiveness studies have not been applied extensively to command and control systems but they have frequently been applied to weapons systems where costs are very great. However, the fraction of the total cost of the fighting force represented by command and control is increasing every year and it is, therefore, reasonable to predict ever-increasing attention to the subject. It should be borne in mind that cost-effectiveness for command and control is a pioneering and a research effort at this point in time.

This section is an introduction to the subject of costing and effectiveness. The argument is developed in this section that costing should be kept separate from effectiveness measurement. Costs can normally be measured in terms of dollars, but it is extremely difficult to develop quantitative measures for effectiveness. An "overall" approach to costing and effectiveness is discussed in this section. Following this, cost estimation is discussed in Section 2.9 and effectiveness measurement in Section 2.10.

2.8.2 The Increasing Utility of Economic Studies

In the past few years, military system planners and military system managers have begun to think in terms of what is known as "cost effectiveness". Inasmuch as this terminology is new, many have begun to think that the techniques themselves are new. This is not true. In actuality, what now is called costing and effectiveness has been performed both by the various Armed Forces and by civilian engineers for a number of years. The thorough competitive testing given in the past to various small arms by the Marine Corps and to various types of aircraft by the Navy are examples of cost effectiveness studies which vary only in degree of detail and scope from the studies which are so popularly referred to today.

Engineering has always had as one of its main areas of concern, the question which asks of a new product or project "Will it pay?" This question is referred to as the field of engineering economy and the first edition of the outstanding text in this field was written in 1930.*

It is fair to ask why there has been such an increase in interest in costing and effectiveness studies over the past few years, and how this is related to command data systems. The answer seems to lie in three directions.

First, important systems in the national defense inventory are becoming more and more complex. The complexity of these new items in national defense inventories requires thorough analysis of military usefulness prior to the commitment of funds for their procurement. Among the most complex of the new systems available to the Navy are command data systems.

Second, as a concomitant of this complexity and as the state of the technical art advances, these important new items in the defense inventory become expensive to the point where costly analyses are now justifiable to ensure that all identifiable costs have been located and detailed. This is especially true since a future severe cutback in funds for some system, may cut back the purchased usefulness to nearly zero.

The nature of some command data systems requires that they be purchased nearly completely or not at all. For example, the first 10% of an AAW radar system for the fleet has little operational value.

* Grant, E. L., Principles of Engineering Economy, The Ronald Press Co. New York, 1930

Third, there is a marked tendency to use engineering and economic measurements to compare widely disparate alternatives. In the comparison of two different design approaches to DLGNs, many errors in estimating will cancel out (in both cost estimation and effectiveness measurement). However, when the comparison is between Class 637 submarines and MTACCS, or between Polaris and Minuteman, very detailed estimates must be made accurately to present the alternative choices.

2.8.3 Military vs. Civilian Cost and Effectiveness

Military and civilian cost and effectiveness studies are designed to provide answers to the same sort of questions such as:

- 1) What will the new item do for me?
- 2) How much will the new item cost?
- 3) Does this new system seem to be worth its cost?
- 4) Should I choose to do nothing at this time?

and other questions of this nature.

In essence, these questions are:

- 1) What will I pay?
- 2) What will it buy me? and
- 3) Is it worth it?

There is a fundamental difference between military and civilian studies. Engineering economy studies and investment return studies or cost return studies performed in the industrial or business environment measure both costs and effectiveness in terms of dollars. That is, they are truly economic studies. However, the waging of war is both literally and figuratively not an economic enterprise. It is difficult, if not impossible, to measure the effectiveness of military systems in terms of dollars, certainly not in the context of dollars earned or dollars returned per dollar of investment.

It is true that certain types of strategic systems may have their performance measured in terms of dollars of destruction inflicted upon various real to hypothetical enemies. However, communication systems, command systems, radar systems, and control systems must all have their effectiveness measures in some manner not expressible in dollars.

This, then, is the fundamental and pervasive difference between engineering economy studies performed in the industrial environment, and cost and effectiveness studies performed in the military environment for, as Grant* says, "The dollar is the standard of value which makes commensurable, differences which would otherwise be incommensurable." As this distinction between the civilian engineering economy study and the military cost and effectiveness study becomes more clearly understood, it becomes more evident that the engineering economy study is, in reality, one study which measures both cost and effectiveness in terms of dollars, while the military cost and effectiveness study is, in reality, two separate and distinct studies; one measuring cost in terms of dollars and the other measuring effectiveness in any manner reasonable for the problem at hand.

Once this distinction is firmly understood, it also becomes evident that there can be no such thing as a "cost effectiveness" number which defines the efficiency of a certain command data system. Rather, the results of cost studies and effectiveness studies are a series of complex data and measurements which allow senior military and civilian personnel to select a course of action from among alternative complex and expensive courses of action.** For this reason, we treat cost studies and effectiveness studies as two separate and distinct bodies of techniques, although many principles obviously apply to both areas of endeavor.

* *ibid*

** A fine discussion of the problems in making these decisions is found in the 1965 Naval Review, Enthoven, Alain, *Systems Analysis and the Navy*, pp. 98.

2.6.4 Costing, The Total System Concept, and the Total Force Concept

The total system concept requires that all personnel components, equipments, and subsystems which contribute to a given system be considered in any analysis of that system. These components, personnel, equipments, and subsystems include the maintenance, supply, repair, support and training required for the system, as well as the entire array of operational items and personnel.

With respect to costing, the total system concept requires that all contributions to increased or decreased cost made by the system or made because of the system must be considered in any analysis. These contributions must be identified and evaluated at every echelon where they occur and as far up as CINC or DOD level, if it is appropriate.

At first, this seems self evident, but upon closer examination, one discovers that without conscious attention to the concept many small yet significant contributing costs are apt to be overlooked. The thought behind the principle is this: Each new system has associated with it, costs which are difficult to identify and difficult to segregate, yet which contribute substantially to the total monies which must be obligated to initiate a new system. This is particularly true with regard to personnel costs, maintenance cost, supply inventory maintenance costs, repair costs, etc.

Very often when comparing alternative systems, it is found that these costs differ significantly between the alternatives under consideration, although the first procurement cost of the system alternatives may be quite similar. Although these almost hidden costs are difficult to uncover and to specify in detail, they may represent a significant portion of the total cost difference between the various alternatives, and it is the difference in total cost between the alternatives in which we are primarily interested. It is, therefore, necessary to track down and identify in as much detail as possible, all of the costs which will be incurred by the various systems under consideration. It is only in this way that the total difference in cost between alternatives may be uncovered and fairly stated.

For example, if in Command Data System A, all console operators throughout the entire Navy must be Warrant Officers, and in proposed Command Data System B all console operators may be Chief Petty Officers or Petty Officers First Class, and if, between the two systems there is no other cost difference (although this is most unlikely), there is a significant difference in the total operational costs of the two proposed systems. It is nearly hidden costs, such as this, that are difficult to isolate, which can contribute so much to the total cost of a command data system.

The total force concept is the logical extension of the total system concept in that the costing is detailed not at the system level but at the next highest echelon, the force level. The technique of examining at system level all of the possible contributions to the cost of a system, is extended to force level. A tactical force has a number of components and systems. As a new system is added to an existing force, the total cost of that force will vary, and the cost will vary differently as a function of which of the proposed alternative new systems is added to the old (or existing) force. The total force concept says that as the costs of the new system are considered, they must be considered in terms of how the new system will change the cost of the existing force.

Total force analysis can yield two types of valuable data not available at the system analysis level.

First, the use of certain resources normally shared between systems can only be considered by the use of total force cost analysis. Such items as the shared use of dry docks, naval training facilities, airfields, and supply depots can only be appropriately considered in this way.

Second, and particularly important to the Command Data Systems, the addition of Command Data System A to the force may result in a higher effectiveness for the force than if Command Data System B is added. While this is a major consideration in effectiveness measurement, it is of interest to cost analysis as well.

If the effectiveness to be attained by the force is accurately known, and if Command Data System A is used instead of B, fewer missile or surveillance or bombardment systems may be required to attain that effectiveness goal.

Most often, the maximum effectiveness possible is sought. But if only a desired level of effectiveness is required, a total force analysis could show a reduction in total force cost due to the increased capability of Command Data System A and the attendant reduction in other system requirements. This type of cost comparison can only be made through the use of total force analysis.

2.8.5 Effectiveness, the Total System Concept and the Total Force Concept

The total system concept and the total force concept also apply to the evaluation of effectiveness. The concept of measuring the effectiveness of the "whole system" is quite well established for weapons systems. It is not so well established for command data systems, perhaps because their effectiveness is so difficult to measure quantitatively.

The total system concept requires that all the effectiveness provided by a given system be considered when evaluating that system. For weapons systems, effectiveness usually appears in terms of force probably delivered against a specified target. For command data systems, effectiveness may in some instances, only be measurable in terms of the increased effectiveness of subordinate or adjacent systems. It might also be expressed in terms of increased efficiency of some distant supply base.

For this reason, the most meaningful measurement of some tactical command data systems may come from total force analysis rather than total system analysis.

The total force concept requires that all the contributions to a force's effectiveness be considered when performing an analysis.

As an example, consider the AAW effectiveness of a screen of DE, DD, and DLG vessels. Let each vessel's AAW capability be a "System" and the AAW capability of the entire screen be the "Force". Further, suppose that we must evaluate the effectiveness of the AAW function to determine if a new command data system is justified for installation by the increment it adds to AAW capability. We must first select which AAW complex we will evaluate: the capability of the individual vessel to defend itself, or the capability of the screen of several vessels to defend themselves or perhaps some escorted vessel such as an AGC or CVA?

There is a fundamental and critical consideration here:

If the sole or predominant mission of a vessel's AAW capability is to defend itself only, then AAW system effectiveness should be evaluated for individual vessels of each type.

If the predominant mission is to provide protection by operating in conjunction with other AAW systems, then total force analysis must be used. The effectiveness of the force cannot be determined by adding the effectiveness of each unit as individually determined.

The fundamental importance of this question lies in the cooperative nature of most multiple unit combat. In the instance of a single DLG defending itself, one vessel must perform all the surveillance, tracking, target evaluation, battery assignment, fire control computations with no help from other vessels. When more than one vessel cooperates in an AAW engagement, computing loads may be reduced by sharing track and target assignment functions, and the number of batteries engaging the targets and the total rate of fire increases spectacularly. The second case resembles the first only in general mission, AAW.

If naval system planners are called upon to evaluate some current or projected system capability, they must consider the capability of a force of several systems as distinct from the sum of the capabilities of each system. This is especially true for analyses of command data systems, when the effectiveness of the system under discussion may only be measurable in terms of the total effectiveness of the force being commanded.

It may be that a force is less capable than the sum of all its systems due to the increased load of coordination required. It may be that a force is more capable than the sum of its systems for the reasons given above. It is nearly certain that it is not exactly as effective as the sum of its systems, and this fact requires that most tactical systems can be evaluated both as individual systems and as forces composed of these systems. In the case of the AAW function, it could be that the best improvement might be had by improving inter-ship data link or by providing additional centralized track bookkeeping. Single system analysis performance ignores such considerations.

2.8.6 Costing, Effectiveness and Command Data Systems

Command data systems have capabilities and characteristics which have a very direct bearing upon their costing and effectiveness measurement. The most important of these are presented briefly here.

More than any other type of system, except perhaps communications systems, command data systems may be centralized or decentralized, distributed or single-path to the extent that the system planner desires. Figures 2-19, 2-20 and 2-21 show three distinct configurations for a given command data system. These three different system configurations all perform the same operational tasks.

Many more configurations could be shown, all of which meet the same operational requirements. The importance of this capability is that although they will perform the same tasks, their costs will be quite different, as will their mean time between failure, their communications requirements, their resistance to battle damage and many other important characteristics.

This inherent flexibility must be carefully considered by the command data system planner. Simply meeting the basic requirements is not sufficient. The planner must evaluate the increased cost of memories, processors and communications against the increased resistance to battle damage provided by the distributed configurations.

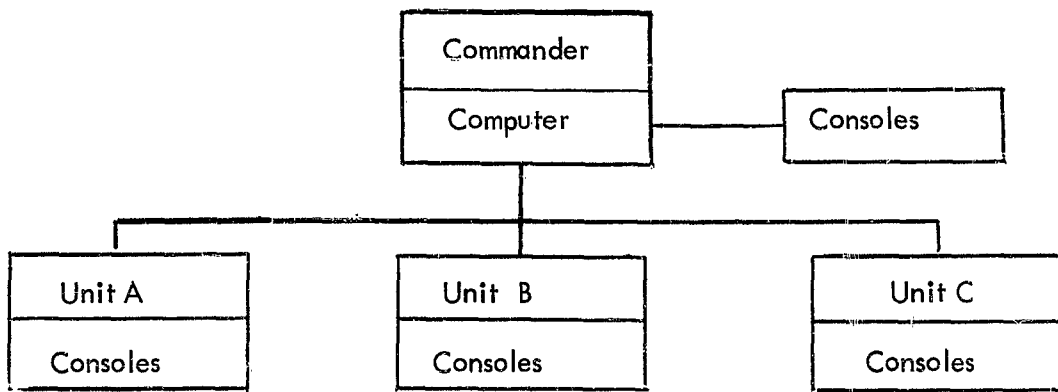


Figure 2-19. Centralized Configuration

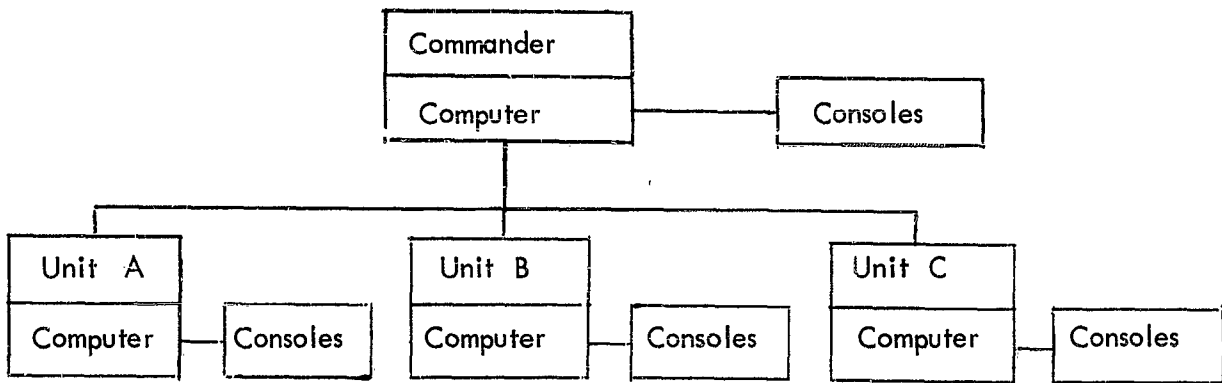


Figure 2-20. Distributed Configuration

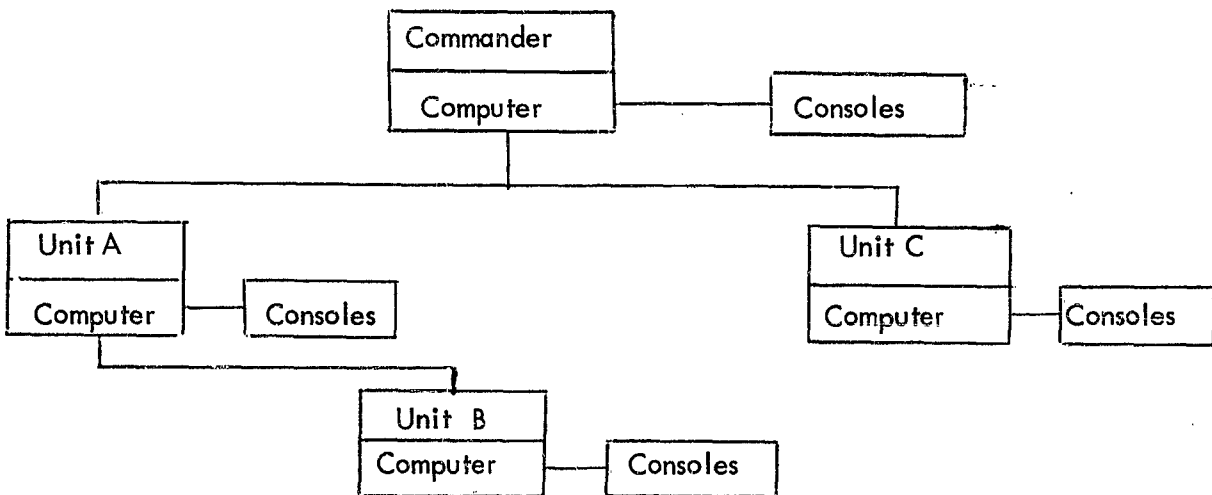


Figure 2-21. Mixed Configuration

When general purpose digital computers and general purpose displays are used in a command data system--different operational tasks can be performed at different times. That is, the system can perform different operational tasks according to its then current environment and the discretion of the commander.

The ammunition accounting function of a command data system could conceivably be used for pay computations and check writing during periods when a flagship was in harbor. This type of flexibility and multiple use possibility places a severe burden upon the naval system planner, both in design and evaluation. He must ensure that the best combination of flexibility, capability damage resistance, etc. is obtained in the system he plans.

The system of maximum total effectiveness is seldom the least expensive. The planner must carefully consider all of the costs and all of the effectiveness before advising his superiors. These highly complex mixtures of different tasks at different times using the same equipment are particularly hard to evaluate--yet they represent a very substantial operating capability to the line commander involved.

The same general purpose nature of modern computing and console equipment also allows the planner to provide for the future expansion of his system to include more operating units, more echelons of control or more operational tasks. Providing for the future expansion of the system calls for advance planning if the future changes are to be made with a minimum of disruption and cost. Very often the current provision of future capability to expand (additional input channels or extra power in display generators) costs more money in the initial procurement.

System planners must be very careful to take all of the costs for the life of the system to make maximum use of this inherent flexibility. What costs far less over 10 years or more of system life may cost far more during the initial years of procurement. The planner must emphasize total force and total system costing and effectiveness not only during the original procurement, but also across the life of the system.

2.9 COST ESTIMATION

2.9.1 General

Modern military operations require that planning decisions be based upon a thorough knowledge of the long range implications of those decisions. This is particularly difficult in an era when the decisions are concerned with the development, the procurement, and the deployment of large-scale command data systems. The day is long past when the senior engineer or military planner could easily approximate the costs of the system under consideration. The figures for command data systems are not readily available, but Large in June, 1963 pointed out, "Over a period of years, the final cost of a number of important weapon systems has been as much as ten times as high as the original estimates. Errors of this magnitude have caused a number of people to ask whether it is really possible to estimate development, procurement, and operating costs of future systems (which cannot be completely defined in advance) with sufficient accuracy to use these estimates as a basis for major program decisions."* Many specialists in the field believe that it is possible to make reasonably accurate estimates of future systems' costs. However, these costs cannot be estimated with accuracy without substantial detailed work and the use of specialized concepts.

The RAND Corporation has undertaken a great deal of work costing strategic bombardment and communications systems. However, there is very little work available on command data system costing. The RAND work known to be applicable to command data systems is referenced in this section as is the computer program costing work performed by System Development Corporation for the Electronic Systems Directorate of the Air Force. This latter work is the only available material on command data system computer program costing.

It can be seen from this scarcity of available material how elementary is the current state-of-the-art in command data system costing. However, enough data and techniques are available to give the naval system planner tools for his initial analyses.

* J. P. Large, ed., Concepts and Procedures of Cost Analysis, RAND Corporation RM-3589-PR, June 1963.

Continuing attention to this area will be required, since procurement programs (even very worthwhile ones) are often cut back to balance cost overruns or estimating errors*.

2.9.2 Cost and Economic Information System

In an attempt to provide a more widespread availability of cost estimating data the Department of Defense has established what is called CEIS (Cost and Economic Information System) by issuing DOD Directive 7041.1, July 7, 1964. The objectives of this system are: **

- 1) Improve cost estimating, cost and price analysis and progress reporting.
- 2) Enhance the effectiveness of planning, programming, budgeting, contract negotiating, and program or project management.
- 3) Provide data necessary for analysis of economic impact by geographic area and industry.

The scope of the proposed activity includes all phases of all DOD acquisitions at the system, subsystem, component and part level, and the CEIS system functions under the guidance of ASD (Comp.). The accumulation of these data and their appropriate indexing and retrieval is of great help to all cost estimators and analysts in the Government.

The Defense Department is providing training courses in the concepts and operation of CEIS. The courses are of 40, 8 and 3 hours in length, and are designed to acquaint the specialist, the supervisor, and the manager with the functioning of the system.

DOD is also requesting the Air Force Institute of Technology, School of Systems and Logistics to expand their training during FY 66 in cost estimating and cost analysis. At the present time AFIT offers a five week course in cost estimating and a 12 week course in advanced cost analysis. These courses are open to all military and civilian personnel of the defense establishment.

* Hon. Robt. S. McNamara, Sec. Def., Statement before the House Armed Service Committee, January 27, 1964.

** July speech of Mr. Chas. Hitch, ASD(Comp), introducing DOD Directive 7401.1, July 7, 1964, "Cost and Economic Information System" to Senior DOD personnel.

2.9.3 The Approach to Cost Estimating

From an academic standpoint, there are two basic approaches to cost estimating: the accounting approach, and the engineering approach. In actuality, a combination of the two approaches is employed. Each has its shortcomings and strengths.

Accounting cost estimation techniques are based upon accounting records which show what charges have been made to which accounts during the production or procurement of some system or component in the past. The charges are then adjusted and extrapolated to apply to the system being contemplated.

Accounting records and analyzes the transactions of a business. To function in a meaningful way, it must be regular and methodical. To accomplish this, it must make regularizing assumptions to smooth the fluctuations of normal business into the confines of a methodical reporting system. The errors possible in using accounting data spring from extrapolating these regularizing assumptions (made for one system in the past) into the future (to be used with a different system).

The two major stumbling blocks are the use of past burden rates and cost allocations for estimating future system costs without a detailed analysis of exactly how these rates and costs were established. This problem is recognized by the professional system cost estimator, who often calls himself a system cost analyst for this very reason.

Engineering cost estimation techniques are based upon the use of experienced engineers to plan in detail how a certain system will be produced. The stages in production; assembly, test, shipping, installation, etc., are all planned in detail. Costs are assigned to all operations; overhead and general and administrative costs are computed. Production quantities and schedules are estimated.

Engineering cost estimation is expensive since it requires the expenditure of so much specialized manpower. This style of cost estimation also has its sources of possible error. These are based upon the difficulty of foreseeing accurately what must be done in the future to place the system in the field.* It is not possible to forecast with certainty the changes which will be made to the production cycle to improve its efficiency. It is also difficult to foresee with accuracy where production problems will occur, or the expense required to solve them.

Most sophisticated cost estimates are produced through the use of both cost accounting and industrial engineering techniques. Past cost records are thoroughly analyzed, and production processes are planned in some detail. Future overhead and administrative costs are estimated and then compared with past records. Wages are inflated by national or industry averages. By skillful use of these two techniques the estimator can increase the accuracy of his costing, but there is no shortcut to a valid system costing. Substantial amounts of highly-skilled manpower are required.

Finally, when the component, subsystem, or system costing is completed, it is compared with one or two costings of similar systems as a check on its approximate accuracy. This constant need to compare and thoroughly analyze cost data on similar processes from many sources makes the data bank of CEIS invaluable for naval systems planners.

2.9.4 Fundamental Factors in Cost Estimating

Costing should emphasize differences - the fundamental purpose of costing is to aid the system planner or manager in making a choice between alternatives. It is at least as important for him to know where the cost differences between two alternatives lie as it is to know the total cost of each alternative. In the second case, he can only tell what his total expenditures could be. In the first case, he also knows what features of the two systems generate the differences.

* The difficulties of forecasting future system problems (and therefore costs) exist with accounting techniques also. However, the biggest problem with accounting is that it is occasionally not an accurate representation of what took place in the past (due to the regularizing assumptious mode).

By emphasizing differences, cost studies can be made at varying levels of detail to economize the use of time and manpower. In system features where alternative systems are insignificantly different in cost, relatively coarse-grained estimating should be used. In system features where there are more tangible differences in cost, finer-grained, more thorough work should be done.

This may seem to be the reverse of good logic, but there are two good reasons for the concept. First, it is the details of the differences which supply the most information to the manager, not the details of the similarities. Second, these differences in cost will be checked against other data, such as effectiveness measurements, production schedules and the needs of the user. It is necessary to have fine-grained data to examine closely what would be paid for those features and what advantage would be gained by buying them.

Non-dollar or other costs - For each new system there is a very substantial set of costs which it is difficult or impossible to evaluate in terms of dollar expenditures. Most non-dollar costs have their greatest impact upon the using command and its supply (or maintenance) organization, and not upon original procurement. This, combined with their non-monetary nature, allows them to be overlooked easily.

Operational costs are those costs (in terms of inefficiency, morale and general disturbance) which accrue to the operational unit receiving the new system or being connected to it. Although a few of these costs may be stated in dollar equivalents, great care should be exercised not to assign a dollar cost to some problem which is unacceptably big to the line commander involved. The ability to state a dollar value doesn't make the real cost acceptable.

The most important non-dollar cost of installing a new system is its interference with the tactical efficiency of the line unit involved. This ranges from putting a ship in the yard for fitting out to the time it takes to get from the final exercise to peak tactical efficiency.

The cost to the line unit in terms of lower tactical efficiency after fitting out is considerable, and one which is very difficult to measure. After the prescribed training there is still a lapse while the commander, his staff, and the operators get the correct feel for the new capability. Each new system improvement brings a cost of this nature. This is one good reason to limit the number of annual major changes for each tactical unit. This cost may differ substantially between system alternatives.

Training costs may be partially expressed in dollars when personnel can actually be identified as being pulled out for assignment as instructors or students. Many training costs remain hidden within the tactical unit. Tours for visiting officers and scientists, familiarization lectures, on-the-job training for officers and operators are all part of training costs which normally remain as non-dollar costs. For certain system alternatives these costs may differ a great deal.

Personnel costs arise from the impact of abrupt change, sporadic training and the problem of mastering one change after another with little intervening time to relax as a competent professional on the job. These costs are reflected in lower re-enlistment, requested transfers, and resignations from the service. While most of these costs arise from the process of change itself, there can be wide differences in impact between proposed system alternatives.

Scarce resource costs arise from the use of certain naval resources by the system alternatives under consideration. There are only so many exercise areas. There are only a few Naval Shipyards. There is a limited number of Naval Training Centers, etc.

In complex systems, such as ACDS, a number of these types of resources is required by each system alternative. When a manager evaluates the costs of system alternatives, he must take into account their requirements for those scarce Naval resources. They are scarce resources since more money added to the program will not readily provide more of that resource. The dollar cost of providing these scarce resources may be estimated on a pro-rata basis. The real cost to the Navy is its being deprived of some future choice as a result of having previously committed some part of these scarce resources.

The biggest difficulty with non-dollar costs is not their non-monetary nature. Their distance from procurement and design activities often leads to their being overlooked entirely. Once they are considered, careful professional judgment is adequate to treat them.

2.9.5 Sunk Costs

Costs start from now. What has happened in the past, the funds that have been committed so far, the funds that have been spent so far; these things have all happened regardless of what managerial decision is made now. Regardless of which system alternative is chosen, or even if no alternative is chosen, these expenditures are already committed to be made or have been made. These costs are called sunk costs.

Assume for example: The Navy has purchased for \$1,000,000, a large plot of land to construct an ACDS Training Center. The buildings have been designed and will cost an additional \$1,000,000. Before the buildings are built, a surplus Army base in the same area becomes available from GSA. The cost of improving that installation will be \$750,000. The Navy wisely turns over the previously purchased land to GSA, spends the \$750,000, and saves \$250,000. The question now is: What was the cost to the Navy of the new ACDS training facility, \$750,000 or \$1,750,000? The answer is \$750,000, since the previously spent money had no effect on and was not affected by the decision to utilize the surplus Army base. The \$1,000,000 is a sunk cost.

In exactly the same manner, those future commitments or expenditures which will be made regardless of which decision is made now are sunk costs as far as this decision is concerned. How the system planner deals with these problems is not quite so clear.

For example, the Marine Corps is required by the Congress to maintain a certain personnel strength. Until or unless the Congress changes this requirement, a certain number of Marines will be recruited each year, will be promoted, will retire, and eventually die. This is without regard to the duty they are assigned to. To a certain extent, all of these costs are sunk costs for the Marines. They are going to maintain this strength, regardless.

When the Marines consider the costs of implementing command data system alternative A versus the cost of command data alternative B, they must consider the cost of assigned personnel. It is carrying the sunk cost concept beyond reason to claim that all the personnel costs are really sunk costs since these personnel would have been Marines anyway. But the limit to reasonable personnel charges would seem to be the active duty assignment to the system. Recruiting costs, Boot Camp, retirement costs and Veteran's Administration benefits seem to be sunk costs and not reasonably chargeable to system A or system B.

Some personnel costs may be thought of as the cost of diverting scarce resources. There are only so many Marines. Those that operate command data systems cannot operate mortars or aircraft. The trade-off in scarce resource cost must be carefully considered. These types of sunk costs are very difficult to deal with. All that this section can do is to mark them for careful attention.

2.9.6 Total System Cost Estimating

The concept of total system cost has developed in Government circles in the last 8-10 years as a direct result of the need to obtain the complete costs of alternative weapons systems so that appropriate managerial decisions could be made. The same concept has been used in sophisticated civilian industrial circles for a somewhat longer period of time. Its spread to governmental use had been hampered by the annual budget concept, but the advent of the DOD Five Year Force Structure and Financial Program (FYFS & FP) has required its use in the cost estimating for most new expensive systems.

Briefly, the concept requires the collection of all costs for all parts of the system* during the entire useful life. This is not a startling or unreal concept, but it does require careful consideration of all stages of system planning, development and use, and of all the possible cost contributions to each stage. The costs are normally grouped into three categories with regard to their occurrence in the system life cycle:

*We are speaking of command data systems here, specifically ACDS. However, the principles remain the same for other systems.

- 1) Research and development costs,
- 2) Investment costs,
- 3) Operating costs.

Research and development includes all of the costs required to prepare a system for procurement and deployment to operational units.

Investment includes the costs of procuring: all operational and support equipment, all facilities and structures--ashore and afloat, all software, all initial spares and replacement units, all initial training and testing, and some miscellaneous charges, including the original deployment of operator and maintenance personnel.

Operation includes all of those recurring costs which are required to keep the system in operation during its lifetime, such as: replacement of equipment, facilities and software, maintenance of those items, pay and allowances, continuing training, spares replacement, magnetic tape, and punched cards.

The costs for these items must all be estimated based upon the specifications of the system alternatives and the doctrine and policies under which the alternatives would be employed. These doctrines and policies would specify the following data:

- 1) Schedules of development and deployment.
- 2) Final number of nodes or units deployed.
- 3) Manning requirements and schedules.
- 4) Maintenance concept and channels.
- 5) Training requirements and schedules.

The cost estimators and analysts then aggregate the estimates for the various alternatives using techniques which tend to isolate and detail the difference between the system alternatives.

It is particularly important to cost all system alternatives across the estimated lifetime of the system. As was discussed in Section 2.8, certain very valuable system characteristics can cost more initially but cost much less over the life of the system. Other system characteristics may cost more initially but enhance the effectiveness of some other system (such as a weapons system) to the extent that the total lifetime cost for a given mission will be reduced. This point leads us to a discussion of total force cost analysis.

2.9.7 Total Force Cost Estimation

There is no clear-cut dividing line between system cost analysis and force cost analysis except that forces are made up of numbers of systems augmented by some non-system activities, such as training centers, supply ships, airfields, etc.

Most non-system costs and shared costs may be dealt with more easily, if we can stop trying to prorate them among various systems, and simply assign them directly to the force which they support. Much fiscal planning performed in support of the FYFS & FP is at force level and is simplified by the use of these conveniences.

One problem in estimating the system costs of ACDS is in the proration of shared costs. For example; how much of the task force's supply mechanism may be charged against an Amphibious Task Force's ACDS can become a detailed and difficult question. If we are supplied with the right data, it is often simpler and more accurate to cost the task force as a whole, first with one alternative--then with another. It is certainly realistic to procede in this manner, for ACDS has no purpose except to support a commander, and that commander must command something. This note of reality in costing is to be looked for, since at its best, costing is still burdened with accounting, economic and engineering assumptions.

One of the goals of ACDS would naturally be to improve the efficiency with which some naval force is applied. It could well be that the total lifetime cost of one or several naval missions could be reduced substantially. Indeed, since ACDS has no operational force of its own, a substantial part of its cost and effectiveness evaluation will depend upon the increased response or efficiency it generates in the forces it controls or coordinates.

Some systems, for example NTDS, probably should be costed (and evaluated for effectiveness) as a component of a task force, a task group or of a screen. Not only can the assumptions for prorating many shared costs be eliminated, but also the effectiveness measurement may have more real meaning. Of course, if the role of any system is partially or predominantly single-unit operation, it should be evaluated for effectiveness in that role as well.

"...total force cost analysis refers to the costing of many different "mixes" or combinations of systems and non-system activities, so that the total costs of various real or hypothetical force structures can be compared. In addition to its inclusive character, total force cost analysis emphasizes the specific timing of requirements for funds and other resources. In its more limited sense, total force cost analysis refers to the costing of particular systems in the context of a force structure otherwise more or less fixed. The cost of a system thus becomes a marginal cost--the change in total cost caused by the addition of the system to the force structure."*

2.9.8 Cost Estimating Relationships

Thorough and effective cost estimating must be based upon the systematic collection and analysis of data on current, future and past systems and projects. These data are analyzed and correlated to provide Cost Estimation Relationships (CER's). These are also called ER's (Estimating Relationships).

An estimating relationship is a quantitative expression of the way in which one system variable affects one or more others. For example: to man one console operator position around the clock for one year might require 4.75 operator man-years to provide for rest periods, mealtimes, off-duty hours, sick leaves, and leaves. This ratio would be an ER or CER. Its use allows the cost estimator and system planner to accumulate data rapidly on the total operator requirements once the number of manned positions is known.

*David Novick, System and Force Cost Analysis, RAND Memorandum 2695-PR, April 1961, p.59.

ER's may be of several degrees of accuracy. If the last order of aircraft X from manufacturer Y cost approximately \$60 per pound, then aircraft W should cost approximately that after adjustments are made for differences in aircraft type, avionics cost, economic trends and the past relative costs of manufacturers Y and Z. Relationships of this nature are of great value to system planners, although finer-grained relationships are required also.

The first concern in developing ER's is the collection and analysis of field data. This is a time consuming job which the recent establishment of CEIS should make much easier. The analyst must then check the accounting assumptions used for the changes made, as well as adjust levels of detail. Different source agencies accumulate costs at different levels of detail. The analyst must be completely knowledgeable with regard to what was included in each charge as well as what was not included.

After this phase of data gathering, the ER's are calculated. Some are easily done by hand. Others involving large amounts of data must be calculated on computers. The resulting ER's may be presented in tables or they may be shown as mathematical equations relating the change in two or more variables. An example of a simple formula might be:

$$\begin{array}{l} \text{Support} \\ \text{Personnel} \end{array} = 500 + 0.4 (\text{Direct Personnel})^*$$

(For a specific type of system at a specific echelon of employment)

Many ER's, of course, are quite complex, but their use allows the system planner to estimate certain costs with great speed. In addition, since they are normally based upon more than one system's experience, they can provide a better set of base data from which to extrapolate.

* See R. L. Petruschell, An Introduction to Estimating Relationships, RAND RM-3215-PR, June 1962

2.9.9 Personnel Costs

One of the more difficult areas to estimate for command data systems is that of personnel costs. Although strategic weapons systems must be manned according to a very rigid plan to remain n% effective on a 24-hour basis, command data systems may be manned on widely varying bases from those originally planned and still remain quite effective.*

In addition, while the local commander has little choice as to how to man an aircraft or an AAW missile battery, he may easily make substantial changes in the manning of his CDS to suit his style, his mission and his available manpower. The estimating problem is not really one of finding the costs of the men required, but of finding the numbers of men required. The system planner will find some previously developed concepts to be of considerable assistance.

The first requirement for personnel estimating is to understand thoroughly how the system will be employed throughout its operational deployment to the user. The numbers and types of personnel required during stand-by, planning, combat and various alerts must be well understood. This must come not only from the system specifications but from knowledgeable line commanders who will use the system. Maintenance and support requirements must also be computed in detail. It has been helpful in costing systems to think of the personnel requirements as having those three parts (operational, maintenance and support).

In addition to operation, maintenance, and support personnel, many service personnel will be used in installation. This will be particularly true of ACDS installations made in naval shipyards. Complete checklists will be required of all types of installation personnel, their effectivity rates, shipyards overhead. Compounding this problem is the variation in effectivity, overhead and wages among the various shipyards. Some ACDS equipment might be installed by private shipyards or contractors and this will require additional cost records to be collected and developed for CDS type work.

* An example of this is found in SAGE. The console manning originally thought to be constantly required is now only approached during periods of alert.

2.9.10 Computer Programming Costs

Computer programming costs have been major parts of total system costs for command data systems; accounting at times for as much as one half of the total cost. General Terhune* has stated that the per instruction cost for SAGE instructions has varied from \$32 to well over \$100, depending upon whether they were included in regularly produced program models or were rush changes expedited into the field.

Such variability in an important cost gives any system manager serious concern, and the major factors determining program costs must be considered carefully.

In most instances where data is available, computer programming costs have been underestimated. There are a number of reasons for this but the foremost are:

- 1) Computer program cost estimates were made by non-programmers.
- 2) The scope and magnitude of the ultimate task were not known.
- 3) The changing nature of system requirements were not known.
- 4) There was little knowledge of the detail factors which affect the costs of programming.

It is possible today for a few business-for-profit contractors to bid on certain programming tasks on a fixed price basis. One contractor warantees that the programs so produced will operate under the specified conditions**. Program errors are fixed without charge. From this, it can be seen that it is possible to estimate the cost of some programs under some conditions. Let us look at some of the reasons for past (and current) poor performance.

Program costs must be estimated by programmers. Only an experienced programming supervisor with extensive costing experience can make an accurate estimate. Economists, accountants and engineers cannot recognize the subtle differences in requirements that spell the difference between an easy task and difficult one. Only a senior programmer can ask those critical questions which provide for efficient program design. Since program costing is performed by the comparison or analogy method (with a few estimating relationships sometimes used), only an experienced programming supervisor can realize what apparently similar programming tasks are, in reality, analogous.

* Commanding General, Electronic Systems Directorate, USAF

** Informatics Inc.

An additional factor confuses the issue for the lay estimator. Productivity rates and wages received fluctuate greatly from contractor to contractor. Individual productivity may vary by a factor of 10 in a large group of programmers. In small groups, it seems to remain more constant within the group, but varies from group to group. Only the supervisor who will direct the task knows the real caliber of the personnel who will perform the work. Other programming supervisors are still able to make worthwhile estimates--not for fixed-price bidding, however.

Scope and Magnitude must be defined. The specifications for a radar must be known accurately before an accurate cost estimate can be made. The scope and magnitude of the programming task must be known before any accurate cost estimates can be made. This seems too evident to need comment, but in many systems the programming requirements are developed after the hardware is designed (or even purchased). At short range a hurried program costing is made and subsequently it is found to be far too low. This can only be remedied by alert system management action.

Changing character of the system not known. Many systems whose programming costs have been painfully high were never conceived of as evolutionary systems, but technological changes and threat changes forced them to become at least partially evolutionary. This modest evolutionary capability has been provided almost entirely by re-programming in most cases.

This characteristic of evolution will be planned for in ACDS, and the costs of the computer programming must be planned for also even if they cannot be accurately estimated.

Little recognition of the important cost factors. Some of this stems from lay estimating and inexperienced professional estimators in the days when there was no experience. There are a number of important variables which are not immediately apparent (such as programmer effectivity and efficiency). One of these is documentation.

For small commercial and scientific programs the cost of documentation is negligible. For large command data system programs, the identifiable documentation costs can be as high as 20-40% of the programming costs.

A broad basis of experience has been established in the programming profession and estimating accuracy is improving in those economic environments which encourage or demand it.

The first openly published investigation of the determining factors of program costs was sponsored in succession by DOD Advanced Research Projects Agency* and USAF Electronic System Directorate.** The results of this study indicate that it should be possible for any computer programming activity to develop reasonably accurate cost estimating relationships. In addition, Farr, et al. show how this can be done.

In Farr's study of one programming agency, the variables most highly correlated to the man months required for program design, code and test were:

- 1) Number of originally estimated instructions required
- 2) Complexity rating of program (range 1-5)
- 3) Number of external document types
- 4) Number of internal document types
- 5) Number of words in data base (\log_{10})

Figure 2-22 shows the cost estimating relationship which uses these variables. It should be noted that this precise equation applies only to the agency studied by Farr. The form of the equation should be examined by all programming agencies for possible application to its estimation tasks.

* OSD-97

** AF19(628)-3418 ESD; and Farr and Nanus, Factors that Affect the Cost of Computer Programming, SDC TM-1447/00, June 1964; and Farr and Zagorski, Factors that Affect the Cost of Computer Programming: A Quantitative Analysis, SDC TM-1447/001, August 1964.

$$M = 2.71 + 121 C + 26 E + 12 D + 22 B - 497$$

Where:

M = Man months to design, code and test program

I = Thousand of instructions in original estimate

C = Complexity rating (1 to 5)

E = Number of external documents

D = Number of internal documents

B = Number of words in the data base

Figure 2-22. The Five Most Predictive Cost Variables in Computer Programming, Shown in Their Prediction Formula (From Farr, et al.)

2.9.11 Uncertainty in Costing

Since system planners and system managers must have system cost estimates made, it is important to have some idea of how inaccurate they might be. The only quantitative studies available show that average system cost estimation errors may be very high, ranging from 200 to 400%. This magnitude of error is considerably better than the 10 to 1 error cited by Large in Section 2.9.1.

There are two sources of this error. The most important error (by a factor of about 10) is that of requirements uncertainty. This error stems from the fact that cost estimation done in the very early stages of a system's planning is subject to a great deal of uncertainty. As a system's design, manning, and schedules develop, there is more certainty as to what is planned and that the plans, as known, will be carried out.

Early system cost estimates, upon which important decisions are made, depend upon incomplete plans and designs which in themselves are quite subject to change. The program definition phase has as one of its purposes the improving of the detail and accuracy of system requirements and design so that the resulting cost estimates may be more accurate.

The second source of uncertainty comes from the estimating process itself. Fisher* cites one study which examined the ability of skilled cost estimators to cost several types of items, some simple, some complex, but all well-known in advance. The results of that study show:

- 1) Variation in cost estimates of manufacturer producing landing gear: average error $\pm 23\%$
- 2) Variation in cost estimates of public engineers estimating construction projects: average error $\pm 15\%$
- 3) Variation in contractors bids for public construction projects: average error $\pm 21\%$
- 4) Variation in direct labor costs and airframe costs for 21 aircraft: average error $\pm 20-25\%$. Specific errors ranging from -40% to $+70\%$.

Given good specifications and base data, estimation can be quite accurate, but it should not be expected that system cost estimators will get much closer than $\pm 35-40\%$ on original or very early estimates for large systems.

Again the advantages to evolution become evident. Smaller increments with more accurately known designs and schedules are amenable to much better costing than large indefinite systems with fluctuating schedules.

Certain cost estimating tools (such as PERT/COST) have been developed for use on computers. These help the system planner and system manager develop a better feel for how much uncertainty is in the system cost estimate and where it comes from.

* G. H. Fisher, A Discussion of Uncertainty in Cost Analysis, RAND RM-3071-PR, April, 1962.

2.9.12 Electronic System Cost Models

Since there is so much clerical work involved with making system cost estimates, it is natural that estimators have turned to computers for assistance.

PERT/COST is a generalized tool which allows the various uncertainties in each cost estimate to be accumulated statistically and presented to the system manager.

Generally, this technique is considered to be an in-process control tool to be used during the implementation of a project to control costs.

There is another type of computer assistance being experimented with at present. This is the electronic system cost model. One has been developed for the IBM 7090/7094 by the Mitre Corporation, and one is under development at the Navy Computational Laboratory at Dahlgren, Virginia.

The system cost model is a computer program which may be provided with all of the specifications of an electronic system and all current estimating relationships. Subject to the details of the program it provides a cost estimate of the specified system.

The loading of the original data into the program is nearly as tedious as computing the costs manually once. The advantage of a system cost model comes in its ability to answer rapidly questions as to the costs of closely related system alternatives.

The system planners for ACDS should investigate the creation or borrowing of computer-based electronic system cost models for use in ACDS planning.

2.9.13 Discussion

Suitable system costing methodology is available for Navy system planners to provide satisfactory cost estimates for ACDS purposes. Much of this methodology exists in areas outside the Navy as well as within the Navy. The cooperation which has been stimulated by the support of DOD Directive 7041.1, dated July 7, 1964 (Cost and Economic Information System) should serve to accelerate the interchange of costing research and techniques among the Services, DOD, and outside agencies such as RAND and Mitre.

* Glazer, M. and Janssen, T., Electronic System Cost Model, Mitre TM-3364.

The use of the CEIS data base (when available) and the careful system definitions required in the program definition phase should increase the accuracy of ACDS system cost estimates over the inglorious historic average. Continuing ACDS system costings will probably benefit from the fact that evolutionary increments are normally well specified before costing.

There are a number of cost accounting and cost analysis groups operating in the Navy today. While they are aware of each other, work steps should be taken to collect and disseminate their concepts and techniques more widely within the Navy and DOD. Perhaps a Navy System Costing Manual could be provided to senior officers and Navy system planners.

Several of the original senior ACDS system management nucleus should have had at least a few weeks specialized training in Naval and DOD system cost accounting or analysis. Many of the important system management decisions will be made early. Training when time is comfortably available will be too late for some purposes.

Continuing research needs to be done (probably on a small scale basis) to develop more effective estimating relationships for electronic and communications equipment.

The work begun in computer programming cost estimation by FARR, et al. should be completed for application to the estimation of Naval computer programming costs. Continuing work needs to be done in reducing, coping with, or factoring out the uncertainties which seem inherent in system costing.

ACDS system planners should borrow or develop a computerized system cost model suitable for use in ACDS costing.

2.10 EFFECTIVENESS MEASUREMENT

2.10.1 General

The decisions made by naval system planners require effectiveness data as well as cost data. But measurements of military system effectiveness cannot be expressed in dollars as can costs. This lack of common units of measurement makes it very difficult for the planner to compare costs with effectiveness. The same lack makes it difficult to measure effectiveness in the first instance and even more difficult to portray the results of the analysis.

The command data system, as exemplified by ACDS will be particularly difficult to evaluate except in terms of how it affects the speed, striking power, effectiveness, etc. of the force it controls or coordinates. This section discusses the outstanding problems of effectiveness evaluation, and presents a new technique particularly applicable to system effectiveness measurement of ACDS.

2.10.2 Military Usefulness

System or item effectiveness is not the correct criterion by which to make system planning decisions. The criterion which must be used is that of military usefulness. Military usefulness considers both the effectiveness of a system in performing its tasks and the military value of having those tasks performed. The military usefulness of the most effective system is zero if the value of that system's tasks is zero.

The determination of the military value of performing a given mission or set of tasks is probably not subject to numerical measurement--and it should not be. It is the responsibility of senior naval officers and their civilian counterparts to determine the relative value of performing various missions and tasks. They must do this using their professional judgment and experience. The task of evaluating effectiveness may be entrusted to competent analysts. The task of determining military value must be retained by those few senior professionals with the experience, training and responsibility for the task.

There are three types of usefulness comparisons according to the items or systems compared. They are:

- 1) Comparisons made within one system,
- 2) Comparisons made between systems having similar missions,
- 3) Comparisons made between systems having dissimilar missions.

Comparisons within a system - In this type of analysis, one item, component, or subsystem is compared with another, both being designed for use in the same system with an unchanged mission. In this analysis, military usefulness depends entirely upon the relative effectiveness of the things being compared since the mission (and therefore, its value) is unchanged.

This type of analysis is used to determine whether repair parts, pluggable units or both should be used to repair ACDS nodes at unit level. Another example is the evaluation of two different intership ACDS data links. In all cases of this nature, the analyst is comparing things very much alike, and the errors of measurement can be rather small.

Comparison between systems having similar missions - In this analysis more senior military judgment is required since the missions of two systems being compared (hence their value) are seldom identical. When they are, however, effectiveness measurement alone will suffice. As the similarity between missions decreases, the role of the senior military professional increases. It now becomes of less relative consequence how effective each system is, and of much more consequence how valuable the performance of each mission is.

Comparisons between systems having dissimilar missions - In this analysis effectiveness measurement is of still less consequence. The matter is now one of which mission is more valuable and by how much. Each of the alternatives becomes an outstanding contender through the process of being evaluated against similar systems. The effectiveness measurements should already have taken place.

Comparisons of this type involve such things as updating three AGC Command Nodes to ACDS configuration as opposed to spending these funds on the COMPHIBPAC training facilities at Coronado.

The Naval system planner must scrutinize all effectiveness studies to ensure that the proper increasing consideration is given to the military value of the mission as the missions become more dissimilar. At one end of the spectrum effectiveness measurement is all that is required. At the other end of the spectrum effectiveness measurement is of much less consequence, and military judgment predominates.

2.10.3 Fundamental Factors in Effectiveness Measurement

Thorough Documentation - Effectiveness studies must be thoroughly documented. If they are not, their value to future system planners may be entirely lost. Cost studies may be recreated by going back to the original cost data, but effectiveness studies must create their own material, and if this is lost as blackboards are erased and notebooks destroyed, there is no way of examining the interior of the study or of validating part of its conclusions for some other application. An additional problem exists in that effectiveness studies are very often conducted by special groups of officers and analysts. Once the study is completed, the group is disbanded. If complete documentation is not available, every step will have to be retraced at some point in the future.

There is an additional reason to document carefully all effectiveness studies. Since the procedures, data base, assumptions, definitions and computations of all effectiveness studies must be created from scratch at the inception of each study, it is not too difficult to warp the numerical results of an effectiveness study to fit some pre-conceived goal. The easiest way in which to escape the accusation of having done this is to have published, in detail, all the requisite data before the study is completed.

Assumptions - It is necessary to make a number of assumptions before an effectiveness measurement study can really begin. These cover such items as the caliber of personnel and training being used to man the system, the operational doctrine under which the system will be used, the replacement and maintenance concept, etc. Assumptions must be made about all conditions which will affect the efficiency of the system under consideration. A few of these assumptions will be critical with regard to the results of the study. These assumptions must not only have some basis in fact, but must also be recorded in sufficient numerical detail to complete the study. Occasionally, the original set of assumptions is not adequate for the study either in detail or in terms of area covered. These new assumptions must be adequately documented as soon as they are made.

Basic Data Sources - As important as the assumptions, are the sources of the basic data upon which the study started. Some of these data sources may cover the assumptions made, others may not. The appropriate documentation of data sources not only attests to the validity of the study but also enables future analysts to update the study when new basic data becomes available. The same ability to update the study applies to appropriate documentation of assumptions and definitions.

Definitions - The purpose of an effectiveness study is to measure the performance of the items at hand with regard to certain characteristics. These characteristics must be rigorously defined and specified at the beginning of the study. Such concepts as flexibility, reliability and operability are far too open-ended to be meaningful unless they are accurately (and numerically, when possible) defined at the beginning of the study.

Failure to define adequately those characteristics being measured in the study leaves even the most numeric and professional study open to serious question.

Procedures and Computations - It is also necessary to document the procedures and computations through which the results of the study were obtained. Given the same assumptions, definition and basic data, different procedures and computational mechanisms will provide different evaluations. So that subsequent references to the study may be of some value, the procedures and computations used must be detailed.

It is not necessary to suggest the importance of thorough documentation of the results of the study. It is advisable to create a number of master documents which have under one cover, the entire history, documentation, and data base of the project, for the simple reason that if one of the above parts is mislaid, the balance of the study becomes almost valueless. This is difficult to accomplish for an extremely involved study but the creation of one or two giant sized volumes is well worth the effort.

Emphasize the Differences - In the same way that it is important for the naval system planner to point out the cost differences between two system alternatives, it is important for him to point out the effectiveness differences between these two alternatives. ~~Again, it is not~~ factors of sameness, but factors of difference which prompt the decisions made by the system planner.

Not only must the effectiveness study emphasize the difference between the alternatives, but it must relate the effectiveness of the system alternatives to the requirements imposed by the mission. In other words, the alternatives are not really compared with each other, but they are each compared with a common standard; that is, the requirements imposed by the mission.

There are two good reasons for arranging the comparison in this way. First, we are interested not in whether A is better than B, but whether or not either will accomplish the task at hand. Second, there are many instances in which, after having compared alternative A and B, we must withdraw alternative A and substitute alternative C. If A and B have been compared with each other instead of with the requirements, a new study must be generated. If A and B have been compared with the requirements, a small amount of additional computation will allow alternative C to be added to the study.

When a study is initiated, it is difficult to tell how many times it will have to be reworked. Comparing alternatives with the requirements instead of with each other is the most efficient manner of computing the results.

2.10.4 Total System and Total Force Effectiveness Measurement

Total System Effectiveness - It has been standard practice for some time to employ the total system concept in the measurement of effectiveness. This consists of attempting to measure everything which contributes to the effectiveness of a system in performing its required mission. It seems to be common analytical instinct to follow this course in effectiveness measurement. Perhaps the only warning needed is that the effectiveness of a system often changes substantially during the life of the system. This factor of changing effectiveness must be considered carefully. Occasionally, the value of the mission changes more rapidly than the effectiveness of the system itself.

With regard to command data systems and ACDS, it is difficult to evaluate them at system level since their mission is to control and coordinate other systems. They are also difficult to evaluate since their ranges of acceptable operation seem to be so broad. Often, their important characteristics are largely non-numerical.

Subsequent sections present some concepts and techniques for measuring command data system effectiveness at the system level. These approaches may also be applied to other than command data systems and may be used at the subsystem and component level, if desired.

Total Force Effectiveness - The concept of evaluating systems in terms of their total contribution to a force was discussed in Section 2.8 and is reasonably well known throughout the military community. Total force evaluations are tedious to make since they normally require operational gaming studies to be performed. This requirement for operational gaming seems to be especially true for command data systems such as ACDS.

There is an additional problem to effectiveness evaluation in the total force environment. It was mentioned in the NTDS AAW problem discussed in Section 2.8. Some portion of the system's mission is single unit operation. Other portions of the mission are multiple unit operation in various types of forces. Granted that new systems should be evaluated as much as possible in a total force environment, how much weight should be attached to the effectiveness of a system as an individual unit as opposed to the weight attached to its effectiveness as a part of several different forces? This is a difficult question to answer with respect to a system such as NTDS which is now in operation. It is much more difficult to answer for an ACDS evaluation to be performed in the system predesign stage. This problem must be considered very carefully during the design of any system effectiveness measurement.

A similar problem exists in measuring the effectiveness of a multiple mission system. For example, assume that some command node of ACDS will normally function as the command node for AAW, ASW, and STRIKE operations. After the node has been evaluated in each of its possible roles, some determination must be made of the relative importance of these roles to the evaluation of the system node. This again is a very difficult question and one which should be answered by the senior naval personnel involved and not by the analyst.

It is very difficult to evaluate the effectiveness of command data systems, and the evaluation of ACDS will be no exception to this. Total force evaluation will allow the comparison of the effectiveness of the total force during the operation of alternative A or B as the command node (or as some functional node as FAAWC). The difference in force effectiveness (if all other items are held constant) will be the result of having employed alternative A or B in its role. This is an indirect technique for evaluation but it is a very respectable extension of the concepts of parametric analysis which are discussed later.

2.10.5 System Characteristics

During the evaluation, every characteristic of the system alternatives must be given careful consideration. There can be almost endless numbers of these characteristics such as accuracy, range, availability rates, speed, response time, etc. Regardless of the large number of characteristics possible, they will be found to group themselves into three categories.

- 1) Operational characteristics,
- 2) Technical characteristics,
- 3) Support characteristics.

In general, systems performance can be evaluated by the answer to three questions:

- 1) Is it technically capable of performing some interesting fraction of the required mission?
- 2) Can it be operated in the field by service personnel and still perform substantially according to its inherent technical capability?
- 3) How easy is it to support in the field with service personnel?

After all of the meaningful system characteristics have been identified and placed in one of the three categories, each characteristic is assigned a weight. These weights should represent the importance of that characteristic to the using commander and the importance of that characteristic to the ability of the system to perform its missions satisfactorily. Weighting is highly subjective and requires that good professional judgment be applied.

Concurrent with the weighting process is the determination of those characteristics which must be present precisely as required for the system to be of appropriate military value. These characteristics are considered absolute requirements. Here, the analyst must be very careful. It is easy to state that a requirement for track resolution capability of a quarter mile at a range of 300 miles is absolute. It is also quite possible that instead of this being an absolute requirement, it is simply a design goal. The determination of absolute characteristics is critical to the conduct of an effective study.

System characteristics require extremely careful definition to ensure against the generation of unfair or ambiguous results.

2.10.6 Scalar Values

Scalar values are those numbers which show the amount or quality of each characteristic present in the system being evaluated. In command data systems, many system characteristics of importance are not numerical in nature. This type of characteristic is called irreducible, that is, not inherently capable of being reduced to numbers. A technique for treating irreducible characteristics is shown in the next section.

There are a number of command data system characteristics which are essentially numerical in nature. A scalar value is assigned to each of these numerical characteristics in the following manner.

Assume that for a certain ACDS node, the data transfer out rate for system A is 1,300,000 bps, and for system B is 900,000 bps. We have discussed previously the advisability of normalizing evaluations to what is required by the specification. In this instance, assume that the specification requires a transfer out rate of 1,000,000 bps. The scalar value for system A is 13, indicating (since these are normalized to the requirements of the specification) that system A has 1.3 times the amount of the characteristic required by the specification. For system B the scalar value assigned is 9, indicating that system B has 0.9 of the quantity of that characteristic required by the specification.

At the end of the evaluation, the scalar values for each characteristic are multiplied by the weighting factor for that characteristic and then added for all of the characteristics of each system. This produces a weighted score for each system alternative being evaluated.

Inasmuch as the weight attached to each characteristic represents the relative importance of that characteristic to the operational user of the system, raw scores have no meaning.

At this stage, the analyst factors out the absolutes. First, the analyst eliminates from further consideration any system alternative which does not have the minimum amount required of any absolute characteristic. It is often desirable to continue the evaluation of the balance of the system for the determination of base data. An additional reason for continuing with evaluation could be that future developments promise to provide the inadequate system with satisfactory capability in that particular characteristic.

The second aspect of factoring out absolutes is to remove from further consideration, those characteristics which have the same rating or scalar value for all the system alternatives under consideration. This is occasionally done for characteristics in which there is a small difference in capability among all the systems when this small difference is of no operational or technical importance.

2.10.7 Irreducible Characteristics

There are several characteristics of command data systems which are not normally thought of as numerical in nature. These are such things as ease of console operation, maintainability, convenience of command post arrangement, etc. To evaluate all the characteristics of ACDS alternative designs, some technique must be used to apply scalar values to the irreducible or qualitative characteristics.

Again, the scalar values should be normalized, (at 10 or 100) to that amount of the characteristic required by the operational specification. For irreducible characteristics, it is rather difficult to determine precisely how much is required by the specification. Careful professional judgment must suffice.

Applying scalar values to irreducible characteristics consists of arranging a set of adjectives which describe the characteristic. Beside this arrangement of adjectives, the analyst arranges a list of numbers from zero to ten or higher. Zero corresponds with "absolutely useless" or "inoperative", while ten corresponds with "exactly meets requirements". Those systems having characteristics exceeding the requirements are assigned numbers in excess of ten (or 100). An example is shown in Figure 2-23.

There is a strong temptation to simplify this process by rank ordering the systems in terms of good, better, best; and assigning a number 1, 2, 3 for each rank ordering under each characteristic. There are two outstanding errors in this technique. The first is that the systems are being compared with each other and not with the operational requirements. This was covered in detail earlier. Second, the number three is three times as large as the number one, and when this number is multiplied by the weighting factor for that characteristic, a disproportionate advantage will be given to the system which may be only slightly better than the systems given the values two and one.

10	Exactly meets requirements
9	Nearly or almost meets requirements
8	Very good
7	Satisfactory
6	Nearly satisfactory
5	Unsatisfactory, but complete output
4	Poor, but complete output
3	Poor, incomplete output
2	Very poor, some output
1	Extremely poor
0	Inoperable
Scalar Value	Adjective Descriptors of Performance

Figure 2-23, Example Arrangement of Scalar Values for an Irreducible Characteristic

2.10.8 Evaluating Non-Optimum Operation

Many thoughtful system designers are concerned over the tendency to conceive of and design systems for operation under optimum conditions. Concurrent with this tendency is the tendency to evaluate the effectiveness of systems at those optimum conditions only. In January, 1960, A. H. Katz wrote a paper which was published in February, 1962. This paper was subtitled "It's Easier to Ensure Against Success than Against Failure". The main theme of this paper was to caution system designers against useless sophistication in their systems. Katz included a tabular evaluation of two systems to demonstrate the shortcomings of too much elegance.

To prove his point, Katz evaluated the two competing systems in operation at the design point, under two conditions worse than the design point and under one condition better than the design point. The table, with some simplifying deletions, is published as it appeared in 1960. (Figure 2-24).

This technique of evaluation shows very clearly, critical system capabilities which would be overlooked if the two systems were compared only at their optimum design point. For example, system D is relatively insensitive to a poor environment, while system A may be inoperative under the same conditions.

System managers and system analysts should evaluate systems at other than their ideal design point. This is particularly true for future systems which plan to take advantage of improvements in state-of-the-art, and for systems which can be expected to operate in hostile environments. The tabular form shown in Figure 2-24 would be sufficient for the evaluation of small increments of change to ACDS. Larger increments would require a somewhat different presentation.

The obvious next improvement upon this tabular technique is the insertion of scalar values for the adjectives "very good", "poor", etc. This insertion cannot be made, however, without considering the probability (however subjective it may be) that these various conditions will occur. Section 2.10.10 presents a further development of this tabular or quadrille technique. The technique shown there permits the insertion of scalar values and probabilities.

	System A	System B
General Description	Elegant, Sophisticated, Highest State of the Art... Sensitive to environment	Crude, Simple, Brute-Forceish... Relatively insensitive to environment
Requirements	Requires: Good Orbit Good Stability Good Temperature Control ---	Does <u>not</u> require: Good Orbit Good Stability Good Temperature Control ---
Results if Requirements Met	Very Good	Very Good if those for "A" are met
Results if Requirements Not Met	Poor	Very Good
Results is Requirements Missed Badly	Horrible	Good
Results if Requirements are Exceeded	Very Good	Excellent

Figure 2-24. The Evaluation of Two Hypothetical Systems (after A. H. Katz, January 1960)

During the evaluation of every ACDS increment and of possible design approaches to ACDS itself, naval system planners should concern themselves with the effectiveness of the alternatives under minimum operational conditions. That is, what happens if one or more of the four computers in the computation node are out of action? What capability is left when one of the four data links is inoperative? Questions such as these must be answered for systems which are subject to battle damage and the damage from high sea states. In addition, such factors as the interference of pitching and rolling with data link transmission must be thoroughly investigated. These questions are not to suggest that every ACDS increment must have available a complete manual backup. It does suggest that questions of this nature be asked and answered during the process of evaluation.

2.10.9 Two Computer Tools for Planners

As systems planners and analysts become interested in evaluating larger systems and using the concepts of total force effectiveness measurement, clerical loads rise appreciably. The use of computer simulation allows analysts and planners to answer a number of questions concerning the performance of systems and of systems designs. Two general families of simulations exist. These can be thought of as internal or design simulations and external or operational simulations. They treat the same sorts of problems. Their primary difference is in detail.

Internal or design simulations may be used to evaluate each small portion of a system or subsystem design under certain conditions which the designer chooses. It may also be used by system analysts to evaluate an entire system under conditions which he chooses. The concern of this type of simulation is with the reaction of the internal technical design to the stresses upon the system.

This type of simulation is used to answer questions such as: With X messages arriving at rate Y at point Z, is computer configuration A or B or C most adequate? Questions of this nature are constantly asked then answered during the system planning process, and must also be answered during the evaluation of any proposed increment to the system.

One particularly valuable technique is known as parametric analysis in which a certain design is held constant while the parameters limiting that design are varied. The performance of the design under consideration is recorded for each of the changes in parameters; the analyst or designer can receive a great deal of information from a parametric analysis such as this. Another form of parametric analysis is to hold the parameters constant, vary the design of the system and record the performance of each design as the parameters are held constant. The use of computer simulation or analyses such as this makes possible the investigation of design alternatives for system change proposals which could not be evaluated using manual techniques.

The second type of simulation is external or operational simulation. This simulation is concerned with how the system reacts externally to external operational stresses. In this type of simulation, designers and analysts are concerned with questions such as: In the middle of strike route planning for Mission A, can ASW operations of X magnitude and AAW operations of Y magnitude be conducted? Another question would be: Given the previous conditions, can the Commander transfer task force command from node A to node B 15 minutes after the start of ASW operations?

Using manual analyses, only a few operational questions of this nature can be asked and answered. Using computer simulation allows system planners and system managers to get appropriate answers to larger numbers of evaluative questions.

2.10.10 The Modified Quadrille Technique

This technique is one which was developed* to allow the addition of scalar values and probabilities of occurrence to the tabular or quadrille technique shown in Figure 2-24. To summarize the important concepts of effectiveness measurement as well as to demonstrate the applicability of the modified quadrille technique to the evaluation of ACDS components, this section evaluates two hypothetical components for ACDS.

The hypothetical component chosen is a general purpose operator console of the Charactron tube type having a small amount of internal high speed memory but no computing capability. In addition to this being a hypothetical component, it is a hypothetical evaluation since

* By E. K. Campbell of Informatics, Inc.

only a very small number of characteristics will be evaluated. These are chosen to demonstrate how a complete evaluation might be organized.

Figure 2- 25 shows the requirements of the specification and the characteristics of the two components when operating under optimum conditions.

Figure 2-26 shows a modified quadrille evaluation made for four characteristics. Two of these characteristics were selected from Figure 2- 25 which showed quantitative characteristics and two characteristics are subjective characteristics (usually thought of as being irreducible to numerical values).

The left hand column of Figure 2-26 shows the characteristic being evaluated, and the second column shows the weight or relative importance of the four characteristics. The third column defines the possible perturbation of the characteristic being discussed. In each instance, the characteristics shown are subject to variation as the environment changes. Many characteristics of operator consoles are not subject to this change, but their consideration does not require the use of the probabilities which Figure 2- 26 is used to demonstrate.

The fourth column shows the probability of the particular perturbation occurring. These probabilities can be very subjective such as the ones shown here in Figure 2-26 or they can be derived from actual service data. In box A note that all probabilities for a given characteristic must sum to 1.0 or certainty. The next two sets of columns are broad bands in which the raw scalar values are recorded and the arithmetic is set down. An open format of this sort is to be desired since it allows any observer to check the computations involved or to instruct himself in the method by which the evaluation was made.

Since this is a hypothetical and an incomplete evaluation, no absolutes were factored out. Reference to Figure 2- 25 shows that scalar values were justified to system requirements. The required mean time between failures is 1000 hours, that being equivalent to a scalar value of 10. Console C is rated at 8 for 800 hours; Console D is rated at 11 for 1100 hours. Similar justifications to the numerical standards of the system requirements must be made in all evaluations. Mean time to repair is justified in this manner also. System C has an MTTR of 1/4 hour. This is four times as good as the requirement states, therefore, setting the requirement equal to ten, the raw scalar value for mean time to repair for System C under conditions of no perturbation is 40.

<u>Requirements</u>	Console C	Console D
General Purpose ACDS Operator Console	Small, Low Power Requirements, Good Switch Layout, State of the Art Engineering	Larger, Modest Power Requirements, Fair Switch Layout, Unimaginative Design
Mean Time Between Failures (MTBF) 1000 hr.	800 hr.	1100 hr.
Mean Time to Restore 2 hr.	1/4 hr.	1 hr.
Message Memory in Bits 100,000	250,000 bits	150,000 bits
Message Recall Time 1/2 sec.	1 sec.	1/4 sec.
Message Output Time 1 sec.	1/2 sec.	3/4 sec.
Other Quantitative Characteristics	Equal	Equal

Figure 2-25 Characteristics of Two Hypothetical ACDS Components

Characteristic	Weight	Perturbation of Characteristic	Probability of Perturbation	SYSTEM C			SYSTEM D			
				Raw Scalar	Weight x Probability	= Rating	Raw Scalar	Weight x Probability	= Rating	
1. MTBF	20	Heat loss in below 32°F	.25	B	x 20	x .25	= 40	C	x 20	= 25
				3	x 20	x .25		5	x 20	
				8	x 20	x .50		11	x 20	
				<u>135</u>			<u>160</u>			
2. MTTR	40	Trainees No Console Exp. No Perturb.	.3	D	x 40	x .3	240	E	x 40	60
				20	x 40	x .7		5	x 40	
				<u>1120</u>			<u>560</u>			
				<u>1360</u>			<u>620</u>			
3. DISPLAY QUALITY	40	Emerg. Battle Lighting No Perturb.	.3	F	x 40	x .3	84	G	x 40	108
				7	x 40	x .7		9	x 40	
				10	x 40	x .7		10	x 40	
				<u>280</u>			<u>280</u>			
				<u>364</u>			<u>368</u>			
4. SWITCH HUMAN ENGRG.	80	Emerg. Battle Lights No Perturb.	.3	H	x 80	x .3	168	I	x 80	96
				7	x 80	x .7		4	x 80	
				<u>504</u>			<u>392</u>			
				<u>672</u>			<u>488</u>			

TOTAL ADJUSTED SCORE

2531

1656

Figure 2-26 Comparison of Two Hypothetical ACDS Components

Examination of boxes B through J shows the ability of the modified quadrille technique to reflect numerically the performance of the two systems in periods of non-optimum operation.

The scalar values used in boxes F through J are extracted from Figure 2-23 and again, these figures are normalized around the requirements of the specification. Although a great deal of additional computation is required to use the modified quadrille technique, it has the capability of reflecting large numbers of subtle differences in system performance under non-optimum operating conditions. Figure 2-27 shows the total adjusted score for the two systems computed in one of the standard evaluation techniques.

Characteristic	Weight	System C		System D	
		Raw Scalar	Rating	Raw Scalar	Rating
1 MTBF	20	8	160	11	220
2 MTTR	40	40	1600	20	800
3 DISPLAY	40	10	400	10	400
4 SWITCHES	80	9	720	7	560
TOTAL ADJUSTED SCORE			<u>2880</u>		<u>2720</u>

Figure 2-27. A Non-Quadrille Evaluation of Consoles C & D

The difference in quantities of arithmetic and analytical thought are clearly apparent. But the standard techniques are unable to reflect the inability of System D to deal with non-optimum operating conditions.

ACDS evaluations will have to consider large numbers of subjective or irreducible characteristics and many types of non-optimum operation. The modified quadrille technique shown here permits these conditions to be incorporated in a numerical effectiveness rating with the minimum possible computational load. The use of this evaluation technique is recommended for ACDS system planners.

2.10.11 Discussion

The planning decisions made by Navy systems planners require accurate effectiveness data as well as accurate costs. There is no shortcut method for accomplishing this, just as there is no shortcut for writing an effective and complete operations order. Both require diligence and high professional capability. Some discussion seems appropriate.

- 1) ACDS effectiveness measurements should be made, as much as possible, by members of the ACDS project staff. They will develop the expertise to cope with the special estimating problems, and they alone will have adequate technical knowledge of the system.
- 2) Effectiveness measurement, particularly for ACDS, should underscore not underplay the important role of professional military judgment in determining military usefulness. Military usefulness is the product of system effectiveness and the value of performing the system's mission. The military value of a task can only be determined by experienced professional military men. The complex role of ACDS requires the exercise of professional judgment in effectiveness measurement.
- 3) The Navy must continue to emphasize the need for thorough documentation for all assumptions and base data used in effectiveness studies.
- 4) The concept of total force effectiveness measurement must be used in all applicable situations. This is particularly true of effectiveness measurements of large portions of ACDS. Its real effectiveness may only be measured by how the combat effectiveness of the total force changes.
- 5) All evaluations should consider the operation of the component, subsystem or system in non-optimal circumstances. This is a tedious procedure but it is necessary to demonstrate adequately the strengths and weaknesses of the design being considered.

Section 3 METHODOLOGY FOR SYSTEM PLANNERS

3.1 INTRODUCTION

Management aspects of methodology are presented in Section 2. This section is directed toward the naval system planner who is charged with developing technical approaches to system implementation. In other words, the subject involves the tools and techniques to develop overall system concepts, to select equipment, to develop the man/machine system, and to evaluate, install and test the system.

One of the most useful tools which the system planner has at his disposal is computerized simulation. It is often difficult to arrive at formulative or numerical techniques to be used for evaluating systems. Simulation can be used to test hypothetical systems or design approaches to systems without having to develop all the precise mathematical relationships. Also, simulation can often be used to test parts of the system which are not amenable at all to a formulative approach. For example, the human factors involved by the console operator can only be analyzed through simulation techniques.

Because of the importance of simulation in system design, considerable time and effort is devoted to analyzing the various uses of simulation in systems design and implementation. Sections 3.2 through 3.5 present the various aspects of simulation from the first considerations of modeling and development to the later phases of system checkout and testing. In Section 3.6 the important topic of simulation languages is presented. Simulation languages are techniques for efficiently designing and programming computerized simulation.

Despite the importance of simulation, the ANTACCS methodology study team believes that insufficient efforts have been expended in formulative techniques in system design and evaluation which are not normally considered to be of a simulation type. In these techniques, mathematical approaches are developed which are aimed at developing

numerical results intended to answer certain questions of design. The mathematical approaches involved may occasionally be handled with the aid of a computer. However, in these techniques the emphasis is on the mathematical expression rather than on the computer process. The computer process is aimed only at providing assistance in handling the expressions.

Mathematical modeling is discussed in Section 3.7. Sections 3.8 through 3.10 present three mathematical techniques for system design. In Section 3.8 a technique is presented in which the real time data handling system is regarded as a queue processor. That is, the various types of tasks arrive at the system at random times, and queues of tasks form. An analysis from this point of view can yield important results. It is important to note, however, that these techniques need more exploration and exploitation before extensive payoffs can be realized for system design.

In Sections 3.9 and 3.10, two examples of formulative techniques in system design are presented. In Section 3.9 a technique for developing quantitative measurements for analyzing information communication storage and retrieval is discussed. This approach is aimed at providing a better understanding of the processes of information transfer, file access, file design, and their software and hardware requirements. In Section 3.10 a figure of merit for digital computers is developed. This takes into account arithmetic speeds, word length, memory size, memory speed, and transfer speeds. It can be of use to Navy system planners who are selecting computers.

3.2 SIMULATION IN SYSTEM DESIGN

3.2.1 General

Simulation is a necessary tool for planning ACDS. The operational concept of ACDS is so large that computer simulation is essential to getting the job done on time in the design, evaluation, checkout and training stages of developing the command data system. There will be many different equipment and system interfaces for ACDS. Management information needs will require that nearly continuous simulation takes place to keep abreast of the evaluations of proposed system improvements and changes.

ACDS must also interface with other command data systems. Changes in these systems can radically affect the command data system. To be ready for such changes planners must be able to evaluate their possible effects. Simulation is the only effective tool that can be used to do this.

During the design phase of ACDS, planners will rely heavily on simulation to prevent fruitless investigation and insure maximum use of budgetary allocations. Since data systems in particular, and defense systems in general, continue to grow in complexity, scope, and cost, it becomes increasingly important for planners to be provided with tools that will let them test proposed configurations without building hardware. Simulation is the most powerful tool available to the planner for this purpose.

Before describing the simulation tools of particular importance to the ACDS planner, some background information is appropriate. This information is applicable to all simulation problems. This background is, however, slanted to the particular problems faced by the planner of the advanced command data system.

The obvious feature of all simulations is imitation or modeling. But a simulation is more than just a model; it has an operator and an objective. The operator adds dynamics to the model. For example, the operator of a ship-to-shore trajectory simulation would be a numerical integration method, a computer program, and a

computer. Common objectives of simulation are analysis, checkout, and training. Command system designers use simulation to analyze the complex operation of contemplated or existing systems. Large systems are checked out with simulated inputs. System operators are trained with man-machine simulations. System design, checkout, and training simulations are all important to planners of an effective command data system.

Once a planner has identified a need for simulation to help him solve a specific command control problem, then he has to decide whether or not it is practical and economic to use simulation. If he decides in favor of simulation he next must decide what type of simulation and how it must be implemented.

There are two problems to be faced in deciding if simulation should be used. First the planner must find out if the specific command problem area can be simulated accurately enough for the simulation results to be valid. Next he must determine the trade-offs between simulating part of the system and using part of a real system. For example, it may be more expensive to simulate a piece of transmission hardware of a tactical data system than it would be to buy the piece of equipment and try it, especially if the equipment is an off-the-shelf item.

If simulation seems appropriate, the next step is to develop a model of the specific command data problem to be solved. Modeling is an art which requires the talent of a specialist. Yet the planner must understand a great deal of this art to plan effectively and wisely. A section of this volume is devoted to modeling.

One point should be emphasized about design of simulations. A simulation should be easy to use. Parameters in the simulation must be easy to vary. Also the simulation should record its results so that they can be readily interpreted.

These, then, are the fundamentals of simulation. Now we will discuss simulation for system design, development, checkout, test and evaluation with particular reference to use in simulation of ACDS.

A system designer does not simulate and model to create system designs but to test system designs. A system designer can test and examine early forms of his design with simple diagrams and hand calculations. His intuition and experience tell him that one equipment configuration is more functional than another. However, as the design becomes more advanced, he finds it increasingly difficult to evaluate design trade-offs. Finally, the design is too complex. He can no longer see the dynamics and interrelationships of the heavy components.

How can he be sure his design will perform as he expects when subjected to stresses in the real world? One method is to build a prototype system and subject it to a simulated real-world environment. Reasons why this is often an unrealistic approach, especially for military command and control systems are:

- 1) Simulated environments, such as military maneuvers, are expensive in time, manpower, and money.
- 2) It is difficult to reproduce real-world environments for repetitive tests of system prototypes.
- 3) System prototypes are expensive and may require years of development.
- 4) Often, scarce resources such as shipyards, cannot be used.

Computer simulation is a fast and inexpensive alternative method. Simulation is limited by the ability of the simulation designer to create an accurate model of the system components and their interaction. System components may be computer programs, people, information channels, sensors, and weapon systems. Each component and its dynamic relationship with others must be represented accurately for valid system simulation. However, the actions of people are relatively unpredictable, especially when involving evaluation and decisions.

Two general classifications of system simulations are man-machine and all-computer simulation. Application of these two types of simulation to the design of command and control systems are discussed next.

3.2.2 Man-Machine Simulation (General)

Operations simulation simulates operation of a command data system at the interface between command personnel and displays. Figure 3-1 shows a command data environment and the simulated man-machine interface.

An operations simulation presents simulated information to command personnel and modifies the information to suit their response. An operations simulation consists of command personnel, communication equipment, and information exchange. How information exchange between command people and communication equipment is controlled depends on information rate and quantity. If small amounts of information are communicated, the information exchange might be done manually with switches or grease pencil displays. Since information rates and quantities are high for command and control systems, operations simulations generally use computers to control information exchange. A computer also simulates other components of the command and control environment, such as sensing and controlling devices, and external world activities.

Objectives of system designers are to increase the effectiveness and functionality of system design and reduce time and cost of implementation. The operations simulation tool can be used by system designers to achieve these objectives. However, these objectives are too general to be used in planning specific simulation runs. Each simulation run or series of runs must produce data to form specific conclusions about system design.

System design is arrived at by using past experience, imagination, projection, and intuition. Many system parameters are difficult to evaluate: type of information displayed, frequency of information updating, number of operators required, performance of the operators under peak loading, reaction time of the operators, types of operator errors, consequences of operator errors, unnecessary control options, and necessary automatic modes of operation. These parameters affect system design; quantity of communication links, size of computer memory, speed of the selected computer, computer software, and so on.

A model of the system may contain hundreds of parameters to be examined. Sometimes only one parameter need be examined with a series of simulation runs. For instance, the effect of aggregate or lumped radar returns during peak loading on the commander's actions could be tested with a series of simulation runs.

Sometimes a system design contains latent parameters as requirements which are illuminated during operations simulation. Simulation stimulates ideas to improve system design.

Operations simulation provides feedback to system designers for improving system design. This reduces time and cost of implementation by reducing modifications to the production model of the system.

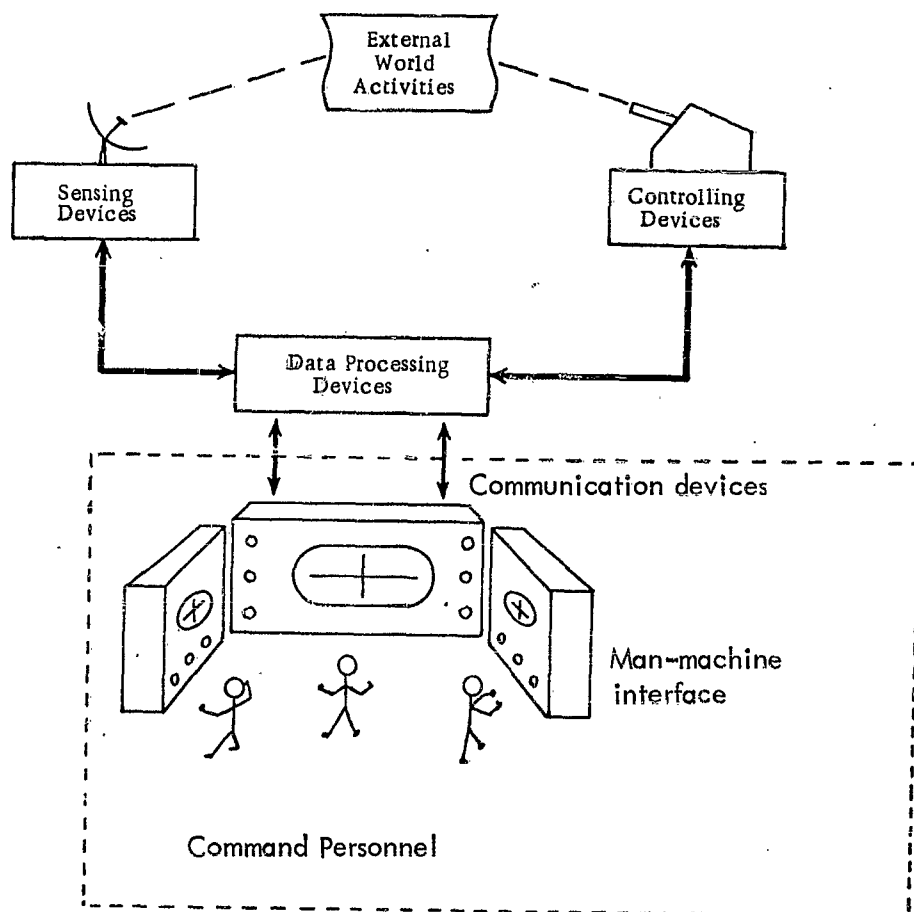


Figure 3-1. Command Data System

Obviously, an operations simulation should only be implemented to meet a definite need. Each simulation run should be planned to yield results leading to specific conclusions which will satisfy that need.

The most common pitfall in simulation is failure to anticipate how simulation results are used. Simulations can produce much data. Data selection summaries, and analyses of significance must be preplanned. Good simulations have been known to fail for lack of this planning.

3.2.3 Man-Machine Simulation (The Laboratory)

A simulation laboratory houses personnel and equipment such as computer hardware, computer software and communication devices.

The laboratory consists of rooms for the hardware supporting the simulation and the type of environment which must be simulated. If a decentralized command and control system must be simulated, a room or compartment may be required for each group of command personnel.

In addition to data collected by the computing facility during simulations, much data is gathered by observation. Each study has an observation area for simulation controllers to study the simulation participants.

Simulation laboratories should be built adjacent to existing computing facilities to take advantage of their data-processing support. Efficient use of computer time can reduce equipment costs, which are high in man-machine simulation.

A fringe benefit of operations simulation is system checkout capability. If the laboratory is large, system hardware can be incorporated into the simulation as it is developed. The system computer can be used in the simulation when ready, and the general computing facility then furnishes system inputs only.

A Simulation Facility (SIMFAC) in Paramus, New Jersey, is a physical model of the SAC Underground Command Post complete with Command/Control personnel stations and with capabilities to produce simulated SACCS hardware printouts and

1
wall displays. From a soundproof observation deck SIMFAC personnel perform actions to simulate all external occurrences, from an intelligence buildup to changes in threat responses. Many of the operational design concepts for command and control have been derived and validated.*

Large general purpose digital computers are generally used to control operations simulations because:

- 1) They use software to develop and modify complex simulation situations programs.
- 2) They have speed and capacity for processing simulation tasks, on-line and
- 3) Many organizations already use this type of computer for data processing.

2
An example is at the Systems Simulation Research Laboratory at SDC where a Philco 2010 controls several man-machine simulations. This computer is normally used for general data processing with an occasional simulation. The computer can also operate in a pseudo multi-programming mode in which a data processing program can be interrupted and saved for later completion while a simulation program is executed.

A digital computer can be used to simulate many complex subsystems of control systems. Simulation avoids developing subsystems equipment until its value has been determined.

In the last five years, computer speeds have increased to suit on-line operations simulation. The most effective type of computer is a large scale general purpose digital machine with interrupt features, real-time clock, and standard display interface equipment.

Multi-programming techniques reduce cost of operations simulation by using computer time more efficiently. Cost also is reduced by using an existing computing facility.

* Anon, Simulation, BRT-12, System Development Corporation.

There are no software packages tailored to implementing on-line man-machine simulation programs. Compiler languages can be used for on-line programming, but on-line programs are seldom coded in a compiler language because the generated code is not efficient and on-line compiler languages are unavailable for most computers. If computer time is not an issue, some efficiency could be sacrificed for the advantages of a compiler language.

Computer software to support a large simulation effort is expensive. It will be modified more than any other part of the total facilities. An on-line simulation programming system should be set up before attempting simulations. The programming system aids in planning and coordinating simulation development. It also makes program modifications and documentation easier by setting up standard procedures. Also, new personnel quickly become familiar with a well-organized and documented system.

Once a flexible programming framework is set up, the development can begin of the many programming segments of the simulation. Some program segments contain actual system software logic; other segments are simulations of real-world elements.

System software logic undergoes an evolution as system requirements solidify through operations simulation. Outgrowth of the evolution is a set of program specifications to fit the needs of the final system with least modification. Computer program specifications are written before the hardware is selected. These specifications are helpful in selecting computer and auxiliary memory units.

Simulated environment software provides substitutes for real-world elements which are absent in the simulation. Software is the implementation of mathematical models to represent radar inputs, weapon effectiveness, threat dynamics, system errors, and the like. Speed of the computer must be enough to process system software logic and real-world models on-line. Time is limiting when designing mathematical models. For example, it may be necessary to compute probable radar detection using a stochastic process rather than to model radar search pattern and testing to find out when the radar beam intercepts target.

Communication devices consist of displays and consoles for communication between a computer and command personnel. Equipment depends on type of information to be communicated. Cathode ray tubes, TV tubes, slide projectors, keyboards, buttons and switches are commonly used. Manual displays, such as weather status and equipment status boards, are economical and often used in early stages of operations simulation.

General purpose display equipment is useful for operations simulation during the design phase of system development. This allows equipment to be reconfigured to test operating modes and display configurations.

After firm communication requirements have been determined, more elaborate consoles and displays may be constructed to refine system design.

Additional equipment may be needed for greater realism; e.g., models or photographs of terrain scanned by closed-circuit TV cameras to display military developments.

The computer for simulation must process system software and simulation models. If system software is elaborate, the computer may not be able to compute both on-line. Another computer may be needed to process the simulation models. This second computer could supply all inputs to and receive all outputs from the computer for system software logic.

Hybrid simulation techniques can also be used to relieve the digital computer of equation solving. An analog computer might simulate an entire air/sea battle involving many interceptors and threats.

Operations simulations use a "gaming" approach. A threat model is designed to present a situation to command people through communication devices. The simulation responds to actions of the commander by displaying consequences of his actions. Goal of commander is to "defeat" the threat model. This technique is extended to add competition to the simulation by using two teams; a "red" team familiar with the system and simulation, and a "blue" team of system designers and operators.

The red team designs threat models and tactics that best challenge the system. The simulation provides the red team with as much flexibility as possible; e.g., allows them to write software models. These models include elaborate tactics which are changed dynamically by the response of the system.

This technique needs referees to monitor the red team's threat design so that it does not violate any rules. The referees also test the simulation to check correct operation before the simulation run.

After the simulation checks out, the blue team is briefed and operate the consoles. They do their best against the threat model. Object is to test capability of the system and locate weak points.

3.2.4 All-Computer Simulation

This Section uses a typical tactical data system design to show the use of all-computer simulation as a design tool.

The system's mission is to defend a fleet unit against attack from approximately 25 attack vessels with an 80% probability of destroying 25 of these vessels and a 95% probability of destroying 20 of them.

The system design consists of five killer units equally spaced around the defended unit. Each killer unit has a computer, command personnel, weapons, detection equipment and tracking equipment. All five are connected through a master control center which monitors the entire system. Figure 3-2 illustrates the system deployment and communication links.

The task of the system planner is to determine a set of parameters which lets the system fulfill its mission. Also he must consider the system cost involved with each selection of parameters such as:

- 1) The detection ranges of the weapons are uniformly distributed between P_1 and P_2 yards.

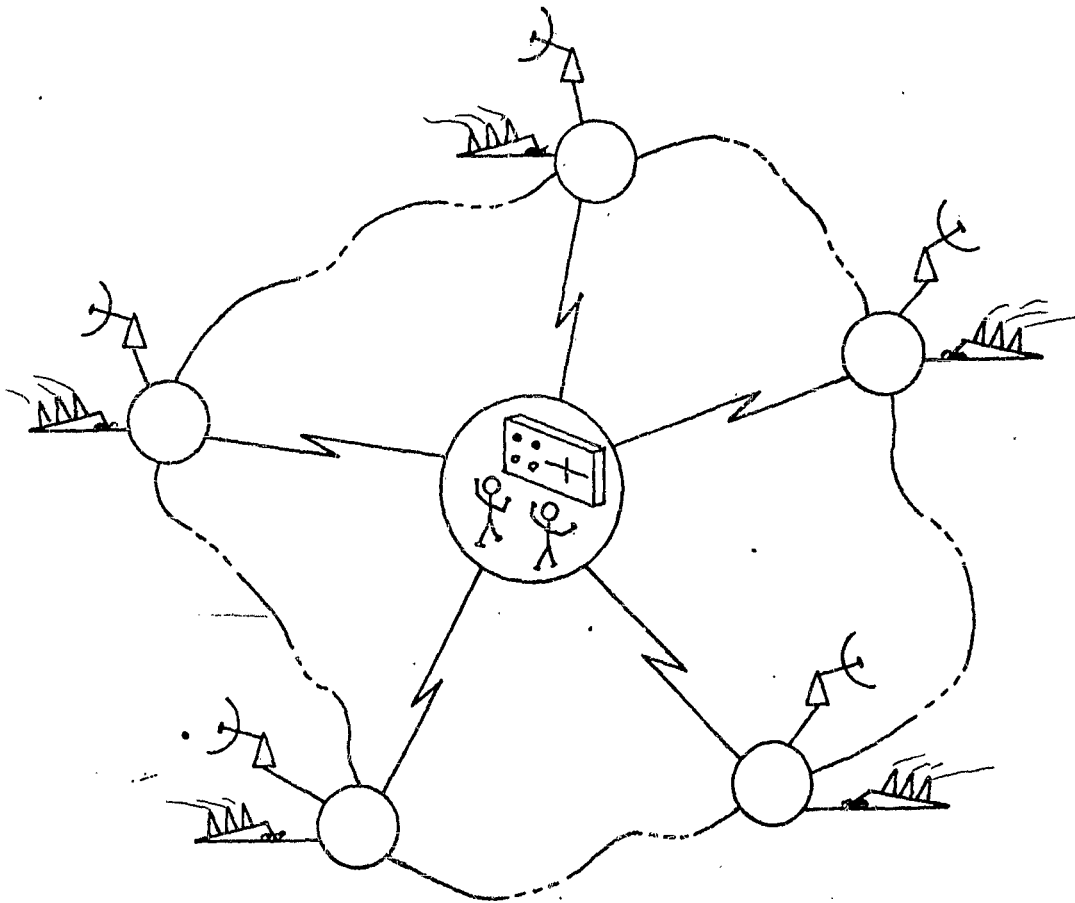


Figure 3-2. Hypothetical System Design

- 2) Each killer unit has P_3 interceptors which can be launched at a maximum rate of one interceptor every P_4 seconds.
- 3) The probability, P_5 , of one interceptor destroying one attacker is a known function of the position and velocity of the attacker at the launch time of the interceptor.

- 4) The time required for the master control center to assign an attacker to another complex is normally distributed about P_7 seconds with a known variance; the assignments are processed in order, one at a time.
- 5) A peripheral killer unit is destroyed if an attacker reaches within P_8 miles of the killer unit.

Decision rules must also be established which govern the use of the system, namely:

- 1) How many interceptors are launched at each attacker?
- 2) How are weapon assignment conflicts resolved?
- 3) How much control is exercised by the master control center?

Can the system designers determine a set of parameters and operating procedures which they feel will give best performance and economy, using their intuition and experience? If they determine a set of parameters, how can they demonstrate the performance of their design for final approval?

Although simulation is not a panacea, it is used to answer the above types of questions. Some evidence of this use is the number of simulation languages now used to study "machine-shop" problems. Block diagrams or flow diagrams describing this class of problems are very similar to the block diagram used to model the tactical data system.

Figure 3-3 is a simple block diagram of the attacker-interceptor model. Even with this simple problem, it is difficult to determine the capability of the model by intuition alone.

A computer program must be written to exercise the block diagram model of the system design. If a simulation language is not used, a tailored computer program must be written.

Many simulation runs are needed to get results from a stochastic system model. The approach to the attacker-interceptor model would be a Monte-Carlo technique:

- 1) Design many attack configurations to span all expected threats.
- 2) Select a desirable set of model parameters and decision rules.
- 3) Run the simulation many times for each attack configuration.

The effectiveness of the model against an attack configuration is proved if 25 attackers are destroyed in 80% of the runs and 20 attackers are destroyed in 95% of the runs. If the effectiveness of the model is inadequate, another set of parameters is tested to improve the performance. The model parameters are varied to study the sensitivity of the model's capability to critical parameters, for example, the number of interceptors launched at each attacker.

A flexible general purpose computer program may be designed to provide system performance data, or a proposed new design or modification to an existing design before equipment is selected and committed, or before any computer program is designed. Total system design, including software and equipment, receives rigorous analysis and evaluation early in design so that key decisions can be made on:

- 1) Kind of equipment to be used.
- 2) Number of each type of equipment.
- 3) Kind of data processing discipline and strategy required.
- 4) Projected performance of the system under varying loads.
- 5) System's maximum capacity.
- 6) System's ability to respond as a function of loading capacity, and environment.

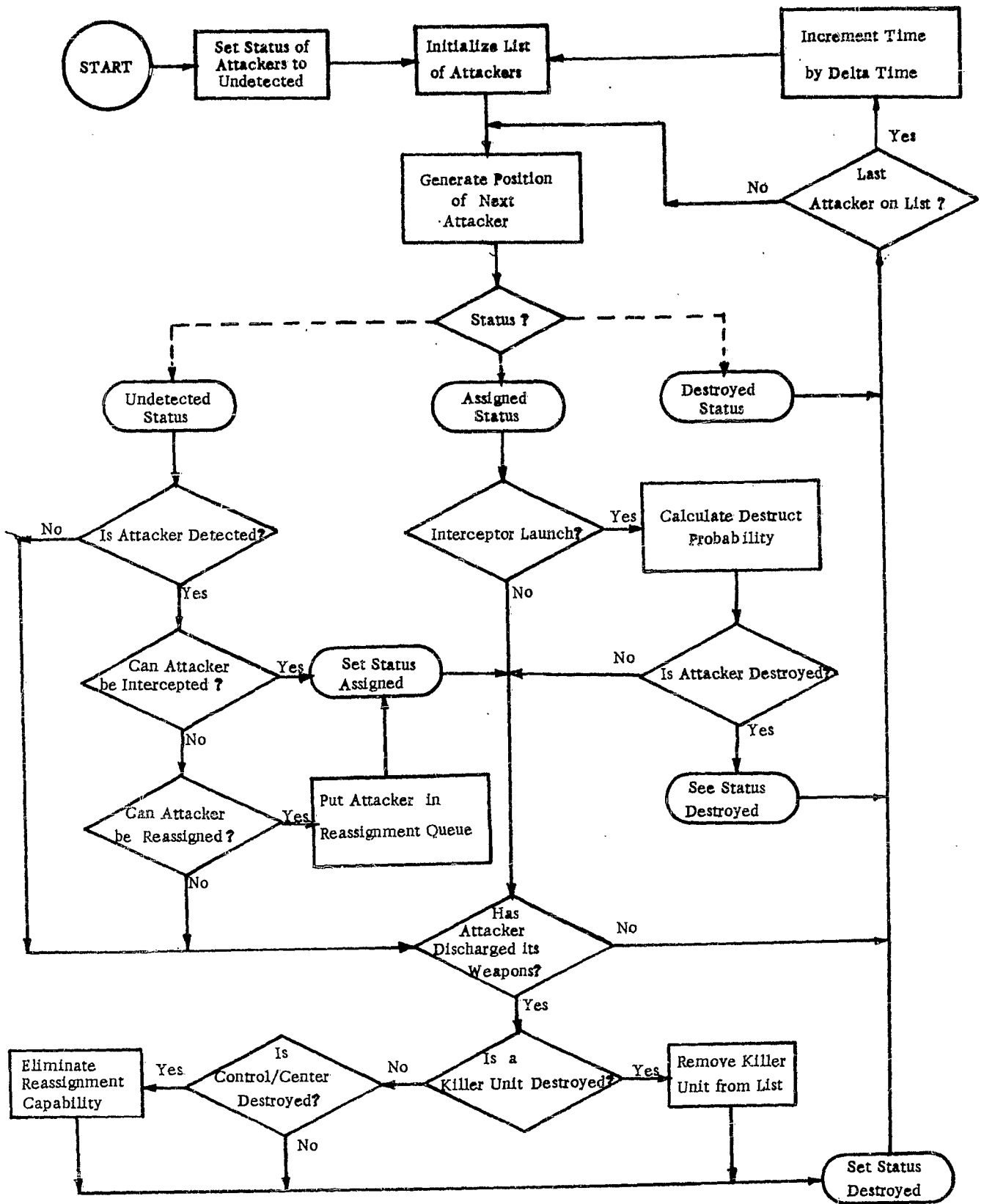


Figure 3-3. Interceptor Model

Validity of the simulation results depends on accuracy of the system model. If one component of the system is modeled inaccurately, and the system is sensitive to that component, results are misleading.

If acceptable models can be developed for human elements, all-computer simulations can save time and money. Models of human elements can be developed using man-machine simulations. This is by recording the reaction of many elements in response to a specific display configuration. The recorded data are used to establish operator decisions, errors, etc. Unfortunately, models developed this way can only be used to simulate one attack configuration.

Even with erroneous components in the system, simulation results can help the planner determine relative importance of system components. If operators in the system made errors frequently which affect the total performance of the system by 25 percent, the simulation might still be used to evaluate the trade-offs between "hardened" killer units and more accurate interceptors.

In summary, operations simulation and all-computer simulation can both be used to answer system design questions. Both types have advantages and disadvantages.

Operations simulation is valuable for determining operational requirements of a command and control system. This is by entering operating personnel into the simulation so they can uncover functional difficulties of the system design before the system is produced and used in the field.

The need for operational control systems is expanding in the military, and the need for improved command systems has been ever present. Simulation techniques prove helpful, pooling and integrating knowledge from many sources and providing the opportunity to integrate and vary the elements of such systems. Most published simulation experiences have involved all-machine models, while man-machine simulation is valuable when problems involve organizational interactions, design of information systems, and conflict or interacting decision rules, since these are developed considerable during simulation.

The main disadvantages of operations simulation are:

- 1) Elaborate hardware equipment is needed, i.e., displays, special interface equipment, larger facilities, etc.
- 2) Longer time to implement.
- 3) Longer time to run, i.e., normally running in real-time requires briefing and debriefing of operating personnel.
- 4) Trained experienced operational crew needed.

The advantages and disadvantages of operations simulation are reversed for all-computer simulation. All-computer simulation requires no elaborate equipment or facilities other than the computer. It is relatively fast to implement, especially when written in a simulation language, and it can run faster than real-time if the system is not too large. However, all-computer simulation generally cannot be used to uncover any functional difficulties of operating personnel.

All-computer simulation is normally used with Monte-Carlo techniques. These require many simulation runs or evaluation of many design alternatives. An all-computer simulation written in SIMSCRIPT used in a logistics study evaluated many scheduling procedures*. The optimum and next to optimum procedures were then evaluated in an operations simulation of the same system.

In this way, operations simulation and all-computer simulation used together can solve system design questions rapidly and economically.

It is recommended that operations simulation and all-computer simulation be employed as early as possible in ACDS planning. Navy operating personnel should be used in an operations simulation to evaluate operating procedures and total system concept.

Operations simulation deals with hardware, command decisions, human interaction, operating procedures, and situational change; all important factors operating in and

* Cohen, The Design and Objectives of Laboratory Problem IV, RM-3354-PR, RAND Corporation, January 1963.

I
about a system. Inputs are identified, performance is observed and measured, and outputs are recorded. This significant extension of simulation technology provides powerful means to assist the design, development, evaluation, and improvement of naval systems*.

(

*** Anon. Simulation, BRT-12, System Development Corporation**

3.3 SIMULATION IN SYSTEM DEVELOPMENT

3.3.1 General

A command data system consists of subsystems: Communications, detection, weapons, transportation, data processing, and programming.

The task of total system design involves integrating these subsystems into the best geometrical and functional configuration. Part of this task is to determine general characteristics of each subsystem. For a detection subsystem these might be range, track capacity, accuracy, watts, size, and weight. After general characteristics have been determined for each subsystem, development of the total system can begin, i.e., design and development of the subsystems.

Problems involved in designing and developing subsystems are generally the same as those involved in designing and developing total systems, namely, integrating a large number of components into an optimum configuration.

3.3.2 Analysis

Simulation plays a major role in the analysis phase of system design. One of the first tasks is to evaluate many alternative designs. Often simulation is used because analytical methods are too difficult. For example, the design may contain many non-linear relationships which would devalue a linear programming solution.

An important application of simulation is in dynamics. The mathematical models of moving vehicles are often too complex for analytical solution. Sometimes the models contain empirical tables, such as atmospheric density functions, which must be represented by series approximations. An accurate solution can only be through numerical integration. Although numerical integration solutions do not resemble man-machine simulation techniques, they are popularly referred to as simulations.

3.3.3 Optimization

System designs which contain only a few parameters can be optimized by evaluating all possible designs. Suppose a system design contains only two parameters and each parameter can assume ten values. All possible designs of this type can be evaluated with one hundred simulation runs.

Few system designs depend for their performance on only two parameters. Optimization of multi-parameter systems is much more difficult. A gradient method is generally applied.

This method begins with estimating the set of parameters to optimize the design. This set is adjusted by small steps until it is no longer possible to optimize the design by further adjustment of the parameters' values.

The inherent power of this method is the technique by which the parameter adjustment is controlled. The amount by which each parameter is adjusted varies with every step, but the vector sum of all the adjustments is constant at each step. The amount by which a parameter is adjusted is proportional to the sensitivity of the optimization function to that parameter. Parameters which affect the optimization function most are modified by a greater amount.

Figure 3-4 graphically represents the method.

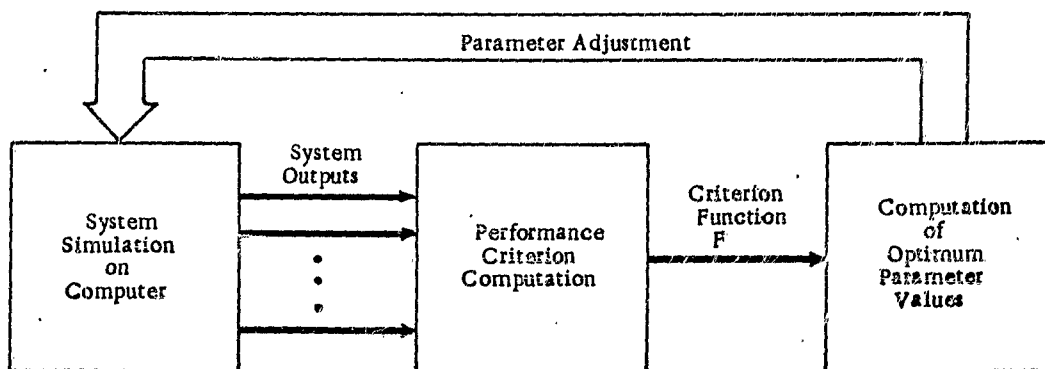


Figure 3-4. Block Diagram of Design Optimization Problem

The gradient method is often implemented on hybrid computers. The analog part of the computer simulates the system design. The digital part evaluates the performance of the simulated system and controls the adjustment of parameters.

3.3.4 Subsystems Development Simulation

The examples presented show where simulation has been used in development of various subsystems of command and control systems. They are intended to indicate simulation applications and not to give a comprehensive treatment.

Communication systems can be complex, especially in a decentralized command and control system. Figure 3-5 shows a complex system which could be analyzed by simulation techniques.

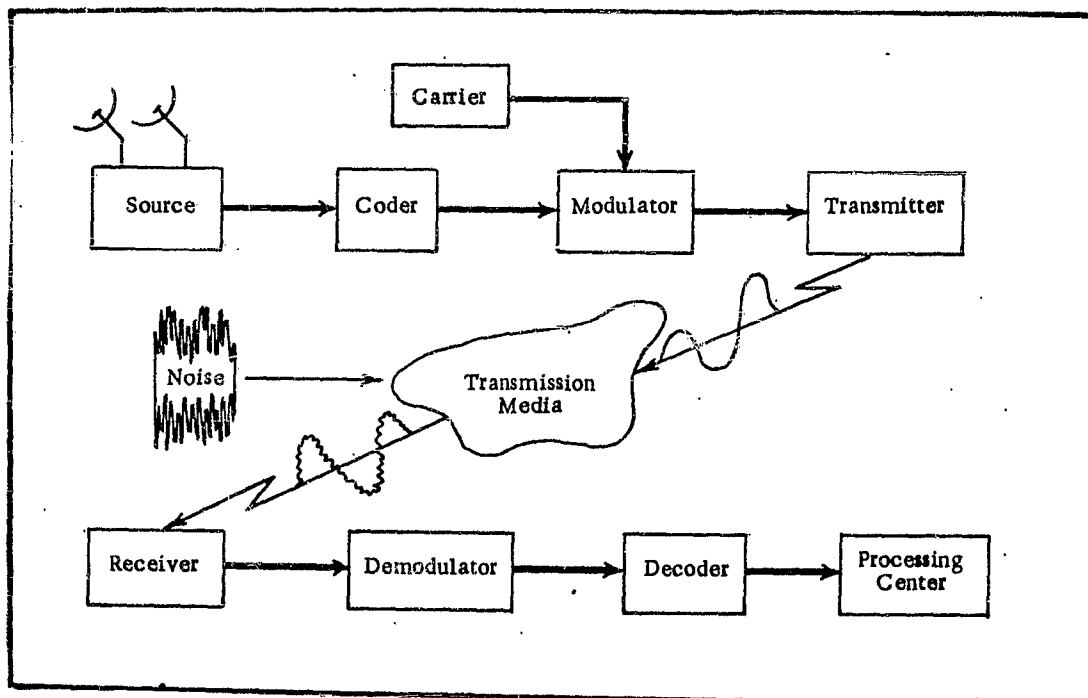


Figure 3-5. Carrier Transmission System

Analog simulation has been used at General Electric to analyze a secure communications system design. The simulation evaluated system feasibility, determined optimum system parameters, and evaluated system performance in various signal environments. The simulation results avoided time and expense of hardware equipment.

Radar, sonar and infrared detection systems have been studied with simulation techniques. Simulation can be used to study the system performance as a function of the system errors, to optimize and improve system parameters, and to analyze measurement accuracy and track ability.

Analog simulation has also been used at General Electric to improve radar system design concepts. Potential areas of difficulty were illuminated by simulation early in the design phase.

An article published in Russia describes how digital simulation is used as a research tool for studying electromagnetic fields around disturbing objects, i.e., plates, discs, cylinders and spheres.

The complete assortment of simulation tools can be used in the design and development of weapon systems.

Digital computer simulation can be used for solutions which require great accuracy. Unfortunately, accurate digital simulations require much computer time. Often calculations must be performed in double precision arithmetic.

On the other hand, analog simulations require very little computer time but are not accurate. Consequently, analog simulation is used when many solutions are required. For example, analog simulation can be used for analysis of guidance techniques or the calculation of kill probabilities of ship-to-air missiles.

Man-machine simulation can be used to study the performance of human components in weapon systems. TV missile systems have been analyzed with man-machine simulation to determine the ability of pilots to guide missiles. An analog computer simulates the motion of the missile in response to the pilot's commands.

Sometimes actual system hardware is studied by simulating the environment of the hardware component. Infrared seeker components have been studied by supplying a moving target to the seeker through an arrangement of lenses and mirrors. The motion of the missile and the target are controlled with an analog computer.

Analog and digital simulations are used to study damped spring mass systems (suspension systems) of transport vehicles. Analog simulation is used more extensively because it is better suited to the solution of differential equations. These simulations are valuable for determining the shock and stresses on delicate components which must be transported: computers, communications equipment, guidance equipment, etc.

General Motors has written a simulation language, DYANA, which is used to simulate complex damped spring mass systems. The input to DYANA is a description of the physical system, i.e., geometry, spring constants, damping coefficients, forcing functions, etc. DYANA translates the input into a set of differential equations which represent the system. A FORTRAN program is punched by DYANA to solve the equations and print out the responses of system components.

Simulation is used at the micro and macro level of the development of data processing systems. One of the largest applications of simulation at the micro level is to check out logical circuit designs. Computer logic can be represented with boolean expressions. This type of simulation operates at the bit-time level for the check out of logical circuits.

Application of simulation at micro level is not limited to computer circuits. Other computer components can be simulated, e.g., drum memory, word structure, and information channels. Simulation has also been used to study error patterns in computer information channels.

A programming system to control on-line processing in a command control system is generally under continual modification. Modifications result from improvements or expansion of the system. They can be checked out by simulation. This is done by simulating input data to the system to test the program's functions.

3.4 SIMULATION IN SYSTEM CHECKOUT

3.4.1 General

Simulation has been applied to many seemingly unrelated activities: numerical integration of equations of motion, command and war games pilot trainers. The term "simulation" is often used whenever an activity is represented by something else. Simulation is also applied to the activity of system checkout. Operation of a system is often initially checked with test inputs which are not received from the normal or "real" environment, in a "simulation mode."

Electronic circuits are checked with signal generators and oscilloscopes. A signal generator supplies an input signal to the circuit, while an oscilloscope displays how the circuit transforms the signal. The signal generator might be termed a signal simulator.

A similar approach is used to check out command and control systems of which the following are three examples relevant to ANTACCS.

3.4.2 Range Safety System

The Pacific Missile Range range safety system is a complex of radars, communication links, computers, and command and control devices. It provides range safety support during missile and space vehicle launches.

Data are processed on-line by an IBM 7090 and displayed in a Range Safety Control Center. Displayed information is used to evaluate performance. If a missile violates any predetermined limits, it may be destroyed.

A set of computer program parameters is prepared for each launch, including the characteristics of the missile, local weather data, program control, etc. A simulation is run to test the parameters and the equipment in the Range Safety Control Center.

The simulation is controlled by computer. When the computer program is in simulation mode, it reads simulated radar data from magnetic tape instead of reading data from

magnetic tape instead of reading data from the radar input buffer. The simulated data is raw radar data recorded from a previous launch. It is processed by the computer in normal fashion. The program output exercises most of the equipment in the Range Safety Control Center. This equipment includes digital-to-analog converters, plug-board switches, plotting boards, and control consoles.

The simulation checks the operation of the program and terminal hardware but not the radars or communication links.

The Range Safety System built up and modified over several years, is a patchwork of many smaller systems. Checkout of all these is laborious and performed for each launch, coordinated by voice communication.

If the computer could monitor or control this routine checkout operation, the operation of the Range Safety System would be more efficient. At present only a few launches can be made each day because of the long preparations.

Checkout of the Range Safety Center is relatively simple because it is under computer control. Routing procedures such as system checkout should be controlled by computers whenever possible.

3.4.3 The Real-Time Data Handling System

The Real-Time Data Handling System (RTDHS) at Point Mugu, California, is a multi-computer system consisting of a prime computer and peripheral computers. The prime computer processes data, presents displays and performs control functions. A typical control function is transmission of aircraft vectoring commands. The peripheral computers receive and process radar data at each radar site and transmit the processed data to the prime computer.

The simulation checkout of RTDHS is similar to that of the Range Safety System. Simulated radar data is read from magnetic tape and used to check out the computer program and associated equipment. However, the RTDHS simulation can be more comprehensive than the Range Safety System simulation. This can be done by transmitting simulated radar data to peripheral computers for processing. After processing

by peripheral computers, data can be transmitted to the primary computer. In this way, hardware and programs at each radar site and the transmission system can be checked out in addition to the operation of the primary site.

Multi-computer systems, like RTDHS, are readily adaptable to simulation checkout because of the flexibility of program control at many places in the system. RTDHS simulation modes can be expanded simply by modifying the computer programs. For example, each peripheral computer could read simulated radar data from tape and transmit the data to the primary computer. The radars could also be included in the simulation because they can be controlled by the computer program through digital-to-synchro converters.

3.4.4 MTDS

Simulation is used similarly in MTDS. The configuration resembles RTDHS. It consists of a central or primary computer - a Q-20 - which receives data from satellite computers. The central computer supports the Tactical Air Command Center (TACC) which monitors the entire "battle." It controls various displays in the TACC. A satellite computer supports operations at a Tactical Air Operation Center (TAOC). Each TAOC identifies, classifies, and assigns weapons to airborne targets and transmits their actions to the TACC.

The MTDS simulation checks almost the entire system. Targets are generated by the Q-20 and transmitted to TAOC's where they are processed. The TAOC's transmit their results to the TACC for display and command/control action.

MTDS also makes use of other smaller simulations for checking system components. The operation of the TAOC's can be checked out individually without involving other parts of MTDS. This is done by supplying simulated targets to the TAOC with a target simulator, the SPS-T2A.

A test director of MTDS said; "Simulations should be designed so they may be set up and operated completely by military operations personnel. Contractor prepared film which was used for the MTDS simulation runs and the time required for the film preparation were too long."

Simulation checkout in MTDS is extensive but provides a valuable troubleshooting and maintenance aid. Component simulations prior to total system simulation avoid using the entire system to find a malfunction in one component.

3.4.5 Conclusion

Simulation is not only very effective in checking out Command Data Systems, it is also now well-known and practiced in the Navy and Marine Corps. In a multi-computer system, as ACDS is likely to be, comprehensive simulated checkout may be performed since there are several general purpose computers available to command various system equipments and to exercise each other. As far as it is practical, system components should include capability to be exercised and checked by the central system control.

3.5 SIMULATION IN TEST AND EVALUATION

3.5.1 Introduction

An example of the role of simulation in testing and evaluating new, complex systems is the simulation facility operated by the Navy at the Naval Missile Center, Point Mugu, California. Experience gained can be extremely valuable to factual data system planners. Not only is the experience of value in examining tactical and attack parameters, such as range at time of firing, but also similar, less sophisticated facilities, can be adapted by the system designer to evaluate alternative design concepts during early phases of system design.

3.5.2 The Simulation Laboratory

The new simulation and vectoring laboratory for the Naval Missile Center contains analog computers and other special purpose electronic equipment, and studies many problems emphasizing simulation testing of Navy weapons systems.

The most general role and function of the laboratory is to use simulation studies for all Navy problem areas which can be effectively studied by simulation.

The facility is used by NMC to simulate all parts of weapons systems by electronic analog computers and to vector a missile-carrying aircraft into correct positions for launching missiles against airborne targets.

The analog computers are of several kinds, the REAC (Reeves Electronic Analog Computer), the Bendix three-dimensional flight simulator, and the PACE computer built by EAI.

A prominent simulation project being carried on is a study of the problems involved in attacking an enemy airplane when the pilot of the missile-carrying interceptor never actually sees his target. The studies are concerned with two basic problems:

- 1) How does a ground or shipboard controller, using a long range search radar, vector the interceptor airplane into a position where its own airborne radar can "see" the target?

- 2) How can the airplane be flown close enough to the target to successfully launch a missile?

The pilot must depend entirely on information obtained from his radar system. Hence, with this "vectoring" problem to study, the most important part of the F4B cockpit simulator is the radar display. Every effort has been made to have the pilot and his radar observer see the same displays that would appear in a combat situation.

Closely associated with the intercept evaluation is the test and evaluation program for the Airborne Tactical Data System (ATDS). This is a computer-automated fleet-oriented system with similar objectives.

The cockpit simulator requires three large analog computers to realistically represent:

- 1) The response of the airplane to the flight controls.
- 2) The geometry (or geography) of the problem, sometimes extending over several hundred miles.
- 3) Simulation of the electronic equipment aboard the airplane which transforms the raw radar information to meaningful displays.

The ATDS is typical of a complete weapon system which must be located in a laboratory where it can be studied in a simulated environment. This system consists of a high-powered search radar and a number of digital computers which automatically interpret what the radar sees, display the information, and direct a number of fighter aircraft to intercept enemy aircraft.

A set of operational ATDS radar-computer-display equipments, as in the airplane, is installed and operating at the Naval Missile Center in laboratory spaces near the analog computers.

By locating the laboratory ATDS near the analog computers, many tests of the automatic detection tracking and reporting functions of the ATDS computers can be performed without actually having airplanes in the air.

3.5.3 A Cockpit Simulator

An intercept simulator was constructed at the Naval Missile Center to aid evaluation of the F-4B/SPARROW III and Airborne Tactical Data System weapons systems. The simulator combines an analog computer with a mockup of an F-4B cockpit and accessory equipment to simulate the flight of an F-4B fighter from combat air patrol to breakaway maneuver.

The intercept flight is simulated by solving, on an analog computer, mathematical equations representing the fighter-target intercept dynamics, and by duplicating with operating hardware the airborne-intercept-radar controls and displays for both the clear and countermeasures environment.

The cockpit provides simulation only through the navigational instrumentation. No attempt is made to provide such effects as landscape, thermal, or gravitational effects.

By combining an analog computer with a mockup of an F-4B cockpit and accessory equipment, the intercept simulator duplicates many flight conditions in Naval airborne intercept tactics. Such tactics, as used in current fleet defense strategy, deploy early warning aircraft around a fleet perimeter, with F-4B interceptors on combat air patrol 100 to 150 nautical miles distant. Early warning radars contact and track approaching aircraft; information is processed by a Combat Information Center and, if the aircraft is hostile, air controller dispatches one or more of the patrolling F-4B's to intercept. Radio communications from the center to the F-4B pilot "vector" him until he can detect the target with his airborne intercept radar. After detection, the target is automatically tracked by the radar until the pilot launches his missiles and breaks away.

Future fleet defense operations are similar, but involve the Airborne Tactical Data System (ATDS) and the Naval Tactical Data System (NTDS). In these systems information is processed automatically by digital computers, and once the interceptor pilot is assigned to a mission, vectoring information is automatically transmitted, received, and presented to him electronically.

The F-4B intercept simulator consists of; electronic analog computer, pilot's cockpit, radar intercept officer's cockpit, CIC station, and AN/ASW-13 digital data communications set.

With these interconnected units, the flight of a F-4B fighter, or any part of it can be simulated. Although no motion of the cockpit takes place, the pilot and radar operator "fly" the F-4B within its design limits, receive vectoring commands from the command center, operate the radar in finding and tracking a target, and respond to radar scope and instrument displays duplicating those in actual aircraft. Countermeasure effects, such as voice jamming, chaff, decoys and range jamming, can be simulated. The intercept simulator allows technical areas of interest, such as vectoring accuracy of the effects of engineering changes, to be investigated; the results are combined with other ground tests and with flight tests in weapons-system evaluation.

Simulation of the intercept problem is by solving, on the electronic analog computer, mathematical equations representing the fighter-target intercept dynamics, and by duplicating with operating hardware the cockpit of the F-4B airplane, the command station, and countermeasure effects. The computer and hardware are cabled together and function as a single unit to simulate a typical intercept situation or problem.

Each of the units in the analog computer performs one or more mathematical operations on the voltages (and, therefore, on the mathematical quantities) fed into it. By interconnecting the units to perform all the operations called for by any set of equations, an electronic scale model of the mathematical equations is produced, and the computer can give a solution. The equations are typically those of engineering or physical systems, in which mathematical operations produce changes with time in the variables such as position in space, velocity, and heading. In the analog computer, the voltages vary continuously with respect to time in a corresponding manner.

The equations to simulate the typical intercept problem can be divided into four main groups. They are interrelated in use, and the quantities obtained are instantaneous.

- 1) Aerodynamic Equations - Represent the flight characteristics of the F-4B aircraft; producing its acceleration, turn rate and climb rate.

- 2) Kinematic Equations - Represent the position and attitude of the fighter and target as viewed from the early warning reference point producing distances north and east from the EW station.
- 3) AI Radar Equations - Represent the basic geometry between fighter and target, producing the elevation and azimuth angles of the target with respect to the fighter.
- 4) Fire Control Computer Equations - Represent identical equations which are mechanized in the Airborne Missile Control System of the F-4B, and produce visual indication on the radarscopes of favorable conditions for firing a missile. Quantities such as the maximum error in heading the missile can tolerate, and range to the missile are obtained.

The quantities -- in the form of voltages -- obtained from the continuous solution of these equations are applied to the units of operating hardware. Quantities obtained from the operating hardware are entered into the equations.

Full size hooded cockpits are provided for a pilot and a radar intercept officer. The pilot's cockpit is equipped with a control stick, rudder pedals, throttle, instrument panel, radarscope and a communications set; the radar officer's cockpit is equipped with radar control set, radarscope, communications set and an instrument panel. External to the cockpit and supplementing the radar is a rack of electronic circuitry which:

- 1) Converts analog computer outputs to video for radarscope displays.
- 2) Provides realistic radar switching sequences and modes of operation.
- 3) Simulates enemy countermeasure effects such as chaff drop, angle and range deception, noise and voice jamming.

The command station is located in a separate room; its main simulation equipment is a plan position indicator and a communications set.

A fully operating prototype of the ATDS weapons system, is being evaluated in another section of the Computer Division laboratory, with a tie-in to the F-4B intercept Simulator. Anticipating this requirement, the cockpits were prewired for installation of the tie-in unit, an AN/ASW-13 digital data communications set. It displays vectoring information automatically from inputs received from the ATDS system. When the simulator is used in ATDS operations, the command station is normally disconnected.

Suppose in a simulated situation a hostile aircraft is detected at a range of 350 nautical miles due north of the command station. The target is at 40,000 feet above mean sea level and is flying south at Mach 0.9. An F-4B fighter is assigned to intercept. The combat air patrol station is angularly removed 40 degrees east of a line extended northward from fleet center; the F-4B is initially flying at Mach 0.9 too.

These conditions are initial conditions which must be specified and set into the simulator before actual operations begin. Each condition may be prescribed over a wide range of values, to simulate several intercept situations. The target and fighter appear as blips on the PPI at the control station. When the simulator is turned on, the air controller notes movements of the blip, and calculates the heading and speed the F-4B should take. He then radios this information to the pilot while communications jamming, if present, interferes. The pilot manipulates the control stick, rudder pedals, and throttle as he would in an actual flight. These motions produce changes in the analog equations, and such changes are instantly reflected in the cockpits as instrument movements and radarscope displays, and in the control center as a scope display.

The intercept can be divided into two phases--search and attack. In search the pilot continues to be vectored by the air controller, while the radar officer manipulates the radar control and searches for the target on his radarscope. Upon detecting the target, the radar locks on and the automatic tracking mode of the radar is simulated. The scope display channels are switched to receive fire-control computer inputs; the attack phase begins. The pilot now has on the scope a visual indication of how to maneuver the airplane to a favorable missile-launch position, when to fire a missile, and when to break away. At any time during flight, the various countermeasure effects can be switched in or out.

Figure 3-6 is a functional block diagram of the simulator.

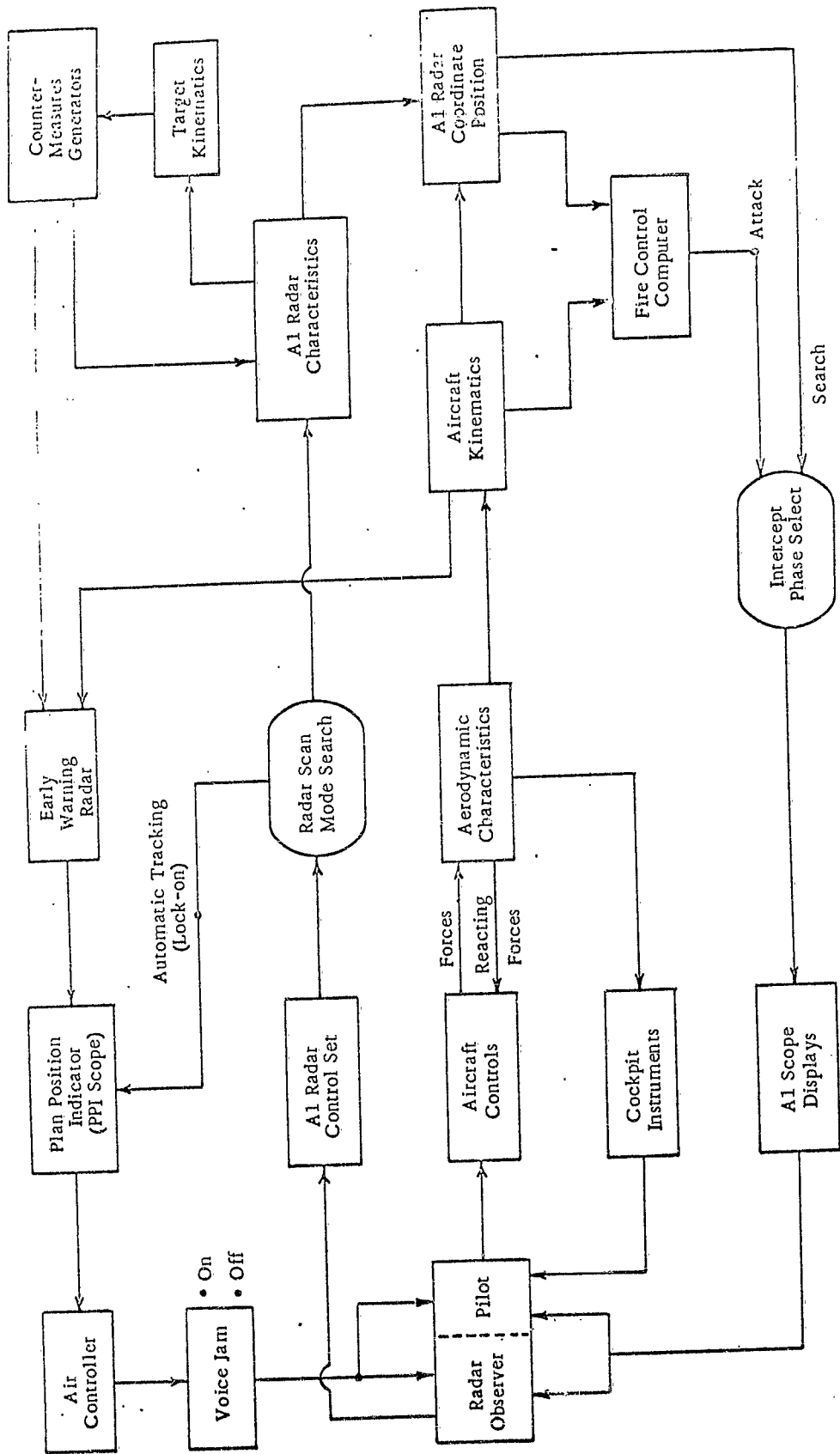


Figure 3-6 Functional Block Diagram of F-4B Intercept Simulator

3.5.4 ATDS Test and Evaluation

The modern tactical environment places increasing demands on mobility, flexibility and dispersion of a Carrier Task Force. Gathering, transmitting and processing tactical information into decision making form for the Fleet Commander and his staff has grown proportionately. The Airborne Tactical Data System (ATDS) has been developed to provide improved data acquisition and cross-tell to permit rapid appraisal of the tactical situation, and rapid solution of detailed problems for the precise control of the Task Force.

The ATDS is an airborne system designed to provide both intercept control and early warning to the fleet.

A basic role of the Naval Missile Center is to conduct engineering test and evaluation of Navy weapons systems. From this point of view, the ATDS is an experimental system, and the purpose of the current test and evaluation activities is to determine the feasibility of this concept.

The ATDS evolved to provide automated processing of many functions such as; automatically processing the radar data and detecting the presence of a target, automatically tracking the target and automatically reporting this target to some surface activity as the Naval Tactical Data System, automatically vectoring an interceptor to a point where its own system takes over control of the intercept.

The modern ATDS carrier-based system utilizes the Grumman E-2A (W2F-1) and has an extensive complement of associated electronic equipment including: display equipment, communication and data transmission equipment, radars, identification equipment and data processing equipment.

The required system command and control functions of the ATDS include:

- 1) Detection
- 2) Acquisition
- 3) Identification of Target
- 4) Evaluation of Threat Potential

- 5) Weapon Assignment
- 6) Transmission of Control Data to interceptors
- 7) Transmission of Tactical Data among the various Elements of the Fleet
- 8) Provide Accurate Navigation Computations

These functions are automatic, semi-automatic, and manual as required by particular missions.

The ATDS command and control functions are implemented by:

- 1) Detection Subsystems
- 2) Navigation Subsystem
- 3) Communication Subsystem
- 4) Data Processing Subsystem
- 5) In-Flight Performance Monitoring

To exercise these subsystems a complex of analog computers and other support devices, such as inertial subsystems and an air data computer, were built so that the system would function in the laboratory as a complete system. Tests of the ATDS systems were run both with the laboratory set and with conventional ATDS craft. Of particular interest is the laboratory-based tests. The test series of the laboratory ATDS was conducted in the following modes:

- 1) Test runs using simulated inputs.
- 2) Test runs in the laboratory using live inputs from radars scanning targets in the sea test range.
- 3) Combination of live and simulated inputs

In addition to these test modes, computer programs for the IBM 7090 were written to do computer simulation of some of the computer functions such as detections, tracking and vectoring. The programs are written to duplicate the computations performed in

the computer equipments of the ATDS craft. Using this technique, the logics of the system can be exercised to verify conditions and tests that would happen only rarely in live testing.

One of the important aspects of the test and evaluation effort is the series of controlled laboratory tests. These test runs in the laboratory ease data gathering and recording and, through the use of simulated inputs, provide a very large data base for subsequent evaluation.

The detection subsystem has three principal components:

- 1) Search radar set
- 2) Radar recognition set (IFF)
- 3) Computer detector

The Search Radar supplies raw video to the Computer-Detector and to the AN/ASA-27 Computer-Indicator Group displays. Detection probability for weak radar and IFF legitimate target return is enhanced by correlating received signals on a sweep-to-sweep basis, to permit lower thresholds than would otherwise be possible.

The Radar Recognition Set transmits and receives IFF data, compares received IFF data with previously stored data and transmits "verification" signals and raw IFF video for display, to the AN/ASA-27 Computer-Indicator Group via the Computer-Detector.

The Computer Detector determines target height by special processing of search radar video. Target position data is converted from polar (R- θ) to rectangular (X-Y) coordinates and, together with target height data, is transmitted to the Computer-Indicator Group for further processing and display.

The Communication Subsystem has two principal aspects:

- 1) Communications between fleet elements.
- 2) Command Data link to and from the interceptors.

The multi-purpose Communications System (AN/ASQ-52 or MPC data link) provides two-way digital transmission of target data between surface units and other AEW aircraft. Transmitted target data consists of such items as:

- 1) Originator's identity and position.
- 2) 3-D target position and velocity.
- 3) Target identifier
- 4) Target type, threat and engagement status.
- 5) Track quality and handover status.

The Digital Data Communications System (AN/USC-2 data link) transmits guidance to all interceptors, and receives status data from interceptors able to reply. Transmitted guidance data includes:

- 1) Controlled interceptor identifier.
- 2) Target slant range and target ground velocity.
- 3) Interceptor/target range and bearing, attack heading and time-to-go.
- 4) Command heading, speed and altitude, target altitude and action to be taken.

Received interceptor status data consists of such items as:

- 1) True Air Speed
- 2) Altitude
- 3) Heading
- 4) Fuel Status
- 5) Armament Status

The Data Processing Subsystem is a complex of computer equipment which has, as one of its principals, the Computer Indicator Group.

Target data received from the Computer-Detector is correlated with target track data stored in the Computer-Programmer to update existing tracks and to initiate new tracks.

Automatic tracking of maneuvering targets is by linear filters and unique three-dimensional adaptive gating techniques in the automatic tracking unit, the special purpose digital computer of the Computer-Programmer. In addition, friendly aircraft are tracked by IFF and beacon returns for greater positional accuracy as well as greater blip/scan ratios than are usually attainable from skin-track.

The Computer-Programmer continually extrapolates the position of all unknown and hostile airborne targets to determine threat potential to a previously manually-entered defended point, and to assign an appropriate threat priority index which ranks targets in order of threat. When the automatic threat evaluation mode has been set up, the target representing the greatest unassigned threat is made available for automatic weapon assignment and is also displayed to the operators. Manually-designated threats automatically receive the highest priority, whether in the manual or automatic threat evaluation mode.

In this operator selected mode, the greatest unassigned threat is submitted to the intercept computer for Interceptor assignment. Stored data on the available controlled interceptors is then automatically examined. On the basis of aerodynamic capability, fuel status, radar/weapon capability and time-to-go, the Computer-Programmer assigns, computes, and transmits intercept instructions to the interceptor that can best counter the threat. This assignment process continues until all available interceptors have been paired with threats. Alternatively, weapons may be assigned manually by the operators, then the operators pair available interceptors one-by-one with a selected threat and, based on the appearance of the display, manually assign one of the interceptors, until all available interceptors have been assigned.

Guidance instructions are automatically and continually computed for simultaneous control of engaged interceptors. These instructions are based on an intercept computer program derived from the characteristics of weapons expected in the operational inventory. In addition, the terminal approach path is automatically computed on the basis of weapon requirements and AI radar characteristics to ensure maximum kill probability. Automatic transmission of guidance instructions to the

interceptors is via the AN/USC-2 data link as well as automatic receipt of interceptor status reports from those interceptors capable of replying via AN/USC-2. Progress of each engagement may be observed on the Control-Indicator CRT displays.

Reports consisting of positional data, velocity and category, on targets selected by the operators for general reporting (or handover to other AEW aircraft or to surface elements), are automatically organized by the Computer-Programmer and transmitted via the AN/ASQ data link. Similarly, the system can receive target data via the AN/ASQ-52 from other elements, can correlate reports with stored target tracks, and can track and display such targets to the operators. Status and order messages are also automatically received, processed and answered.

System performance is automatically monitored in flight by preprogrammed self-check routines in the Computer-Programmer. These routines are performed continually, periodically, or on manual instruction. Self-checking includes automatic assessment of adequacy of performance and system status. System status is displayed on the IFPM test set for operator monitoring and decisions related to operation in a degraded mode. Test targets are carried in the system (in addition to live targets) to provide continual verification of system performance. The IFPM system also expedites fault isolation using only the permanently installed aircraft equipment.

Simulation of input data is of several forms. The input of simulated radar data is shown in Figure 3-7 and consists basically of range and azimuth voltages entered into the system at the point where the true aircraft sensors would pass on this same information. To simulate these inputs, two characteristics of the sensor data must be closely imitated:

- 1) Shape of the pulse
- 2) Time of arrival

The pulse shape is manufactured in either the IFF simulator and the video simulator. The time of arrival at the Computer Detector is controlled by the target generator computer. A computer of some capability is required to produce a correct equivalent of the three radar returns which are normally received from a single target. The

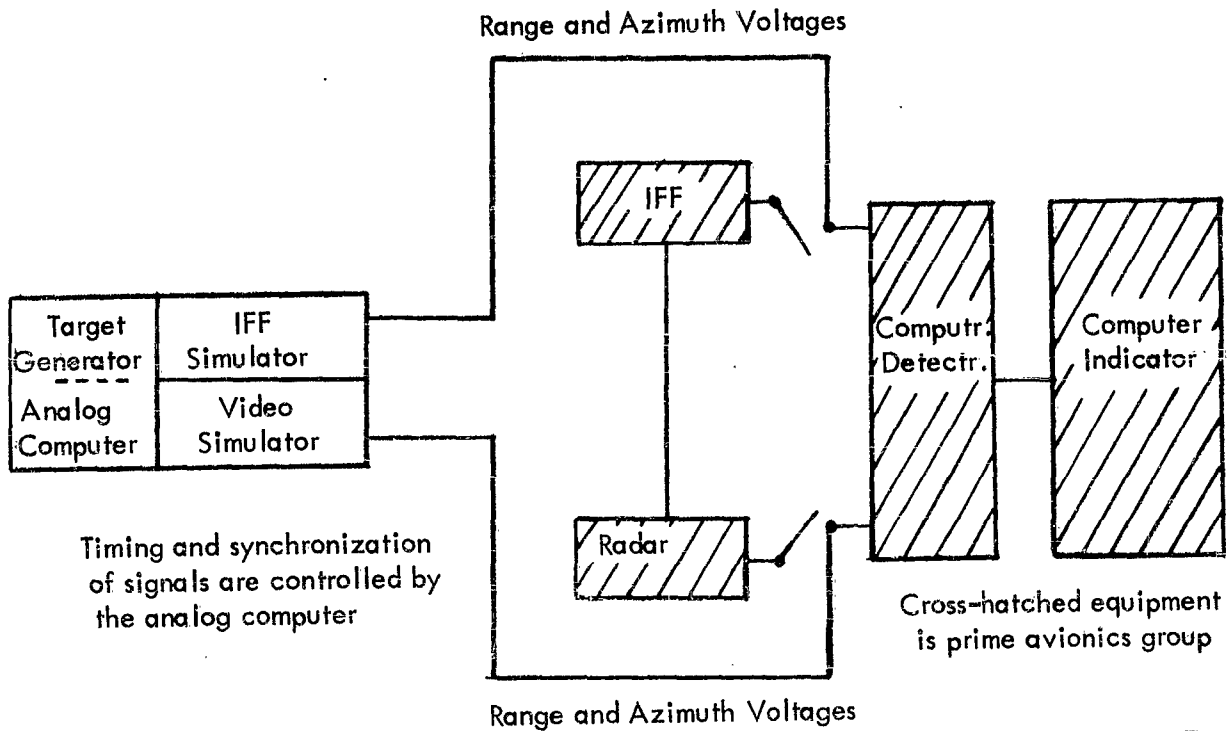


Figure 3-7 Radar Data Input Simulation

first return is direct and allows the distance computation, the second two are bounce returns. The bounce return allows the computation for target height, knowing the time lapse between returns and height of the E-2A aircraft.

The other simulated inputs are also analog computer derived and provide inputs that would normally come from the inertial platform and from the doppler radar.

The ATDS laboratory set can operate with both live and simulated interceptors directed against either live or simulated targets. To use the cockpit simulator the flight characteristics of the type of interceptor it is "pretending" to be are programmed into the analog computers. The cockpit simulator then relates to the ATDS laboratory equipment set as in Figure 3-8. The cockpit communicates with the ATDS system through the ASW-14 and the ASW-13 data link normally found in an operational fleet interceptor.

Two simulation sources are associated with the Communication Subsystem and provide for two types of capability:

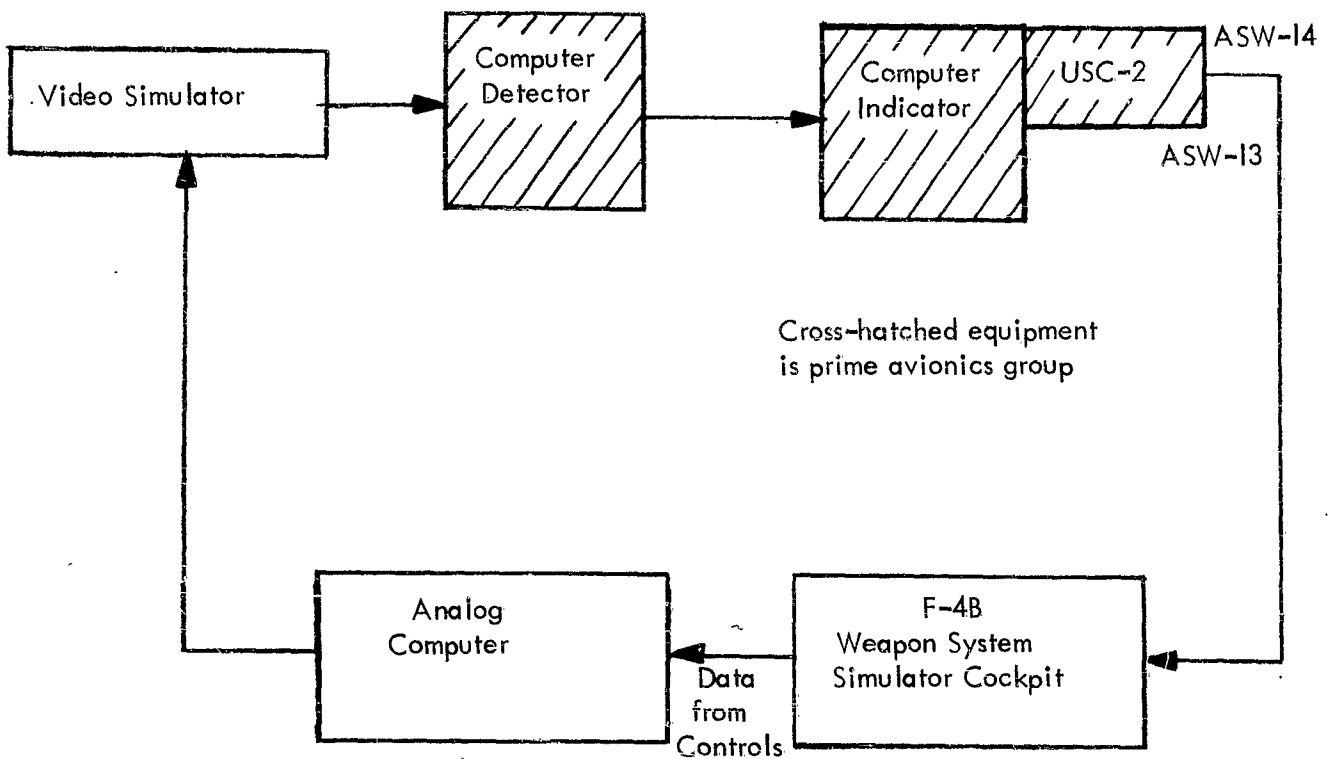


Figure 3-8 Using The Cockpit Simulator

- 1) Playback of previously recorded live inputs.
- 2) Simulation of messages normally generated by other sources; e.g., NTDS.

The laboratory is able to "monitor" any sea test range operation and record any data items of value to its test series. These sets of real world data may be played back repeatedly into the system for isolation of system errors or verification of corrections made to the laboratory model of the ATDS.

3.5.5 ATDS Integration Tests with Companion Systems

The ATDS System provides a far-ranging extension to the fleet-centered NTDS. It is also possible for the ATDS and MTDS (Marine Tactical Data System) to communicate and exchange information about tracking and other target reports. In this case, the ATDS provides a seaward extension of the MTDS.

The Naval Missile Center supervises ground technical tests of tactical data systems. In this role, joint tests involve interaction with ATDS located at Point Mugu, NTDS located at Point Loma and MTDS located at Santa Ana. (See Figure 3-9).

The primary integration concern is with conducting compatibility tests between ATDS, NTDS, and MTDS to investigate interface in:

- 1) Language basis
- 2) Language interpretation

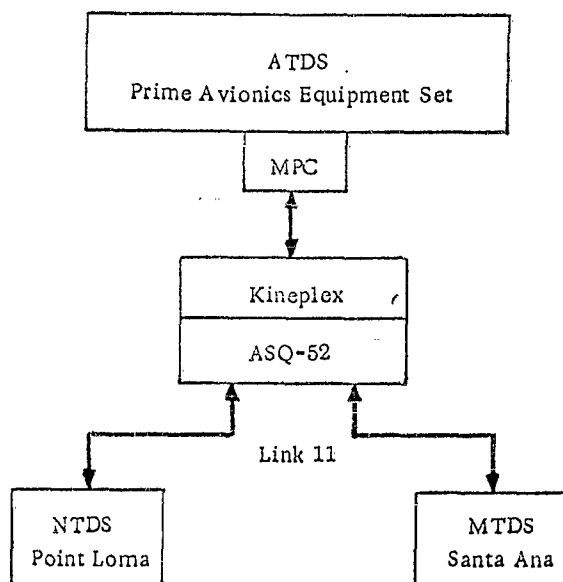


Figure 3-9 Interactions Between ATDS, NTDS, and MTDS

For basis, the interest is syntactic and centers around the allowable symbols used by the system and the rules concerning the various symbol strings of transmission. For interpretation an effort is made to investigate the relative interpretations of these symbol strings. Particular emphasis is placed on investigation of possible sources of intra-system error in such as:

- 1) Track correlation
- 2) Navigation

- 3) Track quality measures
- 4) Target category assignment algorithms
- 5) Mathematical transforms

Compatibility is verified by performing joint tests with communication in pairs between the three installations. The ultimate objective is to achieve an integrated tactical data system complex.

The target reporting function, the air-to-surface link for communication with the various tactical data systems, makes use of the Collins Kineplex ASQ-52. This unit uses parallel transfer of data.

An example of using this link is to provide the MTDS with inputs from ATDS. Often the ATDS outputs required are elementary and can be provided by an ATDS simulator. For example, to send one or two slowly changing targets to assist the MTDS in program de-bug operations does not require the ATDS, itself, to be tied up.

In particular, the ATDS/NTDS interface problem is investigated by tracking common targets and then looking at track correlation and other error sources.

3.5.6 Conclusion

In the simulation laboratory facilities at NMC, Point Mugu, and at other laboratory facilities such as NEL, San Diego, the Navy has amassed considerable experience and equipment devoted to equipment and system simulation. The evolutionary development of ACDS as an operational system requires the use of much of this capability. The Navy has facilities and personnel particularly well suited to checkout and test evaluation simulations required for the evolution of ACDS.

3.6 SIMULATION LANGUAGES

3.6.1 Introduction

Simulation languages are higher order programming languages designed to ease programming, coding and checkout of digital computer simulations.

Simulation (Sim.) languages allow speed in design and construction of a simulation since they provide for routine procedures control, and recording of data. Most sim languages were originated for a specific purpose and have since been expanded to a larger class of problems.

The earliest simulations were coded using octal and binary absolute techniques. Fine simulations may still be produced using machine language or combinations of machine languages and FORTRAN or ALGOL. The use of a simulation language is not required to produce a good simulation program. But a proper sim language makes it easier to produce a sim program, makes the designer's task an easier one, and speeds his progress.

3.6.2 How Sim Languages Work

Construction of simulations calls for lists of things, people or events, to present one of these at a time to be served or operated upon by the logic of a central model of the simulation. These lists may be few and very long, many short, or mixtures of long and short.

In each simulation, at least one operation serves the items waiting in the lists. Often in complex simulations many operations are modeled to serve lists and add items just served to other lists which in turn, are served.

Construction of these complicated models is simplified by using sim languages which provide conventions to specify creation of lists, operation and inter-dependencies of serving models, influence of time or other environmental circumstances. Each sim language uses different conventions, varying in simplicity, power and general applicability. This is because they were all created for specific purposes. Most have been expanded in scope, but the prospective user will benefit if he picks a language originally designed for a problem similar to the one he faces.

The advantage of using a sim language tailored to his problem must be weighed against the difficulties of learning a new sim language or having a computer available for which the language was written. Staying as close to the original area as possible avoids inherent limitations present in all sim languages. The more complex and complete languages may be used to simulate simple relationships and occurrences, but they are often much too ponderous.

The use of a sim language is a multi-step operation:

- 1) Develop rules for processing the lists; mathematical models, stochastic models, or combinations.
- 2) Develop rules for creation of the lists and for items entering and leaving the lists other than by being served by the primary models.
- 3) Develop relationships and linkages relating the lists and models.
- 4) Develop timing and operational considerations to execute the simulation.

The user writes the simulation using the conventions of the language chosen. Next, the computer and the simulation assembly program process the sim conventions, and produces a program in computer language. This may be in machine code such as FAP, but more often in a compiler language such as FORTRAN. This must be compiled into machine code and converted into the binary deck which is finally operated. This operation is the simulation being cycled. Answers and statistical data are recorded and printed out during and after this third pass.

Some sim languages permit use of machine code and compiler or assembly language in originally writing the simulation. Called "enrichment", this process enhances the capability of the sim language. It permits the simulation designer to code some intricate parts of his simulation in machine or assembly language and to bypass various shortcomings of the sim language. Since all simulation requirements cannot be provided for in a sim language, enrichment capability is highly desirable.

The easier a sim language is to learn and use, the more it tends to be stylized and inflexible in what it can describe. The more capable a sim language is, the more complex its rules and the more difficult it is to use.

3.6.3 Some Simulation Languages

Some of the better known sim languages and three which are of considerable interest to simulators at the present time are discussed in this section.

- GPSS II** An IBM product, GPSS I was an outgrowth of the "Gordon Simulator." GPSS II is an enhanced and more flexible version. GPSS handles simulations of communications systems and computer systems, with many lists of varying lengths, but where the central model of the simulation is logically simple. All relationships and types of operations are rigidly specified, and GPSS cannot be enriched with any assembly or machine code. GPSS II is available for the IBM 7090-7094.
- GASP** Developed by the U.S. Steel Corporation to simulate operations in shops of steel mills. Lists may be somewhat shorter than with communications simulations; models of the simulation can be very complicated. GASP is one of the earliest of the powerful, flexible sim languages; permits the use of FORTRAN for enrichment; is compatible with FORTRAN diagnostic tools. It is available for IBM 1620, 7070-7074, 7090-7094, and CDC G-20.
- CLP** Developed by the Industrial Engineering Department of Cornell University, CLP provides engineering students with a general purpose simulation language that could be learned and used in one semester. CLP is simple in its syntactic construction and easy to learn. It is not highly stylized and has flexibility. CLP may be enriched by employing CORC compiler language statements. It is available only for the CDC 1604.
- DYNAMO** A capable sim language designed for the construction of simulations employing differential equation models. Developed at the Massachusetts Institute of Technology for the IBM 7090-7094, it is an interesting and valuable engineering tool but has limited application.
- CSL** Developed by IBM (UK) with Esso (UK) to simulate large corporate operational problems, such as the operation of a port-tank farm--refinery complex receiving crude oil by tanker and shipping output by rail, truck and barge. The real capability of CSL is not in the creation of long lists,

but in the ability to create and manipulate many complex operational models and to cascade these models in complex ways. CSL has a difficult syntax and many formidable construction rules. It may be enriched with FORTRAN. It is a "three-pass" language. The first pass is made on the IBM 1401 (U.K. Model) and the last two passes on the IBM 7090-7094. It is not now available in the U.S. nor outside of IBM (U.K.), but it is expected to be available generally in Britain in late 1964.

SIMPAC Developed by System Development Corporation as a research tool, it is one of the more powerful of the simulation languages. This makes it most difficult to learn. SIMPAC is run on a 7090-7094. While other sim languages for the 7090-7094 operate under FORTRAN control, SIMPAC uses SOS control and requires 14 tape drives. SIMPAC could be run on a large FORTRAN 7090 but would be cumbersome. SIMPAC can be enriched with a machine mnemonic code (SCAT). Many of these limitations are unimportant to a skilled programmer with a large 7090-7094 installation, but represent barriers to many potential users.

SIMSCRIPT Developed at Rand for more efficient preparation of simulations. It operates on an IBM 7090-7094 under FORTRAN control. It may be enriched by FORTRAN statements and by code written in FAP. This feature gives great capability and allows enrichment by the easier-to-use FORTRAN. SIMSCRIPT is complex in its syntax and rules, and is difficult to learn and use well, but has excellent documentation which includes how to get around the grammar to provide more capability. SIMSCRIPT is the most popularly used of the powerful simulation languages and will probably remain so for some time.

MILITRAN A military simulation language developed under contract to the Office of Naval Research by Systems Research Group, Inc.* Designed to run on the IBM 7090-7094, it is for the simulation of military rather than system operations. It has a sophisticated capability for relatively straightforward rules of construction and grammar. It will not be easy to learn, but should prove to be quite useful to naval analysts.

* Nonr 2936 (00)

3.6.4 The Application of Simulation Languages

Simulation programs which sim languages prepare are as efficiently coded as those a skilled programmer could write, but they are available quickly in simulation programming, understanding the problem and deciding what to do takes a long time. But once that is done, simulations may be prepared much more easily, accurately, and speedily using a sim language.

A data systems engineer has two major uses for a sim language. In design, he often wants to check performance of parts of the system or simple sets of interactions. To do this he wants a quickly used, simple sim language. CLP would be ideal, but it is not generally used, although it could probably be made available. He must use GPSS II or something more complex like GASP or SIMSCRIPT. Data system engineers would be more likely to simulate simple problems if CLP or a similar simple language were available.

The second simulation requirement of a factual data systems engineer is to simulate large parts of the system and finally the entire system. This type of simulation is normally not prepared on a short term basis, and the more powerful languages SIMPAC or SIMSCRIPT can be used. CSL, when it becomes available, will be highly desirable for these large scale simulations, and MILITRAN should prove to be very valuable.

3.6.5 Current Developments

More than one computer manufacturer is known or is reported to be preparing simulation languages, at the most powerful end of the capability spectrum. SOL has been developed by a group of system engineers at Burroughs, Pasadena. It runs on the B-5000 and is extremely powerful, reportedly as capable as SIMSCRIPT or SIMPAC. In addition, SOL may be enriched with ALGOL statements, and runs under B-5000 ALGOL control. It is also constructed in a completely different manner from the balance of the sim languages. It is "syntax oriented" which means the compiler and its conventions more closely parallel our natural language in operation, and grammar and construction are much easier to learn to use.

SOL was not mentioned in the previous section since it has not been released to the public by Burroughs, Detroit.

There are no reports of smaller scale languages in development.

3.6.6 Observations

The language spectrum available to the system engineer is thus:

GPSS II	CLP	GASP	SIMPAC
			SIMSCRIPT
			CSL
			SOL
			DYNAMO
			MILITRAN

GPSS II is capable but completely unchangeable since it cannot be enriched. CLP could probably be made available by private treaty, but it is not well known and only runs on the 1604. GASP is old, but capable and runs on several machines including the G-20. SIMPAC has great power but severe limitations. CSL is not available yet, and SOL may never be. The choice is really between GPSS II, GASP or SIMSCRIPT, and with these three languages the simulation requirements of all phases of system engineering may be met satisfactorily. But special applications make CLP SOL and CSL continue to look very promising.

MILITRAN looks especially useful for those simulations directed more at military operations than at the internal functions of some semi-automatic system. It must be realized, however that these two endeavors are often closely related. A true evaluation of MILITRAN must wait for operational simulations following its open publication.

3.7 MATHEMATICAL MODELING

3.7.1 General

Mathematical modeling means the process of deriving mathematical representations of equipment or systems. Mathematical modeling is a preliminary step in solution of a system's analysis problem. The problem might be to evaluate trade-offs between a centralized or decentralized computer organization, to determine the delays in a system caused by queues, or to determine the sensitivity of the system performance to changes in the input.

Sometimes the solution to the problem is obtained by solving the mathematical system model with analytical techniques, occasionally with hand calculations, sometimes with a computer program, and sometimes with a computer simulation. Large system models have a degree of complexity which normally eliminates all methods of solution except computer simulation.

3.7.2 The Development of Mathematical Models

The development process is divided into five steps:

- 1) System Analysis. The first task in developing a mathematical system model is to determine all the factors which affect the system. For a ship-to-air missile model, the study would involve determining all the forces which acted on the missile. This would include forces such as aerodynamic and wind pressures as well as rocket thrust and Earth's gravity. This process is much more difficult when developing a model which is used to evaluate the effectiveness of a command and control system. This task involves the study of many more factors such as radar errors, threat configuration, command organization, weapon deployment, weapon effectiveness, weather, decision errors, etc.
- 2) System Component and Environmental Modeling. Mathematical models must be obtained for each factor which affects the operation of the system. Usually, these models can be obtained by searching the literature in the appropriate technical field. Each technical field has a large amount of standard mathematical models which represent its subject matter.

If a model does not exist for some component or environmental factor, it will be necessary to develop a model. This type of modeling could be considered as basic research. It involves observation, experimentation, correlation, formulation, etc. For example, the development of the kill probability function of a ship-to-air missile might require an analog simulation, plotting results, and curve fitting a polynomial approximation.

- 3) Selection of Models. The third step is selecting a set of models to establish a level of uniformity of detail throughout the system model. For a missile model, the selection of models would be dependent on the size and accuracy of the contributing forces. This selection of models could be considered an error equalization or leveling process.
- 4) Translation of Models. The set of models which comprise the system model generally come from many sources. Consequently, the frame of reference and nomenclature of the models vary. For these models to function together as a unit, they must be translated to a common frame of reference or coordinate system. For example, the forces on a missile must be expressed in the same coordinate system before they can be added. Sometimes, it is necessary to establish more than one coordinate system and derive the transformations from one system to another. The motion of a missile, for example, is normally calculated in an Earth-centered coordinate system. To calculate the radar look angles of the missile, it is necessary to translate the missile's position to the radar's coordinate system which has its origin on the surface of the Earth.
- 5) Integration of Models. After the models have been related, the next step is to define a procedure for performing the required calculations. To calculate the acceleration of a missile, for example, the drag must be calculated; to calculate the drag, the relative velocity must be calculated; to calculate the relative velocity--etc. Figure 3-10 shows a group of models which are graphically integrated to form a system model. Note how the output of one model is the input of

another. There are three models in this system model, the airframe model, the seeker model and the target model.

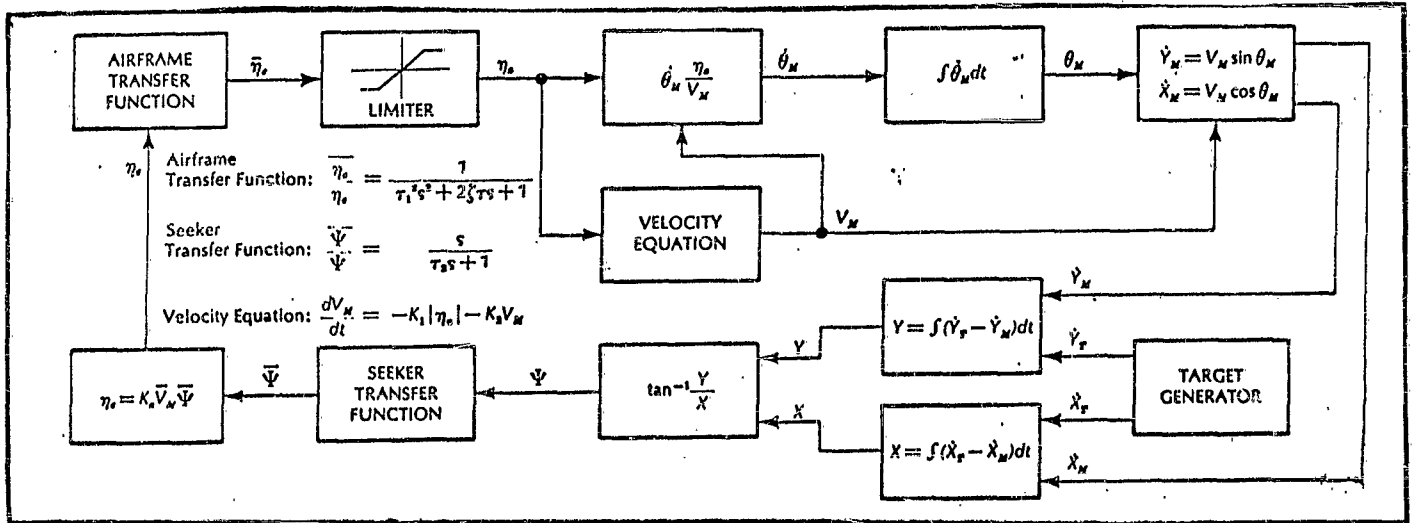


Figure 3-10 Two Dimensional Air-to-Air Missile Simulation Block Diagram

3.7.3 Model Types

There are nearly as many types of basic mathematical models as there are types of mathematics. They are arbitrarily classified into five groups; analytical, geometrical, logical, statistical, and empirical.

The term "analytical" is used to include algebra and calculus functions and is not meant in its strict mathematical definition. Newton's second law, $F = MA$ is a good example of an analytic model. Another frequently used analytical model is the acceleration, g , due to the Earth's gravity,

$$g = g_0 \frac{R^2}{(R + h)^2}$$

where g_0 is the gravity at sea level, R is the radius of the Earth, and h is the altitude above mean sea level.

Geometrical models are drawings which are generally made to scale. They are generally used when analytical models are unduly complex, for example, for graphic solutions such as a radar coverage diagram. Such diagrams can be examined to determine the coverage available at various altitudes.

Logical models are used to describe the relationships, procedures, rules and decisions involved in logical systems such as a 'black jack' playing system. They are generally in the form of block diagrams which can be considered as schematic diagrams of a set of boolean algebra functions.

A statistical model is a mathematical formula which describes the relationship between a classification of erratic data. An average of surface transport traffic would be a statistical model. Statistical models can be used to describe transmission noise, radar errors, weapon effectiveness, human behavior, and other unpredictable phenomena.

Empirical models are tabular or graphical functions such as aerodynamic drag curves, rocket thrust data, spring tension characteristics, steam tables, etc. They are exact functions which do not have analytic descriptions. These models are constructed by accurate measurement of the physical subject. The tension of a spring is a nonlinear function of the displacement. To model a spring's characteristics accurately, the tension in the spring must be measured accurately throughout the range of spring displacement.

3.8 QUEUING MODELS FOR ACDS

3.8.1 General

One of the fundamental analytical problems which face ACDS system planners is to discover how fast a certain node reacts under varying loads of data and operator requests. The random arrival of data at an ACDS node, and the random requirements to perform various tasks, results in the formation of waiting lines or queues of data, tasks, or requests for service. The reaction time of the node depends upon the speed of processing at the node, the rate of data and request arrival, and the size of the queue at any given moment. The analysis of how this reaction time varies is fundamental to the proper internal configuration of a particular ACDS node, and to the manner in which these nodes are netted together.

Queuing models provide a powerful tool for the analysis of these problems. This section shows how simulation and queuing modes may be applied to the ACDS nodal analysis task, but does not present more than the basic concepts involved. A number of fine mathematical texts and papers exist which discuss queuing theory in more than enough detail for the naval system planner.

The reaction time of the node at a particular instant is the total time spent (by a task or a segment of data) waiting in the various queues, plus the time required to perform the requisite processing. This total time is referred to as the "throughput" time for that task or information. The throughput time is a function of the processing capability of the system and the arrival rate of requests for service, as well as the rate of data input. If the arrival rate and the service time are constant or can be controlled, the planning problem reduces to an arithmetic calculation. That is, if one computer can service a request in 18 seconds and requests arrive every 5 seconds, four computers are required.

In command data systems, the problem is much more complex since arrival rates are variable and processing speeds do not remain constant*. A mean arrival rate may be

* In the Target Evaluation Weapons Assignment problem, not only do the inputs arrive randomly, but processing time increases as the system load increases.

1 item every 3 seconds, with a variance of ± 2 seconds, and the service rate of an operator may be 1 item every 6 seconds, with a variance of ± 4 seconds. Although the planner may be able to determine these characteristics quite accurately, he theoretically cannot eliminate the possibility of a waiting line (however short) without providing a large number of operators. He is confronted with sets of trade-off decisions.

A typical planning problem is: How many processors should there be to have an acceptable queue of requests for service? For processor we must think of either computing machinery or human operators. The acceptable length of a queue depends upon the importance of the requests, taken in the context of the environment of the system at that moment in time. Twenty seconds is likely to be an acceptable waiting time for persons requesting communications circuit information, but unacceptable for short range missile assignment to an enemy attacker.

Statistical (or analytical) queuing techniques, and simulation queuing techniques, can be used to study the throughput time of ACDS by analyzing characteristics of the waiting times of the various queues. The total system must be considered as a composite of four kinds of activities: an input, a queue, a processor, and an output. A particular process may have many of each kind of activity. Statistical queuing techniques attack the problem by representing these activities with analytical and stochastic or probabilistic models. Simulation queuing techniques attack the problem by representing the logical relationship of these four activities by flow diagrams, and subsequently by building a computer simulation which performs the functions of the system in the manner specified by the logical flow diagrams.

3.8.2 Statistical or Analytical Queuing Models

The object of analytical queuing techniques is to obtain a statistical model of the queue or the throughput time. The model might be the probability of a job being processed in X minutes as a function of the variation in arrival rate or the probability of a job waiting in line Y minutes as a function of the mean processing time.

Building the model is by combining and manipulating statistical models of the input and processors. The inputs are generally modeled by Poisson distribution functions,

and the service rates of the processors are generally modeled with exponential distribution functions. These models are used because they are rather readily manipulated.

A statistical queuing model was made to analyze the performance of the RW-40 data processing system.* This is a multi-computer system which processes randomly arriving jobs requiring varying types of data processing. The entire system is controlled through a central exchange by the operation of a master control computer program.

The model represents several operations which will probably be required of part of ACDS. Among the functions of the master control program are:

- 1) Detection of input requests for service
- 2) Interpretation and classification of requests
- 3) Assignment of requests to the proper area for solution
- 4) Assignment of slave computers to problem lists
- 5) Supervision of internal operation of the system
- 6) Supervision of the "assignment" rules of the central exchange
- 7) Monitoring progress
- 8) Supervision of handling of queues internal to the system
- 9) Supervision of the handling of system malfunctions
- 10) Dissemination of results
- 11) Liaison with human operators

To perform a queuing analysis of this system, these four simplifying assumptions are made:

* Rothman, Stanley, RW-40 Data Processing System, June 1959, Data Systems Project Office, Ramo-Wooldridge Div., Thompson-Ramo-Wooldridge, Inc.

- 1) Jobs are serviced on a first come first served basis.
- 2) Reliability, repair and preventative maintenance are lumped into a flat reduction of available computer time.
- 3) The arrival of problems is a Poisson distribution.
- 4) Problem-service time, computer down time and computer up time have negative exponential distribution.

Except for the first assumption, these are not unreasonable assumptions for an ACDS nodal analysis. Even within a tactical system there are many individual processing areas where queues are served on a "first come" basis. The development of queuing theory has progressed today to the point where unusual queue behavior and processing requirements may be modeled. Some of these newer modeling techniques are of interest to ACDS analysts and planners.

There are distinct limits to the detail the ACDS planner can obtain if he considers the node as one set of queues and one processor. A greater degree of detail may be achieved by considering the node as a system having sets of queues and several processors. This is probably a more faithful concept of an ACDS node. In addition to the queue which holds the original input, there may be numerous internal queues during processing due to a multi-stage processing requirement. For example, the processing may require retrieving information from an auxiliary storage device which could result in a queue of information requests. There could also be queues for the use of output devices.

Queuing theory can also be used to model a network of queues where the output of one queue/processor is the input of another queue/processor. This is possible partly because of queuing phenomenon. Namely, if the input arrivals have a Poisson distribution, then the output also has a Poisson distribution, independent of the service rate distribution. However, the theory is somewhat limited by the number of models which may be used for the inputs and processors.

In addition, analytical statistical techniques only provide gross information about the system service characteristics as they interface with the service requests. A queuing analysis may show what the mean service rate and variance should be to achieve an

acceptable throughput time, but it cannot give much insight as to what the equipment management should be to obtain such system characteristics. For example, the RW-40 system detects input requests in two different modes; an interrupt mode and a sense mode. When the traffic is heavy, it operates in a sense mode where it periodically scans input alert indicators. When the traffic is light, it would be inefficient to scan the indicators frequently for possible input. Consequently, the computer enters the interrupt mode which interrupts other useful processing whenever an input arrives. The critical problem here is to determine at what point light traffic becomes heavy and vice versa. Solving this type of problem normally involves too much detail for use of analytical means. When problems become this complex, simulation should be used to attack them.

3.8.3 Simulation of Queuing Models

Queuing networks which are modeled with flow diagrams can incorporate the detailed characteristics of the network, because these flow diagram models are implemented with computer programs which can be, for practical purposes, unlimited in complexity.

The operation of queuing systems are studied by using the Monte Carlo simulation technique. This technique simulates the operation of the system by generating a large number of service requests using computer programming techniques. These service requests (or other functions such as service time) can be generated so that they approximate any desired distribution function. The length of the queues, the average throughput time, or other unknown characteristics, are recorded as the simulated queuing system is operated in its simulated environment.

It might be discovered, for instance, that the queue waiting to use a teletype varies sinusoidally between five and ten messages. This information might require the adoption of a selective teletype output program to select messages on a priority basis, rather than a first come, first served, basis.

Figure 3-11 is a simplified block diagram of a queuing model simulation for the RW-40. The logic of the master control program is shown on the right side, and recording required to obtain the analytic data, is shown on the left of the figure.

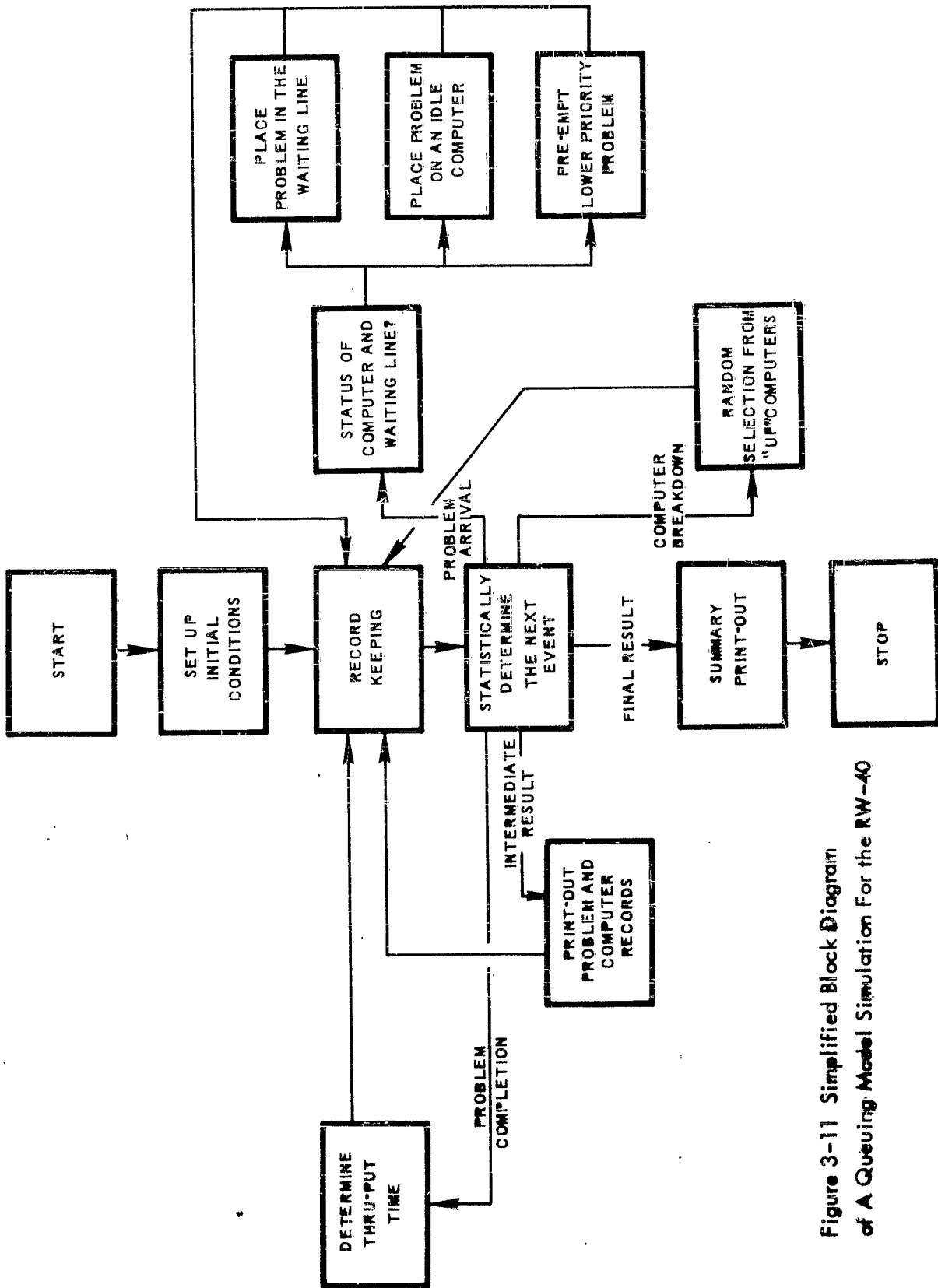


Figure 3-11 Simplified Block Diagram of A Queuing Model Simulation For the RW-40

Figure 3-12 lists the results of a series of RW-40 simulations. The "Remarks" column shows how the environment was varied to test the system. The "Problem Class" column shows the probability, "Prob", that any problem in that class of problems would be completed in time "t".

This type of data is of great value to system planners. It enables them to check the validity of their designs during the planning stage.

3.8.4 Simulation Languages for Queuing

Simulation languages have been written which reduce the time and effort required to program the simulations of queuing systems. These languages provide the building blocks for construction of the model itself, and the program system to run the model and perform the recording.

While all other popular simulation languages are more complex than IBM's GPSS II, the following excerpt from the GPSS II manual describes the nature of all simulation languages. Simulation languages are discussed in Section 3.2.6 of this volume.

"The simulator allows the user to study the logical structure of the system. The flow of traffic through the system may be followed, and the effects of competition for equipment in the system may also be measured. Computer output may be arranged to provide information on:

- 1) The volume of traffic flowing through sections of the system.
- 2) The distribution of transit times for the traffic flowing between selected points in the system.
- 3) The average utilization of elements in the system.
- 4) The maximum and average queue lengths at selected points in the system.

Various statistical and sampling techniques may be introduced into a GPSS II model. Levels of priority may be assigned to selected units of traffic, and complex logical decisions may be made during the simulation. It is also possible to simulate the interdependence of variables in the system, such as queue lengths, input rates and processing time."

REMARKS	PROBLEM CLASS										CONCLUSIONS
	2		3		4		5		t	PROB	
	PROB	t	PROB	t	PROB	t	PROB	t			
1 BASIC PROBLEM MIX	90%	26S	90%	43M	90%	14M	90%	14M	33%	7HR	SLIGHTLY UNSTABLE CLASS 5 PROBLEM WAITING LIST GROWING
2 BASIC PROBLEM MIX BUT RESTART PENALTY REDUCED FROM 15 SEC. TO 1 SEC	90%	26S	90%	30M	90%	6M	90%	6M	90%	7HR	STABLE SITUATION
3 BASIC PROBLEM MIX BUT CLASS 2 PROBLEM ARRIVAL RATE UP 50% WHILE CLASS 5 PROBLEM ARRIVAL RATE DOWN 50%	90%	26S	73%	40M	83%	35M	83%	35M	74%	24HR	UNSTABLE CLASS 5 WAITING LIST GROWING
4 BASIC PROBLEM MIX BUT CLASS 4 SERVICE TIME UNIFORM 0-1½ MIN INSTEAD OF 0 TO 3 MIN	90%	25S	90%	35M	90%	8M	90%	8M	36%	5HR	STABLE SITUATION

Figure 3-12 Typical Results of A Queuing Study

3.8.5 Queuing Analyses for ACDS

Queuing analysis is an evaluation tool to determine the capability of system designs. Queuing analysis can also be used during the design phase to establish design limits or guidelines, such as minimum computer speed, transmission speed, memory size, etc. Here the accuracy of the models is not critical. The models of the inputs selected are the upper limits of what the system is required to service.

After the inputs or requirements of the system are defined, models of plausible system designs are developed. These models are matched with the input models to evaluate the service performance or throughput time of the system design. When the service rate of the system can always accommodate the input rate, queuing analysis is not required. However, as the service of the system becomes marginal and queues form, it becomes very difficult to estimate the performance without a queuing analysis, especially if the arrival and service functions fluctuate; and in command data systems this is nearly always so.

The results of a queuing analysis may show that one queue causes the early degradation of the system. By increasing the capability of the system in that one area, it may be possible to upgrade the throughput of the system. This might involve adding another teletype, increasing a transmission rate, adding another computing unit, etc.

In this way, the queuing analysis can be used to evaluate the performance of a number of candidate systems and to show their maximum capability. The outgrowth of the analysis is a general description of a system which meets the requirements of the input model. These general characteristics such as computer configuration, information exchanges, processing speed, memory requirements, etc. can be used as guidelines for detailed design.

Note that the models of the system inputs need not be accurate as long as they are accepted as the upper limits on the system. Consequently, queuing analysis cannot be applied to design problems until the system user is prepared to define the inputs or requirements of the system.

However, after the design is complete, the model must represent the design as accurately as possible. This is necessary to obtain an accurate estimate of which

queues lengthen first, and where the performance of the system degrades rapidly. This information is then used to further refine the system design and its operational limitations. This cycling of model to design and back to model is ideally suited to the process of evolutionary improvement to existing capability. Once the basic model is established and refined, proposed changes may be "plugged in" for their evaluation.

3.8.6 Summary

Queues are major components of on-line systems such as ACDS. As the inputs or requests to these systems increase and the queues lengthen, the throughput time or service degrades. Queuing analysis can be used to study the performance of systems under these circumstances.

In general, queues are very sensitive to changes in the input rate and service rate of a system, especially when the queues are long. This is because queues have non-linear characteristics which make them increasingly sensitive to system inputs as the queue becomes long.

Queuing analysis can be used to provide information about the service capability of command data systems and insight to delays in the system. Analytical queuing techniques are better suited for determining gross characteristics of systems, such as initial queue lengths and throughput time of the total system.

Simulation of queuing systems can provide more detailed information about the system, such as the effect of equipment management on the throughput time. Simulation languages are effective tools for reducing the time and effort of implementing queuing models.

However, the validity of the results obtained with queuing analysis and queuing system simulations is dependent upon the accuracy of the models of the system and the system inputs.

In some systems which normally operate with queues (such as a large telephone exchange) the arrival of service requests may be predicted quite accurately. If there is some abnormal number of requests for service, a temporary degradation of service is acceptable.

Large economies in equipment may be effected in these systems by the sharing of equipment on the basis of predictable loads and queue lengths.

This is not so with most tactical data systems. It is difficult to model accurately the combat arrival rates of data and requests. It is also intolerable to have abnormally long queues in certain sensitive parts of the system. This results in increased equipment costs because of the normally unused capacity required to prevent excessive queuing during abnormal conditions.

This does not mean that queuing analyses cannot be applied to ACDS. On the contrary, it should be applied to ensure that ACDS nodes are designed which will only form queues of acceptable length.

3.9 DATA BASE USAGE AND UPDATE MODEL*

3.9.1 General

To arrive at a reasonable preliminary design concept for one node of ACDS, it is necessary to have (among many other pieces of information) four parameters which define the data base of that node. They are:

- 1) The total size of the data base for that node.
- 2) The location of that data base. This is, how it is fragmented among various physical locations, such as separate computing facilities or even separate ships? Or, is it centrally located in one computing complex?
- 3) How often is the data base refreshed? This determines the update traffic.
- 4) How often is the data base used? This determines the access traffic. This is two-way traffic to the data base (question going out - answer coming back) if the data base in question is not in the same computer as the Command Post Program.

The purpose of the model described is to provide a tool for estimating the update traffic and the usage traffic once the size of the data base has been estimated.

For most planning purposes, the data base of an ACDS node is that information which the nodal computer programs must have regular access to -- not all of the information in the system. For example in Figure 3-13, the Command Node Computing System contains one portion of the nodal data base, but other portions of the nodal data base are in Subordinate Unit, Staff Unit, and Adjacent Unit computing systems to the extent that the Command Node System can ask for the values of specific items and get answers automatically.

* The discussion is based upon unpublished work performed in 1963 by J. W. Hedenberg and E. K. Campbell of Informatics Inc.

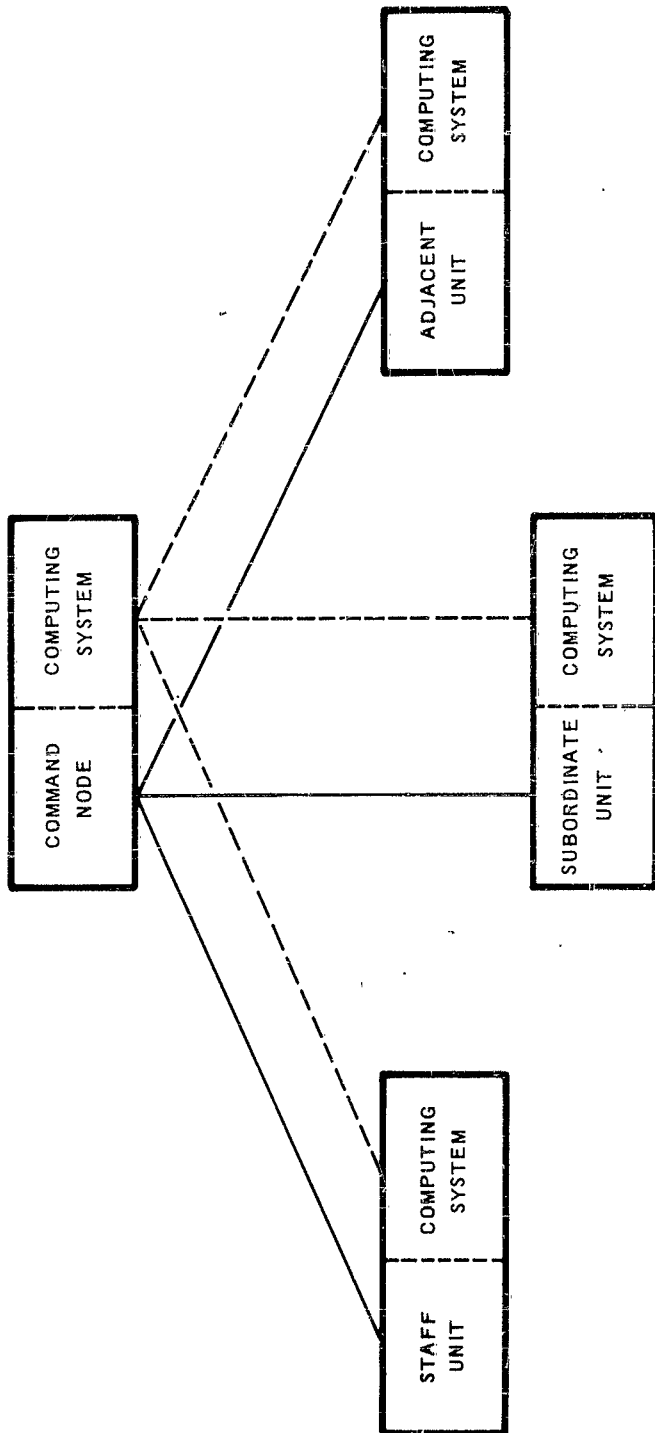


Figure 3-13 Relations of Computing Systems

information which is automatically available -- even though dispersed in systems of connected elements is, in reality, an integral part of the Command Post's data base. This is so since, under some other possible system configuration, it could all be located in a bulk data store at the Command Post Computing System.

The purpose of the model developed here is to show how data base update and access traffic varies as a function of data base size and according to certain arbitrary assumptions. This section explains the mode, its concept and its use in the analysis of ACDS planned problems.

This model is only of use after the size of the data base has been estimated.

3.9.2 Basic Concepts Affecting Data Base Treatment in ACDS

To arrive at useful conclusions concerning the possible configuration of a Strike Task Force Command Node data processing facility, it is first necessary to make some assumptions about the logical structure of the system, and about the nature of the data base to be contained within it.

For purposes of example, the structure of the Command Data System is conceived to be as follows:

- 1) At some central location there is a computerized data processing installation with a fast access memory of unspecified but sufficient size.
- 2) At various remote locations, (data process centers of adjacent and subordinate units) there are lower level data processing facilities, with associated stores of pertinent data, used by each staff or line commander in performing his duties.
- 3) Between the central and remote locations, there exist secure digital data communications of sufficient capacity to forward tell to the center any required individual unit data.
- 4) The Command Node does not sample the environment directly with sensor systems of its own; it depends completely upon the facilities of subordinate units as data sources. Some of these subordinate units may be sensor units.

In an ACDS Command Node as above, the overall data base consists of two general types of data; Type I - Data existing at the subordinate and adjacent unit level, some varying portion of which is forwardtold or lateraltold to the Command Node, and Type II - Command Node Proprietary data (such as summaries or intelligence reports) that exist at now lower or adjacent level of command.

Of these two types of data, Type I is almost certainly the larger in quantity by far. It is the only type having any important effect on both required communication line capacity between the center and subordinate units, and required central memory size. The amount of the second type of data affects central memory size, and is not considered further in discussing this model. After the reader understands the concept of the two models to be described, he may use them to analyze the traffic between the central computer and its own internal data base - that is - the Type II data mentioned here. This requires some new assumptions but the use and update processes can still be modeled in the manner to be described. From the nature of Type I data base, it is apparent that estimates must be made of several of its properties. The most important of these are:

- 1) The absolute size of the data base (how many characters in all).
- 2) The distribution of frequency of changes and individual item values.
- 3) The distribution of frequency of item usage by the processing system.

Precise estimates of the size of the Type I or Type II data base are difficult to make for a system like ACDS which is essentially unlike any that has existed previously. However, a study of the missions and tasks of the various subordinate units, together with consideration of the probable needs of the Command Post Center for its mission, can lead to a reasonable estimate of the total size of the Type I and Type II data base. However, the estimate of the size of the data base for the Strike Command Post is not of direct importance to this discussion since the model itself is independent of the size of the data base. The model discussed is constructed so that it describes the characteristics of update and access of any large data base of an arbitrary size of "N" characters. Future references to the model are in terms of "N."

The rate of which data changes value is of as much importance as the size of the data base. It is obvious that some items change slowly while others change much more rapidly.

Still others have intermediate average rates of change. The precise latitude and longitude location of a submarine pen is an example of a very slowly changing item. The position of an ATDS aircraft is an example of a rapidly changing item. And the relative positions of ships in a Task Force are examples of items changing at intermediate rates. The total amount of data is large and it is necessary to arrive at some combined distribution function of average frequencies of change from which the overall rate of naturally occurring item changes can be computed and related to data base size.

In this type of an analysis the estimate of the relative frequency of usage of items in the data base is very important. At one extreme is the hypothetical and improbable case in which every item is used in processing as often, on the average, as every other item. At the other extreme is the equally hypothetical (and even more improbable) case in which one item is used constantly, and none of the others is ever used. In the first extreme, it is plausible from an operational viewpoint (though perhaps not from a cost point of view) to consider maintaining the entire external data base in central memory, updating items as change messages come in from subordinate and adjacent units.

In the second extreme, it makes sense to maintain only one item in central data storage. The true relative usage frequency lies somewhere between the hypothetical extremes, and the usage function has an important effect upon the optimum size of central memory. It remains to examine the relationships among data change rate, data usage rate, and data base size.

A system of the type shown in Figure 3-13 can be configured in several ways. It is conceivable at one extreme to have no central data storage whatever, relying upon specific requests to staff, subordinate, and adjacent units for individual data as required for central processing. It is also conceivable to duplicate in central stores, everything that subordinate and adjacent units have, relying for data base maintenance upon update messages received from these units as the values of individual items change. Obviously, the data traffic is very different in these two extremes. An intermediate type of system maintains, in central stores, some fraction of the units data bases,

updated as changes occur during operations, and to request additional specific items as the values are needed. The models developed here allow the designer and analyst to estimate the data traffic required to use and update the data base as the data base varies in size and in degree of distribution.

3.9.3 Data Base Models for ACDS

The Change Rate Model

We can select a reasonable functional model in the following way:

Let f_D = the fraction of the total external data base (N characters) that includes items changing at, or greater than, some average rate r_C .

r_C = the average change rate for the most slowly changing item in f_D , in characters/hour.

We must now select reasonable specific values for pairs of the parameters f_D and r_C , and connect the parameters functionally with an equation having constants determined by the chosen specific values. For our purposes, we may make the reasonable suppositions that:*

- 1) No item changes at an average rate greater than once every 10 seconds, or 360 times per hour.
- 2) That 0.1% of all the data base items change at a rate equal to or greater than ten times per hour.
- 3) That 100% of all the data base items change at a rate equal to or greater than 0.00001 times per hour (about once in eleven years).

* If, in planning a particular system, the planner feels that these three assumptions are not correct, he should feel free to change them. The technique, however, remains valid.

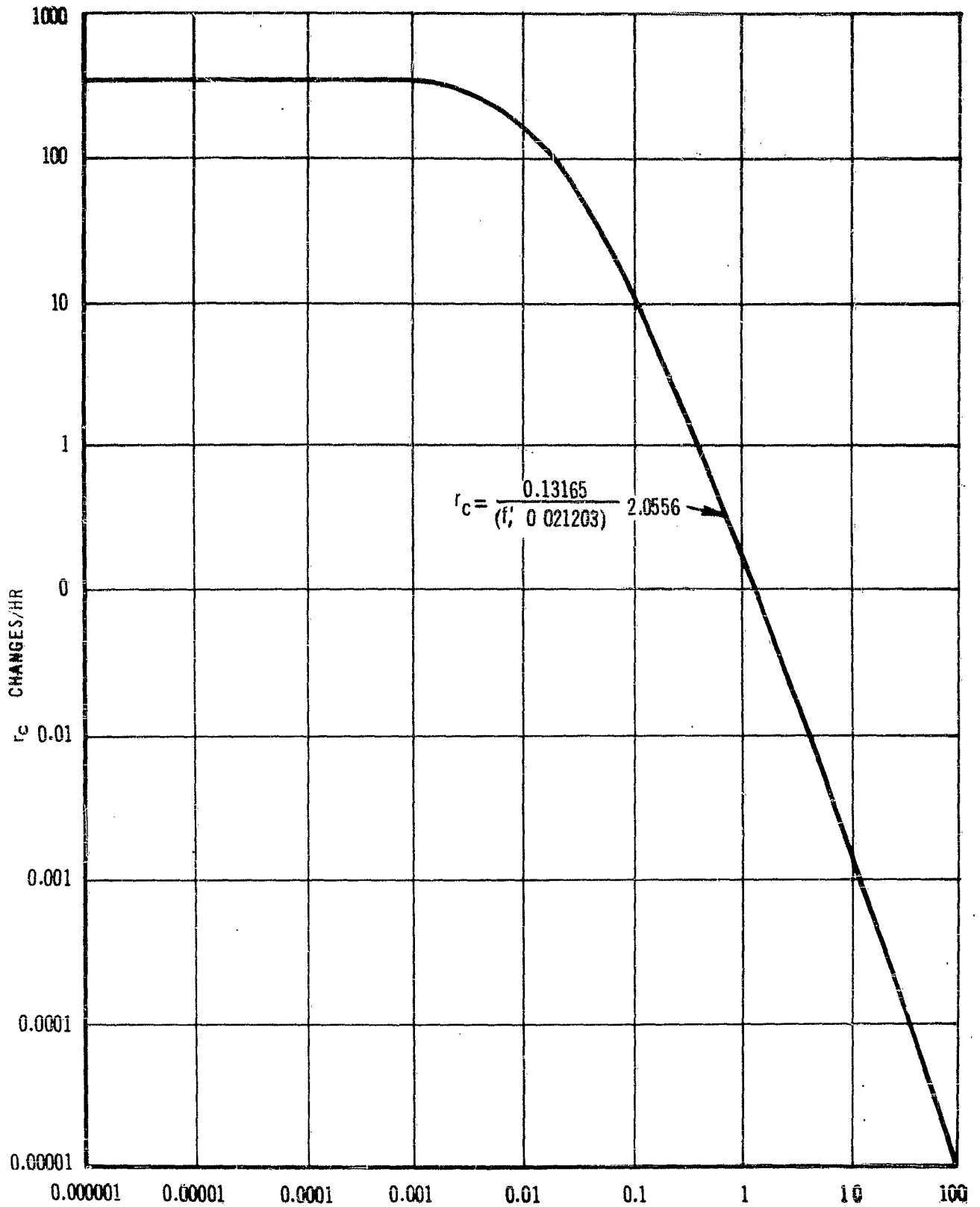


Figure 3-14 Effects Of Change Of Data Base Items

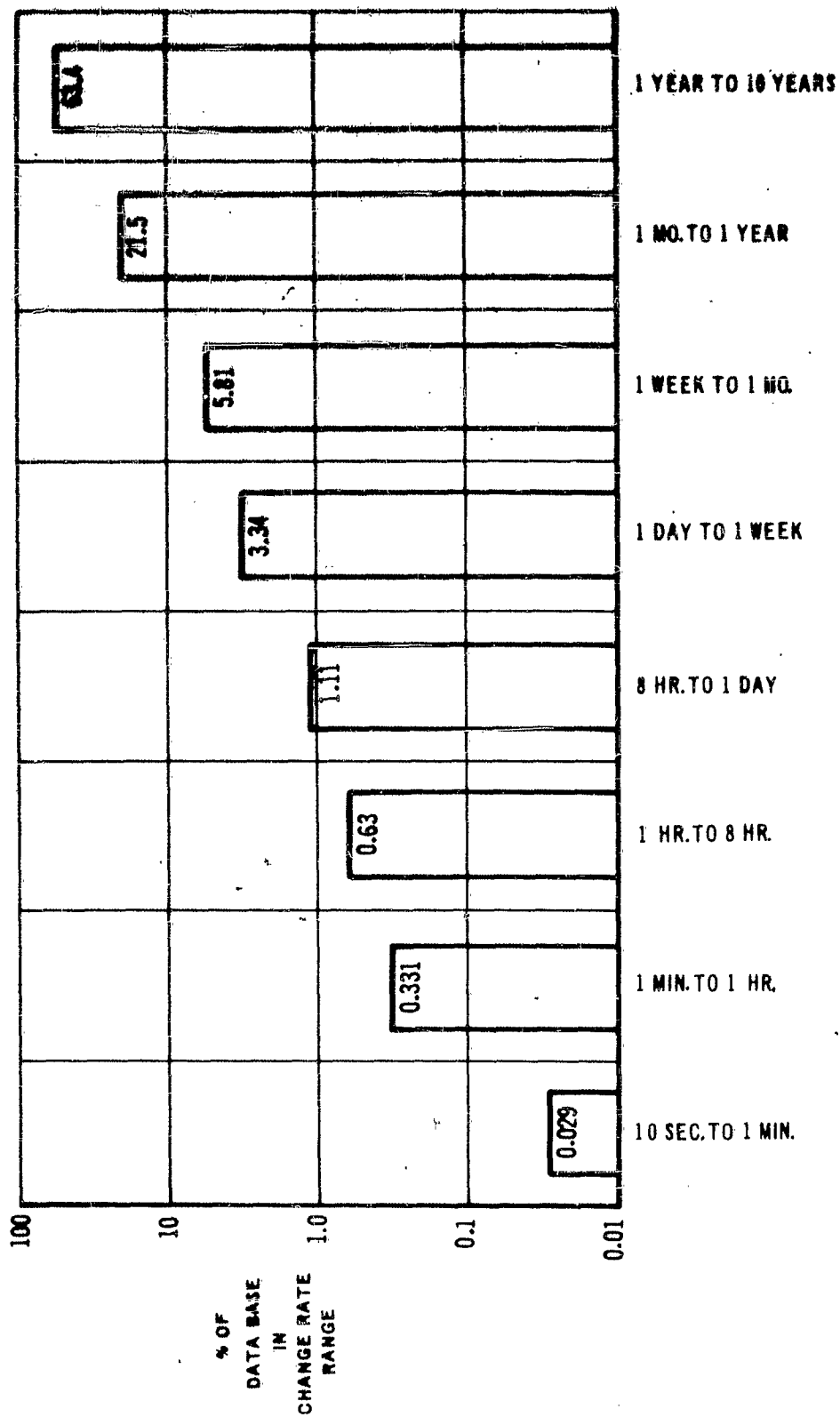


Figure 3-15 Model For Distribution Of Data Base Change Rates

These three conditions are sufficient to establish the constants in a relationship of the form

$$r_C = k_1 (f_D + k_2)^{k_3}$$

Substituting the proper constants gives (see Figure 3-14),

$$r_C = 0.13065 (f_D + 0.021203)^{-2.0556}$$

which yields, when integrated, an overall average change rate of 0.0822N characters per hour. Figure 3-15 shows the distribution of change rates in non-cumulative form, for further clarity. Thus, if the suppositions made in arriving at the constants are valid, and if we were to choose a configuration of ACDS Command Post in which the entire data base was maintained centrally, incoming data traffic would consist wholly of change updating messages that would add up to about 8% as many characters per hour as there are in the entire data base.

It is important, however, to have some idea of how much the occurrence of changes of data (and hence data traffic) could be expected to fluctuate about this average value. In all discussion up to this point, the overall rate of change of items in the data base has been treated as though it were constant in time, though the qualifying adjective "average" was used. This is not precisely true. The instantaneous rate of change can be expected to undergo excursions about the long-time average, since the occurrence of changes can be considered a random Poisson process, with specific probabilities attached to the occurrence of 1, 2, 3 ..., n changes in any given time interval.

If, for the purpose of discussing the model, we estimate the size of the data base, which is subject to change to be very large, that is $\approx 5.5 \times 10^8$ characters, the Poisson distribution resulting has an expected value (average) of 0.0822N, or 45.2×10^6 character changes per hour, which is very far from zero. The distribution is, therefore, only negligibly skewed and can be treated as very nearly Gaussian, but it still has the standard deviation of a Poisson distribution, which is given by the square root of the expected value or

$$\sigma = \sqrt{E}$$

For emphasis, we can switch from the hourly rate to the expected number of changes per minute, which will be:

$$\frac{45.2 \times 10^6}{60} = 0.754 \times 10^6 \text{ characters/minute}$$

Therefore, on this basis,

$$\sigma = \sqrt{0.754 \times 10^6} = 868 \text{ characters/minute}$$

Five times this value is

$$5 \sigma = 4340 \text{ characters/minute}$$

If we take a 5σ excursion limit on the rate of change in characters/minute, we are then in a position to say with 99.99+% certainty that the change rate will fall outside the range $(0.754 \pm 0.00434) \times 10^6$ characters/minute of only one minute out of 3-1/3 years on the average. It is quite justifiable, then, to treat the overall change rate as though it were a true constant, since the data base size cannot be known accurately enough to make this exceptionally high level of confidence a limitation! For much smaller data bases the update traffic will be correspondingly smaller, and the 5σ confidence level for excursions of the instantaneous change rate about the average will have to be re-computed using the method shown.

3.9.4 The Usage Rate Model

When we quantify usage rate functions, things become less certain, since there is no a priori experience to guide us in deciding what percentage of the data base items furnish what percentage of processing usage in computations. However, it is certain, as discussed above, that the proper function lies somewhere between equal usage of every item and exclusive usage of one item. If we now conceive of the entire collection of external data base items as being rank-ordered in terms of decreasing frequency of usage, and

let F_D = some fraction of the N items beginning with the most frequently used, and

f_u = the corresponding fraction of usage supplied by the less frequently used $(1 - F_D) \times N$ items

then the "equal usage" case is expressed by the linear relationship

$$f_U + F_D = 1$$

and cases of decreasingly less uniform usage are expressed by hyperbolas, of increasing inflection, of the form

$$(f_U - a) (F_D + a) = c$$

Several examples are shown in Figure 3-16. But the question of deciding on a proper representative choice remains.

Having estimated the types of relationships, we must decide what constants to place in the equations. There are no openly available studies regarding the usage of items in a data base. However, in an analogous situation, one concerning the relative usage of parts in a large inventory, it has been found that a particular distribution of parts value versus parts percentage of total stock, holds surprisingly constant regardless of the product.*

This concept of distribution of popularity has been found to represent accurately many sorts of popularity such as groceries in stock, parts in inventory, finished items in inventory, etc.

It appears reasonable to assume that a similar relationship holds true for data base item usage frequency versus item fraction. For purposes of this analysis, such a selection was made, and the corresponding curve appears as one of these in Figure 3-16. In this case, the appropriate constants assume the values $a = 0.02499$, $c = 0.0256$.

3.9.5 System Data Transfer

Having selected models for both change rate distribution and usage rate distribution, it is possible to combine the two and examine the effects on the rate of digital data transfer from subordinate units to the Command Post in the system as the fraction of the

* Dickie, H. F., ABC Inventory Analysis Shoots for Dollars, Not Pennies, Factory Management and Maintenance, Vol. 109, No. 7, pp 92-94, 1951

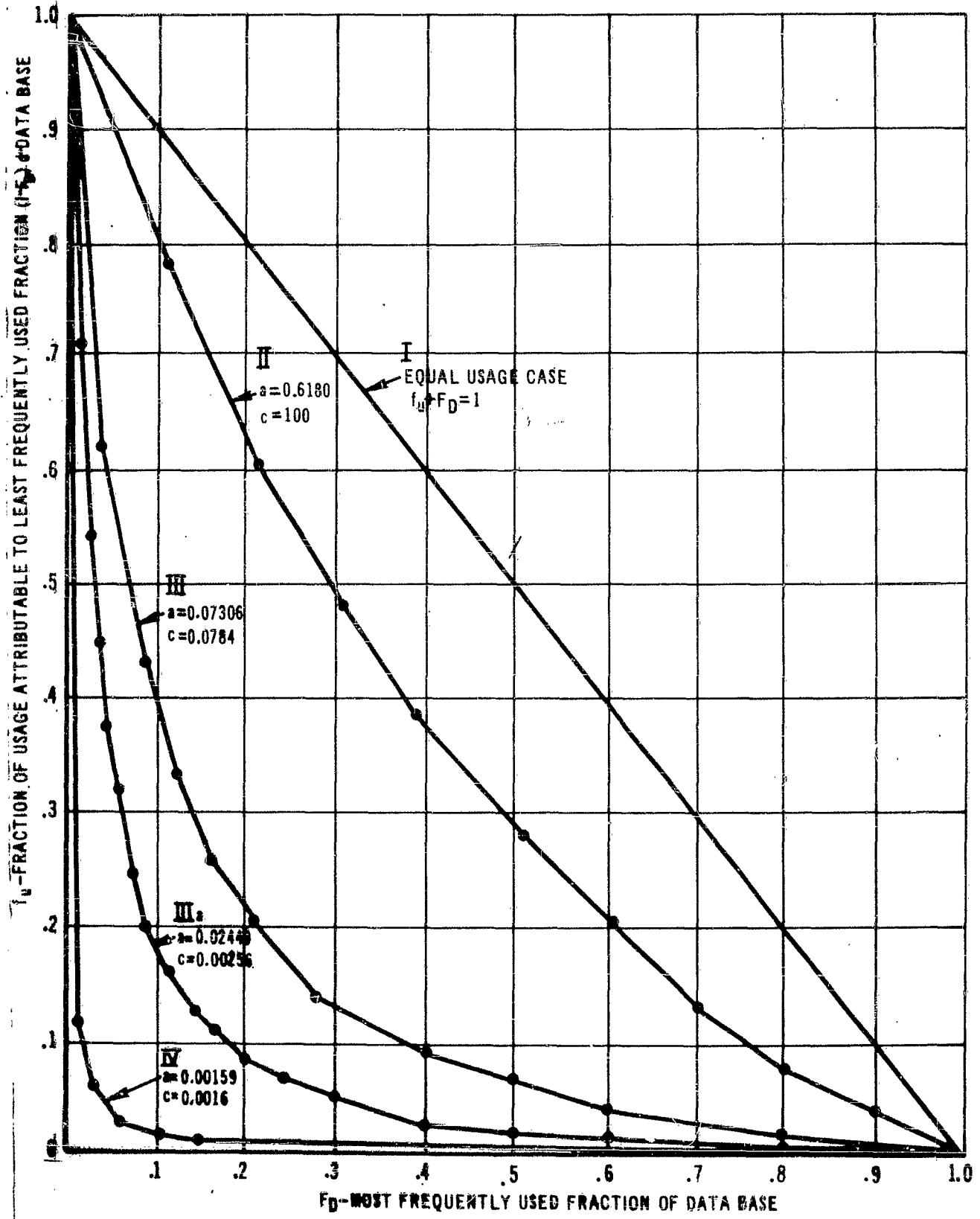
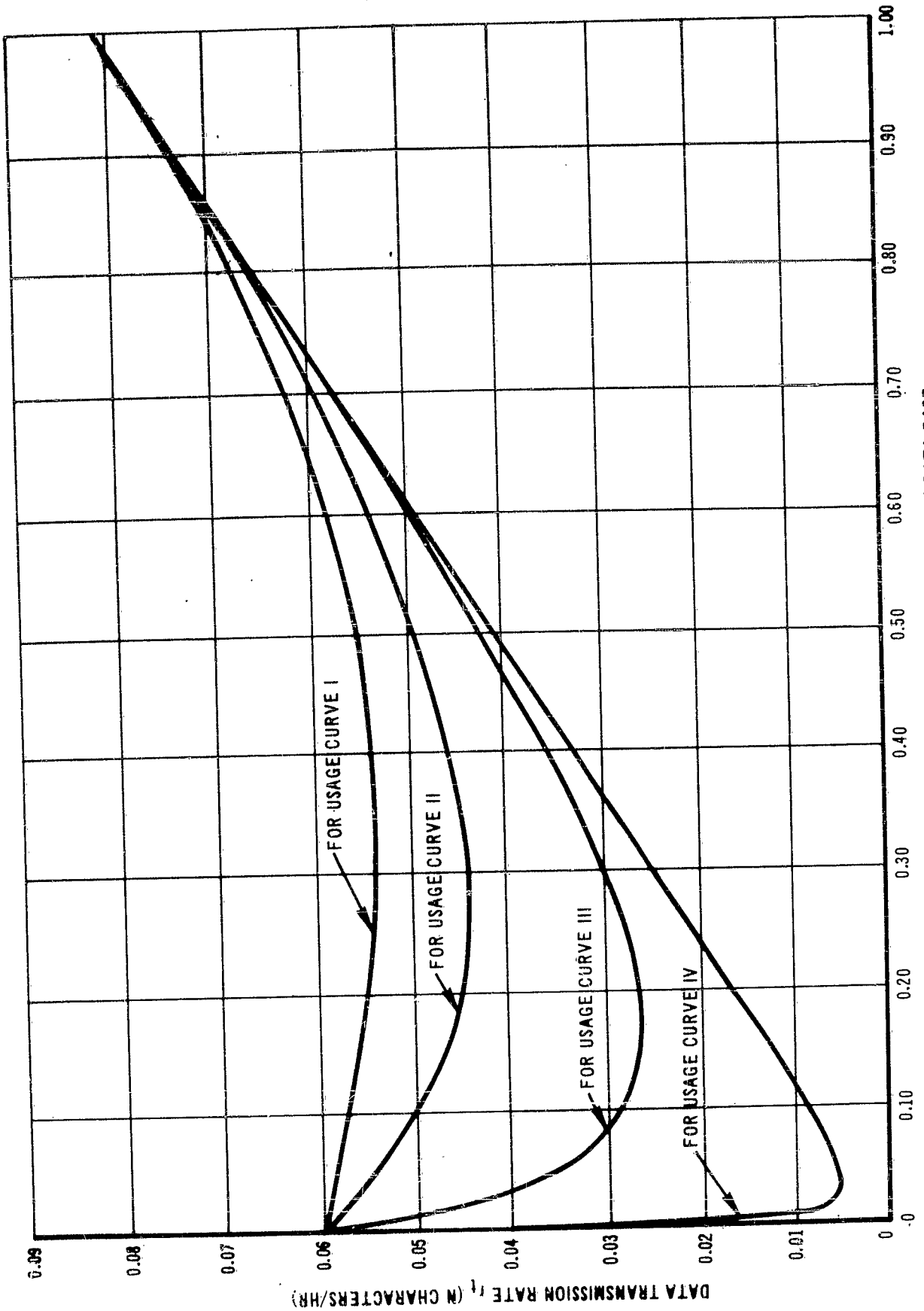


Figure 3-16. Model for Data Base Usage in Central Processing
 Non-Equal Usage Bases are Expressed by $(F_U + a)(F_D + a) = c$



F_D - MOST FREQUENTLY USED FRACTION OF DATA BASE

Figure 3-17. Data Transmission Rate as a Function of Fraction of Data Base Centrally Maintained ($K = 0.0822N$, $R = 0.73$)

data base maintained in the Command Post is varied, and as the ratio of overall data usage rate to overall data change rate is modified.

We first conceive of the entire system in Figure 3-13 as having a flexible quantity of central memory storage in the computing system at the Command Post, such that any desired fraction F_D (the most frequently used items) of the Command Post's data base can be maintained there. The data maintained centrally are updated by change messages from adjacent and subordinate units as, and only as, their values change. Data not maintained in the computing system at the Command Post are transmitted to the center only as needed and requested. Thus, for any value of F_D greater than zero and less than 1, data traffic will consist of two kinds: update messages (type A), and request-answering messages (type B). If $F_D = 0$, all traffic will be of type A; and if $F_D = 1$, traffic will depend only on the overall data change rate and will be exclusively of type B. But if F_D has an intermediate value, traffic will be of both types, and its volume will depend both on the overall data change rate and on data usage rate at the Command Post.

3.9.6 System Data Transfer Analysis

The analytical problem here involved is to determine quantitatively, the relationship between incoming data traffic and the size of the fraction of the external data base that is maintained centrally, for various rates of central data usage. Obviously, if there is an optimal value for the fraction to be stored centrally (F_D) that minimizes hardware costs by balancing the cost of transmission facilities against the cost of central memory, it is desirable to find it. The present analysis demonstrates a way of doing so.

In addition, it may be that the data base within the computing system at the Command Post is to be held in several types of storage media such as tape, drum, disc or high-speed memory. These models provide tools for analysis of the use and update data traffic required within the Command Post computing system itself.

If we let:

r_t = incoming data traffic rate, in characters/hour

R = the ratio of overall usage rate (characters/hour)
 overall change rate (characters/hour)

$$K = \int_0^1 r_C df_D = \text{overall change rate (characters/hour)}$$

and further make the assumption that there is zero correlation between the usage rates and change rates of individual items in the data base, then,

$$r_t = K F_D + R K f_u (1 - F_D)$$

where the first term represents update messages for centrally maintained items and the second term represents question-answering messages for items not centrally maintained. Using the models described above, the appearance of this curve is as in Figure 3-17. But the relationships between f_u and F_D has been established as

$$(f_u + a)(F_D + a) = c$$

or

$$f_u = \frac{c}{F_D + a} - a$$

therefore

$$r_t = K \left\{ F_D + R (1 - F_D) \left[\frac{c}{F_D + a} - a \right] \right\}$$

which can be differentiated with respect to F to find what value of F_D (for any R) will give a minimum value of r_t . If we take $\frac{dr_t}{dF_D}$ and set it equal to zero, we find that the condition for a minimum r_t is given by

$$F_D = \sqrt{\frac{c(a+1)}{a + \frac{1}{R}}} - a$$

from which it can readily be seen that the centrally maintained fraction (F_D), of the external data base that gives minimum data transmission rate (r_t), is a function of the properties of the usage distribution of the data base and the ratio (R) of overall usage rate to overall change rate.

At this point in the analysis it is desirable to see how sensitive the location of the minimum is to these parameters. In Figure 3-18 optimum values of the centrally maintained fraction (F_D) are plotted against the ratio (R) for the range of usage curves in Figure 3-17. It is easily seen that optimum F_D is very sensitive to changes in R for usage curves at the "equal-usage" end of the range, becoming less so as the usage curves become more highly inflected. In particular, a curve selected as very likely to be valid for this type of analysis, gives a change in optimum centrally-maintained fraction (F_D) only from approximately 0.09 to 0.39 for a change in the ratio R from 0.5 to 8.0. There is, therefore, reason to expect that a design choice of F_D with corresponding choice of central memory size, at some appropriate point in this range, allows operation at off-design values of the ratio R without resulting in too great a change in data transmission rate (r_t). Naturally, since the usage rate varies unpredictably from low values in periods of calm to high values during emergency periods or exercises, such a state of affairs is very much to be desired.

It is, therefore, of considerable interest to extend the analysis further and discover just how much the data transmission rate r_t changes for a given change in the ratio R , given particular design values of R and F_D . Rewriting the expression for r_t in dimension form, we have

$$\frac{r_t}{K} = F_D + R(1 - F_D) \left[\frac{c}{F_D + a} - a \right]$$

Incrementing both R and $\frac{r_t}{K}$,

we have

$$\frac{r_t}{K} + \Delta \left(\frac{r_t}{K} \right) = F_D + (R + \Delta R)(1 - F_D) \left[\frac{c}{F_D + a} - a \right]$$

and by rearrangement we arrive at

$$\frac{\Delta \left(\frac{r_t}{K} \right)}{\frac{r_t}{K}} = \frac{\frac{\Delta R}{R}}{1 + \frac{F_D}{F(1 - F_D) \left[\frac{c}{F_D + a} - a \right]}}$$

Using this expression, given design values of the ratio R and the centrally-maintained fraction F_D , it is possible to calculate the proportional change in $\frac{r_t}{K}$ resulting from any hypothetical change in R. As an illustration, suppose we select the design point given by $F_D = 0.198$ and $R = 2.0$, using the selected usage curve.

In words, this corresponds to design which is optimized for maintaining very nearly 20% of the external data base in central stores, and for using data from the external base in central processing at a rate of twice as many characters per hour as there are natural data character changes per hour. The relation then reduces to

$$\frac{\Delta \left(\frac{r_t}{K} \right)}{\frac{r_t}{K}} = 0.423 \frac{\Delta R}{R}$$

Which says that, in this case, a 100% increase in the operating value of the ratio R at any time would result in only a 42.3% increase in $\frac{r_t}{K}$.

Since $\frac{r_t}{K}$, when operating at the design point is only 0.342, the off-design value of $\frac{r_t}{K}$ would only rise to 0.478. Hence, data traffic is still less than half the overall change rate K , even though data usage has doubled over the design value. Other examples can be similarly calculated.

3.9.7 Data Transmission Requirements

Based on the results of the analysis outlined above, a number of possible design points have been selected as examples, according to the following rationale.

We cannot know what usage rate to assume as being most probable in any real system of the kind discussed, though we know that it will vary as the real-world situation changes from calm to emergency. However, a ratio R value of 0.5 or 22.6×10^6 character/hour seems intuitively reasonable under normal circumstances, that is, a usage to update ratio of 1:2.

We, therefore, pick $R = 0.5$ as one ratio to design an optimum system around. For this value, a centrally-maintained fraction of $F_D = 0.0885$ is optimum, resulting in $\frac{r_t}{K} = 0.180$, which represents a transmission rate of 0.0148 N characters/hour for a data base of 5.5×10^8 characters.

The value of R might be larger. Suppose it is four times larger, giving $R = 2.0$. This could be handled two ways. The previous system optimized around $R = 0.5$ could be allowed to operate off-design, or another system optimized around $R = 2.0$ could be used. In the first case, $\frac{r_t}{K}$ rises to 0.454, while in the second F_D is increased to 0.198, and the resulting $\frac{r_t}{K}$ is 0.342 (thereby decreasing data traffic from what it would be in the first method, at the expense of increased central storage capacity in the second method).

The value of R might be larger yet. If we increase it by a factor of four again, to $R = 8.0$, we can extend the process described immediately above and see what results in a system optimized for $R = 0.5$ but operated at $R = 8.0$, a system optimized for $R = 2.0$ but operated at $R = 8.0$, and a system optimized for $R = 8.0$. In the first

two cases, $\frac{r_t}{K}$ rises to 1.552 and 0.777, respectively; while in the last the centrally-maintained fraction F_D becomes 0.393 with an associated $\frac{r_t}{K}$ of 0.569.

Finally, for comparison and completeness, we add systems designed for $F_D = 1.0$ (in which case everything is centrally maintained, the ratio R has no effect, and $\frac{r_t}{K} = 1.0$), and for $F_D = 0$ (no centrally-maintained data) with $R = 2.0$ and $R = 8.0$.

Thus, there are, in all, nine potential schemes selected to serve as examples of how the models are to be used. For greater ease in reference, the pertinent data described above are also tabulated in Table 3-1.

3.9.8 Summary

To arrive at useful conclusions concerning the possible configuration of an ACDS node it is first necessary to make some assumptions about the logical structure of the system and about the nature of the data base to be contained within it. These assumptions have been made for a design which utilizes a large data base and the usage/update model has been applied to that type of analysis.

The models are presented in such a way that the analyst may use any constants he believes necessary to model his system.

The generality of the model and its concept make it applicable to the analysis of the many configurations of data base usage and update problems in ACDS planning, including those entirely contained within a single computational node.

Table 3-1 Selected Possible Design Points

Design R	<u>A</u> 0.5	<u>C</u> 0.5	<u>G</u> 0.5
Operating R	0.5	2.0	8.0
Design F_D	0.0885	0.0885	0.0885
Operating r_f	.0148N	.0373N	.1277N
Design R		<u>D</u> 2.0	<u>H</u> 2.0
Operating R		2.0	8.0
Design F_D		0.198	0.198
Operating r_f		.0281N	.0639N
Design R			<u>I</u> 8.0
Operating R			8.0
Design F_D			0.393
Operating r_f			.0467N
Design R	<u>✓ B</u> 2.0	<u>F</u> 3.0	<u>E</u> --
Operating R	2.0	8.0	--
Design F_D	0	0	--
Operating r_f	.1643N	.6575N	.0825N

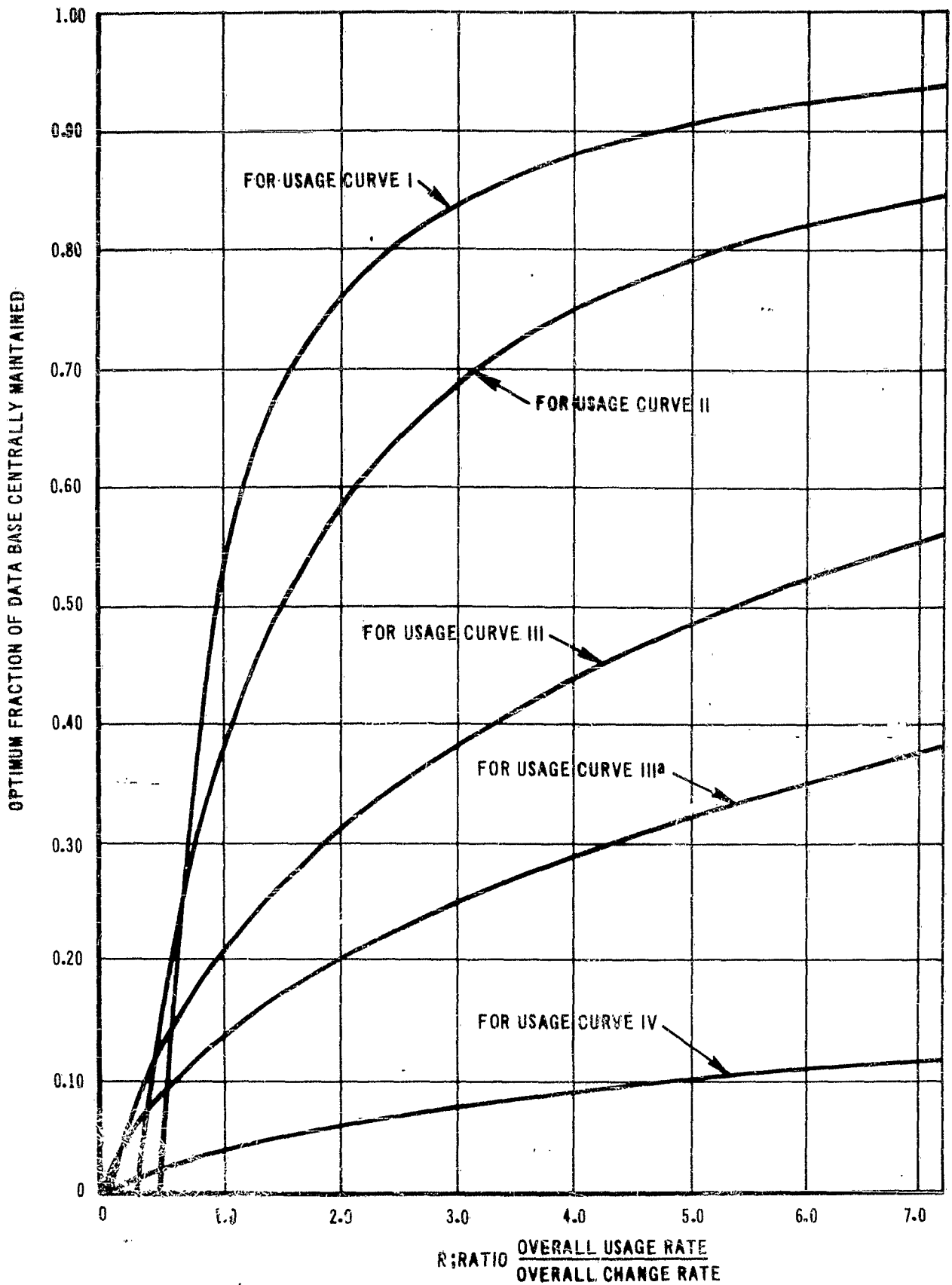


Figure 3-18. Optimum Values of the Centrally Maintained Fraction as a Function of the Ratio of Overall Usage Rate to Change Rate

3.10 FIGURES OF MERIT FOR DIGITAL COMPUTERS

3.10.1 Introduction

This section discusses several approaches to determining arbitrary numerical measures for comparing the "computing capability" of electronic digital computers. Comparisons of various digital computers are normally required several times during the planning of any command data system node. The figure of merit technique is an attempt to simplify and regularize the consideration of many important computer comparison factors. The measures discussed here, and others like them, consider only the "main frame" and high speed memory capability of the computer being examined. That is, they consider only the size of high-speed memory, the speed with which data is transferred into the computer from memory, and the speed of computation.

Since one of the crucial limitations of modern data processing equipment is often input-output capability, these figures of merit approaches clearly leave much to be desired. However, we must bear in mind that normally the purpose of computer installations is not to perform input-output (I/O) functions but to manipulate data. Regardless of input-output limitations, this work is done by the central computer, and figures of merit have real value in the comparison of central computer capability without regard to type of computer or the application for which the computer is used.

To complete any worthwhile analysis, considerations such as instruction repertoire, I/O capability, amount and type of low speed storage, mean time between failures, mean time to repair, etc. must be studied carefully. Nevertheless, figures of merit offer substantial advantage to the system analyst who understands their rationale and limitations, and who confines their use to "rough-cut" first approximations*.

3.10.2 The Bench Mark and the Figure of Merit

There are two distinct general approaches to measuring the capabilities of computing machinery. Only one of these (the figure of merit) is analyzed in this report. But to understand this one technique fully it is first necessary to understand the other (the "bench mark" technique) to a limited degree, and to compare them briefly.

* Rector, R.W. Measuring the Capability of Computing Equipment. Private Communication - unpublished.

1) The Bench Mark Technique

This approach to measuring computer capability is problem oriented. That is, machines are evaluated on their ability to perform certain problems or selected parts of the total task proposed. These problems may be entire real problems, parts of real problems, or synthetic problems made to resemble real problems closely. This technique is called the "bench mark" method since it compares machines by examining their differential capability (normally speed) to perform the same "bench mark" problem.

The bench mark technique (if carefully executed) can be quite accurate, but it is very costly in talent and time, and requires an accurate and precise definition of the total task to be performed. In addition, any bench mark problem which is not the complete task ultimately to be demanded of the computer, takes on certain aspects of simulation and is subject to many of the limitations of simulation.

2) The Figure of Merit

This approach attempts to evaluate the capability of an individual machine without regard to how that capability is used. This is much the same thing as a power station being given a kilowatt rating without regard to how much electricity is used or how it is used. At first, this may seem a little foolish since the only reasonable purpose of computers is to solve real problems. However, system planners find it very useful to be able to think of and measure main frame and memory capability in the abstract. Figures of merit permit them to do this.

3.10.3 The Figure of Merit Rationale

Figures of merit may be used to provide preliminary answers to a number of problems without the need to prepare a bench mark analysis. Among these problems are questions such as:

- 1) I am now processing data at rate R . My work load will increase to about $7R$. What various machines should I consider acquiring?
- 2) My old machine needs to be replaced. What will I have to pay for a new machine, and how much capability could I have left for expansion? This is really a new statement of question #1.
- 3) Company A charges \$5,000 per month for machine 1. Company B charges \$7,500 for machine 2. Is the difference worthwhile in terms of data processing?
- 4) The new system I am planning should have the computing load of about half that of System X, which uses a CDC 6600 at about full capacity. I plan to split the computing load among four computers, A, B, C, and D, where $B = \frac{A}{2}$ and $C = \frac{A}{3}$. Allowing for 20% expansion, what machines should I think of for my system?

These and other similar questions of a preliminary planning and design nature can be answered by using some figure of merit technique.

The entire figure of merit approach is based upon the premise that "more" is "better." The question "Is 10% more also 10% better?" is discussed later. The more fundamental question "More what?" is answered (depending upon what figure of merit is considered) by "more internal speed," "more high speed memory" or some combination of both. How these qualities are combined differs from case to case and is discussed by individual case.

In general, it can be said that more speed is better in direct proportion to the increase. That is, a four-fold increase in speed is four times "better," and a six-fold increase is six times "better." Another way of looking at this is; a machine which can do work in four hours that was previously done in eight is twice as beneficial to the user. This is particularly true of machines used "on-line."

From the standpoint of the usefulness of high speed memory to a user, more is better, but probably not in direct proportion to the increase. That is, to go from a size of 500,000 bits to 1,000,000 bits is more beneficial to the user than to go from 1,000,000 bits to 2,000,000 bits — even though the increase is by the same factor.

There is, however, some difference in opinion as to how much the worth of memory changes as size of memory grows larger. The manner in which the incremental utility of larger memories decreases is generally felt to be logarithmic (or some function so close to logarithmic that the difference is not worth worrying about.) Remember, the search is for some numerical way to express professional opinion, so accuracy is greatly to be preferred to precision. Accuracy is faithfulness of conceptual replication, while precision refers to the degree of refinement of the measurement. It is easy to have one without the other; but precision without accuracy is misleading, at best, while accuracy without precision is often very useful.

For some applications, perhaps one such as message switching, memory requirements may be thought of as absolute. That is, the high-speed memory must be big enough to do the job — but size increments beyond that point are of little use. For these applications, and those where time constraints are severe, more attention should be paid to the efficiency of the computation process than is normally done.

A technical discussion of several types of figures of merit, their applications and shortcomings is now appropriate.

3.10.4 The "Classic Method"

Rector* has applied the name to this method, and while it may not be "classic" in the most pristine sense of the word, the method has been applied in much of the literature. The calculation is a simple one:

$$\text{Classic Figure of Merit (CFM)} = \log_{10} \frac{M}{T}$$

Where M = high speed memory capacity in bits
and T = access time in seconds

Various forms of memory arrangement must be converted to give a total reading in bits. Sign bits and parity bits should not be included.

Access time is the time required to fetch a word (or character or set of characters) from memory. In destructive-readout memory machines the data cannot be operated upon until that small portion of memory is restored with the data just read out

* Ibid

destructively. This takes one more memory access time. The two times together are called a memory cycle. Most data are given in cycle time and must be divided by two. However, in non-destructive memory machines, operations begin immediately after access time.

Since most published tabular data presents the time in microseconds (The Adams Chart, for instance), it is most convenient to use, and subsequent calculations in this paper use microseconds.

The classic method allows calculation of the CFM for many storage and access devices, not just computers alone. Some values calculated in this manner are shown in Table 3-2.

Several points must be completely understood by the system planner contemplating the use of measures such as this one. These are:

- 1) The logarithmic nature of the CFM number.
- 2) The equal treatment of memory and speed increases.
- 3) The implicit relationship of computation speed and access time.

The CFM is, by definition, the logarithm of a decimal number. Its being logarithmic has several implications for a user.

The human mind apparently thinks in linear terms as a normal course of events. Even when presented with a table and the certain knowledge that the CFM is a logarithm, it somehow seems more real to think of terms varying from 100,000 to 45,000,000 than from 4.9 to 7.6. Our world of experience is linear, and dealing with logarithms can be quite illusory.

Therefore, on Table 3-2, where the 910 is 4.9+ and the 6600 is 7.6+, this would mean to many people that two 910's are a little better than one 6600. Of course this is not true, and the error comes from treating logarithms as decimal numbers. In reality, the table states that the capability of the 6600 is three decimal places greater than the capability of the 910; namely, the 6600 is between 100 and 1,000 times as powerful as a 910.

Table 3-2. Classic Figure of Merit

	Max. Wds.	Bits/Wd	Total Bits	Storage Cycle Time (in musecs)	$\frac{\text{Cycle Time}}{2}$	$\frac{\text{Bits}}{\text{Access}}$	Log_{10}	Bits Access
CDC 6600	262K	60	15,720,000	0.7	0.35	44,910,000		7.6523
IBM 7030	262K	64	16,768,000	2.2	1.10	15,240,000		7.1829
Hughes H-330	181K	48	6,288,000	1.8	0.90	6,969,000		6.8432
Philco 212	65K	48	3,120,000	1.8	0.75	4,160,000		6.6191
RCA 601	32K	56	1,792,000	1.5	0.75	2,389,000		6.3783
Univac 1107	65K	36	3,340,000	4.0	2.00	1,170,000		6.0682
SDS 9300	32K	24	768,000	1.75	0.87	882,800		5.9459
CDC G-20	32K	32	1,024,000	6	3.00	341,300		5.5332
CDC 160A	32K	12	384,000	6.4	3.20	120,000		5.0792
SDS 910	16K	24	384,000	8	4.00	96,000		4.9823

Basic Data From Adams (Nov., 1963)

if true, this is useful information, but it cannot be said that it is intuitively obvious. We find, then, that direct comparisons between the very high and very low ratings on the scale may be open to some question. It is also open to question as to how meaningful this 1,000 to 1 ratio could be even if it were quite accurate.

The illusory nature of logarithms and the abnormal compression of the scale should be looked at again. This time look at three computers clumped at the center:

Hughes 330	CFM = 6.8432
RCA 601	CFM = 6.3783
Univac 1107	CFM = 6.0682

These machines appear to be very close together in capability, particularly since they have the same first digit in their CFM. One might imagine that they are indistinguishably close. By reference to Column A it is seen that the quotients prior to the taking of the logarithm lie in the relationship of 6.9:2.4:1.2. This is a considerable difference, indeed, and it is in adjacent areas of this long table that comparisons of CFM's have a great deal of usefulness and reasonable credibility.

There are three fundamentals of logarithmic tables which must be thoroughly understood by any system planner who uses the CFM technique.

- 1) Logarithmic representations are used to place extremely large numbers and very small ones in the same table conveniently, and to allow these numbers to be manipulated pleasantly.
- 2) The use of logarithms obscures the true linear relationships of many types of data, and can stimulate logical errors by all but the most cautious users.
- 3) Arithmetic operations must be performed upon the antilog of the CFM not the CFM itself, that is, the quotient before the \log_{10} is obtained.

The data in Table 3-2 is used to solve problem 4 in Section 3.3.5.3. This crystallizes the points discussed so far.

The proposed system has a load of about one half of System X which uses a CDC 6600 to about full capacity. Allow for 20% expansion. Use four machines A, B, C, and D, with $B = \frac{A}{2}$ and $C = \frac{A}{3}$. Confine the problem to machines from the table.

$$\begin{aligned} \text{CDC 6600} & \qquad \qquad \text{CFM} = 7.6523 \\ \text{Antilog}_{10} & \qquad \qquad 7.6523 = 44,910,000 \\ \frac{44,910,000}{2} & \qquad = 22,455,000 \\ 120\% \times 22,455,000 & = 26,946,000 \end{aligned}$$

Split the load derived among four machines. The load must be allocated 6/13 to A, 3/13 to B, 2/13 to C and 2/13 to D.

$$\begin{aligned} \frac{26,946,000}{13} & = 2,072,769 \\ A = 6 \times 2,027,769 & = 12,166,614 \\ B = 3 \times 2,027,769 & = 6,083,307 \\ C = 2 \times 2,027,769 & = 4,055,538 \\ \log_{10} 12,166,614 & = 7.0853 = \text{CFM}_A \\ \log_{10} 6,083,307 & = 6.7841 = \text{CFM}_B \\ \log_{10} 4,055,538 & = 6.6580 = \text{CFM}_C \end{aligned}$$

A smaller than maximum size 7030 does well for machine A. An H-330 is close to exactly right for machine B, and the 212 could be used for machine C.

The outstanding shortcoming of the Classic Figure of Merit is that it treats increments in storage as being equally beneficial.

Stating the CFM equation again:

$$\text{CFM} = \log_{10} \frac{(\text{High speed storage in Bits})}{(\text{Access Time in Microsecs.})}$$

The logarithm₁₀ does not apply to either the numerator or the denominator, but to the quotient, and therefore treats increases in speed and increases in memory as equally beneficial. For speed this is desirable. For memory size this is not really acceptable.

The worth of machines is often estimated by specialists to look something like

$$\text{Merit} = \frac{\log_x (\text{high speed storage in bits})}{\text{access time in microseconds}}$$

This expression satisfies much of the discussion here and something like it is treated later.

In the classic figure of merit and in some others, the only computer speed considered is cycle or access time. In destructive readout machines, cycle time equals two access times. Most instructions also require integral numbers of access times for their execution. This is because internal speeds are governed by a clock (in synchronous machines), and hence by how fast that clock permits instructions to be executed.

Normally, the fastest tasks of logical testing or shifting control unconditionally occupy one access time, and more complex instructions more integral units of access time. Thus, a reasonable approximation of the internal processing speed may be had by looking at access time. However, for a really accurate estimate of the internal computational speed of any machine, reference must be made to instruction time. This is treated in a subsequent section of this report.

In asynchronous machines, front parts of each instruction may be thought of as overlapping with the final parts of preceding instructions, and therefore access time is not as reliable a measure of computation speed. Still, computation is wedded to the speed with which numbers can be shifted into and out of memory, and access time is a reasonable indicator of that speed.

When these techniques are used with non-destructive readout machines, extreme care must be taken to use access time for non-destructive machines and cycle time for destructive machines. This is because, in non-destructive machines, computation can begin as soon as the number is brought in, while in destructive machines one additional access time is required to restore the number to its original memory location.

In figure of merit computations, considerations other than those of the main frame, memory, and some approximation of computation speed are entirely ignored. The capabilities of input/output peripheral equipment for each system must be studied in detail according to the requirements of each system, and they are not amenable to approximation before the requirements of a system are reasonably well known. It must be remembered that some relatively slower machines have fine input/output and peripheral equipment and, thus, more than make up for their so-called "speed deficiencies".

3.10.5 Information Channel Capacity

Data processing machines that are used primarily for switching purposes, and have memories which meet the absolute minimum required by the problem, may be compared by the use of a slightly more involved technique which treats only the internal speed of the computer.*

$$\text{Channel Capacity or } C = \frac{L}{\frac{N}{P} + T} \cdot Q$$

Where L = word length in bits

N = number of bits required for the execution of an operation

P = clock rate in bits per second

T = average wait time

Q = number of simultaneous operations performed

* This technique was developed by Amelco, Inc. in a study performed for Douglas Aircraft as a part of the Army/Navy Instrumentation Program, Data Processing, ANIP Research, June 1961, Amelco, Inc.

This approach does yield a good measure for the internal effectiveness of a computer used solely as an information switch. Its shortcoming is primarily that, since the approach does not consider memory requirements as other than absolute, the approach has little general application.

This method also has the disadvantage of considering word length (longer = better) without considering memory size. The result of this is two-fold. First, machines with long words come out better than machines with short words - even if they have the same number of bits in memory, which is hardly reasonable. Second, it is quite possible for a machine with the longer word to be less efficient, (even given an equal-sized memory) than a short worded machine, for the following reasons.

Most command data system processing consists of setting and testing items (parts of words), not of making arithmetic computations using full words.* To do this, a word with many bits must be shifted or cycled a larger average number of bit positions than a word with fewer bits. This takes more time. There are machines having special logical circuitry which allows the testing and setting of a few bits without manipulating the entire word. In other than those machines, it is misleading to say "the longer the word, the better". Often this may be completely incorrect. This argument assumes the same number of bits in memory.

However, the reason for including this number (L) in the computation here is: the more bits in the word the more data can be transferred in parallel from memory, and this is an advantage - though somewhat diluted sometimes by an increase in shifting time.

* Picket, R.S., Investigation in Search of a Measure of Data Processing, Unpublished, April 1962.

Campbell, E.K., The Determination of the Meaningful N-Tuples of instructions in a Computer Program, TM-865, 30 Nov. 1962, The System Development Corp., Santa Monica, California.

Anon, Dynamic Instruction Count of a Real Time Program, IBM Federal Systems Division, Kingston, N.Y., 21 Oct. 1960.

As with other figures of merit, this one does not evaluate input/output or peripheral equipment. It is included here primarily to show a good method for evaluating internal timing.

3.10.6 Efficiency Index

The general concept of indices of efficiency is that they measure the ability of the device examined to produce output equal to the input provided.

When we compute the "efficiency index" of digital computers, dollar cost is used as input and the efficiency measure is supposed to show how much "computational ability" per dollar cost is delivered by various machines.

One of the many possible manners of computing an index such as this is shown below.

$$\text{Efficiency (E)} = \frac{n}{t \cdot C_a}$$

Where n = number of bits per word
 t = add time + 0.01 multiply time
 C_a = cost of arithmetic and control units *

This measure has several shortcomings. Nearly any measure using the same terms has the same disabilities, regardless of how the terms are accumulated arithmetically.

- 1) Using the word length alone in the numerator has the same weaknesses it had in channel capacity measurement.
- 2) Using cost in the computation of the index itself has three serious disadvantages
 - a) It is very difficult to obtain the bare cost of the arithmetic unit and of the control unit by themselves for a large array of computers. Granted that it can be done for any particular computer at will - it is still a formidable task for the more than 75 computers now available in the U.S. The GSA

* Anon, Mathematical Models for Information Systems Design and Calculus of Operations, Magnavox Research Laboratories, MRL Report #R-451, 27 Oct. 1961.

electronic supply catalog has the prices of the pieces, but customer engineers have to be questioned to make sure the correct set of prices is used to produce the total cost.

- b) The total cost of the various systems is not any constant function of the arithmetic and control unit. Some computers have low priced units, others high, and any system must all be bought and installed to obtain whatever efficiency is inherent in the two units discussed here. It is only the whole cost of the whole system that is of any importance to us.
 - c) Regardless of what cost is used, it is subject to considerable fluctuation, irrespective of what is published by GSA. This is true since costs are not physical constants of the machine itself, but are derived by management fiat. By using rather vague and fluctuating data in the computation, particularly in multiplication or division, the entire result is open to the most serious question. Of course, prices should be considered, but they should be considered separately from the physical constants of the machine itself.
- 3) The most serious consideration in this type of measurement is the use of

$$t = \text{add time} + 0.01 \text{ multiply time}$$

Naturally, internal computational speed should be considered in evaluating any computer. The classic figure of merit does this indirectly as stated earlier. The construction of the factor t implicitly states that the programs, yet to be designed and coded, call for two times access time instructions (like add) 100 times as often as they call for 8, 10, 12 or more times access time instructions (such as multiply and divide). The construction of " t " is not interpreted to mean that add and multiply themselves are most popularly used, or occur with this relative frequency, only that instructions requiring that number of access times occur with that frequency.

The consideration is this. The future use of the computer is being simulated or guessed at. If the guess is good, the answers are very good (barring other flaws in the computation of these indices). If the guess is not close to correct, the answer is terrible.

It is desirable, however, to get a better reading of internal computational speed than is done indirectly by the CFM, and this is a very reasonable way to do so. Analysts using this technique should be aware of its possible shortcomings. That there is some possibility of error should not prevent the consideration of the technique.

- 4) This figure of merit cannot evaluate the efficiency of the entire computational system since it cannot estimate the input/output and peripheral equipment accurately before the system is planned. This shortcoming is not peculiar to the efficiency index alone, but is shared by all figures of merit.

3.10.7 Babbages

C.J. Shaw* has developed, but not documented, a figure of merit which avoids many of the shortcomings of those discussed previously. The numerical answer is in terms of "Babbages", a unit of measure he has originated.

The Babbage rating of a computer is obtained by using the following equation:

$$B = \frac{L \log_2 M}{T}$$

Where: L = length of word (in bits) transferred to/from memory during the access time, T
M = total number of bits in high speed memory
T = access time in microseconds for transferring in L bits in parallel

* Of the System Development Corporation.

The introduction of the term L in the numerator as a multiplier, gives a much higher rating to those machines which transfer more bits per access time. This does not mean that, all other things being equal, longer words mean better computers. It means simply that the more bits that are transferred at each access, then the more information reaches the computer each access. In this respect more is better. As was stated earlier, there is a possible shortcoming here. Machines with proportionally longer words consume more time cycling and shifting data into the correct position (once it is transferred) if they do not have some character and/or partial word logic, as well as full word logic. The consideration of this term, then, while highly desirable, is capable of producing some error if the analysis does not guard against it.

The $\log_2 M$ term in the numerator states that each successive bit of storage added to memory is 1/2 the benefit to the user of the immediately previous bit of storage. This may be too severe a judgment upon the marginal value of increments of storage. In most discussions with programmers and system analysts, it has been found that the feeling is: "each bit is almost as valuable as the preceding bit. Almost — not quite."

There is a shortcoming in the construction of Shaw's "Babbage." When the logarithm of a number is multiplied by another number, the product is the logarithm of the original number, but to a new base. What this new base is is determined by the number used as the multiplier. A different number gives a different base. The equation governing this relationship is:

$$\log_x Y = \frac{1}{\log_{10} X} \cdot \log_{10} Y$$

This means that the logarithm of any number can be found to any base desired, given the presence of a table of common logarithms (\log_{10}). But it also means that in the Babbage computation the logarithmic base, used to evaluate the size of memory, varies inversely as the size of the word transferred from memory during the access time.

Stated another way, the error says that as the number of bits transferred from memory gets larger, the more valuable to the user is each succeeding bit of memory. How

valuable is dependent upon what size the word is; but here are three examples:

<u>If the multiplier is:</u>	<u>The percentage value to the user of each new bit in terms of the preceding bits is:</u>
6.8	71%
12.6	83%
24.1	90%

It is likely that each succeeding bit is something from 0.7 to 0.9 as valuable as the preceding bit, as discussed before. However, it is a flaw to have this value function fluctuate between computers – depending upon something else entirely. There is a method to consider word length transferred without encountering this difficulty, which is discussed later.

An interesting point is that since the log of the numerator is operated on arithmetically by the formula, the resultant Babbage reading can be manipulated arithmetically without the logarithmic difficulties mentioned in the discussion of the CFM.

The Babbage Method goes far toward providing a very useful measurement. It produces reasonable comparisons when the result is tempered by good professional judgment. It is worthwhile, however, to examine one more attempt to provide a figure of merit measurement.

3.10.8 The Highland Method

The Highland Method of computing figures of merit has been developed by E.K. Campbell.* It represents an attempt to produce a figure of merit method which obviates the internal logical and mathematical difficulties which appear in these approaches mentioned previously. It does not suffer from most of the logical and mathematical difficulties of other techniques, but is still subject to the inherent limitations of figure of merit.

$$HM = \frac{K (\log_{10} M)}{A \cdot \frac{T}{B}}$$

* Of Informatics, Inc.

- Where: **K** = conversion constant (see below)
 M = total bits in high speed memory
 A = add time (in microseconds)
 B = bits transferred in parallel during one access time
 T = memory access time (in microseconds)

K is the constant required to change the $\log_{10} M$ to the log of M to another base depending upon what value is selected for K. Table 3-3 which follows, shows some values to use for K, depending upon what value is selected for the marginal utility of additional memory.

Table 3-3. Values of the Multiplier "K"

Incremental Value of Additional Bits of Memory	Value of Multiplier "K"
0.40	2.5
0.50	3.3
0.71	6.8
0.77	8.7
0.83	12.6
0.90	24.1

The use of K allows the analyst to adjust the evaluation to reflect his professional judgment as to the incremental value of memory for the application. It is reasonable to believe that for most applications the value of K is somewhere in the vicinity of 0.7 to 0.9, though for some it could be much higher (or lower). The method of computing new values for K is as follows:

$$\log_x V = \frac{1}{\log_{10} X} \cdot \log_{10} Y$$

The incremental value is $\frac{1}{X}$.

Therefore: If the incremental value of bits added to memory is to be 0.4,

Then,

$$\frac{1}{X} = 0.40$$
$$X = 2.5$$

and, from the first equation,

$$\log_{2.5} Y = \frac{1}{\log_{10} 2.5}$$

$$\log_{10} 2.5 = 0.39794$$

$$K = \frac{1}{\log_{10} 2.5} = \frac{1}{0.39794} = 2.5$$

M is the total number of data bits in memory; that is, the total number of bits excluding sign and parity bits. \log_{10} is used since tables of this function are easily obtained, and multiplier K changes \log_{10} to whatever base it is wished to use.

A is the add time of the machine. It is necessary to use some direct measure of instruction time since it is possible for a machine to have a fast access time and a much slower instruction time than comparable machines. Add time is used since the type of circuit logic which makes add slower or faster normally makes other instructions slower or faster. In addition, two access-time instructions (such as add) are very frequently used, and add time by itself is not an unreasonable representation of computational speed.

The term $\frac{T}{B}$ allows consideration of the number of bits transferred in parallel (B) in the denominator, and thus avoids the difficulties involved in multiplying logarithms. T is in the denominator (of the entire expression) since a smaller time is better and this increases the size of the answer. Since T is divided by B, the result grows even smaller as B increases.

$\frac{T}{B}$ is multiplied by A to remove any undue advantage which could accrue to very cheaply built machines having a very fast transfer rate and something slow like a ripple-shift add logic. In addition, any slight advantages in computational speed by one machine over another should be fairly portrayed, since it is computation and not transfer rate that gets the task accomplished.

Table 3-4 shows the machines evaluated by the Highland Method.

In the Highland method there are a number of improvements over the other methods. As with the Babbage, the resulting Highland number may be operated upon arithmetically to solve analytical problems. This may be done since the rating number scale, after having been both multiplied and divided, is now linear (or very close to it) instead of logarithmic.

The Highland method measures what is to be considered in a logical and mathematically consistent manner. The resultant ratings may be manipulated analytically. Finally, the analyst has a method for adjusting the marginal value of incremental memory to the potential user.

3.10.9 Summary of Figures of Merit Comparisons

It must be understood that figures of merit have severe limitations both in their field of application and in the scope of factors which they consider. However, they are of great value to the analyst who understands them thoroughly. They can be, at the same time, professionally threatening to the executive or administrator who uses them casually -- without an understanding of what they mean or measure.

There is no satisfactory way at this time to bridge the gap between having a data processing requirement and selecting the appropriate machine for it, except to perform a detailed analysis of the task to be done. This analysis necessarily includes a bench-mark analysis unless the requirements are well-known in relation to the capability of a particular computer. Only then does a figure of merit comparison yield any meaningful results directly. Even so, the next step is often a bench-mark analysis.

Table 3-4. Highland Method Figure of Merit (With K for ~ 0.8 Value)

	Total Bits Mem. (M)	Access Time (T) Microsecs.	Bits Transferred in Parallel (B)	Add Time (A) Microsecs.	$\log_{10} M$	$K \log_{10} M$ (K = 12)	$\frac{T}{A \cdot B}$	$\frac{K \log_{10} M}{\frac{T}{A \cdot B}}$ (The Highland Rating)
CDC 6600	15,720,000	.35	60	.7	7.19645	86.3574	.00408	21,166
Philco 212	3,120,000	.75	48	.6	6.49415	77.8298	.00936	8,315
IBM 7030	16,768,000	1.10	64	1.5	7.22453	86.6944	.0256	3,386
Hughes H-330	6,288,000	.90	48	1.8	6.79851	81.5821	.0337	2,420
SDS 9300	768,000	.87	24	1.75	5.88536	70.6243	.0633	1,115
RCA 601	1,792,000	.75	56	5.7	6.25334	75.0401	.0752	997
Univac 1107	2,340,000	2.00	36	4.0	6.36922	76.4306	.222	344
Univac 490	960,000	3.00	30	4.8	5.98227	71.7872	.480	149
CDC G-20	1,024,000	3.00	32	15.0	6.01030	72.1236	1.40	51.5
SDS 910	384,000	4.00	24	16.0	5.54158	66.4990	2.67	24.9
CDC 160-A	384,000	3.20	12	12.8	5.54158	66.4990	3.42	19.4

The next limitation of figures of merit is that they necessarily cannot evaluate input-output capability or peripheral equipment configuration, since these are system (or problem) oriented and cannot be adequately determined in advance of problem definition.

Some additional key factors which are not considered by figure of merit methods are; instruction repertoire, amount and type of low speed storage, mean time between failure, mean time to restore, and amount of memory cycle overlap. These factors must all be carefully weighed in any complete analysis.

Figures of merit may be used quite well to evaluate the relative power of various central computers and their high speed memories independently of their application to a specific problem. Not only can they be used to solve the analytical problems posed earlier and other problems closely related, but also they can be used quite effectively to evaluate, from a cost-effectiveness point of view, proposed changes to data processing systems.

When memory size is considered, parity bits and sign bits should be excluded from the total, since they store little or no information. Some are required, but others may be superfluous for the task. The number (M) to be used is the largest memory size that the particular machine can be expanded to.

The illusory potential of logarithmic scales is completely covered in a previous section. This quality must always be kept in mind by the analyst. It begins to fade as linearity is restored by operating on the log arithmetically. Unintentional changing of the base of the logarithm results, however, if care is not exercised with these manipulations.

Access time and cycle time must be used carefully in evaluating destructive and non-destructive readout machines.

Another effect must be guarded against. In some machines memory banks may be arranged so that access time may be reduced by referring to these banks in rotation. This is called "overlapping." Some machines have this capability - others do not. The amount of allowable overlapping varies among models and as a function of how

many blocks of memory are purchased. Since the number of memory blocks to be required cannot often (if ever) be accurately determined at this stage of analysis, overlapping should be considered by the analyst; but not in the figure of merit computation.

One of the very low access times quoted by one manufacturer results from maximum overlapping (which cannot be used unless all possible memory banks are acquired), while a very low access time quoted by another manufacturer can still be reduced to about 2/5 of that quoted by the use of his maximum overlapping capability. So much for the technical content of descriptive literature. The competent analyst must be certain where each of his numbers came from and why.

Add time is probably as good an indicator of internal computational speed as can be found, and using it alone does not inject the tincture of simulation mentioned earlier. In certain situations where the internal speed of the machine is quite critical, the information channel capacity technique should be considered. Normally, the technique used in the Highland method should be adequate.

The concepts concerning word size have been treated in previous sections, but it is important to remember that big words are not always tantamount to better machines.

Since cost cannot accurately be predicted early in the analysis, and are subject to change due to the pressures of competition, they must remain outside the computation. This is true even though costs must be considered in any worthwhile analysis.

When only a small proportion of the high speed memory of a particular machine is of a much higher speed than the balance, such as 128 registers of thin film versus 32 K registers of core, then the thin film speed may be neglected entirely for the figure of merit computation. However, if machines are postulated which have 5-10% or more of main memory operating ultra-high speed, then this clearly must be considered in the computation. Just how to do this best is open to discussion at the moment. In the Highland method this factor likely appears as some sort of multiplier in the denominator.

The Highland method produces reasonable and rapid comparisons of computer main frame and memory capability for ACDS planning purposes when employed by planners experienced with data processing equipment.

Section 4 BIBLIOGRAPHY

4.1 MANAGEMENT METHODOLOGY

Affel, H. A., Jr., System Engineering, International Science and Technology, November 1964.

DOD Directive 7401.1, Cost and Economic Information System, July 7, 1964.

Enthoven, Alain, Systems Analysis and the Navy, the 1965 Naval Review, p. 98.

Farr and Nanus, Factors that Affect the Cost of Computer Programming, SDC, TM-1447/00, June 1964.

Farr and Zagorski, Factors that Affect the Cost of Computer Programming: A Quantative Analysis, SDC, TM-1447/001, August 1964.

Fisher, G. H., A Discussion of Uncertainty in Cost Analysis, RAND RM-3071-PR April 1962.

Glazer, H., and Janssen, T., Electronic System Cost Model, Mitre TM-3364.

Grant, E. L., Principles of Engineering Economy, The Roland Press, N. Y., 1950.

Hitch, Charles, Speech to Senior DOD personnel, Cost and Economic Information System, July 1964.

Katz, A. H., On Style in Research and Development, RAND P-2030, January 1960.

Kirchmayer, L. K., Planning of Systems, IEEE Transactions on Military Electronics, April 1964.

Large, J. P., ed, Concepts and Procedures of Cost Analysis, RAND Corporation, RM-3589-PR, June 1963.

McNamara, Hon. Robt. S., Statement Before the House Armed Services Committee, January 27, 1964.

Novick, David, System and Total Force Cost Analysis, RAND Corporation, RM-2695-PR, April 1961, p. 59.

Petruschell, R. L., An Introduction to Estimating Relationships, RAND Corporation, RM-3215-PR, June 1962.

Stock, Paul R., Sr., Estimating Procedures, Section 8, Ch. 1, Industrial Engineering Handbook, 1st ed, Maynard, H. B., ed, McGraw Hill, N. Y. 1956.

4.2 PLANNING METHODOLOGY

- Alexander, L. T., Man-Machine Simulation as a System Design and Training Instrument; System Development Corp., SP-331/000/01, Sept. 27, 1961.
- Anon, Mathematical Models for Information Systems Design and Calculus of Operations, Magnavox, MRL #R-451, October 1961.
- Anon, Real-Time Automobile Ride Simulation, WJCC, Vol. 17, 1960, P.285.
- Anon, Simulation of an Information Channel on the IBM 704 Computer, WJCC, Vol. 15, 1959, p. 87.
- Anon, Simulation, BRT-12 System Development Corporation.
- Anon, General Purpose Systems Simulator II, Reference Manual, IBM B20-6346.
- Anon, RTDHS Primary Site Programming System, Informatics Inc.
- Anon, Simulation, BRT-12, System Development Corp.
- Anon, Dynamic Instruction Count of a Real Time Program, IBM, Kingston, N. Y. October 1960.
- Anon, Data Processing, ANIP Research, Amelco, Inc., June 1961.
- Arnold, C. R., Digital Simulation of a Conformal DIMUS Sonar System, Phase I, AD-265398, 28 February 1961, p. 37
- Ashley, J. Robert, On the Analog Simulation of Mechanical Systems with Stiff Position Limit Stops, Simulation, May 1964, p. 21.
- Bekey, George A., Optimization of Multi-Parameter Systems by Hybrid Computer Techniques, Part I, Simulation, Feb. 1964, p. 19.
- Bekey, George A., Optimization of Multi-Parameter Systems by Hybrid Computer Techniques, Part II, Simulation, March 1964, p. 21.
- Bishop, W. A., and Skillman, W. A., Digital Simulation of Pulse Doppler Track-White-Scan Radar, IRE Internat. Convention Record, Vol. 10, Pt. 4, p. 94.
- Campbell, E. K., The Determination of Meaningful N-Tuples of Instructions in a Computer Program, SDC, November 1962.
- Cohen, The Design and Objectives of Laboratory Problem IV, RM-3354-PR, Rand Corp., January 1963.

Daev, D. S., Serdinov, A. I., Tarkhov, A. G., Model Simulation of Problems Bearing Upon the Method of Radiowave Sounding, Izv Akad Nauk SSSR. Geotiz, 1963, No. 6, p. 936 or (English trans.) Bull. Acad. Sci. USSR, Geophys., Ser. No. 6, p.573 (June 1963; publ. October 1963).

Geisler, M. A., and Steger, W. A., The Use of Manned Simulation in the Design of an Operational Control System; WJCC, p. 51, 1961.

Grabbe, Ramo-Wooldridge, Handbook of Automation Computation and Control, John Wiley & Sons, Inc., 1961.

Hara, Hiroshi H. Special Techniques for Two-Dimensional Air-to-Air Missile Simulation, Simulation, May 1964, p. 29.

Hicks, C. L., Analog Simulation of an Acquisition and Tracking Radar System with Command Capability, Eastern Simulation Councils, Mtg. 16, July 1962.

Katz, J. H. Optimizing Bit-Time Computer Simulation, Communication of ACM6, Nov. 1963, p. 679.

Kepcke, J., Computer Simulation of a Complex Secure Communication System, Eastern Simulation Councils Mtg. 16 July, 1962.

Meissinger, H. F., Simulation of infrared Systems, Simulation, March 1964, p. R-23.

Pickett, R. S., Investigation in Search of a Measure of Information Processing, unpublished, April, 1962.

Rector, Ronald W., Measuring the "Capability" of Computing Equipment, unpublished, November, 1962.

Rector, Ronald W., Predicting the Cost of Computing Equipment from its Physical Characteristics, unpublished, November 1962.

Robin, F. A., Pardee, R. S., Scheffler, D. L., Holland, F. C., A Computer Driven Simulation Environment for Air Traffic Control Studies; WJCC, Vol. 24, 1963, p. 437.

Rothman, Stanley, RW-40 Data Processing System, TRW, Inc., June 1959.

Smith, William E., A Digital Systems Simulator, WJCC, Vol. 11, 1957, p. 031.

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) Informatics Inc. 15300 Ventura Blvd. Sherman Oaks, California		2a. REPORT SECURITY CLASSIFICATION Unclassified	
		2b. GROUP —	
3. REPORT TITLE Advanced Naval Tactical Command and Control Study, Volume IV, Methodology			
4. DESCRIPTIVE NOTES (Type of report and inclusive dates) Summary of work concluded in period Jan 1964 to Jan 1965			
5. AUTHOR(S) (Last name, first name, initial) Bauer, W. F., Campbell, E. K.			
6. REPORT DATE 15 January 1965		7a. TOTAL NO. OF PAGES 296	7b. NO. OF REFS —
8a. CONTRACT OR GRANT NO. Nonr-4388(00)		9a. ORIGINATOR'S REPORT NUMBER(S) TR-65-58-2	
b. PROJECT NO. —		9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)	
c. —			
d. —			
10. AVAILABILITY/LIMITATION NOTICES U. S. military agencies may obtain copies of this report directly from D. D. C. Other qualified D. D. C. users shall request through the Advanced Warfare Systems Division of the Office of Naval Research.			
11. SUPPLEMENTARY NOTES None		12. SPONSORING MILITARY ACTIVITY ONR	
13. ABSTRACT Reviews a study to set forth improved methods and procedures for Navy planners to make decisions in development, design, and implementation of improvements to tactical command and control systems. This volume reports on the first year's study to analyze planning tools for system design and evaluation, and interprets their use in planning tactical command and control systems. The report discusses in detail planning for system management and the procedures to be followed in system planning. It discusses the role of cost effectiveness and how effectiveness can be measured. Methodology for system planners is treated, covering the role of simulation in system design, development, checkout, and test and evaluation. Simulation languages, mathematical modelling and queuing models are discussed. A new and improved method of determining figures of merit for digital computers is given. The volume recommends a management system for naval tactical command and control systems and concludes with a bibliography of management methodology and planning methodology.			

Methodology for Navy Command and Control Systems
 Tactical Command and Control
 Planning of Navy Data Systems
 Methodology for Systems Management
 Evolutionary Implementation
 Systems Management
 Naval Procedures for System Plan
 Planning Procedures
 Simulation in Systems
 Simulation Languages
 Mathematical Modeling
 Computers, Digital, Figures of Merit
 Management Methodology
 Planning Methodology

LINK A		LINK B		LINK C	
ROLE	WT	ROLE	WT	ROLE	WT

INSTRUCTIONS

1. **ORIGINATING ACTIVITY:** Enter the name and address of the contractor, subcontractor, or corporate author issuing the report.
- 2a. **REPORT SECURITY CLASSIFICATION:** Enter the overall security classification of the report. Indicate whether "Restricted Data" is included in accordance with appropriate DoD Directives.
- 2b. **GROUP:** Automatic groupings are provided in DoD Directive 5200.10 and Armed Forces Manual 3-10.1. Show that optional groupings have been used.
3. **REPORT TITLE:** Enter title in capital letters. Titles in all caps should be unclassified. If a meaningful title cannot be given, show title classification in all capitals in parenthesis immediately following.
4. **DESCRIPTIVE NOTES:** Enter report, e.g., interim, preliminary, final, specific reporting period as covered.
5. **AUTHOR(S):** Enter name of author(s) as shown on report. If military, show branch and principal author.
6. **REPORT DATE:** Enter month, year, or month/year on the report, as applicable.
- 7a. **TOTAL NUMBER OF PAGES:** Enter total pages, including front and back matter.
- 7b. **NUMBER OF REFERENCES CITED:** Enter the total number of references cited in the report.
- 8a. **CONTRACT OR GRANT NUMBER:** If appropriate, enter the contract or grant number under which the report was written.
- 8b, 8c, 8d. **PROJECT NUMBER:** Enter the appropriate identification, such as project number, task number, etc.
- 9a. **ORIGINATOR'S REPORT NUMBER(S):** Enter the report number(s) in the document which is being cataloged.
- 9b. **OTHER REPORT NUMBER(S):** Enter the report number(s) of other reports by the originator, or by the sponsor, etc.
10. **AVAILABILITY STATEMENTS:** Enter the classification of the report, other than that imposed by security classification, using standard statements such as:

- (1) "Qualified requesters may obtain copies of this report from DDC."
- (2) "Foreign announcement and dissemination of this report by DDC is not authorized."
- (3) "U. S. Government agencies may obtain copies of this report directly from DDC. Other qualified DDC users shall request through _____."
- (4) "U. S. military agencies may obtain copies of this report directly from DDC. Other qualified users shall request through _____."
- (5) "All dissemination of this report is controlled. Only DDC users shall request through _____."

If the report has been furnished to the Office of Technical Services, Department of Commerce, for sale to the public, indicate this fact and enter the price, if known.

- 11. **SUPPLEMENTARY NOTES:** Use for additional explanatory notes.
- 12. **SPONSORING MILITARY ACTIVITY:** Enter the name of the departmental project office or laboratory, or the name of the research and development, in full or address.
- 13. **ABSTRACT:** Enter an abstract of the report and factual information of the document indicative of the report, though it may appear elsewhere in the report. If additional space is required, an abstract sheet shall be attached.

It is highly desirable that an abstract of classified reports be unclassified. Each paragraph of the abstract shall be marked with an indication of the military security classification of information in the paragraph, represented by (C), (S), (U), or (U).

There is no limitation on the length of the abstract, however, the suggested length is from 150 to 250 words.

14. KEY WORDS: Key words are individual or meaningful words or short phrases that characterize a report and may be used as index entries for cataloging the report. Key words must be selected so that no security classification is required. Identifiers, such as equipment model numbers, trade names, military project code name, geographic area, may be used as key words but will be followed by an indication of technical context. The assignment of links and weights is optional.