

MASTER FILE COPY

R-1855/1-PR

April 1976

The Computer Resources Management Study

S. M. Drezner, H. Shulman, W. H. Ware,
G. K. Smith, M. R. Davis, R. N. Reinstedt, R. Turn

A Report prepared for
UNITED STATES AIR FORCE PROJECT RAND



ARCHIVE
RAND/R-
1855/1

The research described in this Report was sponsored by the United States Air Force under Contract No. F44620-73-C-0011 — Monitored by the Directorate of Operational Requirements and Development Plans, Deputy Chief of Staff, Research and Development, Hq USAF. Reports of The Rand Corporation do not necessarily reflect the opinions or policies of the sponsors of Rand research.

R-1855/1-PR

April 1976

The Computer Resources Management Study

S. M. Drezner, H. Shulman, W. H. Ware,
G. K. Smith, M. R. Davis, R. N. Reinstedt, R. Turn

A Report prepared for

UNITED STATES AIR FORCE PROJECT RAND

Rand
SANTA MONICA, CA. 90406

PREFACE

In January 1975, the Air Force Chief of Staff, General David C. Jones, requested that Rand undertake a high-priority study of policy and organizational aspects of the management of Air Force computer resources. The terms of reference for the study were developed and work was begun in February 1975, and the substantive work was completed in the late fall of that year. A progress briefing was presented to the Chief of Staff and various Air Staff deputy chiefs in mid-June. Major findings were reported on 24 July 1975 in a briefing presented to a similar group, and in September a summary of the study results was published as R-1855-PR, *The Computer Resources Management Study: Executive Summary*.

The present report provides a more detailed and complete description of the findings and recommendations of the study. Although issued some months later, this report was initially prepared in the late fall of 1975, and the text represents organizations and events as they were at that time. *No attempt has been made to incorporate any of the subsequent changes that have been made in the Air Force organization and management of computer resources.*

This study was performed as part of the Project RAND research project "Management of Computer Resources."

SUMMARY

Computer technology has become central to the Air Force's ability to perform its role and mission. Computer hardware failures and inadequate software can mean that an aircraft cannot fly or cannot carry out its design missions, that command and control systems cannot communicate with each other, or that important management information is not received in a timely or accurate manner. But, like other organizations, the Air Force is having trouble dealing with the management of its computer resources. The objective of the present study is to identify and recommend alternative courses of action in policy, management, and organization that would enhance the Air Force's ability to acquire, utilize, and exploit its computer resources.

In the course of the study, team members visited ten Air Force projects, each of which contains a substantial computer component. Some of the projects are now, or have been, in trouble; some are obviously headed for trouble; and others are generally considered to be in satisfactory condition. The projects visited included the Advanced Logistics System, STALOGS, BASE-TOP, the Advanced Airborne Command Post, the B-1 (both the System Project Office and the contractors), the AWACS, the F-16, the NORAD command-control upgrade (427M), the SAC command-control upgrade (436M), and TACS/TADS. In addition, team members interviewed a large number of individuals throughout the Air Force and the other military services, as well as several commercial firms with experience in developing and using large computer systems. These included United Air Lines, Hughes Aircraft, MITRE Corporation, and TRW. Finally, we reviewed an extensive body of literature on earlier computer development projects and on previous studies of computer resource management.

Based on information thus collected, we concluded that the difficulty experienced by the Air Force in managing computer resources stems principally from the failure to follow an adequately structured and properly managed development process. Additional problems stem from the multiple and overlapping sets of regulations and command organizations used to manage computer resources.

We, therefore, make the following recommendations to improve Air Force management of its computer resources:

- Establish a focal point in the Air Staff for all computer resources management, creating an Assistant Chief of Staff for this purpose and extending the focalization concept to major command level.
- Strengthen the development channel in the Air Staff and in AFSC so that these organizations can assume responsibility for managing and developing computer systems in all functional areas. This does not imply that these organizations should assume all responsibilities for computer efforts now performed in the various major commands.
- Establish centers of expertise in computer technology within selected organizations in the Air Force.
- Strengthen the computer-related career field.
- Strengthen or establish selected groups within the Air Force to assure that the requirements generation process adequately reflects both the genuine need of the user and technical feasibility.

ACKNOWLEDGMENTS

The study reported herein was performed by a team of thirteen
Rand staff members:

Louis Catlett	Anthony I. Robinson
Malcolm R. Davis	D. Kenneth Shelton
Stephen M. Drezner	Herbert J. Shukiar
R. Stockton Gaines	Hyman Shulman
Raymond A. Pyles	Giles K. Smith
Robert N. Reinstedt	Rein Turn
Willis H. Ware	

The authors of this report gratefully acknowledge the extensive and valuable assistance provided by the other team members. A special note of appreciation is due to Lt. Col. John Marciniak (then of the Directorate of Data Automation, Hq USAF) for the outstanding and complete support that he provided us throughout the study.

CONTENTS

PREFACE	iii
SUMMARY	v
ACKNOWLEDGMENTS	vii
TERMS AND ACRONYMS	xi
Chapter	
1. INTRODUCTION	1
Background	1
Organization of the Present Study	6
2. AIR FORCE MANAGEMENT OF COMPUTER RESOURCES	
IN 1974-1975	9
Management Structures and Regulations	9
Observations	15
Management Attitudes and Beliefs	20
3. BASIC PREMISES	22
Premise 1: A Systems Approach Must Be Used	23
Premise 2: Similar Management Procedures	
Should Be Used for All Systems	24
Premise 3: Development Management Methods	
Should Be Used	26
Premise 4: Life-Cycle Management Is Required	28
4. SUMMARY OF THE SYSTEM DEVELOPMENT PROCESS	30
Mission Requirements Formulation and Review	35
Technology/Requirements Validation	37
Design and Development	39
System Tests	42
Production	43
Operational Testing and Use	43
Maintenance	44
Application to Computer Resources	44
Summary	45
5. SUMMARY OF RECOMMENDATIONS	47
The Management Problem	47
Recommendations	49
6. IMPLEMENTATION DETAILS AND SUPPORTING RATIONALE	55
Focalization of Management at the Air Staff	55
Strengthening the Development Channel Within	
the Air Staff	67

Strengthening the Development Channel Within AFSC	78
Strengthening the Role of Single Managers in the MAJCOMs	81
Miscellaneous	83
Appendix	
A. TERMS OF REFERENCE FOR THE RAND STUDY ON AIR FORCE MANAGEMENT OF COMPUTER RESOURCES	85
B. PROJECTS, MILITARY ORGANIZATIONS, AND FIRMS VISITED	87
C. PERSONNEL AND TRAINING ISSUES	90
REFERENCES	95
BIBLIOGRAPHY	97

TERMS AND ACRONYMS

ACS	Assistant Chief of Staff
ACS-CRM	Assistant Chief of Staff/Computer Resources Management, Hq USAF (proposed new office)
ADC	Aerospace Defense Command
ADP	Automatic Data Processing
ADPS	Automatic Data Processing System
ADS	Automated Data System
AF/ACD	Directorate of Data Automation, Hq USAF
AF/ACDCA	AF/ACD Mission Systems Branch
AF/ACDCB	AF/ACD Management Support Systems Branch
AFDSC	Air Force Data Services Center
AFDSDC	Air Force Data Systems Design Center
AFIT	Air Force Institute of Technology
AFLC	Air Force Logistics Command
AF/LGY	Directorate of Maintenance, Engineering, & Supply, Hq USAF
AFM	Air Force Manual
AF/PRP	Directorate of Aerospace Programs, Hq USAF
AFR	Air Force Regulation
AF/RD	Deputy Chief Staff for Research and Development, Hq USAF
AF/RDP	AF/RD Directorate of Development and Acquisition
AF/RDPE	AF/RDP Electronic Systems Division
AF/RDPK	AF/RDP Computer Systems Division (proposed)
AF/RDQ	AF/RD Directorate of Operational Requirements and Development Plans
AF/RDQLM	AF/RDQ Management Office
AF/RDQP	AF/RDQ Support Division
AF/RDQPK	AF/RDQ Computer-Based Systems Office (proposed)
AF/RDQR	AF/RDQ General Purpose Forces Division
AF/RDQS	AF/RDQ Strategic Forces Division
AF/RDR	AF/RD Directorate of Reconnaissance and Electronic Warfare
AF/RDS	AF/RD Directorate of Space

AFSC	Air Force Systems Command
AF/XOO	Directorate of Operations, Hq USAF
ALC	Air Logistics Center
ALS	Advanced Logistics System
AMA	Air Materiel Area
ASD	Aeronautical Systems Division of AFSC
ATE	Automatic Test Equipment
AWACS	Airborne Warning and Control System
BASE-TOP	Base Automated System for Total Operations
CEIP	Communications-Electronics Implementation Plan
CEM	Communications-Electronics-Meteorological (Board)
CID	CEIP Implementation Directive
COE	Center of Expertise
CPDP	Computer Program Development Plan
CRISP	Computer Resources Integrated Support Plan
CRT	Cathode Ray Tube
DAR	Data Automation Requirement
DCA	Defense Communications Agency
DCS	Deputy Chief of Staff
DoD	Department of Defense
DoDCI	Department of Defense Computer Institute
DPD	Data Project Directive
DPI	Data Processing Installation
DPP	Data Project Plan
DSDC	Data Systems Design Center
DT&E	Development Test and Evaluation
ESD	Electronic Systems Division of AFSC
ESD/MCI	Information Systems Technology Office in ESD
FEDSIM	Federal ADP Simulation Center
GSA	General Services Administration
ICBM	Intercontinental Ballistic Missile
IOT&E	Initial Operational Test and Evaluation
JCS	Joint Chiefs of Staff
LMC	Logistics Management Center
MACIMS	Military Airlift Command Integrated Management System
MAJCOM	Major Air Command

MSS	Management Support System
NBS	National Bureau of Standards
NORAD	North American Air Defense Command
OCR	Office of Collateral Responsibility
OFP	Operational Flight Program
OPR	Office of Primary Responsibility
OT&E	Operational Test and Evaluation
PAR	Proposed Automation Requirement
PEM	Program Element Monitor
PMD	Program Management Directive
PMP	Program Management Plan
PPBS	Planning, Programming, and Budgeting System
R&D	Research and Development
RFP	Request for Proposal
ROC	Required Operational Capability
RRG	Requirements Review Group
SAC	Strategic Air Command
SAF/FM	Assistant Secretary of the Air Force for Financial Management
SIM	Simulation
SM	System Manager
SPO	Systems Program Office
SSP	Subsystem Project Plan
STALOG	System for the Automation of Logistics at Base Level
TACS/TADS	Tactical Air Control System/Tactical Air Defense System
USAF	United States Air Force
WWMCCS	World Wide Military Command and Control System

Chapter 1

INTRODUCTION

Computer technology has become central to the Air Force's ability to perform its role and mission, but the Air Force, like many other organizations, is having great difficulty in dealing with it. Historically, computer technology has been widely exploited in the Air Force, both in command and control systems and in management support systems, notably in such areas as logistics, comptroller functions, and base operations. In recent years it has also become an essential subsystem of modern aircraft, with regard not only to weapon delivery, both offensive and defensive, but also to navigation, to stores management, to on-board monitoring of performance sensors, to automated ground-support equipment, to crew and mission training simulators, and even to the actual flight control of the vehicle.

Thus, computer hardware failures or inadequate software can mean that an aircraft cannot fly or cannot carry out its intended missions, that command and control systems cannot communicate with each other, or that important management information is not received in a timely or accurate manner. Given such essentiality, it becomes increasingly important to master and exploit the computer, not only from a technical, but also from a management and policy point of view. The Air Force, although it has done a superb job in some instances and continues to make improvements over time, still faces significant difficulties in managing its computer resources.

BACKGROUND

In reflecting on a number of large, complex computer systems, Dr. Frederick Brooks [1] of the University of North Carolina, likened large computer programming efforts to the mortal struggle of prehistoric beasts trying to escape the tar pits:

Large systems programming has over the past decade been such a tar pit, and many great and powerful beasts have thrashed violently in it. Most have emerged with running systems--few

have met goals, schedules, and budgets. Large and small, massive and wiry, team after team has become entangled in the tar. No one thing seems to cause the difficulty--any particular paw can be pulled away. But the accumulation of simultaneous and interacting factors bring slower and slower motion. Everyone seems to have been surprised by the stickiness of the problem, and it is hard to discern the nature of it. But we must try to understand it if we are to solve it.

For many years, the Air Force too has had its own version of the tar-pit problem. The difficulties have appeared in all functional areas, including management support systems, command and control systems, and computers embedded in weapon systems. Furthermore, the rate at which complexity has increased in these systems has kept the quantity of unsolved problems at a high level, despite progress by the Air Force in dealing with some of them. It can be anticipated that military needs will continue to require systems that are barely within (or just beyond) the reach of the technology.

The consequences of developing large, complex systems at the frontier of current technical and management knowledge are illustrated in the case histories of projects that fell short--sometimes spectacularly--of meeting originally stated cost, schedule, and performance goals. In the Air Force, some of the more well-known examples include the ALS program,¹ elements of WWMCCS, and portions of the F-111D program. It is worth noting that the Air Force is not alone in the computer system development quagmire--the other military services, other government agencies, and commercial firms have experienced similar failures and frustrations. Even the computer industry itself has not escaped the cost overages and missed schedules of large computer system development, especially the development of software components.

It would not serve the purpose of this report to recount in detail the experiences of past computer system projects. That has been done in previous studies, some of which are summarized in this chapter, but the widespread commonality of the difficulties is an important element in the formulation of the present study. The art of computer resources

¹See p. xi for the definitions of acronyms used throughout this document.

management is still rapidly evolving, and the practitioners are still far from a consensus on major issues.

Computer problems are not new, nor are they a new subject for study by technical and management analysis teams. During the past ten years, such problems have been extensively studied in the Air Force, but in most cases the studies and workshops were directed at specific functional areas, as opposed to the general computer resources management problem. Some emphasized technical problems, while others stressed management: some were aimed at identifying problems only, while others were directed at both problems and solutions.

It is worth reviewing briefly some of those studies. As early as 1964, high-level Air Force management had already recognized that computer technology was developing rapidly and that expanding requirements would dictate a proliferation of applications. It was predicted (accurately) at that time that the Air Force computer inventory would increase by 50 percent in the ensuing two or three years [2]. In July 1964, a contract was awarded to the Planning Research Corporation for a "Study of Application Effectiveness and Problems of Air Force Information Processing Systems" [3]. Phase I was to synthesize an approach for providing summarized ADP² experience information to Air Force decisionmakers for support of the new automation proposal review process. The study recognized that there had been a concentration of management attention on hardware acquisition, despite the fact that software even then accounted for more than half of the ADP costs and far more than half of the problems.

Phase II [5] produced a pilot version of an ADP experience handbook, using data from 18 systems, but subsequent work revealed that the proposal review process was more complex and the management problem was far broader than originally envisioned. Accordingly, a recommendation

²In this report, the term "Automatic Data Processing (ADP)" is applied only to the use of computers in management support systems, and "ADP equipment" denotes computer equipment available in standard commercial product lines. Where specialized computers are used as components of weapon systems, they are referred to as "embedded" computers. This term was introduced by Lt. Col. John Manley in a paper presented to the Joint Logistics Commanders Workshop at Airlie House, Virginia, in May 1975 [4].

was made for the development of a complete ADP Management Information System with a goal of serving members of the Air Staff involved in the entire life cycle of ADP systems (proposal, development, and operation) [2]. The recommendation was never implemented.

In January 1970, General John C. Ryan (then Air Force Chief of Staff) established the USAF Select Committee on Computer Technology Potential. His action stemmed from concern about the ability of the Air Force to achieve maximum potential benefit from computer technology. In the midst of the study, a change of emphasis directed the committee to consider problems of computers integral to weapon systems. While the specific problems of such embedded computers were never adequately addressed by the committee, a number of recommendations for alternative organizational structures were made [6], the preferred structure involving the establishment of an Information Systems Office at the Assistant Chief of Staff (ACS) level. None of the major recommendations were ever implemented. Considerable planning was done in an attempt to implement the recommended establishment of a Computer Technology Center, but the idea was eventually abandoned, although a miniature version of the center was created at ESD.

Studies of the computer problem have not been limited to the Air Force. In 1967, the DoD undertook a study of procurement procedures, DoD computer inventory, and the use of ADP equipment [7]. The primary result of the study was a recommendation that changes be made in DoD Directives 5100.40 and 4105.55. In 1970, some of those changes were incorporated in a new version of DoD Directive 5100.40. In addition, responsibility for ADP was transferred from the Assistant Secretary of Defense/Installations and Logistics to the Assistant Secretary of Defense/Comptroller.

In 1970, the DoD initiated another study on ADP to be conducted by the Blue Ribbon Defense Panel [8]. The principal recommendation was that an office of Computer Systems and Services be established at the Assistant Secretary of Defense level, with broad powers and responsibilities for providing all ADP support to the military departments and other DoD components. However, such an office was never established, and consequently none of the other recommendations of the study were implemented.

The first significant study directed solely to embedded computers was initiated in 1972 by AFR-20-1, which established "Project Pacer Flash" to conduct an in-depth study of long-range computer software requirements for weapon system computers; the study group was chaired by the Air Force Logistics Command. The following are three of the twelve major recommendations made by the study [9]:

1. The Air Force should move in directions which will increase its organic capability for software support.
2. Software support should be made the explicit responsibility of the weapon system manager (SM) at the AFLC AMA where he resides and for the aircraft for which he is responsible.
3. Recognize common and unique requirements for software support in the ATE, OFP, and SIM areas as delineated in the report and physically and organizationally locate functions outside of AFLC (including AMAs) where these functions are most cost effective and responsive to user mission requirements. The SM retains configuration management control.

Pacer Flash was criticized because many believed it fell short of its goals in a number of ways. For example, it dealt only with aeronautical systems software, rather than all Air Force weapon systems computers, as had been directed by the charter. Also, it failed to compile a complete inventory of computers installed in weapon systems, although an incomplete list was published. In spite of its deficiencies, the study did bring the problem to the attention of the embedded-systems community, and it was a precursor to creation of the very important AFR 800-14 regulation.

While Pacer Flash was in progress, the Engineering Directorate of the Aeronautical Systems Division/AFSC reached the conclusion that the study was formulating draft policies and proposals for life-cycle management of embedded computer systems with which AFSC probably could not concur. Consequently, ASD instituted a special study activity to formulate its own position. The results of that study were published in October 1973 [10], and the principal disagreement was in the way Operational Flight Programs (OFP) maintenance would be performed.

Neither study resolved the problems of embedded computer systems, but together they did identify many of the issues. Probably the most significant specific action taken as a result of these studies was the initiation of work on AFR 800-14, which has become the prime Air Force regulation governing implementation of software. The studies also set the groundwork for the Aeronautical Systems Workshop in Dayton, Ohio, in April of 1974, during which a broad range of embedded-computer problems were discussed at length by experts from AFSC, AFLC, user commands, other government agencies, and members of industry [11].

Although the various studies had some effect on the computer system acquisition process, they failed to achieve the positive action needed to bring it under management control. There are a number of reasons for this, some of which are related to the limitations of the studies themselves, and some to the organizational climate that existed at the time action was being considered. For example, some studies produced recommendations that were too general in nature and consequently were not actionable. Furthermore, because the pervasiveness of the computer problem was not fully appreciated, the studies were focused on problems at local levels or at specific functional areas. The overriding global issues were missed. Attempts at action were further confounded by an organizational arrangement that has traditionally diffused responsibility for computers into different areas; people and organizations, as would be expected, maneuvered to protect and maintain their purview.

ORGANIZATION OF THE PRESENT STUDY

In recognition of the major difficulties many of its elements were continuing to experience in the development and use of large computer systems, the Air Force, in early 1975, requested The Rand Corporation to undertake an examination of Air Force management of computer resources. The terms of reference for the study are reproduced in App. A.

From the above discussion, it is apparent that most of the previous studies focused on specific functional areas as opposed to the general computer resources management problem. Some emphasized technical problems, while others stressed management aspects. Some identified problems

only, and others were directed at both problems and solutions. None looked at the matter in the large. Thus, the Project RAND study was initiated under a very broad charter, encompassing all functional areas and dealing with the overall management of computer resources. Furthermore, the Air Force emphasized from the beginning that the study must not only identify problems, it must also ". . . propose alternative courses of action in policy, management, and organization which address both immediate and long-range problems [and] identify the important consequences of implementation for the alternatives."

In response to this very broad charter, the study was structured around the following guidelines:

1. It should draw from the experience and insights of both the users and the developers of large computer systems.
2. It should encompass the entire set of computer applications in the Air Force (i.e., management support systems, command-control-communications systems, and weapon systems).
3. It should focus on broad management policy and organization, without attempting to define in great detail the implementation of policies.

The first step was to review the history of several major development projects, each of which involved computers in some large way. Most of them were Air Force projects, but a few from other government agencies and commercial firms were included (see App. B). At the same time, the then-existing organization of computer management in the Air Force was reviewed, including the external constraints imposed by Federal law and by DoD policies and regulations; this aspect is summarized in Chap. 2.

Our reviews of management practices and the actual outcomes of projects conducted under them provided background data for the next phase of the study: a digest of the "conventional wisdom" about computer system management, followed by a careful and critical examination of that wisdom in light of actual experience. This resulted in the definition of four basic premises which seemed to fit experience best; they are discussed in Chap. 3.

Another important result of the analysis was the formulation of a conceptual structure of the development process as a system life-cycle chart. The latter has proved to be very useful in communicating essential features of the issue and in providing a basis for conceptualizing and presenting the final recommendations; this chart is described in Chap. 4.

After assuring ourselves that the basic premises were consistent with the information obtained from the interviews conducted during the study, we undertook the final phase, devising Air Force organizations and management procedures that would encourage and facilitate the management of computer resources. The results are summarized in Chap. 5, and discussed in greater detail in Chap. 6.

Chapter 2

AIR FORCE MANAGEMENT OF COMPUTER RESOURCES IN 1974-1975

This chapter outlines the management structure for acquisition of computer resources as it existed within the Air Force when this study was begun. It represents the baseline for the changes recommended in later chapters of this report.

In this discussion, many subtleties of the organization and management process are omitted for the sake of brevity. The intent is to describe a management concept, while not complicating the description with all the details necessary to make the concept work. The salient conclusion is that the existing structure and its implementing regulations cannot meet goals of:

- Designing a total system.
- Treating computer hardware and software together rather than separately and serially.
- Emphasizing motivation of user-stated requirements rather than details of the regulations.

MANAGEMENT STRUCTURES AND REGULATIONS

The management of most of the computer resources in the Air Force in early 1975 was conducted under one or the other of two different sets of procedures: (1) the AFR 300 series regulations (Data Automation) that were rooted in Federal legislation (the Brooks Bill, PL 89-306, dated October 1965) and that regulate use of standard, off-the-shelf commercially marketed computers for management support systems, and (2) the AFR 800 series regulations (Acquisition Management), particularly AFR 800-14. The latter procedures regulate the weapon system development process and apply to computer resources embedded in weapon systems which are exempted from the Brooks Bill (and AFR 300 series) requirements. In addition, certain computers used in telecommunication and command and control systems were acquired under the AFR 100 series regulations (Communications-Electronic Activities)

and under several other regulations (AFR 57 series, AFR 56-2, AFR 71-11, AFR 80-1 and AFR 102-5). There were overlaps in the applicability of these regulations, and some systems, such as command and control, could be developed under AFR 300, 800, or 100 series regulations, or be subject to all three series. The language used in these regulations tended to address computer *equipment* (the hardware) rather than the overall computer system, indicating a general lack of management attention in the acquisition of software.

Standard ADP Systems

Responding to the Brooks Bill requirements, the DoD named the Assistant Secretary of Defense (Comptroller) to be responsible for the management of ADP resources subject to the Bill. In the Air Force, the Assistant Secretary of the Air Force (Financial Management) was designated as the Senior ADP Official.¹ Subsequently, the following management structure was established by Secretary of the Air Force Order 560.1, AFR 300-1, and 300-2:

- Authority to manage the Air Force ADP Program was delegated by the Chief of Staff to the Director of Data Automation under the direction of the Comptroller of the Air Force. All proposed data automation requirements above certain thresholds had to be reviewed and approved by the Director of Data Automation.
- Command ADP Program Single Managers were established at major commands and at separate operating agencies and were made responsible for the ADP programs within their organizations. The Director of Data Automation delegated authority to Single Managers for acquisition of command-unique computer resources that were below certain cost and manpower thresholds.

¹A similar line management structure, with ADP resources controlled by the financial management office, was common practice in commercial firms at that time.

- The Data Automation Panel of the Air Force Board was established to review and monitor proposed and existing data automation programs.
- The Air Force Data Automation Agency, commanded by the Director of Data Automation, was established. It consisted of the Air Force Data Systems Design Center (AFDSDC), the Air Force Data Services Center (AFDSC), and the Federal ADP Simulation Center (FEDSIM). AFDSDC developed standard software packages for the Air Force, AFDSC provided data-processing support for the Air Staff, and FEDSIM supported all agencies of the Federal government.

Under this management structure, which is presently in effect, a request for new ADP resources or services may be submitted in the form of a Data Automation Requirement (DAR) by any Air Force organization to the designated approval authority as determined by the level of resources involved. A DAR identifies the requirement for which computer resources are sought, presents an economic analysis, describes benefits expected (e.g., a saving in manpower requirements), and suggests alternatives. After approval by the ADP Single Manager at the appropriate level, a Data Project Directive (DPD) is issued to the developing agency, which in turn prepares a Data Project Plan (DPP). Both are submitted for approval to the appropriate ADP Single Manager.

In summary, the AFR 300 series regulations, in theory, provide the Air Force and the MAJCOM ADP Program Single Managers with ample means for critical review of DARs and their implementation. In practice, however, the Single Managers at all levels are placed in an advocacy role for data automation requests. The review of a DAR frequently becomes an exercise in rewriting in a language that enhances approval, rather than a critical examination of underlying rationale and economics.

Embedded Computers

The resource requirements for computers that are part of a weapon system are usually included in the overall weapon system Required

Operational Capability (ROC) document submitted to AF/RD in the Air Staff for approval and funding. AF/RD then issues a Program Management Directive (PMD) to the appropriate developing agency (usually AFSC), authorizing that agency to initiate work on research and development of the system and to provide direction to the implementing and participating commands. The developing agency then produces a Program Management Plan (PMP). An approved project is managed in AFSC by a program manager who has full authority over the project, including the computer subsystem selection or development, and who retains responsibility until the project is turned over to the requesting command for operational use and to AFLC for maintenance.

The management of computers embedded in a weapon system is specified in AFR 800-14, Vol. 1 (first issued in May 1974). This directive requires that computer resources in such systems be managed as elements or subsystems of major importance during all phases of system development, that the management responsibility for integration of computer hardware and software into the system remain centralized for the life of the system, and that this responsibility be transitioned along with the entire system. It further specifies that PMDs require (and PMPs provide for) establishing computer technical and managerial expertise at Program Offices.

Some additional guidance for the management of embedded computer systems is provided in AFR 300-2, which states that the ADP Single Manager's role is to review and assist in the preparation of those portions of ROCs, PMDs, and other documents that involve ADP resources. During the development process, the ADP Single Manager's role is to support the Program Manager. However, AFR 300-2 does not establish clear lines of authority and responsibility. Presumably, those were determined individually for each project at the time the PMD and PMP were drawn up.

Near the end of the Project RAND study, the Air Force issued AFR 800-14, Vol. 2, *Acquisition and Support of Computer Resources in Systems*, which addresses in the fashion of a manual the development of software for embedded computers. This regulation goes into greater detail than any previous regulations do, and it recognizes some of the peculiarities

of software and its development. In particular, AFR 800-14 points out that satisfactory performance of computer software may not be completely demonstrable and assessable until the completion of the operational test and evaluation (OT&E) phase in the system's cycle, and that even after the system is in operational use, changes in software may be required to remove errors, improve efficiency, adapt to changes in system requirements, and incorporate knowledge obtained in operational use. It emphasizes that *software cannot be a turn-key component*.

The regulation specifies in considerable detail the type of direction that must be given by a PMD for the acquisition of embedded computer resources and software. The development of a Computer Resources Integrated Support Plan (CRISP) is required early in system acquisition. The objective is to identify the offices of primary responsibility for management and technical support of computer resources in the post-transition phase of the life cycle. The development of a Computer Program Development Plan (CPDP) by the implementing command or the contractor is specified, and the establishment of a Computer Resources Working Group is suggested for preparing the CRISP. The regulation describes the life cycle of implementing and supporting computer software² and discusses in detail the management of each phase.

Communications-Oriented Systems

The AFR 100 series regulations govern the procurement and acquisition of computers integral to telecommunications systems, telecommunications services in support of ADP systems, and terminal equipment. The Air Force Command, Control and Communications Program includes most of the communications-electronics work that comes under the AFR 100 series. Development is initiated by the submission of a Communications-Electronics Implementation Plan (CEIP) to the command-level Communications-Electronics-Meteorological (CEM) Board for review. After approval is obtained, the CEIP is forwarded to the Air Staff for review and

²The term "software" is used here in the broadest sense to include operating systems, mission application programs, and all utility and support programs.

approval. The latter issues a CEIP Implementation Directive (CID) to the implementing command (AFCS, AFSC, or a designated command), which appoints a Project Manager, who produces a Subsystem Project Plan (SSP) for the procurement or acquisition of the system.

Command and Control Systems

Wide latitude has been permitted in approaches to the development of computer resources for the Air Force command and control systems, with a general overview being maintained by the Air Staff. Developments could be performed under either the AFR 300 or 800 series regulations, or both. For example, if a command and control system development involved a large system engineering and programming job, it might be done in the R&D community under the AFR 800 regulations. The initial part of the 436M SAC Command and Control update program (part of the Air Force transition to the WWMCCS configuration) was done in such a manner. Subsequently, a number of in-house tasks brought the system to a full operational capability. Most of these were done via the DAR process under AFR 300 by the Command ADP Program Single Manager in SAC.

Another example was the 427M development at Air Defense Command, which began with an AFSC SPO and several industrial contractors. As that program approached an operational capability, management of the development was assumed by ADC, and the remaining software development was taken over increasingly by in-house groups within ADC. In contrast, a command and control system could be developed in-house from the beginning under the AFR 300 series regulations. The Military Airlift Command's MACIMS system (Program 415M) was developed in-house, with AFDSDC providing software development support and the MITRE Corporation providing some of the engineering support.

Practically all command and control systems at Hq USAF and the major commands are parts of the World Wide Military Command and Control System (WWMCCS) and thus must comply with the WWMCCS development goals of a high degree of commonality in equipment and software, and interchange of information with command and control systems in other services and at the national level. Centralized ADP support for WWMCCS

was provided by the Joint Technical Support Activity, a field activity of DCA.

AFR 102-5, *USAF Management Policies Governing Development, Acquisition, and Operation of Command Control Systems*, provides one set of broad guidelines for the Air Force WWMCCS program. Development in incremental steps is called for, with off-the-shelf inventory or commercial equipment to be used where feasible and practical. Standard, nonfunctional software is to be centrally controlled by Hq USAF or by a designated agency. The unique functional software is to be controlled by the user, whose participation throughout the development life cycle is called for, with a strong emphasis on maintaining and utilizing in-house hardware and software development capability.

Further direction is provided by AFM 300-12, Vol. II, where Chap. 3, "Management of the USAF World Wide Military Command and Control System (WWMCCS) Automatic Data Processing System (ADPS 80)," states that JCS-prescribed procedures are to be used in conjunction with the AFR 300 regulations and manuals for management, development, and operational responsibilities associated with ADPS. Centralized control, but decentralized development and implementation, is called for, with AFDSDC providing technical support and maintaining complete information on hardware and software configurations.

OBSERVATIONS

As mentioned earlier, several separate computer resource management regulations existed in the Air Force when this study was undertaken. Each of these organizational structures assumed that a functionally different system (i.e., a management support system, weapon system with embedded computers, or command and control system) called for a different approval justification and a different management treatment in the acquisition phase. At the same time, frequent uncertainty about which set of regulations was applicable (particularly in the command and control areas) has caused both confusion and maneuvering. The latter resulted not from determining which regulation was most pertinent, but from decisions about which one seemed to promise the highest probability of approval. The system has not precluded submitting

requests through two management chains simultaneously or, if denied approval under one regulation, submitting the request under a different regulation.

In addition to the problem of multiple and overlapping management procedures, several other features of the Air Force computer resource management system deserve special notice; these are discussed below.

Advocacy

A definite, although probably unintended, bias has existed toward emphasizing advocacy in the Air Force computer resources management structure. Thus, the same entities in the management chain that were responsible for validating and approving a computer acquisition request have also been responsible for generating or providing assistance in the generation of that same request, and after approval, have also sometimes been responsible for presenting it in proponent roles to higher approving authorities. For example, according to AFR 300-2, the Major Command ADP Program Single Manager "will . . . develop or assist in developing ADP requirements in support of the command mission" and also "evaluate, approve, and implement, or disapprove suggestions concerning unique ADP systems." The Hq USAF Staff Offices, likewise, are to "develop and validate ADS requirements," "develop and review requirements," and "support the acquisition of ADP resources. . . ." While this arrangement undoubtedly has been responsive to the shortage of managers with computer expertise, it is unrealistic to expect someone involved in advocating a new system to simultaneously conduct a critical review of the requirement or the development plan for that system.

In-House Versus Contracted Development

It is Air Force policy not to manufacture any equipment in-house, and this includes computer hardware. In the area of computer software, however, the policy is different. AFR 300-1 states that Air Force in-house capability should be developed and used, whenever technically or economically advantageous or essential to mission accomplishment, in the design, development, acquisition, implementation, operation, maintenance, and management of ADP systems. However, no such policy

statement is made in AFR 800-14, as the selection of the developing organization (usually a contractor) is entirely within the authority of the Program Manager.

The policy encouraging in-house software generation reflects some of the differences in hardware and software. No large capital investments are required to set up a software development facility--only manpower and access to an appropriate computer are required. Hence, in management's view, it appears advantageous to use programmers already at an ADP facility to develop the new software and thereby avoid the problem of requesting additional funds for that task.

Software development performed by an outside contractor inherently includes a confrontation process via the contract negotiations, whereas an in-house software development task is typically subjected to considerably less critical review. For example, AFR 300-2 specifies that the approval authority of a command ADP Single Manager in acquisition of a commercial software package is limited to \$50,000, and in acquisition of contractual services is limited to \$100,000 per contract. However, the Single Manager can authorize organic system design and programming up to 25 man-years--a significantly higher cost ceiling than either of the above. Without stringent review and control of the proposed in-house software developments, the development efforts tend to be underestimated and the expertise of the in-house programmers overstated.

External Constraints

Numerous requirements of the Brooks Bill, the Armed Services Procurement Regulations, and various rules established by the General Services Administration are applicable to Air Force acquisition of ADP resources. Those requirements apply in particular to the acquisition of ADP *equipment*, thus placing the emphasis in system development on hardware, rather than on the entire system, which comprises both hardware and software. Indeed, those requirements make only fleeting references to software. As a result, a great deal of attention is focused on hardware procurement in the acquisition of ADP systems--an emphasis that is inconsistent with the relative costs of hardware and

software. More importantly, it is inconsistent with the fact that the goal is to design a total system, with hardware and software treated jointly rather than separately and serially. Furthermore, motivational emphasis tends to be on the details of the regulations rather than on the user-stated requirements.

The acquisition and procurement of commercially available hardware is emphasized in regulations governing ADP systems. They emphasize competitive selection of ADP equipment from commercial vendors; sole-source acquisition is discouraged. Specifications in RFPs that might favor a particular ADP equipment or vendor, or limit the number of vendors who would qualify, are prohibited, as are specifications of delivery or installation times that may be regarded as unreasonably short.

A primary objective of the Brooks Bill is to foster economy in ADP acquisition and utilization by Federal departments and agencies. The Bill calls for the establishment of an ADP sharing program, requires competitive bidding, and urges coordination of equipment needs so that the government can act as a volume purchaser. Unfortunately, those provisions have, in some cases, resulted in the acquisition of less-capable equipment, when a more advanced item was available from only one vendor but sole source acquisition was difficult to justify. Also, accepting the lowest bidder in hardware procurement sometimes results in a proliferation of different types of ADP systems, with new systems unable to utilize existing software and sometimes even having inferior system software capability.

In the case of upgrading the equipment at a computing installation, a more subtle effect can occur. The objective in this activity should be to convert to the new equipment at the lowest total cost; but by directing attention to the hardware aspect with little consideration for software cost, a lowest-bidder competitive procurement can have the effect of incurring reprogramming costs for the functional-area programs that far outweigh the savings from the low bid. The importance of this phenomenon is due to the ever-increasing quantity of programs that have to be moved to new machines; and, as installations continue to become larger, software problems will be of steadily increasing importance.

Personnel and Expertise Problems

Over the years, the Air Force has developed many large computer systems and conducted many studies of the computer systems development and management process. The experience gained in system development and the insights and recommendations produced by the studies represent a wealth of information and knowledge, but, unfortunately, it is distributed among many people who move from one job to another rather than concentrated in fewer people of long tenure. Thus the Air Force has not been able to fully capitalize on its own experience, nor has it been able to reap the full benefits of its many studies. In contrast, private organizations and the computer industry have a much greater ability to learn from past experience.

One of the reasons for the Air Force's inability to capitalize on its experiences is the personnel rotation policy that moves people to new assignments, often unrelated to their previous ones. Although there are many valid reasons for following the rotation policy, it creates a serious handicap to the establishment of a high-quality, organic technical management capability in a rapidly changing area like computer technology; the art is moving much too fast for large numbers of personnel in rotation to acquire experience in depth. The rotation policy also inhibits development of a "corporate memory" of experiences, both good and bad, in computer system development.

Part of the problem is that people, not organizations, learn; and when people move on, the only thing left behind is documentation in the form of reports, experience handbooks, regulations, study outputs, etc. Even this process is imperfect; the Air Force personnel normally do not stay with a project throughout its development life cycle, and no single individual has the complete picture or sufficient insight to prepare detailed experience documentation.

The next stumbling block comes in the utilization of whatever experience documentation and study results are available. New projects are never exactly the same as those in the past, and experience with the latter requires extrapolation to the new problems. Furthermore, each project management team brings its own experiences (and biases), and a project manager is not likely to spend time reading documents

that he believes relate only indirectly to his present project. Historically, the civil service work force has supplied the needed continuity and corporate memory; however, their ranks are very thin in the computer system development area.

Another problem is that few decisionmakers in the Air Force have experience with design and implementation of systems that depend heavily on computer technology. Thus, they may approve requests for new capabilities without full awareness of the difficulty of their implementation or of the consequences of the decision. Additional personnel issues are discussed in App. C.

MANAGEMENT ATTITUDES AND BELIEFS

Computer awareness among the managers of systems acquisition projects has been increasing rapidly in recent years. Nevertheless, there still exists a set of management attitudes and beliefs which reflect a lack of understanding of computer system development and management processes. In general, subscribers to such beliefs and attitudes tend to underestimate the cost, time, and expertise required to develop computer systems and, in particular, computer software. Almost invariably, this results in a lack of attention to the computer software early in the system's development phase and, consequently, substantial additional expenditures of resources during the system's integration, testing, and operational phases.

It is important to the rationale of the recommendations we make in this study to explicate the nature of those attitudes and beliefs; we describe them below as a set of myths whose validity will be refuted in subsequent chapters. While some may appear overstated and others not entirely self-consistent, collectively they are the basis for the existing management attitudes and practices toward implementation of computer-based systems. As such, they partially explain the development difficulties that have been encountered:

Myth: It is relatively simple to state valid and complete requirements for a new computer system.

- Myth: Hardware and software can be purchased or developed *separately* and fitted together later, and then they can be fitted into the administrative and procedural environment even later.
- Myth: Because requirements tend to evolve rapidly, the development of command and control systems (and, to a lesser extent, management support systems) must be performed *only by the user* and must be developed in a *continually evolving* manner.
- Myth: Since software is somehow different from hardware, its acquisition must be managed differently, or it may not even need to be managed at all.
- Myth: Management review mechanisms used for hardware are *superfluous* for software or are *impossible to conduct*.
- Myth: Many hardware deficiencies or inadequacies can *easily* be offset by changes readily made in the software.
- Myth: Acquisition of software can be treated as a *production-like* process, similar to procurement of standard hardware.
- Myth: Software, once developed, *never again* needs to be changed.
- Myth: Maintenance of software is essentially *the same* as maintenance of hardware.

Chapter 3

BASIC PREMISES

In the course of this study, considerable information was collected on the histories of projects that involved computer system development over the past decade. This information was gathered from the Air Force, from other elements of the DoD, and from private industry. The objective was to develop a substantial data base on varied projects which would not only include the factual outcomes but would lend itself to analysis of why those particular outcomes occurred. In the examination of these experiences, we sought first to separate cause and effect and then to distill the causes to a common basic set. Finally, from the varied cause-and-effect relationships, we sought to construct a conceptual model that would explain and account for the diverse problems and successes that we noted. This model forms the foundation for our recommendations for improving the outcomes of future computer system projects.

This process led to the identification of four basic premises which we believe to be fundamental to the successful management of computer resources. Since the specific recommendations outlined in the subsequent chapters of this report are based on these premises, it is appropriate to begin by identifying them and describing the basis for the importance placed on each; then we can discuss our recommendations for management actions and organizational changes.

As in most matters of importance, there are divergent schools of thought on these four issues, and none are subject to a rigorous proof. There was, in fact, sharp dissent among members of the study team on some issues during the early phases of the study; but as the evidence accumulated and as the debate became more focused, a consensus emerged that was further tested through conversations with experienced management personnel throughout the computer community. We found widespread concurrence with our views.

Finally, it should be noted that not all of the points expressed below are completely new; several have, in fact, been recommended in various forms during one or more earlier studies. It is also proper

to observe that the existing Air Force procedures for managing computer resources contain elements that are responsive to one or a part of each premise. However, taken as a whole, acceptance of the views outlined below would necessitate a substantial revision of the present Air Force computer resources management system. Such a revision is outlined in subsequent chapters.

PREMISE 1: A SYSTEMS APPROACH MUST BE USED

The phrase "computer system" usually refers to the combination of hardware and software needed to perform some task. In its broadest context, the phrase must also include hardware maintenance, software support, documentation, data sources, personnel training, administrative practices, and operational procedures. However, the acquisition and operation of these various parts are frequently managed separately and are sometimes even overlooked, with little or no attention being paid to the overall integration problem.

It became clear to the study team that a computer project must be managed as a system, rather than as a series of separate parts. In particular, the software and hardware must be considered as integral parts of one system. While quite different in a product sense, both are created by intellectual and manual effort of people. While the distribution and kind of required skills may be different, each is subject to a common, integrated management approach throughout the life-time of the system, from concept formulation to phase-out. The differences that do exist amplify rather than reduce the need to consider software and hardware as closely interrelated components of a system, rather than as individual entities to be managed separately and in different ways. It is also essential that system designers be provided an environment which not only permits but even encourages them to address the composite job. The computer is one element of a total system; therefore, designers need to be able to treat it as a holistic subsystem during most phases of the design process.

It is especially important to address the hardware-software interaction during the initial phases of a project in which both new hardware and new software are being developed. The interactions between

them are many and intimate; unless the development of the two is closely integrated, endless opportunities arise for each to accumulate design characteristics inconsistent with the other. Frequently such inconsistencies are not visible until very late in the development, and their reconciliation can then be costly and time-consuming.

Moreover, there are tradeoffs between the two that can only be examined jointly. Hardware features can facilitate development of software or even decrease the job to be done; conversely, software can do some things that are difficult to realize by hardware.

PREMISE 2: SIMILAR MANAGEMENT PROCEDURES SHOULD BE USED
FOR ALL SYSTEMS

A careful examination of the computer systems now in operation and under development in various functional areas (such as command and control systems, management support systems, and computers embedded in weapon systems) has failed to reveal any basic differences that would justify different management approaches. Perhaps the greatest demonstrable difference among them is that in management support systems and command and control systems the hardware is often procured as standard commercial items, whereas in weapon systems new hardware may be developed for a specific application. However, there are occasions when special-purpose hardware is developed along with the software for management support and command and control systems as well, so that management in even these areas must be able to accommodate total system development. Another difference that now exists among functional areas is that the software for management support and command and control systems is frequently developed organically, whereas software for weapon system computers is almost always acquired under contract. These differences fail to justify a different management approach; certainly the management of in-house software development should involve the same management controls that are applied to commercially contracted software.

An argument frequently used to justify a different management approach for developing some systems, especially command and control systems, is that defining operational needs and functional specifications

for the computer system is so difficult that an "incremental"¹ approach is necessary. A number of factors reinforce this argument, including the complexity of the system being developed, the degree to which the functions are well understood or definable vis-à-vis new functions that are not well understood, and the likelihood that the specifications will rapidly change for legitimate reasons. Some systems can be completely specified during the requirements validation phase with great confidence that subsequent events will not soon invalidate the details. In others, there is a substantial risk that the specifications will need to be changed even before the system development has been completed. There may even be occasions when it would be necessary to carry out operational tests of system components or prototypes in order to prepare complete functional specifications. While this discussion has focused on command and control systems, it should be noted that management support systems can exhibit the same characteristics, especially as they progress from simple record-keeping systems to dynamic, interactive, and even decision-oriented ones embedded in the day-to-day operation of the user.

Thus, there are indeed certain differences between systems that can be well specified in advance and systems that may require an "experimental" approach to requirements specification. However, the differences are not fundamental; the experimental approach can be viewed as a sequence of small steps to the end goal, with each managed as an orderly development process; where a complete set of initial specifications is possible, the entire development may occur in one large step, managed in the same orderly way. The same system development process

¹The terms *evolutionary* and *incremental* take on a variety of meanings when applied to computer development. In the present study, we have adopted the following precise set of definitions: *evolutionary requirements* are achieved by *incremental development* according to a *phased* program where each phase is a *complete* pass through an orderly process of development, producing an increment of capability.

The inappropriate phrase "evolutionary development" can lead to the impression in managers' minds that the process of development does not have to be managed or controlled but can be allowed to muddle forward with a guaranteed successful outcome. Thus, we prefer not to associate "evolutionary" in any way with development; we use it only as a descriptor of system requirements.

applies to both; except possibly for scale and perhaps a difference in details or formality, the same management practices apply to both; the structured management doctrine applicable to well-defined systems can also be applied to a phased incremental development.

PREMISE 3: DEVELOPMENT MANAGEMENT METHODS SHOULD BE USED

The Air Force has a long history of leadership in the formulation of management procedures for controlling the development of complex weapon systems. Unfortunately, the process of acquiring computer systems has frequently been treated as a simple procurement task, when in fact it is a development process which demands the most sophisticated R&D acquisition management techniques.

Both the hardware and software components of new computer-based systems require highly disciplined development management for several reasons:

1. Both involve considerable innovation, creativity, and ingenuity throughout the development stage. As a result, precise scheduling, cost estimation, and performance prediction are extremely difficult. Both components must be treated as items for acquisition, not simply procurement.
2. Both are subject to piece-part testing during the development phase, and both require extensive testing of the complete system to validate the proper integration of the many parts and components.
3. Both are vulnerable to tinkering and modification, during development as well as after being placed in operation, so that configuration control becomes an important and essential function. Configuration control of software is an especially critical task because of the ease with which seemingly simple changes can be made without adequate management visibility, and with unperceived implications for the future.

It is especially important that software acquisition for a new computer system be managed according to development principles, not

only for the above reasons, but also because software presents some special problems that require careful adherence to a structured development process.² For example, whereas the basic laws of physics clearly constrain the performance of hardware systems, and these constraints are widely recognized and understood, those on software performance are poorly defined, incompletely known, and largely experiential. The body of experience and expertise on software which can be drawn upon and utilized throughout the development phase is much smaller than that for hardware; many aspects of system design, performance prediction, and resource estimation are based on knowledge from similar past projects. Thus, there are several important consequences for software projects, especially major projects:

1. Software cost and schedule estimates can easily have relatively large uncertainties.
2. The preferred design approach--or, in some cases, the only workable one--may not become apparent until a substantial amount of preliminary design and development work has been done. To reach this point, it may be necessary to use prototypes, simulation, and even field tests.
3. The body of specifications, design standards, preferred practices, and documentation standards for software is still relatively small.

These characteristics raise special problems for software acquisition. Even the present systems development management procedures (the AFR-800 series regulations) are based on the premise that reasonably precise cost and schedule estimates can be made at the beginning of full-scale development and that a body of specifications and design standards does exist. It is clear that the art of software development has not yet reached such a degree of stability and predictability.

Finally, it should be noted that software has no significant "production" phase. An important consequence of this is that the development

²This phrase, applicable to the overall process, is not to be confused with "structured programming."

cost for software is a significantly larger fraction of the total life cycle cost than it is for hardware. This accentuates the need for both complete specification of functional requirements and thorough preliminary and detailed design prior to initiation of full-scale software development.

These factors in combination further strengthen the view that development of large software systems is very much an R&D type of activity (as opposed to a production one) and calls for the same style of management that has evolved for large noncomputer systems, one that is alert to the subtleties and special needs of this particular type of development.

PREMISE 4: LIFE-CYCLE MANAGEMENT IS REQUIRED

Both the hardware and software components of large computer systems require considerable support after delivery to the customer. Hardware may have to be modified, and it always needs maintenance because it fails or wears out. Software is inevitably subject to design oversights or errors not caught in testing, and the cost of modification, redesign, and functional improvement is frequently a major item in its total life-cycle cost. While the break between development and operation is relatively clear and precise for hardware, it is much less so for software.

There are basic differences between hardware and software support. Software does not "wear out" in the usual meaning of the term, and the phrase "software maintenance" is a misnomer in such a context. During the operational phase, however, software invariably requires substantial amounts of support, in the form of product improvement. The mechanics of changing software are straightforward and simple; the replication of changes in many copies of software is also straightforward. This has led to the mistaken belief that software is easy to change. What is missed is that software is an intricate fabric of interactions and mutual dependencies; the effects of changes in one part of a computer program on other parts are hard to perceive and can, in fact, introduce new and unforeseen problems and anomalies.

The design of software improvements is technically very difficult and challenging. It requires deep insight and knowledge of the original software system to analyze the impact of specific changes on total system performance. Consequently, it is frequently difficult or even impossible for someone other than the original developer to successfully make major changes in a software program without investing a substantial effort in becoming familiar with the program. Ideally, it would be expedient for the original developer to retain a close organizational tie to a software program throughout its operational life. Alternatively, the agency responsible for operational support must be thoroughly involved in the development program, must participate in the acceptance test, and must acquire an understanding of the software details that approximates that of the original developer.

Chapter 4

SUMMARY OF THE SYSTEM DEVELOPMENT PROCESS

Acquisition of any major system by the Air Force or other DoD agency is usually visualized as an orderly R&D process that consists of the following phases: military requirement; concept of operation; validation of requirements; full-scale development; production; deployment; and operational support. Sometimes a prototype phase is included to resolve uncertainty or to test different ideas. A project advances from one phase to another only after it satisfies specific requirements and criteria. Many project shortcomings have resulted from failure to follow such a process, or from incomplete understanding of the nature of the full life cycle of a system.

Figure 1 synthesizes a system life cycle in the form of a flow-chart. This chart, while similar to traditional models of the development process, contains several differences that are critical for the life-cycle management of computer system projects. In the remainder of this chapter, we shall discuss the structure and the elements of the chart in more detail to provide a background and rationale for the management recommendations presented in Chaps. 5 and 6.

We recognize, of course, that not every project should be made to follow the procedure outlined below in every detail. However, to simplify the presentation and to lend emphasis to what we believe is a very important central theme in computer system development, strong declaratory language is used throughout the description with only an occasional reference to the possibility of variations in detail application.

The chart shows the sequence of phases in the life cycle of any system. Each phase is represented by a large "box," and a line coding system is used to indicate the types of organizations that are principally involved in each phase: The functions enclosed in heavy solid lines (boxes A, D2, F) are performed by users; those in light double lines (boxes B and D1) ideally should involve independent nonadvocacy groups with computer expertise; and those in the light single box (box C) principally involve a computer system design and development organization.

ILLUSTRATIVE "SYSTEM DEVELOPMENT" LIFE CYCLE

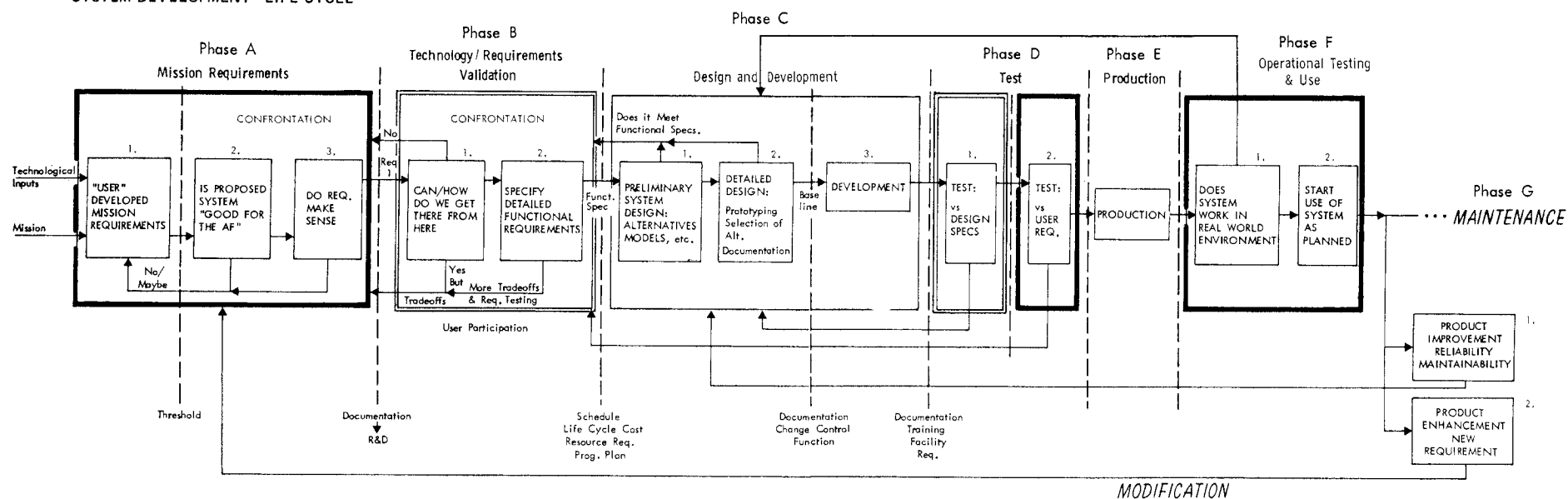


Fig. 1—Illustrative "system development" life cycle

A single military command could conceivably be responsible for all the functions indicated, but the various organizations within the command that are to establish the requirements, develop the system, perform the reviews, conduct the tests, and operate the system must be separate to encourage productive confrontation, and they must be sufficiently separated organizationally to avoid the compromise of one function by authority imposed by another.¹

The major subfunctions and decision criteria for each phase or function of the life cycle are also shown in the chart. We have deliberately avoided the use of jargon, but the chart can be related to the corresponding terminology, check points, and management decisions in the AFR-800 series regulations. For example, the Preliminary Design Review would occur at the end of phase C1, the Critical Design Review would occur at the end of phase C2, and the Test phase (D1, D2) represents DT&E and IOT&E, respectively.

The iterative nature of the development process is underscored by the many feedback paths between functions. Numerous reviews and iterations occur not only in the various phases of the development but also during any modifications later in its life cycle. Development, by its very nature, is a problem-solving activity rather than a straightforward application of a predefined set of steps, and all the details of the design cannot be expected to be right the first time. If design reviews are established at appropriate points in the development process and if all of those involved--managers, designers, users, reviewers--understand in advance that a review may well result in repeating a previous part or parts of the development activity, then the process can proceed in a controlled and rational manner.

¹The phrases *organizationally separated* and *independent* occur frequently throughout the discussion, implying an aspect of degree. The intent is to assure that an organization required to perform some function (e.g., testing) cannot be thwarted by organizational authority or preempted from completing its function without the topmost levels of management becoming aware of what has happened. For example, a slippage in the development schedule must not be arbitrarily compensated by shortening testing time unless the end user and all relevant levels of management understand and accept the performance risk that will be incurred.

Indeed, one of the important and deep causes for many unsatisfactory system development efforts has been the expectation and understanding, explicit or implicit, that the developer must produce the right design the first time and may not admit that errors have been made or take adequate steps for correction. Perhaps even more serious is the belief that the user can produce functional system specifications simply by sitting down and writing them. Several iterations, involving personnel from both the user and the computer system development community, are usually needed before specifications are complete and valid.

Finally, before proceeding with a more detailed discussion of the chart, we would like to emphasize the following general points about the chart and its applicability to the development of computer systems:

1. The system development and life-cycle process depicted in the chart applies equally well to a turn-key system, to each phase of an incremental development of a system, or to each step forward in the phased progression of a system. The chart specifically does not imply that a turn-key process is necessary or desirable in the development of a computer system.

2. The importance of *constructive confrontation* in the Mission Requirements Formulation (A) and the Technology/Requirements Validation (B) phases must be explicitly emphasized. Clearly, the requirements generation function must be a user-dominated, user-controlled phase in the life-cycle process. Some of the tasks represented in phase B may be part of the requirements generation process in that the output of phase B is a functional specification. In some instances, the user may have to prototype parts of the system, or even the whole system, in order to resolve uncertainties of feasibility, desirability, and operational payoff. Such prototypes are not necessarily the mechanization that will be followed in the formal system development, but are merely a tool in the requirements generation process.

It is also important that the user remain involved during the subsequent design and development process, but in a carefully controlled manner to avoid interfering with its orderly conduct. Frequent or uncontrolled changes in functional requirements, especially when the system is in full development (phase C3), can have disastrous effects. The user must also be responsible for the final test phase (D2) prior

to production, because only in this way can he be assured that his real operational requirements have been satisfied.

MISSION REQUIREMENTS FORMULATION AND REVIEW

The mission requirements formulation phase (A) is exceedingly important in that it provides the performance inputs that drive the rest of the system development process. This function must be controlled by users rather than developers.²

Requirements formulation (phase A1) is a relatively informal and sometimes lengthy process. When the operational need is sufficiently defined to permit consideration of the development of a system to meet it, the process of review and confrontation (phases A2 and A3) can be undertaken. In phase A2, the stated mission is reviewed from a top-level Air-Force-wide point of view, focusing on the functional rather than the technical aspects of the mission. Among the criteria that must typically be considered are the following:

- Does the Air Force really need the proposed capability?
- Is the proposed capability or system consistent with the long-range goals, policy, and planning of the Air Force, as well as with external directions from the DoD or the Congress?
- Do systems exist within the Air Force that could be used to satisfy this need? Is there a similar need elsewhere in the Air Force? Will the proposed system be exportable to the rest of the Air Force, or could it be so designed?

These questions serve the dual function of establishing the overall importance of the proposed system and alerting those who may in the future become involved in adapting the system to standard Air-Force-wide use.

Given a satisfactory review of the stated need for which a system is to be developed, it is important to address the adequacy of the

²By *developer* we mean the group of computer specialists plus other technologists who work from a detailed functional specification to produce a complete operational system that satisfies the specification.

performance requirements that have been proposed for use in the design of the system (phase A3). That is, do the requirements, as stated, "make sense" in that they represent what is really needed for the mission?

The questions asked in such a review should include the following:

- If automation is proposed, is an automated system really necessary? Computing is not a free good.
- Do systems exist in the Air Force that could satisfy the stated requirements, or are such systems in the development process? Duplication of effort should be avoided.
- Can the stated requirements be understood by the developers? Are they understood (in detail) by the users themselves?
- What impact will the proposed capabilities have on other Air Force functional areas?
- Are there interfaces or interorganizational issues that must be addressed?

At the end of phase A3 a set of requirements will have been produced that can usually be documented in the form of an approved ROC or DAR and submitted for technology/requirements validation (phase B). This is not to imply that phase B is undertaken only after a ROC or DAR has been fully approved. In some cases, the requirements are sufficiently uncertain that ROC or DAR approval is premature. In those cases, the user must perform further studies, and some of the phase B processes can be helpful or necessary to guide him. More specifically, the user may need to simulate the functional operation, build laboratory prototypes, or even build operational prototypes of the proposed system in order to understand basic questions such as exactly what he wants, whether it will work, and what the payoff is. If an exploratory development project is needed, and if it is felt that the system and resources must be especially accounted for, then an approved ROC or DAR would be necessary, under the present system. The need for exploratory work during the requirements formulation process should be explicitly recognized and provided for in the management procedure. While this is recognized in the basic research/exploratory development/advanced

development sequence within AFSC, there are no corresponding opportunities for a phased approach in nonweapon systems.

TECHNOLOGY/REQUIREMENTS VALIDATION

In phase B the mission requirements are refined to a more detailed stage. The goal of the phase B functions is the further validation of the original requirements from a technical and cost viewpoint. The process involves *confrontation* between the user and the system developers to determine the technical feasibility of implementing a computer system that satisfies the stated requirements; it is a task that requires close interaction between users and computer system developers. The user organization³ is still in complete control at this point, because the phase B function is a prelude to computer system development. It is important that activities in this phase be closely *reviewed* by a group organizationally *independent* of both the user and the actual system design and development organization. Technical experts are needed to identify and analyze technical problems and to perform tradeoffs. Satisfactory answers must be given to the following questions:

- What degree of technical risk or uncertainty is involved in meeting the requirements? Is any research indicated? It must be clear that the developer will be able to translate the user's requirements into a working system at reasonable time and cost, and if he cannot, the technical questions needing resolution must be identified.
- Is the system commercially available? While this is primarily a hardware consideration, the question may also be applicable to certain software packages.
- Do the requirements, as stated, provide sufficient information to allow a realistic cost estimate?
- In view of any technical risk, is the projected development schedule appropriate?

³The term *user* or *user organization* does not necessarily refer only to the ultimate user, but rather is a generic term meaning a functionally oriented (rather than computer oriented) entity that represents the end user.

- Are there any special technical aspects that would suggest extraordinary management measures? Schedule or funding contingencies?

After completion of the technical review--and there may be several iterations of modifying functional requirements and evaluating the technical aspects of their implementation--the requirements and technical inputs are amalgamated into detailed functional requirements (phase B2). These are again reviewed against a set of criteria which must include:

- Do the detailed functional requirements specifically and accurately define the desired capabilities and the necessary resources, and is there an accurate match between the user's stated mission and the requested resources?
- Is the proposal ready for design and development (phase C), or is further research necessary?
- What, if any, beneficial tradeoffs can be identified from an examination of the requirements? This feedback mechanism may present the user with choices such as a less ambitious but less costly system, or a system that, while providing less, can become operational at an earlier date.
- What technical management has been incorporated in the proposal to assure successful implementation? How is it assured that data automation aspects of the system will get a high level of management visibility?

It should be noted that a substantial amount of time and technical work, including the development and testing of a prototype system or a portion of one, may be needed to satisfactorily answer these questions. The organization that performs (not reviews) these tasks will probably change as a function of the particular situation. In some instances the technology/requirements validation is performed by a non-user organization

at the request of the user. For other systems (the ones possessing higher uncertainty with regard to various key questions), the user's involvement needs to be more intimate. This does not mean that the user organization must duplicate all the highly specialized skills one would expect to find in a computer system design organization, but it does suggest that the user needs ready access to those skills.

The final product of the technology/requirements validation phase is a detailed functional specification that has been subjected to thorough functional and technical reviews and that includes a set of firm performance specifications for implementing the desired functional capabilities. A Program Management Directive then defines a management approach consistent with the technical characteristics of the project, including consideration of anticipated technical risks.

DESIGN AND DEVELOPMENT

The design and development phase (C) is often called full-scale development. This step requires, for the first time in the process, a joint consideration of hardware and software. Some initial attempts at understanding the feasible and possible hardware/software configurations are included in the technology/requirements validation phase, but up to this point the process has been requirements-oriented and relatively free of implementation considerations.

The design and development phase is performed by an organization (either in-house or an outside contractor) that, ideally, is completely independent of the user organization.⁴ The design process naturally continues to be an iterative one throughout this phase. A careful distinction must be maintained between detailed requirements documents and corresponding design proposals, but at the same time they must be correlated until a design emerges that reflects both acceptable performance and technical practicality.

This phase proceeds into the preliminary system design step (C1), in which analyses and tradeoffs are made to produce a set of design alternatives that are then evaluated on the basis of such criteria as

⁴It should be noted that this policy has not been followed in some recent Air Force managed support systems and command and control system developments, with undesirable consequences.

performance, cost, and reliability. Important considerations that must not be overlooked include:

- Does the preliminary system development plan, especially those parts that include computer equipment and software, provide adequate management visibility?
- Is there a plan for proceeding through the development process in an orderly manner? Such a plan is especially needed for software, where specific provisions must be made during the design phase to allow subsequent testing, integration, and support activities.
- Is there a plan for orderly transition to the user and to the support organization?
- Does the design meet the functional specifications?
- Does the design skirt or satisfactorily hedge against identified technical risks?

After the completion of a preliminary design review,⁵ the detailed design of the system (phase C2) begins. In some cases, this may involve the construction of prototypes; in others, such as in a large software system, a major part of the entire design may have to be partially implemented to explore and determine such things as logic, interfaces, and control flows. An iterative process is to be expected, returning to earlier steps if the effort to produce a detailed design reveals some inability to meet functional specifications. The eventual detailed design must then be subjected to a thorough review, the critical design review.⁵ Such a review is especially important for software systems, and the essential questions include:

- Is it clear that the correct design choice was made?
How was the choice justified? What is the evidence?
- Have relevant hardware/software tradeoffs been adequately performed?

⁵Reviews must be conducted by people having sufficient technical and operational expertise to insure that the right questions are asked.

- In view of the technical insights now available, are the mission requirements still properly stated for smooth progress? At this critical juncture in the development process, we can still afford to review this fundamental question.
- Have significant changes been made in the cost estimates and development schedule? Since more information is available now than when the estimates were first made, does the additional information indicate that new cost and time estimates should be made? If so, does this influence any earlier decision to proceed in a particular way?
- Does the original design strategy still appear valid? Does the system development plan still seem appropriate? Is the projected management structure still proper?

In the development phase (C3), the detailed design is implemented in a form that can be tested and, in the case of hardware, then passed into production. For software, the final development step consists of writing the actual computer code in the chosen programming language. Except for testing, this completes the software development process, since no significant production step is involved. Throughout the development phase, the project management must remain aware of considerations such as these:

- Are there any technical problems that must be solved before proceeding with the development effort?
- Is the development effort conforming to the Program Management Plan?
- Are milestones, benchmarks, and other checkpoints being met? If not, what remedial actions are necessary?
- Are there any factors at this stage that require major revisions in the development process? In the schedule? In the technical approach?

With the completion of the development step, the system is ready for a sequence of tests.

SYSTEM TESTS

The testing phase (D) comprises two distinct steps involving distinct organizations. However, the first step is not necessarily totally separate from the development phase. Testing of subsystems can proceed as soon as their design and implementation is complete; consequently, there may be an appreciable amount of overlap between the design phase (C) and the first test phase (D1).

Testing of the system performance against the design specifications (D1) must be performed by a technologically competent organization independent of the system developer. The principal question is whether or not the system performs as specified in the detailed design specifications produced and accepted in phase C2.

The second test step (D2), which is performed by the eventual user organization,⁶ involves the system's test from the point of view of functional performance: Does it satisfy the needs of the user? If the detailed functional specifications are not met, it may be necessary to return to the design and development phase. Tests may also show that even though functional specs are fulfilled, the system is not truly responsive to the mission requirements, necessitating a return to the detailed functional specifications (B2) or even to an earlier step in the life-cycle process. For example, in some cases (such as the decision-focused parts of command and control systems) the functional requirements are difficult to define and may not be implemented exactly right at first. In other cases, the system capabilities may need to react to changes in high-level doctrines or policies as well as to changes that may have occurred in operational environments. In any such instance, however, the current product may still be able to provide an initial increment of capability; the development process will have produced a basic design that can be modified as indicated. If functional requirements have changed significantly before the system is fully designed, there is an obvious management choice

⁶In contrast to the footnote on p. 37, the actual end user is to be involved at this point.

to be made. Several options are available: (1) complete the development and put the limited-capability system into operation; (2) halt development and modify the design as necessary; or (3) scrap the development and start at the beginning to define and develop a new system.

Consistent with the emphasis placed on independent review of each phase of the development process, a special review group is needed to oversee the testing phase. Such a group should be represented in every major testing activity (e.g., the formal qualification test), should evaluate the reports on system module tests and integration tests, and should assure that the tests specified in the detailed design and in the Program Management Plan are conducted and the objectives met.

PRODUCTION

For hardware systems, completion of the test step is followed by conventional factory production (phase E). For software subsystems, however, the production phase is minimal, consisting of the automated preparation of error-free copies of the original programs for distribution to the users. The absence of a production phase in software is a major distinction between hardware and software, especially in terms of funding. Nearly all of the software acquisition cost must go into development, thereby putting additional strains on a traditionally tight budget category.

OPERATIONAL TESTING AND USE

The final test of the system's ability to satisfy the user's specified need comes in the actual operational use (phase F) of the system in the real-world environment. At this point, the user has formally accepted the system from the developing organization and, as with many systems, the responsibility for further support has been transferred to a supporting organization (e.g., AFLC or AFDSDC). Transitioning of the system is an important life-cycle step that is often overlooked at the beginning of the design process. This has been especially true of computer systems, and consequently, in a number of them, the necessary operational support equipment was not ready when the system became operational. The recently published AFR 800-14,

Vol. 2, recognizes this problem by requiring the preparation of a Computer Resources Integrated Support Plan (CRISP) as early in the system development process as possible.

MAINTENANCE

The maintenance function (phase G) assures that the system remains operational despite the natural deterioration or failure of the system's hardware. In computer software, there is no intrinsic wear-out process; however, errors sometimes exist in the accepted software, and these represent a latent problem for the support organization. Hence, the term *maintenance* in the context of software may be misleading, since it does not correlate with hardware maintenance or traditional logistics usage. A better term (and one that is being increasingly used) is software *support*, which spans product improvement (e.g., increasing its efficiency, reliability, or supportability), product enhancement (providing new functional capabilities), and the correcting of anomalous behavior due to design oversights or unfound errors. These attributes of support frequently force a return to much earlier phases of the software development process; e.g., sometimes extensive redesign is necessary, or functional performance must be degraded or at least changed to accommodate a software deficiency with existing hardware. This is often not recognized, and managers or commanders who believe the myth that "software is easy to change" are apt to ask for software changes much more casually than they ask for hardware changes. Clearly, some suggested product improvements require going through the complete life-cycle development process. In all cases, however, the user, or user organization, should have management control of changes to his system. This is true even when the user is not providing the software support.

One consequence of this special nature of software support is that software is usually at some level of continuing development; therefore, it appears to be never completed to any set of managers and auditors who believe the myth that turn-key software can be produced.

APPLICATION TO COMPUTER RESOURCES

While the process depicted in our life-cycle chart is normally

followed in the development of major weapon systems, its application to computer-based capability (either as an independent system or embedded in another system) has been sporadic. However, case histories show that skipping any phase in the orderly process schematized by the chart greatly increases the risk that a given system will not meet performance, schedule, or cost expectations. It is essential that the development of computer resources be subjected to strict discipline.⁷

This total life-cycle process is not a part of the existing Air Force computer management organization (except in the case of some embedded computers). It will be necessary to apply vigorous and independent management review practices to assure that these time-consuming, and often troublesome, but vital procedures are adhered to in the future.

The listed functions themselves are generally agreed-upon elements of the life-cycle development process. Yet some organizations have tried to omit one or more of them; others have tried to perform them simultaneously; and others have behaved as if the steps were a sequential set to be followed from left to right without iteration. In addition to these counterproductive practices, there are organizational questions as well. Who should perform each function? Who should be responsible for the various reviews? Some organizations have combined the tasks of development and testing under common management; others have split the design and development management to address hardware and software issues independently. There are potentially many reasonable alternative responses to these questions. We have already indicated some of our preferences. In the following chapter we suggest an overall alternative that we believe provides the most benefits.

SUMMARY

The essence of our argument is that development must be an orderly process that includes certain functions carried out by appropriately independent organizations. While the "right sequence" and the "right organization" are central issues that we emphasize, even together they cannot assure success; they are necessary but not sufficient conditions.

⁷Sometimes, the thrust of our argument is described as "application of engineering discipline to software."

Among other things that can bedevil a project are technical problems-- an aspect we do not consider. Even though the process we describe is carried out to perfection, major technical difficulties, procurement delays, catastrophic events, or a major misjudgment on availability of some technical item can ruin an otherwise properly managed and conducted development program.

Chapter 5

SUMMARY OF RECOMMENDATIONS

Because of the vital role of computer technology in the Air Force's ability to perform its mission, it is imperative that solutions be found to the problems that beset the management of the Air Force computer resources. Unreliable hardware or inadequate software can mean that weapon or other systems are not available when needed or cannot fully perform their intended missions. These operational consequences are the result of management problems whose rectification requires that the Air Force achieve control and comprehensive management visibility of its computer resources.

In the previous chapters of this report we have identified some of the critical issues and limitations in the current Air Force organization and policies for managing computer resources; now we shall describe a set of management actions designed to ameliorate the major problems. The issues and recommended solutions are summarized in this chapter. Chapter 6 presents an expanded discussion of implementation details and some of our reasons for selecting these specific recommendations from the broader set of possible "solutions."

THE MANAGEMENT PROBLEM

While many individual problems and limitations can be identified in the present Air Force management of computer resources, there appears to be one basic problem that has led to many of the current operational deficiencies:

- Failure to consistently follow an adequately detailed and structured process of system development such as the one depicted in Chap. 4.

This basic management problem has its source in some key organizational limitations and management misconceptions that we have identified and discussed in previous chapters:

- A lack of knowledge, awareness, and skills at the management level in the various phases of the system development life cycle.
- An inability to implement uniformly the intent of the present regulations. This appears to be partly a skills problem at the management level, partly an organizational problem, partly a lack of managerial insight, and partly a matter of attitudes and philosophy with respect to the required management process.
- An inadequate focus and emphasis on the mission requirements process depicted by the first box (phase A) in Fig. 1 and the technology/requirements validation process depicted by the second box (phase B).
- A lack of adequate confrontation in the technology/requirements validation process, and thus inadequate analysis of tradeoffs between mission requirements and computer technology. The mechanism does not exist at present to ensure that the necessary tradeoff studies are performed and the results utilized.
- An inability of the Air Force financial reporting structure to account for the cost of development, procurement, and operations.
- A lack of insight into the problems of software development (as opposed to hardware) by Air Force system managers; software tends to be given the "black box" treatment. The matter is pushed off onto contractors or superficially described without a clear perception of the real state of the software art. This results in overly optimistic schedules, underestimates of cost, and unrealistic performance expectations.
- Inadequate technical support in the formulation of management decisions. Reviews intended to form checkpoints in the development process only superficially examine critical technical questions.
- Uncontrolled user interaction with the design and development phase. This tends to cause repeated

modifications to a system during development, and consequently the system may never be completed and put into useful service.

RECOMMENDATIONS

We believe that several key organizational and policy changes are needed if the Air Force process for acquiring computer-based systems is to be properly managed and controlled. These changes involve reorganization of existing entities, reassignment of responsibilities, and creation of new procedures. Our recommendations are summarized as follows:

- Establish a focal point in the Air Staff for all computer resources management, creating an Assistant Chief of Staff for this purpose and extending the focalization concept to major command level.
- Strengthen the development channel in the Air Staff and in AFSC so that these organizations can assume responsibility for developing computer systems in all functional areas.
- Establish centers of expertise in computer technology within selected organizations in the Air Force.
- Strengthen the computer-related career field.
- Strengthen or establish selected groups within the Air Force to assure that the requirements generation process adequately reflects both the genuine need of the user and technical feasibility.

Implementation of these changes, plus others implied by them, should create the organizational and policy structure necessary to insure that computer system design is approached from a total systems point of view, that software acquisition is treated as a development process, and that there will be a uniform management and policy treatment of all computer systems regardless of functional area.

Focalization of Management at the Air Staff

The overall process whereby a computer-based system is brought into being, from initial requirements through final user acceptance, is inherently the same for all systems--command and control, subsystems embedded in a weapon system, or management support systems. Thus, the Air Force must establish uniform policies, procedures, management controls, and approval processes for every kind of computer-based system that it acquires. It is imperative that (1) adequate oversight be exercised to assure compliance with the intent, not just the form, of the system development process; (2) visibility of computer resources development be provided at the highest level of Air Force management; and (3) an unambiguous signal of concern be given to the entire Air Force on the seriousness of the computer problem.

To accomplish this, we recommend creating within the Air Staff a focal point for computer resources management that will establish policy, monitor the life-cycle development process, act as a central source of information and expertise on computer matters, provide an authoritative and visible point of contact, and act as an Air Force sponsor for the computer career field. Analogous focalization in the major commands is also recommended.

In particular, we recommend that an Assistant Chief of Staff of the Air Force for Computer Resources Management be created to serve as a focal point in computer matters. This action, the strongest alternative, has the advantage of transmitting the required signal of concern. It can provide the focal point with a stature that will permit productive interaction with the existing power structure in the Air Staff, provide access to important top-level deliberations and participation in decisions, and establish credibility and acceptance in the Air Force.

Strengthening the Development Channel Within the Air Staff

To bring acquisition of computer-based management support systems into the familiar Air Force development process, it is necessary to strengthen the development channel that traditionally has been the DCS/Research and Development (at Hq USAF) and AFSC. It seems appropriate that an augmented and strengthened AF/RD and AFSC should accept

the responsibility for the development of all types of computer systems needed by the Air Force. Therefore, we recommend that the AF/RD and AFSC not only continue their present responsibilities but also accept the additional area of management support systems. However, it is not necessary for all computer systems to go through the major development channels of AF/RD and AFSC. For example, a major command could develop a relatively small system (from a funding or manpower point of view) or a command-unique system with in-command resources--following, of course, the same development life-cycle process and subject to the same kind of management review discussed in Chap. 4.

As stated above, the acquisition of a computer-based system must be managed as a development process in all functional areas--and we stress that management support systems are included. It is, therefore, natural for AF/RD to be the technical advocate for these systems within the Air Staff. Thus, both Data Automation Requirements (DARs) and Required Operational Capability statements (ROCs) would flow into AF/RD for coordination. At present, ROCs flow into AF/RD, but DARs flow into the Directorate of Data Automation (AF/ACD).¹

We recommend that AF/RDQ and AF/RDP be augmented to handle the OPR and PEM functions for the management support systems in addition to their present responsibilities for embedded computer systems and some command and control systems.

Strengthening the Requirements Generation Process

An underlying characteristic of the Project RAND recommendations is that the computer system development process must be organizationally separate from, but carefully interfaced with, the system requirements generation process. It is clear that the user must be heavily involved with requirements generation and must stay connected to the development process because compromises and adjustments will inevitably have to be made, interpretations will be required, and requirements can change.

¹The reader is reminded that this document does not reflect any changes that the Air Force has made in response to the recommendations contained in the previously published summary of this study, R-1855-PR; thus, some of the organizations referred to may no longer exist.

However, the user cannot be allowed to exploit his interaction by injecting a continuous stream of changing requirements as he conceives new ideas or refines his perception of need. While he must be deeply involved in and even control the initial requirements phase of the development cycle, during other phases, especially design-development, he must interact only in a carefully controlled way under strict management control. Even with an incremental approach that accommodates evolutionary requirements, each cycle of progress must be completed before expanded or revised requirements initiate the next cycle.

This separation of responsibilities and careful control of interactions must be exercised not only at the Air Staff but wherever it might occur throughout the Air Force. The user must be given the resources and responsibility to define the systems that will increase his effectiveness. "Separation" does not imply different geographical locations, but rather assurance of sufficient organizational distance to guarantee clearly delineated life-cycle development responsibilities and productive confrontation--an essential ingredient of a strengthened development channel. To assure that the requirements generation process adequately reflects the genuine need of the user and also technical feasibility, it is necessary to create or to strengthen, as the case may be, the requirements generation process within the Air Staff and elsewhere; this must be done wherever a functional application area exists that requires a computer-based system. In some parts of the Air Staff, adequate groups already exist, but in some functional areas (particularly those that cut across commands) and for systems that cut across functional areas, there is no requirements focus.

Strengthening the Development Channel Within AFSC

To further strengthen the development channel, we recommend that the part of the Data Systems Design Center that responds to user-stated requirements and creates appropriate management support systems for either operating bases or major commands become the nucleus for a new division of AFSC, which we will refer to as the Data Systems Division. This newly formed Data Systems Division should have both an in-house design and development capability to do such tasks as can best

be done by an organic staff, and a project-oriented component through which it could contract and conduct outside acquisition as necessary. This action would give AFSC a capability to implement management support systems.

It is important to note that the total-life-cycle involvement of the proposed Data Systems Division is significantly different from the way normal AFSC product divisions operate. In particular, some systems created by the Data Systems Division would never transition to AFLC or a major command for product improvement or product enhancement; the Data Systems Division itself would retain responsibility for their support.² This new and quite different responsibility for AFSC is a vital element of the recommended Data Systems Division concept.

In order for Hq AFSC to cope with the new responsibilities of management support system development and support within the Data Systems Division, it must be strengthened organizationally and technically. This could be accomplished by the establishment of a new directorate under either DCS/Systems or DCS/Development Plans, or Hq AFSC might elect to assign the new responsibilities within the current organizational structure.

Establishment of Centers of Expertise

The computer systems development channel should be further strengthened by establishing a "center of expertise" at each of the product divisions of AFSC or wherever else is indicated, e.g., a MAJCOM heavily involved in software development or in software support. This recommendation derives partly from the observation that computer expertise must be expanded and strengthened throughout AFSC to support system development program managers and project offices and to assist in the technical aspects of management decisions and partly from noting that organizations such as Air Logistics Centers or MAJCOMs have similar needs. The recommendation recognizes that while the steps, mechanisms,

²This recommendation is applicable to those systems that are expected to require significant product improvement throughout their operational life.

and processes involved in creating a computer-based system are independent of the functional area of application, the detailed technical elements of particular systems are sometimes quite different; hence local centers of expertise are more appropriate than an Air-Force-wide one. This is also consistent with our belief that the relatively decentralized development mode the Air Force has been using for computer systems remains the most effective way to carry out the wide range of computer-based development projects the Air Force requires to accomplish its mission.

Strengthening the Role of Single Managers in MAJCOMs

We recommend that within each major command the Single Manager, whose position was originally created to be a focal point only for management support systems, now should have two functions: He should be responsible for all computer systems in the command that are not subsystems of weapons, and he should oversee, control, and manage the development process--especially of software--within his command. The Single Manager should have no formal authority relationship to the recommended Assistant Chief of Staff for Computer Resources in the Air Staff, but there should be a functional relation in the same sense that the operations people of a command relate to DCS/Plans and Operations, or the logistics people of a command relate to DCS/Systems and Logistics.

Chapter 6

IMPLEMENTATION DETAILS AND SUPPORTING RATIONALE

Having summarized the Project RAND recommendations for improving computer resources management in the Air Force, we shall now provide additional information on the detailed implementation of them, together with background and supporting rationale.

It is obvious that many different organizational arrangements could be devised, even within the spirit of the broad recommendations outlined in the previous chapter. In a few cases (the organizational location of the Air Staff focal point is the most notable example), we will discuss some of the alternatives and explain our reasons for picking one particular solution. In most cases, however, we describe only the arrangement that seemed most appealing to us, recognizing that others might work equally well.

FOCALIZATION OF MANAGEMENT AT THE AIR STAFF

The recommendation that a focal point for computer resources management be established within the Air Staff is a direct consequence of our recognition that the acquisition of computer systems (including software) is a development process to be conducted from a system- and life-cycle-oriented point of view, and that this process is equally applicable to all types of computer systems acquired by the Air Force. It is critically important that a systematic process (such as that described in Chap. 4) be followed in the development of all types of computer systems, wherever in the Air Force they are undertaken. Since the present situation is characterized by multiple and overlapping organizations and sets of regulations for acquiring computer systems, it is necessary to develop new uniform policy and uniform management methodology that will be applicable to computer resources in every functional element of the Air Force. Such uniform policy and management methodology can come only from a single source that has authority as well as technical and management expertise.

The recommended focalization is essentially an extension and expansion of the scope of the Single Manager concept that presently

applies to computer resources acquired under the AFR-300 series regulations. We have enlarged the concept to cover all Air Force computer resources management, and we have moved the Air-Force-wide focal point to a position of higher authority.

Purposes

The first and foremost purpose of establishing a focal point for computer resources management in the Air Force is to set up an organization that can assure the disciplined application of management reviews and controls throughout the development life cycle of a computer system and assure confrontation at appropriate places in the development process. More specifically, the purposes of focalization are to:

- Develop, and assure the application of, uniform policy and management controls to all Air Force computer system acquisitions.
- Assure the presence of independent technical expertise at Air Staff decision points.
- Provide improved visibility of computer systems to the highest levels of Air Force management.
- Assure the exploration of technical alternatives so that development risks can be identified very early.
- Minimize the risk that the Air Force will apply computer resources to unproductive goals. From the vantage point of authority, visibility, and technical expertise, the focal point can assure that the recommended confrontation steps in the development process are carried out.
- Assure that technically adequate reviews are conducted at appropriate times during the development process.
- Minimize counterproductive conflicts between different functional areas, by the creation of a single authority to adjudicate such disputes.
- Provide a single point of contact to facilitate interaction among computer users in the Air Force, the AFSC developers, and the non-Air-Force computer community.

- Identify and manage interface problems among systems. Interconnection among the Air Force's computer systems, as well as with the computer systems of other services, is increasing. The successful implementation of such automated data interfaces requires special management attention and control.

Additional objectives of focalization are less directly related to management of the computer resource development process but are no less important for improving Air Force ability to apply computer resources. They are to:

- Reduce the need for scarce resources. Consolidation of the elements that presently deal with computer resources in the several functional areas into a single entity of the Air Staff will reduce overlap in activities and free personnel with computer expertise for other, more important uses of their talents.
- Provide a strong, positive impact on the computer career field in the Air Force by emphasizing the importance of this area, establishing the focal point as a sponsor of the 5LXX field, and increasing the stature of the top positions in this field.
- Provide a high-level input into the Air Force training and education process to ensure that the computer career field keeps up with the state of the computer art. The focal point will provide inputs directly to the Air Training Command, the Air Force Institute of Technology, and other training and educational elements in the Air Force and the DoD.
- Provide for the generation and maintenance of a "corporate memory" on the development of computer resources. The focal point will learn of and maintain a data base of both the development successes and the difficulties. This information can then be disseminated throughout the Air Force.

One other key characteristic of the recommended focal point must be emphasized: To develop a computer-based system successfully, the requirements for the system must not only reflect the users' needs, they must also be technically feasible and within the state of the art. Thus, the personnel involved in the technology/requirements validation task must have qualifications in both functional and technical areas. We urge a procedure that encourages confrontation between those who are requesting the system and those who will have to develop it. One purpose of the focal point is to assure that such a productive confrontation takes place. It is therefore important that the focal point not serve as an initiator or participant in generating users' requirements, nor should the focal point ever be an advocate of any proposed computer system until after a thorough analysis has been performed, all requirements have been validated, and cost tradeoffs have been completed.

Implementation of the Focal Point

It is our recommendation that the focal point within the Air Staff be an Assistant Chief of Staff for Computer Resources Management (ACS-CRM). This choice was selected in preference to other possible alternatives for the following three reasons:

1. The focal point must have enough stature to enable productive confrontation with the existing power structure and to institute the recommended changes.

As stated earlier, one function of the focal point is to assure that review and confrontation are applied to every computer system development proposal. It is conceivable that participation in such reviews may bring the focal point into conflict with advocates at very high levels in the Air Force. In such situations, the ACS-CRM may be unable to stand by his views unless he himself has a high level of authority. Along the same line of reasoning, and noting that the recommendations involve major changes in the present way of acquiring computer systems (and these may not be entirely popular throughout the Air Force),

we are convinced that the focal point should start in the strongest possible position of authority. Such a position is needed to generate and implement new regulations and procedures. Also, interaction with the DoD and the Congress on matters dealing with the Brooks Bill requirements call for a focal point with an authoritative stature.

2. The focal point must have access to top-level Air Force management and must participate in the formal and informal decisionmaking structure within the Air Staff.

First, the decisions from the focal point must be recognized as representing top-level Air Force management. Second, because of the importance of computer systems in the present-day Air Force, matters dealing with their management and acquisition are important in many highest-level Air Force decisions. Furthermore, decisions in non-computer areas may have important impacts on the ability of the Air Force to provide computational resources. For these reasons, the proposed ACS-CRM should be a participant in the normal top-level decision process in the Air Force, including representation on the Air Force Board and Air Force Council. He cannot perform his function if he is regarded as only a resource person to be called in when specific information is needed on computer systems.

3. The position of the focal point in the Air Force organization must transmit an unambiguous and strong signal of intent to cope with the computer problem.

Creation of an Assistant Chief of Staff as the focal point would emphasize Air Force resolve to cope with the computer technology problem and thus would improve the acceptance of his new policies and management controls. The action of establishing a focal point with high authority and stature is an unmistakable message that the Air Force is serious about solving its computer problems.

To enhance his acceptance and credibility within the Air Force, the ACS-CRM should have personal competence in the computer field but

he need not necessarily be a computer specialist. Competence with computers will enable him to judge the advice given to him, to feel comfortable with the technology, and to discuss computer matters during his participation in the Air Force decisionmaking process. He will need a supporting staff of technical experts, as well as persons familiar with the special problems of the three main types of computer systems in the Air Force--command and control, embedded, and management support.

It is important for the success of the focal-point mission that the establishment of an office as powerful as that of an Assistant Chief of Staff have the wholehearted concurrence of top-level management. The new focal point must have the full cooperation of the management structure if they are to collectively solve the Air Force computer problem.

It should be noted that the establishment of an Assistant Chief of Staff to guide an important new technology or to develop a new capability has precedents in the Air Force. For example, in 1954 an Assistant Chief of Staff for Guided Missiles was established to enhance the development of the U.S. ICBM capability; the office was eliminated six years later when management of the ICBM program was well in hand.

Organizational Alternatives for the Focal Point

Our recommendation that an Assistant Chief of Staff be established as the focal point for computer resources management was decided upon only after considering several other alternatives:

- A Special Assistant to the Chief of Staff on computer technology matters.
- A Director of Computer Resources Management in the office of an existing Deputy Chief of Staff.
- A separate Operating Agency of the Air Force.

None of these alternatives provide the stature, authority, credibility, or strong message that are essential to a focal point. In none of these positions would the focal point be assured of involvement in

the top-level decisionmaking process or of being able to take a stand in high-level confrontations. There are also the following considerations.

Historically, the function of a Special Assistant to the Chief of Staff has been primarily to keep the Chief of Staff informed about the progress of a particular development project. This appointment is not recognized as a position on the Air Staff, and such posts have been created in the past for ad hoc purposes only, such as for AWACS or the B-1. Substantive management questions have always been handled by a Program Manager or the appropriate Deputy Chief of Staff. Special Assistants report directly to the Chief of Staff and, in the present structure, do not consult with or participate with Air Force management groups except through him. Thus, the role of a Special Assistant would not be appropriate for handling the Air-Force-wide problem of computer resources management.

The establishment of the focal point as a Directorate under a Deputy Chief of Staff was given careful consideration, but this alternative too has serious shortcomings. Some problems stem from the strong functional orientation of the present DCS structure and the fact that an Air Staff Directorate is inevitably identified with, and strongly influenced and constrained by, the parent DCS and its associated functional organization. Such a Directorate could not avoid becoming an advocate for the functional area represented by the DCS. This is contrary to our strong conviction and recommendation that the focal point must not be an advocate for any particular system but instead must concentrate on effective management of all computer matters, on adherence to a disciplined and systematic development process, and on effective exploitation of computer technology wherever such exploitation can be shown to produce important benefits.

Another problem with putting the focal point at Air Staff Directorate level is that it would be difficult for any Directorate to exercise appropriate review authority over a computer system throughout the entire life cycle of that system. Finally, a Directorate-level focal point could not exercise the authority or transmit the strong message of intent that we believe is necessary to achieve the desired reforms in computer resources management.

If the focal point were implemented as a Separate Operating Agency of the Air Force (similar to the present Data Automation Agency), it would provide neither the required strong message nor an adequate position of authority. It would be regarded as some sort of data-gathering agency rather than as an entity to be contended with when proposing a computer system development.

At the present time, it is essential that the focal point have a strong starting position. In a few years, the situation may have changed and the special attention of an ACS for computer resources management may no longer be necessary. At such time, it would indeed be appropriate to rethink the organizational position of the focal point; some of the alternatives that we have rejected at this time may then be more suitable.

Duties and Tasks of the ACS-CRM

Having presented the rationale for the establishment of an Assistant Chief of Staff for Computer Resources Management (ACS-CRM), we next outline our recommendations for the tasks that his office should perform.

The major task of the ACS-CRM would be to coordinate and review all major Air Force computer system acquisitions, both in-house and contracted. In part, he would do this through a process of confrontation--between himself and other relevant parties, or between appropriate other parties with the ACS as a referee and organizer. He would provide independent judgment and expertise in reviewing Air Force development projects that deal in whole or in part with computer resources, with special emphasis placed on the development of software. He would, therefore, have responsibility for evaluation, modification, and consolidation of existing regulations in the AFR-300, 800, and 100 series, or for developing new regulations that would establish an orderly development process for computer systems and reflect a uniform approach to life-cycle management. Appropriate changes in the development organizations are discussed in the following subsection.

The ACS-CRM would have the task of assuring effective and efficient use of all computer resources in order to derive maximum advantage from available technology, while assuring that the requirements established by the Brooks Bill and DoD regulations are applied to the systems covered

by them. To increase computer expertise, he would be responsible for assuring effective computer-related career management, education, and training in the Air Force. Furthermore, he should be responsible for ongoing tasks in the areas of computer system acquisition, maintaining cost visibility, and fostering information exchanges.

In the area of computer systems acquisition, the ACS-CRM would be expected to:

- Evaluate, modify, and/or prepare policies, procedures, standards, guidelines, and regulations governing the acquisition of major computer systems (in-house and under contract). Such material should provide for the appropriate technical and management reviews in the various stages of the acquisition process.
- Evaluate, modify, and/or prepare standards and guidelines for the documentation required at each stage of the acquisition process, including the documentation needed to effectively utilize, operate, and support a computer system throughout its life cycle.
- Monitor the acquisition of computer systems by participating in and conducting reviews and exercising approval authority for certain steps in the process.

The task of monitoring the acquisition process can be related to the life-cycle development model discussed in Chap. 4 and depicted in Fig. 1.

During the mission requirements formulation phase, the role of the ACS-CRM would be primarily a supportive one; he would assure that considerations related to computer technology are addressed in the requirements generation process whenever they become significant. He would not be expected to participate in any nontechnical requirements-related confrontation, but there would be grey areas where he could raise technically oriented issues--for example, must a proposed system be on-line or would remote job entry be sufficient?

In the next development phase, technology/requirements validation, the ACS-CRM role would become much more active. Here, the full technological and management expertise of his office would be brought to bear

on technological feasibility, anticipated system performance, cost, and schedule. He would determine whether systems with similar capabilities are already in existence or being developed elsewhere in the Air Force. The ACS-CRM should not himself perform the requirements validation task but should assure that it has been completed and should lend assistance to the process as required. Passage from the technology/requirements validation phase to design and development should not be permitted without ACS-CRM approval. This would be a qualified approval in that it would relate only to the technical and management process for computer-based systems and would not imply a value judgment regarding, for example, the validity of the functional requirement. Such approval at this stage would imply ACS acceptance that the functional system specifications were generated by an appropriate process and are at the required level of detail for the next steps to begin. This is one of the most important tasks of the ACS-CRM during the technology/requirements validation phase.

The active role of the ACS-CRM would continue in the design and development phase. Before full development could begin, he should have expressly reviewed and be satisfied with the preliminary as well as the detailed design at the system level. Some reasons for disapproval might be: (1) insufficient consideration of design alternatives, (2) incomplete design, or (3) insufficient consideration and planning for testing, support, enhancement, etc. Such review consideration is especially applicable to computer software.

In the testing phase, the ACS-CRM should review the testing criteria and plans for thoroughness, effectiveness, and relevance, and he should assure that all planned tests are conducted. The ACS-CRM would have to be satisfied that the test results adequately demonstrate satisfactory implementation. Subsequently, his role would revert to that of maintaining awareness and routine enhancement activities and of the operational use and support of the system, and monitoring and reporting the costs that are incurred. He would have the prerogative, however, to review at any time the operational use and support activities of a system to reassess its effectiveness and efficiency. We expect that in such reviews, the ACS-CRM would establish and chair the technical review

groups involved and would apply the monitoring and review process to Air Force in-house development projects as well as to those that are contracted.

In his role as a point of contact for computer matters, the ACS-CRM can be expected to:

- Provide guidance on efficient utilization of computer resources and act as a central source of information and expertise on computer matters.

The ACS-CRM office should act as a repository of information and records on the total computer resources in the Air Force. In particular, the ACS-CRM should provide cost visibility by monitoring and recording all costs associated with computer system acquisition, development, and operation. Furthermore, he should evaluate, modify, and/or prepare policies, procedures, standards, and guidelines to facilitate the cost-monitoring function and assure that appropriate cost breakdowns are provided. It is particularly important to identify the various components of software development and support costs. With respect to demands from other agencies, such as the DoD and the GSA, the ACS-CRM should act as the office of primary responsibility for the preparation and submission of cost records, computer equipment inventory and utilization data, and other such information that may be required.

As a source of expertise, the office of the ACS-CRM should maintain a constant assessment of the state of the art of computer technology and closely associated technologies as they relate to Air Force needs and should initiate and manage studies on current and possible future trends of these technologies. Furthermore, it should be the task of the ACS-CRM to:

- Provide a knowledgeable and visible point of contact that can encourage and expedite interactions within the Air Force computer community and between the Air Force and the computer community at large.

In this task, the ACS-CRM could provide short-term support to users who need advice or assistance. A strong professional relationship should be established with the Air Force ADP Single Managers at major commands and elsewhere. As discussed below, the Single Manager organization must also be strengthened and provided with a degree of autonomy such that, within their organizations, Single Managers could mirror the purposes and roles that the ACS-CRM would provide on an Air-Force-wide basis.

It would also be within the purview of the ACS-CRM to foster beneficial exchanges of information between functional areas on computer technology, its applications, and the management of computer development projects. The ACS-CRM should examine all functional requirements entailing uses of data automation and computer resources and identify those which may lend themselves to common development efforts. In these deliberations, however, the ACS-CRM must avoid any advocacy role for specific systems.

Another important task for the ACS-CRM would be the sponsorship in the Air Staff of the computer career field. In particular, he should work in conjunction with the DCS/Personnel and the Air Training Command to:

- Prepare, evaluate, and recommend policies, procedures, and guidelines that promote the development and retention of technically competent specialists in the Air Force, that promote education and training programs in computer technology, and that promote computer-related management training and education to Air Force military and civilian personnel.

The development of an attractive career path is essential to the effective utilization of computer systems in the Air Force. The ACS-CRM should assign a high priority to this task.

While the personnel matter was not considered in enough depth during the present study to enable us to make suggestions or recommendations, some related issues are discussed in App. C.

Brooks Bill Requirements

While it is essential that the development of all types of computer systems in the Air Force be uniformly managed, there is no need to extend the 300-series requirements to those systems presently exempted (i.e., computer systems integral to or embedded in weapon systems). The reporting chain for the systems subject to the Brooks Bill presently runs from the Assistant Secretary of Defense (Comptroller) to the Assistant Secretary of the Air Force for Financial Management, and through the Chief of Staff of the Air Force to the Director of Data Automation (AF/ACD).¹ For major systems that are subject to these requirements and require resources above certain thresholds (see AFR 300-2), the Assistant Secretary of the Air Force for Financial Management (SAF/FM) is the final approving authority.

With the establishment of the ACS-CRM, we expect that the Brooks-Bill-related duties and the approval authority previously delegated to the AF/ACD would be transferred to the ACS-CRM. While it may be desirable to provide more authority to the ACS-CRM by increasing the thresholds at which SAF/FM approval is required, there seems to be no need for any restructuring of the management chain between the ACS-CRM and the SAF/FM.

STRENGTHENING THE DEVELOPMENT CHANNEL WITHIN THE AIR STAFF

The acquisition of any computer-based system should be treated and managed as a development process, independent of functional area. Since AF/RD is the part of the Air Staff that is responsible for the development process, it seems appropriate that it should have responsibility for all major computer developments. We therefore recommend that AF/RD add management support systems to its present areas of responsibility, computers in weapon systems and in command and control systems. Thus, both Data Automation Requirements (DARs) and Required Operational Capability statements (ROCs) would flow into AF/RD. At present, ROCs flow into AF/RD, but DARs flow into the Directorate of Data Automation (AF/ACD).

¹The reader is reminded that this document does not reflect any changes that the Air Force has made in response to the recommendations contained in the previously published summary of this study, R-1855-PR; thus, some of the organizations referred to may no longer exist.

A DAR is usually initiated by a user at a major command; this is also the current procedure with ROCs. Since computer expertise exists at the MAJCOM under the Single Manager, all DARs should go through him. Thus, the MAJCOM Single Manager should perform a role analogous to that of the ACS-CRM. For DARs that can be fulfilled at MAJCOM level,² the Single Manager would have full responsibility for acquisition and support of the computer system. For larger systems that must have Air Staff or DoD approval, the MAJCOM Single Manager would assist the user in refining the DAR or ROC for submittal. The process in the case of ROCs has historically included an interaction with Systems Command for technical feasibility; the same procedure should now be followed in the case of DARs. This would assure that initial consideration has been given to the technology/requirements validation phase of the life cycle. Interaction with Air Staff and mission analysis people would also assure that the operational requirements and the potential value of the proposed system to the Air Force would be evaluated.

Validation of DARs and Computer-Oriented ROCs Within AF/RD

The proposed management-control and approval-process flow for DARs and ROCs is shown in Fig. 2. The procedure is based on the existing management structure within AF/RD, supplemented by the changes necessary to handle DARs in the same manner. Figure 2 replicates the general process shown in Fig. 1, but with additional details such as:

- Interaction points for the ACS-CRM.
- The inputs expected from the ACS-CRM.
- The inputs expected from organizations at various points.
- The actions required of various organizations.

DARs or ROCs that must be reviewed by the Air Staff would be submitted to the Management Office (AF/RDQLM) of AF/RD's Directorate of Operational Requirements and Development Plans (AF/RDQ). (At present, the ROCs that deal with reconnaissance and electronics warfare or space

²Financial thresholds would still be specified for both ROCs and DARs, but they need not be the same for each, nor are the present levels necessarily proper.

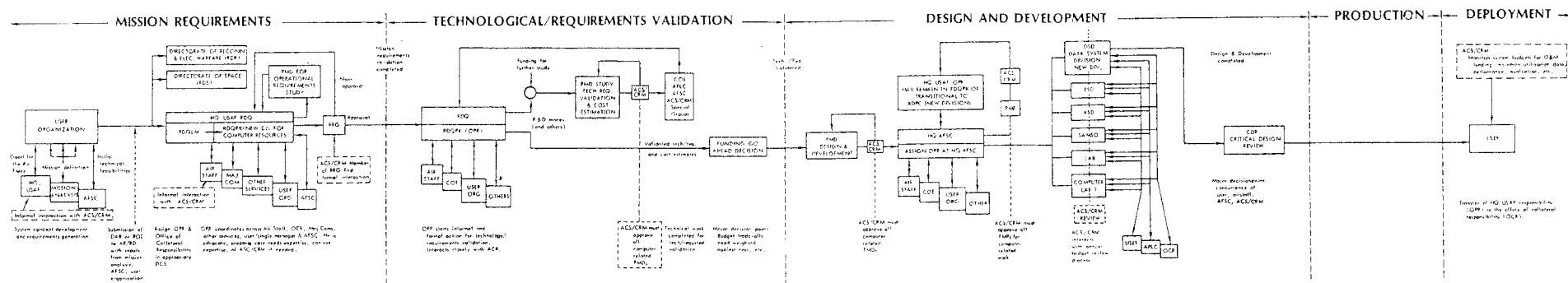


Fig. 2 — Proposed approval process flow for DARs and computer-oriented ROCs

are submitted to the corresponding directorates, AF/RDR or AF/RDS.) There is no apparent need to create a special directorate for computers.

Past procedures have called for AF/RDQLM to assign ROCs to an action office (OPR) within one of the Deputy Directorates in AF/RDQ: General Purpose Forces (AF/RDQR), Strategic Forces (AF/RDQS), or Support (AF/RDQP). The OPR fulfills the advocacy role, coordinates the ROC throughout the Air Staff, and retains responsibility for it until it is validated. Part of the coordination task is to expose the mission requirements to the Air Staff and others for scrutiny. If necessary, study directives should be issued to substantiate operational requirements prior to submitting the ROC to the Requirements Review Group. This is indicated by feedback paths in Fig. 2.

We recommend that DARs should follow the same general procedure as ROCs; however, considerable computer-related expertise is required to review and evaluate DARs. Consequently, the AF/RDQ staff would have to be augmented with people who have such expertise. An organization would have to be established within AF/RDQ to serve as a central contact point for computer-based systems. Such an organization is shown as AF/RDQPK in Fig. 2. With these changes, development responsibility for all computer-based systems would reside within AF/RD. Thus, AF/RD, together with the functional areas involved, would become an organizational advocate for every Air Force computer-based system, including embedded systems. A nucleus for this new division in AF/RDQ could come from parts of the Management Support Systems Branch (AF/ACDCB) and the Mission System Branch (AF/ACDCA) of AF/ACD. Additional strengthening may well be needed in the areas of command and control and weapon systems computers.

AF/RDQPK should have responsibility for bringing DARs or computer-involved ROCs to the point at which they are ready for presentation to the Requirements Review Group (RRG). It would be completely inappropriate for the proposed ACS-CRM to handle the OPR function because the OPR must take an advocacy role; advocacy for any specific system must not be a part of the charter of the ACS-CRM.

The current RRG membership includes the Directors of the AF/RDQ (Chairman), AF/RDP, AF/RDR, AF/PRP, AF/XOO, and AF/LGY. We recommend that the ACS-CRM participate as an additional member of the RRG. The

RRG review of the requirements would be the first time the ACS-CRM would officially interact with the development of a proposed system. This interaction is shown in Fig. 2 as a prerequisite to passing to the technological/requirements validation phase.

After approval by the RRG, the mission requirements validation function in the development process is completed. If the ACS-CRM or other members of the RRG have concerns about the mission requirements, they should express and adjudicate them at this time.

The next step, technology/requirements validation (phase B in Fig. 1 of Chap. 4), should be the responsibility of the OPR in AF/RDQPK. Either the OPR should have sufficient technical expertise available to perform this function, or it should issue a PMD to a center of expertise requiring analysis, tradeoffs, and other activities to see that the phase is properly performed and that confrontation takes place.

As discussed earlier, the proposed ACS-CRM would establish a review group for ascertaining that the functions of technology/requirements validation are properly performed; it would then be the responsibility of AF/RDQPK to interact with the review group. This can be assured by requiring that ACS-CRM approve all PMDs related to DARs or to ROCs that have a computer component. In cases where a PMD requires further studies, the ACS-CRM could ascertain whether or not sufficient direction and goals have been specified. This situation is covered in Fig. 2 by including the ACS-CRM in the feedback loops in the technology/requirements validation phase.

After the mission requirements have been approved, the technology/requirements validation process completed, and the project funded, AF/RDQ sometimes transitions control to the Directorate of Development and Acquisition (AF/RDP), i.e., the OPR switches to AF/RDP, where development programs are normally monitored within the Air Staff. If this procedure is followed for computer-based systems, it will be necessary also to augment the staff of AF/RDP where PEMs normally reside. Alternatively, it may be beneficial to augment AF/RDP by establishing a new computer systems division within AF/RDP--call it AF/RDPK. The placement of this new division in the management flow, under Design and Development, is shown in Fig. 2. The division would provide software management and data-automation expertise as needed to monitor computer

development projects. A nucleus for such a group could come from parts of the AF/ACDCB branch of AF/ACD, but it would have to be supplemented with expertise for embedded and command and control systems.

While this overall approach may appear to give computer technology a very special treatment, we would like to reemphasize that computer resources have become so vital to the Air Force that special attention and special organizational arrangements are needed to deal with them.

Additional Air Staff Changes

The need to strengthen the requirements generation process for computer-related systems has been emphasized throughout this report. The traditional functional areas (e.g., operations, logistics, finance, manpower) already have within the Air Staff appropriate offices of primary responsibility which represent the user in terms of the orderly development of formal requirements documentation for desired new capabilities.³ However, even in these traditional areas, the capabilities required to develop system requirements do not always exist. Thus, these functional areas need to acquire the skills to complement their present responsibilities. Moreover, in some cases it will be necessary to develop new organizations in or associated with the Air Staff to manage the generation of requirements for a capability that either will cut across functional areas or command lines or will serve functional areas that now do not have adequate representation at the Air Staff level.

We have argued earlier that organizational separation is essential between those who are advocating new systems and the ACS-CRM. Thus, any Air Staff element focused on generating system requirements or advocating a particular capability should not be a part of the proposed ACS-CRM.

Reassignment of AF/ACD Resources

One effect of the foregoing recommendations would be reassignment

³It is worth noting here that such Air Staff offices should be concerned only with the definition of requirements and should avoid specifying the desired hardware solution or computer system design details. Unfortunately, that separation of responsibilities is not always observed.

of the responsibilities now centralized in AF/ACD. The Director of AF/ACD presently commands the Data Automation Agency, which consists of three components: the Federal Simulation Center, the Data Services Center, and the Data Systems Design Center.

The Federal Simulation Center is a governmentwide organization for which the Air Force is executive agent. It provides computer-simulation expertise for the analysis of user requirements at the detail level and relates them to specific computer hardware configurations and software capabilities. In general, this group assists in the same kinds of studies and decisionmaking that the proposed ACS-CRM would be expected to participate in. Though the focus of the Federal Simulation Center is the entire U.S. Government, not just the Air Force, as long as the Air Force is the executive agent it seems appropriate that the Federal Simulation Center be continued as part of the office of the proposed ACS-CRM. The Data Services Center, in the basement of the Pentagon, supports the entire Air Staff. This organization might also become a part of the office of the proposed ACS-CRM, in which case the latter could also function as the Single Manager of computer matters for the Air Staff; or it might equally well become a part of the Hq USAF Command that now manages all other administrative and support facilities for the Air Staff. The Data Systems Design Center requires a more extensive discussion.

The Air Force has institutionalized in the Data Systems Design Center (DSDC) all development efforts and ongoing support activities for standard functional software packages that run both at base level and in the major commands. Situated at Gunter Air Force Station, the DSDC currently employs many hundreds of people who help to specify user requirements and who design, implement, and operationally support base-level management support systems. It is involved in a special way with Air Force computer users because (1) it delivers capability to the user on an incremental basis, (2) it has responsibility for managing much of the inventory of computers on which all the standard software packages run, and (3) it is expected to provide operational support for functional area users. The DSDC should be realigned organizationally but not necessarily relocated geographically.

The design of computer-oriented systems requires in-depth functional knowledge as well as knowledge about computers; providing such a mix has been a problem for every organization that has tried this kind of activity. Many solutions have been tried over the years, and through experience much has been learned about some procedures that work better than others. Ideally, one person would possess all the knowledge needed to design the system from both the user's side and the technical side. Unfortunately, that cannot work except for the smallest systems. As system sizes have increased, it has become clear that no one person could possibly know everything needed to design them, so some kind of team approach is unavoidable.

During the early 1960s, a number of organizations were formed to develop data-automation systems in support of various Air Force functions. Among them were the Supply Systems Design Office (to support the Air Force Directorate of Supply and Services) and the Maintenance System Design Office (to support the Directorate of Maintenance Engineering). Each was organized to develop ADP systems in support of a particular functional area. At a later date, the Data Automation Design Office was established under the Hq USAF Directorate of Data Automation to support all functional areas. Since these several early organizations were located near one another in the Washington, D.C., area and all were supported in one way or another by the Data Automation Design Office, they were collectively referred to as the Data Automation Design Center.

In 1967, the various separate but related groups were combined into a formally established Data Systems Design Center. While the new DSDC retained some internal alignments corresponding to the various functional areas for which work was being performed, all personnel reported to the DSDC commander who, in turn, reported to the Air Force Chief of Staff through the Director of Data Automation. Thus, the present DSDC is essentially a vertically oriented organization; for a particular functional area (logistics, for example), there exists within DSDC a team comprising functional area people, systems design people, and application programmers. Another section of DSDC contains computer scientists who deal with more generalized problems of computer hardware configurations, software operating systems, etc.

Theoretically, requirements for a particular system are generated not by the DSDC but by a functional user. However, the user sometimes does not have within his own staff the computer expertise necessary to formulate detailed functional specifications for a new system and thus tends to generate only a general statement of desired capability. Such a statement is then formalized as a DAR and, when approved, is submitted to DSDC as a firm system development requirement. It is therefore necessary for the functional-area people in DSDC to assist in the preparation of detailed functional specifications, a task that usually involves substantial modifications to the original statement of desired capability. While the end user is consulted during that process, prime responsibility is centered in the DSDC functional staff who work for the DSDC Commander, as do the systems designers and application programmers. The difficulty with such an arrangement is that the user who originally states the need for a proposed system via the DAR has lost control over a vital phase of the requirements process, i.e., the translation of general performance requirements into detailed functional specifications. Once we agree that most systems cannot be developed, from concept to operational system, by one individual or organization, the issue of how to separate the various functions emerges. The current DSDC organization is one arrangement. When we recommend that the DSDC should be realigned organizationally, we are suggesting another alternative. The major point which should not get lost in the details of a particular alternative is that we perceive a significant problem with the current arrangement of responsibilities.

This problem is not unique to the DSDC. Every organization involved in the development of large computer systems eventually recognizes the need for separation of user-requirements generation from system development. Where this problem has been successfully dealt with, organizational changes have been made that differentiate the roles of user and developer rather than confounding them.

The interaction between the requirements generation process and the development process must be carefully controlled. We have considered two possibilities for strengthening the part of DSDC that helps generate user requirements for the various functional areas: The Logistics Management Center (LMC), plus various special operating agencies such as

the Military Personnel Center and the Finance Center, could represent the user in the preparation of detailed functional specifications for various base-level activities. Or, and this would be more difficult to implement, the Air Force could create a Base Management Center that would address itself to all computer-based functional-area applications at base level. We have not taken a specific position on this point. In either arrangement, the user-oriented organizations should work closely with computer systems development experts in DSDC but should retain line management ties to the functional user.

The net effect of the recommended shift in responsibilities and organizational realignment of certain portions of the present DSDC staff would be to change the nature of a DAR and to change the point in the development cycle at which a final approved DAR would be issued. Under the proposed arrangement, the functional user of the proposed system would retain complete control over and responsibility for the requirements-generation process up to the point at which a detailed functional specification for the system has been completed and approved and its associated cost and schedule projections have been acknowledged as reasonable by the organization charged with full-scale development of the system. The final DAR, and the associated Program Management Directive, would therefore be founded on much more complete information than is normally provided under the present arrangement; the design specifications would be sufficiently complete that the development organization (the DSDC) could proceed directly with the detail design.

The part of the DSDC that responds to user-stated requirements and creates appropriate management support systems for either bases or major commands (i.e., the design and development part) could become the nucleus of a new division of AFSC, called perhaps the Data Systems Division. We feel strongly that such a new organization should not become part of the Electronic Systems Division, for several reasons: First, ESD is a weapon-system-oriented organization with little or no experience in development of management support systems. Second, and very important, the way of life in the weapon-system development process to which ESD is accustomed is different from the total-life involvement of the DSDC.

The proposed Data Systems Division should have both an in-house design and development capability to accommodate tasks that are appropriately done by an organic staff, and a project-oriented component through which it could contract and conduct outside acquisitions as necessary. The Division could remain at Gunter Air Station; the functional requirements people and the design/development specialists could still be colocated, but appropriate organizational controls would be needed to assure a proper interface and interaction between the two.

The Data Systems Division would deal only with management support systems; its proposed structure is consistent with that suggested elsewhere for Air Staff reorganization. It is important to note that the total-life involvement for the proposed Data System Division is significantly different from the way other AFSC product divisions operate. In particular, a system created by the Data Systems Division would normally never transition; it would never be turned over to AFLC or to a major command for software support. Rather, the Division would retain responsibility for it throughout its life. AFSC would have to be willing to accept this new and quite different responsibility. The user organization would, as discussed previously, retain management control and responsibility for initiating and specifying product improvement and product enhancement.

STRENGTHENING THE DEVELOPMENT CHANNEL WITHIN AFSC

Changes are needed in two areas of the Hq AFSC organizational structure to enable it to fulfill the responsibilities suggested by these recommendations. First, since AFSC would be accepting responsibility for the development of management support systems, it would have to acquire capability in this new area. Second, management of the computational support centers distributed throughout the Command must be strengthened; this matter will be discussed later in relation to the Single Manager question.

When a development project is approved, a Program Management Directive is forwarded by AF/RD to AFSC. An OPR is assigned at Hq AFSC and a product division or laboratory is designated to conduct the project. An OPR is currently assigned to one of three places at Hq AFSC, depend-

ing on the nature of the program. If the directive specifies research or exploratory development (6.1 or 6.2 funds), the OPR is usually within the Directorate of Science and Technology; if it specifies advanced development (6.3 funds) the OPR is usually within DCS/Development Plans; and if the program is engineering development (6.4 funds), the OPR is usually within DCS/Systems. There is currently no management structure or expertise in these three organizations that can cope with the development of management support systems. Consequently, AFSC must strengthen at least one of them to assure that the development of management support systems is managed in the same way as other developments.

In Chap. 3, it was argued that despite apparent differences, all major projects, regardless of the functional area, need to be accorded the same management perspective. It was also argued that the acquisition of computer-based systems is a development process, not a production process. In this regard, software development needs at least as orderly and disciplined an approach as hardware does. If these arguments are accepted, with the corollary that AFR-300 series regulation acquisitions should be managed the same way as AFR-800 series regulation acquisitions, it follows that the OPR for all computer systems at a similar level of technological development should reside in the same office in Hq AFSC.

Centers of Expertise

The Project RAND recommendation for reorganizing computer resources management in the Air Force includes the establishment of centers of expertise (COEs) at each of the AFSC product divisions, including the proposed new Data Systems Division. The concept of a COE at each of these divisions derives from the observation that computer expertise must be strengthened throughout AFSC to support both System Project Offices and program offices and to provide technical support for management decisions. An attractive image is that of ESD/MCI, which lends experienced computer people to projects for varying periods of time. Although creating three or more COEs would require more personnel, the eventual goal of the Air Force is to have computer expertise wherever needed; thus, it is appropriate to create several COEs now.

Multiple COEs would have the collateral benefit of providing an excellent training ground for computer specialists who could acquire multifaceted, multi-functional-area experience and therefore would be qualified for later career assignments in System Project Offices or the Air Staff.

A single COE operating from Hq AFSC might appear as an outsider or intruder to a product division and therefore could be largely unwelcome. An in-division COE, being "in the family," should have better acceptance by the Systems Project Offices and program offices.

The principal functions of the COEs in the AFSC product divisions would include consultation and assistance to the SPOs, in-house software development projects, and contract managers in various phases of a computer system development life cycle. In order to stay current with the fast-moving computer field, it is absolutely essential that the COE staffs vigorously interact with professional societies, universities, and industry. In addition, the COEs must be encouraged to perform research in areas that have potential for high payoffs for the Air Force or to undertake projects of general value to the Air Force; e.g., it may be highly advantageous to become completely knowledgeable on a particular higher-order language or to maintain an inventory of available and exportable software packages. Only by encouraging the COEs to work on the technological frontiers can the Air Force be assured of identifying and utilizing the latest technology while avoiding commitment to projects that are beyond the state of the art. An effort is currently being made to build this character into the COE at the ESD/MCI; it presently devotes about 60 percent of its effort to SPO support and 40 percent to work in computer security, performance evaluation, and monitoring outside research contracts.

The COE organization and operation must avoid any actions that could be construed as advocacy for the projects it is supporting. Advocacy and confrontation are important elements in the formulation and management of a development project (as discussed elsewhere in this report), but a COE will lose its value to a project office if it slips into the role of advocating a partisan approach or solution. The COEs must vigorously and objectively provide technical advice, but it must

be presented to the supported project in a professional and impartial manner.

The other military services, especially the Army, are already operating organizations that could be regarded as COEs and that provide assistance in computer development. For example, the Army's Computer Systems Command has organized Field Assistant Support Teams (FAST) to provide help on software matters; the Computer Systems Support and Evaluation Agency provides technical assistance; and the ADP and Communications Laboratory of the Electronics Command has a Center for Tactical Computer Science (CENTACS) that conducts research and exploratory development but also provides assistance to SPOs.

Hence, there is adequate reason to believe that, given experienced personnel, a COE can perform the tasks we expect of it. To be sure, there may be some resistance from the organization being helped, especially if the assistance is unsolicited. Incentives should be established to encourage the seeking of help from the COEs, and it should be possible to convince the developing organizations and project managers that requesting or receiving help from a COE is not an admission of failure but rather is a positive and productive use of available technical skills to assure project success.

STRENGTHENING THE ROLE OF SINGLE MANAGERS IN THE MAJCOMS

We have argued the need for strong organizational separation but controlled interaction between the requirements generation process and the development process in any major command engaging in computer system development. This by no means implies that major commands should discontinue development--quite the contrary. But any such development should be subjected to the controls and disciplines described throughout this report. The organizational structure that seems appropriate for computer systems development is that of the Strategic Air Command, where the Single Manager is an Assistant Chief of Staff who can properly confront and interact with functional-area people in SAC who request computer-based capabilities. For this reason, we propose that the MAJCOM Single Manager, who was originally created to be a focal point

only for management support systems, become the focal point in each major command for all computer systems that are not weapon subsystems. He would also oversee, control, and manage the development process--especially of software--within the command. The MAJCOM Single Manager would have no formal authority relation to the proposed ACS-CRM in the Air Staff, but he would have a functional relation in the same sense that the operations people of a command relate to the Air Staff DCS/Plans and Operations or the logistics people relate to the Air Staff DCS/Systems and Logistics.

A major command that is heavily involved in computer system design and development might need its own COE reporting to the Single Manager; alternatively, technical support could be obtained from AFSC as needed. One effect of this recommendation would be that the experience and knowledge of the MAJCOM Single Manager organization would have to be expanded and its authority and responsibility broadened. Such an action would not affect the present scope and responsibilities of the command, but rather would represent a modification of organizational responsibilities within some of them.

The exact organizational arrangement and the scope of responsibility of the MAJCOM Single Manager, given these additional areas of management, will almost certainly vary from one command to another. In particular, the AFSC Single Manager needs special discussion because of the several AFSC major computing centers, each oriented toward a different functional or technical area. For example, the computing centers at the Aeronautical Systems Division, the Flight Test Center, and the Electronics System Division are much more diverse and support a much broader workload than those at the various Air Logistics Centers (ALCs). While each ALC may be responsible for the support of different weapon systems or different components, they all perform similar types of computing work. However, the different divisions in AFSC do substantially different kinds of work and demand computing services that vary in an important degree from one division to another. It seems likely, therefore, that in AFSC more authority could properly be delegated to a headquarters-level Single Manager than would be appropriate for other commands; a corresponding observation holds for division-level managers.

Obviously, various organizational arrangements could be made to manage the support of the AFSC computing centers. One that seems appropriate would be to establish the Hq AFSC Single Manager as an Assistant Chief of Staff/Information Systems. This would give the Commander of AFSC the kind of independent advisory support needed in this critical area. In addition to the advisory role, the AFSC Single Manager would have responsibility for operating the Headquarters computing center and for developing computer programs to support the Headquarters system. He would also monitor existing standard Air Force management support systems for commandwide usage and would develop new AFSC-unique management support system programs. The AFSC Single Manager should maintain a professional relationship with the division-level Single Managers and should be a focal point at Headquarters for various computation centers in the acquisition of upgraded capabilities. He should maintain an upward professional relationship with the proposed Air Staff ACS-CRM for policy and management and should function as a channel for interaction between the ACS-CRM and the AFSC computing centers regarding inventory, usage, and matters of acquiring upgraded systems.

MISCELLANEOUS

A recurring theme, not only in the Air Force but in the civilian world at large, is that communications and computers are rapidly becoming a common field. While it is true that they share an electronic capability for solid-state switching, processing, and storage, and therefore ultimately depend on a largely common technology, there seems to be no reason at this juncture to force the two into a common management structure. We suggest that the Air Force should make no decision now that would preempt joining the two areas in the future.⁴

Some of the suggestions that we have made concerning the Air Staff may have repercussions for the Office of the Secretary of the Air Force. There are some obvious issues related to career management, training, personnel management, and career progression, and Air Force regulations

⁴In fact, the Air Force, in September 1975, did assign both areas to the newly created ACS/Communication and Computer Resources Management (AF/KR).

may need to be restructured or redrawn to be consistent with the framework proposed. In the present study, we have not addressed these points in sufficient detail to support specific recommendations.

Appendix A

TERMS OF REFERENCE FOR THE RAND STUDY ON AIR FORCE
MANAGEMENT OF COMPUTER RESOURCES

1. Background. Problems associated with the development, acquisition, operations, and support of computer resources have been extensively studied from many viewpoints. Generally, these studies have concentrated on the narrower aspects, such as standardization, documentation, and improving technology. Recent discussions at the Chief of Staff level have clarified the need to take a broader look at the overall issue.

2. Scope. The study will consider the broad aspects of management, policy, and organization relevant to the computer resources for selected major weapon, command and control, telecommunication, intelligence, and management and support systems. The study will direct major attention to the top levels of management and must not focus on individual narrow areas or produce overly detailed analyses.

3. Tasks. The study will (1) examine existing Air Force policy, management, and organization at higher levels; (2) propose alternative courses of action in policy, management, and organization which address both immediate and long-range problems; (3) identify the important consequences of implementation for the alternatives.

4. Relevant Policy-Related Issues.

- a. External policy from the Congress, the DoD, and other executive agencies (e.g., GSA, NBS) on the Air Force management of computer resources.
- b. Impact on the using/supporting commands of having different Air Force policies for weapon systems, command and control, telecommunications, intelligence, and management information systems.
- c. The adequacy of the planning, programming, and budgeting system (PPBS) to provide visibility, coordination, and control of computer resources.

- d. Current procurement policies and practices for the acquisition of computer resources.
 - e. Personnel management policies and procedures to control and assign personnel with expertise in computer resources development, acquisition, operations, and support.
5. Relevant Management Issues. Identification, approval, coordination, prioritization, and control of computer resources and the supporting technology.
6. Relevant Organizational Issues. Planning, coordination, direction, and control of computer resources development, acquisition, operations, and support.
7. Study Program.
- a. Interim Reporting. Informal progress reports will be briefed to the Vice Chief of Staff and the Air Force Focal Point during the course of the study as travel to Washington by the study directors permits. This reporting will begin with an outline of the study approach by the end of February 1975.
 - b. Formal Reporting. A briefing and executive summary will be prepared for presentation to the Vice Chief of Staff by 30 June 1975. A final report will be provided as soon thereafter as possible.
 - c. Findings. The final report should present findings, analysis, and recommendations with sufficient support and detail to permit the Air Force Chief of Staff to select alternatives and take specific actions to improve the management of Air Force computer resources.

Appendix B

PROJECTS, MILITARY ORGANIZATIONS, AND FIRMS VISITED

PROJECTS

Advanced Logistics System

STALOGS

BASE-TOP

Advanced Airborne Command Post (contractor and project office)

B-1 (contractor and project office)

AWACS (project office)

F-111 (project office)

F-16 (project office)

NORAD Command-Control Upgrade (427M)

SAC Command-Control Upgrade (436M)

TACS/TADS (Air Staff OPR and test location)

AIR FORCE ORGANIZATIONS

Secretary of the Air Force

Assistant Secretary of the Air Force for Financial Management

Assistant Secretary of the Air Force for Research and Development

Headquarters USAF

Chief of Staff

Vice Chief of Staff

DCS/Research and Development

Assistant for Policy and Management Systems (AF/RDMO)

DCS/Plans and Operations

Director of Doctrine, Concepts, and Objectives (AF/XOD)

DCS/Systems and Logistics

DCS/Personnel

Director of Personnel Planning (AF/DPX)

Director of Civilian Personnel (AF/DPC)

Comptroller of the Air Force (AF/AC)

Director of Data Automation (AF/ACD)

Director of Command, Control, and Communications (AF/PRC)

Major Commands

Air Force Logistics Command

DCS/Material Management: Weapon Systems Computer Resources
Management and Technology Office

DCS/Data Automation

DCS/Plans and Operations

Air Force Systems Command

Commander

DCS/Development Plans

Director of Data Automation

Electronic Systems Division

Commander

Director of ADPE Selection (ESD/MCS)

Director of Information Systems Technology (ESD/MCI)

Aeronautical Systems Division

Rome Air Development Center

Space and Missile Systems Organization

Air Defense Command

ADP Single Manager

Deputy Chief of Staff for Logistics

Tactical Air Command

ADP Single Manager

Strategic Air Command

ADP Single Manager

Military Airlift Command

ADP Single Manager

Other Air Force Facilities

Data Systems Design Center

Military Personnel Center

OTHER MILITARY AGENCIES AND OFFICES

Director of Defense Research and Engineering

Assistant Secretary of the Navy for R&D

Tactical Data Systems Office, Naval Air Systems Command

Chief of Naval Operations, ADP Management Planning Office
Chief of Naval Operations, ADE Equipment Selection Office
Office of Army Chief of Staff, Management Information Systems Directorate
Army Material Command, Management Information Systems Directorate
Defense Systems Management School
DoD Computer Institute
Industrial College of the Armed Forces

COMMERCIAL FIRMS

MITRE Corporation
Logicon Corporation
United Air Lines
Hughes Aircraft Corporation
TRW
The Boeing Company
General Dynamics Corporation

Appendix C

PERSONNEL AND TRAINING ISSUES¹

PERSONNEL ISSUES

While the primary emphasis of this study has been on areas other than personnel and training, there are several personnel issues that merit further investigation.

Voluntary separation rates for field-grade computer technology specialists (51XX AFSCs) appear to be significantly higher than those for field-grade personnel in general. Many of these officers leave the service at the 20-year point, even though most have six to eight years remaining before mandatory retirement. This phenomenon appears to be common to many of the specialist classifications.

Moreover, many field-grade officers who have had exposure to computers but whose primary experience is not in computer technology (e.g., officers in logistics, maintenance, personnel) are reluctant to have computer technology appear on their automated personnel records. This reluctance is a direct result of their concern that such identification would tend to narrow their career opportunities--they want to avoid "career channelization," as one officer put it.

These two observations indicate the need to examine at least two issues of Air Force officer management and assignment. The first deals with the role of a specialist in a generalist Air Force: What are the costs and benefits associated with alternatives to the generalist concept? Do specialists choose voluntary separation well in advance of mandatory retirement because of poor career opportunities? Are incentives available that would encourage these officers to remain? Should these incentives be provided? The entire career management issue deserves further investigation, especially as it applies to the Air Force's "whole man" concept.

A second, related issue focuses primarily on the classification of officers by Air Force Specialty Codes and other experience and

¹This appendix was prepared by D. Kenneth Shelton and Herbert J. Shukiar.

skill identifiers. There are a number of field-grade officers who have had excellent exposure to computer technology but whose specialty codes are not related to this technology. These officers have taken steps to ensure that their automated personnel records contain no reference to the computer-technology exposure, thereby avoiding what they feel would be a narrowing of career opportunities and a reduction in promotion chances.

In addition to questions about the generalist/specialist issue and the issue of career opportunities for the Air Force technical specialist, there are also questions about the ability of the officer classification system to produce qualified candidates for assignment. If a logistics officer, for example, is reluctant to be identified with a technical specialty such as computer ~~te~~chnology, candidates for logistics billets requiring computer technology expertise cannot be effectively identified.

While our interviews with Air Force officers do not provide an adequate basis for drawing firm conclusions, sufficient questions exist to warrant an investigation of the issues described above.

TRAINING ISSUES

Air Force training associated with computers and computer systems should be viewed from two distinctively different perspectives: the technical and the managerial.

Technicians--programmers, operators, design engineers, etc.--require in-depth training and exposure to the computer systems and tools available. They are the people who use those tools directly; they are responsible for preparing the instructions that enable the computer to perform. Technicians, therefore, work in a highly specialized and detailed area, developing expertise about how to make the computer work.

As with other Air Force technical training programs, computer courses are designed specifically to train an individual to work on a given system. These courses are typically "How to Work on System X" or "How to Program Using Language Y on System X." Most of the technical training in the Air Force (for both officers and airmen) is

conducted at three centers: Lowry AFB, Colorado, Keesler AFB, Mississippi, and Sheppard AFB, Texas. There is also some interservice training available, e.g., the Navy hosts the effort on the AN/YUK systems.

Managers, on the other hand--DPI managers, certain SPO personnel, development project managers, etc.--typically view the computer from a different, detached perspective. In today's complex Air Force environment, the manager must depend on computer systems for much of the information he needs in order to make effective decisions. He needs to develop an appreciation of how the computer can help him perform more effectively, not how the computer can be made to perform.

Air Force training (or more appropriately, education) in management philosophy and techniques is presently provided in (1) Air Force Schools and Programs and (2) Joint Military Schools.² Specific avenues for educational advancement are as follows:

1. Air Force Schools and Program
 - Air University (Air War College)³
 - AFIT Program (civilian schools at AF expense)
2. Joint Military Schools
 - Industrial College of the Armed Forces and National War College³
 - Department of Defense Computer Institute (DoDCI)
 - Defense Systems Management School

Generally speaking, the AFIT programs that are specifically designed to provide computer expertise produce technicians, i.e., in computer sciences or electronics. These technicians may be either

²The participants in this training are primarily field-grade officers who represent the middle-management decisionmaker.

³These schools are considered senior military colleges, and only one may be attended (in resident status) by any selected individual during his career. Military students are "fast-burning" O-5 and O-6 (as is typical of all the senior military schools), and thus a large percentage of the general officers in the Air Force have gone to one of these schools.

software- or hardware-oriented; they can build compilers or black-box configurations. Management (or business/public administration) programs produce either technicians (e.g., programmers) or general managers who have been introduced to the computer via a survey course and, possibly, some canned package application such as statistics. However, many of the civilian school programs are fairly flexible, and if the student is motivated, he can explore the computer area in greater depth.

The Air Force and joint military schools (excluding DoDCI) train their students to be military managers. Their exposure to computers is limited to the use of tools to support a specific function or as a specific part of a case-study problem. They generally achieve only a cursory appreciation of what a computer can do, although the other extreme can also be realized; it is not unusual to have introductory-level programming courses offered. The benefits from this type of course with regard to management are questionable indeed.

The Air Force is faced with two basic problems regarding computers and training:

1. How do you train or equip an individual with the necessary skills to do software programming, engineering, and management?
2. How does the Air Force develop managers who can understand and appreciate the management problems and technical complexities of computers, either as they are used and applied to the Air Force mission and functions or in the actual development of a computer-based or computer-supported system?

Computer expertise is an art as well as a science: How do you train someone to be a programmer or a software development project manager, since both of these tasks require talent in addition to training? Individuals can be exposed to the tools and taught how to use them, but experience and aptitude are necessary ingredients for software technicians and computer development managers. The problem

is not unique to the Air Force but is a state-of-the-art concern of the computer community at large.

The Air Force has essentially two different objectives that must be satisfied with respect to middle managers and computers: (1) to develop managers that appreciate the capabilities and complexities of the management support provided from third-generation computers, and (2) to develop personnel who can effectively function with an understanding or awareness of pitfalls, day-to-day problems, etc., in the development process of computer projects. Efforts should be directed to ensuring that

- Computer specialists in the Air Force will be exposed to generalized military management and its problems as they relate to Air Force missions, i.e., the purview of the specialist should be broadened through the various training mechanisms now available.
- Non-computer-specialist managers (e.g., SPO-type personnel) understand the various impacts of the computer on the management role and, more importantly, on (or in) the various development projects undertaken by or for the Air Force.

As previously noted, the Air Force does provide an opportunity for its managers to obtain some institutionalized exposure to the computer, but this exposure is at best limited. Perhaps a reexamination of the course material content covered in the various military professional schools is in order. Redirection of the curriculum for the general student to include an in-depth exposure to computers, their management support role, and system development pitfalls would be useful.

REFERENCES

1. Brooks, F., "The Mythical Man Month," Addison Wesley Publishing Company, New York, 1975.
2. Gradwohl, Alan J., et al., "Report on Use of Air Force ADP Experience to Assist Air Force ADP Management," Tactical Planning Division, Directorate of Planning and Technology, Electronics Systems Division, Air Force Systems Command, United States Air Force, L. G. Hanscom Field, Bedford, Mass., ESD-66-671, 3 Vols., December 1966.
3. AFSC/Electronics Systems Division, "Study of Applications Effectiveness and Problems of Air Force Information Systems," RFQ ES-4-SL, 40827, 1 July 1964.
4. *Final Report of the Joint Logistics Commanders Electronics Systems Reliability Workshop*, Departments of the Army, the Navy, and the Air Force, 1 October 1975.
5. Montague, George H., et al., "Air Force ADP Experience Handbook and Primer (Pilot Version)," Tactical Planning Division, Directorate of Planning and Technology, Electronics Systems Division, Air Force Systems Command, United States Air Force, L. G. Hanscom Field, Bedford, Mass., ESD-TR-66-673 and ESD-TR-66-672, December 1966.
6. The USAF Select Committee on Computer Technology Potential, "Air Force Organizational Ability to Exploit and Manage Computer Technology," 1970.
7. Assistant Secretary of Defense (Installations and Logistics) Report of Ad Hoc Group on "Computer Selection Policies, Procedures, and Techniques in the Department of Defense," July 1976.
8. Report to the President and Secretary of Defense by the Blue Ribbon Defense Panel, Appendix I, "Staff Report on Automatic Data Processing," July 1970.
9. Finnegan, Robert J., et al., "Project Pacer Flash: Executive Summary," Vol. II, Appendix A, "Automatic Test Equipment (ATE)"; Vol. III, Appendix B, "Operational Flight Program,"; Vol. IV, Appendix C, "Air Crew Trainers (Simulators)," Air Force Logistics Command, Wright-Patterson Air Force Base, Ohio, August 1973.
10. Archibald, Robert W., and Edward S. Hinton, *USAF Aeronautical Systems Operational Software Maintenance Study*, Avionics Engineering Directive AFSC/ASD, Wright-Patterson Air Force Base, Ohio, Technical Report ASD TR-73, 46, October 1973.

11. *Proceedings: Aeronautical Systems Software Workshop*, Air Force Systems Command, Aeronautical Systems Division, Wright-Patterson Air Force Base, Dayton, Ohio, AFSC-TR-74-03, April 1974.

BIBLIOGRAPHY

Acquisition and Use of Software Products for Automatic Data Processing Systems in the Federal Government, General Accounting Office (GAO), Washington, D.C., Report to the Congress No. B-115369, 30 June 1971.

Air Force Systems Command, *Air Force Laboratory Support to Systems Acquisition*, May 1974.

Air Force Systems Command, *Definition of an Approach to Establishment of an F-15 Avionics Software Support Capability*, Report No. ASD-TR-74-30, Wright-Patterson Air Force Base, Ohio, July 1974.

Aircraft Avionics, Digital Avionics Study Report, Vol. I, *Final Report*, Vol. IV, *Cost Trade Analysis*, Appendix C, United States Air Force, ASD-TR-73-18, April 1973.

Aircraft Avionics Tradeoff Study, Vol. I, R. K. Heitzmann, et al., *Executive Summary*; Vol. II, *Concept Development and Tradeoff*, Part 1, P. S. Kilpatrick, et al., *Avionics Function and Requirements*, Part 3, Thurber, et al., *Data Processor Tradeoffs*; Vol. III, R. K. Crowe, et al., *Concept Application Evaluation and Implementation*, prepared by Honeywell, Inc., Systems and Research Center, for Deputy for Development Planning, Aeronautical Systems Division, Wright-Patterson Air Force Base, Ohio, Report No. ADS/XR 73-18, September 1973.

Automated Data Systems Documentation Standards, DoD Manual 4120.17-M, DoD Directorate for Automation, Policy, Technology and Standards, Washington, D.C., December 1972.

Boehm, B. W., *Some Information Processing Implications of Air Force Space Missions*, The Rand Corporation, RM-6213-PR, January 1970.

Boehm, B. W., et al., *Proceedings of the TRW Symposium on Reliable Cost-Effective Secure Software*, TRW-SS-7414, March 1974.

Cooper, Stephen R., *Provision of Maintenance and Support for Avionic Weapon Systems Computer Software*, Master's Thesis, University of Maryland, 14 May 1972.

David, E. E., Jr., *Some Thoughts About Production of Large Computer Systems*, Software Engineering, NATO Science Committee, January 1969.

Department of the Air Force, "Management of Automated Test Systems," Air Force Audit Agency, Comptroller of the Air Force, Audit Report (draft) 47010,

Department of the Air Force, *The Air Force Automatic Data Processing Plan (AFADPP)*, 1 March 1975.

Department of the Navy, *Software Management*, Tactical Software Division, Laboratory Department, Naval Missile Center, November 1973.

Department of the Navy, *Tactical Computer Software Acquisition and Maintenance Staff Study*, 31 October 1973.

Electronics Systems Division, Air Force Systems Command, *Support of Air Force Automatic Data Processing Requirements Through the 1980s*, Vol. 2, *Background and Requirements*, Vol. 3, *Technology*, Appendix VI, Vol. 4, *Concepts and Configurations*, Vol. 5, *Economic Analysis*, Vol. 6, *Program Planning*, L. G. Hanscom Field, Bedford, Mass., SADPR-85 Report, June 1974.

Fisher, David A., *Automatic Data Processing Costs in the Defense Department*, Institute for Defense Analysis, Paper P-10216, October 1974.

Gammill, R. C., *Software Reliability: Philosophical Underpinnings*, The Rand Corporation, P-5306, October 1974.

Gates, Howard P., et al., *Electronics X: A Study of Military Electronics with Particular Reference to Cost and Reliability*, Volume 2: *Complete Report*, Institute for Defense Analyses, AD/A-001 065, prepared for Defense Advanced Research Projects Agency, Office of the Director of Defense Research and Engineering, January 1974.

Industrial College of the Armed Forces, "An Analysis of Problems in the Processing of Requirements for Automatic Data Processing Capabilities in the Air Force," abstract of Student Research Report No. 99, 17 February 1971.

Information Management System Steering Group, Ad Hoc Committee on Software, *Report of IMS Software Study for Manned Shuttle Payloads*, Vol. 1, National Aeronautics and Space Administration, January 1973.

Inquiry into Management of Automatic Data Processing Systems, Report to the Committee on Appropriations, House of Representatives, Comptroller General of the United States, 13 August 1970.

The Johns Hopkins University, Applied Physics Laboratory, *DoD Weapon System Software Management Study*, May 1975.

Kay, R. H., "The Management of Large Scale Computer Projects," *AFIPS Proceedings*, Spring 1969, pp. 425-434.

Kosy, D. W., *Air Force Command and Control Information Processing in the 1980's: Trends in Software Technology*, The Rand Corporation, R-1012-PR, June 1974.

Liebowitz, B. H., et al., *Procedures for Management Control of Computer Programming in Apollo*, Bellcomm, Inc., NASA Contract NAS-W-417, June 1967.

MITRE Corporation, *DoD Weapons System Software Acquisition and Management Study*, MTR 6908, Vols. I and II, June 1975.

Naval Air Systems Command, *Software Management*, Staff Study, 3 March 1972.

Navy Airborne Software Support Activity, Navy Management Branch, Tactical Software Division, Naval Mission Center, Point Mugu, Staff Study, 14 December 1973.

Practice of the Naval Ship Systems Command Related to the Procurement of the AN/YUK-7 (V) Computer System, General Accounting Office (GAO), Washington, D.C., Report No. B-163074, 22 May 1972.

Ratynski, M. V., *The Air Force Computer Program Acquisition Concept*, Spring Joint Computer Conference, 1967, Electronics System Division, USAF, Bedford. Mass.

"Software Development Project Assurance," Air Force ADP, Department of the Air Force (draft).

Space and Missile Systems Organization, AFSC, *Information Processing/Data Automation Implications of Air Force Command and Control Requirements in the 1980s (CCIP-85)*, Executive Summary and Vols. 1-11, October 1973.

Stanford Research Institute, ONR Contract N 0014-74-C-0028, SRI Project 3272, AD 777121, Menlo Park, California.

A Survey of Avionics Simulation Facilities, FEDSIM Report, MV-409-012-TAC/AFDAA, Washington, D.C., August 1974.

System Development Corporation, *A Report on Air Force Logistics Command Operation Flight Program Support: Introduction and Summary*, TM-5439/000/000, 9 December 1974.

System Development Corporation, *Standardization of Avionics Information Systems*, TM-5159/000/00A, 31 August 1973.

Tactical Computer Software Acquisition and Maintenance, Deputy Assistant Secretary of Defense (Production Engineering and Materiel Acquisition), Staff Study, 31 October 1973.

Weiss, David M., *The MUDD Report: A Case Study of Navy Software Development Practices*, Naval Research Laboratory, NRL Report 7909, May 1975.

Zempolich, Bernard A., *Executive Summary*, Vol. II, *An Analysis of Computer Software Management Requirements for Operationally Deployable Systems*, Vol. II-A, *Appendices for Chapter 3*, Industrial College of the Armed Forces, Research Fellows Program, Department of the Navy, April 1975.

Zempolich, Bernard A., "History of ADP in Government with Regard to Policy," draft prepared for ICAF.