AD NUMBER: AD0902515
LIMITATION CHANGES
TO:
Approved for public release; distribution is unlimited.
FROM:
Distribution authorized to US Government Agencies only; Test and Evaluation; 16 Aug 1972. Other requests shall be referred to HQ SAMSO/XRS, P.O. Box 92960, Worldway Postal Center, Los Angeles, CA, 90009
AUTHORITY
SAMSO ltr dtd 29 May 1974

AD902515

DDC FILE COPY

UNITED STATES AIR FORCE

INFORMATION PROCESSING/DATA AUTOMATION IMPLICATIONS OF AIR FORCE COMMAND AND CONTROL REQUIREMENTS IN THE 1980s (CCIP-85)

VOLUME XI

ROADMAPS

MAY 1972

see form 1473

DISTRIBUTION STATEMENT B

Distribution limited to U.S. Government Agencies only; Test and Evaluation, 16 Aug 72. Other requests for this document must be referred to HQ SAMSO/XRS, P.O. Box 92960, Worldway Postal Center, Los Angeles California 90009

COPY 29 OF 50 COPIES

SAMSO/XRS-71-1

900 031L

INFORMATION PROCESSING/DATA AUTOMATION IMPLICATIONS OF AIR FORCE COMMAND AND CONTROL REQUIREMENTS IN THE 1980s (CCIP-85)

NOTICE TO RECIPIENTS

The views presented in this report are those of the Study Group and do not necessarily reflect the policy or position of the Air Force or the participating Commands on any issue. The work of the Study Group has been reviewed for technical quality and adequacy by an Advisory Review Group of qualified operational and technical specialists.

DISTRIBUTION STATEMENT "B"

Distribution limited to U.S. Government Agencies only; Test and Evaluation, 16 Aug 72. Other requests for this document must be referred to HQ SAMSO/XRS, P.O. Box 92960, Worldway Postal Center, Los Angeles, California 90009

THIS PAGE INTENTIONALLY BLANK

STUDY REPORT: LIST OF VOLUMES

THIS PAGE INTENTIONALLY BLANK

TABLE OF CONTENTS

UNCLASSIFIED

UNCLASSIFIED

LIST OF ILLUSTRATIONS

LIST OF TABLES

THIS PAGE INTENTIONALLY BLANK

GLOSSARY

AABNCP	Advanced Airborne Command Post
AADC	Advanced Airborne Digital Computer
AAP	Advanced Associative Processor
ABAAP	Airborne Associative Array Processor
ABNCP	Airborne Command Post
ADP	Automatic Data Processing
ADPE	Automatic Data Processing Equipment
AED	Algol Extended for Design
AWACS	Airborne Warning and Control
CAI	Computer-Aided Instruction
EOS	Executive and Operating System
FGSS	Flexible Guidance Software System
HICAP	High-Capacity Airborne Computer
HOL	Higher-Order Language
IDHS	Intelligence Data-Handling System
IGA	Information-Gathering and Analysis
JCL	Job Control Language
JTEA	Joint Test and Evaluation Agency
MIPS	Millions of Instructions per Second
NMCS	National Military Command System
OCR	Optical Character Reader
R&D	Research and Development
SACCS	SAC Command and Control System
Semanol	Semantic-Oriented Language
TAG	Time-Automated Grid
TIPI	Tactical Intelligence Processing and Interpretation
WWMCCS	Worldwide Military Command and Control System

I. INTRODUCTION

The Roadmaps presented in this volume are a culmination of the findings of the Mission Analysis on Information Processing/Data Automation Implications of Air Force Command and Control Requirements in the 1980s (CCIP-85). They indicate what should be done to alleviate problems identified in the body of the study. They do not indicate who should do a task, or where it should be performed. Although these issues are important, CCIP-85 is not a management study. As to level of detail, the Roadmaps indicate what to do at the functional level. They furnish guidelines and background information for preparing more detailed specific task summary statements.

Most importantly, the Roadmaps are integrated internally and externally. They are coordinated with each other as well as with current research and development.

An additional point concerns the "R&D iceberg" described in Volume I (page 103). Air Force R&D represents only the tip of a large iceberg of information processing R&D nationwide. Wherever possible, the Air Force should attempt -- through such means as seed money, workshops, and standards -- to orient these other R&D efforts toward Air Force needs, rather than duplicating the R&D internally.

II. INTEGRATED ROADMAP SUMMARY

Each Roadmap consists of a prose section, a section on task descriptions, and a graphic ("facts and figures") section. The prose sections describe the problem data, demonstrate their importance to command and control, and illustrate the need for new effort to correct the situation. The graphic section displays the tasks in a time-phased array and shows how the different tasks are integrated and interdependent. Each block is numbered and can be related to the preceding section where specific funding information, related efforts, and past R&D are coordinated. Table XI-I lists some of the symbols used throughout the graphic sections of the Roadmaps, with examples.

Roadmaps have been prepared in the following areas:

- System Design/Exercise Technology
- Software System Certification
- Software Timeliness/Flexibility
- Computer Hardware Survivability
- Data Security
- High-Capacity Airborne Computers
- Multisource Data Fusion
- Communications Processing
- Source Data Automation
- Image Processing
- Computer System Performance Analysis
- Associative/Parallel Processor Exploitation
- Software Transferability
- Computer-Aided Instruction
- Hardware Destructibility
- WWMCCS Conversion
- Interservice Coordination
- Hardware Laboratory

The summary of funding, for all Roadmaps, is shown in Figures XI-1 and XI-2 for minimum and reasonable efforts.

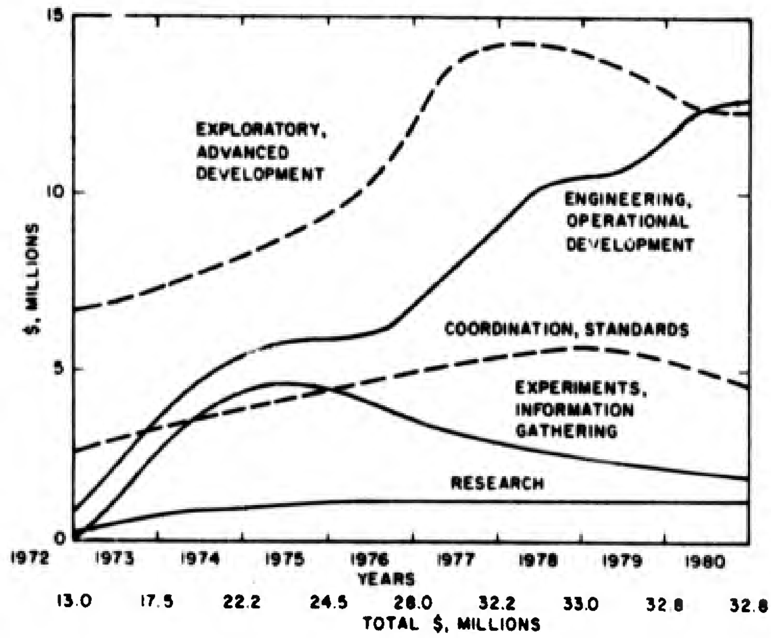


Figure XI-1. CCIP Roadmap Totals - "Minimum" Efforts

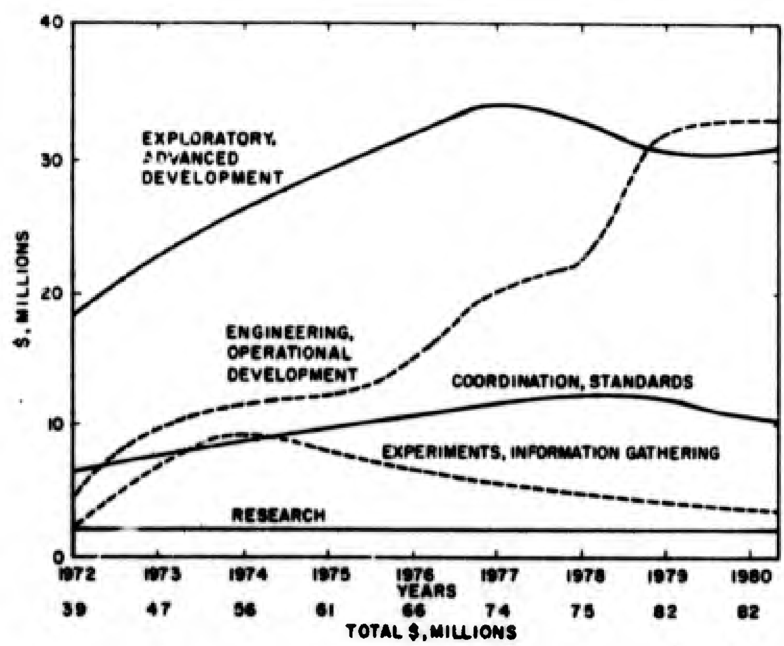


Figure XI-2. CCIP Roadmap Totals - "Reasonable" Efforts

TABLE XI-I

ROADMAP GUIDELINES: PROJECT TYPES

Type	Title	Examples
RES	Research (6. 1)	graph theory
EXD	Exploratory Development (6. 2)	query languages
AVD	Advanced Development (6. 3)	TACC data automation
EGD	Engineering Development (6. 4)	exercise evaluation model
OPD	Operational Development (6. 5)	system performance measures
OPD/X	Experimental Operational Development	alternate software development in AED, FGSS, structured programming, conventional methods
IGA	Information-Gathering & Analysis	software effort measurement
LIB	Library & Tools Inventory	data-management system evaluation
SVC	Service Functions	workshops, training, expert aid
STD	Standards Development	for compilers, OS's, hardware
MTH	Methods & Procedures Development	configuration management
ISC	Interservice Coordination	joint interface test planning
RQA	Requirements Analysis	WWMCCS computer sizing

As an example o how to read the "facts and figures" sheet, here is the information Tab e XI-II provides on project 1.4, Requirements Analysis Methods and Service (described below on pages 9 and 10). This activity begins in 1973, phases up to its maximum level by 1975, and stays at a constant level there through 1980, generally the last year considered in the Roadmaps. This constant level should be at least $0.5 million per year (in 1972 dollars) to achieve a minimum critical mass for the activity ("peak annual rate" column); a more reasonable operating level would be at $1.0 million per year, but even a "crash" effort would reach a point of diminishing returns above a level of $4 million per year. One related project is already planned in the area; it is described in Section 2.4.6. of Volume X, "Current Information-Processing R&D Programs," with a proposed total expenditure of $0.5 million over the three years FY 1973-1975. In this instance, the "number of efforts," listed parenthetically behind the "peak annual rates," indicate that all activities will be carried out from a single center. The distribution of type of effort is 50 percent development of methodology and 50 percent service on-site at user commands. (Table XI-I describes the various categories of effort, with examples.) There is no funding overlap with other Roadmap projects.

III. ROADMAP #1: SYSTEM DESIGN/EXERCISE TECHNOLOGY

A. BACKGROUND AND OBJECTIVES

The force structure analyses upon which the Air Force bases its major procurement and deployment decisions involve force effectiveness calculations which generally assume that command and control functions are carried out to perfection. No delays, garbled messages, uncoordinated decisions, or mistaken identities mar the performance of the ICBM force in the usual U.S.-U.S.S.R. missile exchange calculation or aircraft penetration calculation. However, in practice, command and control systems -- and resulting system performance -- fall far short of this ideal.

A good exercising capability for command and control systems can provide, on a routine basis, the same sort of valuable insights as a post hoc high-level committee, with better data collection and without the need of consuming the time of so many high-level decision-makers. However, the exercise capabilities available today are highly insufficient, mainly because of their high dependence on the manual processes of scenario generation, script generation, operations monitoring, event recording, and analysis. For example, the annual High Heels exercise requires an 18-month preparation and analysis activity. Much of this could be performed by computers to provide a much more rapid, thorough, and responsive exercising capability for Air Force command and control systems at a relatively modest R&D cost; details are provided in Annex A of Volume VII ("Integrated Design"). Although a highly automated exercise capability would certainly not eliminate all future errors, it would strongly increase the combat readiness of Air Force command and control systems and sharply reduce the incidence of future high-level post hoc investigating committees.

Another indicator of the difficulty of problems in requirements analysis and design is the degree to which command and control computer programs have to be rewritten once they are initially completed. For example, 95 percent of the software for the SACCS 465L system had to be rewritten, and 67 percent of the software for Seek Data II had to be rewritten in order to make the systems responsive to operational needs.

Difficulties such as these are by no means unique to the Air Force; they are characteristic of many attempts to automate complex information systems in other organizations such as the U.S. Army, U.S. Postal

Service, and airline reservation systems. Much of the problem stems from the relative newness and intrinsic difficulty of the fields of requirements analysis and integrated design for automated information systems.

Clearly, any automated aids which the Air Force could supply to designers (to simplify their job and assure that their results are reconcilable) would be a great help. None of these aids exists right now, but there are some which appear to merit further investigation, such as ISDOS,[1] FOREM (sponsored by RADC), and IBM's TAG (Time-Automated Grid).[2] Information flow simulation capabilities such as SCERT, CASE, CSS, SAM, and ECSS represent a complementary approach to the problem.[3]

Unless more R&D effort is devoted to advancing technology in the requirements analysis, design, and exercise areas during the 1970s, the Air Force command and control in the 1980s will be called upon to perform much more difficult and delicate functions with very little assurance that the system is capable of carrying them out. Some likely consequences would be:

1. Reduced Force Effectiveness -- The error rates in High Heels 1967 indicate how seriously an unexercised system could degrade Air Force capabilities.

2. Unsatisfactory Force Structure Decisions -- Unless better assurance can be provided that dynamic force management can be performed effectively, decision-makers may reject Air Force initiatives for weapon systems to provide such capabilities.

3. Reduced Deterrence Credibility and National Prestige -- Future command and control performance which leaves a less-than-surgical impression in the minds of leaders of other nations will cause them to question our ability to back up our defense commitments and policies.

4. Impaired Human Performance -- Lack of practice and confidence in exercising a command and control system generally lead to a highly redundant mode of operation for the commander and his staff, with little use of many of the helpful features which the system provides.

Clearly, the added capabilities obtained by further R&D efforts in automated aids to requirements analysis, design, and exercising of command and control systems would not eliminate all problems in the area, but they would contribute a significant increase in Air Force combat readiness for the complex missions of the 1980s.

Further, the R&D would eventually pay for itself, for the Air Force pays a great deal for inefficient and inappropriate requirements analyses and design techniques. Roughly 35 percent of the effort in a software project is typically spent on analysis and design. Given the annual Air Force software expenditure of roughly $1 billion a year, a set of techniques which saved one man-day of analysis and design effort per man-month would save the Air Force roughly $17 million a year. More incisive analysis and design techniques would effect at least as much additional savings because unnecessary or inappropriate system hardware and functions would be eliminated; but these savings are more difficult to isolate or estimate specifically.

B. TASK DESCRIPTIONS

Roadmap #1 is subdivided into 14 activities. These are described below.

1. Evaluation of Automated Aids to C&C Requirements Analysis (1.1)

This activity examines and compares the advantages and disadvantages of existing automated aids to requirements analysis for C&C systems (e.g., ISDOS, TAG, PATTERN, etc.).

2. Experiments with Automated Aids to C&C Requirements Analysis (1.2)

Based on the evaluation in activity 1.1, an experiment is carefully structured and performed to compare the performance of selected automated aids in aiding a C&C system designer in performing his task. Factors evaluated would include ease of learning, ease of use, specification efficiency, error modes, and design effectiveness.

3. Advanced Automated Aids to C&C Requirements Analysis (1.3)

Based on the results of the experiments of activity 1.2, a decision can be made on whether to expand some of the existing tools into general Air Force capabilities, to design new Air Force capabilities, or to submit to further experimentation. The funding summary corresponds to the second alternative; clearly, the other alternatives would have other funding levels.

4. Requirements Analysis Methods and Service (1.4)

This activity maintains a cadre of requirements analysis experts who spend alternate terms in the field and at a center. In the field, the expert works as the leader or a senior member of a requirements analysis team on a specific operational C&C project. At the center, he incorporates his field experience into the body of knowledge accumulated in other activities in the System Design/Exercise Technology area to produce a standard reference work for the area, and to develop standards and advanced techniques for specific problems. Examples of such problem areas are discussed in detail in Volume VII, Integrated Design; they include interoperability, enervation analysis, command system technology, intelligence-operations interfaces, and configuration management techniques.

5. C&C Usage Study (1.5)

This is a continuing activity to study the way command and control systems are used in practice: the available information that is more and less relevant; the types of information and decision aids that are desirable but unavailable; and the interaction and information flow between levels of command and staff. This collection of information serves as a reference for all the efforts in the area.

6. C&C Usage Study of a Selected System (1.6)

To serve as a base for the activities in developing decision aids and displays, exercise capabilities, and simulation capabilities for C&C systems, a single representative C&C system should be selected and studied in detail.

7. Decision Aids and Displays (1.7)

This activity covers many of the existing Air Force R&D efforts in the area. Under this plan, these efforts would be combined with additional efforts to concentrate on developing decision aids and displays for the selected representative C&C system of activity 1.6. These could then be evaluated in practice in activity 1.9, using the prototype exerciser of activity 1.8.

8. Prototype Exerciser (1.8)

This activity develops a prototype system of automated aids -- scenario generation, script generation, external inputs generation, exercise monitoring, data collection and analysis, etc. -- to exercising the representative C&C system of activity 1.6. Enough generality is maintained to allow subsequent evaluation of the exercises on a different C&C system.

9. Prototype Exerciser Evaluation (1.9)

Once the prototype exerciser is developed, it can be used to exercise the selected representative C&C system under a wide variety of conditions, to evaluate both the C&C systems and the exerciser. It can also be used to evaluate the decision aids developed in activity 1.7 by substituting the new decision aids for existing ones in an exercise. Finally, the exerciser can assess how well the different information flow modeling and simulation techniques of activity 1.12 predict the performance of the exercised system.

10. Advanced Exerciser (1.10)

Based on the experiences of evaluating the prototype exerciser on its nominal C&C system and one other C&C system, a decision can be made on whether to expand the prototype into a general Air Force exerciser capability, to design a general Air Force exerciser capability, or to pursue further experimentation. The funding summary corresponds to the second alternative; clearly, the other alternatives would have other funding levels.

11. Simulation Tool Evaluation (1.11)

This activity examines and compares the advantages and disadvantages of existing analytical modeling and simulation techniques for investigating information processing aspects of C&C systems (e.g., COMET, MAMO, CASE, CSS, ECSS, SAM, SALSIM).

12. C&C Simulation Experiments (1.12)

Based on the evaluation in activity 1.11, an experiment is carefully structured and performed to compare the performance of selected information flow modeling and simulation techniques in modeling the performance of the C&C system of activity 1.6. Factors evaluated would include ease of learning, ease of specification, modeling efficiency, error modes, and accuracy of describing and predicting the performance of the C&C system. Final evaluations would involve comparisons with the selected system in its exercise mode.

13. Advanced Simulation Development (1.13)

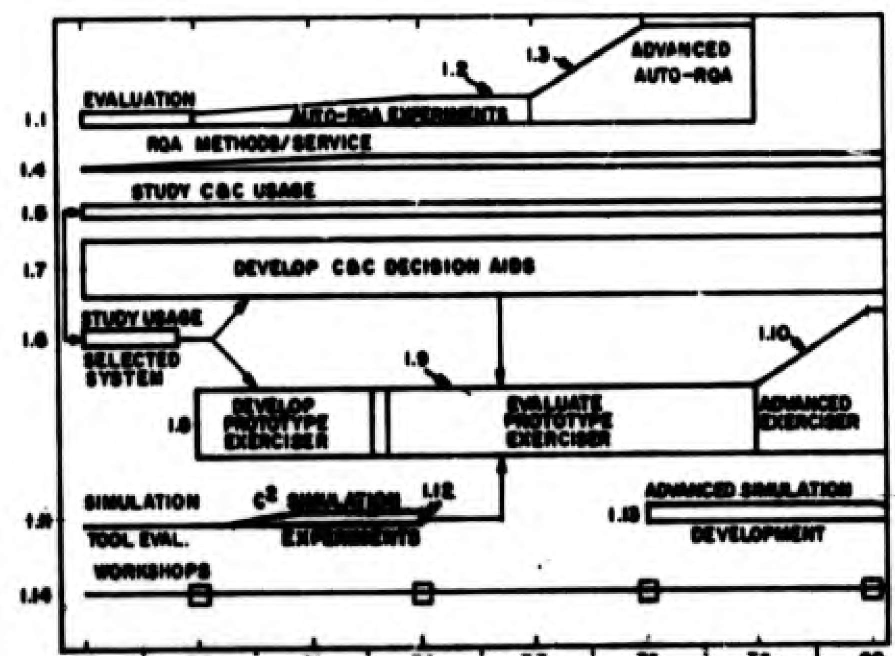
Based on the experience in the simulation experiments of activity 1.12, a decision can be made on whether to expand some of the existing tools into general Air Force capabilities, to design new Air Force information flow simulation and modeling capabilities, or to submit to further experimentation. As in activity 1.10, the funding refers only to the second alternative.

14. Workshops (1.14)

These are held periodically to bring together C&C operators, SPOs, experimenters, simulators, decision aids researchers and developers, and requirements analyzers to concentrate on common problems such as choice and strategy of experimenting with a selected representative C&C system. Comparisons of simulators or automated aids to requirements analysis may be reviewed. Throughout the process, more cohesiveness will be imparted to the R&D area of System Design/Exercise Technology.

C. FACTS AND FIGURES

Table XI-II presents the pertinent facts and figures for Roadmap #1: System Design/Exercise Technology. Figure XI-3 is a graphic display of Roadmap #1.



$, MILLIONS	FY 74	FY 75	FY 76	FY 77	FY 78	FY 79	FY 80
MINIMUM	8.0	7.6	7.8	10.6	10.1	16.1	13.1
REASONABLE	11.3	16.4	16.4	23.0	22.5	34.5	26.5
CRASH	34.7	34.1	34.1	43.5	42.5	62.5	50.5

Figure XI-3. Roadmap #1: System Design/Exercise Technology

TABLE XI-II

ROADMAP #1: FACTS AND FIGURES

No.	Title	Fiscal Years Active	Peak Annual Rate, $\$10^6$ (No. of Efforts)			Existing Projects			Types
			Min.	Reas.	Crash	ID	year	$\$10^6$	
1.1	Auto-RQA Eval.	73 → 74	0.2 (1)	0.4 (1)	0.6 (1)				LIB-80, RQA -20
1.2	Auto-RQA Exper.	74 ↗ 76 → 77	1.0 (2)	2.0 (2)	4.0 (3)				OPD/X-80, RQA-20
1.3	Auto-RQA Advanced	77 ↗ 78 → 79	3.0 (1)	8.0 (2)	12.0 (2)				EGD-80, RQA-20
1.4	RQA-Method/Service	73 ↗ 75 → 80	0.5 (1)	1.0 (1)	4.0 (1)	2.4.6	73-75	0.5	MTH-50, SVC-50
1.5	Study C^2 Usage	73 → 80	0.4 (1)	0.7 (2)	1.0 (3)	2.5.2 2.5.3		3.0	IGA-100
1.6	Usage, Select System	73	0.4 (1)	0.7 (1)	1.0 (1)				IGA-100
1.7	Decision Aid Development	73 → 80	2.5 (1)	5.0 (2)	12.0 (3)	2.2.2 2.2.5 2.3.2		0.25 1.0 2.5	RES-15, EXD-30, AVD-55
1.8	Prototype Exerciser	74 → 76	3.0 (1)	6.0 (1)	12.0 (1)				EGD-100
1.9	Eval. Exerciser	77 → 79 ↘ 80	3.0 (1)	6.0 (1)	10.0 (1)	2.4.7		0.7	EGD-50, IGA-50
1.10	Adv. Exerciser	79 ↗ 80	6.0 (1)	12.0 (1)	20.0 (1)				OPD-100
1.11	Sim. Tool Evaluation	73	0.1 (2)	0.2 (3)	0.4 (4)				LIB-100
1.12	C^2 Sim. Experiments	74 ↗ 75 → 76	0.2 (1)	0.4 (2)	0.6 (3)				OPD/X-100
1.13	Adv. Sim. Development	78 → 80	0.5	1.5	3.0				EGD-100
1.14	Workshops	73 → 80	0.2 (3)	0.3 (5)	0.5 (9)				SVC-100

IV. ROADMAPS #2 AND #3: SOFTWARE/SYSTEM CERTIFICATION AND SOFTWARE TIMELINESS/FLEXIBILITY

A. BACKGROUND AND OBJECTIVES

1. Software/System Certification

Dynamic force management in the 1980s will be performed within a context of far more complex Air Force systems and world situations. The ever tighter time constraints and ever higher data flow rates will cause the commander and his staff to rely more and more on automated aids to data analysis and decision-making. Since the computer software required to manage this data processing will need to be far more complex, it becomes increasingly more difficult to guarantee that the software is indeed producing the right results at all times, rather than producing misinformation which could critically degrade Air Force capabilities or inappropriately escalate a crisis situation.

The software/system certification problem has been difficult even with past or current systems. For example:

- Early in its lifetime, the software in the BMEWS system allowed the rising moon to be mistaken for a massive Soviet missile raid.[4] Had such an error occurred during the Cuban missile crisis of 1962, the consequences could have been grave.
- During the 1963 NORAD Skyshield exercise, a software patch was inserted which virtually incapacitated the entire air defense capability, routing misinformation to and from radars and control centers throughout the system.

As the examples above should indicate, software certification is not easy. Ideally, it means checking all possible logical paths through a program; there may be a great many of these. For example, the number of possible paths in one cycle of the Titan III guidance and navigation software is about 2×10^{18}. Even if one could check out one path per microsecond, the certification job would take 60,000 years. Furthermore, each time the program is modified, some considerable fraction of the testing must be repeated. Even for small software modifications, one should not expect error-free performance thereafter.[5]

Some simplification of the problem can be achieved by imparting structure to the software in ways which permit thorough verification at various levels of the software control hierarchy. Considerable success with "structured programming" techniques has been achieved at universities in Europe and in a single experimental project at IBM,[6] but it is not clear to what extent the success is also due to the expertise of people involved. A good deal of further experimentation and R&D are necessary to evaluate and extend these and related software technology, management, and testing concepts to provide capabilities for operational Air Force use which can, at the least, partially keep up with the increased complexity of, and reliance on, command and control software. More thorough discussions of promising certification concepts are provided below and in Volume IV on software technology.

Many of the error modes characteristic of software bugs are similar to those occurring in underexercised command and control systems, and many of the aspects of software testing can be considered as aspects of system exercising. Thus, insufficient R&D in software certification carries risks -- of reduced force effectiveness, rejected Air Force initiatives, reduced deterrence credibility and national prestige, and impaired human performance -- similar to those cited in Roadmap #1. As a highly visible example, associated with force structure decisions, one of the major arguments put forth in the public forum against the deployment of the ABM was that its software would be too complex to certify.[4]

A related risk caused by insufficient R&D effort in certification involves the critical instability of software-based systems because of the unpredictable nature of software errors. Command and control software which has worked perfectly for five years has been known to suddenly blow up and send false signals when confronted with a particularly unlikely sequence of inputs. A similar incident could cause a provocative strategic escalation or degradation of defense capability at a critical time.

Again, additional R&D in the certification area would be likely to pay for itself in the long run. Roughly 45 to 50 percent of the total effort in a software project is typically spent on checkout and test. Again applied to the Air Force software expenditure of $1 billion a year, advanced certification techniques which could save one man-day of checkout and test activity per man-month would save the Air Force roughly $20 to $25 million a year. Additionally, since this testing effort is generally on the critical path of the overall system, savings in checkout and test time imply earlier operational readiness for Air Force systems; this will be discussed more fully in the next item.

2. Software Timeliness/Flexibility

As emphasized in Volume IX, dynamic force management in the 1980s will require the software capability to support a rapid response to unexpected events with quickly planned or even improvised combinations of aircraft, weapons, sensors, and command-control-communication elements; further, the software will have to be thoroughly validated for correctness. Providing such capabilities requires a large degree of built-in flexibility and versatility in command and control software, which has always translated into a long gestation time, and has frequently translated into situations in which the software causes delays in delivery of an entire system to the using command.

Experience with one current project provides a good example. It shows well the multiplier effect of software's being on the critical path of the system: direct software costs of roughly $2 million for a six-month slippage lead to virtual losses of roughly $100 million worth of command and control capability. Further, it provides an example of minimizing the risk of further software delays by cutting back on some of the versatility and flexibility in the original software specifications.

Such tradeoffs can often lead to difficult problems. For example, the SACCS 465L software was designed using a one program-one file concept which greatly simplified the development effort; but it produced a system which was highly inflexible. Thus, even six months after Biggs AFB had been shut down, the maintenance programmers had not been able to discover each reference to Biggs in each program and file. The result, when the commander requested status reports and received them with references to Biggs AFB still present, was a greatly decreased confidence in overall command and control system credibility.

The length of the software gestation period creates frustrating delays for currently desired capabilities. For example, the 485L Seek Flex-Seek Digit system has had two years of system design studies and is currently entering its hardware acquisition phase; but it is not scheduled to attain combat operational capability until 1976.

Again, however, the problem is at least as difficult outside the Air Force. The software of the IBM OS/360 cost over $200 million and was over a year late.[7] The FAA's Air Traffic Control Center software is roughly 60 months into a 17-month schedule; costs have escalated from $1.8 million to $19 million.[8]

Software is virtually incompressible with respect to elapsed time; adding more manpower to an established project generally increases rather than decreases its duration.[9] Thus, the avenues by which technology can improve the situation involve developing tools to increase each individual's software productivity, enhance software teamwork by improved project design and organization, and initiate software development activity earlier in the system development cycle, thus moving it off the critical path.

Primarily because of advances in higher-order programming languages and improved libraries of modularized programming aids, the average number of machine instructions per programming man-month has increased from about 120 in 1955 to about 350 in 1970, and will probably reach about 1000 by 1985. These figures are averaged over all types of software production; the averages for command and control software are considerably lower. In the 1966 SDC sample, command and control software productivity averaged about one-third of the productivity in non-C&C applications.

The AED and FGSS (Flexible Guidance Software System) packages are systematic approaches to providing modularized programming aids (as are data management systems and executive operating systems when used by applications programmers). A preliminary analysis of potential programming savings which FGSS could provide during the various phases of software development revealed an estimated overall savings of 42 percent.

AED and FGSS may also be categorized as "structured programming," which includes the European and IBM-New York Times efforts referred to above under certification. The IBM-New York Times effort also included an experiment along the second avenue of enhancing software teamwork, the "chief programmer" approach; together, these measures led to an estimated 50-percent savings in software cost and a 75-percent savings in elapsed time on a 30,000-instruction system. However, it is not clear how much these savings are due to the use of exceptionally good personnel, and to what extent they could be approached on the typical Air Force project.

To a large degree, however, the problems of software productivity are problems of management: thorough organization, good contingency planning, thoughtful establishment of measurable project milestones, continuous monitoring on whether the milestones are properly passed, and prompt investigation and corrective action in case they are not. In the software management area, one of the major difficulties is the transfer of experience from one project to the next. Some of these management considerations have been embodied in the 375-series (lately revised to 800-series) software configuration management guidelines.[10, 11] Even with these, however, many of the lessons learned in SAGE and published ten years ago in Hosier's 1961 article[12] (on the value of integrated measurement capabilities, formatted debugging aids, early prototypes, concurrent system development and performance analysis, etc.) are often ignored in today's Air Force software developments. There is a strong need to revise the 375/800-series software guidelines to more appropriately reflect unique software considerations,* and to supplement the guidelines with additional detailed facts and experience which should be considered in structuring and managing the project. Some good forward steps are being made in this direction; AFSC Regulation 800-2 includes

---

*For example, "First Article Configuration Inspection" is an appropriate term to apply to the first item in a hardware production run; but it is inappropriate for software, where the series of production steps is traversed only once.

provisions for an external project review board, a device used successfully in the one recent software project to avert further slippages.

The third avenue of enabling software development to start earlier in the system development cycle is addressed by the concept of "software-first" machine which is described in more detail below.

In the software productivity area, several promising tools and techniques exist, such as AED, FGSS, and structured programming, which need further analysis followed by controlled experimentation to determine under what conditions they are appropriate vehicles for software development in an Air Force command and control environment. With all the current Air Force and DOD emphasis on prototyping and "fly before buy," it is surprising that the concept is not applied more to alternative software development tools. The potential payoffs are large; an increase in average software productivity from 10 to 11 instructions per man-day translates into an annual Air Force software savings of $100 million.

Concepts such as the "software-first" machine provide the potential of telescoping the software development cycle from six to four and a half years, with additional side benefits of hardware-software tradeoff flexibility and hardware that is two to three years newer when the system is initially operational. Were such a capability available today, it would be possible to achieve an operational Seek Flex-Seek Digit System by late 1974 instead of 1976, equipped with 1973 hardware technology instead of 1970, and with much more flexibility to adapt to changing requirements for tactical command and control support, as their needs became known during the intervening years.

Without considerable R&D support in the area, it is likely that software delays will continue to deny the Air Force the early use of new inventory items such a remotely piloted vehicles and advanced space sensors, and that compromises in software flexibility to meet deadlines will be likely to result. If so, some instances of heavy-handed use of force in delicate situations could result, such as in the 1983 New Guinea scenario in Volume VIII, which would adversely impact not only the immediate mission objectives but also the long-term U.S. image of dynamic force management capability.

B. TASK DESCRIPTIONS

This combined Roadmap for Software/System Certification and Software Timeliness/Flexibility consists of 18 tasks which are described here.

1. Certification Research (2.1)

The "Certification Research" project consists of research into areas such as program proof theory, theory of hierarchical software, and theory of parallel processes. Researchers should be periodically exposed to actual C&C software certification problems and data in some depth.

2. Software Research (2.2)

Software research is a group of projects exploring the basic specification of computing processes, analyzing alternative combinations of hardware and programming language capabilities with respect to the ease of specification, efficiency of realization, and ease of certification of the computing process. Part of the effort should be devoted to obtaining maximum value from such non-Air Force projects as ARPA's effort in automatic programming.

3. Experimental Structured Programming (2.3)

The "experimental structured programming" project (at its "reasonable" level) involves originally selecting a small C&C software project and using carefully balanced and representative software teams to develop parallel versions of the software, by employing typical current techniques, AED, structured programming, and FGSS or other candidates, if available.

Considerable planning is necessary to assure appropriate information-gathering and analysis, and a sound experimental design. Based on the results of the initial experiment, the two most promising candidates are selected for an additional experiment on a larger C&C software project. Analysis also continues through the maintenance phase.

4. Experimental Hardware/Software Aids (2.4)

This activity involves developing and evaluating potential hardware and software aids to software development and certification (e.g., expanded instruction sets, extra debugging bits and status registers, directly executable higher-order language machines). As in task 2.2, evaluation could involve parallel development of C&C software, using the software-first machine (2.13 - 2.15).

5. Information-Gathering and Analysis (2.5)

In order to determine procedures and criteria by which refinements can be made to the development and certification of software, it is useful to gather statistical data on the software production process. By analyzing these data, it is hoped that correlations can be determined which will be beneficial to the production and certification of software. The "number of efforts" in Table XI-II refers to the number of different C&C software projects on which data are being gathered.

6. IGA Standards and Use (2.6)

Once the base of experience in gathering and analyzing data on software production is accumulated, it will be possible to develop standards and procedures for a continuing information-gathering and analysis effort on all C&C software projects.

7. Industry Product Requirements (2.7)

This project builds on the results of the other projects to develop Air Force (and perhaps DOD) standards for software product capabilities. Some examples in the certification area might include required diagnostics and cross-reference tables for compilers; tracing and status monitoring for operating systems; and extra status registers or debugging bits for hardware. Such standards would provide powerful Air Force leverages on the "submerged" portions of the R&D iceberg -- industry R&D, corporate R&D, and R&D-equivalent SPO developments.

8. Software Testing Team (2.8)

This project maintains a cadre of software/system testing experts within the Air Force staff organization for information-processing technology mentioned elsewhere, who spend alternate terms in the field and at their home bases. In the field, the testing expert works as the leader or the senior member of the software/system test team in developing and carrying out test plans and interfacing with the software developers. At his home base, he incorporates his field experience into efforts in industry product requirements determination (2.7), information-gathering and analysis standards (2.6), or "assembly line" certification procedure standards.

9. Software Development Assistance Team (2.9)

This project operates in a manner similar to the software testing team in task 2.8 above. Experts in such areas as data base design, computer-communications interfaces, and software/system conversion would alternate between assisting user commands in the field and improving Air Force-wide standards and procedures at their home bases.

10. Workshops (2.10 and 2.11)

The workshops on certification and other software topics should bring together researchers, system developers, and users of C&C information-processing systems to assure that each is working with sufficient realization of the others' potential capabilities and operational constraints.

11. New Techniques (2.12)

This project is a major attempt to build certification-oriented software tools which may serve as Air Force standards. These developments are necessarily based on the insights gained in the earlier and continuing experimental and information-gathering projects.

12. "Software-First" Machine Studies (2.13)

This task would be the first step in the potential design, development, and utilization of a "software-first" machine: a general-purpose computer which, through microprogramming techniques, could be reconfigured to look like many other machines. This would provide the capability to design software, perform hardware-software tradeoffs, and write and test software for a "target" machine before the "target" machine was procured and installed. Components of the study task would include considerations of underlying hardware technology, range of architectures needed for Air Force applications, higher-order language support and compatibility considerations, and implications for procurement and configuration management procedures.

13. Software-First Machine Prototype (2.14)

Based on the studies in 2.13, a decision is made on whether and to what extent to develop and evaluate a prototype software-first machine. Evaluation should involve parallel development of a representative C&C software system using conventional procedures and using the software-first machine prototype.

14. Software-First Machine Development (2.15)

Based on the results of task 2.14, a decision is made on the full-scale development of an Air Force software-first machine and establishment of associated support functions.

15. Experimental Assembly Line (2.16)

This project develops the software integration and testing techniques outlined in Annex A and applies them to a large C&C software project.

16. Assembly-Line Certification Standards and Use (2.17)

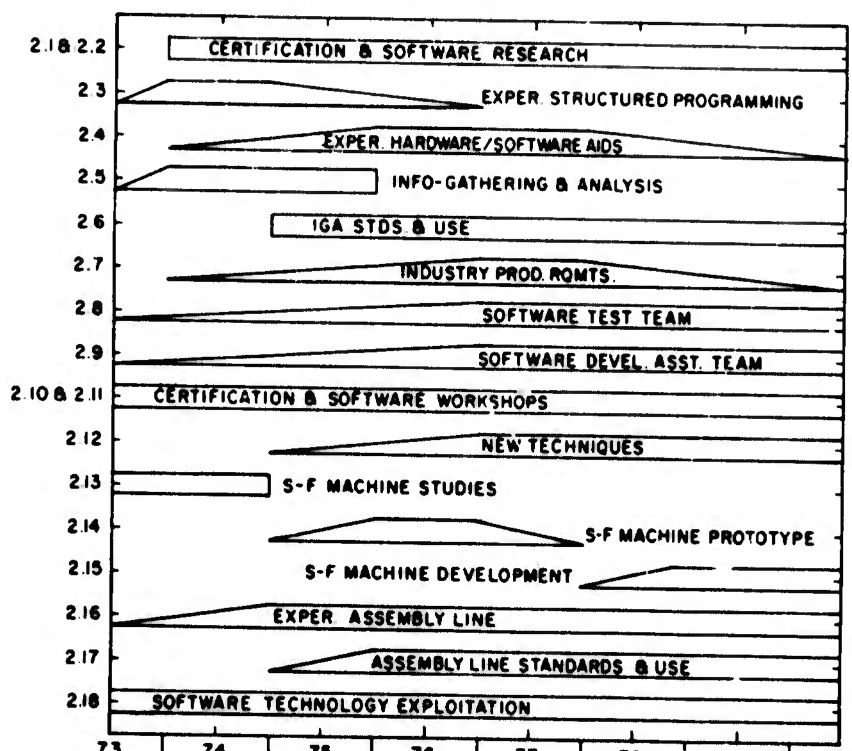
This project develops configuration management procedures based on the experimental assembly line (task 2.16) and others, and applies these to future C&C software development projects.

17. Software Technology Exploitation (2.18)

Based on data gathered in the C&C usage studies (tasks 1.5 and 1.6 of Roadmap #1) and the software information-gathering activity of task 2.5, the software technology exploitation project evaluates current software R&D products such as languages, compilers, operating systems, and data management systems, and adapts them to Air Force use as appropriate. The activity should also include additional software R&D efforts when analysis of the above requirements and technology data indicates it is appropriate.

C. FACTS AND FIGURES

Table XI-III presents the pertinent facts and figures for Roadmaps #2 and #3. Figure XI-4 is a graphic display of these Roadmaps.



$, MILLIONS	FY 74	FY 75	FY 76	FY 77	FY 78	FY 79	FY 80
MINIMUM	8.13	16.13	15.73	15.13	18.13	16.23	16.23
REASONABLE	16.90	32.90	31.90	28.90	36.90	32.90	32.90
CRASH	35.05	64.05	60.05	54.55	69.55	63.55	63.55

Figure XI-4. Roadmaps #2 and #3: Software/System Certification and Software Timeliness/Flexibility

TABLE XI-III

ROADMAPS #2 AND #3: FACTS AND FIGURES

No.	Title	Fiscal Years Active	Peak Annual Rate, $\$10^6$ (No. of Efforts)			Existing Projects			Types
			Min.	Reas.	Crash	ID	Years	$\$10^6$	
2.1	Cert. Research	74→80	0.1 (1)	0.75 (2)	0.5 (3)				RES-100
2.2	Software Research	74→80	0.5 (4)	1.0 (7)	3.0 (10)				RES-50, EXD-50
2.3	Exper. Structured Prog.	73 ↗ 74→75 ↘ 77	1.0 (1)	2,0 (1)	5.0 (2)	2.3.1b	71-74	0.75	OPD/X-90, IGA-10
2.4	Exper. Hardware/ Software Aids	74 ↗ 76→78 ↘ 80	1.0 (1)	2.0 (1)	4.0 (3)				AVD-30, OPD/X-60, IGA-10
2.5	Info. Gathering & Analysis	73 ↗ 74→76	0.5 (2)	1.0 (4)	1.5 (6)				SGA-100
2.6	IGA Stds. & Use	75→80	0.5 (1)	1.0 (1)	1.5 (1)				IGA-80, STD-20
2.7	Industry Prod. Rqmts.	74 ↗ 77→78 ↘ 80	0.8 (1)	1.5 (1)	2.5 (1)				STD-100
2.8	Software Test. Team	73 ↗ 77→80	0.25 (5)	0.5 (10)	0.75 (15)				SVC-100
2.9	Software Devel. Asst. Team	73 ↗ 77→80	0.5 (10)	1.0 (20)	1.5 (30)				SVC-100
2.10	Certification Work-shops	73→80	0.03 (3)	0.05 (5)	0.15 (9)				SVC-100
2.11	Software Workshops	73→80	0.05 (5)	0.1 (7)	0.15 (9)				SVC-100
2.12	New Techniques	75 ↗ 77→80	5.0 (1)	10.0 (2)	20.0 (3)				AVD-25, EGD-30, OPD-45
2.13	S-F Machine Studies	73→75	0.4 (1)	1.0 (3)	4.0 (5)				EXD-100
2.14	S-F Machine Proto.	75 ↗ 76→77 ↘ 78	2.0 (1)	4.0 (1)	6.0 (2)				AVD-100
2.15	S-F Machine Devel.	78 ↗ 79→82	5.0 (1)	10.0 (1)	20.0 (1)				EGD-40, OPD-60
2.16	Exper. Assy. Line	73 ↗ 75→76	1.0 (1)	2.0 (1)	4.0 (2)				MTH-30, OPD/X-60, IGA-10
2.17	Assy. Line Stds. & Use	75 ↗ 76→80	0.5 (1)	1.0 (1)	1.5 (1)				MTH-100
2.18	Software Tech. Exploitation (OS, DMS)	73→80	2.0 (2)	4.0 (2)	8.0 (2)				AVD-50, EGD-50

V. ROADMAP #4: SURVIVABILITY

This Roadmap is classified and is being issued to the appropriate agencies under separate cover.

VI. ROADMAP #5: DATA SECURITY

A. BACKGROUND AND OBJECTIVES

A number of multiuser, time-shared, remotely accessible data management systems are being developed for Air Force command and control use. Some examples are SAC's SATIN IV, MACIMS, and the various intelligence data-handling systems (IDHS). The resources of such systems--storage, data files, processors--are shared by the concurrent users of the system. In order to fully realize its potential, a resource-sharing command and control data management system should allow storing and processing of classified and unclassified operational data and compartmented intelligence information, while allowing concurrent access to users with various levels of security clearances and need-to-know authorizations. This, as well as the protection of classified and compartmented data from unauthorized access, requires development and implementation of effective data security techniques.

To date, the general data security problem has not been fully solved. In fact, there are instances of resource-sharing computer systems containing classified information which have been completely and undetectably penetrated with only a few man-weeks of effort. Individuals at DCA have estimated that 100 man-years would be necessary to secure the IBM OS/360 operating system and that, even then, the system would not be completely certifiable. Only within highly secured environments, such as intelligence facilities, is the processing and storage of compartmented information in a resource-sharing system allowed (DCI Directive No. 1/16, 7 January 1971). Classified operational data (excluding compartmented information) are likewise excluded from the general resource-sharing systems except at times when it can be guaranteed that only users with appropriate clearances have access and the system can be adequately sanitized (declassified) after the classified processing. A set of guidelines toward achieving the desired general data security has been published by ARPA,[13] but the software and hardware data security techniques that have been developed to date by industry and Air Force have not been found adequate.

Nevertheless, Air Force data management systems are being developed with the expectation that data security problems will be solved by the time of their IOC dates (1975 time period and earlier). It is not at all apparent that this will happen unless the Air Force works actively within the national security community to establish a high-level R&D program that will develop software- and hardware-implemented data security techniques that will be acceptable to the user community (or, at least, will be accredited for classified operational data); also required is a body of methodology for testing the adequacy of the security techniques, as implemented in data management and command and control systems.

The U.S. spends about $6 billion a year on intelligence. Suppose a unified data base were established for all WWMCCS information processing (as has been seriously considered) and its data security were completely and undetectably penetrated: the value of the information thus gained by the penetrator would compare quite favorably with the information gained during a year's operation of the U.S. intelligence agencies--and the information would be more reliable, more coherent, and in a machine-readable form. Proceeding with such a unified WWMCCS data base without extremely good data security techniques could result in a loss of intelligence competitiveness equivalent to $6 billion.

A major advantage of good data security techniques would be that they would reduce the tendency to overcompartmentalize information; or, at least, they would eliminate one of the excuses for doing so. As mentioned in the discussion on exercising, overconcern with compartmentalization can be a serious hindrance to dynamic force management.

B. TASK DESCRIPTIONS

Roadmap #5 is divided into the four tasks discussed below. The data security research program should consist of these tasks. (See also Volume III, the report on intelligence subsystem requirements.)

1. Data Security Software Development (5.1)

The objective of this program is to develop logic and algorithms for access control, auditing, file structuring, and monitoring; methodology for writing fail-safe executive systems that can be totally validated; methodology for introducing changes in validated executive systems; and software techniques for rapid shut-down of the system upon detection. Prototype software would be produced to permit testing of the adequacy of the techniques, and hardware design implications would be investigated.

2. Data Security Hardware Development (5.2)

The objective is to develop tamper-proof hardware implementations of security techniques: access control devices, failure monitoring logic (including detection of software failures or "confusion"), hardware for rapid declassification of the system upon imminent failure, and prevention of operation until failures or unauthorized access attempts have been resolved. This task includes development of hardware for cryptographic protection of data files, prototype development of the devices, and investigation of software implications.

3. Data Security Testing and Validating Methodology (5.3)

This task will develop methodology for testing data processing software and hardware which have implemented data security techniques to detect inadequacies in the security techniques implemented. The testing techniques used must satisfy national agencies tasked with data security. Included are analyses of vulnerability to covert penetration of various types of data processing hardware and software systems.

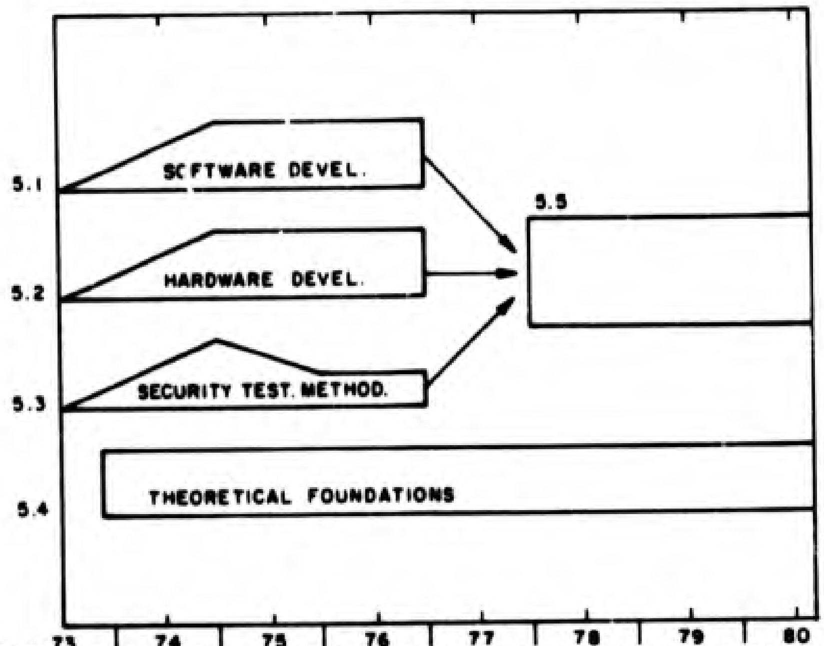
4. Theoretical Foundations of Data Security (5.4)

This is a research effort aimed at the formulation a theory of data security which is applied to derive measures of effectiveness for data security techniques, security criteria, and a methodology for optimization of software/hardware technique mixes for given operating environments, data bases, and threats.

All of the above tasks can be initiated concurrently; however, it is essential that close coordination and information exchange be maintained between the tasks. An intensive effort should be initiated in order to satisfy the data security needs of the command and control and data management systems planned for the 1975 time period. Subsequent to that, tasks 5.1 through 5.3 can be combined into task 5.5, allowing emphasis to be placed on security techniques or testing methodology as circumstances dictate.

C. FACTS AND FIGURES

Table XI-IV presents the pertinent facts and figures for Roadmap #5: Data Security. Figure XI-5 is a graphic display of Roadmap #5.



$, MILLIONS	FY74	FY75	FY76	FY77	FY78	FY79	FY80
MINIMUM	1.1	1.1	1.1	1.1	0.6	0.6	0.6
REASONABLE	2.1	2.1	2.1	2.1	1.0	1.0	1.0
CRASH	4.95	4.95	4.95	4.95	1.6	1.6	1.6

Figure XI-5. Roadmap #5: Data Security

TABLE XI-IV

ROADMAP #5: FACTS AND FIGURES

No.	Title	Fiscal Years Active	Peak Annual Rate, $\$10^6$ (No. of Efforts)			Existing Projects			Types	Overlap	
			Min.	Reas.	Crash	ID	Years	$\$10^6$		ID	%
5.1	Data Security Software Development	74 ↗ 75 → 77	0.25 (1)	0.50 (2)	2.0 (3)				EXD-30, AVD-45, LIB-25	15	15
5.2	Data Security Hardware Development	74 ↗ 75 → 77	0.50 (1)	1.00 (2)	2.00 (3)				EXD-50, EGD-50	15	25
5.3	Security Testing Methodology	74 ↗ 75 → 76 ↘ 77	0.25 (1)	0.40 (2)	0.60 (3)	2.3.1d	71-74	0.84	MTH-100	2	35
5.4	Theoretical Foundations	74 → 80	0.10 (1)	0.20 (1)	0.35 (2)				RES-100	2	25
5.5	Combined 5.1 through 5.3	78 → 80	0.50 (1)	0.80 (1)	1.25 (1)				EXD-30, AVD-20, LIB-10, MTH-40	15	25

VII. ROADMAP #6: HIGH-CAPACITY AIRBORNE COMPUTERS

A. BACKGROUND AND OBJECTIVES

Several planned Air Force command and control system programs, such as the Airborne Warning and Control System (AWACS) and the Airborne Command Post (ABNCP), have established requirements for high-capacity airborne data processing and storage capability for handling sensor data, for force control information, and for performing targeting or weapons control operations. These requirements can be expected to increase in the future as more command and control functions and larger data bases will be transferred to survivable airborne command post and data processing facilities.* In addition to performing complex calculations in real time, these systems will need to function as resource-sharing data management systems.

The current airborne data processors are relatively low capacity machines (typically 0.2 to 0.3 MIPS) oriented toward avionics use. Larger airborne processors, such as the IBM 4PI series, are in the 0.5 to 0.7 MIPS range. In the development stage, however, is the Navy's Advanced Avionics Digital Computer (AADC), which will be capable of 2.5 MIPS and can operate in the multiprocessor configuration. The AADC breadboard prototype is scheduled for 1975. It is expected to satisfy Navy data-processing needs for the decade of the 1980s. In the airborne bulk storage area, the RADC plated wire memory has a 10^8-bit capacity.

The above discussion indicates the need for further R&D to meet the computational requirements of dynamic force management aboard the AABNCP along the following major lines:

- Exploring promising new technologies (e.g., electro-optical), which may provide a breakthrough easily encompassing the AABNCP requirement;
- Accelerating development of airborne uniprocessor technology to more rapidly approach the AABNCP requirement;
- Developing efficient multiprocessor architectures and operating systems to achieve the AABNCP requirement via parallel operation; and

---

* Thirty million instructions per second (MIPS), or more, and 10^{11} to 10^{12} bits of bulk storage seem to be required for the Advanced Airborne Command Post (AABNCP).

PRECEDING PAGE BLANK - NOT FILMED

- Separating the AABNCP functions into relatively independent tasks which could feasibly be performed by independent computers.

In addition, the points made earlier on the likely divergence of the commercial computer hardware market from military interests hold also for other airborne and militarized computer components. Thus, Air Force R&D efforts (or cooperative DOD R&D efforts) are needed to provide efficient militarized versions of mass memories and computer peripherals, ranging from ground-transportable printers to airborne laser memories.

As illustrated by the functional breakdown of the AABNCP computer workload in Volume I, a lack of high-capacity airborne computer hardware could strongly limit the range of functions needed to support dynamic force management aboard the Advanced Airborne Command Post. At least as important, however, is the need for high-capacity airborne computer hardware to reduce software costs and complexity. This study reports data in Volume I that establish general trends in support of the following points:

- Overall system cost is generally minimized by procuring computer hardware with at least 50 percent to 100 percent more capacity than is absolutely necessary.

- The more the ratio of software-to-hardware cost increases (as it will markedly during the 1970s and 1980s), the more excess computing capacity one should procure to minimize the total cost.

- It is far more risky to err by procuring a computer that is too small than one that is too large. This is especially important since the initial sizing of the data-processing job often tends to underestimate its magnitude.

Of course, buying extra hardware does not eliminate the need for good software engineering thereafter. Careful configuration control must be maintained to realize properly the benefits of having extra hardware capability, as there are always strong Parkinsonian tendencies to absorb excess capacity with marginally useful tasks.

The above analysis can be applied to show the potential payoffs of high-capacity airborne computing power. If one assumes that the "ideal software" cost for the AABNCP of the 1970s is roughly equivalent to the $30 million estimated for converting the SACCS ground computer software to the upcoming WWMCCS machine, then providing enough airborne capability to make the AABNCP computer 50-percent loaded instead of 85-percent loaded saves a factor of about two in software costs, or roughly $30 million.

B. TASK DESCRIPTIONS

This Roadmap consists of the six tasks that are described below.

1. High-Capacity Airborne Computer (HICAP) Processor Development (6.1)

This task commences with a set of studies of the HICAP processor design objectives, environmental requirements, input-output interfaces, required software tasks (data management, operation in resource-shared manner), reliability and maintainability concepts, and implications of the command and control data processing and management of large data bases. The studies are directed to examine the findings of the AADC studies as well as findings of the CCIP-85 study.

The study results are amalgamated to specify the design of the HICAP prototype. Prototype construction is carried out, and tests of environmental hardness and ability to handle large airborne command post requirements are performed.

2. Airborne Laser Memory Development (6.2)

This task includes feasibility studies of producing a laser memory with a capacity of 10^{11} to 10^{12} bits and random access within microseconds. If found feasible from the engineering and environmental standpoints, a prototype will be constructed and tested in flight.

3. Solid-State Bulk Memory Development (6.3)

The development of a nuclear hardened solid-state bulk memory (10^{11} to 10^{12} bits) with access in microseconds that is qualified for airborne applications is the goal of this task. It includes feasibility and design studies of suitability of various solid-state technologies (magnetic domain wall motion devices, ferro-acoustics, charge-coupled devices, semiconductor LSI), construction of a prototype bulk memory using the most suitable technology and environmental testing.

4. High-Capacity Airborne Computer (HICAP) Software Studies (6.4)

Studies of software requirements for the HICAP processor and bulk memory for command and control applications, particularly multiprocessing executive systems, comprise this task. It includes determination of the suitability of existing programming languages for direct higher-order language (HOL) processing by the HICAP processor, as well as continued studies of software requirements for airborne command and control systems. There would be no actual development of operational software.

5. Airborne Computer Technology Research (6.5)

This task includes continuing studies of, and stimulation of, research in computer components, manufacturing techniques, packaging, and architectural features for improving the HICAP designs. The foundation would be laid for producing higher capacity airborne computer systems beyond the HICAP time frame.

6. Militarized Peripherals (6.6)

Task 6.6 involves continuing studies of, and stimulation of, research and development of computer peripherals (mass memories, printers, displays, etc.) for use in airborne or field operations.

7. Basic Computing Technology (6.7)

Continuing studies of, and stimulation of, research into promising new methods of physical representation and processing of information (electro-optical, atomic-molecular, etc.) are the goals of this task.

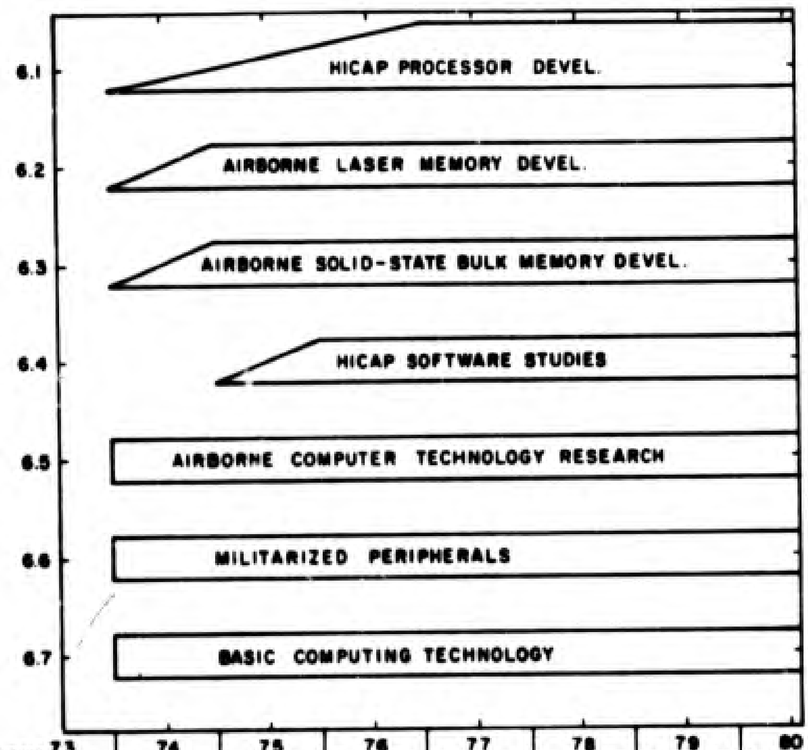
C. FACTS AND FIGURES

Table XI-V gives the facts and figures for Roadmap #6. These data are presented pictorially in Figure XI-6, along with some of the funding information.

TABLE XI-V

ROADMAP #6: FACTS AND FIGURES

No.	Title	Fiscal Years Active	Peak Annual Rate, $\$10^6$ (No. of Efforts)			Existing Projects			Types	Overlap	
			Min.	Reas.	Crash	ID	Years	$\$10^6$		ID	%
6.1	HICAP Processor Development	74 ↗ 76 → 77 ↘ 80	1.0 (1)	2.0 (2)	3.0 (2)	2.2.4 2.2.3 2.2.5	70-77 70-77 70-77	1.0 2.9 23.5	RQA-20, LIB-80	8-20 7-40 15-10	11-20 4-20
6.2	Airborne Laser Memory Development	74 ↗ 75 → 80	0.5 (1)	1.0 (2)	2.0 (2)	2.3.1 a-c	71-75	5.95	EXD-30, AVD-70	7-30 8-10 13-30	15-10 4-20
6.3	Airborne Solid-State Bulk Memory Devel.	74 ↗ 75 → 80	0.4 (1)	0.8 (2)	1.5 (2)				RES-20, EGD-80	7-30 8-10 13-30	15-10 4-20
6.4	HICAP Software Studies	75 ↗ 76 → 80	0.1 (1)	0.25 (2)	0.4 (3)				AVD-100	2-25 17-20 8-40	5-25
6.5	Airborne Computer Technology Research	74 → 80	0.2 (1)	0.5 (2)	0.8 (2)				RES-50, EXD & AVD-50	7-20 15-10 5-10	4-30
6.6	Militarized Peripherals	74 → 80	0.4 (1)	0.8 (2)	1.5 (2)				EXD-50, AVD-50		
6.7	Basic Computing Technology	74 → 80	0.5 (2)	2.0 (4)	5.0 (6)				RES-100		



$,MILLIONS	FY 74	FY 75	FY 76	FY 77	FY 78	FY 79	FY 80
MINIMUM	3.0	3.1	3.1	3.1	3.1	3.1	3.1
REASONABLE	7.1	7.35	7.35	7.35	7.35	7.35	7.35
CRASH	13.8	14.2	14.2	14.2	14.2	14.2	14.2

Figure XI-6. Roadmap #6: High-Capacity Airborne Computers

VIII. ROADMAP #7: MULTISOURCE DATA FUSION

A. BACKGROUND AND OBJECTIVES

A current problem, particularly in tactical command and control is an inability to thoroughly exploit a superabundance of input data from various sources--voice, text, radar, imagery, acoustic sensor data, and the like.

The total volume of input and the slowness of manual processing makes it impossible to detect significant patterns in operational trends, even with current manpower levels. With the increasing data-gathering capabilities and the lower Air Force manpower levels predicted for the future, there is a strong need for effective automated aids to recognize such patterns in data, or at least to indicate that they need human review.

In Vietnam, the United States has had an unprecedented amount of information about operations along the Ho Chi Minh Trail. Intelligence from a wide variety of sensors, visual pilot reports, other voice reports, and various types of photographs, coupled with a host of background reports, provided a data base capable of predicting every move along the trail. Yet, the tactical command and control system was unable to match optimum weapon systems with target systems, traffic was always heavy when ordnance was not available, or other factors seemed to keep the command and control response out of synchronization with enemy actions. One of the key contributors to this situation was the inability to time-correlate the masses of multisource data within the time window required for optimum operational responses. Clearly, automated methods of relating multisource, perishable data with historical data would greatly enhance the credibility of the entire command and control effort.

The TIPI (Tactical Intelligence Processing and Interpretation) system currently under development will provide an initial step toward such a capability, primarily on the basis of existing information processing and display aids. It will also provide a framework into which advanced techniques can be placed. To exploit this framework properly, more fundamental studies are necessary to develop and evaluate advanced automated aids for such tasks as:

- Reading and interpreting input data correctly;
- Correlating related data items;
- Automatically maintaining the data files;
- Eliminating or rejecting nonrelated or trivial data;

- Understanding the impact and meaning of the correlated data;
- Developing alternatives through synthesis based upon a set of objectives;
- Selecting an alternative based upon current constraints of the system; and
- Feeding the results to a human for decision-making.

To properly assess the benefits of such studies, one must consider the billions of dollars spent in gathering such information during the last decade (e.g., $4 billion for Igloo White), and the additional billions which will undoubtedly be spent on data gathering in the 1980s. Without advances such as the above, and with likely future Air Force manpower constraints, a large fraction of the value of these data will never be recovered.

The total fusion process problem may not be automated, but there are several areas which appear promising for further development. Among these are:

- Eliminating the requirement for manual or interactive message handling tasks;
- Eliminating the requirement for manual or interactive update and maintenance of data files, except for the data man creates as a result of reviewing the data base, computed solutions, or whatever;
- Eliminating the requirement for manual or interactive supply or reiteration of search parameters necessary to correlate data;
- Eliminating the requirement for manual or interactive synthesis of the correlated data; and
- Eliminating the requirement for manual or interactive generation of products.

The ultimate objective of automating the fusion process would be to reorient man's function to one of decision-making. Some of the characteristics an automated fusion model would have to exhibit were listed above as capabilities to be provided with advanced automated aids (e.g., reading and interpreting input data correctly, correlating related data items, etc.).

B. TASK DESCRIPTIONS

This section describes the three specific tasks of this Roadmap.

1. Fusion Usage Study (7.1)

To serve as a base for the other R&D activities in fusion and image processing (Roadmap #10), studies are needed which identify and categorize common types of Air Force fusion processes, and analyze the procedures by which human analysts perform fusion of multisource data.

2. Develop Algorithms (7.2)

Based on the usage studies above, algorithms and advanced information display capabilities can be developed to aid Air Force C&C personnel in data correlation, synthesis, and product generation.

3. Relational Data Files Techniques (7.3)

Based on the usage studies in 7.1, techniques of categorizing relevant information and associated processing algorithms can be developed to facilitate and partially automate the process of drawing inferences from textual data to support C&C decision-making.

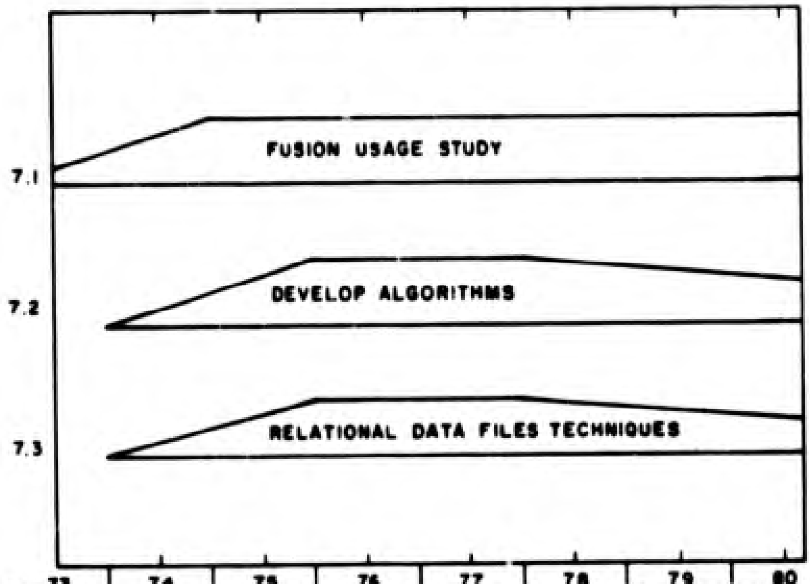
C. FACTS AND FIGURES

Table XI-VI gives the facts and figures for Roadmap #7. Some of these data are presented pictorially in Figure XI-7, along with funding information.

TABLE XI-VI

ROADMAP #7: FACTS AND FIGURES

No.	Title	Fiscal Years Active	Peak Annual Rate, $\$10^6$ (No. of Efforts)			Existing Projects			Types
			Min.	Reas.	Crash	ID	Years	$\$10^6$	
7.1	Fusion Usage Study	73 ↗ 75 → 80	0.2 (1)	0.4 (1)	0.6 (2)	7.7.1.5 4.3			IGA-100
7.2	Develop Algorithms	74 ↗ 76 → 78 ↘ 80	0.6 (1)	2.0 (2)	4.0 (2)	2.2.4.1 2.2.5.1 2.2.6.1 4.2		0.02 0.04 0.04	EXD-25, AVD-25, OPD-20, IGA-30
7.3	Relational Data Files Techniques	74 ↗ 76 → 78 ↘ 80	0.4 (1)	1.8 (1)	2.0 (2)	2.2.6.1 2.4.8 2.5.1		0.04	EXD-20, AVD-20, EGD-20, OPD-10, IGA-30



$. MILLIONS	FY74	FY75	FY76	FY77	FY78	FY79	FY80
MINIMUM	1.2	1.2	1.2	1.2	1.2	1.2	1.2
REASONABLE	4.2	4.2	4.2	4.2	4.2	4.2	4.2
CRASH	6.6	6.6	6.6	6.6	6.6	6.6	6.6

Figure XI-7. Roadmap #7: Multisource Data Fusion

IX. ROADMAP #8: COMMUNICATIONS

The recently completed mission analysis on communications to support theater air command and control[14] identified much of the R&D required in this area. A continuing liaison with that effort should be maintained to ensure support of the command and control information-processing requirements highlighted in this study.

X. ROADMAP #9: SOURCE DATA AUTOMATION

A. BACKGROUND AND OBJECTIVES

Dynamic force management is as susceptible to the "garbage in, garbage out" phenomenon as any other process. It depends strongly on the timely receipt of accurate information from on-site personnel -- often a pilot who may be preoccupied with other pressing concerns. Current experience, particularly in Vietnam, indicates that the information received is often delayed and inconsistent, resulting in lost opportunities, inappropriate C&C decisions, and general inefficiency of the C&C systems as data consistency committees evolve to compensate for the erroneous inputs.

The most reliable source of data for a command and control system continues to be visual reports from the operators themselves. No number of intelligence reports, sensor reports, or paper reports can refute a live voice report from a pilot saying that he has sighted the enemy and that they are engaged in some particular activity. Unfortunately, these reports are very unresponsive to operational needs because of the time lag and other inefficiencies of voice transmission, including the large number of errors introduced. There is an additional time lag if manual encryption is required, multiplying the opportunities for errors. The above factors combine to cause either 1) overreliance on personal input data that are grossly in error or out of date, and/or 2) loss of confidence in current data that may be the best available.

Advanced computer technology can provide improved source data automation capabilities which could improve speed and accuracy, reliability, cost, weight, and volume characteristics. There would be additional capabilities for preprocessing inputs. For example, PLRACTA[15] offers an example of a unified, advanced source data automation approach to aircraft position reporting, which is a largely manual, time-consuming, and error-prone process today. Similar source data automation approaches or extensions could be applied to event reporting, using special keyboards with preformatted message generators, head-mounted locators, and the like.

Voice data input is likely to mature into operationally acceptable capabilities by the 1980s, at least with reasonable restrictions on vocabularies, which would present no major drawback to their use in C&C systems. The Advanced Research Projects Agency is initiating a major thrust in the area, at a level of roughly $5 million per year; the Air Force should initiate counterpart efforts to increase the likelihood that the recognition problems ARPA researchers solve are those of prime interest to the

Air Force (e.g., voice recognition with suitably realistic vocabularies and noise backgrounds). The Air Force should also try to assure that the ARPA results, as well as others, are promptly developed into Air Force operational capabilities.

In general, there is a continuing need not only for Air Force research into providing new source data automation capabilities (such as those mentioned above, computer graphics, optical character recognition, etc.), but also for counterpart efforts to ensure that the new capabilities can and promptly do solve Air Force operational problems in the source data automation area. Otherwise, input data delays and errors will result in command and control breakdowns, with serious adverse effects in overall system performance, national image, rejection of Air Force initiatives, and lack of confidence in the C&C system. On the other hand, appropriate automation and preprocessing can eliminate the need for many clerical functions and permit the Air Force to actually increase its effectiveness in an era of decreasing manpower availability.

B. TASK DESCRIPTIONS

Roadmap #9 is subdivided into five activities. These are described below.

1. Command and Control Source Data Automation Usage Study (9.1)

The opportunities to saturate operational channels with information are expanding with the introduction of each new sensor system. It is most important that all other tasks in this area be preceded by a usage study to ensure that each effort is linked to operational effectiveness.

2. Source Data Automation Technical Assessment and Development (9.2 and 9.3)

These two efforts should be accomplished in an integrated fashion over the range of activities listed below.

a. Low-Cost CRT Terminals -- In 1970, the use of simple CRT units for source data input and query-and-response or question-and-answer operations is spreading rapidly in the commercial area. This is assumed to be partly the result of the cost reduction achieved recently and partly a byproduct of the growing use of time-sharing, remote access, and transaction-oriented processing. This line item represents studies to be made in utilizing these consoles; in conjunction with the output from the standardization studies, it is estimated to lead to development of a standardized low-cost CRT console for USAF use in connection with many different C&C computer applications and many different computers. A militarized model and an airborne model are also assumed to be developed and procured. The absence of such standardization will result in excessive programming costs for accommodation of the differences, time delays for program transferability, and likelihood of errors in transferring. Secondly, the use of multiple different consoles will limit the potential cost savings that would be available by mass production by competing sources.

b. Graphic Display Studies -- Commercial graphic display development has produced the BR 90 type of CRT display system as an outgrowth of previous military developments. Many low-cost displays based on the newer storage tubes have been developed. The major areas of development are the lowering of cost for the refreshing type of displays (e.g., BR 90) and the increase in flexibility of the erase (add-subtract) capabilities of the storage tube display. Commercial efforts, supported in part by military orders and industry R&D, will produce improved displays: by 1975, a full color display with three-dimensional rotation and translation capabilities should be available at more reasonable costs (the LSD-1 has these capabilities now at $200,000). By 1980, a helmet-mounted display could be produced for operating on stored data and inputs of new data. This would eliminate the bulk and weight of the display units (the total weight would be on the order of pounds). By 1989 or 1990, low-cost digital storage may be available that would allow the storage and retrieval of completely digitized images at costs similar to that of paper and paper storage devices. At this point, the nature of source data automation might change from predominantly text and table data to predominantly vector-picture types of data. Software studies are included.

c. Military Graphic Display Development -- Because of the intense commercial development in the graphics console area, military development is required only for meeting special problems such as security, airborne operation (including radiation and EMP), tactical size-weight-volume, and for all uses the important problem of reliability. It is assumed that such units will be required for AWACS, AABNCP, TAC, TIPI, and survivable command posts.

d. Data Security Studies for Remote Terminals -- This line item covers the problems of remote console operation in a mixed security computer system used for C&C purposes. Suitable crypto devices, preferably capable of asynchronous operation, and automatic user identification devices (whether by voice, handwriting, fingerprints, blood characteristics, odor, visual image, or other characteristics) may be used. The development of security within the computer and computer storage is not part of this item since it is a major area covered elsewhere.

e. Optical Character Reader (OCR) Input Devices -- Prior military support has contributed significantly to the development of OCR devices, many of which are now available commerically. Because of the extensive input requirements for textual material for intelligence and C&C activities, continuing military support is needed, particularly in the areas of reading low-quality typewritten material, printed materials, foreign character sets, and mixed texts and diagrams. The primary limitation on the widespread use of OCR input is the cost for the electronics. This cost is expected to drop significantly over the next 15 years. One important area is that of error-correcting equipment that uses the redundancy of text for making corrections and checking the equipment. Another significant advance is the use of adaptive mechanisms that can be "taught" to read new fonts or recognize old fonts.

f. Digitizers - Plotters -- The inputs of data such as map data and photointerpretation data can be achieved by digitizers. Major development of highly accurate digitizers is considered to be a byproduct of the military photointerpretation system requirements. Continued military support is assumed to be required.

Plotters, on the other hand, have had significant advancements from commercial activities (although some of this resulted indirectly from military requirements). The semiconductor industry in particular has requirements for highly accurate plotters for mask design. For many future C&C requirements, plotters are far too slow and CRT presentation methods are expected to be used.

g. Sensing Systems -- The use of a multiplicity of sensing devices in a network to monitor or control activities is an area of growing use. As an example, thousands of sensor elements have been integrated into a monitor-control system for a commercial enterprise. The design and fabrication of such systems can generally be achieved commercially for most military systems. (Those like Igloo White are not included in this general input discussion). One special case is the development of multiplexing systems for aircraft to reduce cable weight and improve reliability. Electromagnetic pulse considerations could be another area of military concern.

3. Voice Input-Output (9.4)

Commercial equipment to produce speech output is currently available. The vocabulary is limited, however, and the equipment is relatively expensive. It is assumed that commercial development will provide continued advancement in this area. One expected military use will be that of providing added paths of information to a console operator. In particular, the speech channel can be used to alert to unusual situations without requiring excessive operator attention.

The use of speech input to computer systems will be one of the more important fundamental advances to be made in C&C systems over the next 15 years. This method is rapidly becoming a reality for very simple inputs, such as numbers. A major research program by ARPA is likely to produce major advances in the next five years. Speech input is considered to be highly advantageous, in that the speed of speech input can be several times that of typewritten input. The use of speech input for control of a device such as a graphics console or a text-editing console frees the hands for other functions and should make the overall operations more simple, efficient, and accurate.

Based on the ARPA program and the current military program, extensive user studies should be started in the 1975 time period leading to projects to produce equipment for three categories of operation as follows:

- The input of numerical and selected small vocabulary data by a selected or trained speaker; field use by 1980 should be reasonable.
- The input of computer control data and programming data to be used by programmers in inputting program instructions, editing programs, and debugging programs. Such a capability could be in field use by 1983.
- The input of free text with a large vocabulary such as is used in producing intelligence reports, briefings, engineering data, status reports, etc. This capability will result in reduction in time delays in the overall system operations and in reduced costs (for the inputting, creation, and editing operations). Better reports, more complete and more accurate reports should also be a result. A first mode can be expected by 1985 (or before, if development is emphasized).

By combining the results of the speech input program, the automatic translation program, and the speech output devices, a real time language-to-language translation device can be obtained. By obviating the need for an interpretation and the possible bias of an interpreter, significant improvement in high level politico-military C&C problems could be made. Such a system could be applied to "hot line" communications, for example, to allow direct speech between heads of state.

4. Source Data Automation Standards (9.5)

Based on an extrapolation of past history, the question of source data automation standards will be a continuing one for the period up to 1990. A cohesive program to replace the multiple efforts currently in progress is expected. This study will cover such diversified matters as the standards for the following:

- CRT text/table consoles including codes, line lengths, number of lines, transmission standards, character and symbol sets and shapes, control codes, etc;
- Graphic input-output consoles (including the factor for text given above), vector representation, number of vectors, accuracy, resolution, repeatability, control

keys, light pens, tablets, balls, joy sticks, and data structures to represent graphic data;

- Interactive job control language -- the standardization of language for starting up the console, identifying the user and privileges, calling up jobs, interrupting jobs, terminating jobs, and signing off, as well as standards for accounting for resources, etc;

- Formats for messages between computer and between users such as those involved in the ARPA net type of communications and formatting of reports by people to interactive data bases;

- Standards for data security to input-output remote CRT terminals used in mixed security systems;

- Standards for speech output and speech input, verbal commands for editing, correcting, and deleting verbal inputs and output;

- Standards for electrical and mechanical interfaces to communication lines and computers;

- Standards on reliability and repairability of devices, the allowable error rates (including through communications), and the error correction methods (ALA-ARPA NET), etc;

- Standards for documentation for the above areas and for training and tutoring personnel;

- Standards for the degree of modularity and interchangeability between source data devices of various manufacturers;

- Standards for digitizers, plotters, and similar devices; and

- Standards for sensing systems to convert and detect data automatically or simultaneously such as inventory bin counters, transaction counters, etc.

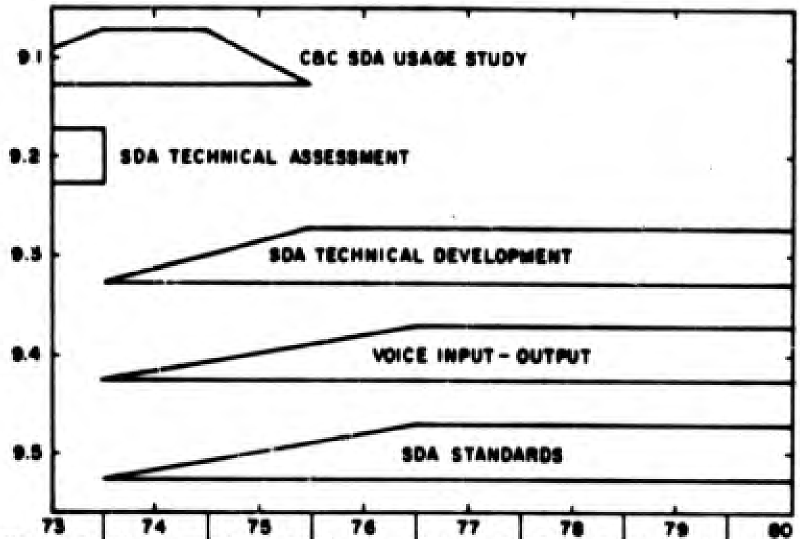
C. FACTS AND FIGURES

Table XI-VII gives the facts and figures for Roadmap #9: Source Data Automation. Figure XI-8 is a graphic display of the Roadmap.

TABLE XI-VII

ROADMAP #9: FACTS AND FIGURES

No.	Title	Fiscal Years Active	Peak Annual Rate, $\$10^6$ (No. of Efforts)			Existing Projects			Types
			Min.	Reas.	Crash	ID	Years	$\$10^6$	
9.1	SDA Usage Study	73 ↗ 74 → 75 ↘ 76	0.3 (1)	0.6 (1)	0.8 (1)				IGA-100
9.2	SDA Technical Assessment	73 → 74	0.2 (1)	0.4 (1)	0.6 (1)				LIB-100
9.3	SDA Technical Development	74 ↗ 76 → 80	0.75 (1)	1.5 (1)	4.0 (1)				EXD-50, AVD-50
9.4	Voice Input-Output	74 ↗ 77 → 80	1.0 (1)	3.0 (2)	10.0 (4)				EXD, AVD-100
9.5	SDA Standards	74 ↗ 77 → 80	0.4 (1)	1.0 (1)	2.0 (1)				SVC-100



$, MILLIONS	FY 74	FY 75	FY 76	FY 77	FY 78	FY 79	FY 80
MINIMUM	2.65	2.45	2.45	2.15	2.15	2.15	2.15
REASONABLE	6.5	6.1	6.1	5.5	5.5	5.5	5.5
CRASH	17.40	16.80	16.80	16.0	16.0	16.0	16.0

Figure XI-8. Roadmap #9: Source Data Automation

XI. ROADMAP #10: IMAGE DATA PROCESSING

A. BACKGROUND AND OBJECTIVES

An automated image-processing capability to identify a truck, radome, or missile site under a wide variety of shadow patterns is both of extreme potential value to Air Force operations and of extreme technical difficulty. The Air Force should not expect any miracles of automated pattern recognition to sweep away the image processing problem, even by the 1980s.[5] Specific, mission-oriented, image-processing functions can be defined, such as change detection, outline recognition, and semiautomated aids to photointerpreters on realistic image inputs. Further R&D on these functions can produce results leading not only to improved C&C capabilities but also to a richer base of data and insights upon which future basic research efforts can build.

Many billions of dollars have been spent on vehicles, sensors, and operations involving image data collection, and additional billions of dollars are anticipated to be spent during the 1970s and 1980s. An appreciable fraction of the expense is devoted to image processing. An even higher fraction of the available information on the images is unrecovered because of time and manpower limitations. Additional R&D would be likely to pay for itself in reducing the large total costs and increasing the fraction of useful information recovered, with additional benefits of timeliness in critical situations, straightforward expandability to higher workloads, and reduced demand for increasingly scarce manpower.

1. Previous Efforts

Many attempts to define algorithms and devices for image data processing have been made during the past twenty years with but limited success. In the area of height finding using stereo pair photography, an automatic countour-following device has been successfully demonstrated. Alignment of essentially identical photographs by the "cross correlation" process has been successfully accomplished in the contour tracker and in other photographic processing devices. Both digital and analog change detection devices for high resolution side-looking radar have been successfully demonstrated. The rate at which change detection has been done, however, limits the economic application of the existing devices. General pattern recognition is in the research phase with many different theoretic approaches and is enjoying limited general success. In general, commercial activities appear to be attempts at application of previous military-supported developments and have very limited funding support.

2. Current Efforts

RADC is continuing to develop the analog optical correlation processor for change detection, while one commercial firm is continuing to develop digital change detection under its own sponsorship.

3. Observational Foundation

The R&D activities in this Roadmap should be started only after a solid observational base of C&C usage requirements from Roadmaps #1 and #7 has been gathered and analyzed.

B. TASK DESCRIPTIONS

This Roadmap consists of five tasks which are described below.

1. Semiautomatic CRT/Digital Computer Aids to Photo-Interpretation (10.1)

This line item covers investigation of means for reducing the time delays and the costs of performing photointerpretation. The activities at The RAND Corporation under the support of ARPA could result in a field test model by 1974. Users studies should begin essentially immediately for the field test. One key element of the overall system must be an adequate text-editing system since it has been found that a major fraction of the analyst's time has been consumed in preparing and editing the textual material of the final report. Recent improvements have been made with PACER, AIDS, and the TIPI program. An improved storage system for the analog electrical pictures is a likely development required for field use of the system. A continual development program extending into the 1980-1990 time period is probably required to achieve the full potential of this approach. Some non-imaging processing support to planning of missions and use of data such as the inertial system position data in processing and selecting film material can also reduce time delays. Storage of film images and indexing of film images is not covered by this Roadmap.

2. Change Detection R&D (10.2)

Change detection techniques for side-tracking synthetic aperture array radars have been demonstrated. User studies have been made. An analog and a digital computer model exist in experimental form.

By 1974, a field test model of a change detection system should be developed. Operational use by 1980 can be expected. A laser scanner change detection model can be tested at the same time, and operational use also expected by 1980.

An IR scanner change detection system should be experimentally tested to determine if results warrant full scale development. The utility of change detection for other sensors should also be investigated.

3. Area Image Outline Recognition R&D (10.3)

This line item covers development of techniques and equipment for defining the outline of areas. Currently, the ESSA program is utilizing aircraft to take multispectral photographs of terrain and using computer programs to define areas. Examples are water, wooded areas, corn fields, etc. A satellite system ERTS A and ERTS B, is intended to be flight tested. By 1976 a combination of these techniques, together with initial results from the other pattern recognition work, could lead to an operational use by 1981.

4. Theoretical Research Studies (10.4)

An expansion in theoretical research studies of image data processing in the area of pattern recognition should be planned for the near future. Larger and faster computers should be made available to support this work, perhaps via the ARPA network and the ILLIAC IV computer. Better interactive capabilities such as the "TENEX resource" should be provided via the ARPA network and a large fast solid-state mass memory provided to reduce inefficiencies in the research program. Data compression of images is a pertinent research area.

5. Object Description Pattern Recognition R&D (10.5)

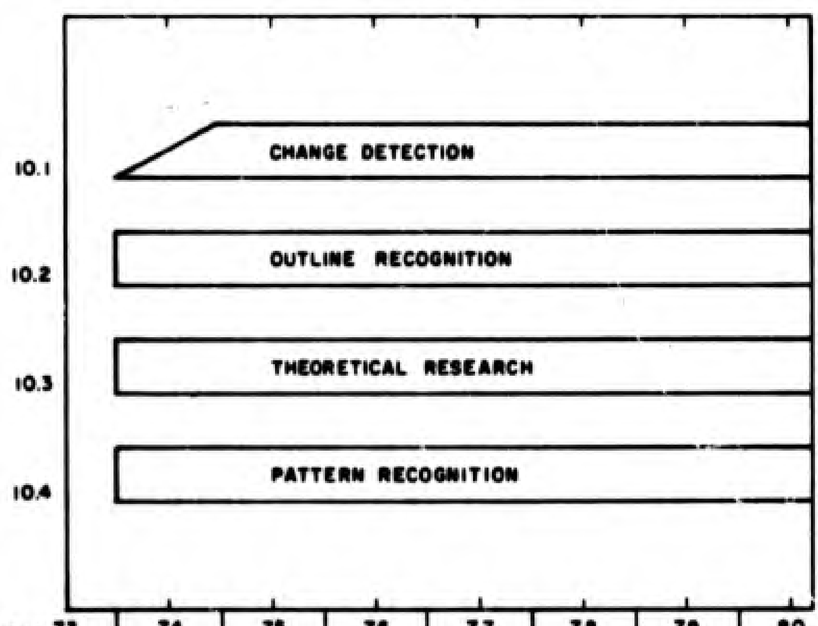
The pattern recognition field is one of the potential areas for long-term significant advancements in imagery processing for command and control. The automatic target detection characteristics of pattern recognition applied to imagery such as photoreconnaissance are the main interest in this Roadmap (pattern recognition may also apply to a number of other C&C Roadmaps). Use can be made of the capability of finding and tracking mobile targets such as mobile ICBM and mobile tactical targets.

By 1975, increased R&D effort should allow design of a model of such a system. A goal should be a six-foot resolution capability. It is assumed that the extensive data processing job required could be aided by an associative array processor or other specialized processors. Design could start about 1975, and the first model could be tested in 1978. A field test could be completed by 1982 and initial operational use started by 1983 for specialized problems.

Concurrent in 1975 with the above program, appropriate basic R&D in the areas of multidimensional pattern recognition techniques, heuristic pattern recognition machines, and self-teaching (adaptive) neural net types of machines should be initiated along with any other significant candidates that are discovered by theoretical research. This program should provide a basis for improved operational capabilities for detecting, classifying, and identifying military targets automatically by the end of the 1980 decade.

C. FACTS AND FIGURES

Table XI-VIII presents pertinent facts and figures for Roadmap #10: Image Data Processing. Figure XI-9 is a graphic display of Roadmap #10.



$, MILLIONS	FY 74	FY 75	FY 76	FY 77	FY 78	FY 79	FY 80
MINIMUM	1.0	1.0	1.0	1.0	1.0	1.0	1.0
REASONABLE	3.0	3.0	3.0	3.0	3.0	3.0	3.0
CRASH	14.0	14.0	14.0	14.0	14.0	14.0	14.0

Figure XI-9. Roadmap #10: Image Data Processing

TABLE XI-VIII

ROADMAP #10: FACTS AND FIGURES

No.	Title	Fiscal Years Active	Peak Annual Rate, $\$10^6$ (No. of Efforts)			Existing Projects		Types
			Min.	Reas.	Crash	ID	Years $\$10^6$	
10.1	Change Detection	74 ↗ 75 — 80	0.4 (1)	1.2 (2)	4.0 (3)			AVD-100
10.2	Outline Recognition	74 → 80	0.1 (1)	0.4 (2)	2.0 (3)			AVD-100
10.3	Theoretical Research	74 → 80	0.2 (1)	0.4 (2)	2.0 (3)			RES-100
10.4	Pattern Recognition	74 → 80	0.3 (1)	1.0 (2)	6.0 (3)	5581	72-74 1.0	EXD-100

XII. ROADMAP #11: COMPUTER SYSTEM PERFORMANCE

A. BACKGROUND AND OBJECTIVES

The discussions on associative and parallel processing in Roadmap #12 indicate a future need for computer system performance analysis to ensure that a reasonable number of net MIPS would be applied to the problem at hand from all of the gross MIPS available. Even today, though, many uniprocessors run at about 30 percent of their potential. Performance analysis techniques on such computers have usually increased efficiency by 25 to 30 percent and sometimes by factors of four and five.[16] Applications programs provide another source of savings; for example, one joint RAND-Air Force effort yielded a 40-percent savings on a heavily used Air Force program. Even though hardware is getting much cheaper, the opportunities for inefficient operation are increasing just about as fast.

Concentration on developing and using advanced computer system performance techniques, such as hardware and software measurement devices, statistical analysis, and simulation, will provide the Air Force with a number of direct and indirect benefits in the following areas (the first two apply Air Force-wide; the last three are more significant):

1. Economy of Operations

The Air Force annual computer hardware budget is roughly $300 million.[17] If savings of 25 percent can be applied on only 25 percent of the Air Force's computers,* the annual savings are still about $20 million. The major avenue to savings is through providing an alternative to upgrading to a higher-capacity machine, which also saves the Air Force from the attendant conversion problems.

2. Better Procurement Procedures

Even with current computers, a vendor having advance knowledge of the nature of benchmark applications programs can tune his system to

*Peacetime efficiency is not a major concern for most C&C computers, which must have excess capacity to handle emergencies.

process them 50 to 100 percent faster. A computer system procured on such a basis is roughly equivalent to specifying a requirement for a Mach 2.5 fighter, and then buying a Mach 1.5 fighter which has demonstrated a capability for a Mach 2.5 power dive. Future computer systems will have tuning improvement factors of at least three or four.

3. Software Relief During System Development

As discussed in Volume I and elsewhere, software costs escalate radically as hardware capacity is strained. Performance analysis techniques can reduce the problem significantly: for example, IBM used simulation techniques to reduce computer speed requirements on their Skylab project from 115 percent to 75 percent of capacity and fast memory requirements from 60,000 words to 12,000, without significant loss of responsiveness.

4. Management and Exercise Benefits

While a C&C computer collects and analyzes data on its own performance, it is also collecting and analyzing a record of overall C&C system performance. The SAGE regenerative recording capability[18] was found to be extremely useful to managers, providing insights on both exercises and actual operations.

5. Contributions to Certification and Data Security Problems

Certifying a computer system implies being able to measure what it is doing. Yet, many systems are still designed or implemented simply to maximize performance or to minimize response time, with little or no attention to facilitating measurement. Only later does the necessity for measurement arise, resulting in costly retrofits and poorer performance.[5]

B. TASK DESCRIPTIONS

Roadmap #11 is subdivided into nine activities. These are described below.

1. Central Help (11.1)

The "Central Performance Help Group" program maintains a cadre of performance analysis experts and necessary monitors. The primary mode of operation is for two or more analysts to go into the field for short periods (one week to six months) to help installations with system performance analysis. However, they will be available on a rotating basis to provide continuing aid for particularly important systems such as WWMCCS. While resident at a central location, these experts will develop techniques (such as in task 11.7 and coordinate other C&C performance analysis activities.

2. Controlled Experiments (11.2)

Performing controlled experiments on interactive systems is quite difficult because of the randomness of human response times. This project is needed to create operationally feasible ways to determine meaningful load descriptions, and load the system in a controlled manner as demanded by test designs. Part "a" will deal with the problem in the current environment, while part "b" will consider the added problems caused by such future techniques as new sensors, computer networks, and distributed processing.

3. Operating System and Compiler Aids (11.3)

Current software performance measurement aids are added to operating systems and compilers with no effort at integration. This project will investigate the aids now in use and consider which ones should be made available when integrated into the basic system software. Special problems exist in C&C since the aids must not degrade reliability or performance, and must operate in very dynamic environments.

4. Products Requirements (11.4)

This program begins by specifying those aids to C&C system performance analysis which are already known to have a need for integration. After appropriate research projects have been completed, the specification activity will be expanded to include additional items. This effort will decrease as the rest of the computer industry becomes increasingly involved. However, the Air Force must continue its C&C involvement to avoid specifications from deviating too far from utility in such systems.

5. Hardware Aids (11.5)

Software performance monitors introduce an overhead which is often unacceptable in C&C systems. In addition, certain data cannot be obtained through strict software (or hardware) monitoring. Hardware aids to monitor C&C software may include special registers to turn software monitors on and off and integrated hardware monitors which may be interrogated with software.

6. Performance on Secure Systems (11.6)

This task involves the analysis and performance improvement of potential multilevel secure systems. Security safeguards in C&C systems seem to have the universal characteristic of heavy overhead and offer a potential area for widespread performance improvement when effort is devoted to their redesign.

7. Measurements for New Techniques (11.7)

The software-hardware technological environment of computers is continually changing, and the Air Force must devote effort to the measurement of the new technologies with respect to C&C systems. In the very near term, cache memories and very large capacity storage devices ($\geq 10^{12}$ characters) present a challenge. Networks of C&C systems and/or microprogrammed C&C systems may appear, and measurement techniques will be needed for these systems. Results from this program will be important in the Air Force's determination of appropriate industry product requirements.

8. Represent System (11.8)

Project 11.8 will provide a series of documented ways to represent system interactions. New simulation programs have a tendency to assume that certain types of representations will be useful, meaningful, and measurable; little experimental testing is done, particularly in C&C systems. This program's objective is to develop techniques so that each simulation analyst will not be forced either to accept unvalidated assumptions or to invest in extensive investigation.

9. Workshops (11.9)

A number of performance workshops are held throughout the computer industry, and C&C personnel need to attend those meetings. Many areas of R&D for performance analysis have not been noted in this Roadmap because access to such developments has been assumed for C&C performance analysts. They must be familiar with performance-sensitive areas upon which to concentrate effort, standard measurement techniques, and data reduction techniques developed by others in the industry.

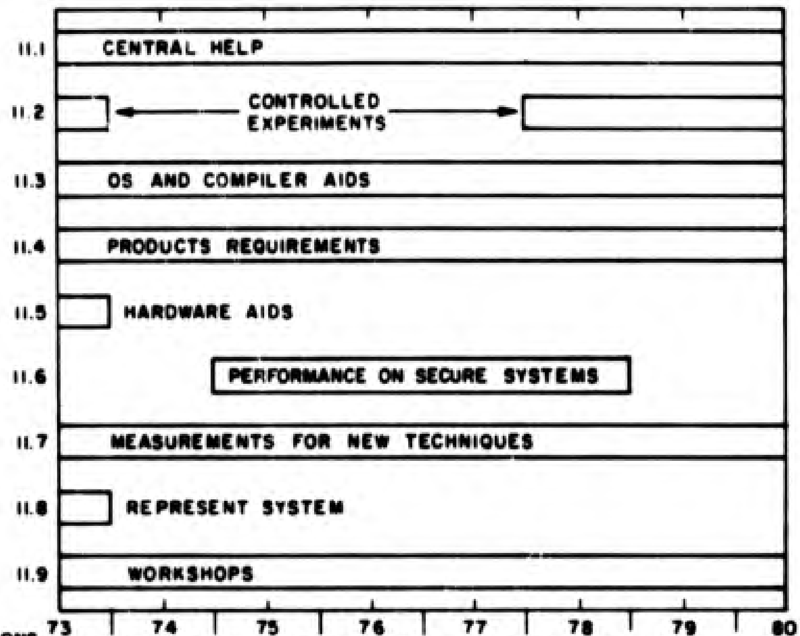
C. FACTS AND FIGURES

Table XI-IX presents the pertinent facts and figures for Roadmap #11: Computer System Performance. Figure XI-10 is a graphic display of the Roadmap.

TABLE XI-IX

ROADMAP #11: FACTS AND FIGURES

No.	Title	Fiscal Years Active	Peak Annual Rate, $\$10^6$ (No. of Efforts)			Existing Projects			Types
			Min.	Reas.	Crash	ID	Years	$\$10^6$	
11.1	Central Help	74→80	0.2 (1)	0.4 (1)	0.5 (1)				SVC-100
11.2	Controlled Exper.	73→75, 78→80	0.2 (1)	0.5 (2)	0.6 (4)				
11.3	OS and Compiler Aids	73→80	0.2 (1)	0.4 (1)	0.6 (3)				OPD-100
11.4	Products Requirements	74→80	0.2 (1)	0.3 (1)	0.4 (1)				STD-100
11.5	Hardware Aids	73→74	0.25 (1)	0.5 (2)	0.6 (3)				OPD-100
11.6	Performance on Secure Systems	75→79	0.2 (1)	0.5 (2)	1.0 (4)				OPD-100
11.7	Measurements for New Techniques	73→80	0.4 (1)	0.8 (1)	1.5 (2)				EGD-100
11.8	Represent System	73→74	0.2 (2)	0.4 (3)	0.6 (4)				EGD-100
11.9	Workshops	73→80	---	---	---				SVC-100



$, MILLIONS	FY 74	FY 75	FY 76	FY 77	FY 78	FY 79	FY 80
MINIMUM	1.65	1.40	1.20	1.20	1.40	1.40	1.20
REASONABLE	3.3	2.9	2.40	2.40	2.90	2.90	2.40
CRASH	4.8	4.6	4.00	4.00	4.3	4.6	3.60

Figure XI-10. Roadmap #11: Computer System Performance

XIII. ROADMAP #12: ASSOCIATIVE/PARALLEL PROCESSING

A. BACKGROUND AND OBJECTIVES

Several of the C&C information processing requirements above have been expressed in millions of instructions per second. For the central C&C functions as sized in Volume II, these requirements are seen to be less than 50 MIPS; however, for some related C&C functions, particularly in the real-time sensor and image processing area, requirements can be extremely high--as much as 10^9 to 10^{15} MIPS for real-time processing of radar strip maps.

Extrapolations of current electromagnetic circuit technology indicate that advances in speed will taper off as physical limits of circuits are approached. Unless new technologies (electro-optical, bionic, atomic lattice, etc.) become feasible, the only remaining way to achieve large numbers of MIPS is by exploiting parallelism.

However, a note of caution is in order. There can be a huge difference between the gross MIPS and the net MIPS which are being usefully applied to solving a problem.* On a problem which is inherently serial, as are some aspects of central C&C such as penetration route calculations, 63 of the 64 processors on the ILLIAC IV would be generally idle, making the net MIPS of the ILLIAC IV on this problem little different from the net MIPS of a uniprocessor on the problem.

Fortunately, parallel processing approaches are effective for many sensor data-processing problems. However, a good deal of study is needed to find out the most appropriate matches between applications, computer architecture, and algorithms; otherwise, there is no guarantee that unexpected sensor input data patterns would not cancel out all the expected gains of parallelism just when they are most needed. Currently, the most appropriate Air Force study vehicle is the Goodyear STARAN IV associative processor, developed with RADC support and

---

*As an extreme case, the ARPA Network (consisting of the ILLIAC IV, a CDC 7600, five large IBM 360s, 12 PDP-10s, and several other large computers linked together with wideband communications lines and 26 minicomputers) could be viewed as a 180-MIPS multiprocessor. But it would be folly to try to coordinate it to work 100 percent on a single problem at one time.

currently undergoing successful tests in an FAA air-traffic control experiment.[19] In addition to the STARAN IV, other associative processors such as that developed by Texas Instruments are potential research vehicles. Further studies of alternative parallel architectures should be made to ensure that the associative processor studies are placed in proper perspective.

As one example of the benefits of such studies, RADC estimates that the STARAN IV could be used to increase the track-handling capacity of an AWACS from 200 to 1600, making the AWACS far more effective in areas with high density traffic (e.g., Europe).

B. TASK DESCRIPTIONS

This Roadmap consists of two tasks which are described below.

1. Associative/Parallel Processor Studies (12.1)

These are the experimental development studies, which have cost from $250,000 to $600,000 per year from 1963 to 1970. The 1971 program was on the order of $300,000. In the next several years, the budget has been estimated at $1 to $4 million per year. The key element is the potential for use in the AWACS system and AABNCP and for ELINT data processing. It is assumed that studies will be started for command and control application (both ground and airborne) based on the CCIP-85 reports and that other uses will be investigated. Applications for increasing the capabilities of the WWMCCS and TIPI (as well as other C&C applications) are expected to be examined. This item covers development of special components and high-transfer-rate mass memories for associative processor use. By 1975, it is predicted that these studies could result in firm projects for an Airborne Associative Array Processor (ABAAP) for the AWACS system.

2. Sequential Development Studies (12.2)

This task consists of a set of integrated efforts in advanced and engineering development designed to exploit the advantages of Advanced Associative Processors (AAP) and parallel processors for specific operational applications.

a. AABNCP AAP Unit -- The ABAAP units for the ABNCP should be flight-tested for both C&C and weather data-processing applications onboard the aircraft.

b. Image Processing and Sensor Data Processing -- In the sensor area, a field test of an AAP for ELINT processing and some other intelligence application areas can be expected. One key study will be that of the use of the AAP in photographic imagery processing.

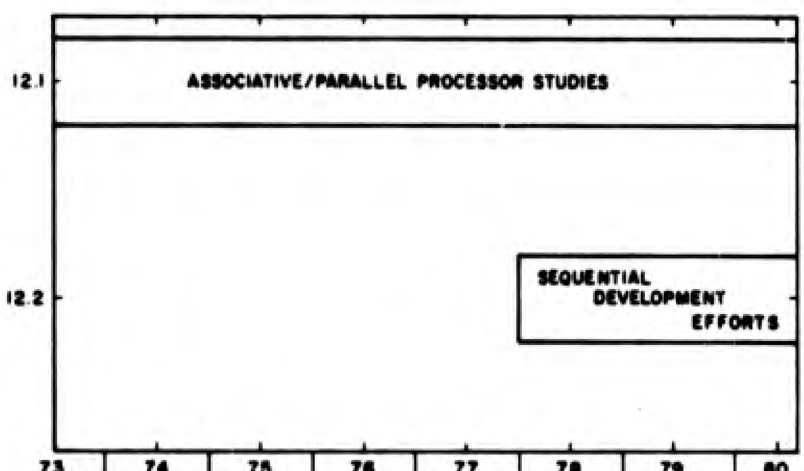
c. Pattern Recognition Application of AAP -- A new application in pattern recognition R&D will be the use of the AAP for analysis of speech to provide data and control input to computer systems and to translate text material between languages.

d. AAP Satellite Preprocessors -- In 1978, satellite preprocessor application studies and development are expected to start, with field tests occuring by 1985.

e. AAP Space Application Research -- In 1978, space application studies and development are expected to be underway.

C. FACTS AND FIGURES

Table XI-X gives the facts and figures for Roadmap #12; Figure XI-11 is a graphic display of the Roadmap.



$, MILLIONS	FY74	FY75	FY76	FY77	FY78	FY79	FY80
MINIMUM	1.0	1.0	1.0	1.0	2.0	2.0	2.0
REASONABLE	7.0	7.0	7.0	7.0	13.0	13.0	13.0
CRASH	13.0	13.0	13.0	13.0	22.0	22.0	22.0

Figure XI-11. Roadmap #12: Associative/ Parallel Processing

TABLE XI-X

ROADMAP #12: FACTS AND FIGURES

No.	Title	Fiscal Years Active	Peak Annual Rate, $\$10^6$ (No. of Efforts)			Existing Projects			Types
			Min.	Reas.	Crash	ID	Years	$\$10^6$	
12.1	Associative/Parallel Processing Studies	73→80	1.0 (1)	7.0 (5)	13.0 (8)	AAP	72	0.3	EXD-80, IGA-10, RQA-10
12.2	Sequential Development Studies	78→80	1.0 (1)	6.0 (5)	9.0 (10)				AVD-40, EGD-30, EXD-30

XIV. ROADMAP #13: SOFTWARE TRANSFERABILITY

A. BACKGROUND AND OBJECTIVES

1. Software Transferability and Maintainability

Software transferability is defined as that property which allows a user's computer program to be operable or executable on all computer systems. Maintainability is that property which allows a user's computer program to be easily and readily changed or modified as would be required by either a change in the desired function of the program (because of a change in the host computer facility hardware) or its supporting utility software.

a. General Status -- Air Force R&D study efforts on software transferability have been limited in the past, primarily because initial study results have indicated that the magnitude of the R&D investment required to guarantee transferability of software would be very large. Furthermore, it seems to have been generally assumed that commercial interests would shortly provide significantly improved capabilities. Despite an expressed interest by USAF in standardization of computer languages, significant commercial improvements have not been forthcoming. The USAF programming problems associated with lack of transferability continue to grow in both cost and complexity.

This Roadmap has been prepared because requirements for transferability can now be clearly defined; further, it is assumed that a major R&D program for transferability will be initiated. The major costs of the project will be covered by utilizing the money saved by improving procurement specifications for new computer systems. Compliance with transferability standards under this project would be mandatory under those specifications. This is to be compared to the purely voluntary aspects of past and most current standardization efforts.

Many of these past efforts have really represented the definition of what could be standard, rather than the required implementation of these standards. As an example, the existence of a standard FORTRAN language specification has not resulted in significant practical implementation of a standard language FORTRAN compiler.

An additional problem exists with COBOL. Because of the large number of options that are available under the definition, the probability is extremely low that a program programmed for one machine will find a similar set of options implemented on another machine.

Previous USAF R&D has consisted of studies of JOVIAL J3 standardization specifications by RADC (assisted by a group of consultants), some JOVIAL and COBOL standardization testing at RADC, an examination of the problem of standardization by MITRE Corporation, a transferability study of costs of transferring some military programs, and a study of operating system characteristics under ESD sponsorship. The only continuing efforts from these past studies are the continuation of the RADC JOVIAL J3 study and the COBOL tests.

Standardization studies by ANSA have resulted in mutually accepted FORTRAN and COBOL standards. JOVIAL and PL-1 standards have been considered, but no schedule is known. Studies of the feasibility of standardization of Job Control Language (JCL), including a time-sharing command language, have been made.

The Navy has done extensive work on the standardization of a subset of COBOL, including the development of verifiers and preprocessors.

The National Security Agency has partially standardized, within its own environment, an ALGOL compiler which is being used on four different systems.

Contributions toward the achievement of transferability and maintainability are resulting from the work being performed on AED, metacompilers, FGSS, and structured programming. These efforts are primarily intended to increase programming productivity, reduce costs, reduce time delays, and reduce errors in system programs. Transferability is considered only secondarily.

A more direct contribution to transferability is the development of data management systems, including means for the transfer of data between systems and means for describing data characteristics in an explicit manner.

The foregoing discussion has addressed primarily the transferability and maintainability of new programs (job programs, application programs). One completely different problem area is that of standardizing operating systems, compilers, data management systems, utility systems, library systems, and library routines. In general, these have been supplied by the machine manufacturers (although unbundling may have some effect).

Further, these system support programs are generally written in an assembly language for the given machine and are, therefore, inherently nontransferable. To the manufacturer, this is usually an asset, in that this incompatibility prohibits the use of programs by a competitor.

With respect to this practice, the Air Force should require that all system support software be written in a higher-order language; furthermore, it should obtain the right to use the resulting software on machines made by other manufacturers. This would create an environment which would lead to a unified standard software system. A new magnitude of problems for maintainability will be posed by the introduction of validation and military security aspects into system support software such as operating systems and data management systems. This is expected to result in increasing emphasis on modular hierarchical programming (AED, FGSS, structural programming) to reduce the problem of revalidating as maintenance changes are made.

Little effort has been directed toward standardizing computer hardware characteristics (e.g., computer word length, number system, arithmetic roundoff operation) or peripherals such as displays (character set, line length, number of lines per display). A recent military procurement of core and computer peripherals from independent suppliers can lead to a significant contribution to the standardization of core, peripherals, and electronic interfaces.

b. Current Funded Efforts -- Currently, programmed projects on standardization at RADC include JOVIAL validation ($450,000), which is to be complete by January 1974. Research and development on the Semanol (Semantic-Oriented Language) is under contract at a cost of $90,000 for FY 1972 with construction of an experimental language processor for JOVIAL in 18 months. (A follow-on program of an additional $50,000 is planned.) At SAMSO, the FGSS system is currently funded at $1,750,000.

c. Future Efforts (Funded or Proposed) -- Under the 64XX advanced development plan (see Table XI-XI), with a 1974 start, ESD has proposed computer-aided instruction (CAI) for programmer training, documentation standards, documentation aids, and software library system. A contributory ESD program under this project is Air Force use of AED.

TABLE XI-XI

FUTURE FUNDING

Item	Yearly Funding, $Millions				Total Funding, $Millions
	1974	1975	1976	1977	
CAI	95	150	50	50	345
Documentation Standards	100	50	---	---	150
Documentation Aids	---	70	225	75	370
Software Library	95	250	400	300	1045
AED	70	290	220	---	580

B. TASK DESCRIPTIONS

1. Software Transferability R&D and Cost/Benefit Analysis (13.1)

This item represents a continuing effort to determine fundamental techniques by which transferability and maintainability can be achieved and to perform cost/benefit analyses of the value of transferability to the Air Force and the costs associated with obtaining the desirable improvements.

2. Language Standardization R&D (13.2)

This task involves the development of technology for defining languages, writing compilers, and other language system support software such as development of metacompilers, procedures for validating compilers in respect to completeness and accuracy of translation, fabrication of compiler preprocessors to detect the illegal or prohibited use of language features or sequences that hinder transferability, and the development of language-to-language translation techniques.

3. Specific Language Standardization Development (13.3)

Task 13.3 emphasizes the preparation of detailed specifications for procurement of standardized languages and includes the actual R&D procurement, test, and operational use of a number of special languages on a number of different computers.

A text-editor language differs slightly from the usual well known language. This language can be used for the editing of photointerpreter reports, situation reports, intelligence reports, engineering reports, project planning operation, etc. By 1975, this text editor should have a classified capability on a remote on-line interactive CRT type of terminal. A graphics capability is also required for diagrams, briefing materials, etc. The JOVIAL J3 standardization effort and the text-editor effort should be supported because of their direct contribution to command and control.

4. Operating System Service Standardization (13.4)

This task covers the following aspects of operating system standardization:

- The determination and implementation of standard services;
- The standardization and implementation of a language for calling operating system services;
- The use of higher-order languages for writing operating systems and the procurement and implementation on DOD computers;
- The standardization of JCLs and their procurements;
- The standardization of time-sharing terminal commands and procedures and their procurement; and
- The specification and procurement of a unified standard executive.

As is well known, the standardization of languages is only a part of the solution to the problem of obtaining transferability. Another part is the standardization of the operating system services provided to the user program and of the language for invoking these services. This applies to the JCL and time-sharing command languages as well.

Over the period, the continual effort to standardize the operating system should finally result in the ability to obtain a unified operating system written in a higher-order language for use on DOD computers. It may be implemented by microprogramming or supported by microprogrammed routines.

5. Documentation Standardization (13.5)

Documentation standardization is primarily a problem of providing adequate direction and emphasis on the standards, and adherence to the standards, as well as providing supporting facilities that reduce the effort required to produce standardized documentation and to use and modify the documentation. The item includes the development of a suitable text-editor system using CRT types of remote terminals on-line with the disc file storage. Such editors currently exist in primitive form but require emphasis on CRT use and on improvement of language (including display, format selection, etc.). A graphic capability is required (for example, a normal C&C program may involve a thousand flow charts).

6. Programming System Standardization (13.6)

The programming system must be standardized if the resulting operational programs are to have a high degree of standardization and maintainability. The library services and routines must be standardized between systems so as to provide a consistent and dependable use of a routine or service without major worries about its reliability, accuracy, etc. Such a library is implied in the AED and FGSS systems. Emphasis here is on the inter-system transferability and compatibility. Why have ten different sine routines, one for each of ten systems, especially when they produce ten different values?

Modular programming is a major area of standardization activity, both because of its contribution to standardization and maintainability, and because of its own need for transferability. The current major programs are AED, FGSS, and structured programming. These efforts are being supported primarily because of their ability to increase productivity of programmers and their contribution to reducing time delays and errors. They also contribute to transferability. The transferability acquired by the use of AED and FGSS systems will enhance software production technology.

Maintainability is expected to be improved significantly by such modular programming methods, particularly when verification and validation are required (as in future C&C systems).

The text-editor system serves a major need in all programming systems. The primary intent of these editors is to reduce the programming costs; however, they can have a major effect on transferability and maintainability as well, because of their ease in making uniform and systematic modifications to programs and operating systems. By 1978, the use of a text editor for producing documentation and maintaining programs should be required. By 1980, a graphic capability should be required for creating and maintaining flow charts (assuming they are still used).

Standardization of a debugging system is a necessary ingredient of overall transferability and maintainability. There is a widespread difference in the capability of debugging systems today. The most widely known one is probably the DDT system which is used on-line with the PDP-10. Such debugging tools should be considered as much a part of a standardized language as are the arithmetic forms of the language. A major fraction (e.g., 40 percent) of a system programming effort is typically used in debugging (or testing) systems.

7. Data Management Systems Standardization (13.7)

Data management systems (DMS) represent an operating system type of service. Where a DMS has been used in a program, significant transferability can be achieved only if the same DMS is also used on the other computer. Thus, one objective of a transferability program should be to see that the DMS system or systems are standardized across computers and are fully implemented.

Data management systems are also required if significant transfer of complex data between computers is to be achieved. This is one of the major functions expected from current DMS R&D programs.

One improvement from DMS is in maintenance of application programs which interact with a large data pase. This is provided by the ability to modify the form of the data base without having to find and make modifications to the many job programs of a large C&C system. This ability provides a high degree of flexibility in the maintenance and a fast maintenance response time, two characteristics of importance in C&C systems. Further, major economies are obtained, which are of significance to all large systems.

In the near future, a multiplicity of DMS systems is expected. It is assumed that, by 1978, a significant reduction in the number of DMS systems can be achieved for C&C purposes. Perhaps this can be reduced to a single comprehensive system.

Query languages and report generation languages for use by C&C personnel should be standardized, although a diversity of individual languages may be desired in order to enhance the coupling of experts in a given field to the computer system. Standardization here means the uniform definition and implementation of each language, not necessarily the reduction in their number.

8. Hardware Standardization R&D (13.8)

The area of hardware standardization is one of the more difficult areas, not because of inherent technical difficulty, but because of the major vested interests by the commercial computer manufacturers in their own hardware standards. Despite these difficulties, the Air Force, perhaps in conjunction with other DOD or U.S. Government or international users, can standardize the computer and peripherals it procures.

The intention should be to specify and require a standardization by some future date such that adequate time is available for vendors to meet these requirements. As an example, all vendors can currently produce machines that use two's-complement arithmetic and 32-bit word lengths. Assume that, in 1975, it is specified that only machines with these particular capabilities will be procured by 1980; the vendors could not claim that the action (when finally implemented in 1980) was arbitrary, premature, capricious, etc., as they appear to do currently when such changes are desired. Further, by providing a clear indication of USAF needs in this regard, vendors can demonstrate their willingness to meet and anticipate USAF requirements. The WWMCCS procurement problems are examples of what can happen if such standards are not set.

The elements considered for such standardization include mainframe and core, storage (disc and tapes), display -- text type, display -- graphic type, printer, and emulator for R&D testing and development.

a. Mainframe Standardization -- Between 1971 and 1974, studies and emulator system development and use should be carried out. By the beginning of 1975, some production specifications and a schedule for standard elements should be available. A typical schedule could be:

1975 - Dynamic microprogramming capability required; input-output block length specified and required

1978 - Paging operation and hierarchical memory operation required; asynchronous mainframe - "core" storage required

1980 - Two-complement machines, 32-bit series word length, and eight-bit characters required; standardized roundoff, etc., specified for math operations; standardized overflow, etc. operataions; and standardized interrupt operations

1982 - Variable-length arithmetic operation and significant bit arithmetic specified and required; indirect addressing to be specified under program controlled option required

1985 - Uniform microprogramming instructions and higher-order language direct execution specified and required; standard subroutines for operating system, DMS, etc. required; security operations specified and required

1988 - Uniform electronic interface required

Besides the elements indicated in the representative schedule, enhanced power instructions should be considered for standardized inclusion in the hardware to facilitate programming and, thus, improve security control, program verification, faster response to software changes, and less costly programming. The previous direction of reducing instruction set complexity to reduce hardware costs (because of infrequent use of the more powerful instructions) should be reversed to better accomplish software objectives. Of particular sensitivity in this regard is the indirect addressing capability which is not included in some of the major computer lines. Sample operations of this nature that should be considered are the following:

- Bit extractor and testing and setting (with indexing, etc.)
- Field extractor-depositor (with indexing, etc.) within word
- Full extractor and depositor across word boundaries (full regular length)
- Table lookup instructions
- Hash code computations
- Mathematical operation -- sine, cosine, sigma, log, etc.
- Bit count
- List processing operations

b. Storage System Standardization (Disc type or equivalent -- The standardization of secondary storage systems will obviate the problems of transferability between systems caused by the secondary storage system. By 1973, the specification for transfer block length and addressing could be available and used for procurement specifications. The hierarchical storage system should be required. Because of the large number of independent suppliers of storage systems, a definition of uniform electronic interfaces should be specified early -- say, by 1977.

c. Displays (CRT, text, on-line) -- One expected standardization problem, which has not yet become evident in field operations, is that of standardized programming for test types of displays. Since the use of CRT consoles is only now becoming widespread, and the number of job programs that are using CRT text consoles is limited, the effect of CRT text consoles on job program transferability is not yet evident. Considerable reprogramming is required to accommodate differences such as a shorter line length on a display than the previous line length, differences in the character set and character codes, differences in the number of lines per message, and the effect of slow transmission speed on a display system originally based on high speed transfer. The total effect might be such as to render the transfer of the original program unreasonable.

The current time is a good one to start standardization of line length, message line length, character set, character code, and message transfer rate. By the beginning of 1973, if studies are started immediately, a requirement for an initial standardized CRT text display could be implemented. Recommended values are 80 characters per line, a 128-character set, ASC II coding, 32 lines (with one, two, four, eight, and 16 subsets) and a 4800-bit-per-second transfer rate (600 characters per second, full display in approximately four seconds for a full 2240 characters and less time for fewer characters -- e.g., one line in approximately 0.10 second).

There are many more elements which need to be standardized (e.g., key layouts, key functions, interrupt to computer standard, etc.). A second major area is the interfacing with communication line and with non-communications (i.e., direct to computer system). By 1982, communication lines interfaces should be required to be standard (including error detector and correction in both directions) and, by 1985, noncommunication electronics interface (including error detector and correction) should be standardized. By 1988, a fully standardized unified unit for all computer systems should be required.

d. Displays (CRT graphics -- on-line) -- At the current time, the field of graphics displays shows wide dispersion and essentially no standardization at all. Some computer manufacturers even produce several versions of graphic systems that are not compatible even within the company products line.

By 1975, if studies are started immediately, specifications for data representation and control representation could be made and required on future procurement. Three-dimensional rotation and translation characteristics should be required. By 1978, a uniform electronic interface could be required for both communications and noncommunications systems; a unified unit should be required ten years later.

e. Printer -- The problem of noncompatible printer character sets and line lengths is the cause for many of the transferability problems of today. By 1973, a character set and code should be specified and required; by 1976, line length should be specified and required; and, by 1978, a common electronic interface should be specified and required. A unified unit could be available for competitive procurement by 1982.

9. Codes and Formats (13.9)

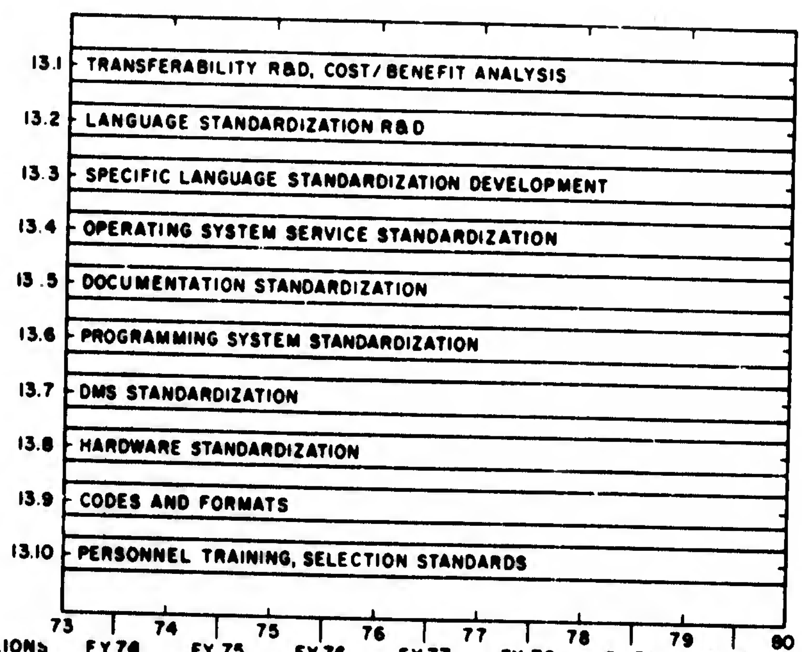
The standardization of codes, formats, etc. is and will continue to be a requirement. The problem of deciding on the ASC II code is a good example. The standardization of data formats has been a continuing problem in interservice use of data. Specific problems should be attacked.

10. <u>Personnel Training and Selection Standards</u> (13.10)

It should not be forgotten that the quality of the software is always related to the quality and training of the personnel doing the work. Computer-aided instruction techniques should be actively developed for programmer training, maintenance, and system operations. The development of means for measuring the quality and productivity of programmers should receive R&D funding. Means for selecting programmers should also be developed. Such uniform selection and training should result in increased transferability and maintainability of software.

C. FACTS AND FIGURES

Table XI-XII presents pertinent facts and figures for Roadmap #13: Software Transferability. Figure XI-12 is a graphic representation of the Roadmap.



$,MILLIONS	FY 74	FY 75	FY 76	FY 77	FY 78	FY 79	FY 80
MINIMUM	2.0	2.0	2.0	2.0	2.0	2.0	2.0
REASONABLE	8.3	8.3	8.3	8.3	8.3	8.3	8.3
CRASH	2.40	2.40	2.40	2.40	2.40	2.40	2.40

Figure XI-12. Roadmap #13: Software Transferability

TABLE XI-XII

ROADMAP #13: FACTS AND FIGURES

No.	Title	Fiscal Years Active	Peak Annual Rate, $\$10^6$ (No. of Efforts)			Existing Projects			Types
			Min.	Reas.	Crash	ID	Years	$\$10^6$	
13.1	Transferability R&D, Cost/Benefit Analysis	73→90	1.0 (1)	0.3 (1)	1.0 (3)				RES-100
13.2	Language Standardization R&D	73→90	0.1 (1)	0.3 (1)	1.0 (3)				EXD-100
13.3	Specific Language Standardization Devel.	73→90	0.2 (1)	2.5 (5)	5.0 (10)	5581	71-72	0.3	EXD-50, EGD-50
13.4	Operating System Service Standardization	73→90	0.15 (1)	1.0 (3)	5.0 (10)				EXD-30, AVD-30, EGD-40
13.5	Documentation Standardization	73→90	0.05 (1)	0.2 (2)	0.5 (5)				EXD-40, EGD-60
13.6	Programming System Standardization	73→90	0.1 (1)	1.0 (3)	5.0 (3)	FGSS AED	71-74 74-76	0.75 0.2	EXD-10, AVD-20, EGD-40, OPD-30
13.7	DMS Standardization	73→90	0.05 (1)	0.3 (1)	0.5 (3)				AVD-50, EGD-50
13.8	Hardware Standardization	73→90	0.25 (1)	2.5 (6)	5.0 (6)				EXD-20, AVD-80
13.9	Codes and Formats	73→90	0.05 (1)	0.1 (2)	0.5 (4)				EGD-100
13.10	Personnel Training & Selection Standards	73→90	0.05 (1)	0.2 (2)	0.5 (4)				EGD-100

XV. ROADMAP #14: COMPUTER-AIDED INSTRUCTION

A. BACKGROUND AND OBJECTIVES

Even if all the techniques recommended in the preceding Roadmap are developed, they will be of relatively little value without well-qualified people to apply them, or to see that they are applied properly. One institutional roadblock is the current lack of familiarity within the Air Force with the unique aspects of computing technology, particularly software. This is evident in the following areas.

1. Establishment of Unrealistic Software Schedules

One example of this situation was provided in a briefing to the Study Group. It involved the use of remotely piloted vehicles. The application had major new C&C software implications and was planned to be operational in 1974, even though almost no thought had been given to the software. With luck, the software would have been only a year late.

2. Gold-Plating

The less familiar a contract monitor is with computing and software, the more tempatation there is for contractors to add marginal or unnecessary features to a system, sometimes if only to protect themselves against "soft," or ill-defined, requirements.

3. Major Design Retrofits

The 95-percent reprogramming on SACCS and 67-percent reprogramming on Seek Data II imply needed improvements not only in C&C information system requirements analysis and design techniques, but also in the requirements analysis and design personnel.

The lack of trained personnel is one institutional roadblock which technology may be able to help remove. The technology of computer-aided instruction (CAI) currently shows promise, but has at least two major drawbacks:

- A great deal of marginally competent work goes on;[20] and
- There is usually a large communications gap between the computer specialist and the expert in the field of instruction.

If the field of instruction is computing, the second difficulty is largely removed. By selecting a thoughtful, thorough, top-level CAI development group, the Air Force could provide via terminals at each airbase a system which, through appropriate use of problems, games, and simulations, could make learning about computing an effective and enjoyable process.

B. TASK DESCRIPTION

Roadmap #14 consists of one task which will be developed to support the goals outlined in the narrative above.

C. FACTS AND FIGURES

Table XI-XIII presents facts and figures for Roadmap #14: Computer-Aided Instruction; Figure XI-13 is a graphic representation of the Roadmap.

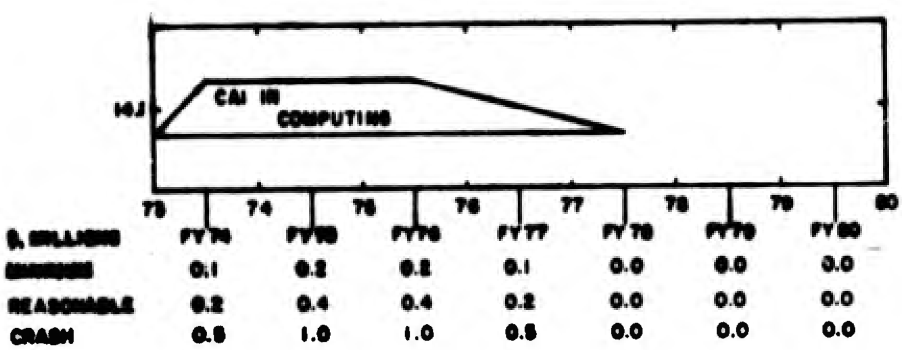


Figure XI-13. Roadmap #14: Computer-Aided Instruction

TABLE XI-XIII

ROADMAP #14: FACTS AND FIGURES

No.	Title	Fiscal Years Active	Peak Annual Rate, $\$10^6$ (No. of Efforts)			Types
			Min.	Reas.	Crash	
14.1	CAI in Computing	73 74 76 77	0.2 (1)	0.4 (1)	1.0 (2)	EGD-100

XVI. ROADMAP #15: DESTRUCTIBILITY

A. BACKGROUND AND OBJECTIVES

Emergency destruction or denial of classified material is defined as the abnormal means and procedures used to prevent the unauthorized seizure or disclosure of classified material. It is characterized by the use of destruction or denial mechanisms (hereinafter called devices) which detonate and accomplish massive permanent destruction of classified material in a matter of minutes. The devices are used only when physical seizure or disclosure is imminent or the material cannot be removed or withdrawn to a protected environment for any operational reason. An arbitrary maximum limit of 120 minutes was set to define the boundaries of emergency which justify the use of emergency devices. A user who certifies that he will have at least 120 minutes to destroy his classified material holdings in an emergency would tend to employ conventional (normal) means. On the other hand, one who had only one minute or less to destroy his classified material holdings would tend to employ more exotic means.

Air Force Manual 205-1 (Chapter 5, paragraphs 5-5 and 5-11) discusses procedures for normal destruction and lists some devices for accomplishing it. Paragraph 5-6 discusses emergency destruction but does not define it. The methods it suggests are time-consuming and primitive. Roughly translated, the paragraph authorizes the responsible authority "to do whatever is possible." The drawback is that in many cases nothing is possible because the responsible authority simply did not possess the devices to accomplish the task in the time available. The Palestinian belligerents who hijacked the Pan American 747 and TWA 707 may have pointed the way to emergency destruction techniques by blowing up the 747 shortly after landing. Although the classified material carried onboard has been considered compromised, there is some doubt whether they had sufficient time to seize the material from the cargo hold (i.e., less than ten minutes). There are more pressing "combat" examples such as the Viet Cong attack on Tan Son Nhut Air Base during the Tet offensive which seem to amplify the need for emergency devices.

Preliminary investigation revealed that the only technical need for emergency destruction devices has been surfaced by ESD for AWACS (TN-ESD-11-71, -02, -03, -04). There is, however, reason to believe that requirements for emergency devices have been buried in specifications for the procurement of classified equipment. Two contractors have produced emergency devices as part of the overall system program package. None of the evidence indicates the construction of denial devices. The concept of denial is therefore introduced as an alternate means of preventing unauthorized seizure or disclosure without irradiation by an extremely radioactive isotope (gas, powder, or liquid). This concept may prove more effective and less costly than an exotic explosive mechanism.

The objectives of normal and emergency destruction are identical--material must be destroyed to such an extent that the data recorded thereon or stored therein are rendered unrecognizable or illegible (presumably by any advanced reconstitution technology). No guidelines are given for quantifying what is, or is not, recognizable or legible; however, some are presented here for validation and verification.

- Equipment should be destroyed to such an extent that manufacturer, nomenclature, and function cannot be determined.

- Circuits should be destroyed to such an extent that quantity and logical pattern of electrical flow cannot be determined.

- Electromagnetic storage media should be destroyed to such an extent that no more than 40 contiguous bits per million bits (one standard word) of original information can be determined.

- Bulk storage media should be destroyed to such an extent that no more than:

  - One printed word per 1000 pages of print can be determined;

  - One punched or interpreted word per 10,000 punched cards can be determined; or

  - One square millimeter of original image per ten images can be determined.

B. TASK DESCRIPTIONS

This Roadmap consists of four tasks which are described below.

1. Define Candidates for Emergency Destruction/Denial (15.1)

The objective of this task is to study and document USAF items of permanent, semi-permanent, and nonaccountable inventory which may be classified or be capable of storing classified information. A preliminary outline for definition and categorization is:

- Equipment
  - Ground
    - Electronic
    - Electromechanical
    - Mechanical
    - Facility/Vehicle

- Airborne
  - Electronic
  - Electromechanical
  - Mechanical
  - Vehicle

- Circuits
  - Internal
  - External (exposed wire)

- Electromagnetic Storage Media
  - Magnetic
    - Cassette
    - Cartridge
    - Dizum
    - Data Cell
    - Disc
    - Tape
  - Electrical
    - Relays
    - Core
    - Buzzers
    - Registers
  - Plasma
  - Flat Plate/Rod

- Bulk Storage Media
  - Punched Cards
  - Punched Paper/Mylar Tape
  - Microform (microfiche, roll, etc.)
  - Paper

It may be necessary to identify media used to house the media listed above, since storage media may have to be destroyed in order to assure destruction of their contents. For example, it may be necessary to destroy a safe or a vault to guarantee destruction of contents when, in actuality, only the contents were required to be destroyed.

Each item should be catalogued according to its taxonomic relationship in a hierarchy and the amount of destruction necessary to "certify" it destroyed in accordance with security regulations. To the maximum extent possible, the cataloguing should incorporate the symbols, data elements, and codes of the automated logistics system. The items should appear as a line item in a USAF technical manual of emergency destruction/denial devices and procedures.

2. Develop a Catalog of Emergency Destruction/Denial Devices (15. 2)

The objective of this task is to accomplish the research and engineering necessary to develop a range of general-purpose emergency devices which are sensitive to use, cost, and time.

Preliminary investigation indicates that no single device including large-yield thermonuclear weapons is a panacea for emergency destruction/denial. There seems to be a fairly wide range of timing requirements (i.e., time available to command, detonate, and destroy) and an equally wide range of applications for such devices. For example, destruction of provocative recce/intelligence platforms may have to occur in seconds while destruction of the contents of a safe located on a CONUS air base may have to occur in less than 120 minutes. The implication is clear that the shorter the "time to destroy," the more exotic and dangerous the device used will have to be.

Some devices have already been developed by two companies for equipment destruction, and there may be others. A study survey may be required to determine the current state of the art. Nevertheless, it is believed that technical and operational feasibility studies, prototyping of the promising devices, and testing for possible acquisition should occur in the following areas:

- Mechanical
  - Pulverizers (separation)
  - Crushers (compaction)
  - Shock (shatter)
  - Hypervibration (shatter)
- Thermal
  - Fire (combustion)
  - Heat (liquefy)
  - Energy Beam (vaporize)
- Flooding (water or chemical)
- Conventional Explosives
- Biological Agents
  - Corrosive
  - Pathological
- Chemical Agents
  - Corrosive
  - Pathological

- Cryogenic Agents
  - Change State
  - Change Property
- Thermonuclear Devices
  - Low-Yield
  - High-Yield
- Radioactive Isotopes
  - Corrosive
  - Pathological
- Electromagnetic Force Fields
  - Super Magnets
  - Mini Magnets

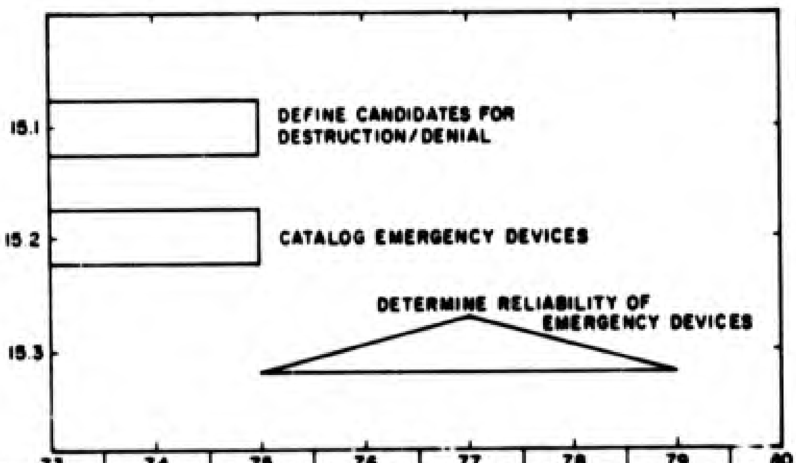
The areas specified are meant to serve as guidelines. They should not be interpreted as the only areas for investigation. Nor should they be interpreted to mean a device should be developed in each area. Moreover, development of the devices must include command detonating techniques (i.e., salvo, ripple, individual) and mechanisms.

3. Develop and Determine Reliability of Emergency Devices (15.3)

The objective of this task is to establish human reliability criteria for each device developed. It can be postulated that detonating an emergency device will cause swift, effective destruction; but, if it is used improperly or detonated accidently, it will cause severe or fatal injury and unnecessary destruction of material. One corporation indicated that it has fabricated several devices which would "do the job"; but these devices were considered too dangerous for the safety and welfare of personnel during normal operations. Clearly, there is a concomitant requirement to define the human reliability factors associated with the use of each device. It must be expected that severe timing requirements will require exotic, dangerous emergency devices. Psychiatric certification of personnel having access to the devices could be required. Triple "unlock and command" could be required to prevent disaster. Nevertheless, stringent human reliability controls should not prevent the design and fabrication of desperately needed emergency devices. Nuclear safety directives and procedures have proved extremely effective in routine handling of these dangerous weapons. It is entirely possible that development of human system reliable emergency devices will permit the employment and routine use of classified systems heretofore deemed too risky in terms of compromise.

C. FACTS AND FIGURES

Table XI-XIV gives the pertinent facts and figures for Roadmap #15: Destructibility. Information is also presented pictorially in Figure XI-14.



$, MILLIONS	FY 74	FY 75	FY 76	FY 77	FY 78	FY 79	FY 80
MINIMUM	0.3	1.3	1.0	1.0	1.0	1.0	1.0
REASONABLE	0.6	2.6	2.0	2.0	2.0	2.0	2.0
CRASH	0.9	1.9	5.0	5.0	5.0	5.0	5.0

Figure XI-14. Roadmap #15: Destructibility

TABLE XI-XIV

ROADMAP #15: FACTS AND FIGURES

No.	Title	Fiscal Years Active	Peak Annual Rate, $\$10^6$ (No. of Efforts)			Existing Projects			Types
			Min.	Reas.	Crash	ID	Years	$\$10^6$	
15.1	Define Candidates for Emergency Destruction/Denial	74→75	0.2	0.4	0.6				RQA-100
15.2	Develop a Catalog of Emergency Devices	74→75	0.1	0.2	0.3				LIB-100
15.3	Develop Emergency Devices & Reliability Factors	75↗ 77→79	1.0	2.0	5.0				EXD/AVD-50 ENG/OPD-50

XVII. ROADMAP #16: WWMCCS CONVERSION

A. BACKGROUND AND OBJECTIVES

1. Introduction

A current interoperability effort is the integrated WWMCCS computer procurement for which the vendor was recently selected (October 1971), culminating a process consuming over four years of elapsed time and procurement process expenses roughly equal to the $50 million hardware procurement. Even now, it will take another three years for SAC, for example, to phase its new computers into operation. During this time, 200 programmers will be working full-time on conversion. This is half of the six-year stated operational lifetime of the WWMCCS machines. Thus, SAC will begin in 1975 to operate computers whose procurement was initiated in 1967, whose basic hardware and operating system architecture are vintage 1965, and which are planned for obsolescence by 1978.

During the WWMCCS procurement, however, a large body of experience has been accumulated which, if properly collected and analyzed, will help the Air Force to ensure that the next generation (WWMCCS II) computer procurement process in the late 1970s does not yield similar unfortunate results for the 1980s. The Air Force should begin such a "lessons learned" analysis now, followed by advanced studies of Air Force aspects of WWMCCS II functional requirements, hardware/software requirements, intraoperability and interoperability standards, testing, and transition management planning. This should be accompanied by a continuous effort to build and maintain a conversion data base. Such a data base would assure that converting the Air Force portion of the estimated $722 million worth of upcoming WWMCCS I software to WWMCCS II machines would be less time consuming and expensive than the current conversion process.

2. Automatic Data Processing (ADP) System Description

Automatic Data Processing systems are used extensively throughout the WWMCCS to assist all command and staff levels in a variety of tasks associated with their assigned missions. In general, the ADP centers can be considered to be manned 24 hours a day, seven days a week.

a. Computer Subsystem--The computer subsystem consists of the computer itself, common peripheral equipment, and vendor-supplied documentation.

b. Real-Time Subsystem--The real-time subsystem consists of AUTODIN terminals and signal converters/conditioners (such as analog-to-digital converters) through which data are introduced directly into the computer subsystem without local manual intervention.

c. Remote Console Subsystem--The remote console subsystem consists of manual input-output devices remotely located from the central processing unit (such as cathode ray tube displays, typewriter keyboards, console printers, logic pushbuttons, plotters, and other devices situated apart from the computer location) through which on-line interactive requests can be made and system responses provided.

d. Software Subsystem--The total software subsystem consists of the files, programs, and documentation provided by the user organizations and the vendor. All user and vendor programs operate in conjunction with an operating system. Thus, any necessary program can be called for by the computer operator, the data user at his console, or through the real-time subsystem.

e. Processing--All processing is accomplished under supervision of the operating system which maintains control of the computer system and allocates available resources. Machine processing can be initiated by the computer operator, staff users through the remote console subsystem, or by a message arriving via the real-time subsystem. Processing jobs initiated by the computer operator may provide their own data processing routine or may require the execution of other nonfunctional programs (called in by the operating system). All processing jobs initiated from remote consoles or by the real-time subsystem require use of some nonfunctional programs, at least for processing the requests. Jobs may be updates on the files, direct queries of the files, computational requests providing their own data, or combinations of these. A particular job may call upon several programs, the output of some being the input to others. Several programs being processed for several jobs may be in main memory concurrently.

3. WWMCCS Objectives

The major objectives of the standardization of ADP systems are outlined below.

a. Increased Personnel Utilization--The goal is to improve personnel utilization by allowing programmers, operators, maintenance personnel, and staff users to more readily use their past experiences and training when moving from job to job, through standardization of equipment, procedures, and nonfunctional system software.

b. Improved Logistic Support--The objective is to reduce the cost and increase the effectiveness of logistic support through standardization of equipment and maintenance procedures.

c. Reduced Overhead Costs--The objective is to reduce the duplication of effort involved in hardware and software acquisition, configuration control, program development, and maintenance through central management of standardized equipment and nonfunctional system software.

d. Improved Data Interchange--The objective is to improve the speed and ease of data interchange between centers through standardization of equipment and system software.

e. Improved System Responsiveness--The objective is to improve the effectiveness of the ADP support by taking advantage of modular, high speed, real-time capable systems which provide rapid access to large data files.

f. Increased Expandability and Flexibility--The objective is to provide for future redesign and modular expansion of capability without the necessity of conversion to new ADP equipment, with its attendant delays and expense.

g. Reduced Operating Costs--The objective is to reduce duplication of effort in functional program development and to improve the effective utilization of functional programs available for exchange throughout the system through standardization of equipment, system software, and the associated reference documentation.

4. Requirement for Orderly Conversion Planning

The systems which currently comprise WWMCCS were not designed with conversion to latter generation equipments in mind. The design activity concentrated primarily on the operational problems each was facing at that time. Additionally, no standard methodology to determine new requirements within the context of the WWMCCS objective was utilized. As a result, the planning, scheduling, and determination of WWMCCS I requirements was accomplished on an ad hoc basis. This led to false starts, misinterpretation, and confusion. This WWMCCS Conversion Roadmap outlines an orderly approach to alleviate this problem for the 1980-1990 time period.

a. Air Force WWMCCS Schedule--Currently, only four Air Force or Air Force-supported activities are approved to acquire WWMCCS I ADPE equipments. These are as follows:

Organization	WWMCCS Machine Number
Strategic Air Command	1 and 8
Air Defense Command SCC	4
CONAD IDHS	6
Military Airlift Command	7

The Air Force has planned that the following activities would additionally receive WWMCCS standard ADPE.

AFSC/ESD	Hq USAF
21st Air Force	CINCPACAF
22d Air Force	5th Air Force
Tactical Air Command	13th Air Force
Air Force Communication Service	USAFE

The latter equipments were planned to be acquired in FY 1973 and FY 1974. It is assumed for the purposes of this Roadmap that these additional acquisitions will be approved by DOD; it is further assumed that the WWMCCS equipments will have a useful operational life of ten years.

Conversion planning to transition from WWMCCS I, and the derivation of equipment performance requirements and interoperability standards for WWMCCS II, will be a significant problem to contend with due to the expected advances in hardware architecture and computer configurations.

b. Conversion Planning -- Converting WWMCCS I to WWMCCS II will be a complex and lengthy process requiring several years and the coordination of many activities. Only if the new EDP system is compatible with WWMCCS I will the conversion process be simple. If the objectives of the WWMCCS I system are modified, or if changes are required in its organization, components, and content, then the conversion process will be complex. Extensive planning is necessary to coordinate the conversion activities and to formulate strategies for the transfer of computer programs. To assess the complexity of conversion planning, it is necessary to look at conversion objectives, the WWMCCS I system, and the functional and technical characteristics of WWMCCS II. The conversion objectives may dictate extensive functional changes to the computer programs or only those changes necessary for the transfer to new equipment. If additional needs force changes in computer programs, the transfer is constrained (i.e., additional design and programming must be performed.

In part, the WWMCCS I system will determine the complexity of conversion. For example, its software may be fairly machine-dependent or the physical facility for equipment may already be crowded, which would limit the possibility of running the systems (old and new) in parallel during phaseover. If the organization of the WWMCCS II equipment is radically different or if the executive software is entirely replaced, elaborate plans must be made for testing the new system, verifying its operation, and retraining its operators or users.

Conversion almost always involves design changes to the system to accommodate new requirements. The fewer the requirements for change, the easier the conversion will be. Design changes stem from additional system requirements. For example, the replacement of magnetic tape with disc storage to satisfy a system requirement for improved response time will cause changes to the file processing logic. Design changes should be identified and introduced into the new system design to help plan the conversion. Reliable cost estimates and schedules require a stable design for the WWMCCS II system.

The following discussion on system conversion assumes that one set of ADP equipment completely replaces another (even though this may be accomplished in stages). In other conversions, where new equipment is added without replacing old, the solutions to the problem of continuity in system operation will differ: for example, operations of the total system may be divided among the equipments, or only new applications may be placed on the additional equipment.

c. Events in System Conversion--The five sequential steps in system conversion are to establish the design, plan the conversion, develop the system, test the system, and phase the system into operation.

The design of the WWMCCS I system is a primary input in establishing the design of the new system. Outputs of the design activity include equipment and system software specifications, identification of the software needed that can be transferred, and additional software requirements. These inputs should be based on empirical data acquired by testing the current system in a controlled operational simulation to ensure the validity of the requirements. Cost evaluation and resource assessment should be used as inputs to help plan the conversion; the outputs of the planning activity are schedules, resource allocations, transfer strategy, phaseover strategy, and requirements for testing, training, and new system operation. In the next stage of the conversion, software is developed; this includes the transfer of old programs to the new system as well as preparation of any new programs needed to start operation. Inputs from the current system include data structures and software to be transferred. Additional inputs are facilities, new equipment, systems software, and production personnel.

The next phase of the conversion process is testing. Inputs from WWMCCS I are test procedures and test data. New or modified procedures and data may be needed for testing the new system. An activity for comparing the inputs of the WWMCCS I system with those of the new system is also required. In the final step of the conversion process, operations of WWMCCS I system are phased over into the operation of the new system. At this point, the new system is initiated with data from the current system, and the user has full responsibility.

The transfer problem exists during the development and test activities. In the development stage, the problem concerns the use of data structures and software from the current system: this is the crux of the transfer problem. Data structures and software are transformed and modified for use in the new system according to the form and structure of the new equipment and the objectives of the new system.

Training requirements dictate that personnel be trained before the development phase, and users be trained before the testing phase.

d. Design Activities--Design activities should precede other planning activities because factors that affect subsequent planning are outputs of the design activity. The design activity should determine why the system must be changed and identify factors that force change, so that the computer programs that can be transferred are identified and so that EDP system and additional software requirements can be developed. Requirements for system changes may come from many sources: the need for additional capacity, storage, or throughput; a different input methodology; a new user interface, etc.

An important factor in the transfer of computer programs is the set of conversion objectives (i.e., the reasons for replacing the EDP equipment). These objectives should be based on controlled testing of the current system. A broad objective like cost reduction may be narrowed to reducing maintenance costs for computer programs or to reducing the maintenance or operating costs of old equipment. This orientation gives some perspective to the transfer problem: it indicates

- The individual computer programs or groups that can be transferred;
- What portions of these can be transferred; and
- Which transfer methods are suitable.

The following example illustrates how the conversion objective influences the planning and selection of transfer strategies.

Suppose that the conversion objective is to integrate operations. The user wishes to integrate operations that are currently performed separately in his system. For example, the user may wish to combine, under central control, disjointed processing functions (or subsystems) in order to utilize the equipment more effectively; or, because he now requires a multiaccess system, simultaneous access to multiple functions is needed. Another possibility is that the user wishes to combine the operations of several computers, to share processing loads or utilize processor time more efficiently. Under this circumstance, it is apparent that the executive computer programs that control his current system must be replaced. It is also apparent that there is the need for reworking the parts of the application programs that interface with the old executive--for segmenting them for a multiprocessing environment, revamping their access to data, and modifying their interface with the user.

Design requirements, such as the requirement for computational accuracy, also emanate from the current system specifications. If these are not documented, they must be determined before a new design can be developed. The design of the new ADP system will surely cause problems in the production and system verification efforts if the status of the WWMCCS I system is not well documented.

e. Planning--In planning for conversion, it is necessary to consider how the new system will be developed, how it will be tested, and how phaseover from the current system to the new system will be accomplished. Planning requires, as inputs, cost evaluation and resource assessment. Cost factors will be useful in determining tradeoffs for transfer that influence the efficiency of the new system and its adaptability for the user. Resource assessment will measure personnel availability, software transferability, facility constraints, lead time, etc. Planning must be applied to schedules, resource allocation, transfer strategy, training requirements, test requirements, and phaseover strategy. The considerations that go into planning personnel retraining and allocation, testing, and configuration management are addressed below.

f. Personnel--Operational capability is a combination of equipment, software, and a user. When significant differences exist between current and new systems, the user must also be, in a sense, "transferred." Retraining is an aspect of the problem of allocating the manpower resource for the conversion, including both the personnel who will produce the new system and those who will use and operate it. Sometimes it is possible to divide responsibilities so that some of the user's personnel are working with the object system while others are working with the target system. In other cases, it is necessary to pool the user's resources, making all of the user's personnel familiar with the target system. If the same personnel are responsible for both the maintenance and operation of the current system and the development (or operation) of the new system, then all of them must be retrained prior to system production (or phaseover). If responsibility is divided between separate groups, retraining may be phased over a longer period.

Pooling the responsibilities for the production of the target system with those for the maintenance of the object system is effective only when the requirements for maintenance are light, although it exploits the experience of the personnel, particularly in respect to software that will be transferred. Pooling the responsibilities for the use or operation of the object and target systems may be an effective means for involving the user early in the development of the target system as well as for enhancing the system verification effort. However, if current operations place a heavy load on the user, or if the user interface design undergoes a radical change, divided responsibilities may be more appropriate.

g. Testing--Requirements for verification of the WWMCCS II system should also be established early. The design of tests to verify reliability will affect how the system is developed, when equipment must be installed, how phaseover will take place, when user personnel are trained and so forth. In addition to verifying that the target system performs to specifications, the test design should provide means by which the user's ability to operate the system can be measured and, if the system is adaptable, a measure of the user's ability to add capabilities to the system within a specified time. This testing should be in the context of some simulated operational problem areas.

h. Configuration Management--Configuration control or management, a critical problem in conversion planning, is the coordination of design changes to the current system with the development of the new system. Ideally, design of the new system is frozen once the design has been agreed upon. But the poor state of the art in design methods, coupled with changes in the environment, inevitably lead to design changes. Hence, tight control is necessary to ensure that changes made to the current system will be reflected in the design of the new system, that they are accounted for in the test design for system verification, and that they are reflected in training requirements for user personnel.

i. Phaseover Strategy--Planning for phaseover must be applied to operational continuity, assurance of system reliability, equipment phasing, training, etc. During phaseover, utilization of the old and new equipments may occur in parallel. The equipments are used in parallel for a period of time during which the target system is verified or applications from the current system are gradually phased over to the new system. For various reasons, however, it is not always possible to run the equipments in parallel. The most likely reason is insufficient space in the equipment facility to house two sets of EDP equipment. Inability to run the new and current equipments in parallel may demand a more elaborate phaseover strategy, such as gradual replacement of current equipment components with new equipment components, utilizing throw-away interface packages of computer programs.

B. TASK DESCRIPTIONS

1. WWMCCS Engineering Support (16.1)

This project consists of developing prototype conversion data bases for Air Force WWMCCS members and determining interface requirements and training requirements. Additionally, it includes prototype engineering on new state-of-the-art programming features, programmer aids, and multiuser data base features to be employed in WWMCCS II.

2. WWMCCS Operational Support (16.2)

This activity includes operational engineering support for operational testing, increasing effectiveness of the current system (WWMCCS I), and determining required system modifications.

3. WWMCCS Operational Simulation (16.3)

Operational simulation includes the development and application of controlled operational simulation to the WWMCCS I system to identify deficiences and derive requirements for WWMCCS II analysis of system operations and identification of conversion requirements.

4. WWMCCS Design (16.4)

This project consists of analyzing the functional requirements for WWMCCS II equipments and determining speed, storage size, and architecture of replacement ADP equipment. Effort includes development of engineering specifications identifying the Air Force requirements for WWMCCS II.

5. WWMCCS Standards (16.5)

This task concerns developing necessary methods, operating procedures, and standards to permit interoperation of all components of the WWMCCS system, including interservice interoperability.

6. WWMCCS Conversion Plans (16.6)

This activity includes plans and incorporation of design and standard activities.

7. WWMCCS Procurement Planning (16.7)

The planning task includes development of test and demonstration criteria, bench mark problems and programs, and procurement approaches.

8. WWMCCS Procurement Activities (16.8)

This activity includes contractor support in the preparation of documentation requirements, analysis of testing results, and other consultation.

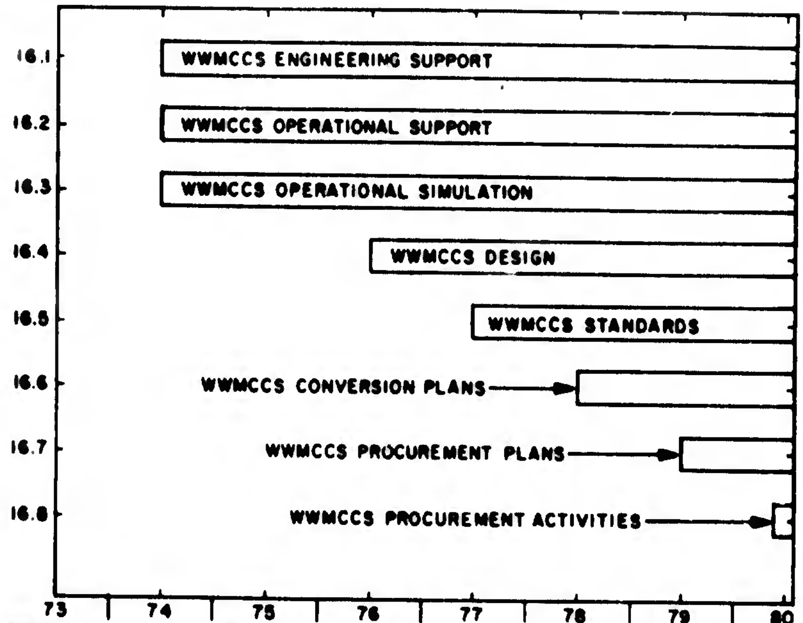
C. FACTS AND FIGURES

Table XI-XV presents facts and figures for Roadmap #16: WWMCCS Conversion. Relevant data are also given in Figure XI-15.

TABLE XI-XV

ROADMAP #16: FACTS AND FIGURES

No.	Title	Fiscal Years Active	Peak Annual Rate, $\$10^6$ (No. of Efforts)			Existing Projects			Types
			Min.	Reas.	Crash	ID	Years	$\$10^6$	
16.1	WWMCCS Eng. Support	74→80	0.5 (2)	1.0 (3)	2.0 (4)	2.5.1 2.5.2	75-77 72-77	3.5 1.0	EGD-50, OPD-50
16.2	WWMCCS Opn. Support	74→80	0.5 (2)	0.7 (4)	1.0 (6)				OPD-100
16.3	WWMCCS Opn. Simulation	74→80	0.2 (1)	0.3 (2)	0.7 (3)	2.4.7 2.5.4	72-74 72-77	0.7 0.64	OPD-100
16.4	WWMCCS Design	76→80	0.2 (1)	0.4 (1)	0.7 (2)	2.5.3	72-77	2.2	OPD-100
16.5	WWMCCS Standards	77→80	0.1 (2)	0.3 (3)	0.6 (4)	2.4.3	72-75	0.15	OPD-100
16.6	WWMCCS Conversion Plans	78→80	0.1 (1)	0.3 (2)	0.5 (3)				OPD-100
16.7	WWMCCS Procurement Plans	79→80	0.07 (1)	0.2 (2)	0.4 (3)				CPD-100
16.8	WWMCCS Procurement Activities	80→84	0.05 (1)	0.1 (1)	0.2 (2)				OPD-100



$, MILLIONS	FY 74	FY 75	FY 76	FY 77	FY 78	FY 79	FY 80
MINIMUM	1.2	1.2	1.4	1.5	1.6	1.67	1.72
REASONABLE	2.0	2.0	2.4	2.7	3.0	3.2	3.3
CRASH	3.7	3.7	4.4	5.0	5.5	5.9	6.1

Figure XI-15. Roadmap #16: WWMCCS Conversion

XVIII. ROADMAP #17: INTERSERVICE COORDINATION

A. BACKGROUND AND OBJECTIVES

This Roadmap identifies and discusses the technical and operational programs for interservice coordination to achieve interoperability of a command, control, and communication system in the 1980-1990 time period. The requirements for Air Force C^3 systems to interoperate in coordinated joint or combined operations is well established and documented. By 1985, this requirement will increasingly impact overall Air Force C^3 operations--especially in the tactical area--because of the following factors:

- Tactical C^3 systems of U.S. services and agencies will increase in usage in operational situations as compared to strategic or defense systems.
- The advances in communications and computation technology will increase the availability of C^3 data both laterally and up or down command links.
- Allied nations will become increasingly sophisticated in the procurement and use of C^3 equipment and systems.
- The National Command Authority will be able to take advantage of new and advanced C^3 technology so as to make all phases of tactical operations more immediately responsive to national objectives.

Current and completed joint service and combined allied C^3 system interfaces were analyzed and used as a basis for projecting the requirements in this Roadmap. Concentration is on the U.S. tactical system interface and interoperation with indigenous forces since these are the most complex. By 1985, the impact of the Nixon Doctrine of self-reliance will combine with an increase in the availability of C^3 technology to cause a situation wherein U.S. allies will have acquired significant C^3 capabilities. The C^3 systems of the foreign nations will not necessarily reflect U.S. systems but will tend to reflect their own national goals, viewpoints, military structures, and doctrines. Extrapolating from the current difficulties that U.S. agencies are experiencing in achieving interoperability of U.S. and foreign nation systems will be a significant problem.

Coordination between or among military services and Defense agencies to effect C^3 system operability is required whenever new or existent systems are interfaced with new or existent systems that are under the control of some other service or agency. Once a need for interoperation is defined, coordination is required throughout the life cycle of the interfacing systems. The kind, extent, and level of coordination required will vary with the type of system and interface development, the organizations involved, etc. The JCS is assigned responsibility for monitoring the coordination, validating the requirements and providing guidance to the services; each service has designated a point of contact and provides copies of each service-validated statement of operational requirement to the other services.

A basic problem with the current procedure is that it causes, or allows, the interoperability requirements of the system to be addressed late in the system's conceptual and design stages or, in many cases, not until the system has been acquired and is undergoing field-testing. As a result, increased costs are incurred and the interoperated systems do not fully realize the potential benefits of their interoperation. Additionally, interoperation of U.S. systems with those of foreign allies is not based on long-range operational and technical plans and usually requires a high degree of modification of equipment and systems which results in lowered potential performance. The projects covered in this Roadmap will help to alleviate these interoperability problems for system operation, both joint and combined, in the 1980-1990 time period.

B. TASK DESCRIPTIONS

1. Interoperability Software Monitoring (17.1)

This project involves initiating action to ensure that future Air Force studies on data base/data exchange include consideration of potential joint and combined applications. Factors which may impact joint or combined operations include data base design methods, data base query techniques, required data base management, accounting and control procedures, security keys and priority standards.

2. Interoperability Hardware Monitoring (17.2)

This activity includes initiating action to ensure that Air Force computer mainframe studies consider interface buffer applications.

3. Project NIMBLE (17.3)

a. Minimum C^3 Analysis--This project involves initiating a mission analysis on required technology to support operational functions and to define the type and amount of Air Force support which can be provided to allied commands in combined operation with minimum in-country participation of Air Force units.

b. Design, Development, and Integration of Modular Prototypes--This task includes designing, developing, and integrating in-country sensors, reporting devices, and displays with out-country ADPE and operational software. Data reporting and data transmission requirements, including RPV and satellite relays, would be identified.

c. Operational Support for Users--This project would aim to provide installation and technical support to allied users.

4. Project NIMBLE Operational Simulation (17.4)

This project involves designing, developing, and implementing multiple-level operational simulation exercises for Project NIMBLE. The multiple-level aspects are concerned with exercising single components, integrated components, and whole systems. These exercises would not only be used as a vehicle to determine effectiveness of system operation and interoperability with U.S. capabilities supporting the indigenous forces; they would also be used as a training and orientation vehicle for the indigenous forces. This latter application is extremely important in achieving psychological acceptance of the equipments, or system, by the indigenous user. For this reason, the operational simulation should utilize controlled normative exercises to provide corrective guidance to the user personnel as they participate in the exercise.

The installation and testing phase is critical since the majority of the user personnel are exposed to the system as an operational entity for the first time. This is a period in which the basic attitudes of personnel who must operate and depend on the system are formed and solidified. If negative attitudes are established, if the users do not acquire confidence in the system, if they are less than cooperative in ensuring that the system accomplishes its goals, if they peremptorily reject the system and refuse to use it, the result may be a disaster from the point of view of both the users and the develcpers. Such disasters can and do happen but can be reduced or removed through the use of controlled exercises.

5. Tactical Software (17.5)

a. Develop QRC for Tactical Software--This task involves initiating an effort to develop and implement quick reaction software change procedures and techniques for use in tactical C^3 systems.

b. Install QRC Software in TACS/AWACS--This activity includes testing, integrating, and installing quick reaction software change capabilities in the TACS/AWACS.

6. NORAD/Navy Interface (17.6)

This project involves initiating a study of NORAD/Navy interfaces including effects of Navy interfaces on NORAD OTH-B, AWACS, and other NORAD systems.

7. Foreign C^3 Interface (17.7)

This project includes initiating action to develop and maintain a catalog describing all significant foreign C^3 systems with which the Air Force tactical C^3 systems may be required to interoperate. This catalog should include consideration of releasibility of sensitive sensor and intelligence data for each potential interface.

8. JTEA & Configuration Management Plans (17.8)

This activity involves developing a configuration management plan for interfaces which ensures that interface management, as defined in applicable military standards, is interpreted consistently throughout all future C^3 procurements. The interface management plan would be promulgated throughout DOD after acceptance.

This project would also include initiating action to study the replacement of the current executive agent concept used to manage C^3 system interface programs with a concept of management employing a joint test and evaluation agency (JTEA), among others. Proposed standards and methods for JCS/DOD consideration would be prepared.

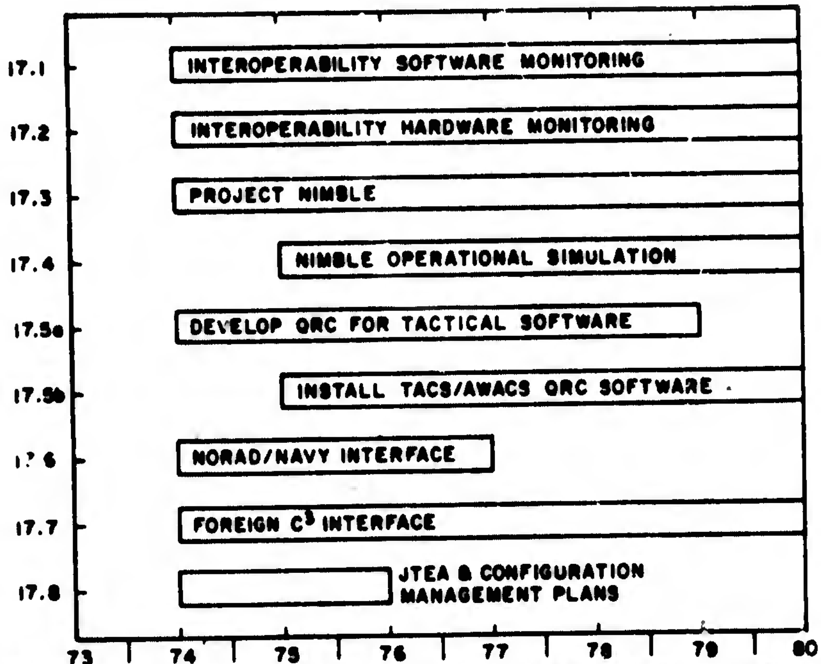
C. FACTS AND FIGURES

Table XI-XVI gives some pertinent facts and figures for Roadmap #17: Interservice Coordination; data are presented pictorially in Figure XI-16.

TABLE XI-XVI

ROADMAP #17: FACTS AND FIGURES

No.	Title	Fiscal Years Active	Peak Annual Rate, $\$10^6$ (No. of Efforts)			Existing Projects			Types	Overlap	
			Min.	Reas.	Crash	ID	Years	$\$10^6$		ND	%
17.1	Interop. Software Monitor	74→83	0.05 (1)	0.1 (2)	0.15 (2)				AVD-50, EGD-50		
17.2	Interop. Hardware Monitor	74→80	0.1 (1)	0.2 (2)	0.25 (2)				AVD-33, EGD-33, OPD-33		
17.3	Project NIMBLE										
	Analysis	74→76	0.2 (1)	0.4 (2)	0.6 (3)				EGD-100		
	Design Develop.	74→81	2.0 (1)	6.0 (1)	8.0 (1)				EGD-100		
	Opn. Support	80→83	0.5 (1)	1.0 (2)	3.0 (3)				OPD-100		
17.4	NIMBLE Operational Simulation	75→82	0.5 (1)	1.0 (2)	2.0 (3)				OPD-100	1.10	25
17.5	Tactical Software										
	Develop QRC	74→79	0.4 (1)	0.7 (2)	1.0 (3)				EGD-50, OPD-50		
	Install QRD Software in TACS/AWACS	75→81	0.2 (1)	0.5 (2)	0.8 (3)				OPD-100		
17.6	NORAD/Navy Interface	74→77	0.1 (1)	0.2 (1)	0.4 (2)				EGD-100		
17.7	Foreign C^3 Interface	74→81	0.2 (1)	0.3 (1)	0.4 (1)				OPD-100		
17.8	JTEA and Configuration Management Plans	74→76	0.1 (1)	0.25 (1)	0.3 (1)				EGD-100		



$, MILLIONS	FY74	FY75	FY76	FY77	FY78	FY79	FY80
MINIMUM	3.15	3.85	3.65	3.55	3.45	3.45	3.95
REASONABLE	8.15	9.65	9.65	9.00	8.80	8.80	9.80
CRASH	11.10	13.80	13.80	13.00	12.60	12.60	15.60

Figure XI-16. Roadmap #17: Interservice Coordination

XIX. ROADMAP #18: USAF HARDWARE LABORATORY

A. BACKGROUND AND OBJECTIVES

There are two prime functions required of the Air Force hardware laboratory. The first is to provide a centralized source of in-house computer hardware expertise; second is to provide a broad spectrum of active computer hardware system support.

1. In-house Expertise

Currently, the Air Force has a broad base of both military and civilian computer experts and professionals. These personnel are generally participating in supporting functions throughout the various commands and civilian agencies. Accordingly, this large source of specialists has no identity by which it can be called and efficiently used (in other than segmented support efforts). Because of the command and control demand for this hardware expertise, it would appear advisable to provide the necessary environment which would attract the personnel required, both civilian and military.

The main function of this core of professionals is to supply and provide the technical direction to and for industry in order to ensure that the Air Force needs are satisfied. Furthermore, these specialists must continually review the present and future needs and provide the forecast, justification, and direction required to sponsor and obtain a progressive R&D budget and program.

2. Computer Hardware Systems

Requirements forecast for the 1980s indicate a need for special-purpose computers and special-purpose hardware support. Some of these special supporting needs are: a flexible and variable computational communications network, hardware computer simulators, modular design engineering, user-transparent hardware systems, software-hardware tradeoffs, a low-cost "smart" display terminal, federated computer interfaces, and command and control application computers.

a. Computational Network--There is a need within the Air Force, and particularly in the command and control area, for a wideband digital communication system among all of the major computer facilities. This computational network is extremely important in/to the fabrication and certification of software. The system must be flexible and variable in configuration, and no fixed installations should be required (i.e., with the transportability and convenience of the telephone modem). The computer hardware laboratory would be a major node in this computational network. The laboratory would supply for this network an on-call inventory of special-purpose equipments for short-term utilization (i.e., fast Fourier transforms, tracking or surveillance algorithms, servosystems, etc.). Mass bulk-storage facilities would be available to the computational network in order to handle peak and emergency digital memory requirements that might occur. The laboratory will also have the capability of combining, assembling, and interfacing special computer systems to meet the special requirements of the users of the computational network.

b. Hardware Computer Simulator--There is a need for a large and sophisticated miroprogrammable computer facility to support Air Force R&D programs. Since a microprogrammable computer can be programmed to operate or function as any other computer, it can be used in a variety of roles. By simulating the instruction set, any machine language computer program can be written, debugged, or operated on it. For special applications, the optimum computer design could be formulated. In a computer sytem testing mode of operation, a number of computers along with their interface hardware could be simulated to whatever extent is necessary, and then the entire system could be checked out and debugged. This hardware computer simulator would be a vital supporting hardware member of the computational network.

c. Modular and Transparent Systems--In order to satisfy the information processing requirements of the 1980s, it is important that the data processing systems employed by the Air Force be modular and transparent. The modularity will provide the capacity for variable system configuration and adequate growth capability. When the idiosyncracies and hardware-peculiar or hardware-dependent characteristics of a system are transparent to the user, the utility of that system is greatly enhanced. In the case of a hardware module, the user could be another hardware module or system, or a programmer. It would be a function of the hardware laboratory to exercise the specified modularity of commercially available modular hardware to verify their building-block capability. It is further necessary to ensure the required transparency by specifying or providing commercially available interfacing hardware to or between modules or systems. Accordingly, the hardware laboratory must be in a position to design, specify, fabricate, and interface modular-transparent computer systems.

d. Software versus Hardware Tradeoffs--With the extremely disproportionate ratio between the costs of software and hardware in Air Force systems, a prime function of the hardware laboratory would be to determine how hardware could be used to equalize this ratio. In this respect, hardware standards for all software interfaces could be identified; these standards could result in the long-term utilization of existing software, as well as providing specifications for future software production. The utilization of direct execution higher-level language machine instructions probably could have the greatest impact in equalizing the cost ratio between software and hardware, and it would be the responsibility of the hardware laboratory to sponsor, develop, and promote this hardware technology.

e. Display Terminals--Because of the expected large total numbers of display terminals that will be required, it is important that the hardware laboratory push and direct the commercial suppliers toward producing a low-cost terminal display which will satisfy the Air Force requirements. Besides being low cost, there are three capabilities which are most important to the Air Force requirements: mass memory, autonomous computation, and non-mechanical. The LSI mass memory is vital to the graphics requirement as well as supplying the recording capability of thousands of characters of information for the interactive alphanumeric display terminal. The self-contained computational capability of the terminal is necessary for the off-line operation. A non-mechanical system is required in order to minimize maintenance and to provide a hermetically sealed package.

f. Federated Computer Interface -- Because of the large number of independent information-processing facilities which must have limited digital communications with one another, there is an important requirement for a convenient adaptable interface hardware processor (computer). This interface computer must not only provide data interfaces (formats, word size, etc.) but must also guarantee the two-way transmission of commands, instructions, and answers. Furthermore, this capability must be transparent to each facility. (In other words, each computer requests and receives instructions, data, and answers in its own software and hardware language as if it were connected to its own twin.) This capability could be satisfied by one of the high performance microprogrammable MINI computers or a modification thereof. Inasmuch as there is also a commercial requirement for such interface equipment, the MINI computer manufacturers would probably be the ideal source for such implementation. The computer hardware laboratory would have a directing role in the design, development, procurement, and implementation of these interface modules.

g. Hardware Redundancy--In most command and control applications in the 1980s, it will be necessary that systems are never down and that data are never lost. Furthermore, when a line replacement item fails, it must identify itself. A capability to replace all failed line replacement items without disturbing the full productive capability of the information-processing equipment will also be needed. This places some severe requirements upon the system configuration and upon the maintenance procedures. It would be the responsibility of the hardware laboratory to see that these requirements are satisfied. Again, these same requirements are needed for commercial applications, and the Air Force could provide the stimulus for their development. Although there are many sophisticated methods for providing these capabilities, the "brute-force" approach of redundancy is the best method available. This is particularly important when hardware forecasts indicate reduced hardware costs with

LSI technology (MSI, etc.), and with labor and design costs increasing. Brute-force hardware redundancy can be provided on the system level or on the lowest replaceable unit level. Total system redundancy probably will suffice for the near-term solution, while redundant lowest replaceable units will provide the terminal solution. The hardware laboratory must play an important role in the last approach.

h. Security by Hardware -- Compartmentation and file security are critical issues in all command and control applications. Most near-term approaches to this problem are using an executive type of software with the possible utilization of some special hardware instructions or limiting registers. This approach is not satisfactory in the long term for, from the point of view of strict security, almost anything that is done by software can be changed by software, either deliberately or accidentally. It appears that the only exacting solution to compartmentation is the utilization of redundant hardware.

In order to avoid a requirement for special-purpose compartmentized hardware systems, the security hardware must be a peripheral add-on piece of equipment such that commercially available systems can be utilized for Air Force needs. The security hardware would not provide any data processing capability but would be a parasitic piece of equipment which would monitor the addressing registers of the data processing unit and simultaneously process its own files to ensure that a breach of security was not about to be made. The only hardware interfacing that would be required would be to the address registers and the start and stop commands of the processing unit. No software interfacing would be necessary. The hardware laboratory could provide the technical direction necessary to achieve these security/compartmentation goals.

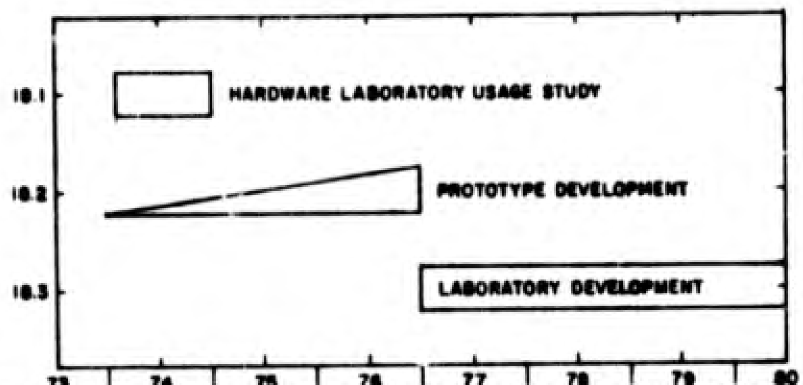
i. The Command and Control Application Computer--The applications computer hardware for the 1980s must be modular with plug-in and plug-out capability. The modules of the applications computer must also be plug-in expandable within themselves (i.e., word length, expansion of pipeline, random access, cache, and/or associative memories, etc.). Further, it is highly desirable that direct execution high-level language machine instructions be available for certain command and control applications. To acquire these capabilities, the hardware laboratory would be responsible for advising, directing, and technically justifying the development, procurement, and testing of these systems.

B. TASK DESCRIPTIONS

Roadmap #18 consists of three tasks which will be developed to support the goals outlined above. These are a hardware laboratory usage study, prototype development, and laboratory development.

C. FACTS AND FIGURES

Table XI-XVII presents pertinent data for Roadmap #18: Hardware Laboratory. Figure XI-17 depicts the research and development program that is required in order to support the hardware laboratory and its responsibilities.



$, MILLIONS	FY 74	FY 75	FY 76	FY 77	FY 78	FY 79	FY 80
MINIMUM	1.02	1.02	1.00	4.00	3.00	3.00	3.00
REASONABLE	3.02	3.02	·3.00	8.00	5.00	5.00	5.00
CRASH	5.02	5.02	5.00	16.00	10.00	10.00	10.00

Figure XI-17. Roadmap #18: Hardware Laboratory

TABLE XI-XVII

ROADMAP #18: FACTS AND FIGURES

No.	Title	Fiscal Years Active		Peak Annual Rate, $\$10^6$ (No. of Efforts)			Types
				Min.	Reas.	Crash	
18.1	Hardware Laboratory Usage Study	74	75	0.02 (1)	0.02 (1)	0.02 (1)	RES-100
18.2	Prototype Development	74	77	1.00 (1)	3.00 (1)	5.00 (1)	EXP-100
18.3	Laboratory Development	77	80	3.00 (1)	5.00 (1)	10.00 (1)	AVD-100

REFERENCES

1. D. Teichroew and H. Sayari, "Automation of System Building," Datamation, pp. 25-30, August 14, 1971.

2. R. V. Head, "Automated System Analysis," Datamation, pp. 23-24, August 14, 1971.

3. D. W. Kosy, "Experience with the Extendable Computer System Simulator," R-560-NASA/PR, The RAND Corporation, Santa Monica, California, December 1970.

4. J. C. R. Licklider, "Underestimates and Overexpectations," in ABM: An Evaluation of the Decision to Deploy an Antiballistic Missile System, Abram Chayes and Jerome B. Wiesner, eds., Signet Books, published by the New American Library, Inc., New York, N. Y., pp. 122-123, May 1969.

5. Barry W. Boehm, "Some Information Processing Implications of Air Force Space Missions: 1970-1980," RM-6213-PP, The RAND Corporation, Santa Monica, California, January 1970.

6. "Software Engineering Techniques," report on a conference sponsored by NATO Science Committee, Rome, Italy, October 27-31, 1969.

7. Fortune, November 1966.

8. EDP Weekly, p. 2, February 2, 1970.

9. Brooks, "Why is the Software Late?" Data Management, August 1971.

10. AF Regulation 800-2, July 27, 1971.

11. "Systems Management: Configuration Management for Systems Equipment, Munitions, and Computer Programs," Draft AFSC/AFLC Manual 375-7, undated.

12. W.A. Hosier, "Pitfalls and Safeguards in Real-Time Digital Systems with Emphasis on Programming," IRE Transactions on Engineering Management, vol. EM-8, pp. 99-115, June 1961.

13. "Security Controls for Computer Systems," Willis H. Ware, ed., R-609, The RAND Corporation, Santa Monica, California, 1970.

14. "Mission Analysis on Command Control and Communications for Theater Air Operations," vol. V, AFSC Study Facility, ESD, Hanscom Field, Mass., August 1971, SECRET.

15. "416M System Package Program, PE 12314F, Back-up Interceptor Control System (BUIC III)," ESSGL-70-001, Hq ESD, January 1, 1970, SECRET NOFORN (exc. Canada).

16. T.E. Bell, "Computer Performance Analysis: Measurement Objectives and Tools," R-584-NASA/PR, The RAND Corporation, Santa Monica, California, February 1971.

17. Benjamin F. Schemmer, "DOD's Computer Critics Nearly Unplug Ailing Worldwide Military Command and Control System" Armed Forces Journal, p. 23, December 21, 1970.

18. J. Sockman, Computers, System Sciences, and Evolving Society, Wiley, 1967.

19. "Associative Processing Technology, A National Resource for Defense Status and Application -- A Surveillance Data-Processing Report," Goodyear Aerospace, January 3, 1969.

20. A.G. Oettinger and S. Marks, Run, Computer, Run, 1967.

ANNEX A

SOME GUIDELINES FOR "ASSEMBLY-LINE" CERTIFICATION

ANNEX A

SOME GUIDELINES FOR "ASSEMBLY-LINE" CERTIFICATION

I. INTRODUCTION

Most of the promising approaches to software certification involve increased concern with certification aspects before and during software production rather than after-the-fact testing. The ideas developed below on "assembly-line" certification provide some guidelines for incorporating such changes into Air Force software development procedures.

The certification of software can be defined as that process which is necessary to assure that the information-processing function is performed according to the desired requirements. In past and present applications, this certification process has been accomplished by massive verification and validation* efforts which have been performed upon the software after the programming is finished. For example, in a recent Air Force survey, it was determined that it took 12 man-months to verify a 13,000-instruction program for the flight computer in the Saturn V's instrument unit using current software production procedures. This verification was in addition to the 48 man-months required to verify the original software design specification.

The validation effort required for software qualification of a computer program can also be staggering. A single, possibly catastrophic, error in a 2000-instruction program might be caught by a carefully designed after-the-fact check consisting of 50 to 100 validation runs through a general-purpose computer facility on the ground. When the program involves 25,000 instructions, it might take 1000 to 2000 runs to provide the same amount of confidence. It can be seen that a great deal of money (typically $750,000 for a man-rated 8000-instruction program) and effort must therefore go into today's validation of software.

*Verification is the process or effort required to assure that the equations and logic flow (program requirements/specification) provide the desired performance of the system which is being supported by the information-processing system. Validation is the process or effort required to assure that the computer program performs the function described by the equations and logic flow as it is defined in the requirements/specification document.

Changes to a validated program can also be expensive. The same Air Force study quoted above cites the time required for reprogramming one to ten instructions for the 9000-instruction Titan III computer program using current procedures as anywhere from five days to two weeks.

If relatively few of these changes are dispersed throughout the computer program, the validation requirements can approach those required for the original new program.

The above statistics are all related to space programs where a catastrophic software error can result in the loss of a space vehicle or the failure of its mission. In command and control applications, a catastrophic software error has far more serious consequences, in that national survival may be at stake.

II. FEASIBILITY OF AFTER-THE-FACT CERTIFICATION

Extensive efforts and certification programs have been initiated in order to try to provide creditable software. To date, the experience has been that, no matter how large the effort, these certification efforts do not result in creditable software (see Aerospace Technical Memorandum ATM 69(4112-34)-96 for a case in point). Furthermore, there is absolutely no evidence that, by following the "make it bigger and centralize" philosophy (e.g., via a centralized Air Force software certification facility), an after-the-fact certification (verification and/or validation) will provice successful creditation of software in the future.

III. THE SYMPTOMS AND THE DISEASE

Because of the importance of command and control software and the implied stakes involved in successful certification, it is important to look closely at the interrelations among current certification procedures, the current state of the art, and the forecasts for the decade between 1980 and 1990. It is immediately apparent that the current after-the-fact certification program is attempting to find the symptoms of the problems (i.e., software errors and shortcomings), rather than trying to cure the disease that is causing them. Specifically, instead of correcting the errors after they occur, the causes of these errors should be corrected or changed. One further observation can be made: the prime source of errors is attributable to human involvement, and minimizing this involvement will also minimize the occurrence of errors.

Figure A-1 is a caricature of current software production procedures, and it is within or during these procedures that the causes and sources of the software errors and shortcomings develop. By evaluating these procedures, it will be possible to determine what the current problems are, what has to be done to correct them, and whether it will be possible to correct them in future command and control systems.

The first column of Figure A-1 shows a group of specialists who are required to prepare the guidance equations for a space mission. These specialists could as easily be associated with any command and control system or subsystem. In the second column, the specialists are verifying their equation and logic flows so that they can prepare an interface requirements/specification document. The verification is being performed using open-loop simulations which supply the spectrum of inputs that can be expected by each subsystem. After the specialist is satisfied that the iterations through the verification loop have resulted in an adequate set of equations and logic, his final interface document can serve as the interface between the subsystem specialist and the systems analyst.

Two points should be noted: first, the software used in the open-loop verification simulation probably has the highest credibility or confidence that can be acquired, as it was prepared by the most qualified individual and checked out by the same expert; second, from this point on, that confidence level will deteriorate. For example, the specialist has only confirmed that a computer program is verified; unless the computer program itself is used, the possibility of an inconsistency between the verified software and the requirements/specification documents exists which degrades the confidence level. This degradation exists independently of the amount of validation, and the system analyst's interpretation through the interface document also provides a source of error.

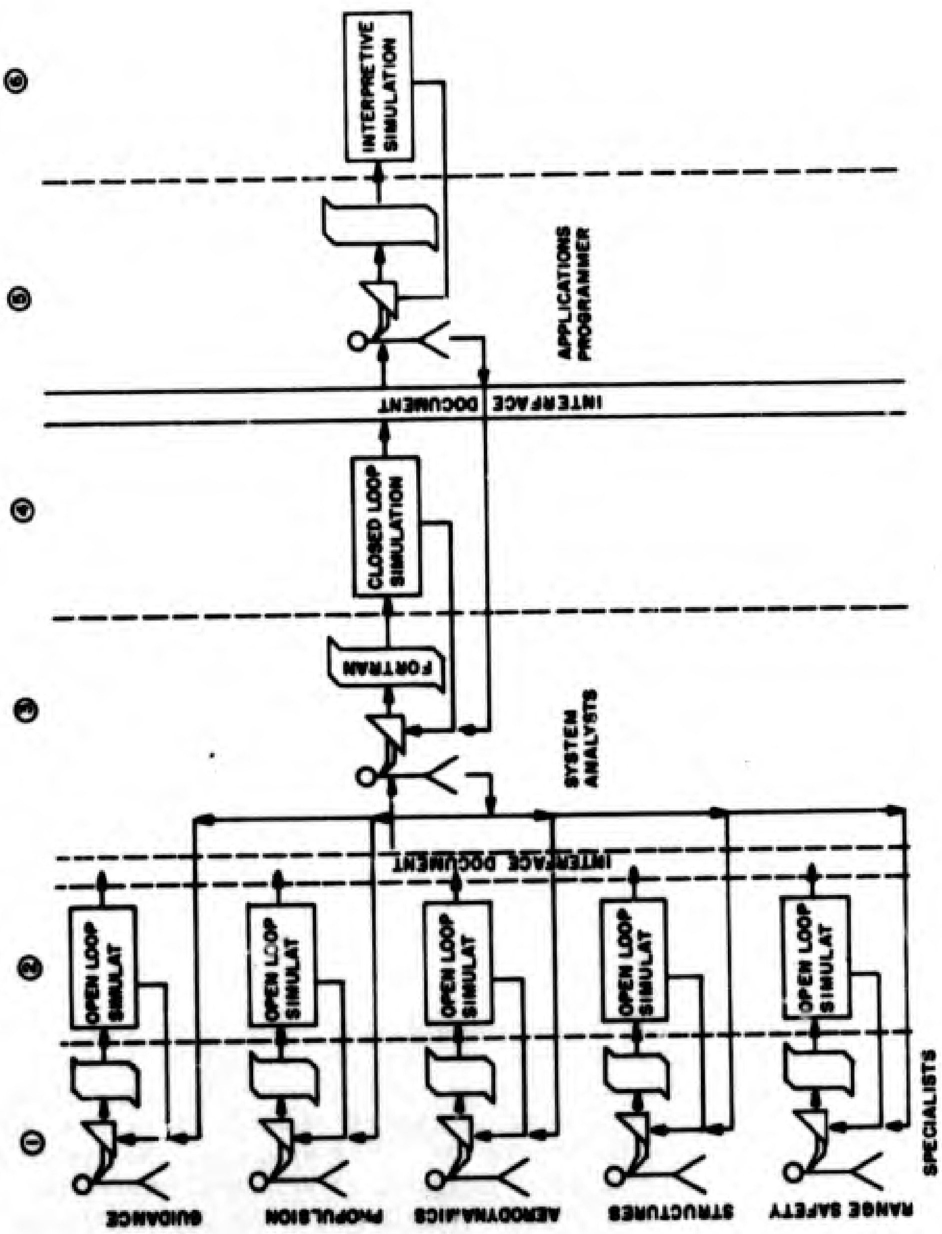


Figure A-1. Current Software Production Procedures

Accordingly, five corrective actions can be taken: first, more than one specialist should work the same problem in order to minimize the element of human error; second, each specialist should write the program in a higher-order language; third, all open-loop check runs should be performed continuously, simultaneously, and automatically on duplicate programs and computers and all discrepancies should be indicated to all specialists (and other concerned parties); fourth, the verified software computer programs should be used as the requirements/specification document; fifth, the human should be removed from the interface, and a reproducible software tool should be used.

Column three indicates the systems analyst who will assemble the various specialists' requirements/specification documents into a working software package. To date, this is generally done by reprogramming the specificataion document. Column four shows the closed loop scientific simulation that is being used to verify systems operation. After the system analyst is satisfied that the systems verification iterations (including an iteration back through the specialists) have resulted in an adequate set of systems equations and logic flows, the systems requirements/specification document is prepared for use as an interface for the implementation programmer.

Several points should be noted here: first, the software that has been verified using the closed loop simulation has the highest level of confidence that the systems software will acquire; second, from this point on, confidence in the system will deteriorate. For example, the analyst can only confirm that the computer program of the system was verified; unless the computer program itself is used, there may be inconsistencies between the specialists' requirements document, the verified system's software, and the system's requirements/specification document. This degradation exists independent of the amount of validation and the applications programmer's interpretation through the interface document will also degrade the confidence.

Accordingly, five corrective actions can be taken: first, the function of the systems analyst should be duplicated to reduce the probability of the human error; second, the systems analysts should write their programming in a higher-order language; third, all scientific simulation loop check runs should be performed continuously, simultaneously, and automatically on the duplicate programs and all discrepancies be indicated to all specialists, system analysts, and all other concerned parties; fourth, the verified software systems computer program should be used as the requirements/specification document; and, fifth, the human element should be removed from the interface, and a software tool should be substituted.

Column five shows the applications programmer, who is programming the specification in machine (or assembly) language for the applications hardware. It is at this point that the most significant degradation in software credibility occurs. This degradation is caused by the human element and also by hardware differences and limitations. The machine language programming is the primary source of human errors. The machine language programming requirement is generally necessary because of uncertainties in the computational speeds and the storage availability of the application hardware. Equation and logic uncertainties result from the changes in word length (accuracies), fixed point arithmetic (no floating point), different interrupt hardware, limited input-output facilities, and so forth.

Column six shows the after-the-fact certification of the software, wherein personnel will try to find and correct the validation errors and verification uncertainties which have been caused and which have been accumulating throughout the entire software production procedure. It should be noted that an interpretive simulation of the application hardware will also be used, thus causing additional degradation of the credibility of the entire process. When limitations of computational speed, memory size, word length, etc., are found to exist in or with the applications computer hardware, the applications programmer must reprogram in an attempt to optimize around these deficiencies. This optimization generally results in more complex coding. Complex coding generally leads to a higher error rate.

The following corrective actions are suggested: first, verified programs of the system should be used for the applications program; second, the applications hardware should not be obtained until scientific verification simulation of the system determines the CPU size, speed, and supporting storage requirements; third, the hardware should be designed with adequate (plug-in, plug-out) input-output facilities; fourth, the application programmer should be replaced by a compiler (or direct application instructions set); fifth, the hardware should be designed as expandable plug-in modules so that word length, storage, and throughput can be modified easily; and, finally, the after-the-fact certification should be a point-for-point and computer-to-computer check between the scientific closed-loop verification analysis and the actual application hardware outputs (no interpretive simulation and no human involvement).

IV. FORECASTS FOR 1980 TO 1990

Given the high risk, it is necessary to review the software and hardware forecasts in order to determine if the corrective actions outlined above are feasible and, further, if they may be reasonably forecast for the 1980-1990 time period.

A. SOFTWARE FORECAST

A review of the software forecasts indicates that the corrective actions outlined above are a small portion of what are called software engineering procedures and techniques. The software engineering procedures and techniques that are important to the creditable certification of command and control software for the 1980-1990 time period can be outlined as follows:

- The command and control requirements should be subdivided into the smallest units of responsibility and qualification;
- The most expert specialist should be assigned to fulfill and specify the equations and logic requirements;
- Each specialist should be backed up with a duplicate analysis, with the depth of redundancy being dependent upon the complexity and sensitivity of the solution;
- The specialist should work and interface with the open-loop verification process with a higher-order language;
- The specialist should work on-line with an interactive console;
- The specialist's verified requirements/specification document should be the verified higher-level program;
- The system analysts should assemble the specialist-verified programs using a higher-order language;
- Each system analyst effort should be duplicated to a minimum redundancy of two and, preferably, three working systems at three different computer facilities, each having different supporting hardware and software;

- All system closed-loop scientific verifications should be iterated between all subsystem specialists and all system analysts and, furthermore, a point-for-point and computer-to-computer check should be made continuously during all verifications;
- The verified closed-loop scientific system's software should be used to specify the applications computer characteristics (i.e., speed, CPU storage, input, output, etc);
- The verified closed-loop scientific system's software should be compiled for the applications computer;
- Final verification should be a point-for-point and computer-to-computer check between the closed-loop scientific verification analysis and the actual application hardware outputs; and
- All support software (operating systems, compilers, interactive programs, etc.) must be certified by a point-for-point and computer-to-computer check before it is used.

B. HARDWARE FORECAST

Although only a few hardware requirements have been specified above, there are a number of hardware support requirements that are vital to the implementation and success of continuous, simultaneous, and automatic quality control of software. A review of the hardware forecast indicates that these requirements will be satisfied by the 1980s.

The hardware requirements that are important to the creditable certification of command and control software for the 1980-1990 time period can be outlined as follows:

- On-line display terminals with independent computational capability are required for each specialist and system analyst;
- A real-time wideband digital computational network is required between contractors and the technical direction command, agency, or contractor;
- The computational network must be transportable or readily linkable with no fixed installation requirements (i.e., the simplicity of the time-shared telephone modem of today);

- The computational network must be secure, and the security mechanization must be implemented by hardware and be transparent to all users, both application and system specialists;
- The computers associated with the digital network must be modular, expandable, and have plug in-plug out capability in order to vary their configurations to meet all information processing requirements;
- The computational network must be composed of multiple computer systems (not a large central computer), and it is preferable that the systems not be identical;
- The network computers must be compartmentized; this compartmentation must be implemented with plug-in and plug-out peripheral hardware and must be transparent to all levels of users;
- The executive and operating system (EOS) functions must be hardware-implemented as part of the instruction set of the network computers and must function as an EOS higher-order language instruction; this would provide transparency of the EOS functions for both systems programmer and application user;
- The applications (field, airborne, silo, etc.) computer hardware must be modular with plug-in and plug-out capability;
- All modules of the application computer must operate asymmetrically and must be transparent to each other;
- The modules of the applications computer must be plug-in expandable within themselves (i.e., word length, expansion of arithmetic pipeline, random access, cache, and/or associative memories, etc.);
- It is important that the application computer have plug-in and plug-out input-output capability which will permit it to become a member of the computer network system;
- It is highly desirable, if not required, that all computers in the computational network have, and operate under, a direct execution higher-order language machine instruction set, thus eliminating the need for, and the certification of, compilers, editors, assemblers, and so forth; and

- Hardware certification of the computational network and its utility software will be accomplished by the continuous, simultaneous, and automatic duplicate computation of all tasks on all computers with a point-for-point and computer-to-computer check of the outputs.

V. ASSEMBLY-LINE CERTIFICATION

Described above are the quality assurance requirements to obtain continuous, simultaneous, and automatic certification of both the software and the hardware throughout the software development. These certification requirements have not been determined by committee vote or sanction, but by analyzing the present certification procedures and problems and then identifying the error sources and certification shortcomings. The resulting certification procedures and requirements are accordingly those which are necessary to eliminate these error sources and shortcomings. It should be apparent that an assembly-line certification has been established (see Figure A-2) wherein a step-by-step certification is continuously being made by continuously checking computer against computer, specialist against specialist, system analyst against system analyst, application computer against application computer, software system against software system, and contractor against contractor. The assembly-line development and certification of software is meant to be identical to and with hardware projects utilizing multiple competitive contractors with the addition, however, of continuous competitive checking and comparison of the software development of all contractors throughout each step of the software production assembly line.

A. METHOD OF OPERATION

The modus operandi of the certification assembly line would be as follows.

1. Developing Blueprint

A detailed blueprint of the step-by-step quantification of requirements through the software production assembly line would be developed and used as a milestone or pert chart. This assembly-line blueprint would identify the step-by-step assembly and scheduling of the software through or into the computational network.

2. Developing First-Order Model

An ideal first-order model of the assembly-line blueprint would be developed by the assigned technical direction (T. D.) command, agency, or contractor, and would serve as the perfect case or mechanism for software comparisons throughout the software development. This model will define standard nomenclature and variable definitions.

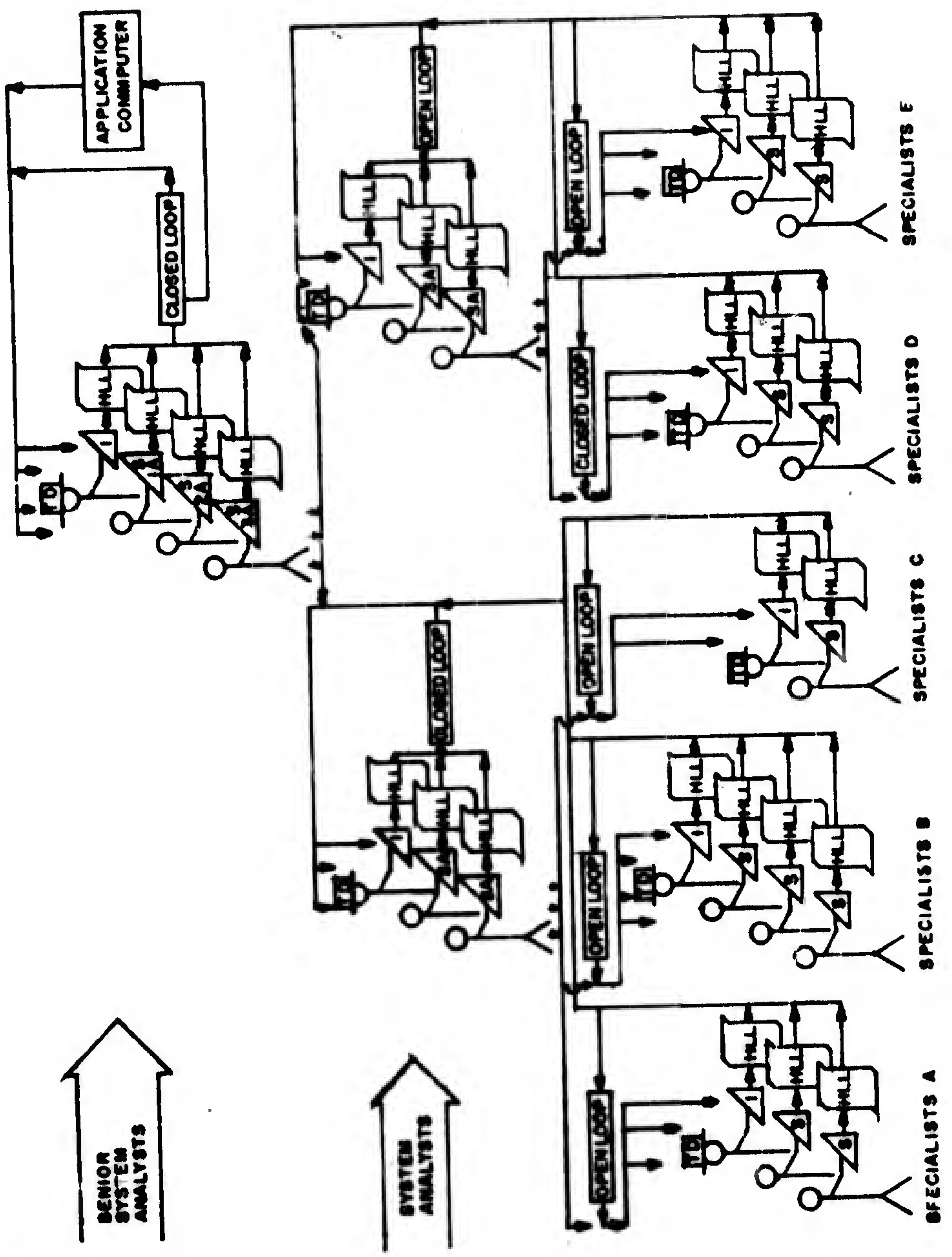


Figure A-2. Assembly Line Certification

3. Developing Testing Procedures

Standard and perturbation tests and/or testing procedures would be developed by the T. D. command, agency, or contractor for each quantification step of the assembly-line procedure.

4. Establishing Requisites for RFP Participation

The facilities of all candidate contractors must meet certain specifications and standards of the computational network in order to be eligible for RFP participation (if the facilities of the network are not being used).

5. Linking Contractors to Network

The multiple competition contractors are linked to the computational network (if the facilities of the network are not being used).

6. Certifying Hardware

Certification of the computer hardware and supporting utility software is performed by testing all computer systems against a standard software/hardware certification pack ge which has been prepared by the T. D. agency, command, or cont actor. This certification is performed by a point-for-point and computer-to-computer check against the required standard outputs. Continuous hardware certification is obtained by shadowing and duplicating the workload of each computer by all of the other participating computers of the network. This will assure against and catch any degradation within the computational network.

7. Developing Step-by-Step Blueprint

Each specialist will develop and satisfy the specification and step-by-step assembly blueprint using an on-line display terminal which has a self-contained computational capability.

8. Testing the Blueprint

When the specialist is ready to check against the ideal model (prepared by the T. D. agency), the specialist updates his computer program that is stored in the network, and a test run is made. Simultaneously, the same run is made on the ideal model as well as all competitive contractor software. The results of the specialist run, the ideal model's run, and all the competitive software runs, as well as their differences, are all displayed to the specialist for his comparison and analysis.

9. Maintaining Software Proprieties

Software proprieties of each contractor are maintained by hardware control of the network compartmentation. Only the T.D. command, agency, or contractor has access to all software.

10. Testing Special Cases

Each specialist can run special test cases against his software package. The same test will be performed by the ideal perfect model and the competitive contractor's software, thus providing a spectrum of tests of all programs (including the ideal model).

11. Providing Contractual Incentives

Contractual incentives are based in part on the performance of each specialist model; accordingly, it behooves each specialist to develop software and tests which show and indicate his software package to be the best.

12. Updating Tests and Test Procedures

It is the responsibility of the T.D. agency (or command or contractor) to update the standard and perturbation tests and testing procedures by including or enlarging upon the individual contractor's test, thus providing the most compatible set which satisfies the specification requirements.

13. Developing System Requirements

Each contractor's system analyst will develop the system requirements necessary to assemble the specialist's software packages that he is responsible for, using a self-supporting interactive on-line display terminal. Initially, the systems analyst will utilize the ideal models, feeding back to the specialist any special problems that must be considered at that level.

When the systems analyst is ready to check against the ideal model (prepared by the T.D. command, agency, or contractor), the analyst's network computer is updated and a test run is made. Simultaneously, the same run is made on the ideal model as well as all the competitive contractor's software. The results of the systems analyst run, the ideal model's run, and all the competitive software runs, as well as their differences, are all displayed to the systems analysts for their comparison and analysis.

14. Performing Special Systems Tests

Special systems tests can be defined by the system analyst. These tests will be run automatically against the ideal model as well as the competitor's contract software, and all results and comparisons will be displayed. This will provide a full spectrum of system test for all systems programs (including the ideal model).

15. Conducting Competitive Contractor Tests

Contractual incentives will also be based upon subsystem and total system software performance, thus supporting the competitive testing described before for the specialists' efforts.

16. Updating System Tests and Test Procedures

It is again the responsibility of the T.D. agency or command to update continuously the standard and perturbated system tests and test procedures by utilizing the contractor's special tests.

17. Providing Feedback to Specialists

Each system analyst will function as a coordinator and will feed back to the specialist all problems associated with his software package.

18. Establishing Baseline Hardware Configuration

A baseline hardware configuration of all of the candidate application computers will be included into the computational network. The baseline configuration will have the characteristics defined by an extrapolation from this ideal system model (as defined by the T.D. agency or facility). The hardware laboratory's microprogrammable hardware computer simulator could be used in the baseline implementation. As inadequacies are determined in the baseline application computer's hardware, its configuration will be changed utilizing the plug-in module (or microprogramming) capability.

19. Formalizing Hardware Configuration

Upon completion of the systems software, the hardware configuration will also be formalized. Furthermore, the certification of both the software and the application hardware will have been accomplished simultaneously.

B. SYSTEM MAINTENANCE AND ALTERATION

All system changes and maintenance after the initial certification will be performed and certified in the same manner and with the same procedures as described above for the original software.

UNCLASSIFIED

Security Classification

DOCUMENT CONTROL DATA - R & D

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

1. ORIGINATING ACTIVITY *(Corporate author)* SAMSO/XRS P.O. Box 92960, Worldway Postal Center Los Angeles, CA 90009	2a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED
	2b. GROUP N/A

3. REPORT TITLE (6) INFORMATION PROCESSING/DATA AUTOMATION IMPLICATIONS OF AIR FORCE COMMAND AND CONTROL REQUIREMENTS IN THE 1980s (CCIP-85), VOLUME XI, ROADMAPS,

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

5. AUTHOR(S) *(First name, middle initial, last name)* (10)

~~Dr.~~ Barry W. Boehm, ~~Lt Col~~ Allen C. Haile, USAF

(12) 139p.

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
(11) May 1972	~~158~~	20

8a. CONTRACT OR GRANT NO.	9a. ORIGINATOR'S REPORT NUMBER(S)
b. PROJECT NO. (14)	SAMSO/XRS-71-1-Vol-11
c.	9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)*
d.	SAMSO-TR-72-200-Vol-11

10. DISTRIBUTION STATEMENT "B" Distribution limited to U.S. Government Agencies only; Test and Evaluation, 16 Aug 72. Other requests for this document must be referred to HQ SAMSO/XRS, P.O. Box 92960, Worldway Postal Center, Los Angeles, California 90009.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY HQ U. S. Air Force

13. ABSTRACT

The Roadmaps presented in this volume are a culmination of the findings of the Mission Analysis on Information Processing/Data Automation Implications of Air Force Command and Control Requirements in the 1980s (CCIP-85). They indicate what should be done to alleviate problems identified in the body of the study. They do not indicate who should do a task, or where it should be performed. Although these issues are important, CCIP-85 is not a management study. As to level of detail, the Roadmaps indicate what to do at the functional level. They furnish guidelines and background information for preparing more detailed specific task summary statements.

PRECEDING PAGE BLANK - NOT FILMED

DD FORM 1 NOV 65 1473

UNCLASSIFIED

UNCLASSIFIED

Security Classification

ABSTRACT

The Roadmaps presented in this Volume are a culmination of the findings of the Mission Analysis on Information Processing/Data Automation Implications of Air Force Command and Control Requirements in the 1980s (CCIP-85). They indicate what should be done to alleviate problems identified in the body of the study. They do not indicate who should do a task, or where it should be performed. Although these issues are important, CCIP-85 is not a management study. As to level of detail, the Roadmaps indicate what to do at the functional level. They furnish guidelines and background information for preparing more detailed specific task summary statements.

UNCLASSIFIED