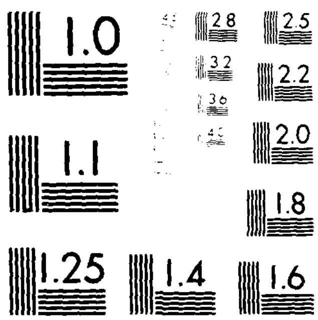
AD-A115 461   TRW DEFENSE AND SPACE SYSTEMS GROUP REDONDO BEACH CA   F/G 9/2
A STUDY OF EMBEDDED COMPUTER SYSTEMS SUPPORT. VOLUME IX. NATION--ETC(U)
SEP 80   F33600-79-C-0540
UNCLASSIFIED   TRW-34330-6003-UT-00-VOL-   NL

1 OF 1
AD A
115461

END
DATE
FILMED
07-82
DTIC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS 1963 A

34330-6003-UT-00

①

A STUDY OF EMBEDDED COMPUTER SYSTEMS SUPPORT

VOLUME IX

NATIONAL SOFTWARE WORKS INVESTIGATION

September 1980

APPROVED FOR PUBLIC RELEASE
DISTRIBUTION UNLIMITED

CDRL 04
Contract Number F33600-79-C-0540

Prepared for
Air Force Logistics Command AFLC/LOEC
Wright Patterson AFB, Ohio 45433

UNCLASSIFIED/UNLIMITED

TRW
DEFENSE AND SPACE SYSTEMS GROUP
ONE SPACE PARK • REDONDO BEACH • CALIFORNIA 90278

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO. AD-A115 461	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Embedded Computer System Support Volume 9-National Software Works Investigation		5. TYPE OF REPORT & PERIOD COVERED Final
		6. PERFORMING ORG. REPORT NUMBER 47169J
7. AUTHOR(s)		8. CONTRACT OR GRANT NUMBER(s) F33600-79-C-0540
9. PERFORMING ORGANIZATION NAME AND ADDRESS TRW Systems Inc One Space Park Redondo Beach, CA 90278		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS HQ AFLC/LOEE Wright-Patterson AFB, OH 45433		12. REPORT DATE Sept 1980
		13. NUMBER OF PAGES 49
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for Public Release, Distribution Unlimited.

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

18. SUPPLEMENTARY NOTES

Recommendations pertaining to administrative and programmatic initiatives are presented in this volume. Companion documents dated October 1981, September 1980, same contract number.

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

Materiel Management	ECS	Computerized Systems
Logistics Support	Weapons Systems	Avionics
Logistics Management	Technolgy Forecasts	
Planning	Program Management	

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

Abstract: This volume presents data that indicates the extent of feasibility and application of the national software works to current and future Air Force Logistics Command (AFLC) support requirements for Embedded Computer Systems. Additionally, a discussion of applicability of computer networking as a generalized aid to AFLC support requirements is presented. (See related volumes: LD 47169A, B, C, D, E, F, G, H, K and L).

ITEM 20 (con't)

Conclusions: See volumes 1 (LD 47169A).

Recommendations: See volume 1 (LD 47169A).

34330-6003-UT-00

A STUDY OF EMBEDDED COMPUTER SYSTEMS SUPPORT

VOLUME IX

NATIONAL SOFTWARE WORKS INVESTIGATION

September 1980

CDRL 04
Contract Number F33600-79-C-0540

Prepared for
Air Force Logistics Command AFLC/LOEC
Wright Patterson AFB, Ohio 45433

ONE SPACE PARK • REDONDO BEACH • CALIFORNIA 90278

TRW®
DEFENSE AND SPACE SYSTEMS GROUP

FOREWORD

This volume is one of nine individually bound volumes that constitute the Phase II Final Report "Study of Embedded Computer Systems Support" for Contract F33600-79-C-0540. The efforts and analyses reported in these volumes were sponsored by AFLC/LOEC and cover a reporting period from September 1979 through September 1980.

The nine volumes are

Volume	Title
I	Executive Overview (CDRL 05)
II	Selected ECS Support Issues: Recommendations/ Alternatives (CDRL 02A)
III	Requirements Baseline: Aircrew Training Devices (CDRL 02A)
IV	Requirements Baseline: Automatic Test Equipment (CDRL 02A)
V	Requirements Baseline: Communications-Electronics (CDRL 02A)
VI	Requirements Baseline: Electronic Warfare (CDRL 02A)
VII	Requirements Baseline: Operational Flight Programs (CDRL 02A)
VIII	ECS Technology Forecast (CDRL 03)
IX	National Software Works Investigation (CDRL 04)

CONTENTS

CONTENTS (Concluded)

TABLES

ILLUSTRATIONS

ABBREVIATIONS AND ACRONYMS

Ada	Standardized high order language
AFLC	Air Force Logistics Command
AGMC	Aerospace Guidance and Metrology Center
AISF	Avionics Integration Support Facility
ALC	Air Logistics Center
ARPANET	Advanced Research Project Agency Network
ATD	Aircrew Training Device
ATE	Automatic Test Equipment
AUTODIN	Automatic Digital Information Network
C-E	Communications-Electronics
DARPA	Defense Advanced Research Projects Agency
DCA	Defense Communications Agency
DOD	Department of Defense
ECS	Embedded Computer Systems
EDMS	Engineering Data Management System
EW	Electronic Warfare
FE	Front End
FM	Foreman
FP	File Package
GIM	General Information Management
ID	Identification
IMP	Interface Message Processor
IOC	Initial Operating Capability
IP	Internet Protocol
IV&V	Independent Validation and Verification
MENA	Mission Element Need Analysis
MSG	Communications protocol
NCP	Network Control Program
NSW	National Software Works
OC-ALC	Oklahoma City Air Logistics Center
OFP	Operational Flight Program
OJT	On-the-Job Training

ABBREVIATIONS AND ACRONYMS (Concluded)

RADC	Rome Air Development Center
RJET	Remote Job Entry Terminal
SM-ALC	Sacramento Air Logistics Center
SMITE	Emulator system
SON	Statement of Need
TBH	Tool Bearing Host
TCP	Transmission Control Protocol
TELNET	Telecommunications Network
TIP	Terminal Interface Processor
T.O.	Technical Order
TRC	Technical Repair Center
WM	Works Manager
WR-ALC	Warner Robins Air Logistics Center

1. INTRODUCTION

This paper presents data that indicates the extent of feasibility and application of the National Software Works to current and future AFLC support requirements for embedded computer systems. Additionally, a discussion of applicability of computer networking as a generalized aid to AFLC support requirements is presented.

1.1 ARPANET

The ARPANET is an operational, resource sharing, host-to-host network linking a wide variety of computers at research centers sponsored by Defense Advanced Research Projects Agency (DARPA) and other Department of Defense (DOD) and non-DOD activities. ARPANET originated as a purely experimental network in late 1969 under a research and development program to advance the state-of-the-art in computer internetting. The network was designed to provide efficient communications between dissimilar computers so that hardware, software, and data resources could be conveniently and economically shared by a wide community of users. As the network successfully attained its initial design goals, additional users were authorized access to the network. Today the ARPANET provides support to hundreds of users located virtually throughout the world with a wide variety of computer types. In 1975 it was considered appropriate to transfer the operational responsibility of ARPANET from DARPA to the Defense Communications Agency (DCA).

ARPANET is an operational, computerized, packet switching digital network which provides a capability for terminals or geographically separated host computers to communicate with each other. The host computers often differ from one another in type, speed, word length, operating system, and other characteristics. Each terminal or host computer is connected into the network through a local node computer called an Interface Message Processor (IMP) or Terminal Interface Processor (TIP). The complete network is formed by interconnecting the IMP's through communications lines (normally 50,000 bits per second capacity) supplied by common carriers.

Each node is programmed to receive and forward messages to the neighboring nodes in the network. During a typical operation a host passes a message to its node; the message is passed node to node through the network until it finally arrives at the destination IMP, which in turn passes it along to the destination host. This process normally takes less than 250 milliseconds.

Hosts communicate with each other via regular messages. A regular message may vary in length from 96 to 8159 bits, the first 96 of which are control bits called the leader. The leader is also used for sending control messages between the host and its IMP or TIP (node). The remainder of the message is the data or test.

For each regular message, the host specifies a destination, consisting of node, host, and handling type. These three parameters uniquely specify a connection between source and destination hosts. The handling type gives the connection specific characteristics, such as priority or non-priority transmission. Additional leader space has been reserved for a fourth parameter, to be used in future internetwork addressing. For each connection, messages are delivered to the destination in the same order that they were transmitted by the source.

For each regular message, the host also specifies a 12-bit identifier, the message-ID. The message-ID, together with the destination of the message, is used as the "name" of the message. The node uses this name to inform the host of the disposition of the message. Therefore, if the host refrains from reusing a particular message-ID value (to a given destination) until the node has responded about the message-ID, messages will remain uniquely identified and the host can retransmit them in the event of a failure within the network.

After receiving a regular message from a host connected to it, a node breaks the message into several packets and passes these through the network toward the destination. When all packets arrive at the destination, they are reassembled to form the original message which is passed to the destination host. The destination node returns a positive acknowledgement for receipt of the message to the source host. This readies the node to receive the next message. If the receipt response is not delivered to the originating host, the source node will detect the situation and automatically inquire of the destination node whether the original message was received until it receives a message from the destination node or "times out" after 30 to 45 seconds.

Users of ARPANET may access local or distant computers over the network. They may also exchange messages, create real time links between users, transfer files from one computer to another, and submit batch jobs.

The ARPANET is a rapidly growing network providing a service which is both cost and operationally effective. The ARPANET will expand from its current size of 66 nodes to approximately 100 nodes by 1983, when some of the subscribers will begin transferring to DOD's AUTODIN II network. Figure 1-1 indicates the variety of computers in use of ARPANET and a portion of the user locations.

While the ARPANET is quite successful, it does have some problems. The basic hardware and software are becoming obsolete. The nodes use minicomputers which were developed in the 1960's and which no longer have sufficient memory and other capabilities to support technical improvements to the network. In addition, the ultimate goal of ARPANET planning is to provide for an ARPANET II which will be a virtual network and will make use of several different physical networks (e.g., AUTODIN II, residual ARPANET, and commercial networks) to provide interconnectivity between users while still maintaining network transparency.

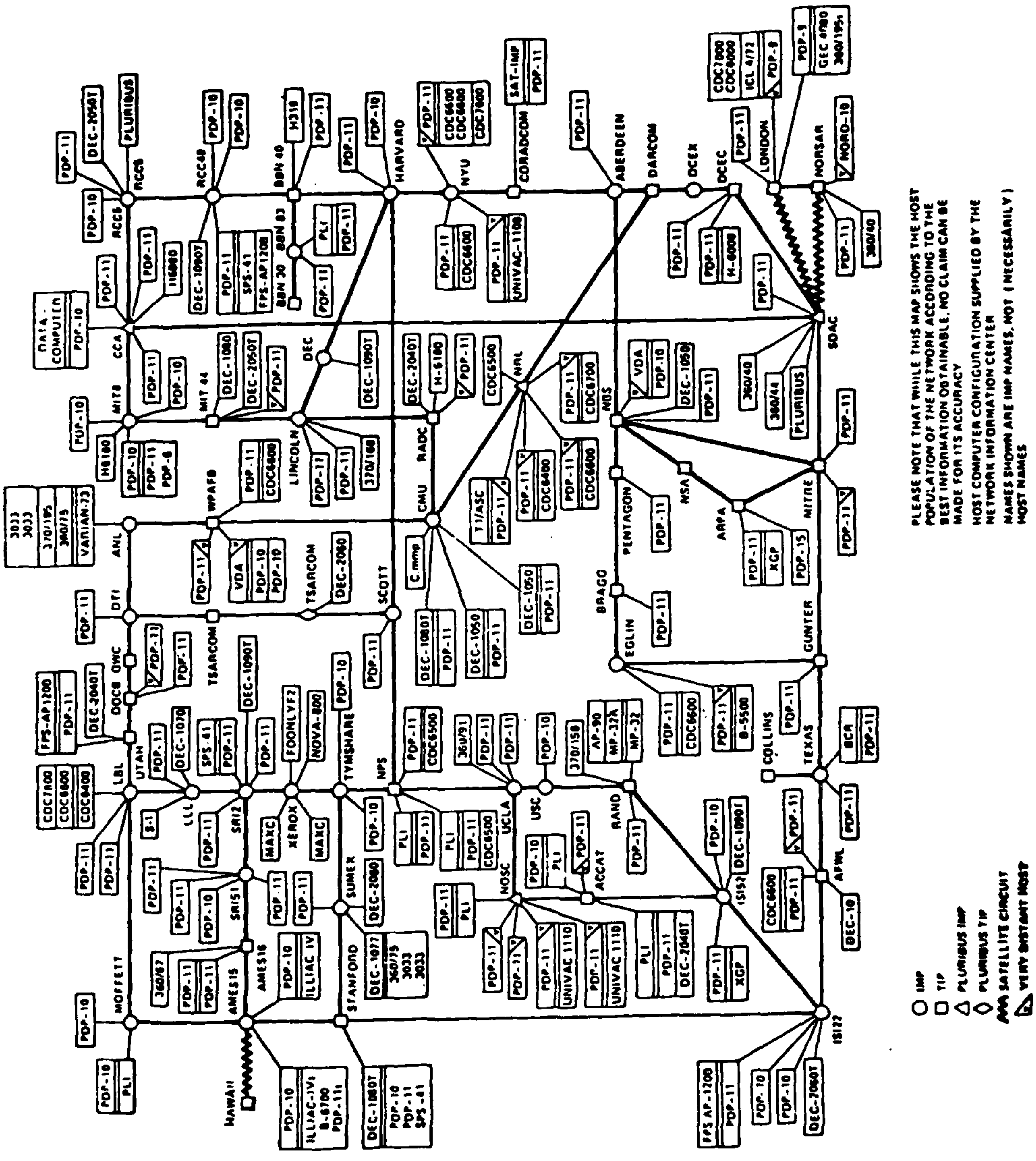


Figure 1-4. ARPANET Logical Map, June 1979

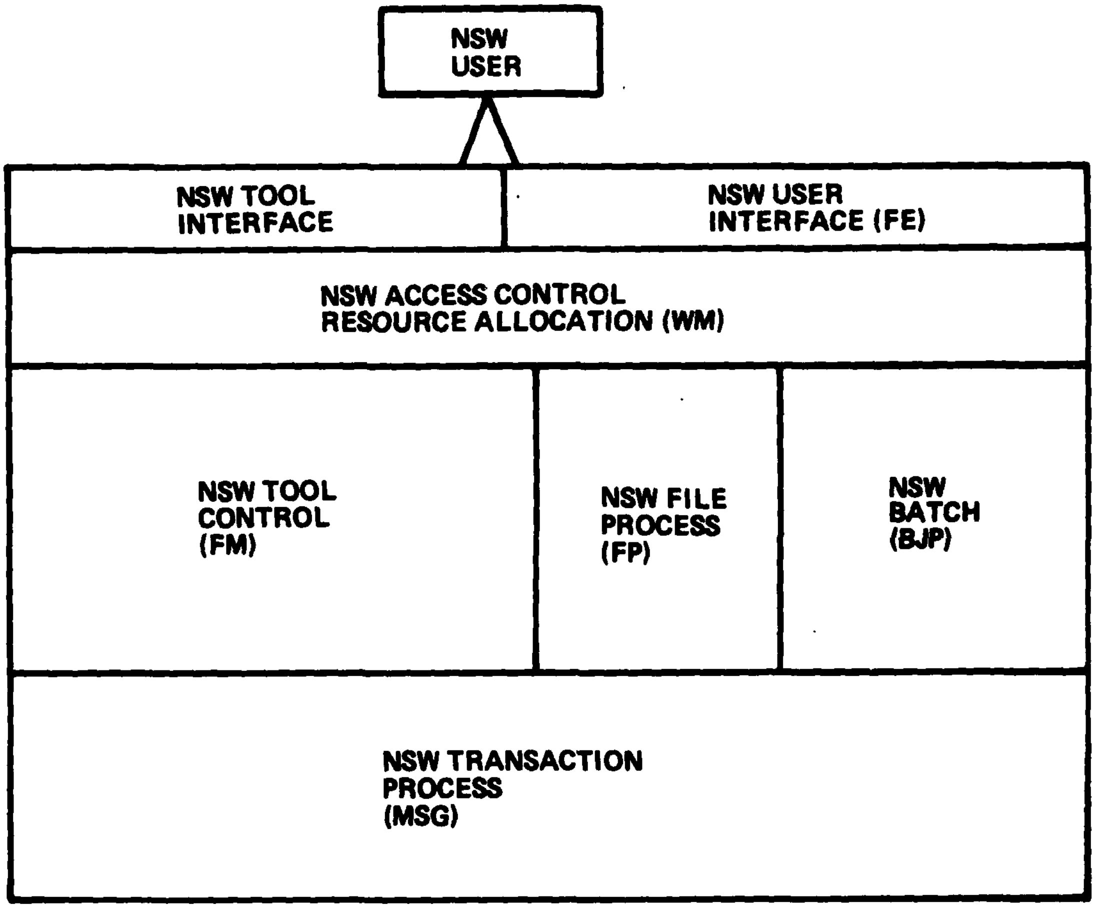
1.2 NATIONAL SOFTWARE WORKS

The diversity of hosts and operating systems associated with ARPANET is a great advantage of network access yet it presents a barrier to effective network use. It is a barrier because a user must have an account and be knowledgeable about the command language and file system on every host that he wishes to access. Recognition of this diversity barrier provided the initial impetus to develop a multi-computer operating system which would serve a set of users. This development effort was known as the National Software Works (NSW).

Need for a system such as NSW was recognized in the early 1970's and active work on NSW development began in July 1974 under the joint sponsorship of the Air Force and ARPA. An initial demonstration was conducted in November 1975 using simulated NSW components and an Initial Operating Capability (IOC) was demonstrated in July 1976 through the use of actual NSW components residing on two different computers, an IBM 360/91 and a PDP-10. Several experiments and improvements have been conducted since that time with an experimental version now available for limited use by several selected users.

NSW is an ARPANET based system designed to provide programmers access to a large range of software tools (e.g., text editors, compilers, assemblers, and debuggers) which can be used in producing software. From the standpoint of the programmer or program manager, the NSW environment consists of numerous software tools running on a variety of computer systems which are geographically and administratively distributed across the country but accessible through a single access-granting, resource allocating monitor with a single, uniform file system.

The following discussions and Figure 1-2 are offered to explain the NSW terminology and the interfacing of the various software components which comprise the total NSW package.



NOTE: NSW COMPONENTS INCLUDE FRONT END (FE), WORKS MANAGER (WM), FOREMAN (FM), AND FILE PACKAGE (FP) PROCESSES. NSW TOOLS RUN ON TOOL BEARING HOST (TBH) COMPUTERS UNDER CONTROL OF FM. THE MSG INTERPROCESS COMMUNICATION FACILITY SUPPORTS COMMUNICATION AMONG THE DISTRIBUTED SYSTEM COMPONENTS.

Figure 1-2. NSW Components

The NSW system consists of the front end through which the users access the NSW, the access granting, resource controlling central component called the works manager, the foreman modules that interface tools on the tool bearing hosts to the work manager, the file package modules which are responsible for file translation and movement, and communications protocols that specify the communications links between the various NSW components.

Each NSW user is interfaced to the system via a component known as Front End (FE). FE provides direct communication between the user and the works manager. Functions performed include establishing appropriate communications paths which enable the user to access software tools and to receive feedback on errors, progress, status, etc. Each FE is peculiar to the specific node processor and it may vary in its sophistication in proportion to the sophistication of the user tasks.

If the NSW user is also a Tool Bearing Host (TBH), then a tool interface program is necessary to allow other NSW users access to and use of the TBH tool.

The Works Manager (WM) is responsible for servicing requests for system resources and verifying the requests are valid. It accepts requests for the performance of work, arranges the initiation of that work, tracks work progress, manages file storage, and cleans up all actions including file closure and saving historical information.

Foreman (FM) is the NSW component responsible for actual accomplishment of a defined and validated work request. There is a foreman, a communication protocol (MSG), and a File Package (FP) for each individual tool bearing host. FM controls user access of tools and provides NSW resources availability to activated tools. Thus a tool gets NSW resources by making a local call on the FM, which then forwards the request to the appropriate NSW component. Effectively, the FM provides the NSW interface for the tool, and tools see through the FM to an extended local system environment.

File Package (FP) functions to create a copy of an NSW file which will be a suitable input for a tool. In other words, one tool output must be acceptable as an input to another tool. FP also facilitates creation of listings, reading and writing tapes, and other input/output functions.

In summarizing the use of NSW, a user at a computer terminal initiates his activity through the FE processor. FE loads a grammar that is used to issue commands to the WM. This permits the user and the WM to converse in a specialized language for WM functions. A communications path is established between the FE and the WM, and subsequently, the FE acts as an intermediary between the WM and the user.

The user identifies himself to the WM and specifies the desired tool and additionally may specify files for the tool to use. Subsequent to WM authentication of user identity and tool authorization the WM sends a set of tables and programs to the FE for interpretation of user commands to the tool. The WM makes any required file copies and sends them to the host bearing the tool. From then on, the user communicates with the tool via the FE and the WM recesses unless an additional file is needed by either the tool or the user.

When the user is done with the tool, any updated files and accounting information are returned to the WM which incorporates them into the NSW file system. At the tool activity termination the user can either log off or choose another tool for use.

Figure 1-3 is presented to indicate the interdependency of NSW to ARPANET and as an example of how the various NSW components fit together.

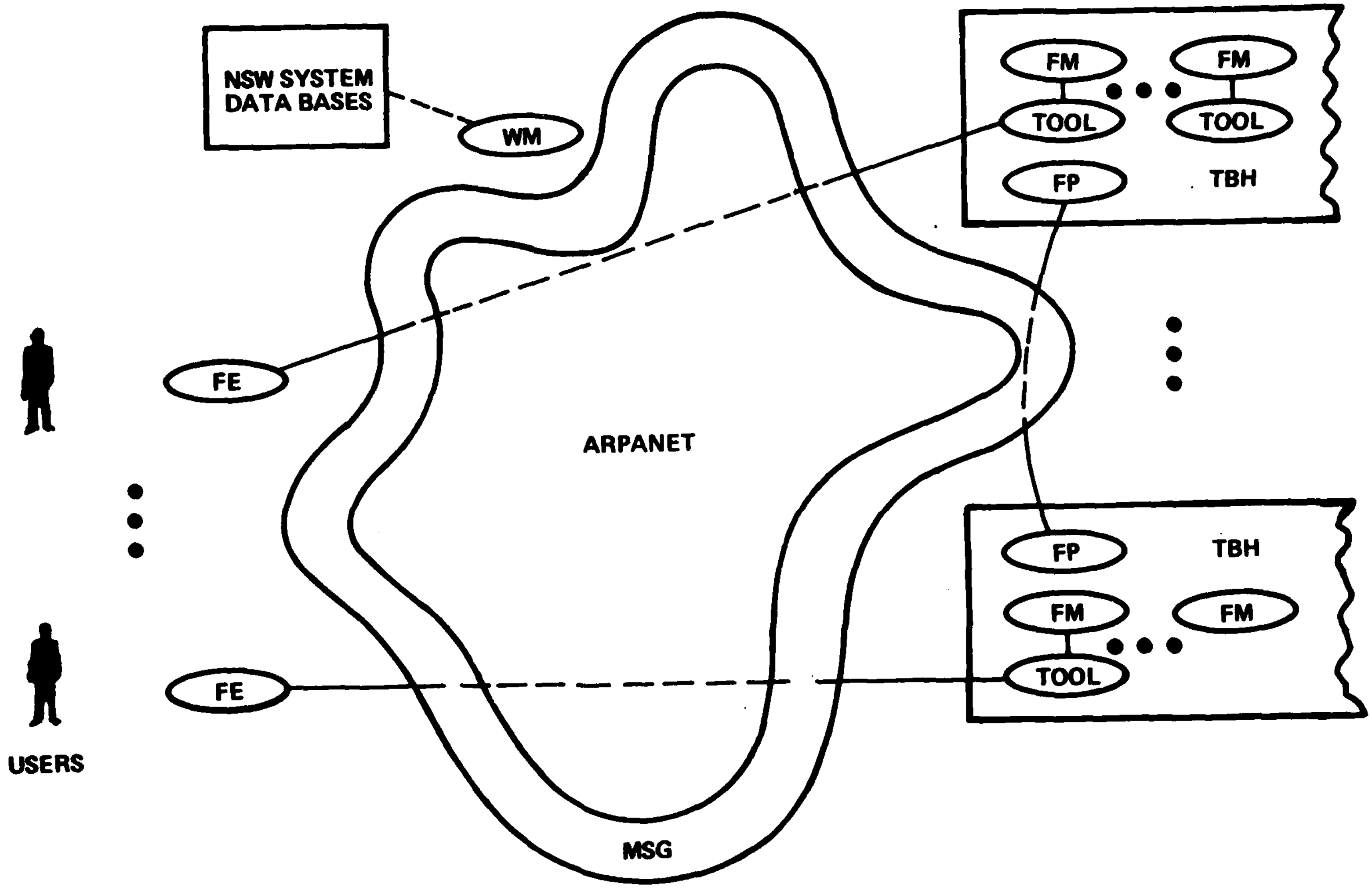


Figure 1-3. ARPANET NSW Relationship

2. NSW CAPABILITY TO MEET CURRENT AFLC REQUIREMENTS

The applicability of NSW to current AFLC software support requirements is dependent upon current NSW software tools, NSW reliability, NSW availability, NSW responsiveness, NSW system capacities, and the AFLC software support requirements. Accordingly, the current requirements are extracted from the Task 2 results of this study. Table 2-1 is a summary matrix of the AFLC requirements, the ECS categories the requirements apply to, and an applicability statement for the current NSW tool repertoire. A few statements are used for each requirement to explain the what and why of the requirement and to justify the applicability rating.

NSW software tools are currently available on only four computers among the nearly 200 ARPANET hosts. The total number of tools available is 72 and the breakout of tasks is outlined in Table 2-2. Appendix A provides amplifying information on the Version 4.1 NSW tools. Specific identificati‹ ns of the compilers and utility programs currently useable by NSW are available but generally they apply to the larger, general purpose computers used extensively for software development. Most of AFLC's requirements are centered around smaller, single-purpose computers although there are exceptions such as the UNIVAC 1108. NSW text editors and management tools apply primarily to NSW or ARPANET and thus do not provide much direct support to AFLC requirements. The following paragraphs identify the AFLC software support requirements for ECS and provide brief rationale for applicability ratings of NSW to the AFLC requirements.

2.1 ECS CHANGE

A fundamental requirement in providing support to any ECS computer program is that a capability to change the basic ECS software exists.

Table 2-1. NSW Applicability Rating to Current AFLC Requests

Source of ECS Requirement	ECS Support Requirement	ECS Category Requirement Applicability	Current NSW Applicability Rating
Task 2 Common	ECS Change	All	Minimal
	Change analysis	All except ATE	Minimal
	Engineering development and test	All	Average
	System integration and test	All except ATE	Zero
	Change documentation	All	Minimal
	Certification and distribution	All	Minimal
Task 2 Unique	Concurrency	ATE	Zero
	Rapid reprogramming	EW, OFP	Minimal
	Frequent change	EW	Minimal
SON/MENA	Classified data/processing	EW, OFP	Zero
	Off-line support	All	Minimal
	Real time simulation	EW, OFP, C-E	Zero
	Hot bench integration	EW, OFP, C-E	Zero
TRW/ SON/MENA	Intelligence data analysis	EW, OFP, C-E	Zero
	Configuration control	All	Minimal
	Automated documentation	All	Minimal
	Management information	All	Minimal
	Software repository	All	Minimal

Table 2-2. Available NSW Software Tool

Tool Type	No. Available via NSW
Category	
Compilers, Assemblers, Inter.	20
Compiler Utilities (Debuggers, Loaders, Xref, etc.)	10
File Utilities (Sorts, Comparators, Copy, etc.)	14
Mail Utilities	0†
Editors	8
Text Editor Utilities (Runoff, Renumber, Spelling, etc.)	9
Simulators, Emulators	5
Application Programs (Stat, Graphics, Math)	0
NSW and ARPANET Management Tools	6
Data Base Management System	0
Total	72

† Zeros denote tools available on ARPANET but not NSW.

That is, the ECS must be loadable, generally in source code, in another environment and/or machine from its own executing processor so a " changed" program can be placed into the executing processor. Normally change is required to correct some deficiency or malfunction with the intended purpose of improving the ECS software capability. All five ECS categories require this capability.

NSW was given a "minimal" rating because the current repertoire of tools available correlates very little with the actual tools used by the various AFLC ECS. It is understood that additional tools could be hosted into NSW to provide more applicability to AFLC tasks; however, rehosting costs would likely be incurred.

2.2 CHANGE ANALYSIS AND SPECIFICATION

This requirement applies to all ECS categories except ATE. All complex computer programs potentially can inadvertantly alter existent capabilities when the programs are changed. For example, it may be that program routine A needs to be changed, but the process of changing routine A influences capabilities in other routines. Accordingly, a support capability must exist which allows analysis of a "trial" program for its operational effects upon the ECS.

NSW was given a "minimal" rating because the current repertoire of tools available correlates very little with the actual tools used by the various AFLC ECS. Additional tools could be rehosted into NSW.

2.3 ENGINEERING DEVELOPMENT AND UNIT TEST

All ECS software changes must be developed and tested prior to incorporation into the baselined software. Development and engineering tests are accomplished in accordance with normal engineering practices. This requirement applies to all ECS categories.

NSW was given an "average" rating because the applicability of utilities programs is most appropriate for this requirement. Some tool rehosting would be required; however, NSW is primarily a development assist so its applicability is good. A maximum rating was not given because NSW does not offer immediate tool availability with high reliability as would a localized, dedicated system.

2.4 SYSTEM INTEGRATION AND TEST

This requirement applies to all ECS categories except ATE. In applicable ECS categories a changed software program may influence other avionics systems because the ECS and the significance of the outside influence, if any, must be determined.

NSW is given a "zero" rating because there are no known or expected NSW tool capabilities that apply to this requirement.

2.5 CHANGE DOCUMENTATION

This requirement applies to all ECS categories. It is the requirement to update pertinent documentation to reflect the incorporation of any changes to the ECS software into the baseline description. Master software programs are also altered to reflect the newest revisions.

NSW is given a "minimal" rating because the current NSW tools can support changing the software programs. Documentation changes can also be supported.

2.6 CERTIFICATION AND DISTRIBUTION

This requirement applies to all ECS categories. Certification is an administrative step that authenticates the changed software and documentation. Distribution is the process of getting updated documentation and software, or in some cases firmware, to the users.

NSW is given a "minimal" rating because it could apply to the distribution of software and documentation changes if the necessary recipients are ARPANET node users. Certification does not currently apply to any NSW tool or capability.

2.7 CONCURRENCY

This is a unique requirement which applies to the ATD category only. Summarily, the requirement demands that a change to an aircrew trainer be made concurrently with any change to the weapon system which the trainer represents.

NSW is given a "zero" rating because there are no known or expected NSW tool capabilities that apply to this requirement.

2.8 RAPID REPROGRAMMING

This requirement currently applies to the EW category and is expected to apply to OFP in the near future. Certain user urgencies require that software programs and/or data be changed as quickly as possible to enable new coverage or capability of the applications software.

NSW is given a "minimal" rating because there is some potential that NSW could apply to this requirement although currently there are no NSW tools which apply.

2.9 FREQUENT CHANGE

This requirement applies to EW only, but conceivably could apply to EW, OFP, and ATD in the future. Change requests in certain instances accumulate at a faster rate than the responses to the requests. Generally, the approach at meeting this requirement is to lump the requests into block changes and respond accordingly with developing and testing the block changes.

NSW is given a "minimal" rating because there is some potential that NSW could apply to this requirement although currently there are no NSW tools which apply.

2.10 CLASSIFIED DATA/PROCESSING

Both EW and OFP potentially could require processing of classified data or using classified software programs. This requirement already persists in EW.

NSW was given a "zero" rating because the NSW system contains no capability to handle classified data or software. ARPANET does interface with or can use encrypting devices although only a few ARPANET nodes actually have them.

2.11 OFF-LINE SUPPORT

This requirement is used by all ECS categories and it defined as any general processor type support. Currenty only ATE is using anything of this nature via a network approach and that is the Remote Job Entry Terminal (RJET). Because ATE support tasks are moving toward uniqueness, the utilization of RJET has diminished.

NSW was given a "minimal" rating because it could apply to this requirement although currently there are few NSW tools which apply.

2.12 REAL TIME SIMULATION

This requirement applies to EW, OFP and C-E categories of ECS. Exercise and/or test of certain ECS software and/or systems require input stimuli or signals which arrive on a real time basis. In some cases these inputs can be simulated and in other cases they must be the real thing (see Hot Bench requirement).

NSW was given a "zero" rating because there are no known or expected tools in NSW which would directly apply to this requirement.

2.13 HOT BENCH INTEGRATION

This requirement applies to EW, OFP, and C-E categories of ECS. Certain ECS tests or exercises require many of the peripheral avionics that normally interface with the ECS to supply inputs on a real time basis.

NSW was given a "zero" rating because there are no known or expected tools in NSW which would directly apply to this requirement.

2.14 INTELLIGENCE DATA ANALYSIS

This requirement applies to EW, OFP, and C-E categories of ECS. In many cases the interpretation or meaning of intelligence data is intricately involved with the ECS so that direct interface with the intelligence data source and analysis of the data is required.

NSW was given a "zero" rating because there are no known or expected tools in NSW which would directly apply to this requirement.

2.15 CONFIGURATION CONTROL

This requirement applies to all ECS categories. Configuration control of all ECS systems to include the software is required. This is a candidate for automation in the future.

NSW was given a "minimal" rating because there is some potential that NSW could apply to this requirement. There currently are some tools which minimally address configuration control.

2.16 AUTOMATED DOCUMENTATION

This requirement applies to all ECS categories, but is not yet fully implemented. Some computer software exists to enhance this capability, but no command-wide system is currently adopted.

NSW is rated "minimal" because the system could host tools to enhance documentation although current tools do not exist.

2.17 MANAGEMENT INFORMATION

This requirement applies to all ECS categories. Management control success is directly dependent upon quality information. This is a candidate for automation in the future.

NSW was given a "minimal" rating because there is some potential that NSW could apply to this requirement although no capability currently exists.

2.18 SOFTWARE REPOSITORY

This requirement applies to all ECS categories and it stems from the fact that back-up software is required for ECS software. The requirement to geographically separate the backup from the implemented software lends itself to future automation.

NSW was given a "minimal" rating because there is some potential for future NSW application. No appreciable capability currently exists.

3. ASSESSMENT OF NSW APPLICABILITY TO CURRENT AFLC ECS SOFTWARE SUPPORT REQUIREMENTS

From the discussions in Section 2, conclusions can be drawn concerning AFLC usefulness of NSW dependent upon the relative importance of the various requirements and whether or not rehosting and/or additional NSW development are worthwhile. Additional factors to be considered are NSW reliability, responsiveness, and configuration control and management.

3.1 RELIABILITY

The reliability of NSW is currently low, that is, there is not a high degree of confidence that files can be accurately generated and preserved. TRW conducted a three-year experiment in which NSW was used to assist in performing a software project. Excerpts of problems encountered and published in the final report dated September 1979 are:

- Short term unavailability due to operational problems
- Long term unavailability due to impending version change
- Inability to access existing files
- Loss of previously generated files
- Loss of batch scheduled jobs
- Inability to reenter NSW once trouble was encountered
- Unusably slow response time
- Faulty implementation of specific tools
- Cessation of NSW services while in use

Problems of these types are not trivial and current versions of NSW are under development to fix part or all of the problems. The following is from a document on network operating systems:

> The NSW approach to reliable operations has been to add reliability mechanisms to the system after it has been designed. This effort involved analyzing system weaknesses after its fundamental architecture had already been established, and then developing specific remedies to the perceived reliability problems. An important guideline in developing the reliability plan was the desire to have minimal impact on the various existing NSW components.[†]

3.2 RESPONSIVENESS

NSW normally takes minutes to respond whereas ARPANET might respond within seconds to the same request.

> For interactive users, responsiveness is a very critical performance measure. Interhost operations are noticeably less responsive than intrahost operations for almost all network and system configurations. Second, the extensive decomposition of the NSW functionality into a number of independent components each with their own private internal resources and data bases has led to a system organization which relies heavily on extensive communication between components. It is not unusual for a single NSW operation to require the cooperation of three or four individual NSW components. In an environment where communication costs or system scheduling overhead are significant, the large number of interactions required to complete an operation can prove to be a serious performance bottleneck. Even when these costs are not significant, the individual process overhead for handling messages can become significant relative to the operations being performed.[‡]

[†] Report on Network Operating Systems, RADC Document No. TR-178-117, Bolt, Beranek and Newman, May 1978, p. 150.

[‡] Ibid., pp. 155-6.

One additional consideration of responsiveness is related to the urgency of the tasks to be performed. In certain ECS software support tasks a capability must be provided as soon as possible, implying that preemption of lesser urgency tasks is expected and desired. Because NSW has no preemptive capability nor is such a capability envisioned in the near future use of NSW for these type AFLC support tasks is impractical.

3.3 NSW CONFIGURATION CONTROL AND MANAGEMENT

In the 15 April 1980 presentation to the NSW working group by IIT Research Institute, the points of inadequate documentation and non-support of NSW tools were made. This indicates the NSW baseline is poorly defined and that steps are necessary to get it to a state from which it could be configuration controlled. These steps are currently in progress by RADC and the NSW development contractors.

An NSW demonstration is scheduled to occur within AFLC over the next three-year period. Primarily, the demonstration objectives are to prove that appropriate tools for AFLC use can be hosted and implemented at AFLC locations. At least three terminal sites are being considered for participation: WR-ALC, OC-ALC, and SM-ALC. The terminal at WR-ALC has been active in use of ARPANET and NSW while the other terminals are currently used to a lesser degree. RADC will fund the major portion of expenditures for this demonstration and the RADC developing contractors will also participate. The scenarios which are candidates for the main thrust of the demonstration include

- Configuration management tool such as Engineering Data Management System (EDMS) or General Information Management (GIM)
- Emulation support environment such as SMITE
- Tool Repository predominately comprised of F-15 support tools
- Common language training environment for Ada.

The demonstration should reveal first-hand evidence to AFLC of the extent to which NSW can assist in ECS software support. Whichever scenario is chosen for implementation should include some measurement guidelines for assessing NSW tool repertoire appropriateness for AFLC use, forecasting NSW responsiveness and availability, and indicating system reliability and maintainability.

The National Software works is presently little more than a sparse set of software tools, few of which have direct relevance to AFLC embedded computer system software support responsibilities, which are available on a very small set of ARPANET hosts. During the early phases of conducting the NSW technology demonstration for AFLC applications, it was determined that the introduction of changes into the current prototypical system implementation produces such stresses that significant improvements are not expected.

The NSW environment is currently heavily dependent upon the composition of system components including PDP-11 base station processors, the UNIX operating system, the ARPANET protocols and the tool bearing host operating systems. Any attempt to amend or supplant either of these introduces prolonged system unreliability and significant costs in time and money.

The Office of the Secretary of Defense has directed that a set of DOD standard protocols be used on all Department of Defense communications networks. This directive applies to the ARPANET. The ARPANET host protocols will be replaced over the next three years with the new DOD standard protocol set. This has a direct impact on host operating systems and some applications programs that use the ARPANET. The ARPANET Network Control Program (NCP) will be replaced by two DOD protocols, the DOD standard Transmission Control Protocol (TCP) and the Internet Protocol (IP). ARPANET FTP and TELNET protocols will also be updated and standardized.

These activities, together with DOD plans for transitioning ARPANET to AUTODIN II, will dictate the investment of significant efforts in order to maintain the NSW status quo.

AFLC should be advised to view NSW with considerable skepticism with regard to both current relevance to ECS software support activities and future capability projections. The evaluations described in Section 3.1 through 3.3 support these conclusions:

- Current NSW tools offer little applicability to AFLC software support for ECS
- NSW is currently unreliable
- NSW is currently not satisfactorily responsive
- No preemptive or classified capabilities currently exist in NSW

Based on the data presented and the current AFLC software support requirements, TRW recommends no near term dependency upon NSW as a capability.

4. NSW CAPABILITY TO MEET FUTURE AFLC REQUIREMENTS

Future AFLC ECS software support requirements are predicted to remain substantially the same as they currently exist. The tools and methods which provide capability to meet these requirements are where most future growth will occur. Some of these improved tools and methods will place additional burdens upon interconnecting communications links and processors. The support concept to be used by AFLC in meeting these requirements will influence the tools to be used and the methods of data interchange and control to which NSW could apply.

In order to define a technical basis for evaluating the NSW applicability to AFLC ECS software support requirements, it was necessary to postulate an operational concept. The postulated concept to be adopted by AFLC will use a centralized control node that is interconnected to implementation nodes. Most likely the control node will be located at AFLC Headquarters with software support provided by one or more alternate locations. Implementation nodes will be located at the Air Logistics Centers and the Aerospace Ground Metrology Center (AGMC). Restated, this concept will use a network to interconnect AFLC to the ALC's and AGMC, and each node will have its own localized network that interconnects nodal agencies. Functions assigned to both the control and implementation nodes will help define the tools and methods by which requirements are met at that node. (For example, software development and test at an ALC may be enhanced by using an interactive graphics system). Common requirements will apply to all nodes although diverse support activities and unique systems will demand some unique support at each node.

Control node functions such as standardization, interoperability, policy, network control, training, and management can be accomplished by meeting the demands of an automated configuration control system, an automated documentation system, an automated management information system, a software repository, and a training system. Each of these demands is discussed and NSW capability ratings are applied. Table 4-1 summarizes these ratings.

4.1 CONFIGURATION CONTROL

Configuration control is necessary at different management levels within the command. It is necessary at the control node to help ensure known baselines, software commonality where practical between the implementation nodes, interoperability, and to facilitate documentation and training. Several attempts at automated systems have been attempted by various AFLC agencies without any large degree of success. The type of configuration control system needed should require simple data inputs that do not overburden the data initiators and data maintainers. Such a system will need to be developed.

NSW is given a "minimal" rating because tools could be hosted to aid the configuration control requirement. Most NSW existent configuration control tools have a limited or specific application and none are of the magnitude and capability required for the control node tasks.

4.2 DOCUMENTATION

Documentation is expected to remain a major concern of AFLC for the next several years. Like configuration control, it requires large data inputs and impacts work activities of engineering developments. The postulated AFLC concept will use an automated system within the next few years for both engineering data and technical orders. Accordingly, the attendant data must be exchanged between the control and implementation nodes and perhaps even distributed directly to the users.

NSW is given a "minimal" rating because there is some applicability of envisioned tools and the communications network. ARPANET is more attractive because it is the communications media itself and

Table 4-1. NSW Applicability Rating to Future AFLC Requirements

ECS Support Requirement	ECS Category Requirement Applicability	Current NSW Applicability Rating
Automatic Configuration Control System	All	Minimal
Automatic Documentation System	All	Minimal
Automatic Management Information System	All	Minimal
Software Repository	All	Average
Training System	All	Minimal

does not contain the larger software overhead associated with NSW. Network support of an automated documentation system is more of a capacity nature than sophisticated processing so the network software required will be simplified programs.

4.3 MANAGEMENT

Management information is normally provided to report status or to provide a basis for management actions. AFLC will adopt an automated system to gather this information and the tie-in to the implementation nodes will likely use some network capability.

NSW is given a "minimal" rating because there is some application of envisioned tools. Network support of management information is more of a capacity nature than sophisticated processing capability.

4.4 SOFTWARE REPOSITORY

Air Force regulations and good development practices require that developed software programs be copied and the copies stored at an alternate geographical location than where the software is developed. For example, software developed for the F-16 may be developed at Hill AFB, Utah, but a copy could be stored at Wright-Patterson AFB, Ohio. The postulated AFLC concept will adopt a system that will copy software developed at the implementation nodes and store it at one or more alternate locations. The system will be accessible to restore or replace software if needed by the implementation nodes. It likely will be used in conjunction with the configuration control system.

NSW is given an "average" rating because tools are available to copy software and transfer it to an alternate location. Assuming that NSW reliability is improved in the near future, NSW could well apply to this requirement.

4.5 TRAINING

Training of AFLC personnel in logistics and technical matters is going to expand within the next few years. In many cases Air Training Command expertise in these areas is not adequate to compete with the demonstrated success of AFLC in-house training experienced through using its own personnel. Certain training programs can substantially be improved and conducted by using a network to enable television based instruction. AFLC will adopt such a system within the next few years.

NSW was given a "minimal" rating because the potential for incorporating training programs is not very high. The data capacity of ARPANET is not sufficient to meet the demands of, for example, a color television link. NSW software places an additional burden on the communications capacity so the real NSW applicability is also a function of the training sophistication required. In other words, NSW could support a low data rate, simple training system but could provide little aid to a high data rate, complex training system.

4.6 FUTURE RESPONSIVENESS TO STATE OF OPERATIONAL NEED

The overall applicability of NSW to future use by AFLC is dependent upon the expected improvements of NSW in the next years. This study is aimed at a support capability for AFLC to use by approximately 1985. As it currently exists, NSW has too many operational problems for a critical mission like AFLC's to rely on. There is little question that if enough resources are dedicated to NSW improvement in the next four or five years the degree of applicability to AFLC use will improve. If such problems as responsiveness and reliability are corrected, AFLC could depend upon NSW as a future capability. It should be noted, however, that despite this assessment of the applicability of NSW to AFLC software support requirements, the lessons learned from the NSW development reveal informative conclusions beneficial to AFLC and other Air Force agencies.

- The concepts of geographically distributed software tools and a software repository have been proven as sound concepts.

- Development of network software is a complex task implying that existent networks should be considered in lieu of any new network development.

- It is more efficient to network homogeneous computers together than heterogeneous computers.
- Additional software tools can be hosted into the NSW tools library.

A Statement of Operational Need (SON) has been prepared by AFLC within the past several months. Although still in the coordination cycle, the SON reflects the thinking of the AFLC technical personnel as to the kinds of systems needed in the next 10 or so years. Extractions from the SON will help to "fill in" the implementation portions of the Long Range Plan Outline. Realizing this, the SON requirements were considered in establishing the control node/implementation nodes concept. NSW has more application potential in the control node arena than to the implementation nodes. A network will be used with each implementation node; however, each network will be structured to enable a common core processor with a standard interface to the control node network. In view of constraints such as security, current and projected AFLC resources, and interoperability, NSW does not offer as much applicability as *AUTODIN II*.

5. APPLICABILITY OF NETWORKING TO AFLC SOFTWARE SUPPORT

Many aircraft and avionics systems use several interconnected processors in a network configuration. The trend is toward more such configurations. Indications for the future reveal increased use of real time communications between multiple processors.

AFLC support problems also are becoming more interdependent as indicated by AISF concepts and implementations. Networks will help to answer the increased interdependency problems. There is little doubt that improved networks will find usefulness in AFLC, but the unanswered questions are how and to what extent.

There are many support tasks and activities that currently apply to ECS software support. Most will similarly apply to future support as well. The emphasis must be made that networking primarily applies to those tasks and activities which require near real time data exchange and/processing. In other words, non real time requirements can likely be done via alternate, more cost effective media such as the U.S. mail, aircraft courier, etc. For example, a copy of a software program, if needed at an alternate location, can be mailed if the urgency for its receipt is not immediate or if no interactive dialogue between the two locations is required. This section relates networking to those activities that require frequent or constant data exchange, tight management control, or maximum usefulness of "multilocation common" software modules. Additionally, the needs of supporting software are the only needs considered in the discussion; however, AFLC has need for support of other activities such as finance, budgeting, personnel, supply, resource utilization, etc.

The May 1980 Statement of Need/Mission Element Need Analysis (SON/MENA) indicates the projected future needs for supporting ECS software in the late 1980's time frame. This document is currently in a coordination cycle but has been examined by TRW. Coupling data from the SON/MENA with corporate inputs has produced a statement of future functional requirements as outlined in Tables 5-1 and 5-2. Table 5-1 indicates control functions which could be applied by Hq. AFLC. Table 5-2 indicates the future implementation functions which could be accomplished by the five ALC's and AGMC. The task requirements, if satisfied, would provide satisfactory future ALC capability to support ECS from a hardware and software perspective.

5.1 POSTULATED NETWORKS

Two types of networks are postulated for AFLC's future support of ECS software and as an example of how to meet the SON/MENA based requirements listed in Tables 5-1 and 5-2. They are: a control network which interconnects the headquarters with the ALC's and AGMC and implementation networks (or local networks) which link the inner activities at an ALC. In accordance with this network approach the support functions were allocated to either the control or implementation networks. Thus, the control functions would be the responsibility of the main control node (Headquarters AFLC) and the implementation functions would be assigned to the ALC's and AGMC. Each local or implementation network contains a common core which links directly to the control network. The common core is configuration controlled by the control node and it acts as the master controller for its own implementation network.

Each implementation node (ALC or AGMC) should have the capability to accomplish generalized functions as listed in the implementation functions column of Table 5-2. Additionally, each node would need a capability to accomplish any unique tasks that will arise as a result of weapon system or task assignments. An example would be unique radar tests for the common multimode radar which will be a WR-ALC responsibility. Figure 5-1 indicates some of the systems assigned to each ALC which could require unique ECS software support for that particular ALC.

Table 5-1. AFLC Control Function Requirements

Control Functions	Task Requirements
Policy	Regulations
	Procedures
	Type information reported
Standardization	Network interfaces
	Local network common core
	Modular software
	Data buses
Interoperability	Protocol
	Languages
	Timing or sequence considerations
Management Information	Data reporting frequency
	Automated management information system
	Automated documentation
Network	Executive software
	Data base control/access
	Security and classified processing
Software	Executive software
	Data base management system
	Automated configuration control system
	Data base security
	Software masters control
	Software repository
	Reproduction system (software)
Training	Master training programs
	Training resources control

Table 5-2. ALC Implementation Function Requirements

Implementation Functions	Task Requirements
Software Development and Maintenance	Off-line processing Dynamic simulation Interactive graphics Testing Software tools Hot bench Support software Input stimuli Emulation
Documentation Update and Distribution	Automated documentation (T.O. and engineer data) Document distribution
Equipment and Facilities Maintenance	TRC and individual system maintenance Maintenance contracts
IV&V	IV&V control system Algorithm and logic tools Module analysis Performance evaluation and analysis Automated test tools
Configuration Management	Local master programs Deficiencies control Prototype development Firmware verifier
Management Information	Periodic data reporting Maintain local status Highlight exceptional or noteworthy problems/status
Training Implementation	Network training On-the-job training Formal training programs
Local Network	Control network compatible Maximize N/W and S/W commonality Modular software, transparent to users

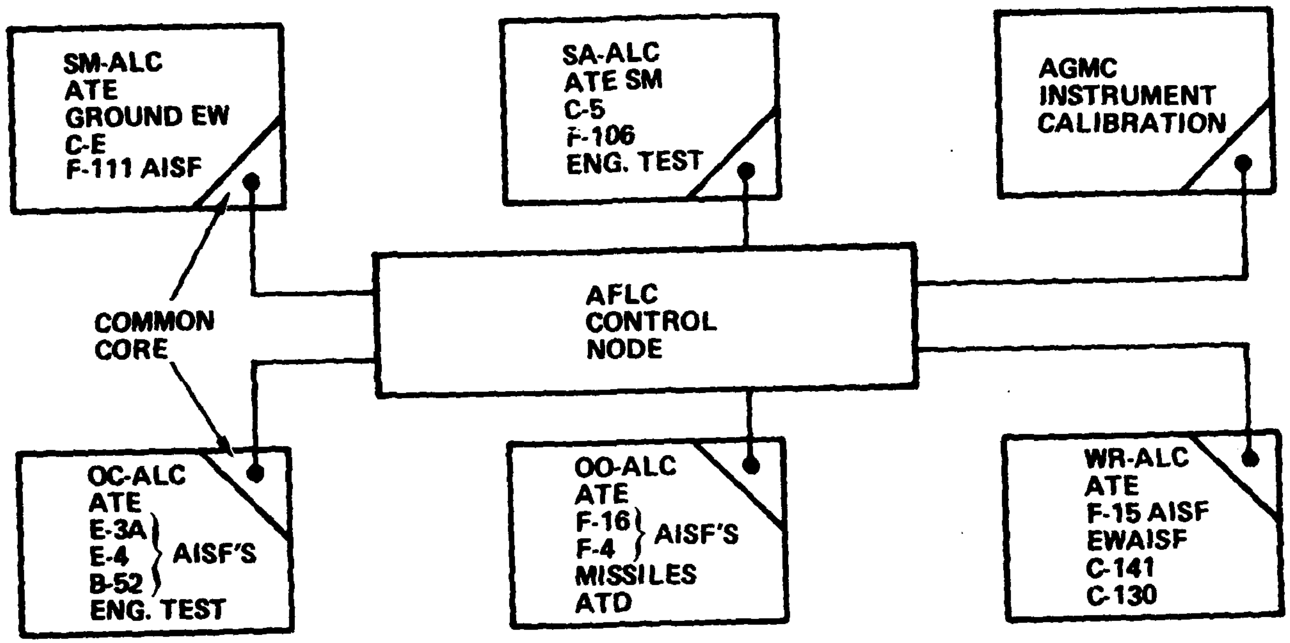


Figure 5-1. ALC Systems Requiring Unique ECS Software Support

5.2 *ASSESSMENT*

The aforementioned example illustrated a method and the extent of networking that could be used to provide software support to AFLC ECS. There are other examples such as all ALC's completely interconnected, more than one control node, reallocation of functions, etc. Whatever the chosen approach, the costs associated with a network are mainly a function of the type and capacity of the communications media, hardware and software acquisitions, and the total task complexity and objectives. Dollar costs for communications media are discussed in Section 2.4 of the Technology Forecast, Volume VIII of this study. Specific costs are determinable when a design for implementation is accomplished to a level of detail that indicates data rate capacities, type terminal equipment, and the associated software programs. Software development and acquisition costs represent the major cost item for the example of network usage already addressed in this paper.

Both ARPANET and AUTODIN II could be candidate systems for the control network mentioned above. Neither system, in its current state, is applicable to the implementation networks. A suitable network would probably need to be developed. The technical risk associated with such a development does not appear to be high although proceeding with the development should not be done until a thorough requirements analysis has been completed.

Two general approaches that AFLC could take in acquiring a network capability are: (1) to take an existing network and see what requirements can be satisfied by its use; (2) to analyze the requirements to see what capability a network must have and then acquire the network. The former approach is parallel to the consideration of NSW to accomplish AFLC tasks. Because NSW was designed for another use (development assistance), it has little application to AFLC tasks nor will it have unless very significant changes are made to it.

Certain constraints also affect the cost and risk of acquiring a network. Such factors as compressed schedules, low budget funding, sophisticated data encryption, etc. have a tendency to drive cost and risk upward. Conversely, maximizing the utility of current capabilities may drive costs downward.

6. SUMMARY

In the near term NSW offers little potential for improving AFLC's ECS software support capability. Far term potential is also quite limited unless some major breakthrough in reliability and responsiveness is realized with the NSW system. Networking, in general, offers substantial potential for improving AFLC's software support posture. The question of the cost effectiveness of tailoring an existent network for future AFLC use versus development of an "AFLC unique" network is unanswered at this point. An answer is obtainable only after an analysis of a specific AFLC operational concept using networks is conducted to determine the cost, schedule, and risk areas.

APPENDIX A

CURRENT NSW TOOLS

In NSW Version 4.1 the name of each batch and interactive tool explicitly identifies the host which houses the tool. The NSW tool name is composed of the generic name of the tool and the following host identifier suffixes:

2 Character Host Identifier	ARPANET Host Name
IC	USC - ISIC
IE	USC - ISIE
RM	RADC - MULTICS
UC	UCLA - CCN

Table A-1 illustrates the type of tool environment that could be assembled from just those tools which have been successfully used in the current NSW release. Tools are grouped by type and arranged to illustrate the NSW hosts on which a tool is installed. For example, the tool XED is installed on two NSW hosts, ISIE and ISIC, under the names XED-IE and XED-IC respectively. The capabilities of some of these tools are exemplified in more detail by the sample tool documentation appended after the list.

Documentation for NSW tools is available at various levels of specificity. At the most general level, an abstract of the tool is available from within NSW. For many interactive tools, once a tool session has begun detailed help is available within the tools, usually in response to a question mark. Reference manuals and beginner's guides are also usually available either on-line at an ARPANET or NSW host, or in hard copy from the tool purveyor. Change bulletins

Table A-1. NSW Tools

Type of Tool	Generic Name	Host Specific Name			
		USC-ISIE	USC-ISIC	RADC-MULTICS	UCLA-CCN
Editor	DISPLAY				
	SOS	SOS-IE	SOS-IC		DISPLAY-UC
	TECO	TECO-IE	TECO-IC		
	QEDX			QEDX-RM	
	XED	XED-IE	XED-IC		
Formatters	MRUNOFF	MRUNOFF-IE	MRUNOFF-IC		
	RUNOFF			RUNOFF-RM	
Spelling Aids	MSPELL			SPELL-RM	
	SPELL	SPELL-IE	SPELL-IC		
Compilers/ Linkers	BASIC			BASIC-RM	
	BCPL	BCPL-IE	BCPL-IC		
	FORTRAN			FORTRAN-RM	FORTRAN-UC
	PL1			PL1-RM	PL1-UC
Debuggers	BDDT	BDDT-IE	BDDT-IC		
System Information Tools	DESCRIBE	DESCRIBE-IE	DESCRIBE-IC		
	HELP			HELP-RM	
	HOSTAT	HOSTAT-IE	HOSTAT-IC		
	NETSTAT	NETSTAT-IE	NETSTAT-IC		

supplement these manuals by describing the impact of new releases of tools. NSW operations publishes "User Impact Bulletins" that describe tool behavior peculiar to NSW implementation of the tool. The following examples of NSW documentation are appended:

- NSW Abstract for BCPL
- NSW Abstract for BDDT
- NSW Abstract for DESCRIBE
- NSW Abstract for HOSTAT
- NSW Abstract for NETSTAT

A.1 NSW ABSTRACT FOR BCPL

The BCPL compiler is available as an NSW tool with no known restrictions. However, potential users are advised to observe the following conventions:

- Observe all upper and lower case distinctions in file names. The NSW file package makes this distinction.
- If you default a file name extension (i.e., to .BCP), the BCPL compiler will always insert upper case.
- Given a compilable file with the name A.B.C, the BCPL compiler will produce two files A.S and A.REL. However, the BCPL debugger BDDT cannot access (Symbol) files with names longer than 5 characters. Users are therefore advised to use file names of 5 characters or less.
- Currently, BCPL has access only to local host directory <bcpl> in addition to the NSW filespace. Do not allow programs to "get" files from other areas.

For assistance, the user may type ? to the BCPL compiler. Full documentation is available in the Tenex BCPL manual.

A.2 NSW ABSTRACT FOR BDDT

BDDT is a high-level debugger for use with compiled BCPL programs. For a given program, it requires a saved core image, a symbol file (generally with the extension .S) and the original source file. Be aware that BDDT cannot currently access symbol files with names longer that 5 characters. Thus, the file name TEST9 is acceptable while the name TEST1Ø is not. There are no known limitations on file extensions. We hope to raise this (5 character) limit in a future release.

Generally, the user may type ? at any point and BDDT will list the options available. Further documentation is available in the Tenex BCPL manual.

A.3 NSW ABSTRACT FOR HOSTAT

HOSTAT is a survey program that gives status information for all hosts on the ARPANET. Execution of the program as well as termination are automatic.

A.4 NSW ABSTRACT FOR NETSTAT

NETSTAT is a survey program giving different kinds of status information concerning the ARPANET. Documentation is available by typing ? at the top level. In some cases, termination is automatic. Otherwise, the user may terminate the program manually by typing QUIT.

A.5 NSW ABSTRACT FOR MRUNOFF

MRUNOFF is a document compiler available on Tenex systems. A document compiler combines the functions of text formatter with capabilities to prepare documents for specialized output devices. The program will ask for an input file containing the document to be compiled. An output file is given by typing "0(esc)" followed by the file name. Execution of the program is started by typing just a carriage return (i.e., blank line). Termination is automatic. Additional documentation is available on <documentation> MRUNOFF.DOC at [BBN].

BIBLIOGRAPHY

1. ARPANET Director, DCA 200-C-641, December 1979.

2. Digest of Papers, "Micros, Minis, and Maxis" IEEE COMPCON' 77 Section 20 (NSW), September 1977.

3. Final Report, RADC Contract No. F30602-78-C-0155, September 1979.

4. Final Technical Report, RADC Document No. TR-78-23, "NSW Feasibility Study", February 1978.

5. National Software Works, Status Report No. 1, Massachusetts Computer Associates, Contract No. F30602-76-C-0094, 30 June 1977.

6. NSW Demo Plan, Palmer Craig, WR-ALC, 1979.

7. NSW Tool Status and NSW Tool Manager Functions, IIT Research Institute, 15 April 1980.

8. NSW User's Reference Manual System Release, Section 4, Massachusetts Computer Associates, Inc., 1 June 1979.

9. Report on Network Operating Systems, RADC Document No. TR-178-117, Bolt, Beranek and Newman, May 1978.

FILMED

7-8