AD-A101 107 NAVAL POSTGRADUATE SCHOOL MONTEREY CA F/G 17/2
COMMAND CENTER NETWORK: SCOPE AND APPLICATIONS.(U)
MAR 81 G K HAMILTON
UNCLASSIFIED NL

1 OF 1
AD
A101107

END
DATE
FILMED
7-81
DTIC

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

COMMAND CENTER NETWORK:
SCOPE AND APPLICATIONS

by

George Kendall Hamilton, Jr.

March 1981

Thesis Advisor: F. R. Richards

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE — READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER

2. GOVT ACCESSION NO. AD-A101 107

3. RECIPIENT'S CATALOG NUMBER

4. TITLE (and Subtitle)
Command Center Network: Scope and Applications

5. TYPE OF REPORT & PERIOD COVERED
Master's Thesis; March 1981

6. PERFORMING ORG. REPORT NUMBER

7. AUTHOR(s)
George Kendall Hamilton, Jr.

8. CONTRACT OR GRANT NUMBER(s)

9. PERFORMING ORGANIZATION NAME AND ADDRESS
Naval Postgraduate School
Monterey, California 93940

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS

11. CONTROLLING OFFICE NAME AND ADDRESS
Naval Postgraduate School
Monterey, California 93940

12. REPORT DATE
March 1981

13. NUMBER OF PAGES
81

14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)
Naval Postgraduate School
Monterey, California 93940

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

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)
Command Center Network, Navy Command and Control Subsystems, Information Subsystems, Transmission Control Protocols, Modularity, Interoperability, Internetworking, Connectivity, Flexibility, Versatility, Tactical Flag Command Center.

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)
The Command Center Network (CCN) is a computer network designed to interconnect a diverse group of heterogeneous shipboard information and Command and Control (C2) subsystems. This local network will utilize a single, high-speed data bus installed on the individual platform. As this network is envisioned, such subsystems as NTDS, NAVMACS, CCIS, SSES, TSA, CV-TSC, and CV-IC will be interconnected in order to correlate information to provide the best possible decision base for the commander. The Tactical Flag Command Center (TFCC) concept, which the CCN is essentially designed to support, is considered by the Navy as the nerve center of future Command and Control. The CCN is envisioned to be the backbone of the TFCC. This thesis examines the system development of the CCN.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

Approved for public release; distribution unlimited

Command Center Network: Scope and Applications

by

George Kendall Hamilton, Jr.
Lieutenant, United States Navy
B.A., Case Western Reserve University, 1974

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN SYSTEMS TECHNOLOGY (C3)

from the

NAVAL POSTGRADUATE SCHOOL
March 1981

Author: George K. Hamilton, Jr.

Approved by: F. L. Richards, Thesis Advisor

G. T. Howard, Second Reader

John M. Wozencraft, Chairman, C3 Academic Group

Academic Dean

ABSTRACT

The Command Center Network (CCN) is a computer network designed to interconnect a diverse group of heterogeneous shipboard information and Command and Control (C2) subsystems. This local network will utilize a single, high-speed data bus installed on the individual platform. As this network is envisioned, such subsystems as NTDS, NAVMACS, CCIS, SSES, TSA, CV-TSC, and CV-IC will be interconnected in order to correlate information to provide the best possible decision base for the commander. The Tactical Flag Command Center (TFCC) concept, which the CCN is essentially designed to support, is considered by the Navy as the nerve center of future Command and Control. The CCN is envisioned to be the backbone of the TFCC. This thesis examines the system development of the CCN.

TABLE OF CONTENTS

LIST OF FIGURES

ACKNOWLEDGEMENTS

I would like to extend my sincere appreciation to Dr. Glen Allgaier and Mr. Merle Neer of the CCN project at the Naval Ocean Systems Center, San Diego, California, for their cooperation and assistance in gathering the material and providing their valuable time. Furthermore, Associate Professors F. R. Richards and G. T. Howard have lent their professional guidance and direction to this effort. Lastly, and most importantly, I would like to thank my wife and son for their constant support, understanding, and encouragement.

I. INTRODUCTION

Command and Control (C2) may be defined as an iterative process of resource allocation in which a recognized point of authority coordinates human and machine generated information, as well as human perceptions, in order to perform missions and validate results. Three key points of the current Department of Defense position regarding Command and Control are:

1) Command, control and communications (C3) are functions performed through an arrangement of personnel, equipment, facilities, and procedures which are employed in planning, coordinating, and controlling the operational activity of military forces.

2) C3 system elements include facilities, warning systems, communications, data collection and processing systems, and procedures.

3) The systems approach to the development and improvement of C3 capabilities is essential. Subsystems need to be interoperable and mutually supporting.

In considering these key points of command and control, the Navy perceives that one of the most critical elements of future C2 will be the ability of the Navy Task Force Commander to maximize his use of available information, and ensure that systems respond to his decisions. Connectivity, interoperability, flexibility, and versatility among existing and contemplated information and C2 subsystems are major goals of future developments.

A. SYSTEMS TO SUPPORT A C2 GOAL

As the tactical arena expands with technological advances in weaponry, such as Harpoon and the Cruise Missile, as well as contemplated developments in the area of Charged Particle Beam Weapons (CPBW), correlation, dissemination, and display of an accurate picture of the tactical environment is absolutely essential. As weapon development advances from considerations of ranges of 30 to 60 miles with weapons travelling the speed of sound, to ranges of hundreds of miles with weapons travelling the speed of light, the need for a coordinated tactical picture grows exponentially. Therefore, the Navy has developed the concept of the Tactical Flag Command Center (TFCC). The TFCC will replace the tactical commander's "flag plot" area. In the past, an embarked commander might have a designated area from which to assess the tactical situation, but the platform sensors immediately available to that area were limited to what can be accessed through an NTDS console. In the TFCC concept, the embarked commander would have access to all shipboard C2 and information subsystems through the use of two consoles. One of these would be used for query of various data bases and displays of charts, graphs, and other pertinent information. The other console would be utilized for an up-to-date tactical picture similar to that which is available through NTDS. This TFCC is envisioned as the nerve center of the Task Force Commander's ability to exercise command and control of forces, weapons, and other assigned assets.

However, in order to correlate all the incoming data, provide an historical data base, display the tactical picture, control forces and weaponry, as well as providing general C2 functions such as reliability, flexibility, versatility, and security, the TFCC nerve center requires a system of transmission, correlation, and interchange of inputs. In a manner similar to that which exists in the human nervous system, a C2 network is generally agreed upon as the best method to support the TFCC.

Therefore, in 1978, the Advanced Command Control Architectural Testbed (ACCAT) at the Naval Ocean Systems Center in San Diego, California, initiated the Local Command Center Network (LCCN) project. In essence, this project was designed to serve as the transmission connector and correlator for the Tactical Flag Command Center utilizing existing technology. In late 1979, the project name was changed to the Command Center Network (CCN), but the goal to serve as the backbone for the TFCC nerve center remained unchanged.

Initially, this program was designed to fulfill the goal of interconnecting C2 and Navy Information subsystems to support the TFCC and produce a working prototype at ACCAT in twenty months. Today, over three years since project formulation, a working prototype at the ACCAT is still in the developmental stage. Meanwhile, the TFCC concept has been employed on the carrier USS Midway and is being implemented on the carrier USS America.

B. SYSTEM CASE STUDY AND QUERIES

The balance of this thesis is designed to outline the complexity of the undertaking that evolved over the course of CCN development. As one progresses through this presentation of the development of CCN, the following questions are addressed:

1) Can worthwhile programs with a mutually agreed upon goal, lose sight of the need necessitating project development?

2) Is the Command Center Network project being developed to respond to the perceived needs of the TFCC?

3) Is complexity of the CCN system and technology driving this program, or are the system requirements the primary focus of the endeavor?

In the succeeding chapters, particular developmental aspects of the CCN project will be explored. Major questions like interoperability, internetworking, and transmission control protocols will be discussed to demonstrate the complexity of the project and to determine if the project addresses these issues in a manner that lends the intended support to the TFCC. In the next to last chapter, the views of the project manager regarding future applications of the network are given. These views suggest that the CCN project development may have wandered away from the system requirements imposed by the TFCC toward the development of new technologies and/or nice-to-have accessories.

As this presentation progresses, one should remain aware of the original goal of this project and the three major questions mentioned earlier in this discussion. It is the author's belief that the CCN project has succumbed to technological temptations. In essence, the light at the end of the tunnel may no longer be interconnecting C2 and Navy Information subsystems in support of the TFCC. The light at the end of the tunnel may have become a runaway train of technological advances. The attempts to make the system climb aboard this train may have created a situation where interconnecting and supporting the TFCC can be fulfilled without the CCN.

II. INTRODUCTION TO CCN

A. BACKGROUND

The multi-platform commander is currently faced with the dilemma of an array of diverse, uncoordinated information systems designed to assist him in the performance of Command and Control functions. Therefore, the Navy has initiated the Command Center Network (CCN) project to address the major issues of coordinating and providing user access to a number of diverse subsystems which contain information of interest to the commander.

B. NAVY C2 AND LOCAL NETWORKS

Previous efforts to integrate subsystems have been impeded by the specificity of equipment development. Each subsystem can be characterized by unique protocols and interfaces, fully committed memories and central processing units, complex, expensive, system-specific software and intelligent human operators specifically trained to perform on the individual equipment.

The goals of the Command Center Network are based on fulfilling three broad criteria:

1) Improvements to Navy command and control must be evolutionary. That is, the present baseline of subsystems making up the Navy C2 system can't be wholly replaced at a given point in time; this is neither affordable nor is it practicable with respect to overhaul cycles and modernization processes with operating platforms and full-time operational use of shore-based command centers;

2) Command and Control improvements, in order to be evolutionary and useful, must be compatible not only with existing systems, but also with expected and/or projected future installations; and

3) Command and Control information needs must be satisfied first from existing subsystem/sources (i.e. the concept of a baseline of system operability). [Ref. 1]

These three criteria seem to be consistent with the system goal to support the TFCC. However, some debate as to whether or not a given system can reasonably anticipate all future developments may arise. It is important to note that a real danger exists if the CCN becomes too conscious of future developments in C2 and Information subsystems instead of concentrating on interconnecting existing subsystems.

C. ISSUES

Upon embarking on a discussion of the major issues the Command Center Network is designed to address, it is important to comprehend the context and scope of Command and Control. Command and Control requires informed decisions by a commander of forces in order to carry out assigned tasks. In that context, information becomes the essential element required, operated on, and disseminated by the generic functions of a command and control center (e.g. TFCC). The generic and interacting C2 nodal functions are:

1) Tasking input/correlation;
2) Information input/correlation;
3) Situation assessment and decision making;
4) Report generation/output; and
5) Directive generation/output.

Furthermore, information input consists of numerous dynamic elements from certain broad classes. These classes or categories of information input include:

1) own forces information (capabilities, readiness, position, etc.)

2) threat forces information (capabilities, readiness, position, intentions, philosophy, etc.)

3) environmental information (weather, sea state, visibility, propagation, bottom/beach/terrain characteristics, etc.), and

4) politico-military intelligence to include postulations regarding force intentions based on assessment of activities.

Within the context mentioned above, the ACCAT is developing the Command Center Network (CCN) to address the following issues:

a. Can available local network technology, in combination with gateways (links) giving access to long-haul networks, facilitate the efficient and effective interconnecting of various existing and newly developed automated fleet subsystems so that information needed for command and control can be obtained from them within a given ship platform or from among several platforms and/or shore-based systems?

b. From among the various configurations of local networks now known, is there one particular configuration or architecture that is significantly "best" for the intended application?

c. Will application of local network technology enhance compatibility for expandability and evolutionary growth of the command and control system and supporting subsystems; that is, will it enhance the ability to add and subtract subsystems as needed to achieve changes in capabilities?

d. Will application of local network technology afford backward compatibility?

e. What C2 nodal functions, in addition to information input, can be enhanced by local data network technology?

f. What significant improvements in C2 functions can be realized by capitalizing on the intrinsic high band-width of local network technology?

g. How well can local network technology support the distribution and exchange of graphic situation displays? Examples: high resolution bit maps of radar images, real-time handling of NTDS displays, real-time update of positions and identifications based on formatted message inputs without human intervention.

h. What are the implied changes, if any, in C2 center procedures, that would result from applying local network technology?

i. What will comprise meaningful demonstrations of the utility of the planned technology application?

j. Will connection to the network still allow for the independent operation of subsystems in case of busline failures? [Ref. 2]

For the most part, these questions relate to such general C2 issues as flexibility, interoperability, connectivity, and versatility. It would seem that if these issues are addressed by and concentrated on in the CCN, then the project could support the TFCC in a manner consistent with project goals. As this presentation continues, it is important to consider whether the project does in fact sufficiently consider these issues in all aspects of system development.

D. C2 AND NAVY INFORMATION SYSTEMS

Figure 1 illustrates the function of the Command Center Network as the universal communications medium between C2 modules and Navy Information Systems. The acronyms used on the figure are explained below:

KG - cyptological equipment

CV-IC - contemplated development of a carrier configured information center subsystem

NTDS - Navy Tactical Data System

SSES - Ship's Signal Exploitation Space (handles processing of Security Agency related information)

NAV - Navy navigation subsystems.

E. CCN TO SUPPORT THE MULTI-PLATFORM COMMANDER

If one can predict and control the crash landing of an orbiting space vehicle, why can't installed computers provide data of interest concerning ships in a task force? This question typifies a commonly

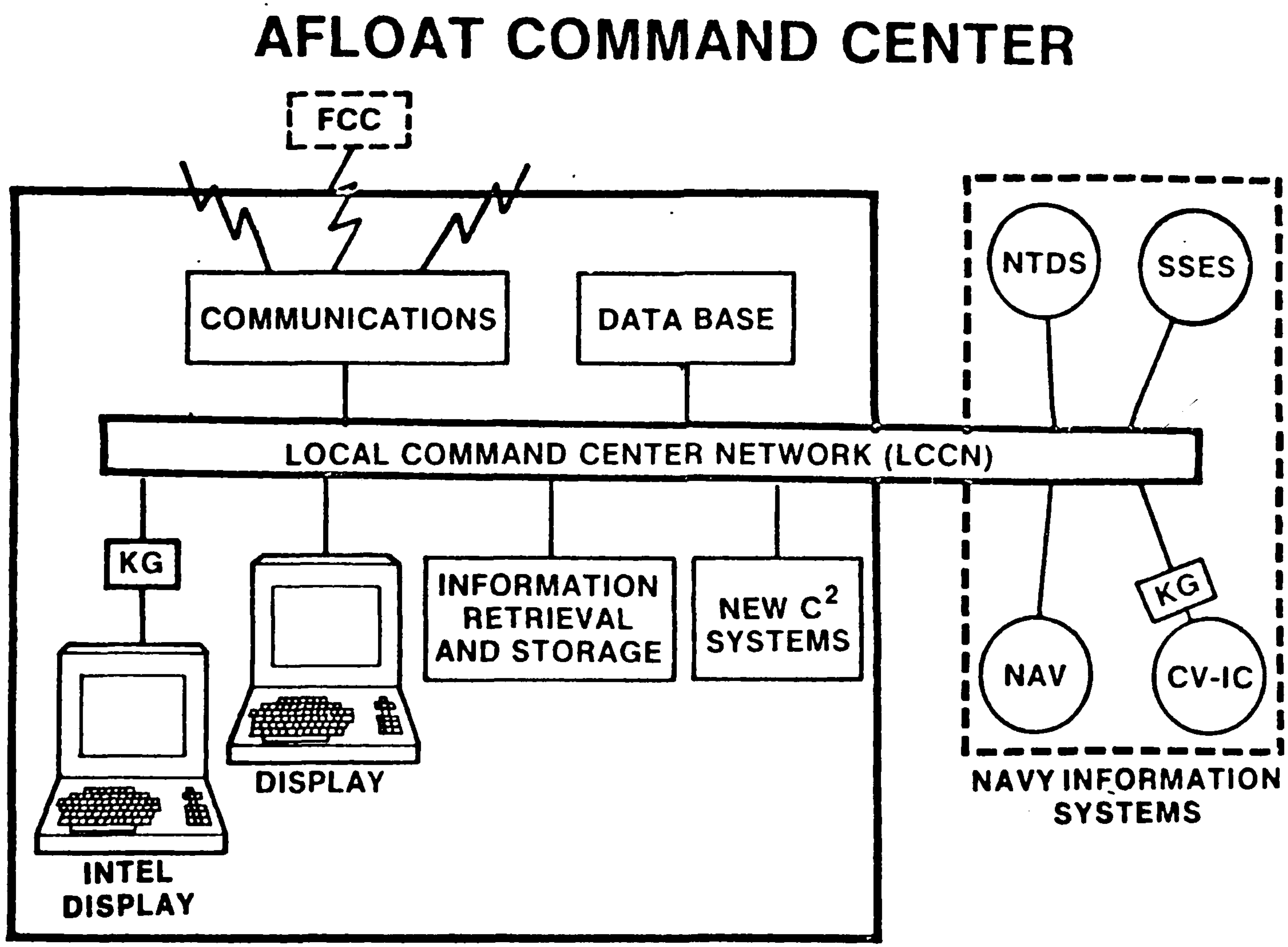


FIGURE 1: RELATION OF C2 TO NAVY INFORMATION SUBSYSTEMS

Source: Navy Command Control Using Local Networks vocalized frustration of multi-platform commanders (i.e. task group, task force, fleet, etc.) and voices a problem with which the Navy continues to wrestle. [Ref. 3]

Throughout the past fifteen years, technology has advanced the capability of a single ship to attack a remote target utilizing a low-flying missile which may have to find its way through an "obstacle course" of intervening platforms. This type of capability necessitates an extremely accurate picture of the situation. A multitude of new systems, from NTDS to Outlaw Shark (AN/USQ-81(V)), have been designed to improve the quality of the information and provide a data bank for the commander. In this way, the commander is able to base his decisions on accurate, up-to-date information. However, integration and correlation of all available information to one or two displays has been limited in applications to the Outlaw Shark program, where three system installations on a single large platform has proved the most acceptable application of totally correlated information. Figure 2 summarizes the functions a multi-platform commander must perform utilizing this information. Some of the considerations related to this information are:

1) There is a lot of data that is needed in order to properly position platforms, control weapons and sensor emissions, and diversify all other capabilities in order to maximize force effectiveness.

2) Answers to queries of the information base are needed quickly (typically between five and ten seconds).

3) The tactical environment must be displayed in a manner that is useful and simple to comprehend.

4) Necessary data may change rapidly, dependent on the mission, friendly and enemy force capabilities and the state of warfare. [Ref. 4]

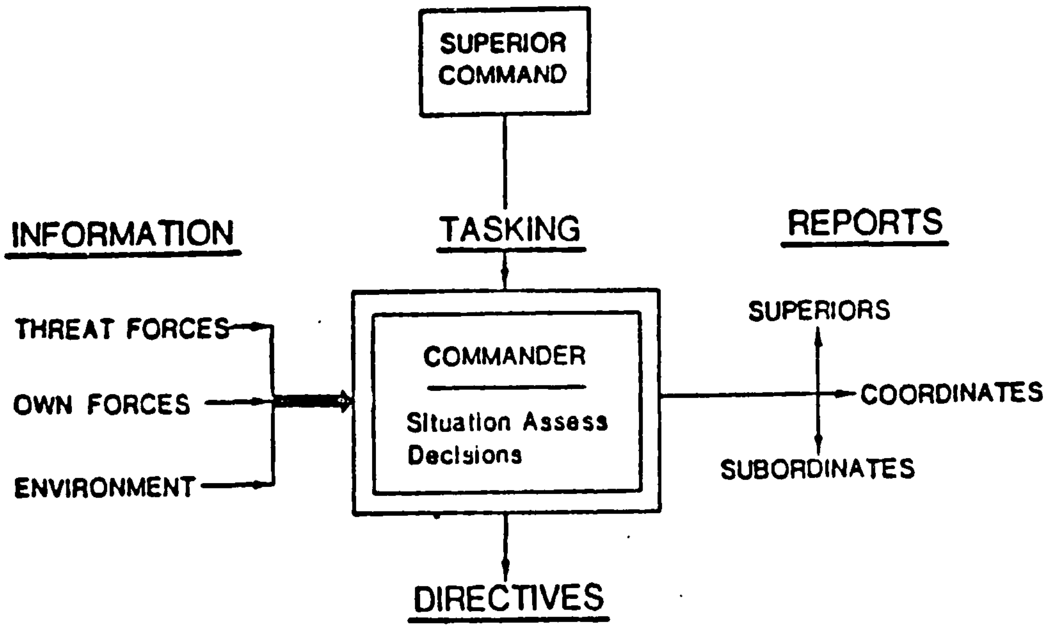


FIGURE 2: FUNCTIONS OF THE MULTI-PLATFORM COMMANDER

Figure 3 shows the traditional method being utilized to provide the necessary information to the multi-platform commander. Typically, responsibilities are divided in the following manner:

1) ASW (Anti-Submarine Warfare)
2) AAW (Anti-Air Warfare)
3) Surface Warfare
4) Intelligence, and
5) Electronic Warfare.

Other methods of assigning responsibilities are nominally at the discretion of the commander. The commander's staff is faced with an exponential growth of available data. Also, the complexity of the tactical situation has resulted in more sophisticated queries by the commander and the staff needing more timely information in order to provide quality responses. Therefore, the staff member's memory is tasked past its limit, and the old grease pencil and status board approach does not lend itself to presentation of information that may become "stale" minutes after plotting.

Within the next ten years, it is envisioned that the Tactical Flag Command Center (TFCC) will become the backbone of the distributed information base of the multi-platform commander. CCN is designed to be the interface mechanism to correlate the crush of data into a simple, straightforward series of displays. On a succeeding page, Figure 4 graphically shows the enormity of the tactical information and command, control integration mission.

CCN is seen by its developers as intrinsic to the correlation of the entire local information data base. Succeeding chapters will discuss how CCN will be designed and implemented to correlate the multitude of

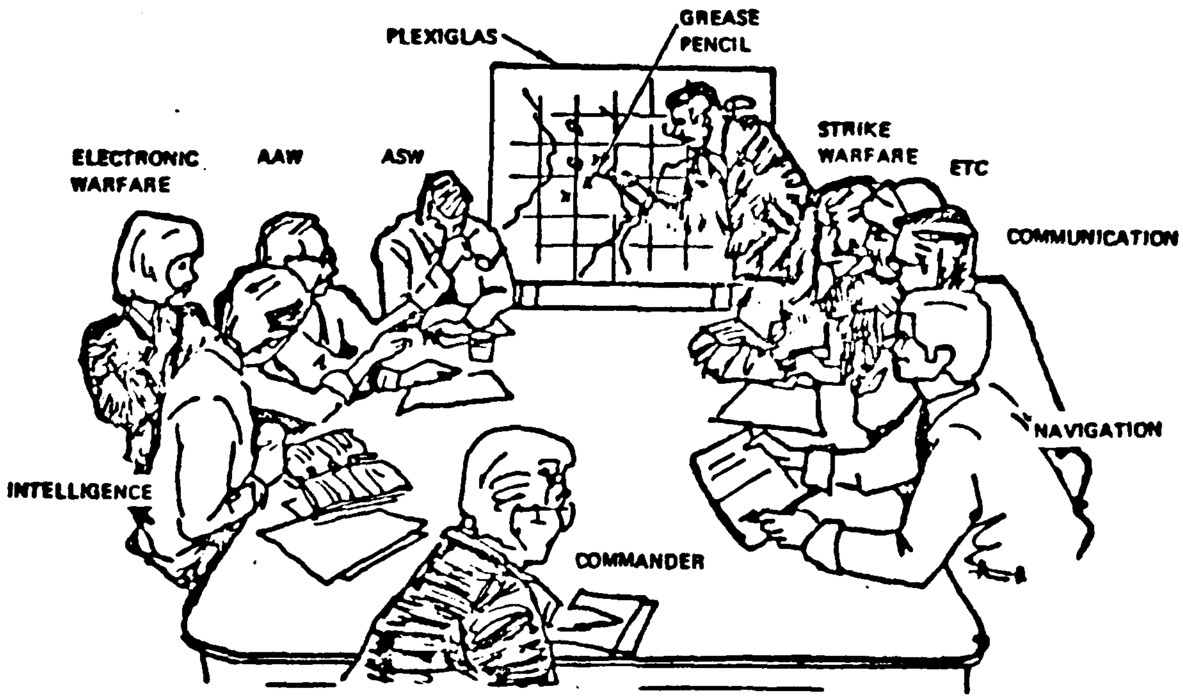


FIGURE 3: THE COMMANDER AND HIS STAFF

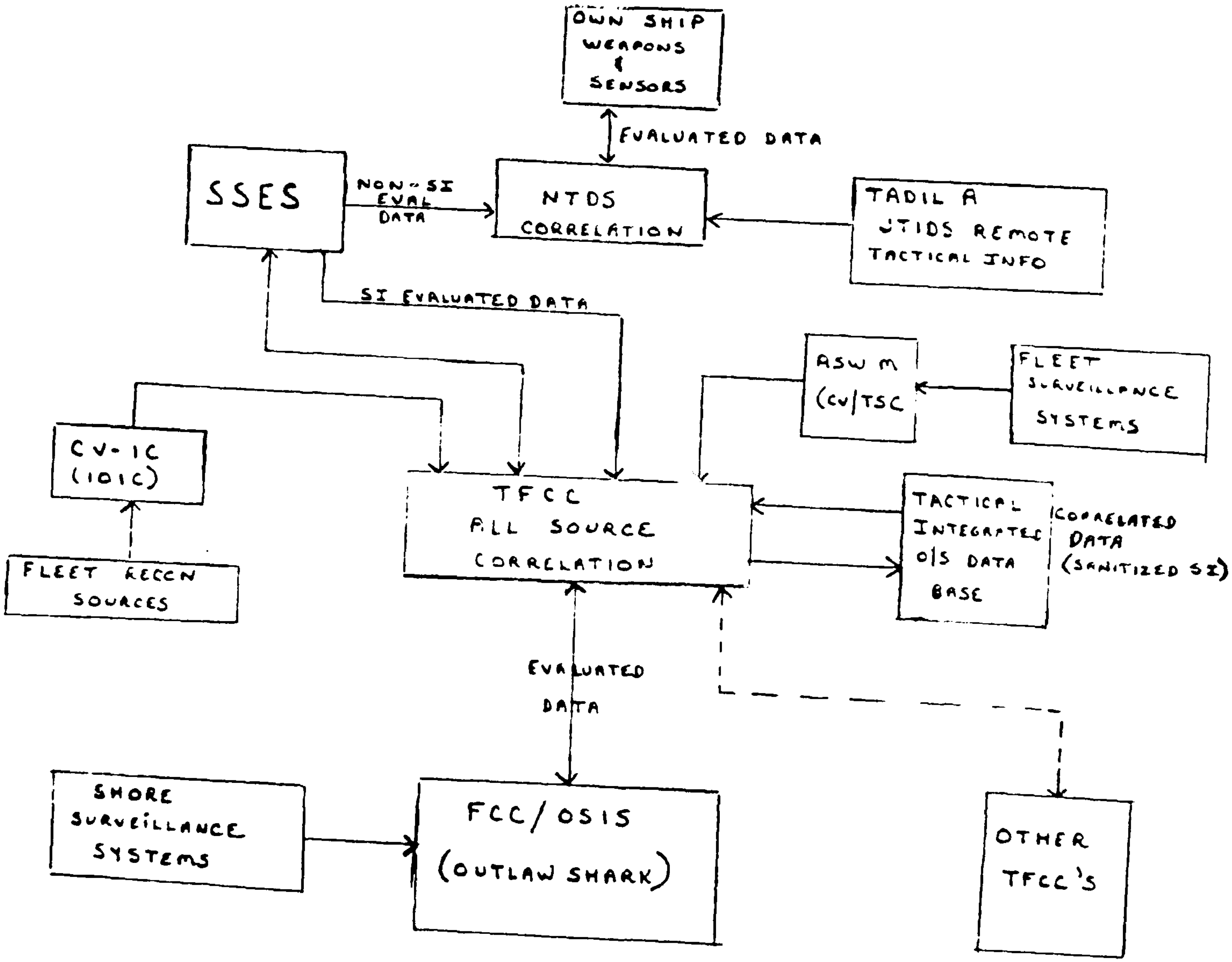


FIGURE 4: TACTICAL C2/I INTEGRATION MISSION

already employed "stand-alone" systems, which were dedicated to particular functions while communicating principally with a human user. Furthermore, the set of interrogations these individual systems are designed to answer is based on the premise that there exists a limited, standard set of questions that a commander might ask. If these systems are interconnected, will there be a standard set of queries? To date, no standard set of inquiries has even been envisioned. This may imply that the boundaries of possible questions may not be definable.

In essence, then, CCN must be capable of interfacing the independent subsystems and providing a protocol system flexible enough to respond to a large number and type of inquiries. [Ref. 5]

III. CCN AND RELATED TECHNOLOGY

A. SUBSYSTEMS CONNECTED TO CCN

Each Navy information and C2 system was developed to stand alone. This and the absence of a standard set of queries by the multi-platform commander affect CCN system design.

When consideration is given to interconnecting such systems, one must consider the nature of systems designed to perform a unique set of functions in a stand-alone mode. Some of the characteristics of such stand-alone systems are as follows:

1) Computers "talk" only to devices which understand their language. Since each subsystem was designed without a requirement to exchange data with other systems, the designer wrote programs which met individual subsystem needs. Thus, subsystems use different word lengths to describe similar parameters, have different symbology instructions, and employ varied protocols for data exchange.

2) The cost of militarized hardware, coupled with a technology lag typical in all military applications has resulted in computer systems which generally run at or near full capacity. Typical of this was the recent update of the Naval Tactical Data System (NTDS) software which required some tradeoff considerations of features to be removed in order to incorporate new capabilities. There are severe limitations which restrict the modification of software within a computer in order to provide "translation" capabilities.

3) The independent subsystems were designed to provide data to a human user. Since humans are quite adaptable, they can understand interference or "noisy" data and displays and, if necessary, repeat queries to the computer.

4) Each subsystem can only respond to the specific set of questions which it is designed to answer. Each new question must, consequently, be programmed into the computer by a specialist programmer familiar with that particular subsystem. Current technological developments provide a more flexible query capability which must communicate in multiple "languages" in order to be used effectively with the Navy systems. [Ref. 6]

These four major characteristics are presented to show some of the problems which the CCN must deal with in order to interconnect existing and contemplated equipments. Since the early 1970's, the Navy and NOSC have undertaken two separate projects to provide the multi-platform commander with a coordinated information bank and display. Both of these projects fell short of the intended goal for two major reasons. The first was the nonexistence of a demonstrated technology of external query processing. The second reason was that "Command and Control Requirements" were constantly changing. The ever-changing nature of C2 requirements resulted in stagnation of the individual projects. Therefore, incumbent on any technological effort, is the requirement that the command center and any other network interconnection allow for future additions and deletions of C2 and information subsystems. The ideal solution, that the commander have access to all "relevant" Navy subsystems, has necessitated consideration of current and future C2 and information subsystems in a "modular" form.

However, in an effort to plan for the addition of future C2 and Navy Information subsystem modules, the goal to interconnect existing subsystems as a baseline to any future applications should remain paramount. With the development and demonstration of a CCN capability, standards and/or specifications can be promulgated for future subsystems, including a proven interface capability with the network. Therefore, in ascertaining whether the CCN project remains focused on its original goal, one must be conscious as to whether the project is designed to utilize existing technology to interconnect existing C2 and Navy Information subsystems.

B. SUBSYSTEM INTERFACE REQUIREMENTS

Two distinct technologies have evolved during the past decade that form the basis for the CCN system. The first is the development of the high-speed (1-100 MBps) data bus for information transfer between closely grouped elements. Such a system connection allows a number of nodes (i.e. users and/or subsystems) to be connected through the same physical "wire". This connection requires the utilization of protocols so that the "wire" can transfer data between two or more users. Bus connections, such as that which is to be applied to the CCN, are already in use in single computers to tie memory and processing units together. They have been shown practical in systems such as SHINPADS (Shipboard Integrated Processing and Display System), which is the current Canadian Navy system for interconnecting shipboard sensors and weapons.

The second technological development is the evolution of a computer network to connect the diverse information subsystems, while meeting the rigid interface requirements of the data bus. The ARPANET, a partial solution to the single data bus interface problem, has succeeded in connecting a variety of diverse computers, called "hosts". These computers differ greatly in type, ranging from PDP-10's and Nova 800's to Honeywell 6000's. Each host or node, characterized by different speeds, word lengths, and other characteristics, has been connected by a "common" language which allows them to communicate with one another. In essence, any user with the proper code or password may communicate with any of these host computers. This capability required the development of standardized protocols to facilitate data exchange. Additionally, special software is required for each host computer to provide a translation between computer specific protocols and the overall ARPANET standard protocols. Thus, the task of interconnecting different subsystems is technologically feasible. By substituting the high-speed data bus for the telephone wire utilized in the ARPANET, one interface problem is solved. However, in contrast to host systems in the ARPANET, existing Navy systems allow for no modification to individual software. [Ref. 7] The Command Center Network (CCN) must deal with this processing problem in current systems and ensure that future subsystems can interface with the network whether or not individual software is modified to accomodate standard network protocols.

In order to demonstrate the complexity of the interfacing problem within the CCN, two of the proposed connected subsystems are discussed.

The Naval Modular Automated Communications System (NAVMACS) integrates currently employed message communications methods and equipment into an automated system offering higher levels of communication capability. It is designed to increase the speed, efficiency, and capability of all phases of Naval afloat and ashore communications, while reducing manhours and error margins. The modular concept of NAVMACS allows the system to be deployed according to requirements without the installation of total packages. This modular concept is represented by two major equipment configurations: NAVMACS A+ and NAVMACS V2. This subsystem serves as an automated shipboard terminal for a satellite link interfacing shore with the Common User Digital Information Exchange System (CUDIXS) network. NAVMACS provides automated accountability for all incoming and outgoing messages. Interconnection with the CCN could allow the commander to determine when the last updated RAINFORM message was received, what the last intelligence summary emphasized, what time his message orders were sent, etc. One of the major functions of integration necessary when connection of this subsystem occurs is that of correlating locating data received from message traffic with positions held by other sensors, whether they be own platform sensors or from other resources available to the task force.

Another subsystem, NTDS, consists of shipboard computers linked by wire to a ship's sensors and by radio data links to other ships in a formation, as well as to aircraft such as the E-2. If two platforms have the NTDS system installed, the interconnection and exchange of tactical information between the two is in the form of computer-to-computer messages. The connection is known as Link 11. If an NTDS ship transfers data to a non-NTDS platform, the format is via message traffic conveyed by radio frequency. This is called Link 14. The exchange of data between a ship and an aircraft like the E-2 is called Link 4A. Essentially, NTDS is based on the concept that sensors aboard different ships and/or aircraft will mutually reinforce, so that their effective ranges will be greatly increased, and the task force commander's tactical picture will be enhanced and more accurate. The goal of the NTDS system, which is essentially to allow multiple platforms to act in concert, is similar to that of CCN. In order for ships to act in concert, information must be consistent, accurate, and highly correlated.

Not only do the NAVMACS and NTDS subsystems provide necessary information to the commander, but they also are networks or part of other networks. The issue of interoperability must consider not only the interconnectivity of such subsystems as NTDS and NAVMACS, but also whether or not the interconnected subsystems interfere with one another. Therefore, the feasibility of interconnecting networks in a CCN configuration must consider the interactions among subsystems.

C. SUBSYSTEM INTEROPERABILITY

Interoperability becomes a major consideration when connecting complex subsystems. Interoperability is the ability of systems, units, or forces to provide services to and accept services from other systems in such a manner that these exchanged services can operate effectively together. In order to be interoperable, the subsystems must operate in a non-interfering manner.

Major interoperability issues are:

1) that each of the CCN subsystems must display backward mobility, which means that individual subsystems may be disconnected from the network and operate independently from a still functional network, and

2) that the connected subsystems must possess an identifiable, shared tactical goal or purpose.

It is important to note that the existence of interfaces does not guarantee interoperability between the subsystems. Items exchanged through interfaces might have a positive or negative impact on the individual subsystems. If even some of the effects are negative, then an interference exists.* Therefore, for the CCN to serve the purpose it is designed to fulfill, the identification of all interfaces and possible interferences is extremely important.

The ability to surface the aforementioned interoperability issues in a substantive manner as early as possible in the evolution of the CCN

*See NAVMACS discussion, p. 38.

project is essential to system success. Even if all the necessary interfaces and interferences cannot be identified either theoretically or during prototype installation, the demonstration of the prototype should significantly address and demonstrate solutions to interoperability concerns. For example, some specific non-interference interoperability concerns are:

1) Will the NAVMACS subsystem operate within its own network simultaneous with connection in the Command Center Network?

2) Will the NTDS subsystem function efficiently as part of its own independent network and provide the necessary information via the CCN?

3) Will the actual performance of NAVMACS and/or NTDS, as well as any other subsystem, be degraded in any manner due to connection via the CCN?

These questions truly address the goal of CCN to interconnect C2 and Navy information systems to support the TFCC. To degrade the performance of any subsystem by connecting it to the Command Center Network could negate the advantages to the commander garnered from such a system.

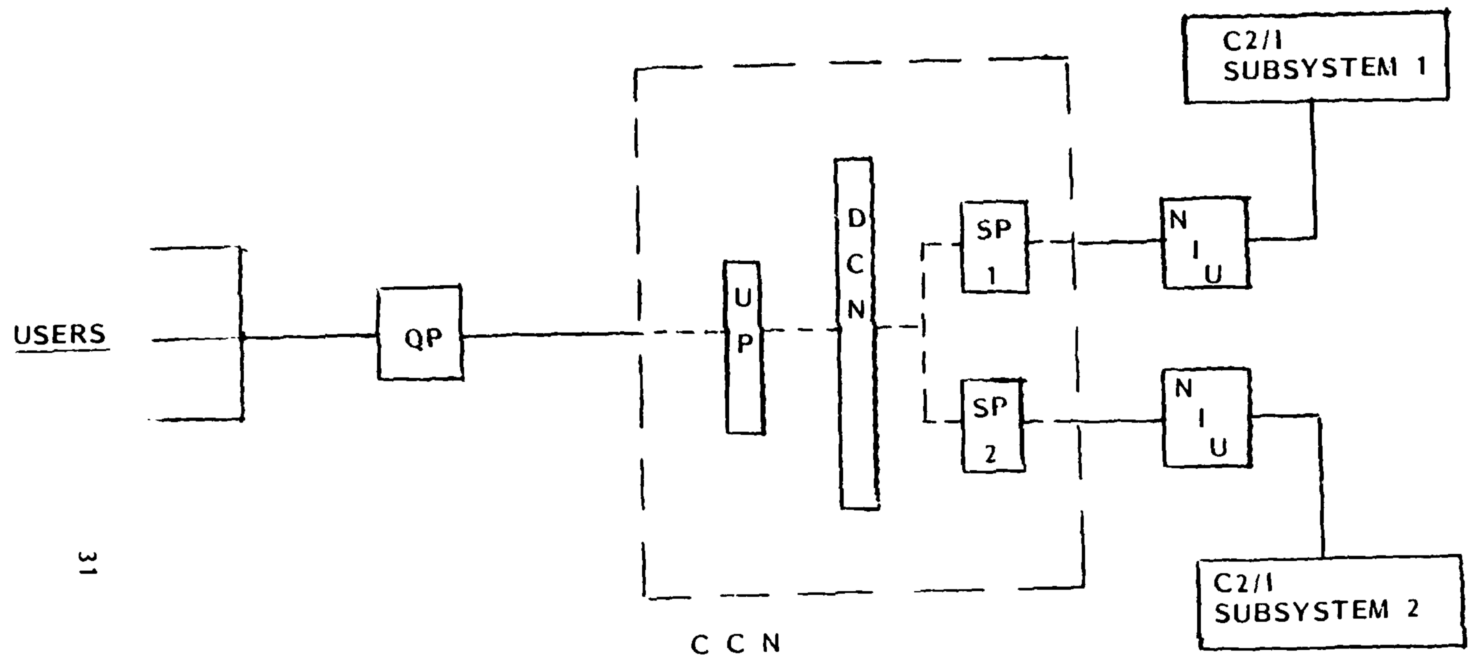
IV. CCN PROTOCOLS AND SOFTWARE

A. INTRODUCTION TO CCN PROTOCOLS

The protocols necessary to carry out the functions of the CCN are outlined in this section. Generally speaking, a set of rules and the associated software to make subsystems interact (i.e. protocols), enable systems to communicate with one another. Protocols provide a critical service in any network. In the CCN, where "remote" (i.e. isolated) subsystems are interconnected, protocols perform such basic functions as the separation of various data streams, reliable sequenced message delivery, and the provision of system-to-system (end-to-end) flow control.

For each C2 subsystem interfaced to the CCN, a set of programs is defined as user and server interacting together. Figure 5 shows the user and server processes as defined in the CCN. The server process occurs through the Network Interface Unit adjacent to the CCN and into the subsystem connected to that particular interface unit. The user process is generally considered as the actions necessary to connect a query into the CCN. For a given user process, there may be one or more server processes necessary. Appendix A explains the user and server process interaction in the NAVMACS subsystem.

The user and server programs will interface to the network via a standard transmission control protocol (TCP). CCN utilizes TCP4, the accepted DOD standard for TCP's. Appendix B outlines the required set of user calls, commands, and user/server function codes required in a TCP.



KEY

QP - Query Processor
UP - User Process
DCN - Data Communications Network
SP - Server Process
NIU - Network Interface Unit

FIGURE 5: CCN USER/SERVER PROCESS

In addition to the standard protocols mentioned previously, protocols must be developed in order for the individual subsystems to be addressed. These protocols must be designed so that the functions of each of the individual subsystems may be made available to the overall user process. In Appendix C, a listing of the functions of the NTDS/DTS subsystem is presented. In order to understand how these individual functions can be made available to the user, Appendix D outlines a typical user/server process for the NTDS/DTS subsystem. [Ref. 8]

B. TRANSMISSION CONTROL PROTOCOL INTERFACE ISSUES

The Transmission Control Protocol (TCP) is a protocol that has been proposed and utilized for process-to-process communication across connected computer networks. It sets up an association between two processes and manages the flow of data between them on a byte-based windowing principle. The first three major implementations of TCP theory were made at Stanford University, Bolt, Beranek and Newman (BBN) in Boston, and at University College, London. [Ref. 9]

The use of a TCP approach stems from the fact that a critical service that must be provided is an end-to-end protocol for communication between two remote processes (i.e. subsystems in the CCN). Such a protocol should provide standardization of mechanisms for performing basic communications functions (i.e. the separation of various data streams, reliable sequenced message delivery, and the provision of end-to-end flow control). Normally, various services are built into lower levels in a network and those which can be provided by an end-to-end protocol need only be a subset of the ones necessary to communicate. In the ARPANET, for example, a great deal of the end-to-end flow control is done by the subnet between the source and destination switching nodes known as IMPs. Some sample activities in this vein are flow control and the generation and control of various acknowledgements and error conditions. The unique functions of the ARPANET normal host protocol (NCP) are to manage the multiple connections, respond to inputs from IMPs, and to provide host-to-host flow control. [Ref. 10]

For communications across connected networks like the CCN, two approaches to TCP may be undertaken. The first, which is utilized most frequently with virtual circuit networks, is to provide mappings between the end-to-end protocols of the various networks. Mappings then are performed in the "gateways" which connect networks. The alternative approach, which is utilized by the CCN, is the "Transport Station" method. In this application of TCP, there is a universal *end-to-end protocol* which generates and controls message flow in a standard transmit format. Packets are thereby treated as data in each network and are imbedded and formatted according to local network rules. Therefore, gateways support the transnet protocol by packet transport from one local net protocol to the next local net protocol.

TCP implementations have suffered from inefficiencies in past experiments, particularly those conducted at University College, London (UCL). The inefficiencies seem to reinforce the need for as much attention to efficient implementation of communication drivers, such as TCP, as to any other part of the system which is frequently utilized. A study of the overhead of supporting ARPANET's normal host protocol (NCP) on Tenex machines indicates that overhead was only 20% greater than that for the same traffic to local devices. [Ref. 11]

One can analyze the costs of implementing TCP by examination of the various functional areas associated with support of such a specification. These areas include:

1) buffer manipulation,

2) table searches,

3) arithmetic computations, and

4) choice of language. [Ref. 12]

A more detailed presentation of these functional areas is included in Appendix E.

Implementation experience suggests that all these factors must be taken into account. In essence, the design of TCP is one of a class of end-to-end protocol designs which are based on a model of several distinct layers of protocol. Each of these layers performs certain clearly defined functions. The main advantage in this approach lies in the simple network structure produced when all responsibility for acknowledgement, sequencing, reliable delivery, flow control, and other features are concentrated at one level. In order for layering to be applied successfully, unnecessary duplication of protocol features in more than one layer must be avoided. In the design of a single network, this may be possible if the protocol designer can assume lower level functions as given. When a protocol is designed for a system where internetworking occurs, this assumption cannot be made. Duplication of function will almost inevitably occur, leading to the possibility of mutual interference, which has been observed in experiments at University College, London.

CCN addresses these problems by considering protocol requirements at each C2 and information subsystem module. Layering of protocols suggests that once the user "opens" connection to a certain subsystem (or subsystems), a "menu" of available commands particular to the module will be displayed. Figure 6 shows the locations of the various protocols in the CCN prototype.

Through the utilization of standard DOD transmission protocols and the modular approach to protocols for the interconnected subsystems, the CCN project has avoided many of the technological problems with network protocols. If an attempt had been made to expand upon the standard TCP or develop a new set of protocols, the CCN project might have become embroiled in philosophical, as well as technical, disputes as to how extensive a standard set of protocols should be. However, in this complex area, CCN remained focused on considerations involved with interconnecting C2 and Navy information subsystems.

C. INTRODUCTION TO CCN SOFTWARE

A functional description is presented for the set of programs necessary to interface some of the subsystems (i.e. NAVMACS and NTDS/DTS) to the CCN. These subsystems serve as a sample of the total program development necessary to initiate the CCN prototype (XDM). C2 subsystems will be interfaced to the CCN by PDP 11/03 microcomputers. The 11/03's are also referred to as Network Interface Units or NIUs. The NIUs consist of a DEC LSI-11 processor, 64k bytes of random access memory (RAM), 4 asynchronous serial lines, a line time clock, and an 1822 communication interface. The 1822 interface can be used to connect the LSI-11/03 to an ARPANET IMP in a direct memory access mode operating at 50 kilobaud. Figure 7 is a cross-sectional diagram of a Network Interface Unit.

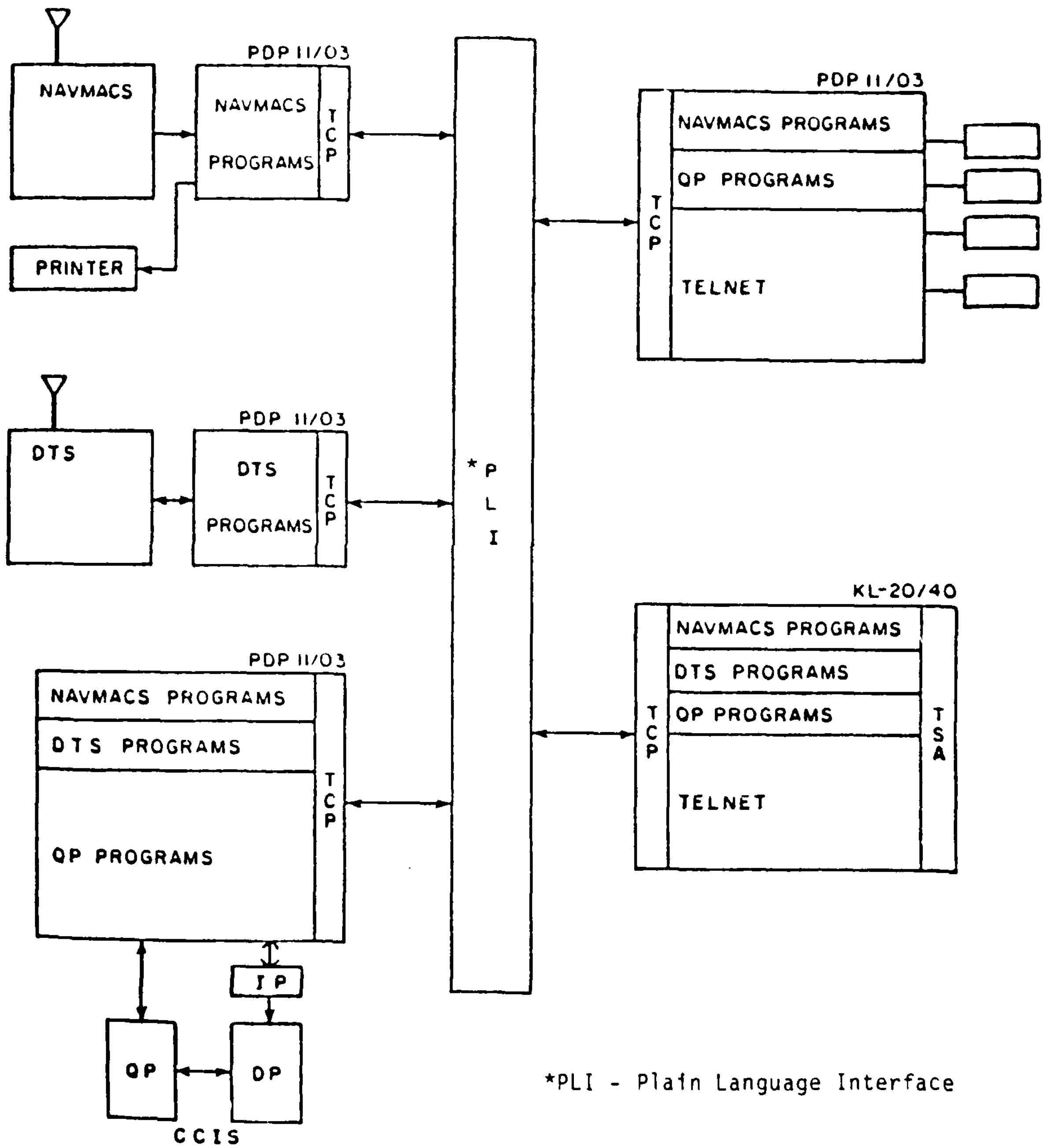


FIGURE 6: PROTOCOL LOCATION IN THE CCN

Source: <u>CCN Protocol Location</u>

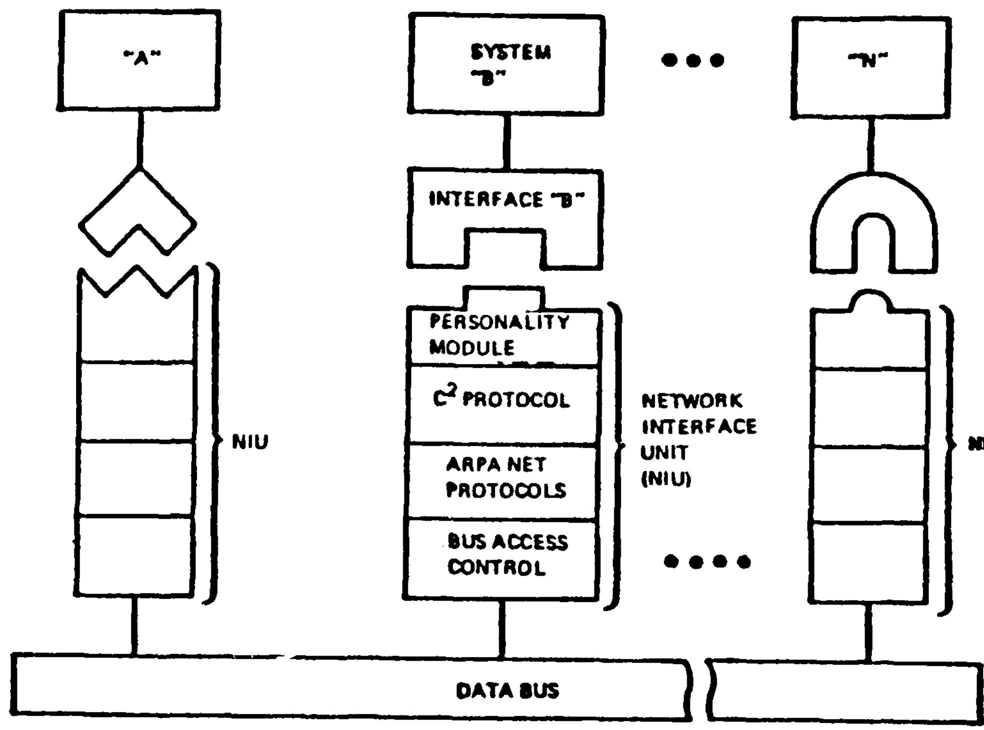


FIGURE 7: CROSS-SECTION OF A NETWORK INTERFACE UNIT

Source: CCN Software Design

Existing software will be used as much as possible, especially for the initial CCN prototype. The NIUs employ Stanford Research Institute's (SRI) MOS operating system and the network protocols Telnet and TCP4. The remaining software that needs to be developed consists of those programs which interface the specific tasks associated with each subsystem. Current planning calls for these programs to be written in a higher order language called BLISS, which is somewhat similar to the "C" programming language.

The following two sections outline the functions of the programs necessary to interconnect the NAVMACS and NTDS/DTS subsystems to the CCN.

1. NAVMACS Programs

NAVMACS messages consist of baudot characters and are, on the average, 2100 characters long. The messages are delimited by a SOM (start of message) and EOM (end of message). Since the NAVMACS processor normally sends the messages to a printer, one way of controlling message flow is setting the printer's ready line to a low voltage. Upon encountering the low voltage level, the NAVMACS processor stops sending the current message. When the ready line is set to a high voltage again, the processor sends the message in its entirety. Messages arrive at the NAVMACS processor over a communications link operating at 75 baud (100 words per minute). The NAVMACS processor sends the characters serially to the printer at 2400 baud. This processor has 32k bytes of storage space available for incoming messages. If this space becomes full, the processor is programmed to keep new messages out of the system.

The functional requirements of NAVMACS programs include:

- deliver NAVMACS messages to terminal users on the CCN and to processes like TSA and IP
- allow a parameter indicating what kind of messages the user is interested in receiving (the user, throughout this discussion, could be a process, a terminal, or a printer)
- require a user to login or a process to authenticate itself
- signal a user when NAVMACS messages arrive
- allow a user to file messages for later retrieval
- allow a user to stop the process at any time
- inform the user of net errors which result in loss of messages
- convert baudot to ASCII
- employ a multi-addressing scheme to deliver the same NAVMACS message to several users
- send each NAVMACS message to the NAVMACS TT624 line printer
- allow the NSM to have NAVMACS messages sent to third parties (the NSM can arbitrarily decide that a process or terminal on the CCN should receive certain NAVMACS messages)
- filter messages based on subject or headers
- allow the user to print out all message headers
- inform the user when the NAVMACS processor is being held off (the processor is held off whenever buffer space is full or hardware is malfunctioning)
- allow a terminal user to direct NAVMACS messages to a third party

- convert RAINFORM formatted messages to CCN format [Ref. 13]

The XDM prototype at the Naval Ocean Systems Center will not be designed to perform all the aforementioned functions. In an effort to produce a working prototype at the earliest possible date, the initial CCN demonstration will perform only the following operations:

- deliver NAVMACS messages
- allow a parameter indicating the type of messages the user is interested in, but limited to all messages in RAINFORM format
- signal user when NAVMACS messages arrive
- allow the user to stop the process at any time
- convert baudot/ASCII
- employ a multi-addressing scheme to deliver the same NAVMACS message to several users (the scheme used will be to send the message once for each interested user, since the TCP currently does not support multi-addressing) [Ref. 14]

It is apparent from the omissions in the above list that the original installation of the CCN (i.e. the XDM prototype) will only touch the edge of the iceberg involving message processing software.

2. NTDS/DTS Programs

Data from the DTS computer comes in binary form over a 30-bit wide NTDS "slow line" to the NTDS computer. The data from/to the DTS is binary, with a start/stop data word and a track number for historical purposes. The NTDS control lines will appear to the LSI-11/03 (Network Interface Unit) as RS-232 control lines so that the existing protocol (as described in NELC TM-119 Interface Design Specification) can be employed in the LSI-11/03.

DTS programs are designed to perform the following functions:

- deliver track data from the DTS computer to interested users
- deliver track data from users to the DTS computer
- require users to login and identify themselves
- prevent transmission over CCN of track reports containing no change in data field
- deliver tracks based on content (air tracks to some users, surface tracks to others, etc.)
- signal the user when tracks arrive from the DTS computer
- employ a multi-addressing scheme in order to deliver the same tracks to several users
- inform users of success/failure of tracks sent to the DTS computer
- convert track data from binary to ASCII
- store track data for later retrieval
- allow NSM to have tracks sent to a third party and filter on subject or content, i.e. the NSM can change the addressee list (for the purpose of insuring that certain processes on the CCN get all air track information or surface track information etc.)
- convert ASCII to binary
- convert to/from CCN format
- allow an option to disable the default of receiving all tracks and receive only certain tracks based on some filter. [Ref. 15]

In order to demonstrate the feasibility of interconnecting DTS/NTDS programs in a Command Center Network, the prototype will perform the following operations:

- deliver track data from the DTS computer to interested users
- prevent transmission over CCN of track reports containing no change in data fields
- signal the user when tracks arrive from the DTS computer
- employ a multi-addressing scheme to deliver the same tracks to several users
- convert track format from binary to ASCII, and
- convert to CCN format. [Ref. 16]

The complexity of NTDS/DTS operations utilized on platforms configured with this subsystem necessitates the development of software to handle more operations than the XDM exhibits. In order for the CCN to truly interconnect NTDS in a manner sufficient to support the TFCC, the remainder of the program functions should be integrated in the prototype.

V. CCN PROTOTYPE (XDM)

The architectural testbed for the Command Center Network, which is called the XDM, is to be installed at the ACCAT facility located at the C3 Site, Naval Ocean Systems Center (NOSC) in San Diego. Figure 8 is a conceptual diagram of the XDM installation. It shows the four major subsystems to be interconnected by the initial demonstration. The arrows in the diagram represent the contemplated flow of information in the XDM.

Within the XDM, there will exist a Data Communications Network (DCN), which is composed of some transmission media. Access to the DCN will be through one of several DCN access modules (DAMS) still within the CCN. A number of Network Interface Unites (NIUs) will also be in the XDM. These NIUs provide connectivity between the XDM and various C2 and information modules external to the XDM. Also, a gateway (transmission connection link) to the ARPANET will be included. This is to provide interconnection to other data networks not directly connected by the XDM. The XDM will include a minimum of four DAMs, four NIUs, one gateway, and associated transmission media.

Figure 9 shows the relationship of the aforementioned hardware in the XDM.

A. XDM GOALS AND DEVELOPMENT

Design goals for the XDM fall into two major categories:

1) Initial Development: Goals which pertain largely to qualities and functional capabilities of the XDM to be realized at the time of installation at NOSC.

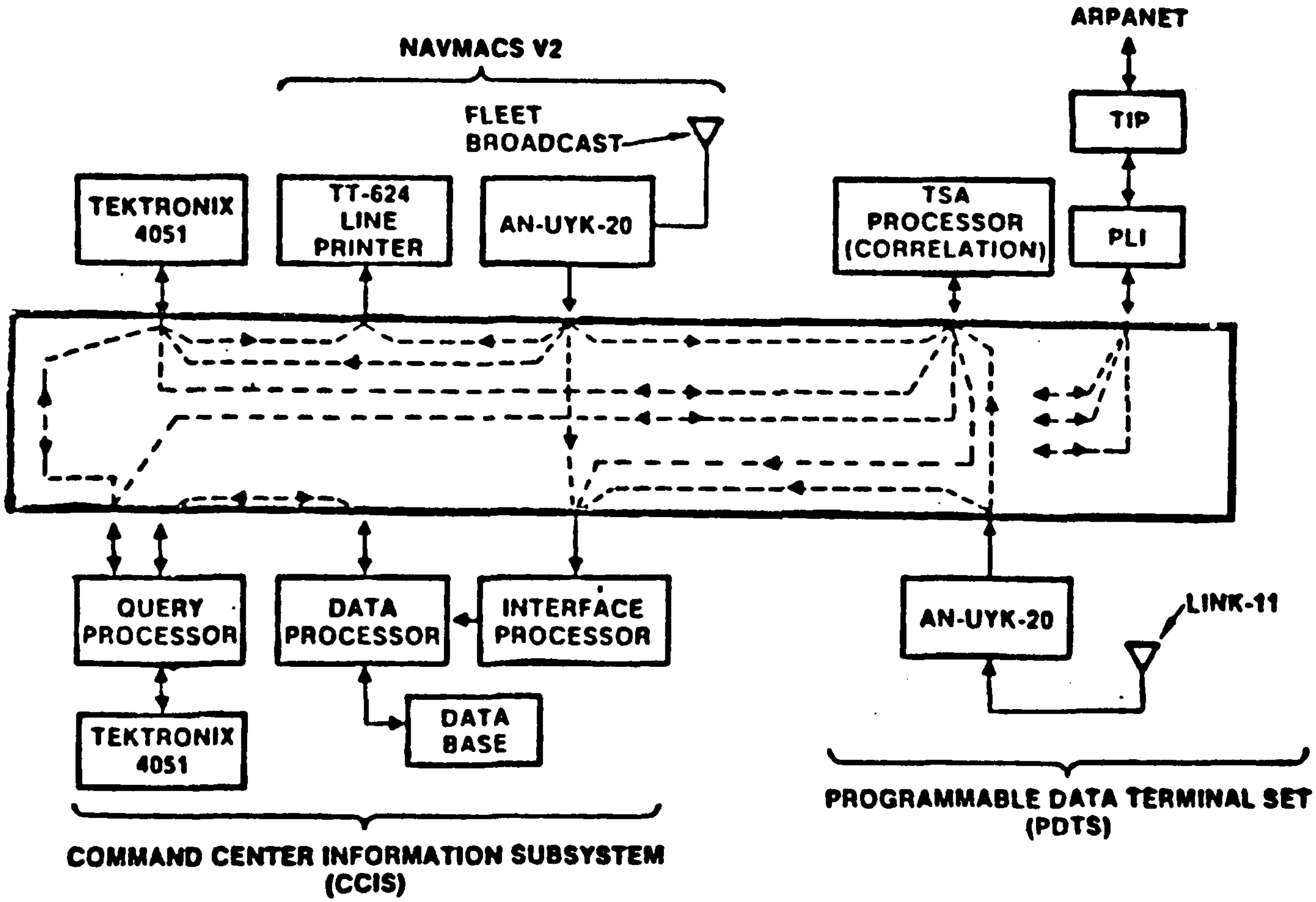


FIGURE 8: XDM SYSTEM CONCEPT DIAGRAM

Source: LCCN Information Flow and Protocol Requirements

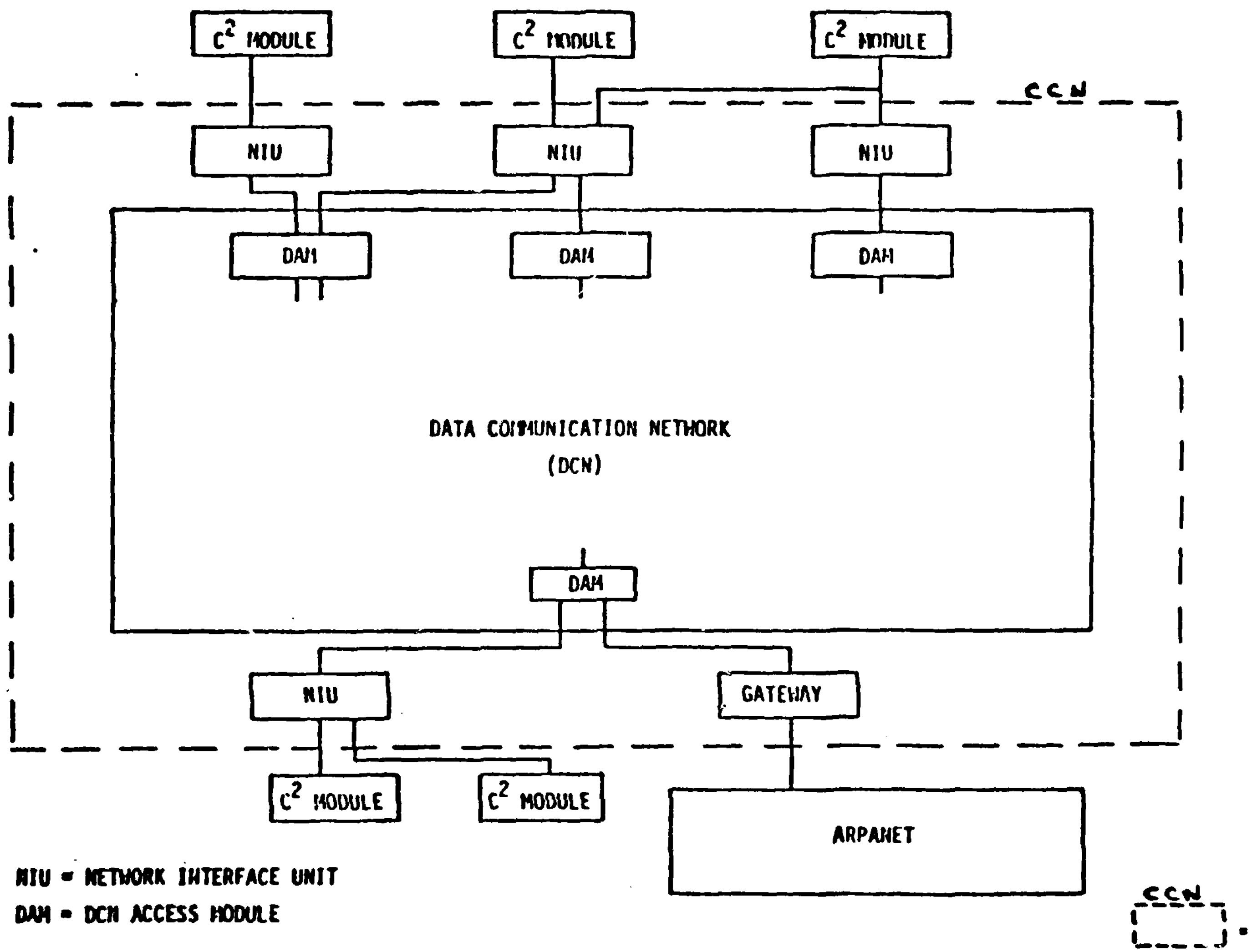


FIGURE 9: CCN EQUIPMENT CONFIGURATION

Source: <u>C2 Modules for Local Command Center Network Demonstration</u>

2) Future Application: Goals which pertain to design characteristics of the initial XDM that make it readily susceptible to changes, modifications, or expansion of capabilities that can be expected to occur during its use as a testbed or prototype in the ACCAT. [Ref. 17]

Initially, the XDM will interconnect selected baseline C2 and information subsystem modules. These modules, which perform well identified and defined functions in support of a command center like the TFCC, will be chosen from either already employed subsystems or those which are in the final stages of development. In this way, it is felt that the technology, capability, and the support of the CCN's overall goal can be demonstrated in a meaningful way.

The XDM test facility is configured to accomodate integration and interoperability testing within guidelines published by NOSC. NOSC defines integration testing as that which is capable of verifying that the appropriate medium has been chosen for the exchange of data between interfaced subsystems, and that the data is appropriately merged as interleaved by the overall system. Interoperability testing is that which is capable of verifying that each participat[illegible] subsystem, as well as the overall system, can successfully generate, transmit, receive, process, interpret, and control the flow of messages either internal to the system or those which originate from external sources. [Ref. 18]

Implicit to the CCN concept is the isolation of users from internal controls, communications, and routing. Isolation of these functions implies the need for a network manager and programmer for each network installation. Those needs further complicate employment of CCN because of training requirements and necessary qualifications. Furthermore, from the developmental standpoint, the simplification of language into a series of fixed commands for the user has necessitated the development of a higher-order language, BLISS, which ensures that communication with the machine or assembler languages of each of the subsystems is maintained. BLISS provides the interfacing language link so that the user can be connected to all resources running in the network.

The distributed arrangements of many military command and control systems, particularly the TFCC, require intersystem protocols, whereby a process running on one C2 or information subsystem may have to task a process running on another system. The XDM must demonstrate this ability without causing any interference in the processes being run on the individual subsystems. In an employed CCN, an example of this capability could be where an intelligence subsystem requires access to messages resident in the communications subsystem message file, and position information resident in the navigational subsystem message file, and position information resident in the navigational subsystem of the tactical data system. Connection, data transformations, file transfers, etc. that are necessary for such services should all be handled by CCN without any need for intervention, direction, or control by the requester. This transparency should accrue from the judicious selection and application of various levels and types of protocols, and is of prime importance in the CCN design and implementation.

The prototype testing must also address the issues of survivability and modularity. With respect to CCN, survivability is the ability to provide network services in the event of internal failures or if subjected to damage or electrical interference from external sources. Modularity refers to the concept that separable network functions should be performed by separate modules. This characteristic is often most applicable to software concerns, where functions are as well formed and as independent of each other as feasible. However, modularity also applies to hardware and it is intended to facilitate system changes.

It remains a challenge to the project managers to ensure that these characteristics are specifically addressed in the initial demonstration plan. If these concerns are adequately addressed, one may conclude that the goal of interconnecting subsystems and supporting the TFCC remains paramount. However, if these issues, and solutions to them, are not part of the XDM demonstration, then one <u>may</u> conclude that a simple demonstration of the ability to interconnect subsystems has become the goal of CCN. In other words, the demonstration of technological feasibility may have replaced part of the original project goal of supporting the TFCC.

B. CCN DEMONSTRATION FUNCTIONS

The initial demonstration of the CCN will interconnect a mixture of existing Navy systems and developmental C2 subsystems. Although limited in scope, the XDM allows the demonstration of the following functions:

1) <u>Gathering of Data</u>: A number of data sources are gathered through the CCN. The first data source is that which is normally transmitted over LINK-11 and stored in the NTDS computers. This data provides pertinent information on platforms which have been detected by the aggregate of sensors on the various task force platforms and thus provide a fairly extensive "local" picture. A second data source is the message traffic which is processed by the NAVMACS (Naval Modular Automated Communications System). Through this channel, data from remote sources, such as shore sites and satellites, is received by the ship. Effectively utilized, it can provide a larger picture of the environment, beyond the sensors organic to the task force. A

third data source included in the initial demonstration of the CCN is the storage of static data such as platform characteristics, rules of engagement, order of battle, etc. All three of these data types (local, remote, static) are stored in a single data base, the Data Processor of the Command Center Information Subsystem.

2) Processing Data: Because of the large amount of data available, some processing is required to reduce it to a manageable size. Since the principal interest of the commander is in getting an accurate view of his environment, a data fusion processor has been implemented as part of XDM (TSA). This processor, not a part of the CCN development itself, incorporates algorithms which implement human reasoning in the fusion of data of different types. It is particularly useful in the identification of platforms which have been detected but not identified. For example, this processor recognizes that a ship which goes out of its way to stay in a storm does so to remain undetected and is more likely to be a warship than a merchant. Here the automated application of human reasoning is applied to the "local" data provided via the Data Terminal Set and the "remote" sensor data provided via NAVMACS. A single, unifying picture is thus available to the commander, uncluttered by the multiple sources required to compose it.

3) Displaying the Data: Two types of display are available to the user; a graphical display and an alphanumeric display. The graphical display provides for the user a geographic presentation and identification of the platforms (surface, subsurface, air; friendly, enemy, neutral) within his area of interest. This geographic presentation includes a "zoom" capability which allows the user to either focus on a small area or to get the picture of a much larger surrounding area. The alphanumeric display allows the user to request data which is stored in the Data Processor. Such data can include characteristics (weapons, sensors, radio call sign, number of screws, etc.) of platforms of interest, or any other data stored in the data base. This alphanumeric display of data is enhanced by a natural language query capability which is inherent in the Query Processor of the Command Center Information Subsystem. This allows the user to ask questions in a "natural" way, eliminating the need for the user to be intimately aware of the computer "language" in order to ask questions. For example, the user may simply type in the question, "Show me all ships within 150 miles of the aircraft carrier". The response will be a listing of all such ships on the display. Although not a part of the CCN development itself, this query capability demonstrates in a powerful way the diversity of capabilities which can be applied once universal access to the data sources has been provided by the CCN. A third method of displaying the data is provided by the TT-624 printer which provides a hardcopy of any alphanumeric

data of interest. A typical usage would be the listing of messages received or some specific ship characteristics which are used frequently.

4) Evaluating the Data: With the existence of the CCN (and appropriate C2 modules), the commander and staff are relieved of the time-consuming process of accessing and manipulating data and are free to perform in the areas where humans excel - evaluation of data and decision-making. The decision-making process is a long way from automation and so the CCN serves principally to prepare the data in a way that the commander can make best use of it in assessing his alternatives. Should such automated techniques be developed and accepted by operational Navy personnel in the future, the CCN structure will already be in place to easily implement it because of the basic design which support the modular addition of such capabilities.

5) Disseminating Information and Orders: Through NAVMACS, the user can prepare and send messages to participating units. These messages may contain amplifying information as well as information. They may be either free text or formatted such as RAINFORM. The other capability provided by this initial selection of equipments is the opportunity to inject into NTDS a new or better track which has been determined by the data fusion node. For example, if the data fusion identifies a particular ship for which only the position was previously known, this information may be prepared in the Link-11 format and sent out via the Data Terminal Set.

6) ARPANET Connectivity: The CCN will be connected to the secure ARPANET. Other secure ARPANET facilities are located at the Naval Postgraduate School, Fleet Numerical Weather Central (Monterey), Naval Research Laboratory, CINCPACFLT, Acoustic Research Center, and Mobile Access Terminals (MATs) which are located on ships. Thus remote users can connect, via telephone lines, directly to the CCN and thus have access to all of the facilities and capabilities described above. For example, a shipboard commander can transmit his data to the CCN and subsequently query the data base remotely. [Ref. 19]

As outlined above, the demonstration of these six functional areas in the XDM serve as the contemplated focus for further expansion of the CCN to include total interconnection of all current and envisioned C2 and Navy Information subsystems. This initial demonstration is currently scheduled for August or September of 1981.

The XDM is configured in such a manner as to adequately address the issues of interconnectivity and modularity. Through the utilization of existing technologies, the CCN prototype will demonstrate interconnection through the correlation of information from four different subsystems. The modularity issue is effectively handled through the utilization of NIUs, which can be connected and disconnected from the data bus. These NIUs serve as the interface to connect subsystems to the CCN. However, two other issues that should be addressed by XDM are survivability and interoperability as it relates to non-interference.

If all of these issues are addressed by XDM. then the focus of the CCN project remains interconnection of subsystems <u>and</u> support of the TFCC.

VI. CCN AND THE FUTURE

Given that the initial CCN demonstration proves successful, the modular structure of the network will allow for future growth and modification. "Universal" access to various data bases provided by the CCN allows for addition of other subsystems through the use of a "personality module" and a standard interface (both of these additions are included in a Network Interface Unit). C2 subsystems operating on the data bases can be added by conforming to standardized CCN interfaces and protocols.

In fact, with the existence of a proven prototype of the network, a host of new capabilities can be investigated. Figure 10 illustrates some of the envisioned additions to the CCN prototype. These additions include:

1) Voice to Natural Language Converter,
2) Natural Language Processor,
3) Text-to-Voice Synthesizer,
4) Sophisticated Graphics and Display Devices,
5) Multiple terminals,
6) Telephones,
7) Bulk Storage,
8) Data Base Management System,
9) Data Fusion,
10) Alerting Elements,
11) Microfilm Retrieval,
12) Decision Aids,

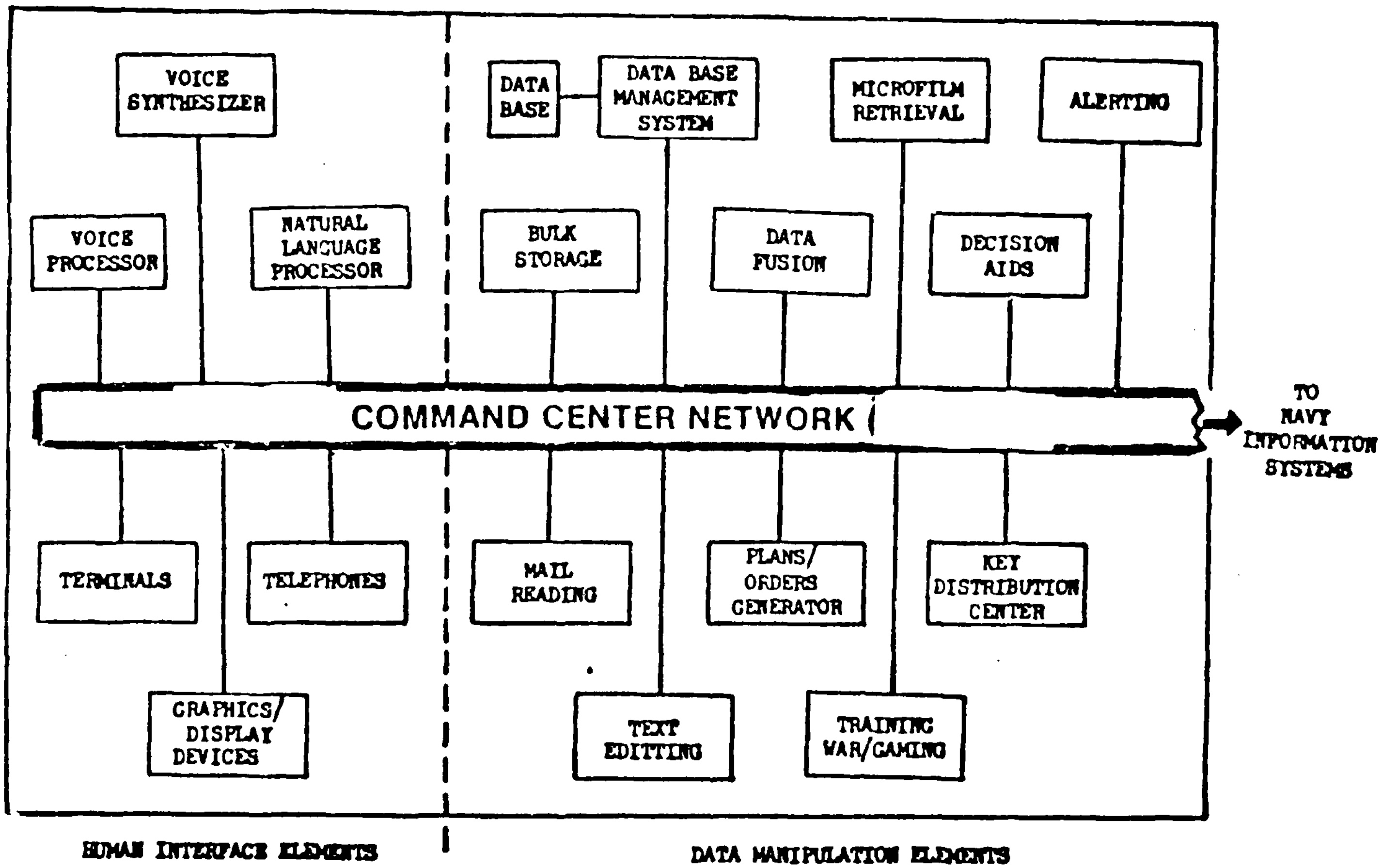


FIGURE 10: CCN AND FUTURE C2 SUBSYSTEMS

Source: Command Center Network....Backbone of Future Command and Control

13) Training/War Gaming,

14) Key Distribution Center,

15) Text Editing,

16) Detailed Plans/Orders Generator, and

17) Electronic Mail. [Ref. 20]

Appendix F addresses each of these additions in more detail.

This rather exhaustive list of possible applications involving the CCN demonstrates the potential of the program. The CCN can be the solution to many of the interoperability problems with current and envisioned Navy C2 and information subsystems.

Dr. Glen Allgaier, the Project Manager of the CCN, envisions the Command Center Network of the 1990's as it appears in Figure 11. These ideas are, for the most part, technological subsystem additions to the CCN. If these additions become the focus of CCN, then application and employment of this system <u>may</u> become secondary to further demonstrations of the capacity to interconnect subsystems.

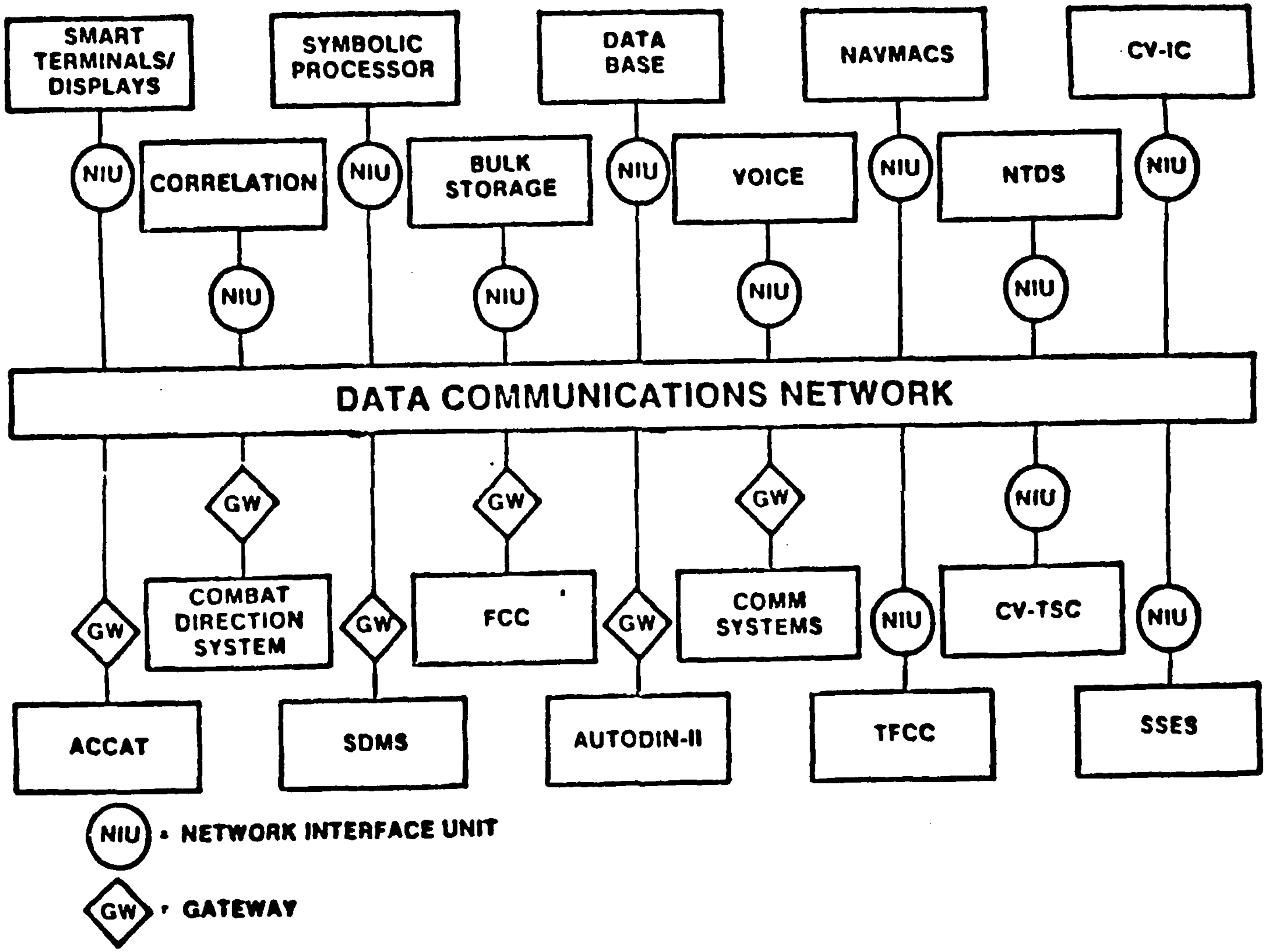


FIGURE 11: COMMAND CENTER NETWORK OF THE 1990's

Source: Command Center Network....Backbone of Future Command and Control

VII. SUMMARY, CONCLUSIONS, AND RECOMMENDATION

The Command Center Network (CCN) project is designed to provide the vital linkage between Navy C2 and information subsystems. As the "bus" connecting the various subsystems envisioned to encompass the Task Force Command Center (TFCC), CCN should be proved compatible with both shipboard and shore-based installations. Throughout this thesis, the complexity of certain areas of the project, such as protocol and software development, has been stressed in order that one can appreciate the enormous task facing the project developers.

The Advanced Research Projects Administration (ARPA) has recognized the multitude of current and future research and development issues which must be confronted. This concern is reflected in the fact that ARPA insisted on the simplification of the project. Some of the concerns of ARPA are expressed in the following comments.

> Since many of the R & D (research and development) issues will surface in the future and because the total future requirements are not totally predictable, this testbed must be designed with a number of basic features. These are as follows:
>
> a. Flexibility: It must be possible to modify various aspects of the testbed (XDM) to incorporate both hardware and software changes. This requires that the design be modular and well-layered, with clearly defined interfaces. In addition, the software must be properly structured and well-documented;
>
> b. Versatility: The testbed must be capable of providing as broad a range of services as possible. This requires that the full spectrum of capabilities be provided at all levels, conditioned by cost and risk of development. Those features which were either high risk or required excessive cost are to be left for later investigation as R & D issues . . . . [Ref. 21]

Within the ARPA community, debate continues as to whether the CCN project is attempting to accomplish more than will ever be feasible. There is little disagreement that interconnecting and ultimately internetworking Navy C2 and information subsystems is necessary to support a commander in a TFCC. However, serious, basic questions remain to be answered by the CCN project. Some of these are expressed below.

1) Is CCN a network in the truest sense of the concept or is it merely a modified tree-structure (i.e. if one of the portions of the bus is disconnected or damaged, are the other nodes still interconnected and functional)?

2) If necessary, can the network be reconfigured at short notice without losing capabilities?

3) Is CCN being developed to support a commander embarked on a large platform (i.e. an aircraft carrier or large amphibious assault platform), or can it be configured to adapt to smaller platforms?

4) If CCN can be installed on smaller platforms, will one network installation be able to query another so that a commander can maximize flexible command structure, as well as command mobility from platform to platform?

5) In the event of emergency, can communications external to the locally installed CCN transfer sufficient data on request to allow continuity of command and control?

6) If, as projected, the CCN intends to interconnect so many subsystems, what kind of access and security structure must be built into the network?

7) What kind of training, maintenance and other support logistics will be involved in a CCN installation?

These are but a few of the questions that arise concerning the Command Center Network project. Some of these questions will be answered during testing of the prototype installation at NOSC.

One must note that it is possible to sink all our technological advances into a system that allows neither flexibility nor versatility. This possibility remains the challenge not only of the Command Center Network, but also for all future C2 developments.

It is recommended that the CCN project re-examine the focus of the XDM demonstration. Is the focus on the need for interconnecting C2 and Navy information subsystems to support a tactical commander? If this development is to maintain its credibility in relation to this goal, then demonstration should be further simplified. Towards the goal of simplification, the author recommends starting with the interconnection of two existing subsystems, like NAVMACS and NTDS/DTS, and demonstrating the following:

1) interconnection of all subsystem functions is feasible from a technological standpoint,

2) while interconnection is made, there is no degradation of the individual subsystem's ability to perform in their individual networks, and

3) if part of the CCN is damaged or destroyed, the remainder of the interconnection is still functional.

Once the above capabilities are demonstrated, subsystems should be added one at a time and the testing or demonstration repeated with the aforementioned attributes remaining principal concerns. In this manner, the author feels that the goal to interconnect subsystems in support of the tactical commander will remain the focus of this worthwhile project.

APPENDIX A

NAVMACS USER/SERVER PROCESS

This Appendix traces the user/server process utilized in one of the major subsystems connected by CCN. The NAVMACS user/server process is presented in this manner to demonstrate and trace how these processes interact in the CCN. It must be remembered that in the overall CCN configuration, one user process can address one or many server processes. For the purpose of simplifying the discussion, the user process will be discussed only in reference to the NAVMACS server process.

The server process reads and distributes messages the NAVMACS processor normally sends to its printer. The messages are distributed to all interested users on the CCN and may be filtered on the basis of message type of content (messages will not be filtered on content in the initial CCN and the only types of messages allowed will be "all" or "RAINFORM"). The messages are also sent to the NAVMACS printer. Users connect to this process via the user program described in the next section. The server program maintains a well known socket via a LISTEN call to TCP. When TCP informs this process that an attempt has been made by a foreign process to connect to the NAVMACS processor, this process will establish the type of traffic the user is interested in seeing by awaiting a CONTROL function code from the user process. If the NAVMACS processor is being held off due to lack of buffer space, the server will return a function code to the user (this will not take place in the initial CCN). If the TYPE is acceptable, an entry will be made in a connection table showing:

1. the user is connected;
2. the TCP connection name;
3. the type of traffic the user is interested in seeing; and
4. a parameter to filter on; for example, a subject or header;
5. an on/off indicator. [Ref. 22]

An acknowledgement signal (ACK) will be returned once the connection entry is placed into the connection table.

As each message is received from NAVMACS, the server process will do some preliminary manipulating before the message is distributed. At first, the process will delimit the NAVMACS message by looking for the start of message - end of message (SOM-EOM) characters. Once these characters are found, the NAVMACS processor will be held off by dropping the ready line to a low voltage. At this time, the server process will convert the baudot characters to ASCII. After the conversion, the server process will attempt to send the message to the printer. The printer is shared so it may be busy with text from some other CCN user. If the printer is busy, the server process will attempt to store the message for the printer process to retrieve later. If no storage device is available, the server process will hold the message with the NAVMACS processor held off until the printer becomes available (in the initial CCN there will be no mass storage capability). The user will be informed via a CONTROL function code of the state of the NAVMACS processor. The server process will start searching the connection table for interested users. If the message is RAINFORM formatted, it will be *converted to* CCN *format.* The CCN format is a standard tactical data format that makes information of this kind available to all CCN users. This format will be defined so that users who want to make use of, for example, DTS LINK-11 data can do so by recognizing the data format. Each connection will be examined in turn to see which ones have interest in the current message. If a connection is interested, the server process will turn it on by making an entry into the connection table. Once the connection table has been searched, the message processing is finished.

The message will then be sent via TCP to each connection which is turned on. TCP will indicate the success/failure of the sent text. If a message cannot be delivered to a user, the user will be informed of the missing message by a CONTROL function code. Once TCP has accepted the message for transmission, the buffer space *will be reused* by turning the NAVMACS processor back on (by raising the voltage on the ready line high). The connection table will be reinitialized, i.e. all connections will be turned off.

This process will respond to a CONTROL by updating the connection table entry for this user and ACKing the CONTROL. If a CLOSE is received from TCP, the connection table will be updated by deleting the entry for that connection.

The user process will supply NAVMACS messages of a specified type to a user process in the CCN. The user must supply a START command to this process to initiate the connection to the server. The user must supply a TYPE parameter indicating the type of messages it wants to receive. The user will be able to have the messages filtered on subject or header. The user process should establish that the TYPE given is valid. The user should supply a parameter indicating whether the messages are to be delivered to a specified user buffer or filed on mass storage. The buffer size at this time is to be on the order of 2K bytes allowing enough room for an average size NAVMACS message. Once the server process responds to the attempt to connect (TCP will signal open), this process will send a CONTROL (type) packet. This CONTROL will be ACKed. This process will keep a status indication which reflects the state of the process at any given time (OPEN issued, type sent, etc.). Once the connection is established (the CONTROL was accepted), this process will inform the user and supply a buffer for incoming messages.

Once messages arrive, they will be filed away or given to the user if so indicated. In either case, the user will be signalled when the messages arrive. The user process will delimit the messages by looking for SOM and EOM. The user will also be informed if the server has indicated that the NAVMACS processor is being held off or messages were lost via CONTROL function codes. This process will handle error messages sent by the server and will respond to a user command to stop at which time it will issue a CLOSE to TCP.

The user process will also maintain a well known socket for foreign processes to access for the purpose of allowing third party transfer. Thus, the user process must be prepared to ask for login information and process it. Once the login is processed, this process will expect to receive a START command as if it were being controlled by a local user.

Terminal users will be asked to indicate what kind of NAVMACS messages they are interested in. This process will establish that the type is legitimate and that the user is authorized. If the type is not legitimate, the user process will return an error indication to the user. If the type is legitimate, this process will then connect to the server via TCP. This process will keep a connection status indicating the successive states:

1. OPEN sent
2. TYPE sent
3. connection established

Once the connection is made, this process will establish the type of messages desired. Once the connection is established, this process will supply a buffer of 2K bytes for incoming messages. Once messages arrive, the user will be signalled of a newly arrived message. The terminal user can then ask for the following actions:

1) print the message on the terminal,
2) print the header on the terminal, and/or
3) file the message away.

The user will be given a time-out period in which to process the message before it is lost. This is necessitated by the fact that the user process must continually read the connection to process the header. A new buffer will be allocated and a new NAVMACS message solicited. The user may change the type of messages being delivered. The user process will send CONTROL (type) with the new type to the server. The server will return an ACK for the CONTROL. The user process will inform the user if the NAVMACS processor is being held off or messages were lost. The terminal user can stop the process at any time via the DONE command. [Ref. 23]

APPENDIX B

TCP SPECIFICS

The following is a list of DOD required minimum user calls for any TCP:

1. OPEN (local port, foreign socket, active/passive)
   The terms port and socket are defined in the specification. If the indication is passive, this call is equivalent to LISTEN. If there is no foreign socket specified, the LISTEN would respond to any attempt by a foreign process to connect.

2. SEND (local connection name, buffer address)
   Here TCP is given the responsibility of delivering the named buffer to the destination. All errors due to transmission over the network are expected to be recovered by TCP.

3. RECEIVE (local connection name, buffer address)
   Here TCP provides a buffer for data arriving over the network.

4. CLOSE (local connection name)
   TCP will delete the connection.

In the initial CCN, SRI's TCP will run in the NIUs and TOPS20 TCP will run in the DEC 20/40.

For users on the CCN to take advantage of the services offered by the CCN user processes, the following commands are defined:

1. START
   This is the command that the user gives to the user process to initiate the connection to the C2 subsystem. The START command will allow for the following parameters:

   - name of the C2 subsystem source and destination

   - user name and password (for login purposes)

   - file name (appropriate to the operating system being used). This file name will provide the user process the necessary information to store incoming data if the user so desires.

   - filter filed: contains an indication to the user process of data that will be used as a filter of incoming data. For example, the NSM might want to have LINK-11 data filtered on content (i.e. all surface tracks and all air tracks).

- type filed: specifies the type of data the user is interested in. For example, in NAVMACS this would be used to indicate RAINFORM type messages only are desired.

2. DONE
When the user is finished with the C2 subsystem, it issues this command to close the TCP connection and terminate the user process and close all files.

3. CHANGE
This command allows the user to change the filter or type specified in the START command.

4. TRANSMIT
This command will include a buffer address and byte count of data to send to the C2 subsystem.

5. RECEIVE
Indicates that the user is ready to process input from the C2 subsystem. If data is available when this command is given, the user process will give a buffer to the user, otherwise, the request will be queued for processing when data does arrive. [Ref. 24]

The user process will pass on to the user responses which TCP gives to the user process: connection established, data on connection, connection closed, etc. The exact nature of the responses depends on the TCP interfaced. For example, SRI's TCP might give one of 12 responses to the user. These twelve responses are appropriate for the commands: START, DONE, TRANSMIT and RECEIVE. The user process will also return a success/failure indication for a CHANGE command.

The user and server communicate with the following function codes:

1. ACK
The C2 packet was accepted.

2. CONTROL
The control function code might be used to change filters or to inform the user of changing events such as: printer OK or DTS ready.

3. ERROR
   This can be sent at anytime and will contain a field with an error code such as: printer down, DTS down, printer out of paper.

4. EOF
   This marks the end of a file (in the CCN, a NAVMACS message, for example, is considered a file). [Ref. 25]

APPENDIX C

NTDS/DTS FUNCTIONS

The protocols contemplated for inclusion in the user process of the CCN are designed to access the following subsystem functions:

- deliver track data from the DTS computer to interested users;
- (deliver track data from users to the DTS computer);
- (require users to login or processes to identify themselves);
- prevent transmission over CCN of track reports containing no change in data fields;
- (deliver tracks based on content (air tracks to some users, surface tracks to others, etc.));
- signal the user when tracks arrive from the DTS computer;
- employ a multi-addressing scheme in order to deliver the same tracks to several users;
- (inform users of success/failure of tracks sent to the DTS computer);
- convert track data from binary to ASCII;
- (store track data for later retrieval);
- (allow NSM to have tracks sent to a third party and filter on subject or content, i.e. the NSM can change the addressee list) (for the purpose of insuring that certain processes on the CCN get all air track information or surface track information etc.);
- (convert ASCII/binary);
- (convert to/from CCN format); and
- (allow an option to disable the default of receiving all tracks and receive only certain tracks based on the same filter). [Ref. 26]

The functions in parentheses are those which will be implemented by future protocol development. Those functions which are not in parentheses will be demonstrated in the CCN prototype.

APPENDIX D

NTDS/DTS USER/SERVER PROCESS

This Appendix outlines the user/server process as it relates to the NTDS/DTS subsystem of the CCN.

The server process reads and distributes LINK-11 track data which the DTS computer normally sends to a NTDS computer. The tracks are sent to all interested CCN users and may be filtered on content (in the initial CCN no content filtering of tracks will be performed). Users connect to this process via the user process described in the next section. The server process maintains a well known socket via a LISTEN call to TCP. When the user process establishes a connection with this socket, it will send a CONTROL packet to the server. Included in the CONTROL message will be an indication of the user's desire to filter tracks. The filter may be based on content or track number (in the initial CCN, there will be no opportunity to filter tracks). If the CONTROL is acceptable, an ACK will be returned by the user process. A data structure will be maintained by the server consisting of one entry per connection containing the following type of information:

1. user name
2. TCP connection name
3. filter/no filter indicator
4. parameter to filter on
5. on/off indicator

As tracks arrive from the DTS, the server process will record the track numbers (and other necessary information) in order that it may prevent transmission over CCN of track reports containing no change in the data fields. Due to the nature of the data receiving process, truncation of track data may occur before the duplicate is detected. If the newly arrived track is not a duplicate, a routine will be invoked to convert the binary track data into a CCN ASCII format. This format will be the standard CCN format for tactical data which will allow processes anywhere on the CCN to recognize and make use of the information. Once this conversion is done, the connection table will be scanned and each connection turned on/off depending on the appropriateness of the track. If a connection indicates some filter, the track will be analyzed to see if the information is pertinent to that user (in the initial CCN, this will not be done). The track will then be sent to all connections that were turned on during the scan. The connections will then be turned off and the process will repeat. If TCP closes the connection, the entry in the connection table will be deleted.

While connected, the user will be able to send codes to the server process. One code will be a CONTROL (filter) so the user can change the filter information on its connection. The server process will respond by inserting the new information into the appropriate entry in the connection table and ACKing the CONTROL. The user may also send new track information to the DTS computer (in the initial CCN, this will not be implemented). The track data will arrive in CCN ASCII format and the server process will convert this data to the binary format required by the DTS computer. The converted data will be sent to the DTS computer. ACK will be returned to the user process to indicate the success of delivering the data to the DTS.

The user process will supply LINK-11 track data to users on the CCN obtained from the DTS computer. The user must supply a START to this process to initiate the connection to the server process.

Included with this command will be an indication of whether the user wants the data delivered to a buffer or to mass storage (in the initial CCN, only the KL-20/40 will have mass storage). The user can supply data to filter the tracks on if he so desires (in the initial CCN, no filter will be allowed). The user process will issue an OPEN to TCP and wait for TCP to signal that the connection is open. The user process will then send CONTROL with the filter parameters if there are any. The server will ACK this CONTROL. When track data arrives, the user process will store it in a file, if the user desires, or deliver the data to a specified buffer. In either case, the user will be signalled that LINK-11 track data has been delivered.

The user can supply commands to the user process by requesting the following actions:

1. stop the process (DONE).
2. change the filter (CHANGE).
3. send track data to the DTS (TRANSMIT).

(in the initial CCN, only the DONE will be allowed)

If the user wishes to stop the process, the user process will issue a CLOSE to TCP and await TCP's CLOSED response before exiting. If the user wishes to change the filter, a CONTROL code will be sent to the server process along with the new filter data. The CONTROL will be ACKed by the server. If the user wishes to send track data, it must issue a TRANSMIT along with the buffer address and byte count.

It is the user's responsibility to insure that data is in CCN ASCII format. Once data has been sent, the user process will keep an indication that it is awaiting acknowledgement for that data. ACKs will come from the server indicating the success of delivering the tracks to the DTS. [Ref. 27]

The user process will also maintain a well known socket for foreign processes to access for the purpose of initiating third party transfer. Thus, the user process must be prepared to ask for and process login information.

APPENDIX E

TCP FUNCTIONAL AREAS

The following is a presentation and description of the four major functional areas associated with supporting TCP specifications:

1. Buffer Manipulation: To make the best use of available space, the system caters to varying message lengths by using dynamic free pools of varying lengths, and the resultant frequent garbage collection. Several free pools must be accessed using the standard system calls (with attendant overheads) for every buffer request, transfer, and return to free space. This overhead has been found to be heavy for TCP and BBN has proposed that fixed buffer sizes should be used for any particular association, together with reassembly direct into the user buffer. This approach removes the need for dynamic pools of varying length, and also reduces the number of transfers required.

2. Table Searches: Associated with multiplexing connections. As TCP allows an association to be any unique source-destination address combination, no short address or index conventions are adapted for it. It is possible then to consume cycles in chaining down the connection list for each incoming message. Some intelligent hashing of addresses will reduce this overhead.

3. Arithmetic Computations: TCP employs a large sequence number space to avoid the possibility of two letters in transit ever having the same sequence number. This requires 32 bit arithmetic to be performed both in generating a new sequence number and in testing that an incoming letter lies within the current window. Check-summing requires the same computational support.

4. Choice of Language: Although the desire for a clear implementation makes the use of a high level language attractive for implementing TCP, known overhead of system implementation languages is around 30-50% and can be much higher. Such overhead can be quite bearable for many applications. For a communications driver such as TCP, performing many actions per message, and even a number of actions per character, the cumulative effect is significant. [Ref. 27]

APPENDIX F

FUTURE CCN APPLICATIONS

This Appendix amplifies the information concerning contemplated additions to the CCN system listed in Chapter VI.

a. Voice to Natural Language Convertor: This element would convert voice to text replacing a keyboard with a microphone. It may include the digital equivalent of a voiceprint which is associated with each user for security purposes. All verbal queries would be routed to this converter. The output of this element would then go to a Natural Language Processor.

b. Natural Language Processor: This processor takes textual queries which are structured much as one would pose them in a "natural" manner, parses these queries, and restructures them in a format which is understood by a data base management system. Development of a natural language processor is currently being done at NOSC in the Command Center Information Subsystem (CCIS) project. These structured queries are then routed to the DBMS (Data Base Management System).

c. Text-to-Voice Synthesizer: This element converts text to voice, providing the commander with verbal reports feedback to his queries. It basically provides the inverse of items (a) and (b) described above.

d. Sophisticated Graphics and Display Devices: These will provide all of the advances in display technology which include: color, conics, shading, etc., all with a very simple user interface, responding to queries such as: "Show all enemy ships as a flashing red light". They will include large screen displays for purposes of briefing large groups. They will range from simple alphanumeric displays to high resolution displays capable of displaying maps and photographs with a zoom capability to show increasing detail. Real-time video displays will be interconnected via the CCN to provide conferencing and other capabilities which require such a display capability.

e. Multiple Terminals: It is projected that between 100 and 200 user terminals will be interconnected via the CCN. Each terminal may have its own software and special purpose applications programs for utilizing the information in the CCN-connected data.

f. Telephones: Each user will have a telephone, headset or microphone for communications with other CCN users.

g. Bulk Storage: This would be a less expensive way of storing vast quantities of data employing disks, tapes, or some other suitable bulk storage media. Generally, the information stored would be slowly changing in nature, not frequently accessed. Examples of the data might be operations orders, ship characteristics, photographs, maps, manuals, etc. The bulk storage may be distributed across a number of machines and would have a file management system for accessing and updating the stored data.

h. Data Base Management System: This subsystem would manage all of the stored data, including that in bulk storage and microfilm retrieval. Capabilities of this subsystem would include:

   1. Provision for significant changes in the structure of the data which are transparent to the user applications programs.

   2. Adaptability to new data base technology.

   3. Generalization of the interface which can be mapped into various data base structures.

   4. Adaptability to interactive queries.

   5. Robustness in maintaining data base integrity.

   6. Optimization of queries to minimize response time.

   Thus, all queries for data from one of the C2 subsystems would be processed by this DBMS which would manage the organization of those data bases which can be managed, query the data base, and return the response to the inquirer. The DBMS would include a knowledge of the structure of the file management system associated with bulk storage. The DBMS would also process all queries to the data bases of other information systems and route these queries to the appropriate system.

i. Data Fusion: This facility would take all pertinent data available and provide an integrated picture of all platforms (friendly, enemy, neutral) within the region of interest to the commander. In addition to the sensor reports, this fusion would consider such factors as political climate, positioning of platforms, personality of commanders, known fuel supplies, behavior of platforms, etc. The Tactical Situation Assessment (TSA) project at NOSC is currently developing such a subsystem. This element would be highly interactive with the DBMS described in (b).

j. Alerting: This element monitors the data traffic on the CCN as well as the content of the data bases to determine if the commander needs to be alerted. Sighting of a submarine that is a threat to a high value unit, priority message traffic, low fuel supplies, and limited defensive capability due to inoperable aircraft are examples of alerting conditions.

k. Microfilm Retrieval: This element would provide access to data which is stored on microfilm, and convert the microfilm image to the proper format for transmission via the CCN.

l. Decision Aids: These may consist of a number of elements, performing distinct tasks as aids to the commander. This subsystem takes as an input the mission and objectives of the commander and provides optional courses of action to achieve them. An interactive capability will be provided, allowing the commander to insert his own preferences, review the explanation trace of the options suggested by the machine and insert his own options. Once an option is selected, this element summarizes the pros and cons, and provides a probability of success and losses. Examples would include:

1. Optimum platform positioning to maximize sensor coverage.

2. Optimum platform position for detection of enemy submarines attacking a high value unit.

3. Optimum weapon allocation against enemy target.

4. Optimum routing of aircraft to achieve a successful strike mission.

m. Training/War Gaming: One of the significant benefits of the CCN, given the subsystems described to this point, is that it can be easily used (and learned to use) by a human while simultaneously providing access to all of the pertinent data affecting the mission of the ship. Software can thus be developed in conjunction with the decision aids of (f) to allow war gaming within the actual environment of the ship. This element thus performs in a manner similar to that currently available in computer chess games of increasing complexity, and allows a commander and his staff to exercise in preparation for the real environment.

n. Key Distribution Center: Data exchanged between subsystems will sometimes be of classification levels which should not be made available to all users of the CCN. In some cases, the commander may also desire to conference with certain staff members without allowing access to the conversation by others. There will thus be a facility which distributes a crypto "key" to each qualified participant on a message by message basis.

o. Text Editing: This element would contain all of those features which assist the commander in preparing orders or directives or reports which require a Textual format. The HERMES or MSG systems developed for the ARPANET are examples of the functions provided. Typical features would include addressee lists, special formats, spelling correction and editing capabilities.

p. Detailed Plans/Orders Generator: This element will be used in the generation of detailed plans for implementing a given option selected in (f) above. In fact, a great deal of interaction is expected between these two elements as the option generator must understand the details of implementation in order to assess the probability of success.

q. Electronic Mail: Similar to the Navy message system, it allows for message exchange between intraship users.

Although all of the above functions are ones which can be implemented within a single ship environment, the CCN also provides the opportunity for the afloat commander to tap directly, via communications systems, the data bases which may be resident on shore or in systems which are connected to other networks such as AUTODIN-II. [Ref. 28]

LIST OF REFERENCES

1. Allgaier, G. R. and Stout, C. C., Navy Command and Control Using Local Networks, paper prepared for the Naval Ocean Systems Center, 1980.

2. Ibid., pp. 1-3.

3. Allgaier, G. R., Command Center Network....Backbone of Future Command & Control, Naval Ocean Systems Center, 1980.

4. Ibid., p. 2.

5. Ibid., pp. 6-10.

6. Naval Ocean Systems Center, LCCN Related Technology Status, report prepared by WESTEC Services, Inc., San Diego, California, July 1979.

7. Allgaier, G. R. and Stout, C. C., Navy Command and Control Using Local Networks, paper prepared for the Naval Ocean Systems Center, 1980.

8. Naval Ocean Systems Center, CCN Interface Requirements, report prepared by WESTEC Services, Inc., San Diego, California, 12 October 1979.

9. Bennett, C. J. and Hinchley, A. J., Measurements of the Transmission Control Protocol, paper presented at the Computer Network Protocols Symposium held at Liege, Belgium, February 1978.

10. Ibid., p. 3.

11. Ibid., pp. 14-17.

12. Bennett, C. J. and Hinchley, A. J., Issues in the Interconnection of Datagram Networks, INDRA Note 637, Department of Statistics and Computer Science, University College London, 1977.

13. Neer, M., CCN Software Functional Descriptions (MOD. #2), Naval Ocean Systems Center, 25 January 1980.

14. Ibid., pp. 7-10.

15. Ibid., p. 11.

16. Ibid., p. 13.

17. Naval Ocean Systems Center, *C2 Modules for Local Command Center Network Demonstration*, report prepared by WESTEC Services, Inc., San Diego, California, July 1979.

18. Ibid., p. 12.

19. Allgaier, G. R., *Command Center Network....Backbone of Future Command & Control*, Naval Ocean Systems Center, 1980.

20. Ibid., pp. 19-24.

21. Naval Ocean Systems Center (Code 8242), *ARPA Comments on LCCN Statement of Work*, Memorandum Serial 824-M-101, 9 May 1979.

22. Neer, M., *CCN Protocol Descriptions*, Naval Ocean Systems Center, 9 April 1980.

23. Ibid., p. 13.

24. Naval Ocean Systems Center, *LCCN Information Flow and Protocol Requirements*, report prepared by WESTEC Services, Inc., San Diego, California, May 1979.

25. Ibid., p. 19.

26. Neer, M., *CCN Protocol Descriptions*, Naval Ocean Systems Center, 9 April 1980.

27. Ibid., p. 30.

28. Allgaier, G. R., *Command Center Network....Backbone of Future Command & Control*, Naval Ocean Systems Center, 1980.

BIBLIOGRAPHY

Allgaier, G. R., Command Center Network....Backbone of Future Command and Control, Naval Ocean Systems Center, 1979.

Allgaier, G. R., Communication Protocols in a Local Computer Network, Naval Ocean Systems Center, 27 July 1978.

Balmain, K. G. and Jordan, E. C., Electromagnetic Waves and Radiating Systems, Prentice Hall, 1968.

Bennett, W. R., Schwartz, M. and Stein, S., Communication Systems and Techniques, McGraw-Hill, 1966.

Clark, D. D., Relevance of ARPA Programs to LCCN, Paper Prepared for the Naval Ocean Systems Center, August 1978.

Naval Electronic Systems Command, Tactical Data Display System Data Book--AN/USQ-81(V) (OUTLAW SHARK), by Naval Tactical Data Programs: Space Systems Division, Lockheed Missiles and Space Division, 15 November 1979.

Naval Electronics Systems Command, System Specification for the Navy Modular Automated Communications Systems "A+" (NAVMACS "A+"), NAVELECSYSENGCEN Document nr. 1 F001-00-AC, 12 March 1974.

Tranter, W. H. and Zeimer, R. E., Principles of Communications: Systems, Modulation and Noise, Houghton Mifflin Company, 1976.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, Virginia 22314	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, California 93940	2
3.	Professor J. M. Wozencraft (Code 39) Department of Command, Control & Communications Naval Postgraduate School Monterey, California 93940	3
4.	LCOL J. W. Johnson (Code 39) Department of Command, Control & Communications Naval Postgraduate School Monterey, California 93940	1
5.	Associate Professor F. R. Richards (Code 55 RH) Department of Operations Research Naval Postgraduate School Monterey, California 93940	1
6.	Associate Professor G. T. Howard (Code 55 HK) Department of Operations Research Naval Postgraduate School Monterey, California 93940	1
7.	Dr. Glen Allgaier (Code 8242) Naval Ocean Systems Center San Diego, California 92216	2
8.	LT G. K. Hamilton 1412 Pinewood Road Jacksonville Beach, Florida 32250	4

DATE

ILMED

8