RESTRUCTURING THE NAVY TACTICAL DATA SYSTEM

LCDR G. S. GARMON, USN

Fleet Combat Direction Systems Support Activity, San Diego

Abstract – Restructuring the Navy Tactical Data System divides the identifying system requirements into functional areas that are further subdivided into smaller independent segments called tasks. Each of the individual tasks is designed to be isolated without interdependency on other tasks. This will allow the modification or replacement of the individual tasks as operational requirements change without affecting other tasks. Each functional area will be supported by a group of tasks that meet the specified operational requirements for that function. Important to the development of the universal program library for all Navy Tactical Data System ships, is the capability to maintain a single source library of tasks written in a host transferable language and a support program that will generate a system tape when a particular NTDS ship class configuration is defined.

Index Terms – Restructured NTDS, tasks, operational requirements, universal program library, host transferable language, system tape generator, ship class configuration.

I. BACKGROUND

Some twenty years ago, the Navy Tactical Data System (NTDS) was developed to computerize the storage and communication of tactically significant data within and between major Navy warships. This recognized the limitations inherent in manual target tracking and assignment to weapons, and it resulted in reduced response times, increased accuracy, and greater data storage capacity. Various other systems have been developed which perform essentially the same function as NTDS. These are sometimes referred to as Command and Control Systems (C&C) and/or Combat Direction Systems (CDS). Systems of this nature have been or are being installed on a wide variety of hardware systems and software structures. [1]

The current NTDS program is a collection of large program segments called modules, consisting of instructions, data and control information, generally pertaining to a single operational function or capability such as tracking. There is significant interdependency between modules with a large dependency on local data stores thus generating an almost intolerable level of Inter-Module/Inter-Computer (IMIC) message traffic. This modular architecture presently requires the modules to be large, functionally orientated, entities resulting in rather heavy internal message traffic that extends the pre-1960 hardware (CP-642) to its limits. Additionally, as a former Chief of Naval Operations has stated, "... as improved weapons are introduced to insure readiness against evolving threats, Naval operations and related tactical command, control, and coordination become increasingly complex. There is a continuous requirement to reduce command decision time. This requirement places demands to improve responsiveness in data processing and to achieve greater automation of information processing in the Navy's Command and Control System. Incremental improvements in processing and display capabilities have resulted in a variety of diverse subsystems which are interfaced in a complex network to form the present tactical Command and Control System." [2]

II. NAVY TACTICAL DATA SYSTEM DEVELOPMENT

It is generally recognized that the optimal system development strategy first identifies the problem (operation requirements to be levied on the Tactical Data System in the case of NTDS) and then delineates the effective employment of available resources in an optimal manner maximizing or taking the greatest advantage possible of each individual feature of the various system components. Then it is reasonable to expect that in developing a new architecture for the Navy's NTDS that (1) operational requirements would be identified, (2) system design to adequately meet or supersede those stated requirements would be developed, and (3) the optimal hardware and software combination would be identified to implement that system design. Unfortunately, Naval tactical system development has not always evolved in this manner. For the most part, the software system design and architecture have been driven by the ever changing hardware procurements with a lack of coordination between hardware and software systems engineers. The recent assignment of "Combat System Engineers" within the Naval Sea Systems Command organization offers some promise for the future in that they will provide the coordination in planning and design.

No necessity to be mutually compatible or interoperable has been stated at the outset of development of many of the Navy weapons and combat systems due to the limited and narrowly defined

HIERARCHY OF /NAVY\ **SYSTEM REQUIREMENTS**

```
                    /\
                   /NAVY\
                  /MISSION\
                 /CONCEPT OF\
                /OPERATIONS  \
               /MISSION AREAS AND\
              /  RESPONSIBILITIES  \
             /REQUIRED OPERATIONAL   \
            /  CAPABILITIES (ROC's)    \
           /                            \
          / TOP LEVEL REQUIREMENTS  /TLR\ SHIP CLASS
         /COMBAT SYSTEM DESIGN REQUIREMENTS/CSDR\COMBAT SYSTEM
        /COMBAT SYSTEM OPERATIONAL DESIGN/CSOD   \COMBAT SYSTEM
       /SYSTEM OPERATIONAL SPECIFICATIONS/SYSTEM   \NTDS SYSTEM
      /   SYSTEM OPERATIONAL DESIGN/FUNCTION         \
     /PROGRAM PERFORMANCE SPECIFICATIONS/SUB FUNCTION  \
    /   PROGRAM DESIGN SPECIFICATIONS/ELEMENT           \
   /PROGRAM DESIGN DOCUMENTATION/SUB ELEMENT             \
  /_____/TASK_____\
```
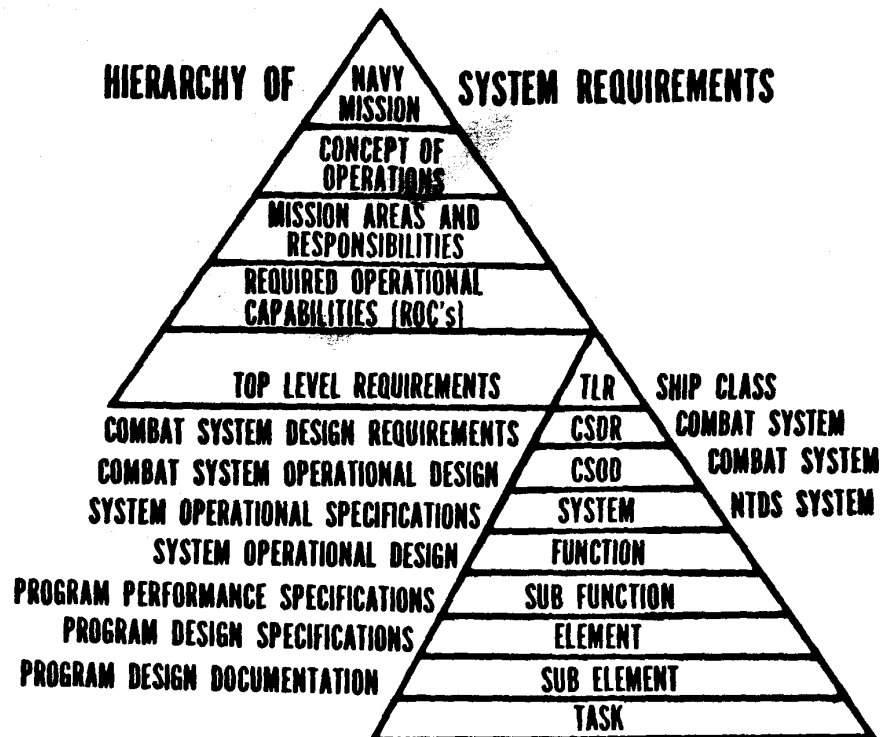
FIGURE 1

operational requirements they were designed to fulfill. Efforts to improve software development policy and practices for weapon system applications are underway within the Department of Defense. [3] It is important at this junction to recognize the reality of the environment in which tactical applications software is most often developed.

In selecting the CV (aircraft carrier) as one of the lead ship classes for the Restructured NTDS Program, the formal identification of the CV Combat Direction System requirements, priorities, and boundaries represented the first step in the program development. These elements of the CV Combat Direction System have never been formally specified. A preliminary step was to define the CV Top Level Requirements (TLR) with the assistance of several offices within the Naval Material Command. This document, in draft form, is undergoing review within the Navy at this time and is supported by some and challenged by others. The challengers are sincere in that they believe this CV Top Level Requirement to be too demanding to be satisfied by "off-the-shelf" systems or the existing capability inherent in the Navy

RATIONAL DEVELOPMENT APPROACH	REAL WORLD DEVELOPMENT APPROACH
IDENTIFICATION OF OPERATIONAL REQUIREMENTS	REQUIREMENTS ARE VAGUE OR NON-EXISTENT
DEVELOP SYSTEM DESIGN TO MEET OR SUPERSEDE STATED REQUIRE-MENTS	SYSTEM DESIGN IS CONSTRAINED BY "DESIGN-TO-PRICE" AND "OFF-THE-SHELF" HARDWARE ITEMS
IDENTIFY THE OPTIONAL HARDWARE AND SOFTWARE COMBINATION THAT IMPLEMENTS THE SYSTEM DESIGN	HARDWARE CONSTRAINTS ACCEPTED AS GIVEN WITH SOFTWARE EXPECTED TO MAKE THE SYSTEM WORK

FIGURE 2

Department. In short, new aircraft carrier construction such as the USS VINSON would fail to meet the requirements based on presently envisioned threat as set forth in the CV TLR. As an example, the TLR might state that surveillance capabilities should extend to a 1000 nm radius based on presently envisioned threats. In reality, this would constitute a top level statement of operational requirements.

A major goal in the Restructured Design Development is to allow traceability of combat design system elements from the overall Navy mission down to the lowest level of task as shown in Figure 1 by making use of requirements languages, such as the University of Michigan's URL/URA, and exercising various techniques of structured program development. [4] Employing automated tools such as these do not guarantee success; however, they can provide some degree of assurance that the NTDS computer program produced by a Navy Fleet Support Activity actually reflects the approved requirements and operational concepts and not simply the efforts of a programmer trying to optimize memory and timing. The high level documentation will also aid in establishing hardware baselines and the associated lead times for program improvement.

III. RESTRUCTURED NTDS ARCHITECTURE

The principle of decomposition of large program modules into small independent tasks is the goal to be achieved in the restructured concept. The procedures that constitute each task are isolated without interdependency on other tasks. This provides the capability to modify individual tasks, as operation requirements change, without affecting other tasks.

In the Model 4.1 NTDS architecture, a "task" is a unit of work which when coupled to other units of work results in the execution of a required operational capability. For example, the ability to engage targets is a required operational capability for most NTDS ships. Reception by the software of an "engage" stimulus from the command level (e.g., depression of an "engage" button) results in the scheduling of the appropriate tasks to notify other force/system operators to initiate additional software stimuli to comply with the "engage" request/order. Alternatively, the "engage" stimulus, when used at lower levels of command results in the scheduling of the appropriate tasks to complete the desired type of engagement. [5] The task communicates, as necessary, via the common data base or the executive as shown in Figure 3.
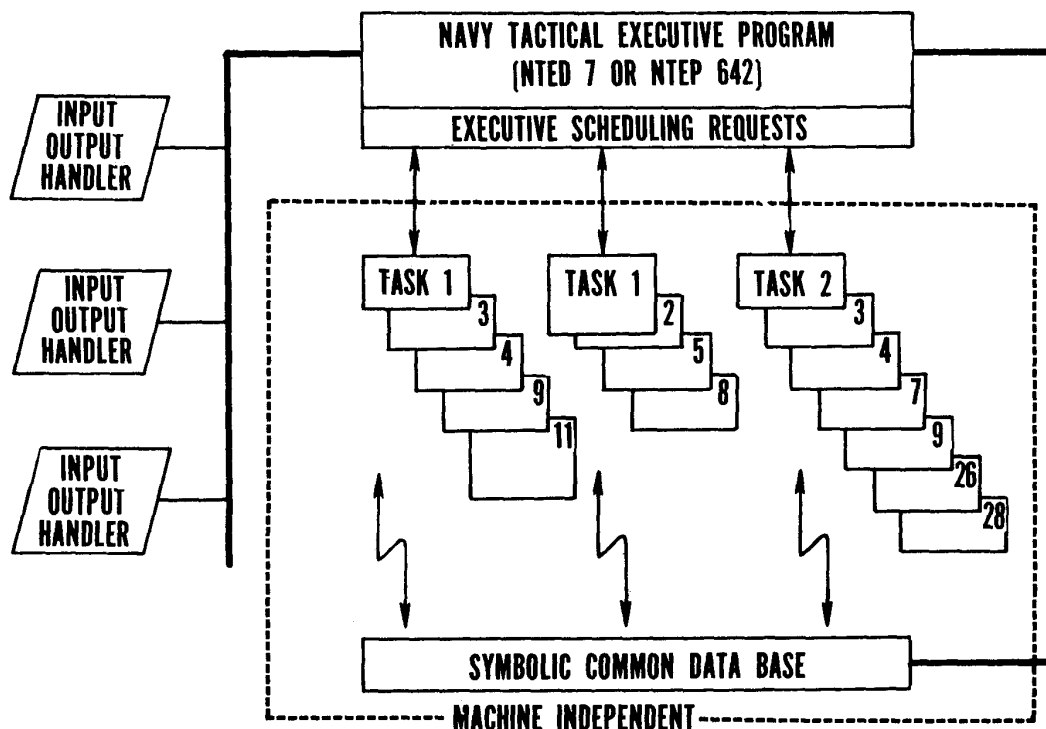

FIGURE 3

A function can thus be defined by assembling a group of threaded tasks that meet the specified operational requirements for that function. The development of a capability to maintain a "single source library" of tasks written in a host transferable language and a support program that will generate a system tape when an NTDS ship class configuration is defined are key elements of the development of the universal program library for all NTDS ships.

The executive program initializes task execution, schedules tasks, controls input/output, provides executive service requests to tasks, and is responsible for interrupt and error management processing. The input/output handlers are responsible for all task interfaces with external equipments and with other on-line computer systems. The common data base is the repository of symbolic source data relative to all such items as track stores, ship parametric data, environmental data, and buffer areas for data input and output, and also may hold program routines used by many task modules, such as arithmetic algorithms.

IV. CONCLUSION

It is apparent that many of the present day problems plaguing the combat systems at sea can be attributed to an absence of overall design guidance related to unstated or previously unknown operational requirements. Unfortunately, this is the real world situation and tactical software development must adapt. The Restructured NTDS development program was required to accept the constraints of today, but has been able to partially minimize the impact of these constraints on future developmental Naval tactical software.

Several benefits can be realized during program development and ultimate life cycle support once the universal user library for all NTDS ships is installed and strict configuration management procedures are implemented. Any NTDS program can be built from the library without modification to the executive or any other task (independent of computer used). During life cycle maintenance, new conceptual operations or hardware changes will affect only discrete tasks or subtasks within a functional area, and will not "ripple" other functional boundaries as in the past modular program. The utilization of small tasks provides the ability for the library to grow in any direction. The cost of change is reduced since each task is a "stand alone" entity and once tested is reliable in all configurations. For new ship classes, or new sensors or weapons, maximum use can be made of "on-the-shelf" tasks unless performance requirements are modified.

It is estimated that a minimum of 60% of the program library will be transferable (common) between all NTDS ship classes, and that for similar classes such as carriers and cruisers, that as much as 80% to 85% will be carried over.

The net effect of this will be to reduce the procurement costs of software for the NTDS program to between 10% to 20% of new software costs using existing programming techniques. It will not be necessary to continually go to outside vendors for total NTDS program procurements since only the new capabilities need to be designed, coded, tested, and placed on the universal library.

The relatively small independent tasks and sub-tasks of the restructured architecture should allow the Navy Tactical Data System to expand into the world of microtechnology and firmware applications with relative ease. It is the intent of the Restructured NTDS Program to provide the Navy with an ability to adapt NTDS changes quickly and easily to rapidly changing fleet operational requirements and associated technical advances in hardware and weapons systems in order to meet the challenge of the future.

REFERENCES

[1] Functional Description for the Restructured NTDS Programs, Fleet Combat Direction Systems Support Activities, pg. 2-1, 1 Mar 1978.

[2] ADM James Holloway, III, USN, "Concept Paper for Tactical Flag Command Center (TFCC)", Chief of Naval Operations, Nov 1977.

[3] B. C. De Roze and T. H. Nyman, "The Software Life Cycle - A Management and Technological Challenge in the Department of Defense", *IEEE Trans. Software Eng., Vol SE-4, No. 4,* pp. 309-318, Jul 1978.

[4] D. T. Ross and K. E. Schoman, Jr., "Structured Analysis for Requirements", *IEEE Trans. Software Eng., Vol SE-3, No. 1,* pg. 12, Jan 1977.

[5] Restructured NTDS System Control and Data Base Working Group, Fleet Combat Direction Systems Support Activities, San Diego and Dam Neck, Working Papers, Aug 1978.

[6] Plan of Action and Milestones (POA&M) for Restructured NTDS Program, Fleet Combat Direction Systems Support Activities, Apr 1977.