DATA EXCHANGE, SYSTEM ARCHITECTURES AND STRUCTURES

BY

J. S. HILL, MA, DPHIL, CPHYS
(*Defence Research Agency Maritime Division, Portsdown*)

ABSTRACT

The concept of a Combat System has been introduced relatively recently into warship design. The need for it has been brought about by the increase in power and complexity of the weapon and sensor systems and the need to integrate them more effectively. Its realization has been made possible by the improvements in the technology for electronic transfer of tactical data and control information between them and with the Combat Management System.

Combat System design has progressed from an *ad hoc* integration of weapons and sensors by the Combat Management System project to the ongoing development of a hierarchic system for the Future Frigate. This article presents some of the ideas on data exchange, system architecture and structure which have contributed to this evolution.

Introduction

The concept of a Combat System, as distinct from a collection of weapons and sensors, is relatively new to warship design. Its emergence, and the recognition of the need for an overall Combat System design[1], has been due, at least in part, to the steady automation of functions which previously could only be carried out by men.

Several factors have contributed to the need for this transition and at the same time have made it possible. The first, obviously, is the increasing pace of warfare and the consequent reduction of safe reaction times, sometimes to below the threshold of human capability. A second is the introduction , and the subsequent decline in cost and increase in power, of the digital computer. But a third, and less obvious, factor is the advance in the technology for moving data between computer-based systems.

This article explores the impact of developments in data exchange on Combat System architectures and structures, and speculates about possible future developments.

The Beginnings of Data Exchange Technology

In retrospect, there have always been recognizable Combat Systems aboard warships. The sensors, weapons and communications were there (telescopes, cannon and flags); the Combat Management System consisted entirely of men; and the primary data exhange mechanism between the Combat System components was the human voice.

The weaponry and, with the introduction of radar and sonar, the sensors improved rapidly in range and accuracy, but developments in data exchange were relatively slow to take off. A beginning was made with the use of synchro circuits, especially for stabilization and for repeaters, but the major spur to advance was the introduction of computers.

The first shipboard computers were bulky and low powered by today's standards. Primarily, however, they were expensive to buy and to program. It was vital to concentrate all data processing into one big machine, and this meant that data had to be imported and exported electronically. A plethora of data interfaces and transfer systems appeared, and the computer's interface with each of its weapon and sensor 'customers' was a point to point link, the subject of a separate negotiation with the relevant contractor. The Combat Management System Computer became the data exchange and management centre of the ship, and Combat System design was implicitly delegated to the Combat Management System project. It is small wonder that this led to difficulties in specification and production and to shortfalls in performance.

The Coming of the Local Area Network

The development of data highways or buses began early in the 1970s. At first they had little impact, partially because they were complicated and expensive, but also because they were an unknown quantity and no one was sure how and where to use them. Indeed, their main advantage was seen by some as a means of reducing the weight of copper cable aboard ship.

In time the big shipwide buses gave place to the simpler concept of the Local Area Network or LAN. These LANs did not attempt to provide every conceivable interface to their users, nor to provide sufficient data bandwidth to meet the entire needs of a major ship, and so were easier to use and less expensive to procure.

Their first use in the Royal Navy came with the use of DEF STAN 00-18 Part 2 (MIL STD 1553B) networks in the UPHOLDER Class of submarines[2], in which the LAN was used to multiplex the historic point to point networks on to a single data handling system.

The decision to install a Combat System Highway in the Type 23 Combat System, this time using DEF STAN 00-19[3], meant that a LAN would be used in a rather more complex procurement and operational environment. It also brought the opportunity to introduce a radical change in the philosophy of data exchange[4]; instead of using actual, or even virtual, point to point links, all data

is broadcast on the LAN by its originator and each recipient selects only the data of interest to it. For this to work, a consistent and complete set of interface and data handling standards were required. These were developed and all users of the LAN were mandated to implement them—a situation made possible largely because most of the Type 23 systems were either new or subject to substantial modification.

The Impact of the LAN on System Architecture

Up to this point there had been no doubt about the architecture of the Combat System. The Combat Management System was at the centre of a star network of point to point data links, as shown in FIG. 1. No other arrangement was possible.

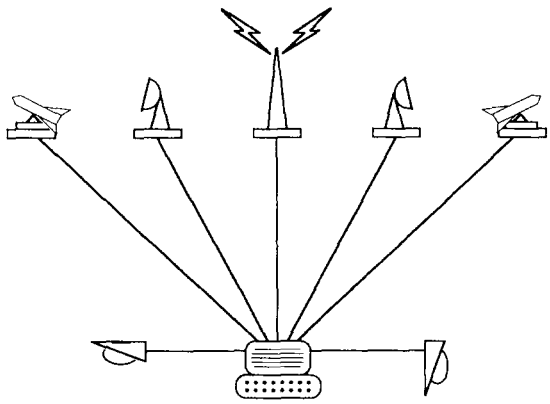

FIG. 1—THE COMBAT MANAGEMENT SYSTEM AT THE CENTRE

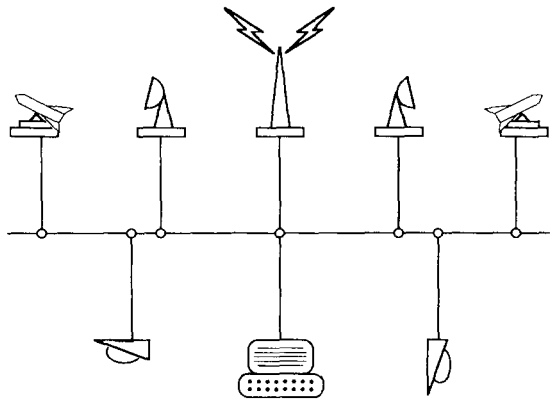

FIG. 2—THE DATA EXCHANGE SYSTEM AT THE CENTRE

The change in data handling technology makes it possible for what had previously been peripheral systems to pass data directly from one to another instead of through the Combat Management System. They thus become equal members (peer systems) with the Combat Management System so far as data handling is concerned. Several important consequences stem from this.

Firstly, the data exchange system replaces the Combat Management System as the central element of the Combat System for data handling (though the latter retains, at least for the time being, its central role as the manager of Combat System resources). This is illustrated in FIG. 2. The resilience of the data exchange system must be greater than that of any of the systems it serves, and this must be reflected in its procurement specification.

Secondly, Combat System design and data management can no longer be swept up into the Combat Management System, but have to be recognized as separate topics in their own right. While the need for ship and fleet data managers has been accepted, other consequences of this have not yet been fully assimilated by the MOD.

Thirdly, it becomes possible to consider alternative system architectures which may permit increased efficiency and reduced cost of the Combat System, and also ameliorate specification and procurement difficulties.

Architecture and Structure—a Digression

The words 'architecture' and 'structure' are sometimes used as though they were synonymous. To avoid confusion, their meanings as used in this article are defined below.

The architecture of a system comprises the rules defining the way it is designed and built. It does not define (though it may be influenced by) the intended use of the system. Several data and system architectures will be explored in later sections.

The structure of a system defines what it is made of; how the various components are incorporated in accordance with the chosen architecture. This article will have little to say about specific structures, though illustrative alternatives may be discussed.

To use the obvious analogy of building architectures and structures, a given building may be designed in Norman, Palladian or modern glass and concrete style; this defines a range of properties from the overall proportions of the building to the shape of the arches, but says nothing about the use of the building. On the other hand, the structure may be a cathedral, a railway station or a block of flats; this says much about what it contains, both in the arrangement of rooms and the purposes for which they will be used, but any one can be (in principle) constructed in any architecture. Some architectures will be more appropriate than others, of course; a Palladian municipal swimming-bath might be thought to be over-kill, while a glass-and-concrete cathedral might raise hackles in certain quarters.

However, the analogy cannot be pushed too far, since it is possible for a system design (and structure) to present different architectures from different viewpoints. It is reasonable for a structure to have a data exchange architecture which is quite different from the control or management architecture. Indeed, it is possible for a data exchange system to present different architectures at the control, the data flow and the recovery levels. In general, however, systems which are architecturally consistent will be simpler and safer than ones which are diverse.

NATO Architectures

In 1983 NATO's Industrial Exchange Group 5 endorsed a paper on Ship System Integration[5]. This document was the result of a long and sometimes difficult debate.

The final version defined three basic data handling architectures (referred to in the main as structures), though the definitions are more widely applicable. The paragraphs below are a direct transcript.

3. Three architectures or structures in data handling are currently in existence or considered for implementation. These are:
   (a) a centralised structure which comprises one (or a small number) central functional processor carrying out all data handling compilations and control functions;
   (b) a federated structure which comprises a number of subsystems which are autonomous in data handling to a certain degree, but can still be controlled to a certain degree by one (or small number) controlling computer;
   (c) a distributed structure which divides the data handling load via the processors in the system without using fixed or central points.
4. These three structures have been identified as basic structures. In practical situations systems are hybrid and form for that reason elements of a continuum spanning these basic structures.

The paper goes on to expand the definitions above. In so doing, it recognizes the inherent dangers of a centralized architecture, the need in designing in a federated architecture to ensure that the federating function is itself decentralized (or protected by redundancy), and the difficulties of data management in distributed systems of more than limited complexity. Finally it accepts that any

practical design will be a hybrid between these limiting architectures, since 'The point at which any ship-wide architecture falls in the continuum of paragraph 4 reflects a trade-off between the easier management and perhaps higher vulnerability on one hand and reduced vulnerability but more complex data management on the other.'

FIGS. 3, 4 and 5 attempt to portray these three architectures in diagram form.

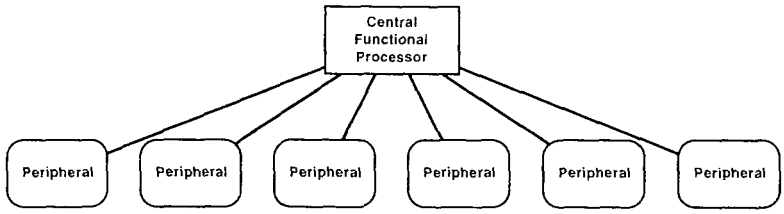

FIG. 3—CENTRALIZED ARCHITECTURE

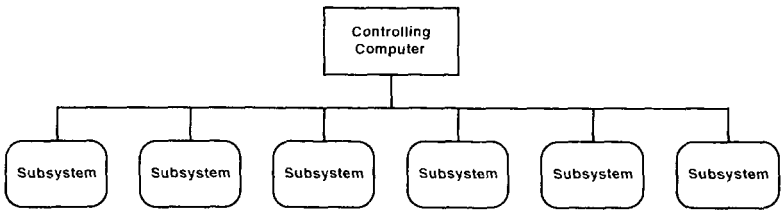

FIG. 4—FEDERATED ARCHITECTURE

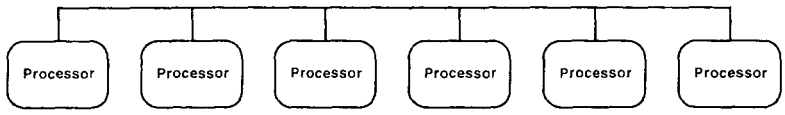

FIG. 5—DISTRIBUTED ARCHITECTURE

Some Existing Systems in Terms of the NATO Architectures

It is instructive to consider the Type 23 Combat System in terms of the NATO definitions. This can be done:

(*a*) from the point of view of the overall Combat System design;

(*b*) from the point of view of the data communications philosophy (the Combat System Highway).

The overall design is clearly federated. The Combat Management System is the management centre of the Combat System, which has only a limited functionality if the Combat Management System is inoperative. Some groupings of equipment can, however, continue to function in the absence of the Combat Management System.

The Type 23 Combat System LAN, DEF STAN 00-19, can be regarded as having centralized management and error recovery. The Highway Controller/Health Monitor offers transmission opportunities to the user systems in turn, and keeps an eye on the state of the LAN for missed or corrupted messages, loss of redundancy, incipient failures, etc. Since it has these vital roles, it is duplicated. However, data exchange is fully distributed; no single system is responsible for organizing the data flow, or determining who transmits what, when or to whom. In fact, messages are broadcast bearing a content address, and potential users of the data select which messages they wish to receive at any given time by this address and reject the remainder.

Both the control and data flow are centralized in the LAN currently used in submarines, DEF STAN 00-18 Part 2 (MIL STD 1553B), in which the Bus Controller commands one system to transmit and another to receive data in a predetermined pattern. Additional protocols have been provided in the submarine implementations to enable the controller to find out which systems are ready to transmit data, and to whom, so that it can issue the appropriate commands.

The Fibre Distributed Data Interface (FDDI) is a LAN which is emerging as the most likely contender for the next generation of installations, both above and below water. It is distributed in both control and data exchange, and also in some aspects of management (though health monitoring and reporting can be regarded as federated).

More Recent Concepts

The NATO statement represented the state of the art at the time. It was based, however, on the presumption of a single shipwide data transmission system serving relatively unsophisticated systems (as can be inferred from the reference to 'one (or a small number) central functional processor'). Developments since then, and in particular

(*a*) the emergence of relatively inexpensive LANs of reasonable capacity and resilience, and

(*b*) the widespread deployment of microprocessors in the systems they serve,

make it possible to look at other architectures. These overlap the NATO definitions to some extent; they must not be regarded as mutually exclusive. The following sections present some alternatives.

Flat Systems

In a Flat structure every separately identified component is at the same level, and can communicate directly with every other component. This implies that a direct connection can be established between any two components; the actual connections invoked would be a very small subset of those possible. SHINPADS[6] was conceived along these lines, and it is at this situation that the ISO 7-layer reference model for open system interconnection is primarily aimed. FIG. 6 portrays a flat architecture. While there is some subdivision of the overall system into subsystems, these have very little practical value; the data transmission system (shown in FIG. 6 as a ring) connects all the components as equals (peers).

It might be helpful to the reader in this and the following diagrams to think of the upper left grouping of five components as representing a Combat Management System (CMS), the centre grouping as representing a radar subsystem (R) and the right-hand grouping as representing an Electronic Warfare subsystem (EW).

Actual data flows can be effectively point to point, as shown in FIG. 7 (as is typical of RN submarine LAN-based systems) or broadcast as in the Type 23. Either example can be regarded at the combat/tactical data system level as flat. This architecture has a close resemblance to the NATO distributed architecture. However, there is no reason why any of the NATO architectures, considered as management structures, cannot be mapped on to a flat data communications architecture.

A significant advantage of the flat architecture is that data transfer delays can be minimized.

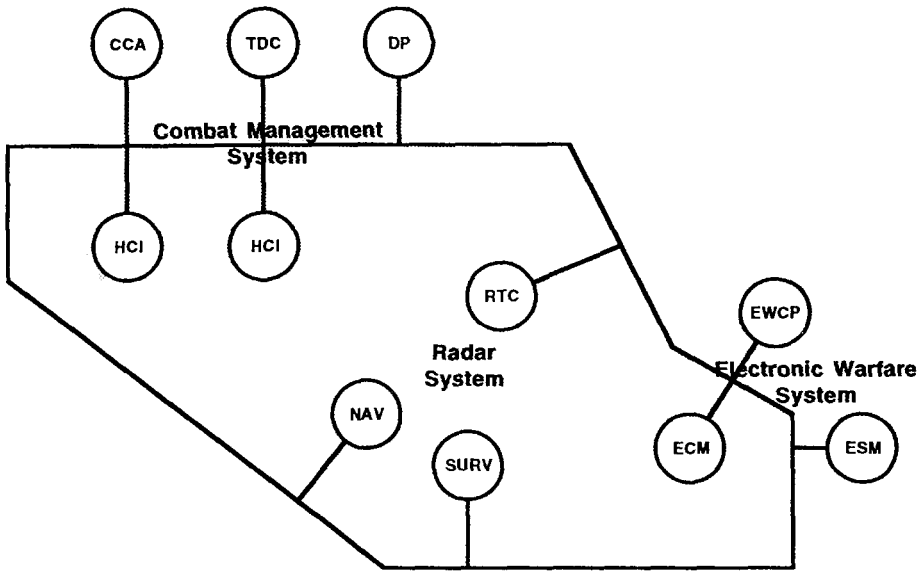

FIG. 6—A FLAT RING-CONNECTED SYSTEM

CCA:Captain's Combat Aid
DP:Data Processor
ECM:Electronic Countermeasures
ESM:Electronic Support Measures
EWCP:Electronic Warfare Control Processor
HCI:Human Computer Inferface (=operator position)
NAV:Navigation radar
SURV:Surveillance radar
RTC:Radar Track Combiner
TDC:Tactical Data Compiler

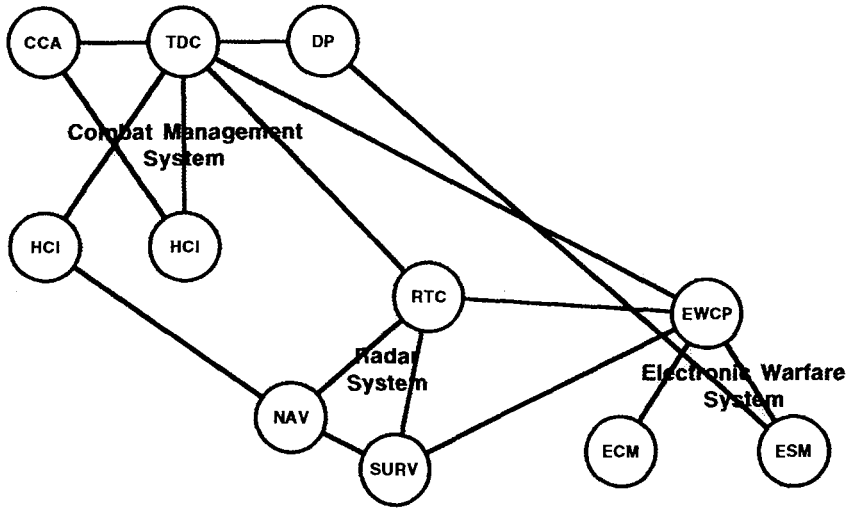

FIG. 7—POSSIBLE DATA FLOWS IN A FLAT SYSTEM

Integrated Systems

The term 'integrated' has been coined to designate an architecture in which each component interacts with a distributed data management system as though the latter was a part of the component. The data held by this management system will be contributed to, and probably held as master copies, by other components. It is the responsibility of the data management system to transfer data as required, to make and maintain local and back-up copies and to ensure data integrity generally. All of this will be hidden from the functional components.

FIGS. 8 and 9 illustrate the architecture. In FIG. 8, each component has a local interface to the data management matrix; it sees this as the interface to a data management system which contains all the data it requires and to which it exports all the data it generates. This interface completely conceals the other components with which it is interacting; the only effect they have is that entries in the data base are changed by some external agency.

The notion of systems and system boundaries is virtually absent in this architecture; each component sits in effective isolation, like a plum in a pudding, with no direct contact with any other component, as in FIG. 9.

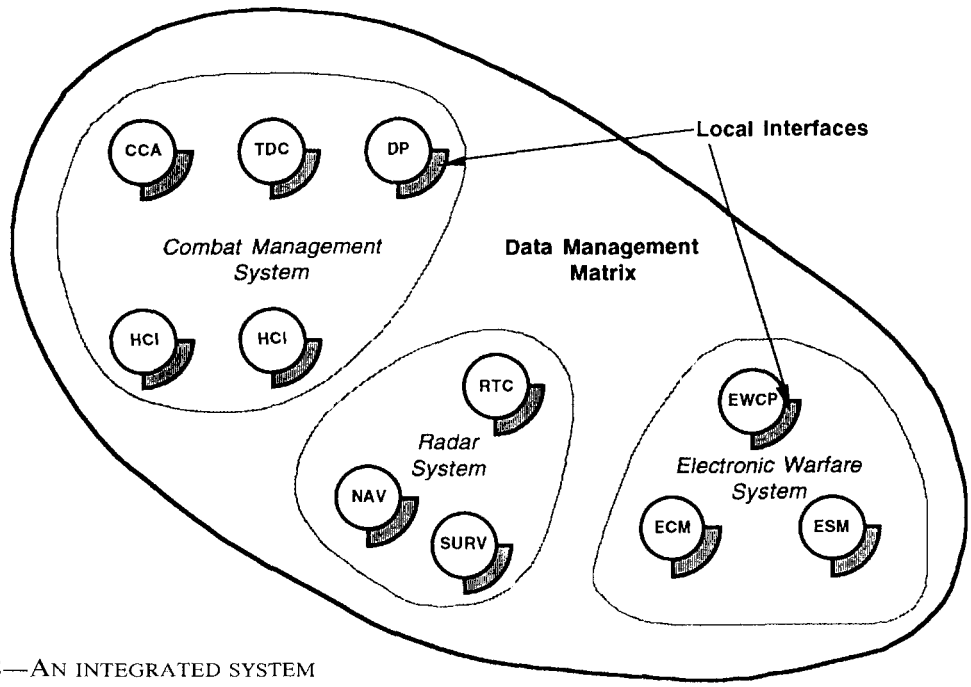

FIG. 8—AN INTEGRATED SYSTEM

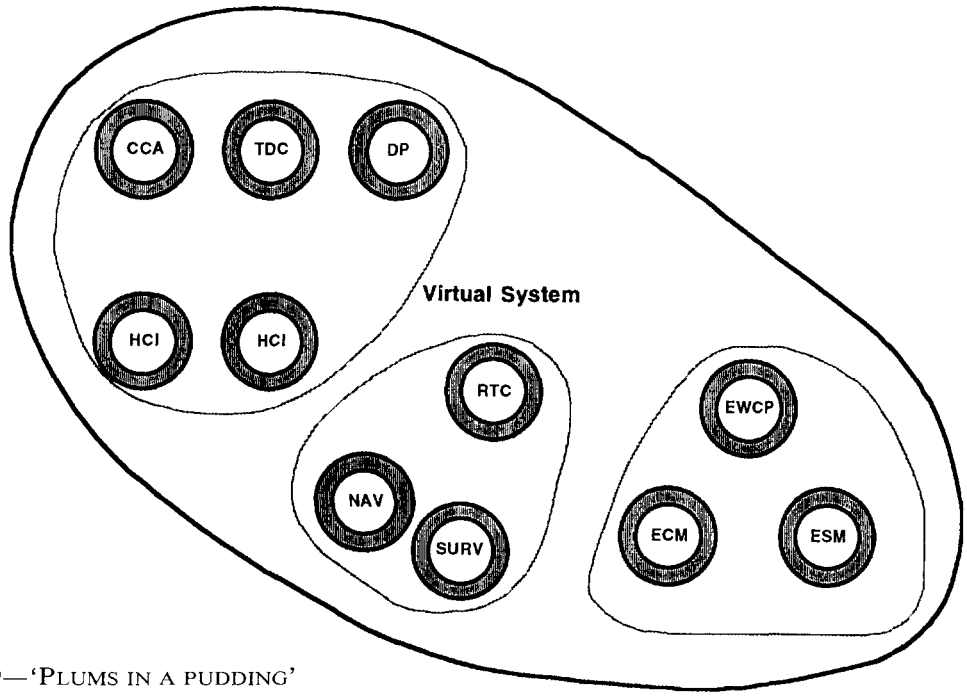

FIG. 9—'PLUMS IN A PUDDING'

The data flow patterns have to sort themselves out within the data management system; this by no means removes the necessity to ensure that it can cope, but it does pass the responsibility to the authority providing the data management system. A means whereby external events can stimulate components must

be provided, either by special data locations which are polled at regular intervals, or by a separate mechanism.

The advantage of this approach is that it simplifies overall system design to a remarkable extent in so far as data is concerned, although problems remain in the passing of directives to other functions. A major disadvantage is that it requires every subscriber to use the same (very complex) data management interface. The difficulties in this are not merely technical; a highly developed procurement management organization would be required to ensure that all components conformed strictly to the rules. This would present problems in the current procurement environment (especially if international procurement is envisaged).

This architecture has much in common with the Virtual Machine architecture which was proposed for the NFR 90 Combat System. It was seen as a very risky venture, and was in fact one of the main factors in the abandonment of that project.

It is questionable whether any system of this type is sufficiently developed to be recommended as the way ahead at the Combat System level for some time to come, if ever, despite its many potential advantages. The DIAS project at ARE, however, has shown that it is feasible to build integrated systems, and at least one recent Combat Management System proposal has offered a virtual machine architecture. (In this case the procurement management problems were containable, because all the hardware and software within the virtual machine structure had a common supplier).

Hierarchic Systems

The discussion so far has been entirely in terms of constructing systems from components, and it has been implied that components are equipments such as individual radars or missile launchers. But what if they are themselves systems? Enlargement of the concept of a component at once leads to hierarchic architectures since the idea can be applied recursively in the manner of Dean Swift's fleas. This opens the door to massive simplification of system structures.

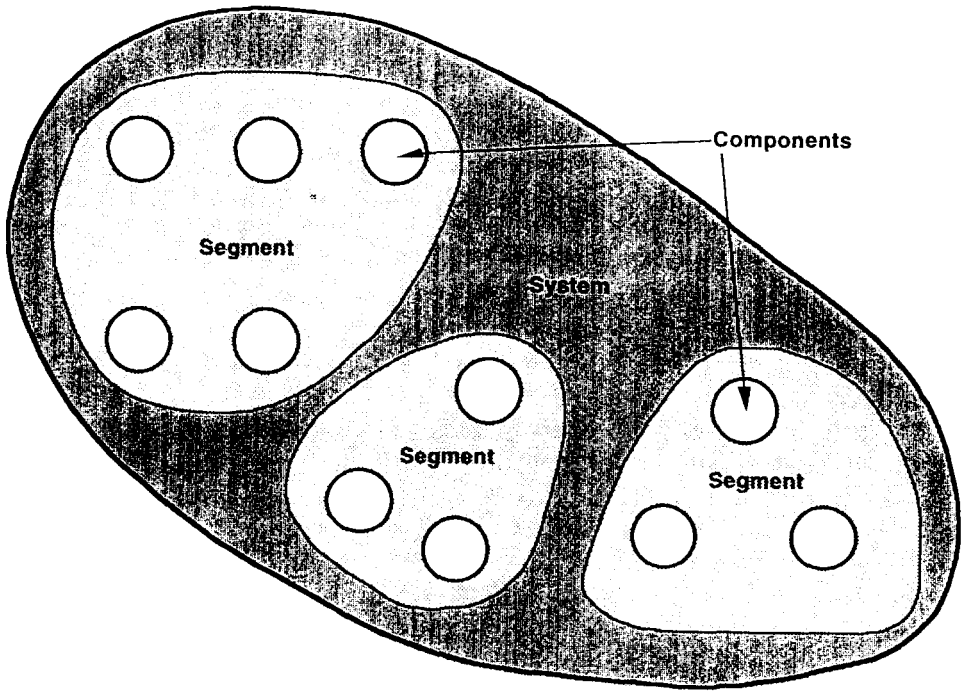

FIG. 10—SYSTEMS WITHIN SYSTEMS

The idea was hinted at in the discussion of flat architectures. Suppose that the radar subsystem suggested there be recognized as a discrete subsystem and treated as a component of the Combat System? The individual radars (and possibly a radar track combiner) may still be procured from different contractors, but the responsibility for welding the radar segment (to use a term which has been adopted from NFR 90 terminology) into a useful Combat System component can be delegated to a suitable procurement authority.

FIG. 10 illustrates this concept. The binding of components into systems is very strong, and it is vital to recognize several points if a successful system is to be achieved.

Firstly, the task of Combat System design begins with Structured Analysis of the operational requirement, and continues with allocation of the functions identified to segments. An interface specification between each segment and the Combat System (i.e. with the aggregation of other segments) must then be prepared. This must be done as soon as possible; nothing can be achieved if the segment specifications are incomplete or inconsistent.

Secondly, the allocation of functions to segments is not a trivial task. Not only must it create a logical and consistent structure, but it must also identify all the data flow paths and assess their impact on performance—for example, data flows that imply urgent action should be contained within a segment, rather than flow across segment boundaries. As a general principle, data should be confined within segments as far as possible, and only exported in refined form.

Thirdly, design may be complicated by the presence of mandated equipments or even systems, which reduce the designer's options. At the Combat System level, for example, systems could be grouped into a Radar segment, a CIWS segment, an Electronic Warfare segment, etc., or they could be grouped into an AAW segment, an ASW segment, etc. The mandation of a system such as NAAWS might force a decision to choose the latter option. Some of the functions currently regarded as part of the Combat Management System segment might be relocated in other segments, their new positions depending on the grouping chosen.

Fourthly, particular attention must be paid to the implementation of the interface between a segment and the LAN connecting it to its peers. A simple communications bridge (through which messages are relayed unmodified) is unlikely to be adequate. The reasons why gateways will be required are worth exploring.

An underlying principle of the hierarchic architecture is that each segment at a given level in the hierarchy conceals its internal structure from the other segments, and data is exchanged between defined interfaces. Routing of data between its interface port and its internal components (the next lower hierarchical level) is the segment's private responsibility. This has the advantage that structural changes, and to some extent functional changes, within one segment have negligible impact on the others. It also means that responsibility for local data exchange is delegated to the segment, instead of leaving it as a global problem. The segment should solve its internal data handling problems to meet its own objectives and to recognize its own constraints. The internal data exchange arrangements of an Electronic Warfare segment, for example, might be quite different from those of a Sonar segment, because one might need to pass small amounts of data very quickly, whereas the other may have to handle large amounts of data but at a more leisurely rate.

The interface gateway must be able to translate between the segment's private data standards and those of the LAN that serves all the other segments at that level. It must be able to filter the data flowing within the segment and export only that which is of interest to other segments, and it must ensure that incoming data arrives at the correct location in the segment without the external