OPEN COMBAT SYSTEMS

BY

J. MILES, PHD
(DERA Sea Systems)

This is an edited version of the article that first appeared in the January 1998 issue of the Journal of Defence Science.

ABSTRACT

Naval combat systems need a computing infrastructure to provide support for the interlinking set of system-specific software. The heavy computing requirements prevent all functionality from being expressed in software at present, but it should soon be possible to introduce a common computing capability for most of a combat system. The complexity of a combat system invites the scope and rigour of a technological demonstrator programme to examine risk, to enable potential users to relate to the implications of the underlying applied research, and industry to gain from participation and to prepare for eventual procurement.

The demonstrator programme proposed is built upon a layered model approach which identifies four areas for investigation—computing infrastructure, integration mechanism, functional integration and functional performance—and these are treated in detail. The programme will integrate a number of combat system components and will carry out experiments designed to measure the performance of infrastructure and integrated applications.

Introduction

A naval combat system includes sensors, communications, actuators and user interfaces interlinked by a complex set of system-specific software applications. The processing, storage and communications needs of this software require a computing infrastructure to provide support. At present, all of the functionality cannot be expressed in software because of the very large computing demands, for example of sensor signal processing. But general purpose processors are increasing in capability and offer the prospect of extending the scope of the interlinking soft system, and it is now possible to consider a common computing infrastructure for most of the combat system which is independent of the application software.

Given the number and complexity of sub-systems and individual software applications that make-up a combat system, the way in which integration is carried out is important in engineering terms and in its effect on operational performance, factors that have been recognized to some extent in current system developments by instituting standard practices for agreeing and validating data exchange between sub-systems. Nevertheless, there remains scope for methods that carry fewer overheads, allow quicker and easier changes to be made, and are not peculiar to the combat system engineering community. With the aim of presenting the many issues relating to combat systems in a structured way so that all parties can see clearly where and how they are to be addressed, a layered model approach has been adopted, as shown in (FIG. 1). This model relates directly to the generally accepted physical layers of a combat system but, more importantly, it separates those layers to which key issues such as the use of COTS (Commercial-Off-The-Shelf) and functional performance apply. The integration mechanism is brought out as a separate layer in this article. The model has been used to propose a structure of four areas for investigation which together will lay the basis for future combat systems that are more efficient and have lower demands on personnel.

While the issues to be addressed are many, some relevant work is already being carried out, and there is scope for collaboration. The levels of risk and uncertainty are appropriate for a Technology Demonstrator Programme

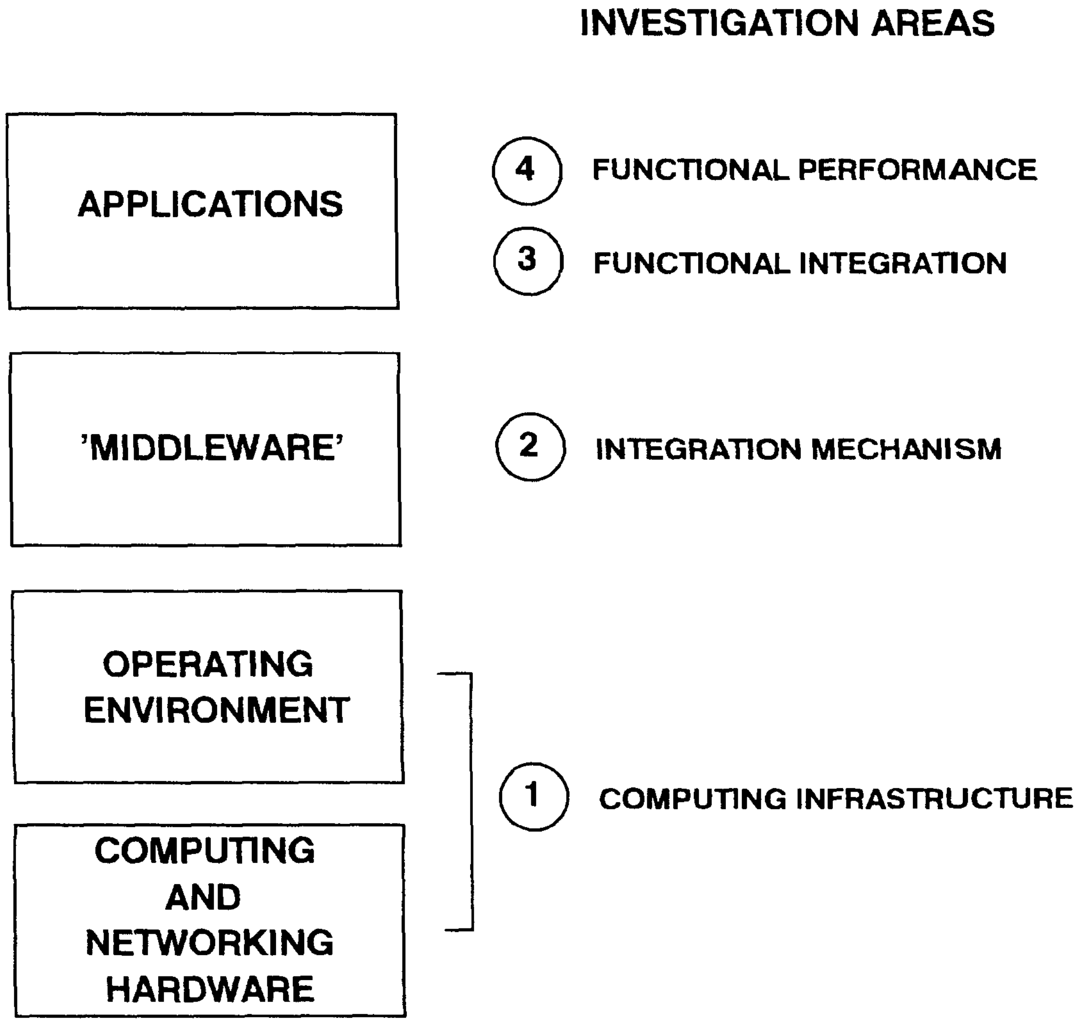

FIG. 1—COMBAT SYSTEM LAYER MODEL

(TDP). Such an approach takes account of the risks of novelty while ensuring that procurement and deployment can be carried out effectively. It also enables customers and potential users to relate to the success, value and implications of the underlying applied research, and industry to participate in construction, trials and ultimate demonstration, to the benefit of essential knowledge skills that will be relevant to eventual procurement. Technical issues relating to future combat systems are therefore presented in the context of a TDP. Specific concepts covered include:

- The use of commercial products and open standards for the computing infrastructure of a future combat system.
- The need for an efficient, robust and open method of integrating functional combat system components.
- Achieving a common architecture populated when appropriate with components meeting common requirements.
- Achieving reuse of specialist and legacy software components.
- The need for interoperability both within and between combat systems and with higher level command information systems.

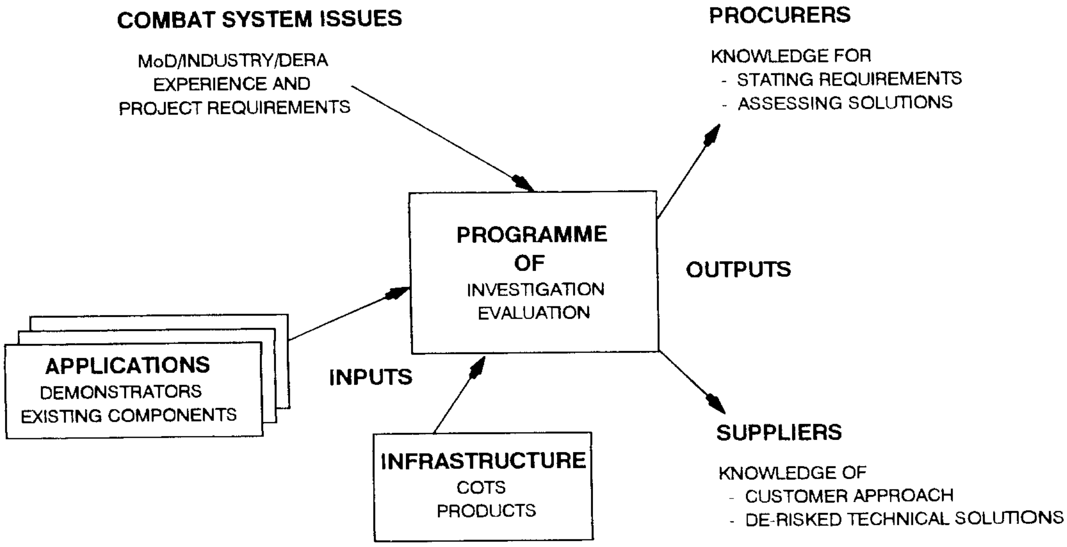

FIG. 2—CONTEXT FOR TECHNOLOGY DEMONSTRATION

A context for technology demonstration is shown in (FIG. 2). The outputs are now hardware of software specifically but knowledge relevant to operational and technical questions and to procurement. While some software engineering and hardware will be required if experience of technologies and of their application to combat systems is to be gained, and true demonstration is to be achieved, the controlling elements will be the studies and investigations carried out under the four headings listed earlier and which are designed to address priority issues.

Area 1—Computing Infrastructure

The term 'computing infrastructure' is applied here to the computing, networking and standard peripheral devices required to support the specialist software application functions of a combat system. Recognizing that the commercial marketplace now dominates the production of such infrastructure components, the key questions for combat system designers and customers are:

1. To what extent and benefit can the required infrastructure be obtained from the commercial market?
2. How can it be used effectively?

Central issues affecting the answers are:

- The suitability of products and standards
- Openness
- Real-time performance and robustness
- The extent of application
- Technology refresh—through-life costs
- Support for security and high availability.

The expectation is that products and standards developed for the commercial mass market will satisfy the majority of processing, networking and operating environment needs for future combat systems even though the real-time demands, particularly in sensor and weapon applications, are stringent. One objective is to learn how to use commercial products in effective ways and to develop strategies to overcome the volatility of the commercial product market. It is expected that there will be a cost benefit but also that a greater level of sophistication and flexibility will result from using products designed for a wider range of applications.

Experience bespoke military solutions has shown that interoperability between systems and sub-systems from different origins is unlikely to be possible without significant development. Integration of systems and upgrades to systems are invariably costly exercises. Such activities often result in less than ideal architectures and yet more bespoke software to develop and maintain. Thus, the trend towards and technologies for open systems emerging in the commercial world are of great interest to procurers and owners of combat systems.

COTS

The term commercial-off-the-shelf is widely used but is not usually defined, with the consequence that interpretations abound. Strictly, anything that is available and can be supplied by a commercial organization is COTS but this could include all the individual electronic components as used in current combat systems, and this is clearly not what COTS is intended to mean. Some combat system sub-systems in procurement and even in operational use claim to be COTS-based because, for example, they make extensive use of off-the-shelf processor cards. The intention here is to be more forward-looking still and to define COTS to be products from a commercial catalogue that only require selection and installation to provide a computing infrastructure. Examples could include:

- The workstation
- Local Area network (LAN)
- Operating system
- Database management system.

Open systems

The vision of open systems is that they will allow systems to be configured from disparate components more easily and thus provide systems that are more adaptable, interoperable and easily upgraded, with more choice over component selection (allowing exchange of components if more capable or cheaper ones become available). This vision can be applied to both the infrastructure and functional levels of the combat systems but, before such discussion, the matter of what constitutes an open system must be addressed.

A straightforward view of an open system is one that uses products and standards that are in the public domain. As is often the case in the computing world, there are formally agreed standards and de facto standards. An agreed open standard is of little value if it has very limited use, and a widely used de facto standard supported by a single supplier also carries risk. The ideal would be widely used and supported open standards but it may be some time before an adequate set emerges from the commercial world. In the meantime, the requirements for openness in combat systems can be explored and metrics developed by which their achievement and value may be assessed.

The use of commercial products to create the computing infrastructure of a combat system inevitably means that hardware and operating software will have to be 'refreshed' during the life of a platform (ship or submarine). To make the COTS infrastructure cost-effective, any such technology updating should be possible without having to change application software and be independent of supplier and particular products (e.g. processor and/LAN types). Thus open standards are needed at the virtual machine level in which the applications reside or, at least at some level within the operating environment, there must be a completely open virtual machine standard.

Area 2—Integration Mechanism

The MoD already uses a common integration method for major components of its shipborne combat systems (which include sensors, weapons and combat management system) involving standard data formats and exchange specifications. However, this method is effort intensive, software is replicated in different languages by different teams working on different sub-systems, and extensive low-level testing is then needed to ensure correctness; any subsequent changes to interfaces are time-consuming and expensive.

Another difficulty experienced in current systems is simply that of access to data within sub-systems. When considering new or improved functionality, it is often a requirement to record or interface to data at particular points so that the effect of changes can be analysed and demonstrated. With current combat system engineering approaches, this frequently means developing special hardware and software. An open approach should allow access to any databases or data flows between software modules.

Openness at the infrastructure level may well allow software applications from different origins to be loaded provided they conform to the same infrastructure standards. However, except for stand-alone cases, such applications would not run owing to an inability to interact with other combat system components. Necessarily, a combat system contains a large quantity of specially defined and formatted data, the majority of which conforms only to local standards. To enable applications from diverse sources to be integrated, the current approach is to establish interfaces and define special messages for data exchange; software changes are then usually required in both the new and existing applications for successful integration. The key question is:

> What technique can be used to provide a more flexible, higher integrity, robust and open means to integrate combat system applications functions?

Issues include:

- Is there a mechanism that is hardware, operating environment and language independent?
- How far do open approaches such as Common Object Request Broker Architecture (CORBA) go to meeting the requirements?
- What level of real-time performance can be achieved with commercial solutions?
- Can a more flexible approach to interfaces be found that does not require all interactions to be pre-defined?
- How feasible is it to reuse old components that cannot easily be modified?

To make progress on this important area of integration mechanisms and open systems, the aim is to conduct investigations between combat systems modules (from existing prototypes) to provide realistic instances of application integration requirements. (Fig. 3) illustrates the experimental system.

Two approaches are being investigated, both using the open standard CORBA as the underlying mechanism. In the first approach, a CORBA-compliant product is providing a machine-and language-independent method of specifying the objects common to two or more modules. It then provides the code, in the desired programming language, for each module to interface with these objects. This common-object approach is being applied to tactical data required by a planning module and supplied from the tactical picture compiler module, as shown in (Fig. 3).

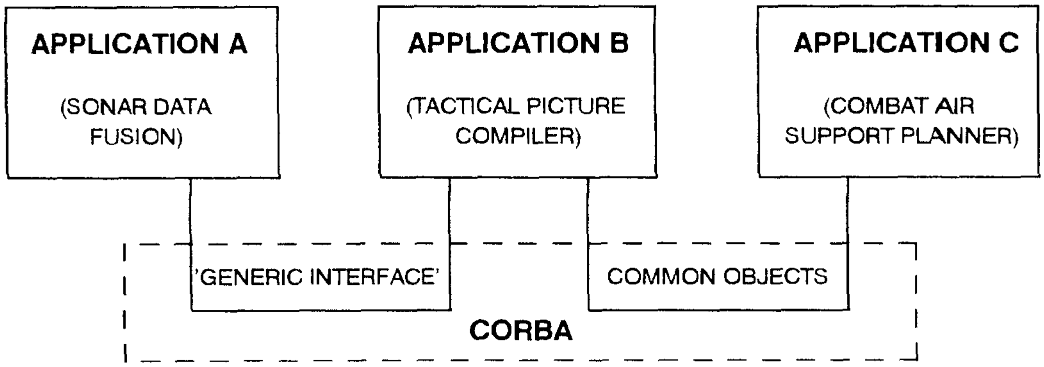

FIG. 3—INTEGRATION MECHANISM EXPERIMENTAL SYSTEM

A second integration example is provided by a sensor-processing sub-system interface to the tactical picture compiler module. This example requires a greater diversity of interactions through common objects, the supplier module, in this case the sensor sub-system, creates and maintains a database of objects representing all the data that any other module may wish to access. Another module, in this case the tactical picture compiler, specifies the data it requires at run-time and accesses the data without the knowledge of the sensor-sub-system.

A key issue that is crucial to the successful use of commercial methods in real-time combat system applications is the real-time performance overheads of these approaches.

Output from this area of work will include recommendations on the mechanism to be adopted, a policy for its application and detailed technical knowledge of how to apply particular mechanisms successfully.

Area 3—Functional Integration

Combat systems increasingly include a large number of complex real-time software functions that are special to maritime operations and sensor and weapon equipments. Much research continues to go into refining the algorithms and knowledge within such functions, and they represent important capabilities and a large investment. To reduce procurement costs and timescales and to enable easier upgrades during their lives, there is a desire to be able to select and integrate the best components that already exist or as they become available. An open integration mechanism will assist this approach but it also requires the functions themselves to conform to a generic functional architecture so that they fit together like pieces of a jigsaw. This area needs to address the question:

> How can the existing and emerging advanced functionality be integrated into effective combat systems to suit a range of platforms?

Issues to be addressed include:

- Is it practicable to standardize combat system objects to facilitate easier integration?
- Is there a generic functional architecture that effectively partitions the functionality and takes the user organization and tasking into account?
- To what extent can functional requirements be common on different platform types?

Some flexibility has been achieved in current systems by the MoD's taking control over the major sub-system interface specifications. This allows sub-systems to be replaced and, to a degree, new sub-systems to be added, but all these sub-systems must be designed to fit with the imposed system architecture, making it difficult to incorporate functionality from elsewhere. Also, the replacement of elements within sub-systems is only practically possible by the original sub-system supplier.

The ability to plug together combat systems functions available from different suppliers requires at least that a generic architecture is recognized and a degree of standardization of the objects common to two or more functions achieved. Steps have already been taken to define objects exchanged between combat systems and command information systems, as this is crucial for interoperability. A key issue is whether it is practicable to extend this to all the objects within a combat system in a way that will assist integration without imposing too fixed a solution.

Outputs from these investigations are likely to include a prototype combat system object library, a baseline functional architecture with proven characteristics, and common application requirements.

Area 4—Functional Performance

The ultimate goal of the integrated combat system is to provide the required military capabilities. A well engineered integration of all the functional components is a necessary but insufficient condition for achieving that goal: in addition, the collective functions must achieve the required level of combined performance. This performance is not related to such things as the processing power, capacity or communications bandwidths—these can be assumed to be sufficient in a well engineered system. The performance here relates to that inherent in the functions themselves, the algorithmic or knowledge performance. Past systems relied heavily on human brains to perform the information extraction, fusion and planning processes, while software was limited largely to data communication, storage, display and calculation The trend now is to automate many of these processes using advanced algorithms and knowledge bases, but this demands much greater attention being given to performance specification—before, the problem was avoided by simply providing more human assistance. Increased capability and reductions in defence costs hinge to a significant extent on the success and continued expansion of such automation.

A particular problem is the difficulty of specifying the quality of information that an automated sensor system will deliver under the range of read-world conditions in which the combat system will have to operated. Given this difficulty, the performance of all the follow-on processes of information fusion, decision support and direction of resources, as generically illustrated in (FIG. 4), is likely to be compromised. Hence it is important to look at overall combat system performance and to solve any problems relating to incompatibilities between one sub-system and another. The question to address is:

> How can the overall combat system performance requirements be specified and applied to ensure that the benefit of the integrated functionality is realized?

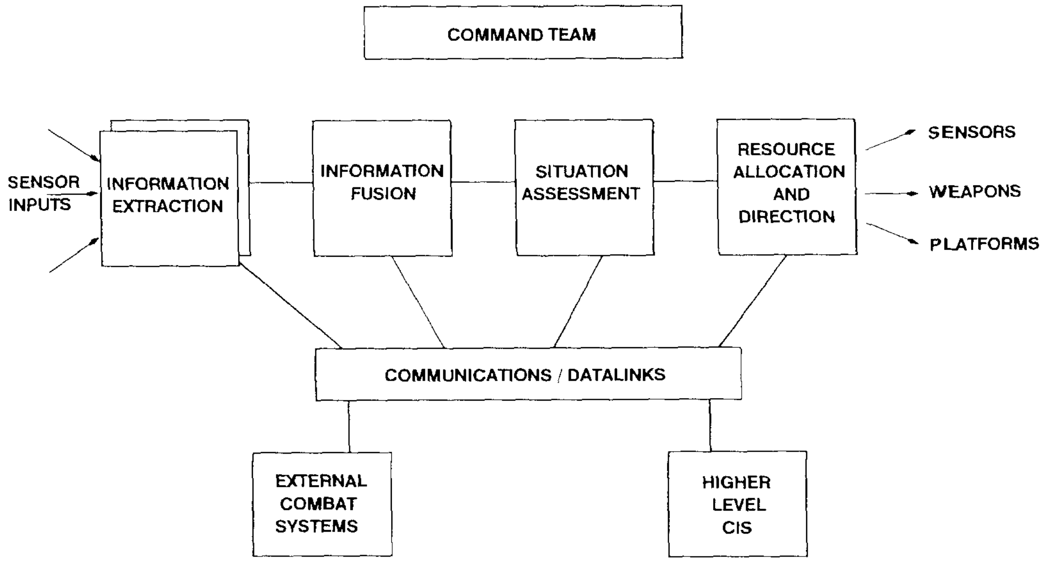

FIG. 4—COMBAT SYSTEM FUNCTIONAL ARCHITECTURE

Issues include:

- Are the operational performance requirements adequately quantified?
- What combat system performance metrics can be applied?
- Are the sub-system performances adequately specified and mutually compatible?
- How can performance be specified so that it is achieved under operational conditions?
- How can human performance factors be taken into account?

The output of these investigations will be an approach to combat system performance specification and assessment.

Summary and conclusions

A programme of technology demonstration to address issues relating to future naval combat systems has been outlined from a technical point of view. A high-level structure has been proposed for what is a complex programme with many detailed objectives. Four investigation areas have been introduced to separate those issues related to computing infrastructure, integration mechanism, functional integration and overall combat system performance. Some of the wide range of issues have been introduced to separate those issues related to computing infrastructure, integration mechanism, functional integration and overall combat system performance. Some of the wide range of issues have been discussed.

The technology demonstrator programme will integrate a number of combat system components, particularly the latest prototypes but also existing functions, and will conduct a series of experiments designed to measure the performance of infrastructure and integrated applications. The programme will need to be developed jointly by DERA and industry to ensure maximum validity and exploitation.