Risk Reduction and Cultural Change in the Procurement and Support of UK Naval Combat Systems using Concept Proof Demonstrations

Mr. Michael Looney
DERA Portsdown West,
Portsdown Hill Road,
Fareham,
PO17 6AD, UK
Tel: 44 1705 336515
Fax: 44 1705 336505
e-mail: mjlooney@dera.gov.uk

Mr. Shane Lunga
DERA Portsdown West,
Portsdown Hill Road,
Fareham,
PO17 6AD, UK
Tel: 44 1705 336522
Fax: 44 1705 336505

Abstract

In attempting to de-risk the development of future Naval combat systems the MoD has initiated a research programme to investigate the new approaches which are being proposed. The paper addresses the background to that programme, identifies the main areas of research still to be carried out and gives some results from experiments carried out to prove some of the concepts which are new to the military domain

1. Introduction

In May 1996, the Directorate of Operational Requirements for Sea (DOR(Sea)) issued the Combat System Technology Demonstration Framework (CSTDF) Strategy Paper which initiated the Combat System Integration (CSI) research programme. The paper stated that future naval combat systems were likely to be unaffordable if the Ministry and Defence (MoD) and industry continued with the incremental evolution of bespoke systems. It considered that an alternative approach was to capitalise on commercial products and standards, particularly those associated with open systems, and with the use of legacy components, to maximise on previous investments made by MoD and industry. The research programme to support this initiative will draw on Applied Research Technology Demonstrators (ARTDs) and Technology Demonstrator Programmes (TDPs) and address such issues as commonality across platforms aimed particularly at the Future Carrier (CVF), the Future Surface Combatant (FSC), and the Future Attack Submarine (FASM). In this paper details are given of the planned programme of research to be carried out under the CSI heading.

Since the original DOR(Sea) paper was written there has been a period of preparation and clarification during which key issues have been identified and various preliminary investigations and studies have taken place. These have addressed a range of areas all of which have contributed to the generation of the CSI research programme. The earlier research into the use of an open systems approach has addressed such issues as the use of Common Object Request Broker Architecture - (CORBA) to support distribution of functionality; the porting of existing legacy subsystems into such an environment and paper studies which have looked into such areas as software portability and the refresh of system components caused by requirements changes or obsolescence. Details from these with some of the results, and lessons learned from the work are given in the paper.

2. Future Programme Overview

The overall requirement is for research to support and de-risk the development and procurement of capable, affordable, upgradeable and supportable future ship and submarine Combat Systems. The CSI research programme will satisfy this requirement by investigating innovative approaches to development and integration including the increased use of COTS components, open systems and commercial standards, coupled with a more rigorous approach to the specification of affordable and operationally justifiable functional and non-functional system requirements. The relationship with legacy equipment and the viability of an evolutionary approach to combat system integration and development is a fundamental aspect of the programme.

The emphasis of the work will be on the key risk areas to future programmes - specifically CVF, FASM and FE. The case for an innovative approach to the integration of these future combat systems will be made by taking specific examples of technological developments, and thoroughly exploring the potential performance, programme, risk and cost implications for these future platforms.

In the initial years the work will concentrate on defining and then gathering metrics to characterise future combat system commercial and legacy components to enable them to be procured (that is, specified, accepted, and supported) cost effectively. This phase will also produce the initial options for future combat system's generic architecture and infrastructure. The CSI programme will also identify where the current combat system policies and processes may need to change.

The second half of the programme will see CSI evolve to investigate the integration of segments/sub-systems of a combat system and assess the ability to achieve the required capability and performance throughout the lifetime of future warships using a component based approach and a given framework. Policies and procedures to derive the specification, carry out the development, and support the through-life needs of such a naval combat system will be investigated.

As part of the CSI research programme, an experimental test bed, known as the CSI ARTD, will be built from ARTDs emerging from other applied research programme (ARP) areas and products available from industry. As indicated the CSI ARTD will be used to investigate combat system issues arising from the integration of commercial and legacy components in an open systems environment and provide a facility for ARTDs or industry to test their capability in a total combat system environment. The test bed will be developed through evolutionary build stages, with each stage designed to undertake specific practical investigations, through experimentation, to enhance combat system integration understanding, methods, and technologies.

The goals of the programme can be grouped into the following broad areas listed below in no particular order of priority:

- To provide a focus for Combat System related research and to establish an emphasis on Joint and Combined Operations ensuring that the programme is consistent with MoD endorsed high level strategies.
- Investigation and practical demonstration of the functional integration of a combat system, focusing on key risk issues, and the use of commercial integration mechanisms to enable functions to be interfaced more efficiently.
- Investigation and documentation of the information management requirements of the Combat System including the interoperability and CIS requirements to support future joint operations.
- Development of open system architecture options incorporating commercial and legacy components and the practical demonstration of the use of commercial components in the military environment.
- Development of processes to support a systems engineering approach to an open component (legacy and commercial) based Combat System utilising both commercial and military standards, and an assessment of the impact on Naval polices and procurement practices including cost and support.
- Production of requirements statements, metrics and guidance to support specification and acceptance of Combat System design solutions including non-functional requirements.
- Investigation of integration anomalies in current Naval combat systems to ensure such issues are alleviated through an improved whole combat system design approach.

2.1 *Proposed Programme Details*

To address this programme of work the set of tasks shown in the following list has been established.

- Item I Exploitation
- Item 2 Performance and Functionality Research
- Item 3 Infrastructure Research
- Item 4 System Engineering Lifecycle Processes
- Item 5 CSI ARTD test bed

This will allow the three main areas of Performance and Functionality, Infrastructure Issues and System Engineering Processes to be progressed with support from Exploitation and a CSI test bed. While these items have been identified in order to structure the work, tasks within all of the items are very interdependent and will be addressed as unified topics. This is particularly relevant to system architectures to support the needs of functional sub-systems in terms of their information flow, performance and through life change requirements.

2.1.1 *Work in Item 1:- Exploitation*

This task will provide for the alignment and co-ordination of the assignment with other activities both internal and external

2.1.2 *Work in Item 2 :- Performance and Functionality*

The research will look into Naval combat system functional and performance requirements and will feature tasks in the following areas:

- Requirements drivers and their associated constraints;
- The definition of a generic functional framework and integration issues associated with the specification of future combat system components;
- The analysis of potential methods for functional and performance evaluation including human factors and the partitioning of combat system components;
- The management of information within a combat system, including human factors and the production of models to assess interoperability and data flow issues.

Current systems are seeing an explosion of computer based information, growing on an equipment by equipment basis with little overall management. There is therefore the potential for duplication, inconsistency and unaffordable support costs. In addition interoperability across the theatre is now seen as a big challenge. The aim of the work package is to take an intra and inter look at the management of information by the combat system bearing in mind the likely extensive use of off the shelf components in future systems.

The work package will survey associated research and programmes to determine:

- sources of information to support the work;
- models that have been used elsewhere which might be candidates for re-use;
- associated initiatives with which the work may be required to be compliant.
- identify information exchanges at the combat system boundaries and examine future trends in wide area information sharing to determine the principal factors that will impact upon combat system design;
- determine the information available within, passed around and through the future combat systems including flows to and from the user, and propose alternative ways of managing and sourcing this information;
- produce a central data object repository containing data definitions to promote common understanding and on which future data interfaces can be based;
- in conjunction with the functional and infrastructure tasks, construct a model of the system to include the information elicited in this work package.

Technical Issues to be Addressed

- Identification of the key drivers in developing a combat system which will be delivered on time, to specification and capable of functional evolution to support the evolving requirement of the platform.
- Production of an information model to support platform, force and joint operations and aid a common understanding and the production of more generic interfaces.

- Production of a generic functional framework, suitably partitioned, to facilitate future evolution and 'plug and play'.

2.1.3 *Work in Item 3:- Infrastructure Research*

This will cover research into aspects of potential infrastructures for future combat systems and will address the following topics:

- A review of current commercial infrastructure models, including their frameworks and structures;
- The derivation of metrics to enable open systems to be specified and measured, including areas such as openness, security and safety criticality;
- The definition of candidate architectures for future Naval combat systems;
- An assessment of commercial components to enable their specification and selection to be achieved for military requirements (including any aspects relevant to their refresh through a system's lifetime);

The generation of the future combat system architecture options will be carried out through the study of commercial infrastructure models and frameworks, the identification of the characteristics of mixed commercial and legacy component based combat systems, the data distribution requirements of information flowing within the combat system and across its boundaries and any areas of commonality across platforms. Both functional and non-functional combat system requirements will be considered, particularly their impact in such areas as real-time operations safety and security. Where necessary models of suitable system structures will be initiated prior to their physical instantiation in the CSI ARTD. This task will specify a number of experiments which will be undertaken, that allow practical investigation of integration issues and the impact of architecture options on combat system performance and capability. A specific output of this task will be the provision of open system infrastructure options for future systems, the methods to be employed in investigating them in the CSI Research programme and a means of characterising open systems.

The component evaluation task will develop a means of specifying the characteristics of software components, both commercial and legacy, which will provide the necessary information for their selection as meeting the requirements for inclusion in a military environment. This work will take account of associated research within the ARP and results emerging from associated programmes both in the UK and abroad.

Technical issues to be addressed will cover how both commercial and legacy components can be used in a manner consistent with the desire for open systems. What the salient characteristics are that software components must exhibit in order to reduce down stream risk in the move to the component based architectures for future combat systems; and the production of a conceptual combat system architecture to support the inclusion of both commercial and legacy components. It will need to:-

- To develop open system architecture options for naval combat systems that allow the incorporation of commercial and legacy components in the military environment.
- To investigate the use of commercial integration mechanisms within the combat system, to enable components to be interfaced more effectively.

- To investigate the impact of information requirements derived in item 2, on the infrastructure of the combat system.
- To produce requirements statements, metrics and guidance to support specification and acceptance of combat system design solutions, including non functional requirements and human factors.

2.1.4 *Work in Item 4:- System Engineering Lifecycle Processes*

This area will address the investigation of those processes needed to support a complete lifecycle system engineering approach to component based combat systems and will cover:

- The lifecycle processes for development of open systems with both commercial and legacy components, including the specification of operational and AR&M requirements;
- The derivation of policies associated with safety, security and ILS;
- The production of cost models for open systems with both commercial and legacy components.
- The derivation of combat systems performance metrics, aligned to MoD's effectiveness assessment methodology, for use in the specification and subsequent acceptance of future procurements;

The development and procurement of capable, affordable, upgradeable and supportable future ship and submarine Combat Systems based predominantly on open system commercial standards, using a combination of commercial, legacy and developmental components, will require changes to the way they are engineered and evolved during the system lifecycle. The task will have a co-ordination role in integrating the outputs of the other assignment items where development of system lifecycle processes are required. Processes will be developed to support a systems engineering approach to software components based Combat Systems using open systems and commercial and legacy components. The aim is to provide an understanding of the process of specification, control and acceptance of open systems procurements that are based on commercial and legacy components.

Technical issues

- To develop processes, methods and policy guidance for combat system engineering during the lifecycle from concept through to support in service, as applicable to combat systems designed using commercial components in an open systems environment.
- To support MoD in the structured specification of non functional requirements which are operationally justified, affordable and testable.
- To investigate the cost of owning such combat systems and recommend approaches to make them more affordable.
- Safety, Security and Integrated Logistic Support (ILS) policies.
- Software licences.
- Cost metrics appropriate to the application of commercial components and open systems and lifecycle cost models.
- MOEs and MOPs appropriate to MoD's operational effectiveness assessment methodology.

3. Completed Research

Having outlined the proposed programmes for this and future years the rest of the paper will address the concepts which have already been examined by experimentation and indicate some of the results obtained form the areas of work completed so far.

3.1 *Practicable Open Systems Technology (POST) evaluation*

The Practicable Open Systems Technology (POST) Applied Research Technology Demonstrator (ARTD) programme, investigated and provided a practical demonstration of an open systems technology approach for potential use in the infrastructure of future maritime combat systems.

The first phase of the programme developed a baseline demonstrator to investigate the design and implementation issues of using a commercial middleware approach (Common Object Request Broker Architecture - CORBA) to integrate three real-time ARTDs namely: the Combat Management Integration Support Environment (CMISE), Sonar Data Integration (SDI) and the Combat Air Support Planning Aid (CASPA).

The approach taken for the evaluation phase was to derive a set of test specifications against which middleware could be evaluated. Within the time-scales and budget constraints of the programme, three areas of evaluation were selected as most valuable in providing an insight to the applicability of middleware in a real-time combat system. These covered engineering, performance and data integrity. This document presents the results of this evaluation phase for two CORBA compliant products, Orbix from IONA Technologies and ORBexpress from Objective Interface Systems.

3.1.1 *The results from the POST evaluation include*:

- No middleware conformance results against a de jure or de facto standard were available at the outset of the POST programme. If the results were available then they would have benefited the programme during the selection of appropriate middleware products and would have assisted in determining inter-dependencies between Commercial off the Shelf (COTS) products used in the system.
- CORBA does support an interoperable heterogeneous computing environment across different hardware, operating systems and implementation languages, including interoperability between different vendors' CORBA products. The use of the CORBA services can assist in developing an interoperable infrastructure for a system, with the potential benefit of reducing the development time of the system, as the infrastructure services are provided by COTS vendors.
- CORBA does provide an infrastructure for the design of scaleable and fault tolerant systems. Particular emphasis must be placed on concurrency, data integrity, multi-tier architectures and load balancing.
- The performance indications of CORBA from the evaluation programme are encouraging. The results are dependent on the specification of the host platform. Anomalous results did

account for approximately 0.5% of all measured values in every test. These have yet to be explained and no assessment has been made of their potential impact.

The POST programme has found no reason to preclude the use of a CORBA compliant middleware within a Combat System. However, the use of CORBA services may have an impact on this and further work is required to ensure that services have no adverse effects. The use of CORBA compliant products for the construction of ARTDs have positive benefits, including:

- a standards based method of interfacing, thus reducing the time to integrate ARTDs;
- the use of IDL as a means of defining an interface is valid whether CORBA is used or not;
- an IDL approach ensures consistency from design to implementation (care must be taken in the handling of dynamic behaviour);
- the availability of wide hardware, operating system and implementation language support;
- it provides a simple solution to problems associated with heterogeneous interconnectivity;
- the availability of commercial CASE tool support for IDL generation and configuration.

The following points were drawn from the data integrity aspects of the POST evaluation:

- Data integrity is an essential aspect of any system particularly a military one. Identifying potential component level data integrity problems early should ensure that system problems are easier to identify and hence correct.
- The data integrity tests showed no errors, however due to the availability of processing platforms only a limited number of test could be performed. A greater sample of data types and structures tested would provide a higher level of data integrity assurance.
- Batching of message data is more likely to induce ordering problems than switching interfaces on a message by message basis. However, ordering problems are more likely to occur as a result of poor application design.
- Market leading middleware products are less likely to suffer from data integrity problems, due to the size of the installed user base, however the potential for error can never be ruled out.

3.1.2 *POST evaluation programme recommendations*.

- Tracking the conformance of COTS products (including middleware products) with de jure or de facto standards should be undertaken to ensure their successful use.
- COTS middleware products should continue to undergo appropriate tests against constantly evolving specifications prior to their use.
- The quality or accuracy of system functions, profiling and measurement tools should be checked prior to the commencement of any evaluation programme, e.g. the granularity of the system clock.

- Further research should be conducted into the cause of the observed latency anomalies, during the execution of the evaluation tests. The possible use of products specifically designed for real-time processing should be considered.
- CORBA provides an infrastructure for the development of a heterogeneous computing environment. However, it is recommended that good system design and engineering methods are still paramount in delivering a high performance, high data integrity, fault tolerant and scaleable system.
- In order to ensure interoperability in a CORBA environment, the Internet inter-ORB protocol (IIOP) should be used.

3.2 *Legacy Porting*

The Practical Portability experiment was instigated to gain first hand experience of undertaking a technology refresh cycle on a set of legacy application components in order to learn lessons from it. The legacy application selected was the Combat Management Integration Support Environment (CMISE) and its components were 'upgraded' from Ada83 to Ada95 and the bespoke infrastructure replaced with CORBA open interconnects.

Two prototypes of the CMISE were produced, the first comprised Ada95 variants of the Data Fusion Module (DFM), the Front-End Processor (FEP), the Tactical Picture Server (TPS) and the System Support Module (SSM) components of the CMISE. The second prototype built upon the first and introduced Ada95 based Common Object Request Broker Architecture (CORBA) mechanisms into the system infrastructure in order to significantly improve the openness of the system.

The successful production of the first prototype proved that it was possible to re-engineer complex legacy applications into a similar language, even one which is more modern and which supports Object Oriented (OO) features. The second prototype illustrated that bespoke communications could be replaced by a far more open mechanism.

3.2.1 *The potential military benefits.*

- shorter periods to port legacy applications;
- improved methods of determining the costs of porting software;
- improved appreciation of the cost of ownership of COTS products.

The capabilities of COTS products may be limited by dependencies on the use of other products from the same supplier. These dependencies are often not obvious in the descriptions of the capabilities of products and their subsequent discovery introduces delays and can preclude the use of a particular COTS product.

COTS products, and CASE tools in particular, offer many different features. Often many different configurations of a product are available, by the use of 'add-ons' to the standard product. The precise configuration required should be identified and specified in detail in the order, although this will not guard against the supplier providing the wrong configuration. Three different versions of the Rational Rose CASE tool were obtained before the 'best' configuration was delivered. This introduced significant delays because of the time required to install and evaluate each version. Greater use of supplier consultancy prior to ordering these products and a better understanding of their capabilities should alleviate these problems.

It may be possible for an individual who has extensive experience of a legacy system to produce a tool which can produce IDL, or another language independent interface definition standard, from the legacy code. However, it is unlikely that for a complex legacy system a tool is likely to be able to fully automate this process unless a considerable amount of effort is expended in producing that tool.

The introduction of a binding layer has the following advantages:

- It provides isolation of application developers from the CORBA libraries.
- It allows the existing types to continue to be used in the applications with the translation to the IDL generated types performed once only in this binding layer.
- It isolates the application components of a legacy system from a change in middleware.

3.2.2 *Legacy Porting Results*

Based upon the outputs of the work undertaken to complete stages 1 and 2 of the experiment, the following results have been derived:

- Four candidate application components have been successfully re-engineered in Ada95 from their original Ada83, still using bespoke communications mechanisms. These components have been individually tested and integrated into the Ada95 prototype for system testing. Indications are that the functional attributes of the re-engineered components are comparable to the original. However, there currently appears to be a higher requirement for CPU usage for Ada95 tasks than their Ada83 equivalents.
- The bespoke mechanisms originally used for inter-component communications have been successfully replaced by a CORBA based byte-stream mechanism. The four candidate application components were integrated into the Ada95/CORBA prototype for system testing. Initial indications are that the performance of the CORBA inter-component communications mechanism satisfies the requirements for CMISE operation.
- The estimates for the amount of effort required to complete stages 1 and 2 of the experiment were based upon the experience gained in estimating CMISE modifications that use Ada83 and bespoke communications mechanisms. These estimates proved to be acceptably accurate, indicating that the development costs associated with Ada95 and CORBA for the chosen application are, at least, no worse than the original process.
- Training of the incumbent development team proved to be the only cost-effective option both because they possessed the detailed application knowledge and because attracting

additional resources with the required skills (Object Orientated design and programming, CORBA, C++/Ada) proved to be extremely difficult.

3.2.3 *The main lessons learned*

- Use of third party software products for greater portability will require increased computing resources for both the development and target systems.
- Use of third party software products for greater portability may necessitate a compromise in the physical design of a system. The effects of these compromises must be accurately assessed before commitment to a product is made. Design options should only include third party software products that already exist. Experience has shown that new and revised products are rarely brought to the market within the time scales cited during ‘pre-sale’ discussions with suppliers.
- The availability of third party software products is increasingly driven by the commercial market place. Careful selection of languages and operating systems should be made if maximum exploitation of these products is to be achieved Keeping in mind that support for less popular languages and operating systems is limited, thereby reducing design options and the potential for competition.
- The licensing arrangements for some third party software products are now ‘library based’. This may significantly increase the cost of software development as any resource that needs to interact with a particular library will need a licence.
- Assumptions, inter-dependencies and many features in the form of ‘add-ons’ to the core product, may force a technology direction to be adopted based upon these factors rather than on a ‘best fit’ solution.
- As third party software suppliers offer precise specification of the product requirements is essential as the capabilities of these products may not scale up to large complex legacy systems.
- Whilst a third party software product may claim, and indeed have, compliance with a particular language or operating system, its inter-operability with other products and legacy systems may be limited by different conventions being adopted.
- It may be possible for an individual who has extensive experience of a legacy system to produce a tool which can produce IDL, or another language independent interface definition standard, from the legacy code. However, it is unlikely that for a complex legacy system a tool is likely to be able to fully automate this process.
- It has proved to be good engineering practice to isolate legacy systems from third party middleware products by the introduction of a ‘binding layer’. This layer can negotiate between different conventions adopted by the products and the legacy systems whilst being fully portable and ‘open’.

3.3 *Rates of Change of Components*

The aim of the Rate of Change and Refresh Study was to understand where and why Combat Systems have changed in the past, to enable future systems to avoid unnecessary change, and ensure that necessary change is implemented in the most cost effective manner.

By looking at changes that have occurred in the past it is possible to assess the likelihood of them occurring again in future systems. The study sought to collate and analyse project and supplier experience from current systems to find which characteristics of change would be applicable to future systems. Not all the areas investigated yielded suitable information for this study. Obtaining data to analyse was in fact one of the biggest problems for this study. Many commercial computer equipment suppliers ignored requests to supply data, and naval projects were cautious about releasing data for analysis. The data that was made available had often been collected for other purposes. However, by adopting a variety of analysis methods and using experts familiar with each system to ensure correct data interpretation, some characteristics and trends have been identified.

The basic drivers of change fell into six categories. Due to the generic nature of these categories, no reasons could be found why these categories would change for future systems.

- new requirements;
- supplier request;
- parallel development;
- obsolescence;
- correction and optimisation;
- the ‘ripple effect’.

From the data available the cumulative cost of change is greatest for support of new requirements and changes stemming from parallel system development, in particular where interface specifications are unclear. Although the problems of change triggered by component obsolescence is causing projects concern, the data examined to date did not provide any insight into the costs associated with this as no changes were clearly attributable to obsolescence.

By considering the drivers identified it could be seen that future systems may be required to deal with higher rates of change due to the increasing rate of change of technology and more importantly the increasing requirements for inter-operation.
The obsolescence of computing hardware between design and installation, or between installations, is seen as a problem that is likely to increase. The rate of obsolescence in the general computing market is making it difficult to supply a common hardware base for all installations. This, in turn, can effect the operating system causing a ‘ripple’ of change, testing and re-validation to occur. This raises key questions for the management of future systems:

- At what point is it cost effective to upgrade previous fits rather than support system variants that have occurred due to obsolescence between equipment purchase, with all the associated testing, configuration and support costs?

- Are there different rates of change for small granularity COTS systems i.e. COTS boards with bespoke infrastructure and large granularity COTS systems i.e. COTS servers and workstations? If so what is the most cost effective approach in the long term.

Due to the early onset of system change due to component obsolescence it was concluded that clear policies for minimising and managing component obsolescence should be established from the outset of the project. More specifically, there is a need to ensure contract payment milestones do not depend on the availability of physical hardware as this encourages early purchase of the elements most at risk from obsolescence.

Change, whatever the cause, can impact all areas of the design, test, documentation and support of a system. Therefore, if through-life costs associated with change are to be reduced not only must system design issues be addressed but also the associated processes. The balance of techniques shows that it is as much a question of management approach as technical solution.

There was a consensus of opinion from the system suppliers interviewed that systems could be designed to support change but that this would not provide a lowest initial cost solution and could therefore not be proposed under current procurement contracts. If support for change is required, a means must be found to make it happen by commercial incentives or clear, testable requirement statements.

General rates of change were calculated for future COTS hardware and software based systems. Based on the commercial rate of change and trends within current CMS systems the following results were obtained:

- During a systems stable phase (i.e. once initial development, correction and optimisation is complete and before the demands of new requirements exceeded the capability provided by the hardware.) a 5-8 year life-span for the main processing hardware, was considered likely.
- To accommodate the needs of new sub-systems and possibly incorporate new versions of operating systems, software will require a 12-18 month update cycle.

These findings are very much based on generalisations. To ensure future upkeep cycles support the requirements of Combat Systems, a clear understanding of the technology refresh and technology insertion a system is likely to undergo is required. The rates of change will be specific to each system's components and design, and should be established during development to inform the Reliability Centred Maintenance Analysis and provide opportunity to improve the RCM 'values' of the resulting system and/or align them with an existing upkeep cycle.

3.3.1 *Recommendations*

There is a wide range of research required to further improve the understanding of change in naval Combat and Combat Management Systems and recommendations are made for further research ranging from 'how to assess a suppliers ability to reduce change by effective component selection and obsolescence management' to 'how to encourage a supplier to design for change'.

In order to reduce the impact of change on any system it is recommended that the following contract, requirement and procedure issues are considered:

- Clear policies for minimising and managing component obsolescence are drawn up during system design .
- Bid assessment includes an evaluation of a company's processes for monitoring components
- The implications of the system supplier entering into long term supply agreement with a hardware or software vendor should be considered and resolved if necessary
- One software configuration item should be completely hosted on one hardware configuration item.
- The accuracy of interface documentation is improved. (currently the approach is based on the specification of an interface not the actual implementation.)
- Interface documents should capture the dynamics of data exchange i.e. timing and ordering constraints .
- The use of interface generation tools is encouraged .
- A clear Combat System requirement, design authority and system funding would provide a clear mechanism and authority for resolving conflicts over services and interfaces .
- Areas of the system at high risk from change should be identified and suppliers required to show how their intended design provides the ability to incorporate change in these areas.
- Technology refresh and insertion programmes are defined during the design and development phase.

3.4 *Portability of Components*

In attempting to make significant reductions in the development costs of naval combat systems several possible initiatives are being examined. These address various technical approaches to providing quicker, more flexible ways to develop and maintain the systems through their in-service life, taking into account the need to make best use of emerging technical developments in the commercial marketplace while still meeting military needs. This report considers what constitutes portability, what must be done to port existing software, how to develop portable software, and how portability might be measured, assessed and specified.

There are only two fundamental approaches to improving portability: make the application easier to port or make the computing platform easier to port to. All initiatives and technologies that attempt to improve portability must address one or other of these.

It is likely that the primary factors limiting portability will be dependence on specialist hardware or a high degree of dependence on a particular operating system.

The conflict between the desire to use advanced, cheap COTS technology and the requirements of naval combat systems must be resolved by using a mixture of COTS technology and bespoke software, with bespoke software in areas where the COTS solutions are not applicable. When software must be developed it should be written in a modular and platform-independent manner, and compiled and optimised for the specific equipment which will be fitted on the ship

There is a danger that specifying a computing platform that allows application portability will add a large overhead in terms of the resources required, particularly time or processing power.

Portability takes more money up front in order to derive benefit in the future. Experience so far shows that developing a software application which is more portable has an extra cost: design and implementation must be more controlled and interfaces be carefully specified. On the other hand, many of the features that make software more portable are considered good software engineering practice even for software where portability is not of primary importance.

An ‘evolutionary’ approach to system development will take longer in absolute time, but is more likely to deliver a bug-free program on the second and subsequent platforms than starting from scratch again.

Practical investigation should be carried out into what is needed to enable an application to be ported to different commercially available computing platforms. This investigation should cover not only the requirements for a newly designed application, but also the approach to be taken to enable portability of an existing application. Output is required in the form of guidelines for the specification of portable applications - preferably in the form of atomised requirement statements, and metrics for the assessment of the degree of portability of a component:

3.4.1 *Recommendations to improve portability*

- When developing a source-code-portable application choose a programming language that has a widely recognised and adopted standard.
- When use of an extension to the language standard or compiler-specific feature is unavoidable care must be taken to limit its use in order to simplify code changes when porting the application.
- Code-checking tools should be used to enforce rules about coding style and to flag up breaches of the rules.
- Software should be structured so that any platform-dependent code is confined to one component of the application.
- Standard good software engineering techniques should be used to increase the level of decoupling between an application and the rest of the system and keep their interaction to single points of contact.
- Treat the operating system like a piece of hardware and limit contact to well defined points.
- Rather than having the application call the operating system directly, use an intermediate, ‘middleware’ layer which subsequently makes the operating system call. Standard middleware should be adopted wherever possible.
- Making middleware calls ‘look’ as much like calls made in other environments as possible will make future ports to those environments much easier.
- With the move towards using commercial standards rather than military standards for military information systems the MoD needs to consider participating in the appropriate standards groups as part of the DSTAN role.

4. Conclusions

From the work carried out so far in support of the CSI research programme, there are already some positive results. It has been demonstrated that the user of commercial components is feasible and that legacy components can be supported on a commercial middleware base. There have been problems with performance but this has quite often been due to lack of knowledge on the how to apply the product most effectively in the domain. In several cases there is still considerable potential to exploit features which have so far not been tried, for example in the use of CORBA to support dynamic registration.

On the other hand there have been some problems. The main ones seem to be in the compatability between products and the resources required to run them. The experience has shown that the integration of commercial components into the military domain is going to require at least an order of magnitude increase in processor power and memory just to obtain the same levels of performance. This is a very worrying aspect as it will certainly increase initial development costs in the hardware areas which may or may not be offset with reduced software development costs. The need to carry out adequate assessment of components at the outset has not been part of the development process in the past, this will be a new cost to consider. This area of costing is likely to be even more difficult when the results of the change study are considered. It was clear that the information available to support this study was limited but it did produce some important relationships. The problem is not just technical and will spill over into the management and policy areas and the new CSI programme will need to look very carefully at such topics.

Using appropriate experimental prototypes to prove out some of the concepts being put forward appears to be a cost effective approach to de-risking future developments. While at present, in the whole of this area , there are still many issues about which too little is known and every piece of research seems to raise just as many questions as it answers, the CSI research initiative will, it is hoped, provide more of the answers and reduce the risk associated with the development of future military combat systems.

5. Supporting Reports

The following are references to the more detailed reports produced in the process of carrying out the experiments covered in this paper.

Software Application Portability
DERA/SS/WI/WP990024/1.0

CSI Practical Portability Lessons Learned
DERA/SS/WI/TR/990018/1.0

POST Evaluation
DERA/SS/WI/CR980224/1.0