NPS-SE-07-004

NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

Open Architecture as an Enabler for FORCEnet Cruise Missile Defense

by

Juan G. Camacho; Lawrence F. Guest; Belen M. Hernandez;
Thomas M. Johnson; Alan H. Kang; Giang T. Le; Brian J. MacGillivray;
Tu K. Ngo; Kyle B. Norman; Franklin Tomei Jr.

September 2007

Approved for public release; distribution is unlimited

THIS PAGE INTENTIONALLY LEFT BLANK

NAVAL POSTGRADUATE SCHOOL
Monterey, California 93943-5000

Daniel T. Oliver
President

Leonard A. Ferrari
Provost

This report was prepared for the Chairman of the Systems Engineering Department in partial fulfillment of the requirements for the degree of Master of Science in Systems Engineering.

Reproduction of all or part of this report is authorized.

This report was prepared by the Masters of Science in Systems Engineering (MSSE) Cohort 5 from the Naval Surface Warfare Center, Port Hueneme, and the Naval Air Warfare Center Weapons Division, China Lake.

Juan Camacho

Lawrence Guest

Belen Hernandez

Thomas Johnson

Alan Kang

Giang Le

Brian MacGillivray

Tu Ngo

Kyle Norman

Franklin Tomei Jr

Reviewed by:

John M. Green
Project Advisor

Paul Shebalin, D.Sc.
Second Reader

Released by:

David H. Olwell, Ph.D.
Department of Systems Engineering

Dan C. Boger
Interim Associate Provost and Dean of Research

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE	*Form Approved OMB No. 0704-0188*

- Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.

1. AGENCY USE ONLY	**2. REPORT DATE** September 2007	**3. REPORT TYPE AND DATES COVERED** Technical Report

4. TITLE AND SUBTITLE: Open Architecture as an Enabler for FORCEnet Cruise Missile Defense	**5. FUNDING NUMBERS**
6. AUTHOR(S) Camacho, Juan G; Guest, Lawrence, F.; Hernandez, Belen, M; Johnson, Thomas M; Kang, Alan H; Le, Giang, T; MacGillivray, Brian J; Ngo, Tu, K; Norman, Kyle, B; Tomei Jr, Franklin.	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000	**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A	**10. SPONSORING / MONITORING AGENCY REPORT NUMBER**

11. SUPPLEMENTARY NOTES

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION / AVAILABILITY STATEMENT • Approved for public release; distribution is unlimited	**12b. DISTRIBUTION CODE** A

ABSTRACT ***(maximum 200 words)*** Advancements in missile technology have made cruise missile capability available worldwide. Current US naval weapon systems lack full interoperability across multiple platforms and full integration of detection, control, and engagement processes against incoming targets. The key to defeating future threats to our military assets is in gaining additional reaction time. This can be accomplished by leveraging collective sensor detection data throughout the battlespace, utilizing the FORCEnet data resources to evaluate the threat, and engaging the threat with a tiered defense. The objective of this capstone project is to address the above issues through the use of Open Architecture (OA) within a FORCEnet environment. This report focuses on the development of a conceptual architecture for Cruise Missile Defense (CMD) that combines FORCEnet architecture requirements with Program Executive Office of Integrated Warfare Systems (PEO IWS)'s OA functional architecture technical requirements. Further, this conceptual architecture is compared with PEO IWS's functional architecture via a series of systems engineering diagrams. These diagrams culminate in a simulation model that analyzes and determines the validity of the conceptual architecture. Results from the simulation model show that the conceptual architecture performed significantly better than PEO IWS's. These results are attributed to the addition of a re-engagement loop called Observe-Orient-Decide-Act (OODA).

14. SUBJECT TERMS FORCEnet; Open Architecture; Cruise Missile Defense	**15. NUMBER OF PAGES 201**
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT Unclassified	**18. SECURITY CLASSIFICATION OF THIS PAGE** Unclassified	**19. SECURITY CLASSIFICATION OF ABSTRACT** Unclassified	**20. LIMITATION OF ABSTRACT** UL

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

Advancements in missile technology have made cruise missile capability available worldwide. Current US naval weapon systems lack full interoperability across multiple platforms and full integration of detection, control, and engagement processes against incoming targets. The key to defeating future threats to our military assets is in gaining additional reaction time. This can be accomplished by leveraging collective sensor detection data throughout the battlespace, utilizing the FORCEnet data resources to evaluate the threat, and engaging the threat with a tiered defense.

The objective of this capstone project is to address the above issues through the use of Open Architecture (OA) within a FORCEnet environment. This report focuses on the development of a conceptual architecture for Cruise Missile Defense (CMD) that combines FORCEnet architecture requirements with Program Executive Office of Integrated Warfare Systems (PEO IWS)'s OA functional architecture technical requirements. Further, this conceptual architecture is compared with PEO IWS's functional architecture via a series of systems engineering diagrams. These diagrams culminate in a simulation model that analyzes and determines the validity of the conceptual architecture. Results from the simulation model show that the conceptual architecture performed significantly better than PEO IWS's. These results are attributed to the addition of a re-engagement loop called Observe-Orient-Decide-Act (OODA).

The views expressed in this thesis are those of the authors and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

EXECUTIVE SUMMARY

This paper focuses on the development of a conceptual anti-ship cruise missile defense (CMD) model that integrates FORCEnet architecture components with the technical requirements of the Program Executive Office for Integrated Warfare System (PEO IWS) Open Architecture (OA) functional domain model. FORCEnet is the enabler of the CNO's vision of SEAPOWER 21 as the transformer of Navy and Marine Corps combat power projection.

The current pedigree of anti-ship cruise missiles (ASCM) sold on the global market proliferates as a poor man's air force by nations frequently hostile to the policies of the United States and its coalition partners. They enable nations of economically modest means to exercise power in response to perceived coalition threats, further political or regional power agendas, or to promote theater-specific mayhem. The cost of fielding these weapons is estimated to be orders of magnitude less than the cost of defending against them. They are lethal to naval forces and are characterized by their high-speed intercept, extended standoff range, advanced seekers, incorporation of multiple reduced observable technologies, and feature maneuvering trajectories making them difficult to detect and counter. The most recent witness to modern cruise missile capability was the coordinated attack of the Israeli Corvette *Hanit* in JUL2006 while patrolling 16 km off the coast of Lebanon. *Hanit* was struck by Hezbollah shore batteries by the second of two missiles in a high/low attack. The missiles consisted a pair of radar guided C-801/802 ASCM's or one C-801/802 and one EO/IR guided C-701, both of Chinese design. The first high missile sunk an Egyptian merchant vessel while the second sea skimming missile inflicted a mission kill leaving four dead and *Hanit* dead in the water.

Research and analysis verified that OA provides the framework for the development of FORCEnet design concepts that enables implementation of a CMD Integrated Fire Control (IFC) and command structure. PEO IWS, chair of the Open Architecture Enterprise Team (OAET), disseminates OA policies and standards iteratively and plans for its implementation in next generation surface and subsurface combatants. Fusion of the FORCEnet information architecture and an OA functional

domain model pose challenges and risks to be identified, managed, and mitigated. To realize the potential of this new architecture, FORCEnet will need to be an operational construct supporting all U. S. Navy forces prior to implementation.

The goal of the conceptual architecture is to fuse time-dependent tactical information from distributed sensor and platform nodes with minimal error and disseminate it in real-time to the decision-makers and Composite Warfare Commanders (CWC). The power of OA rests with the ease in which technology refresh occurs and its promotion of force-wide joint interoperability on the same distributed network. According to the Israeli Navy and Ground Forces Command, a lack of force wide joint interoperability caused the *Hanit* mission kill. FORCEnet, through OA, will expedite data flow enabled by common services and will reduce human interaction in the kill chain. This paper placed special emphasis on joint forces interoperability and prevention-based Information Assurance (IA) to ensure the rapid and accurate flow of tactical data among forces, and to prevent the compromise of information resulting from a breach in network security. IA must preserve the low reaction time needed to counter stressing threats and feature graceful degradation of the command function in the event of a network security breach.

The proposed architecture was developed using the systems engineering process to define the requirements, functions, evaluate capability gaps, and assess the risk of alternatives consistent with the technical characteristics essential for FORCEnet. A wide variety of models were subsequently built, discarded, evolved, and analyzed to verify that the proposed architecture met the OA domain model functionality. The models were constructed relative to three tactical scenarios with emphasis placed on three IFC scenarios including Precision Cue, Launch on Remote, and Preferred Shooter Determination all in the context of the Observe, Orient, Decide, and Act (OODA) loop. IFC is fundamental to improved cruise missile defense and refers to platform-independent sensor fusion and weapons pairing to overcome radar horizon or earth curvature effects that effectively constrain the battlespace volume. Through automated IFC, weapons are not limited to local surveillance and fire control. IFC capitalizes on networked sensors, reduces horizon and terrain limitations, and improves the layered defense against stressing CMD threats.

Two fundamental differences between PEO IWS's and the proposed architectures are that the proposed architecture contains a re-engagement loop after the first salvo is fired and it is horizontally integrated. The re-engagement loop following the kill assessment hastens message flow while horizontal integration simplifies and minimizes the functional interfaces.

To visualize the proposed architecture and its capabilities, strike group formations and CONOPS were developed to form the basis of the simulation needed to validate the proposed architecture. Classical queuing theory formed the foundation of the simulation model defending against arriving CMD threats. The model was based on a discrete-event quadruple serial queue; one arrival and three weapons assignment queues for each layered defense weapon. While the simulation model was based on the discrete-event model, it was built in the process-view of Arena version 10.0 simulation software. The kill chain functions were represented in the simulation in the context of the higher-level aggregation of the OODA loop. Uncertainty was represented by statistical distributions of stressor threat inter-arrival and service times that provides predictive forecasting through statistical inference, which was absent from the conventional OODA loop.

The measures of performance used in the simulation were the means of the following: the number of IA attacks; the number of electronic countermeasures softkills; the number of threat missiles killed by interceptor missiles; the number of reengagements; and the number of leakers. The PEO IWS architecture simulation results were the control group in both the raid and the stream cases.

The simulation revealed that there was no silver bullet and architecture changes alone will not solve the Navy's ability to counter stressing CMD threats. ASCM's successfully perforated the defensive layers resulting in leakers in both attack scenarios. Nonetheless, the simulation revealed that the proposed architecture delivered a statistically significant performance improvement compared to PEO IWS's OA functional domain model. Thus, the authors conclude that the proposed architecture should include a re-engagement loop and retain the human in the decide function of the OODA loop. In addition, the authors suggest re-grouping some functions within PEO IWS's OA Warfare Domain model to achieve improved performance and capability. These re-groupings are explored and explained throughout the report.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. BACKGROUND

Successful Cruise Missile Defense (CMD) for United States (US) Navy ships depends on early threat detection and subsequent engagements. Prosecution of cruise missile threats requires real or near real-time target information. FORCEnet has been identified as the construct that enables the communication of high fidelity data across the battlefield. The ability to provide a commonality of services and easier integration of upgrades is accomplished by enforcing Open Architecture (OA) as a design principle in the development of systems, as well as during legacy system technical refreshes.

The continuing development of FORCEnet will eventually lead to dependence on distributed weapons and sensor nodes. Collectively, these systems will have access to greater resources, which will provide a choice of multiple, semi-automated engagement options with faster response times and earlier intercept times. This report will examine six scenarios with particular emphasis on three current examples of geographically separate integrated fire control (IFC) capabilities and how they may be improved through the FORCEnet umbrella (Young, 2005).

B. PROBLEM ASSESSMENT

Fusion of the FORCEnet information architecture and an OA functional domain model poses specific challenges. Some of these challenges are extracted from Chapter 5 of *FORCEnet Implementation Strategy* (Committee on the FORCEnet Implementation Strategy, 2005):

- The process and tools for translating FORCEnet operational concepts into products, services, and warfighiting capabilities have yet to be fully developed.
- The number of unique interfaces that must be maintained need to be carefully selected and kept to an absolute minimum, or evolution will be hindered by expensive and lengthy integration and testing. One way to do

this is to require that systems partition common functions in a standardized way.

- There has been minimal effort in attempts to characterize how FORCEnet will function in terms of network management, data flow, traffic control, nodal performance, or data access. This information is required to engineer the FORCEnet network management system (Committee on the FORCEnet Implementation Strategy, 2005).
- Command and Control (C2) responsibilities as well as firing authority from remote will need to be addressed. Remote fire procedures and practices will need to become part of the chain of command concept of operations (CONOPS).

In addition to the above integration challenges between FORCEnet and OA, Information Assurance (IA) considerations must be implemented to prevent data compromise through security breaches. Additional security designs will need to be implemented including unauthorized access detection and isolation of a compromised subsystem, re-distribution of workload after a system is down or compromised, and authentication of the message or data source.

The purpose of this project, as stated in the Open Architecture as an Enabler for FORCEnet Statement of Work (SOW, Appendix A), is to address the above challenges. The focus of this report is on the development of a conceptual architecture model that integrates FORCEnet architecture components with the technical requirements of PEO IWS 7's OA functional architecture shown in Figure 1. Furthermore, the conceptual model is evaluated against the model in Figure 1 to analyze and determine its validity. This analysis serves as the basis for providing recommendations for improvement to PEO IWS 7 with regard to its OA functional domain model.

Current system and legacy deficiencies will be identified as well as constraints inherent in the operational environment in order to characterize, understand, and bound the problem space. Relevant operational imperatives are translated into system engineering structures such as concepts, functions, requirements, and solutions necessary to develop the concept.

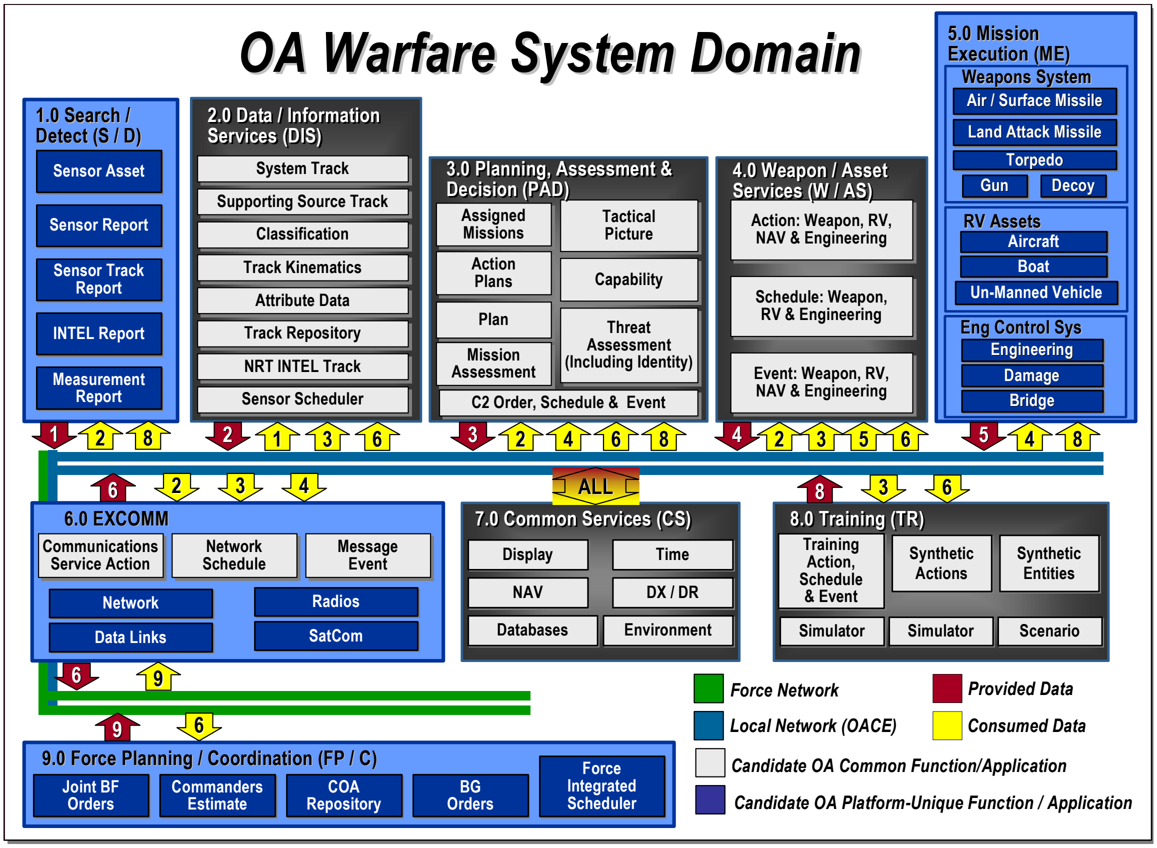


Figure 1. PEO IWS 7 functional architecture (The Critical Network Centric Warfare Enabler, Rushton, 2004).
This architecture is expected to simplify FORCEnet implementation.

IFC capabilities are then introduced during the development of principles for the design and architecting of OA and FORCEnet. Design principles will consider known limitations and constraints of the operational environment such as communication challenges and operator interaction. Communication challenges include unreliability, ad hoc mobile networks, and limited bandwidth. The development of a vision, architecture, and conceptual framework that addresses the problem space is based on the design principles for a distributed system. Automated decision aids will be used to manage warfare resources for collaborative operations.

Next functional representations and system models are developed to express automated resource collaboration concepts and solutions. The final step includes the analysis of the following key capabilities:

- Data fusion techniques and algorithms
- Resource management scheduling and optimization methods
- Weapon and sensor management
- Engagement functionality, initialization, and control
- Situation prediction and war game scenarios
- Tactical planning and battle management
- Opportunities for application of fuzzy logic and neural networks
- Allocation of tasking to people or software
- Information assurance against cyber attacks and for data integrity

Further, this report concentrates on CMD with emphasis on the following three IFC scenarios (Young, 2005):

- Precision Cue – an indication of a possible threat is received from a remote source.
- Launch on Remote – remote sensor data is used to initiate missile launch without holding a local track.
- Preferred Shooter Determination – the optimum weapon from a group of warfare units is selected to intercept a threat.

C. GENERAL ASSUMPTIONS

The following general assumptions were documented as part of the scoping and bounding of the project:

- Threat environments are in both blue water and littoral areas.
- Threats are Anti-Ship Cruise Missiles (ASCM) that can be launched from air, sea, and land.
- Bandwidth and communications pipelines can support real-time data transfer and sensor reporting.

- ASCM raids of 10 maximum per scenario.
- A_o of 98% of the ASCMD system.
- Participating units automatically become part of the theater defense network through a standard credential verification and validation process.
- CMD provided for US Navy assets only.
- System costs are outside the scope of this effort.
- Only US Navy sensors and weapons.
- No open source data for the performance of the PEO IWS architecture was available; therefore, the authors simulated and documented the simulation. This data was held constant except where noted in the simulation of the authors' proposed architecture.

D. RESULTS

An ASCMD simulation model was developed to test both PEO IWS 7's current architecture and the authors' proposed architecture. The simulation model is the compilation of the diagrams developed and analysis performed in the Design and Analysis section. The results show that the proposed architecture performed significantly better than the current architecture in the following: mean number of re-engagements, mean Electronic Warfare (EW) success and IA kills, mean interceptor kills, and mean leakers allowed. These improvements are attributed to the Observe-Orient-Decide-Act (OODA) loop that was added to the simulation model. Results and conclusions are discussed in the Findings and Recommendations section.

E. REPORT ORGANIZATION

This report is organized into five main sections: Introduction, Literature Review, Technical Approach, Design and Analysis, and Findings and Recommendations. The Introduction section provides the background for the project and assesses the problem to be analyzed. The goal of the project is to create a conceptual architecture that combines FORCEnet with OA for CMD. The Literature Review section covers the current state of

FORCEnet, OA, IFC, and CMD efforts. The research in this section provides a foundation for the Technical Approach.

In the Technical Approach section, the System Engineering Design Process used throughout the project is discussed, stakeholders are identified, and the problem space is characterized. The Design and Analysis section covers the analysis of capabilities key to the ASCMD concept, compares the current and proposed high-level architecture concepts, defines the battlespace through CMD and tactical scenarios, provides a detailed functional design analysis of the proposed architecture, and closes with a simulation model that encompasses all analyses herein. This section breaks the analysis effort from the highest-level to the lowest, most detailed level. The Findings and Recommendations section provides conclusions from the simulation model, a final overview of the proposed architecture, and outstanding issues for further study.

II. LITERATURE REVIEW

The team conducted research on the concepts of Open Architecture, FORCEnet, Integrated Fire Control, and Cruise Missile Defense. The purpose of the research is to define each concept, determine existing capability gaps and risks inherent within each concept, and investigate what work has been done to date for each. This section documents the research information found for the concepts above, and leads the reader into the Technical Approach section.

A. OPEN ARCHITECTURE

Definitions for OA vary depending on the person's point of view and organizational philosophy. These definitions vary in scope and content but all have the same general idea of the OA concept. To better understand what OA is, the following key components of OA are defined to establish a common lexicon:

- Architecture. Architecture is the fundamental organization of a system, embodied in its components, their relationships to each other and the environment, and the principles governing its design and evolution. (American National Standards Institute (ANSI) / Institute of Electrical and Electronics Engineers (IEEE), 2000). More simply put, architecture is the human organization of empty space using physical materials.
- Open Systems. Open systems are systems that employ modular design, use widely supported and consensus-based standards for key interfaces, and have been subjected to successful validation and verification tests to ensure the openness of key interfaces (Open Systems Joint Task Force, 2006).

The Open Systems Joint Task Force (OSJTF) states that an open system is characterized by:

- Well-defined, widely used, preferably non-proprietary interfaces and protocols.

- Use of standards, which are developed and adopted by recognized standards bodies or the commercial market place.
- Definition of all aspects of system interfaces to facilitate new or additional systems capabilities for a wide range of applications.
- Explicit provision for expansion or upgrading through the incorporation of additional or higher performance elements with minimal impact on the system.

Next, three definitions of OA are collectively examined with the final goal of resolving these characteristics into one common definition. First, OA is an enterprise-wide, multifaceted business and technical strategy for acquiring and maintaining national security systems as interoperable systems that adopt and exploit open systems design principles and architectures (Mullen, 2005).

The second definition is provided by the Program Executive Office for Integrated Warfare System (PEO IWS): "OA is an architecture that employs common standards, across government and private industry, for key interfaces within a system." (Naval Surface Warfare Center Division, 2004). OA is the high-level technical structure that is designed in accordance with the principles of open systems to achieve mission requirements, functional commonality, and life-cycle supportability goals. Open systems attributes include use of public, consensus-based standards; adoption of standard interfaces and services; use of product types supported by multiple vendors; selection of sTable vendors with a broad customer base and large market share; interoperability with minimal integration; ease of scalability and upgradeability; and portability of applications and users (Strei, 2003).

The third definition, as defined by the Navy Open Architecture Enterprise Team (OAET), states that OA is a multi-faceted strategy providing a framework for developing joint, interoperable systems that adapt and exploit open systems design principles and architectures. This framework includes a set of principles, processes, and best practices that provide more opportunities for competition; optimize total system performance; are easily developed and upgraded; minimize total ownership costs; rapidly field affordable, interoperable systems; employ non-proprietary standards for internal interfaces; and enable component reuse (Shannon, 2006).

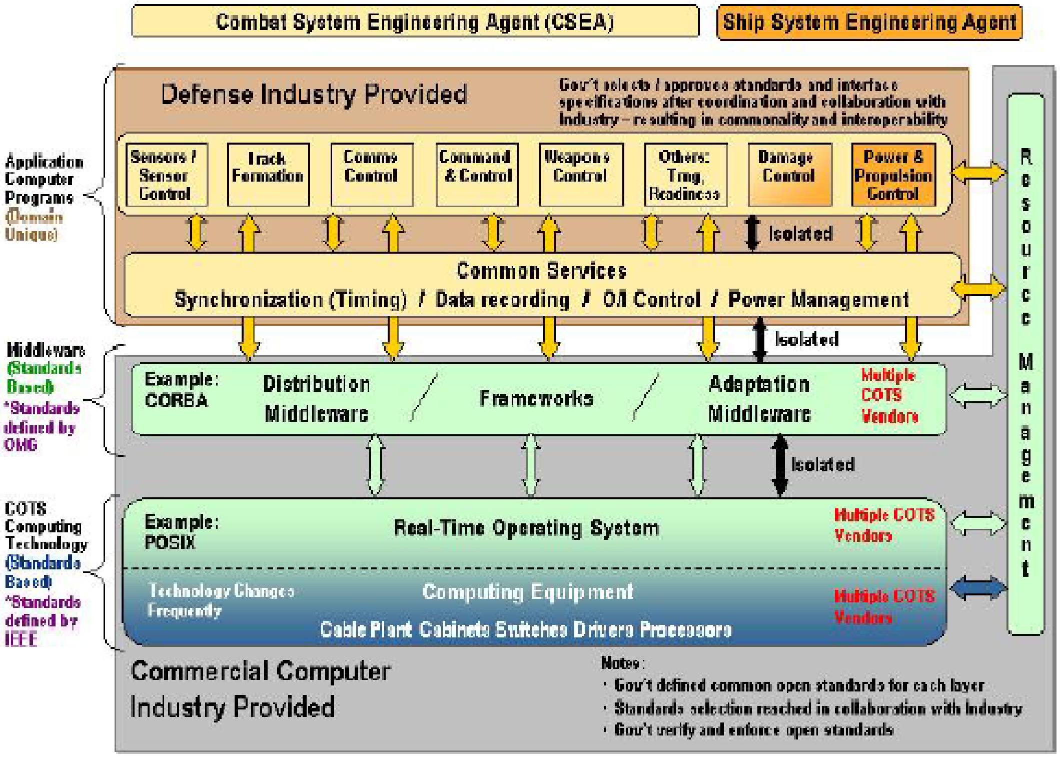
What the above OA definitions have in common is that all of them emphasize the open systems approach, the establishment of standards and design principles, and the implementation of technical architectures. A concern with the three definitions is that the implementation of OA is currently limited to key interfaces within a system. A legacy combat system cannot be reasonably expected to be fully compliant with open systems concepts, but it can benefit greatly by applying some of the key attributes of open systems especially when interfacing with other systems. Technology refresh is an example of leveraging OA into legacy systems. The establishment and adherence to internationally established standards is one of the key attributes of open systems. These standards evolve with time; new standards can also be introduced by a disruptive technology. A disruptive technology is a technology innovation, product, or service that eventually overturns the existing dominant technology or product in the market.

Standards-based architectures lessen the degree of control that the Department of Defense (DoD) can expect to exert. Hence, changes, fixes, and updates are under the vendor's control rather than the associated program office. This has a significant impact on system lifecycle support performance. The Chief of Naval Operations has cited five principles of OA that must be followed in order to reap its advantages (Mullen, 2006):

- Modular design and disclosure.
- Reusable application software.
- Interoperable joint warfighting applications and secure information exchange.
- Life cycle affordability.
- Encouraging competition and collaboration through development of alternative solutions and sources.

The result of our collection of these definitions is the subsequent resolution into a single common definition: *"Open architecture is a technical architecture that employs open specifications and international standards across government and private industry for key interfaces within a system. Furthermore, OA is implemented in accordance with the principles of DoD Directive (DoDD) 5000.1 Modular Open Systems Approach (MOSA) to achieve mission requirements, functional commonality across a wide range of* *systems with minimal change requirements and accomplishment of life-cycle supportability goals."*

PEO IWS 7 continues to refine open architecture policies and standards, as well as planning and implementation of OA into the surface and subsurface fleet. Figure 2 illustrates PEO IWS 7 Engineering Development Model (EDM), which runs on the OA Computing Environment and contains selected communication and specialized war fighting services and applications.



Combat System Engineering Agent (CSEA)
Ship System Engineering Agent
Defense Industry Provided
Gov't selects / approves standards and interface specifications after coordination and collaboration with Industry – resulting in commonality and interoperability
Application Computer Programs (Domain Unique)
Sensors / Sensor Control
Track Formation
Comms Control
Command & Control
Weapons Control
Others: Trng, Readiness
Damage Control
Power & Propulsion Control
Isolated
Common Services
Synchronization (Timing) / Data recording / O/I Control / Power Management
Isolated
Middleware (Standards Based) *Standards defined by OMG
Example: CORBA
Distribution Middleware
Frameworks
Adaptation Middleware
Multiple COTS Vendors
Isolated
COTS Computing Technology (Standards Based) *Standards defined by IEEE
Example: POSIX
Real-Time Operating System
Multiple COTS Vendors
Technology Changes Frequently
Computing Equipment
Cable Plant Cabinets Switches Drivers Processors
Multiple COTS Vendors
Resource Management
Commercial Computer Industry Provided
Notes:
· Gov't defined common open standards for each layer
· Standards selection reached in collaboration with Industry
· Gov't verify and enforce open standards

Figure 2. The visual high-level model of Total Open Systems Architecture (Open Architecture in Naval Combat System Computing of the 21st Century, Strei, 2004).
The model illustrates the nodal open systems architecture and the relationships between the commercial computer industry and the defense industry general and domain-unique hardware, middleware, and software.

B. FORCEnet

The Secretary of the Navy has set forth the Navy's guiding vision in a document entitled SEAPOWER 21 (England, Clark, Jones, 2007). This doctrine is comprised of three operational concepts: Sea Strike, Sea Shield, and Sea Basing. FORCEnet is the enabler of the naval transformation process, transforming the Navy and Marines into a combat organization that is effective against the future complex threats to the United States and coalition forces.

In addition, a fundamental shift has begun to occur in development of shipboard combat systems, transitioning from stove-piped designed systems into systems that interact seamlessly. The evolving Global Information Grid (GIG) is an overarching, interconnected system designed to collect, process, store, disseminate, and manage information. FORCEnet is the Navy's contribution to the GIG with complementary inputs from the Army and the Air Force with expected operational capability by 2020.

FORCEnet is the operational construct and architectural framework for naval warfare in the information age, integrating warriors, sensors, Command and Control, platforms and weapons into a networked distributed combat force (Naval Network Warfare Command, 2007). This definition provides the guidance for architecting FORCEnet. FORCEnet is essential due to shortened response times associated with more complex weapon engagements throughout the battlespace.

FORCEnet relies on two assumptions. The first assumption is that information technology will improve the data source availability, connectivity, and bandwidth. The second assumption is that the non-Navy elements of the GIG will be developed in parallel to provide the information and services necessary to provide coverage for existing gaps in the Navy domain.

In addition to the merits of FORCEnet, a number of risks inherent to the concept have been identified. First, there is a vulnerability to hostile information attack or exploitation. In addition, reliance on the use of information technologies may make the Command and Control less able to deal with natural disasters amid the possibility of degradation. The architecture is based on currently available technologies; however, the bandwidth capabilities may not keep pace with the ever-increasing amounts of information. Finally, the future processor capabilities within FORCEnet will reduce but not eliminate the decision-making load on the warfighter. These risks, as well as other risks, will need to be managed to prevent the collapse of this concept.

The following list identifies required capabilities for an effective integrated FORCEnet and OA structure. Some of these capabilities will need to be resident within the FORCEnet structure while others will be resident within other organizations:

- Robust, reliable communication to all nodes, based on the varying information requirements and capabilities of those nodes.
- Reliable, accurate, and timely location, identity, and status information on all friendly forces, units, activities, and entities/individuals.
- Reliable, accurate, and timely location, identification, tracking, and engagement information on environmental, neutral, and hostile elements, activities, events, sites, platforms, and individuals.
- Store, catalogue, and retrieve all information produced by any node on the network in a comprehensive, standard repository so that the information is readily accessible to all nodes and compatible with the forms required by any nodes, within security restrictions.
- Process, sort, analyze, evaluate, and synthesize large amounts of disparate information while still providing direct access to raw data as required.
- To depict situational information for each decision-maker in a tailored, user-defined, shareable, primarily visual representation.
- Distributed groups of decision makers to cooperate in the performance of common Command and Control activities by means of a collaborative work environment.
- Automation of certain lower-order Command and Control sub-processes and to use intelligent agents and automated decision aids to assist people in performing higher-order sub-processes, such as gaining situational awareness and devising concepts of operations.
- Information assurance.
- Functionality in multiple security domains and multiple security levels within a domain, and to manage access dynamically.

- Interoperability with Command and Control systems of very different type and level of sophistication.
- Functionality of individual nodes while temporarily disconnected from the network.
- Quick implementation of good decisions under conditions of uncertainty, friction, time pressure, and other stresses.

C. INTEGRATED FIRE CONTROL

Within the DoD, the US joint vision is "to build the most effective force for 2020, we must be fully joint: intellectually, operationally, organizationally, doctrinally and technically" (Young, 2005). IFC and CMD are inextricably linked and inherently joint. Their objective is to detach service-unique and platform-specific fire control radars from the weapon for Over-the-Horizon (OTH) CMD engagements. Integrated Fire Control (IFC) is the capability to engage targets by providing fire control solutions with real-time information from one or more non-organic sensors. The literature search revealed that IFC is a single component of the 2010 Theater Air Missile Defense (TAMD) operational concept. The TAMD central theme is that an overhaul of current Command and Control infrastructure and composition is needed to conduct warfare in geographically diverse areas. Historically, the German Air Force was the first user of IFC. The German Air Force and its Command and Control structure first used IFC during raids on London during World War II (WWII). Currently, Command and Control is isolated, with respect to the Area of Operation (AOR), not unlike the way engagements were conducted during WWII.

IFC is the central enabler of joint warfighting capabilities and pillars that include the Single Integrated Air Picture (SIAP), Combat Identification, and Automated Battle Management Aids (ABMA). In this context, the pillars coalesce to increase joint power projection through IFC and enforced interoperability. Through the achievement of IFC, weapons are not limited to local surveillance and fire control. IFC capitalizes on networked sensors, reduces horizon and terrain limitations, and improves the layered defense against stressing targets.

From an operational perspective, several IFC capabilities are identified (Young, 2005): Precision Cue (PC), Launch on Remote (LOR), Engage on Remote (EOR), Forward Pass (FP), Remote Fire (RF), and Preferred Shooter Determination (PSD). These scenarios leverage the distributed assets to achieve collaborative and automated engagements.

The Defense Science Board as early as 1994 studied CMD and determined that our adversaries can quickly and economically acquire Land Attack Cruise Missiles from several sources on the global weapons market and engage them against targets in the US or its allies. The findings of the Defense Science Board were accepted, which led to the establishment of the Joint Theater Air and Missile Defense Organization (JTAMDO) in 1997 (Defense Science Board, 2007). JTAMDO coordinated the efforts of the Commanders in Chief's (Combatant Commanders) and resulted in the operational architecture of the TAMD capability. With the TAMD established, the leadership and processes were in place such that the IFC capability could be jointly developed, evolved, and deployed among and across the services.

D. CRUISE MISSILE DEFENSE

1. Cruise Missile Defense Gaps

The advancements in missile guidance, stealth, and propulsion technologies have made cruise missile capability available worldwide and a prominent threat to naval forces. Cruise missiles are affordable and relatively inexpensive; they are known as the "Poor Man's Air Force," (Feickert, 2005).

Several capability gaps currently exist in US naval combat systems that preclude effective CMD prosecution. Current US naval weapon systems lack full interoperability across multiple platforms and full integration of detection, control, and engagement processes against single or multiple incoming enemy targets. Cruise missiles can avoid radar detection by flying at wave top altitudes, thus making detection difficult. Stealth technologies minimize cruise missile radar signatures, making the weapon systems detect, identify, track, and engage process a very challenging endeavor.

When a cruise missile is first detected, the reaction time, or radar detection to missile launch time of the weapon system, is very small. For example, a shipboard radar located 50 feet above sea level will detect a 50-foot altitude inbound subsonic cruise missile at 17 nautical miles from its own sensor. The radar horizon formula shows that the ship will have 90 to 100 seconds to react and engage a subsonic cruise missile. Reaction time gets even shorter when encountering supersonic cruise missiles. The Navy has made some improvements in the self-defense capability for surface ships against cruise missiles; however, gaining additional reaction time is the most difficult challenge that the Navy currently faces in order to defend against future cruise missile threats (General Accounting Office, 2000).

Another capability gap involves the separation of weapon engagement zones by theater commanders. This separation implies isolation of weapon systems where sensors working independently of each other reduce detection capabilities. A concept called Joint Engagement Zone (JEZ) is currently in development by DoD for theater war fighting. JEZ provides a SIAP to identify threats and an IFC system for offensive and defensive operations among all military services' sensors and weapons. The SIAP and IFC system are key enablers to effective and efficient CMD.

2. Cruise Missile Defense Defined as a Strike Group Problem

Most Navy ships have limited capabilities against CMD. Anti-ship Cruise Missiles (ASCM) are developed in large numbers with the latest in guidance and stealth technologies. Current ASCM's have incorporated advanced target seekers and fly at low altitudes. Additionally, these ASCM's are faster and may arrive in multiple raids due to their lower purchase price. These capabilities create significant detection and defensive challenges.

Addressing ASCM defense requires incorporation of the latest technology and full adherence to MOSA. In addition, existing capabilities have been limited to platform-specific defense assets without taking into account the collaborative capability of sensors and weapons within the Carrier Strike Group (CSG). The current CSG / Expeditionary Strike Group (ESG) watch configuration is separated between tactical operations and intelligence data due to information classification and originating sources. Communications between operations and intelligence watch standers occur via computer chat, email, or through voice reports. This type of communication adds more reaction time to the kill chain.

3. Threat Assessment

The 2005 *Congressional Research Service (CRS) Report for Congress in Cruise Missile Defense* (Feickert, 2005) indicates that the majority of cruise missiles are short range ASCM's. According to the CRS report, there are about 130 different types of cruise missiles in existence today and over 90 countries have the capability to produce the anti-ship version. ASCM's can be launched from air, land-based, and sea-based platforms. Many experts predict that cruise missile proliferation will increase in capacity and level of technological sophistication (Feickert, 2005). Detection of low observable ASCM's and reaction time continue to be areas of concern. The latest radars to be installed on future surface combatants will still lag technically behind next generation cruise missiles (Feickert, 2005).

Surface combatants have greater reaction time against ASCM's launched from land than those launched from undersea. The land-launched ASCM threat response time decreases with proximity to shore. For multiple raids, the current Command and Control system, Ship Self Defense System (SSDS), is not sufficient. The SSDS will require fire control quality data to react against the threat using shipboard weapon systems (Naval Network Warfare Command, 2007).

The next section uses the research above presented to establish the functional need and stakeholder requirements. The problem space characterization is explained in terms of deficiencies, constraints, and assumptions.

III. TECHNICAL APPROACH

A. SYSTEMS ENGINEERING DESIGN PROCESS

Figure 3 represents the iterative Systems Engineering Design Process (SEDP) that is followed through this report's entirety. This report only focuses on two phases of the SEDP: Problem Definition and Design and Analysis. The Problem Definition defines this project's stakeholders' needs and turns them into a functional need and requirements. The Design and Analysis phase decomposes the functional need and requirements into a functional design analysis of the proposed architecture that ranges from a high-level Value System Design to detailed use cases and flow diagrams. The Design and Analysis phase culminates with Modeling and Simulation of the functional design to determine its validity.

The authors use this design methodology to complete the requirements definition, requirements analysis, functional analysis, modeling and simulation, and a value system design in the development of the conceptual model. These activities will create a path for comparison between the authors' conceptual architecture and PEO IWS 7's architecture. Since this report only focuses on a conceptual architecture and not a physical one, no alternatives are considered. Decision-making of alternatives and implementation of the architecture are outside the scope of this report.

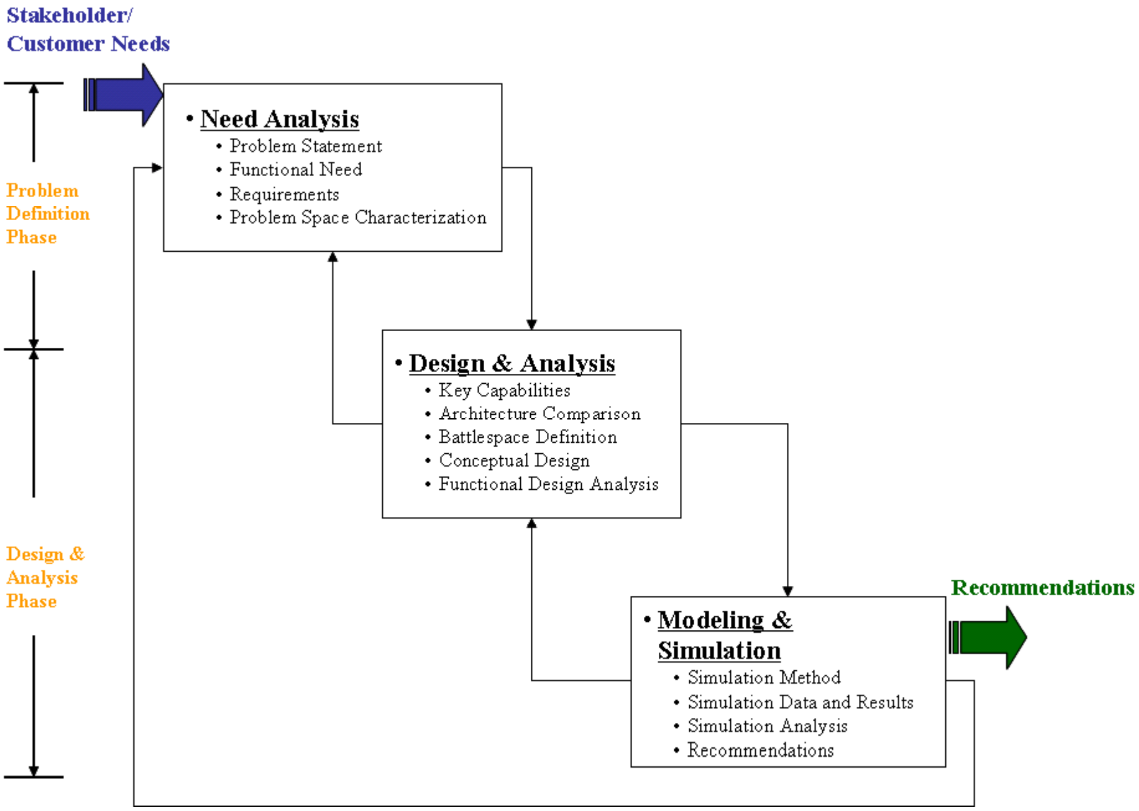


Figure 3. Systems Engineering Design Process.
This report focuses on Problem Definition and Design and Analysis

B. STAKEHOLDERS

The primary stakeholders for this project are John Michael Green of the Naval Postgraduate School (NPS), and the Program Executive Office (PEO) for Naval Open Architecture (PEO IWS 7). The secondary stakeholder is Mr. Adam Simonoff of the Naval Surface Warfare Center (NSWC) in Dahlgren, VA. Mr. Simonoff is a member of the Ship Self Defense System (SSDS) Systems Engineering Team, which supports the Navy Review Team for Open Architecture Combat system design (Simonoff, 2005). Mr. Simonoff served as the advisor for Information Assurance within the conceptual architecture model.

1. Customers

The ultimate customer of the conceptual architecture model herein is the US Navy and its battleforce commanders. The expectation is that this conceptual architecture will lay the foundation for a workable physical architecture. This architecture will improve warfare resource management, effective decision-making, ship self-defense, and mission execution.

2. Functional Need

The current Navy operations environment lacks network centricity. As of today, Intelligence, Surveillance, and Reconnaissance (ISR), Command and Control, and combat systems operate in a stovepiped manner. As explained by Commander Pat Roche of the Space and Naval Warfare Systems Command, each of these areas receives their data through individual communication networks that do not talk to each other (Roche, 2005). For example, ISR data comes from a common data link into an independent server that is directly accessed by users. The ISR data does not automatically correlate with Command and Control and combat systems servers, which may contain additional tracking information.

The functional need is to integrate ISR, Command and Control, and combat systems via networks to create a common operating picture. An integrated FORCEnet and OA model can provide the complete information needed for effective naval operations. The vision is that ISR, Command and Control, and combat systems servers would reside in a distributed services network that receives simultaneous data via GIG and other tactical data links such as Joint Tactical Radio System (JTRS), Joint Tactical Terminal (JTT), Global Command and Control System-Maritime (GCCS-M), Link 16, and others. The distributed services network would in turn enable users to access data across ISR, Command and Control, and combat systems areas. Identified enablers for the distributed services network are Extensible Markup Language, Internet Protocol version 6 (IPv-6), and distributed security.

3. General Characteristics of the FORCEnet Architecture

Chapter 5 of the book *FORCEnet Implementation Strategy* addresses six technical characteristics that are considered essential to achieving FORCEnet (Committee on the FORCEnet Implementation Strategy, 2005). Those characteristics are guaranteed end-to-end quality of service, bandwidth, information assurance, availability, redundancy and graceful degradation, an architecture that supports incremental deployment, and interoperability.

End-to-end quality of service refers to the capability of network warfighting nodes to deliver services needed by specific network traffic from end-to-end (Cisco, 2007). Bandwidth availability and expansion are required to process large amounts of data in a very short period. Information assurance is needed to reinforce the FORCEnet architecture against emerging threats such as cyber attacks. Availability, redundancy, and graceful degradation of network assets must be increased, monitored, and managed to allow for fast replacement in case of failures. An architecture that supports an incremental deployment allows new capabilities to be implemented with minimal impact to the combat system. Finally, the FORCEnet concept must employ common elements, standards, and protocols across its architecture to ensure interoperability.

4. Stakeholder Requirements

The stakeholders for this project tasked the team with developing a conceptual FORCEnet architecture that addresses interoperability and information assurance capabilities. The conceptual architecture must comply with the technical requirements outlined in PEO IWS's OA functional architecture, which are identified in Figure 1. The stakeholders also tasked the team with evaluating the validity of the OA architecture in Figure 1 via Excel and Arena simulations, and to compare it to the conceptual architecture. The design principles used in developing the model must take into consideration known limitations and constraints of the operational environment and be based on automated decision aids.

C. PROBLEM SPACE CHARACTERIZATION

FORCEnet encompasses all Navy command, control, and information-sharing functions necessary to ensure accurate, rapid, and secure transfer of information via the supporting warfighting capabilities listed in Chapter II of this report. Information is dispersed to the forces throughout the battlespace via the FORCEnet information network. Current capability gaps include a lack of common configuration and Fire Control Quality (FCQ) connectivity among platforms that is needed to achieve and maintain a robust cruise missile defense. FCQ is defined as data obtained with the sufficient accuracy and refresh rate to support engagement actions such as launch decision, guidance calculations, and engagement control that may involve sensor tasking or managing the data path (Young, 2005). While FORCEnet is commonly thought of purely in terms of added warfighting capabilities, FORCEnet supports enterprise-wide computing needs necessary for force planning, coordination, and theater-wide sustainment or warfighter operation and support. Table 1 provides a summary list of constraints and deficiencies documented during the problem space characterization.

Constraints	Deficiencies
Form, fit and function	Situational Awareness
Subsonic Threat	Force Planning/Coordination/Management
Maneuvering Threat	Data Latency
Fleet Deployment Tactics (Operational Area under consideration)	CONOPS (over- the-horizon)
Multiple Threat Environment	Lack of Sensor fusion
Counter-Countermeasures	Lack of Common Track Management
Seamless communications	Bandwidth
Unique function/platform	Share resources (chain of command issues)
Schedule/Time	Interoperability (or lack thereof) wrt /US/Allied/Other
Rules of Engagement	INTEL instead of ISR GIG
Sustainment- Joint training, as a constraint, couples with joint interoperability as a deficiency.	Communications gridlock
Cost	Training for Information Services
Current weapons and sensors	Information Assurance
Manpower	The OA warfare domain model features multiple independent entities.
	Lack of common message format
	Target track refresh rate

Table 1. Constraints and Deficiencies Summary.
This Table supports the discussion of deficiencies, capability gaps, constraints, and assumptions.

1. Deficiencies

Deficiencies exist in how the architecture influences the behavior of the weapon system. A thorough understanding of the behavior is required to construct the architecture so that accurate representations of a cruise missile defense system can be modeled and simulated. A desired by-product of improved knowledge of system behavior is that it may enable a more complete understanding of efficiencies and inefficiencies to improve the man-machine interfaces and to drive down the shipboard manpower required in an environment where resources are always constrained. This aspect of the system is especially crucial during cruise missile defense when all the weapon systems critical to the defense of the surface combatant are fully manned.

Currently, there is limited understanding of the combat system posture necessary for complete cyber attack defense. Some of the challenges are posed as part of IA and FORCEnet's capability to detect packets, perform passive detection of a compromised subsystem, isolation upon attack detection, redistribution of information to prevent balking upon restoration after the compromise is excised, and non-repudiation guarantee through multiple information assurance methods that include personal knowledge such as username and personal identification number, smart cards, biometric markers, or a combination of technologies (Schekkerman, 2005).

Dependence on web-based architectures increases the risk of cyber attacks because it is a vulnerability that is cost-effectively exploited through the patience and persistence of our adversaries. Thus, an ironclad IA policy and implementation is essential to effective Command and Control. Implementation of cyber attack prevention instead of the current process of attack detection will move our cyber enemies further outside of our rings of defense. The multi-layered defense is discussed in detail throughout this report.

The OA warfare domain model features multiple entities acting independently. The architecture requires greater interaction among model elements so that there is real-time correlation among intelligence collectors and distributed users. The currently deployed web-enabled command, control, and ISR tools must deliver sufficiently accurate and timely situational awareness as a part of FORCEnet's capability.

The Boyd Observe-Orient-Decide-Act (OODA) loop for C2 response is defined as an information strategy concept for information warfare developed by Colonel John Boyd (Luessen, 2003). The OODA loop does not enable rapid decision cycling, which increases battle force vulnerability and reduces survivability. The current C2 system response is unsTable which means that, from classical queuing theory, the mean service time is greater than the arrival rates of high raid intensity, the C2 decision system simply will not “keep up” with the “arrivals.” Luessens’s concept of the OODA loop is missing the critical element of prediction based on distribution functions of uncertain input parameters. To reflect uncertainty in the C2 response, the OODA loop needs a prediction function inserted into a revised Observe-Orient-Predict-Decide-Act (OOPDA) loop. In turn, the prediction element is a key function missing from current modeling and simulation efforts, which negatively impacts the task force commander’s situational awareness and his ability to effectively plan and coordinate combat forces during cruise missile defense. Where accurate data is available or synthesized, prediction functions should increase the accuracy when modeling system behavior through simulation.

Communications gridlock, or data latency, which refers to the inefficient “flow” of data among entities such as organic and distributed sensors, weapons, ordnance, and delivery platforms, is prevalent within current Areas of Operation (AOR). In event graph conceptual model lexicon, these entities are collectively called nodes that are strung together by suboptimal placement of “arcs” or connections. Other inefficiencies that contribute to communications gridlock include different message formats, low target track refresh rates, shared C2 resources, and insufficient common track management that limits seamless interoperability among platforms and joint forces. These factors additively compromise ship self-defense by limiting the ability to integrate communications, sensors, and intelligence collections with real-time track data.

2. Constraints

Constraints are inherent limitations in resources, technologies, or other limitations that prevent implementation of reasonable and instantaneous solutions. Assumptions enabled the authors to take a snapshot of dynamic events and thereby change the problem from a continuous dynamic state into a series of discrete events. Constraints and assumptions co-exist, through measured interactions, forming the balance necessary to make fundamental design implementation, tactical decision optimization, and improvement through models of the highest degree of fidelity.

The existing inventory of weapons and sensors is a constraint; no additional weapons or sensors will be added to the inventory nor will any be reduced or eliminated.

Current sensor cueing and data fusing methods remain in effect. The weapons and sensors, in their current block configuration, establish the physical form, fit, and function but not the architecture to counter cruise missile threats, regardless of their speed, kinematic capability, raid size, or countermeasures employed. In addition, some of the more advanced cruise missiles have a passive radar capability that allows them to detect and lock to an active jamming countermeasure, making it resistant to electronic countermeasures employed by a defending platform (Defense Threat Information Group, 2005).

Certain platforms would have unique capabilities or functions that other platforms in the task force may not have. The lack of seamless communications among the multiple interfaces that compose these capabilities poses a constraint.

Rules of Engagement (ROE) are a constraint because the existing CONOPS, specific to the weapon, countermeasure, or AOR, remain in effect. The task force commander must make rapid and accurate decisions relative to positive target identification, availability of weapon/platform assets, and collateral damage estimates. Multiple threat environments encompassed by sea, air and land-launched cruise missiles, along with proliferation of these technologies throughout the world, makes it more challenging for the task force commander.

Inbound ASCM warning makes time a constraint. Warning time with regard to cruise missile defense is very limited, and depends on the range of detection with an average time of 2 to 2.5 minutes for a subsonic threat, and approximately 20 to 30 seconds for a supersonic threat. Complexity is added to the equation of a maneuvering threat by means of unpredicTable flight paths.

Bandwidth is a constraint but not the focus of this report. Similarly, processing speed and capacity are expected to be continuing constraints as they are limited resources.

Finally, offensive capabilities can be obtained at a much lower cost than defensive capabilities, according to a Congressional Research Service Report for Congress (Hichkad 2005).

3. Assumptions

In the context of a combat system, IFC requires that ordnance be considered in the same reference frame for all fused organic and distributed sensors, weapons, targets, and delivery platforms regardless of the environment or geography. In other words, active use of a common reference frame reduces the probability that the force commander makes imperfect or imprecise decisions based on an unsymmetrical or myopic view of the battlespace.

It is predicted that the global commercial market will continue to drive information technologies into the distant future. These technologies include interactive products and services that include operating systems and applications directly catering to the communication, information sharing, financial, consumer, and entertainment sectors. The commercial market has embraced the Open Architecture Computing Environment (OACE), but it is expected that mere compliance with OACE specifications alone will not make FORCEnet truly open.

FORCEnet enablers are technology-centric and dependent. Because of high reliability and built-in redundancy, future systems can reduce or eliminate single point of failure scenarios. Such high reliance poses significant C2 and warfighting risk. For instance, theater communications are highly variable, communication in the available frequencies vary continuously, and unpredictably, due largely to environmental conditions over which there is no control. This is a separate issue from frequency spectrum availability and management. Similarly, modeling of the environment with respect to FORCEnet functionality, while worthy of study, is beyond the scope of this paper. In the near term, it is expected that the Navy will trail the commercial sector in OA applications for a variety of reasons including organizational inertia, legacy operating systems, and applications.

Improved sensors, sensor fusing, and integration of ISR data will increase data flux. Adapted from chemical instrumental analysis, data chromatography is the process of separating small amounts of usable information from large and mostly trivial amounts of data. Increased data flux increases the burden on C2 due to greater decision cycle times. While improved sensors or fusing capabilities adds information, it also adds uncertainty, which must be estimated and represented in the decision-making process. Disregarding uncertainty due to data overload, erroneous estimation, interpretation, or application of uncertainty will lead to combination of incorrect, imprecise, or slow decisions that may result in materiel losses. Uncertainty and the measurement, application, and response to it are essential to effective situational awareness (SA).

The functional need and problem space characterization lay the foundation for the next section. The efforts throughout the Design and Analysis take the functional need and expand it into workable systems engineering diagrams that will lead into the ASCMD simulation model. The deficiencies and constraints found in the problem space are taken into account during the analysis.

THIS PAGE LEFT INTENTIONALLY BLANK

IV. DESIGN AND ANALYSIS

In this section, the current PEO IWS 7 OA functional architecture is compared with a proposed architecture model developed by the team. The proposed architecture is first described at the highest level, with detailed decomposition occurring along the way until the lowest level is reached. This section is divided into six analyses in the following order: key capabilities, comparison between current and proposed OA functional architectures, battlespace definition, design principles, conceptual design, functional design, and the proposed ASCMD simulation model.

A. KEY CAPABILITIES

The key capabilities in this section, identified on page 8 of the Introduction, are the major considerations the team will analyze to develop the proposed ASCMD functional architecture. The following capabilities are analyzed immediately below: situation prediction and wargaming, tactical planning and battle management, opportunities for application of fuzzy logic and neural networks, information assurance, and allocation of tasking to people and/or software. The proposed ASCMD simulation model section analyzes the following capabilities: data fusion techniques and algorithms; resource management scheduling and optimization methods; weapon and sensor management; and engagement functionality, initialization, and control.

1. Situation Prediction and Wargaming

Situation prediction is an extrapolation of the analyses to a future point in time. It is the projection of the current situation, which is developed by the various situation assessment and evaluation functional sets, into the future (Young, 2005). The purpose of situation prediction is to estimate the enemy course of action (COA) and potential impact of the battleforce’s planned actions, to predict real-time, near real-time, and non-real-time operational situations.

Some functions are combined to predict the cruise missile defense's behavior as time progresses. These functions include environment prediction, warfighting resource projection, wargaming, and force projection. The environment prediction predicts the environment situation for the area of interest (AOI). The warfighting resource projection is the status and capability prediction of sensors, weapons, and warfighting units' performance. Wargaming predicts the threats; identifies, evaluates, and prioritizes blue force COA; evaluate effects of C2 inputs on blue force COA; predict and evaluate enemy COA and intent; and analyze the historical trend. Force Projection is a prediction of Force Readiness. It is a prediction of overall force readiness and capabilities. All these functions are taken at once in a data fusion level, providing a solution to the cruise missile defense observation.

2. Tactical Planning and Battle Management

Tactical planning is a critical ingredient towards the identification of mission critical resources and identification of strategic goals. An approach for the development of strategic objectives is presented in Figure 4 using the Strategic Creative Analysis (SCAN) process (MBA Tool Box, 2007). SCAN is a process for strategic planning, decision-making, and analysis that supports the development of an effective and efficient battle management plan. The twelve steps required for the SCAN process are depicted in Figure 4. Step 3 requires the selection of the Top Rank Objective (TRO) that is going to help focus on the most important objectives.

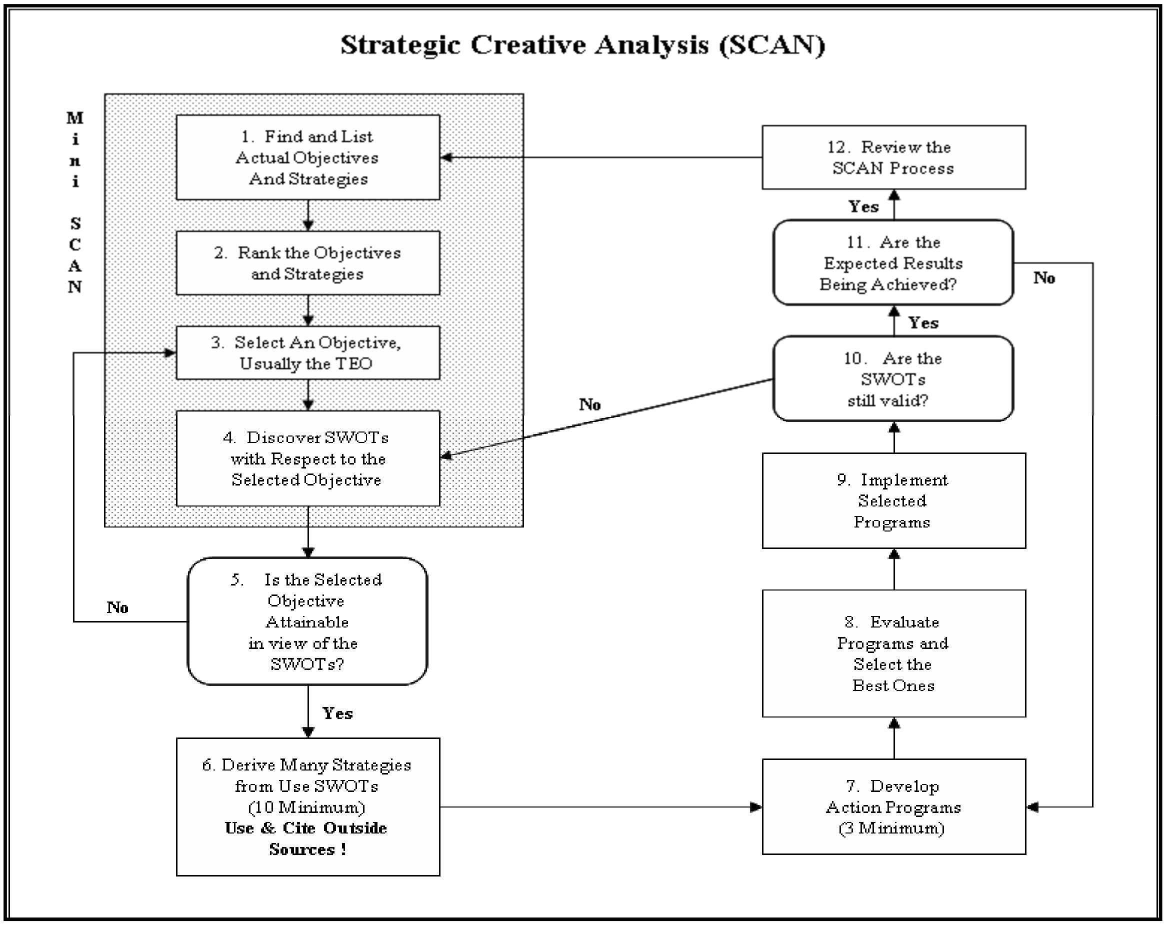

Strategic Creative Analysis (SCAN)
Mini SCAN
1. Find and List Actual Objectives And Strategies
2. Rank the Objectives and Strategies
3. Select An Objective, Usually the TEO
4. Discover SWOTs with Respect to the Selected Objective
5. Is the Selected Objective Attainable in view of the SWOTs?
No
Yes
6. Derive Many Strategies from Use SWOTs (10 Minimum) Use & Cite Outside Sources !
7. Develop Action Programs (3 Minimum)
8. Evaluate Programs and Select the Best Ones
9. Implement Selected Programs
10. Are the SWOTs still valid?
No
Yes
11. Are the Expected Results Being Achieved?
No
Yes
12. Review the SCAN Process

Figure 4. Strategic Creative Analysis (SCAN) process. The SCAN supports the development of battle management planning (Winer, *MBA Tool Box*, 2007).

Step 4 introduces another tool called the Strengths, Weaknesses, Opportunities, and Threats (SWOT) analysis. Once the TRO has been identified, the SWOT analysis can be used to help in the pursuit of that objective or mission objective. SWOT is defined as Strengths: attributes of the platform that aid in the achievement of the objective; Weaknesses: attributes of the platform that are detrimental to achieving the objective; Opportunities: external conditions that are beneficial to achieving the objective; and Threats: external conditions that are detrimental to achieving the objective. The SCAN analysis is an interactive process that needs to be repeated because attributes of the platform and outside conditions could change overtime. In addition to the SCAN and SWOT, analysis processes also needed are Automated Battle Management Aids (ABMA) tools that are required to determine the best use of Command, Control, Communications, Computers, and Intelligence (C4I), weapons, and sensors that facilitate the development of a Battle Management Plan. See Table 2.

Force Planning	Weapon System Capabilities	Mission
Air defense control plan with decentralized execution	Weapon scheduling	Area of Operations (AOR)
Defended asset list	Probability of Kill	Rules of Engagement (ROE)
Distributive collaborative planning	Sensor fusion and sensor planning aids	Threat assessment
Force allocation	Weapons inventory	Concept of Operations (CONOPS)
Force scheduler	Interceptor guidance control	Post-mission analysis, report, replay, and debriefing tools
Frequency management plan	Sensor capability areas	Wargaming Course of Action (COA) and rapid replanning
IFC priorities	Range	Identification and location of friendly forces
Platform capabilities and limitations	Preferred Shooter Determination	Mission logistics support
Tracking and prioritization of warfighting resources	Organic and non-organic support	

Table 2. Automated Battle Management Aids tools.
ABMA tools are needed to aid with tactical planning efforts required for the implementation of an effective CMD strategy.

To effectively conduct CSG, ESG, and Carrier Air Wing (CAW) operations, the authors developed a CMD operations diagram that would function as a two-layered Operations Management Center (OMC) concept as depicted in Figure 5. The Platform OMC operates under a set of policies and ROE's that are delegated by the CSG OMC under the authority of the Composite Warfare Commander (CWC). Under established ROE's, each platform is accounTable and is given full responsibility for the deployment of weapons for self-defense or the protection of other platforms within a particular area of responsibility (AOR) boundaries. These units also receive CWC Force commands via the OMC such as the commands listed in Table 3.

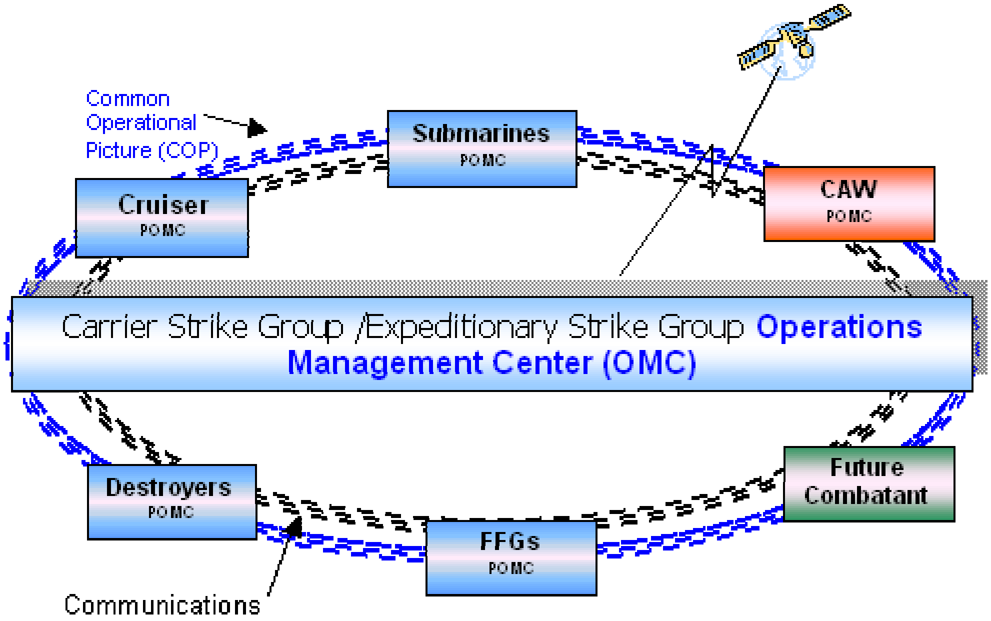


Figure 5. Operations management center concept.
The first layer is the CSG Operations Management Center and the second layer is the Platform Operations Management Center.

CSG/ESG Operations Management Center: This is a single operations center that integrates all Information, Surveillance, and Reconnaissance (ISR) sources of information and generates a Common/Composite Operational Picture (COP) that is maintained using surveillance data from all available sensors. The CSG/ESG OMC is linked to strategic information sources and disseminates to the associated warfare directorates on each unit. The CSG/ESG OMC under the direction of the CWC **(**Ready-for-Sea Modular Course & Handbook, 1999) is responsible for the management and oversight of all CSG resources (ISR sensors, mission planning, ROE's, ordnance inventories, platform status, communications, and logistics).

Based on the perceived cruise missile threat, the CSG/ESG OMC can automatically issue mission updates using force-centric or unit-centric commands listed in Table 3 via secured high-speed and high-bandwidth communication networks to counter single or multiple threats with one or multiple platforms and missiles. Some of these Force Commands include selection of the designated preferred shooters, the designated engagement support platform (fire control data providers, the weapons in-flight control and terminal homing support providers) and other types of engagement orders as needed.

ENTRIC / UNIT-CENTRIC CMD COMMANDS	ORIGINATOR
COMMUNICATIONS PLAN/INTEL REPORTS	**OMC**
EXECUTE ENGAGE ON REMOTE	OMC/REMOTE
EXECUTE FORWARD PASS	OMC/LOCAL/REMOTE
EXECUTE LAUNCH ON REMOTE	OMC/REMOTE
FORCE INTEGRATED SCHEDULER	**OMC**
FREQUENCY MANAGEMENT PLAN	**OMC/Local/Remote**
IN-FLIGHT MISSILE CONTROL/GUIDANCE RELAY	OMC/LOCAL/REMOTE
ISSUE PRECISION CUE	OMC/LOCAL/REMOTE
LOGISTIC PLAN/UPDATES	OMC/LOCAL/REMOTE
NEW TRACK REPORT	OMC/LOCAL/REMOTE
PREFERRED SHOOTER / WEAPON/TARGET PAIRING	OMC
PROVIDE MISSION/KILL ASSESSMENT	OMC/LOCAL/REMOTE
PROVIDE ROE/MISSION PLAN/UPDATE	OMC
REMOTE WEAPONS FIRE ORDER	OMC/REMOTE
SCHEDULE A SENSOR	**OMC/Local/Remote**
TARGET ILLUMINATION	OMC
THREAT ASSESSMENT REPORTS	**OMC/Local/Remote**

Table 3. Force-centric and unit-centric commands.
Mission updates can be automatically issued through these commands.

These commands could be automatically or manually processed and communicated to the designated platform based on ROE's (Young, 2005). Even though the CWC can issue tactical commands via the OMC as described above, the designated platform (CG, DDG, FFG or SSN) retains control authority over all organic warfare assets (sensors, hard kill and soft kill weapons, illuminators, and communications).

Figure 6 provides an example of a typical chain of command structure with Warfare Commanders assigned to different mission areas. This model can also be applied to an ESG with an LHA/LHD substituting for the carrier. Each Warfare Commander is assigned to the platform best suited for the mission as noted in Figure 6.

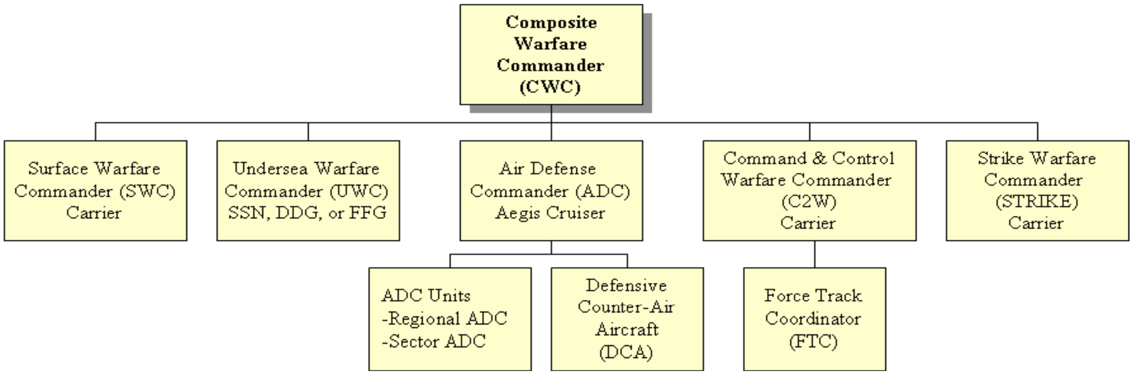

Composite Warfare Commander (CWC)
Surface Warfare Commander (SWC) Carrier
Undersea Warfare Commander (UWC) SSN, DDG, or FFG
Air Defense Commander (ADC) Aegis Cruiser
Command & Control Warfare Commander (C2W) Carrier
Strike Warfare Commander (STRIKE) Carrier
ADC Units -Regional ADC -Sector ADC
Defensive Counter-Air Aircraft (DCA)
Force Track Coordinator (FTC)

Figure 6. Chain of command structure for Warfare Commanders assigned to different mission areas.

The Air Defense Commander (ADC) (call sign AW) is typically assigned to the commanding officer of a USS TICONDEROGA (CG 47) class cruiser operating the Aegis Weapon System. A second Aegis cruiser may act as an alternate AW to allow for 24 hours of operation (12 hours on and 12 hours off). The ADC units are deployed throughout the region or in sectors of interest.

The Force Track Coordinator (FTC) manages all organic and non-organic communications in addition to all Tactical Data Links (TADILs). These include Link 11 (TADIL A and B), Link 4A (TADIL C) and Link 16 (TADIL J). Link 11 provides a standard message format for exchanging digital information among airborne (TADIL-A) as well as land-based and shipboard (TADIL-B) tactical data systems. Link 4A is used to provide vector commands to fighters. Link-16 is DoD's primary tactical data link for command, control, and intelligence, providing critical joint interoperability and situation awareness information.

The CWC, via the OMC, will determine the battleforce readiness condition level that will be in place to allow for each ship to perform offensive and defensive functions necessary to counter cruise missile threats, keep required operational systems continuously manned and operating, perform other command and control functions relevant to the cruise missile threat, and accomplish urgent underway planned maintenance and support functions. The battleforce readiness condition level can last from 24 hours to two months based on the perceived threat level.

3. Opportunities for Fuzzy Logic and Neural Networks

There are continuing efforts in minimizing the reaction time for cruise missile defense. One area that has received more attention is in computational intelligence or "intelligent decision-making system" (Pal, Mudi, 2003). The concepts of neural networks and fuzzy logic are born out of this field, where the approach originates from studies of the central nervous system and human brain. With the increasing complexities of the systems of today and those proposed in the future, it is becoming more difficult to predict and explain the behavior of these complex systems with current methods. New techniques of system control and decision-making are being studied to determine if these concepts can indeed curtail system complexities that currently exist and for future applications.

The concept of neural networks comes from medical research into the human central nervous system. In the central nervous system, neurons collect signals from others through structures called dendrites. The neuron itself sends out electrical pulses through a strand called an axon and each axon is connected to another neutron/dendrites combination. What is of interest in regards to these elements that make up the central nervous system is their information processing capabilities. This type of system performs functions collectively and in parallel by the neurons (units) rather than in a task/subtask hierarchy. In addition, it has been shown that this system has the ability to adapt (learn). Learning is accomplished by electrical activity, which inhibits or excites the surrounding neurons. It is this capability to solve problems collectively and adaptively that can be integrated into the ASCMD functional architecture to eliminate some of the organic decisions made in the kill chain.

Figure 7 (Stergiou, Siganos, 2007) below depicts a simple artificial neuron. This neuron can have multiple inputs and one output. These neurons can be grouped together to form artificial neural networks. Engineers have been studying this concept and have developed artificial neural networks, which can be used to detect trends and extract patterns from data; due to their open structure they can be applied to non-linear applications such as tracking highly-maneuverable targets. In addition, due to the ability to detect trends and patterns, these networks could be used for pattern recognition as seen in radar systems, face recognition, sequence recognition (speech), process control, and data mining.

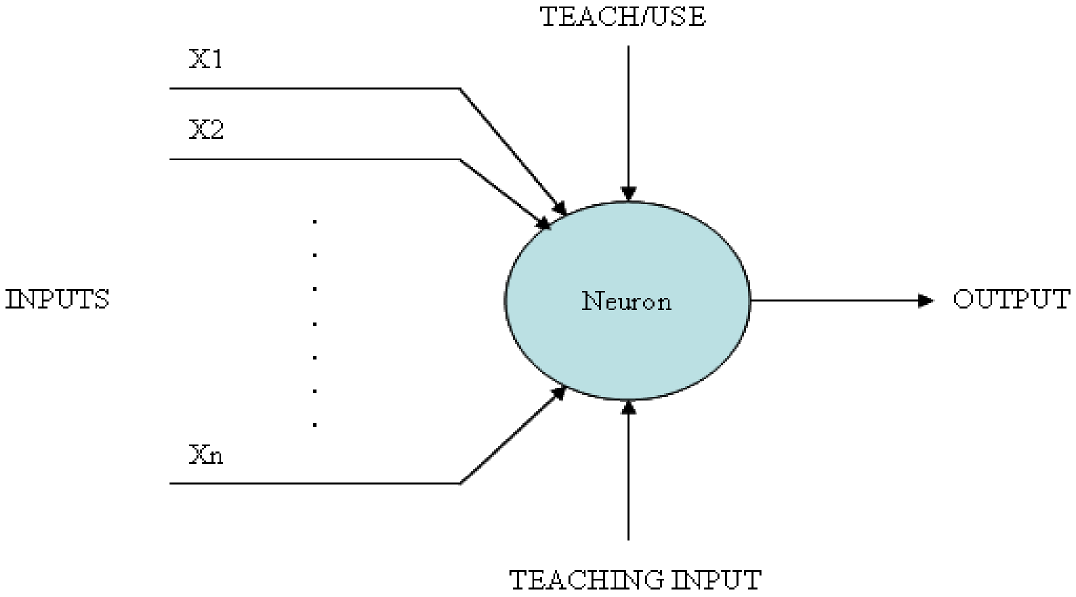


Figure 7. Artificial neuron (Stergiou, Siganos, 2007).
These neurons can be grouped together to form artificial neural networks.

Artificial neural networks take a different approach to problem solving than conventional computers in use today. Today's computers use an algorithm encompassing a set of instructions. The solution to a problem must be known as well as the steps necessary to solve the problem. This concept limits the use of today's computers to those problems and solutions, which are known today. Problems that deviate from what is known cannot be solved with conventional processing capabilities.

As stated earlier, neural networks process information and execute problem solving in ways that are similar to the brain. Processing elements, neurons, work in parallel to solve problems. It has been shown that these neurons can be trained and can adapt based on the input received. In *Neural Networks* (Stergiou, Siganos, 2007), it was shown that if one would define a collection of training modes for a neuron, then 1-taught set of patterns would cause the neuron to fire and 0-taught set of patterns would prevent the neuron from firing. If the neuron was presented with an undefined pattern, it could "compare" the undefined pattern with the defined patterns produced from the taught set of patterns to produce a defined output pattern.

Presently, artificial neural networks are being studied to determine if these concepts could be applied to highly-maneuvering threats. Highly-maneuverable target motions can be difficult to predict. When tracking these types of targets it is difficult to determine where in space the object will next occupy. These types of targets are said to be non-linear in nature and as such can change from the assumed motion model.

A study was conducted by The Space and Naval Warfare Systems Center San Diego in the area of real-time modeling of maneuvers. The Center has developed an artificial neural network multiple model tracker, which has shown to predict the correct system states of a target as it is maneuvering. The model uses the concepts of neural networks to handle the nonlinearities of these types of targets.

Fuzzy logic originated with studies of the human brain and its ability to receive imprecise inputs, evaluate these inputs, and develop an accepTable output. Fuzzy logic is used by people every day. For example, when driving in traffic it is usually optimum and safest to drive with the flow of traffic; however, defining the specific instructions for "driving with the traffic" would be difficult. A number of inputs are received by people as they are driving in traffic, most of which is fuzzy or imprecise at best. Some inputs received are drivers that weave in and out of traffic, drivers going faster than the speed limit, determining how many drivers are ahead, trucks slowing down lane traffic, and number of police officers using speed radars. All of this is imprecise input but people have the ability to take this fuzzy information and determine if it is safe to drive with the flow of traffic.

Fuzzy logic exists in every day items like self-focusing cameras, washing machines, automobile engine controls, subway control systems, and other applications. Fuzzy logic analysis and control can be mimicked in machines to perform tasks somewhat like humans. The method is divided into three main areas; input, processing, and output. For the input, determine what measurements or assessments of the condition of the system are required. Here may be one or multiple inputs, depending on the application. A unit receives one or more stimuli in the form of a measurement or some other assessment of a condition within a system. As an example, the temperature would be the input for a home air conditioner. Then process these inputs according to “If X and Y Then Z” rules. These rules are human-based, expressed in plain language, and need not be as precise as an algorithm found in a conventional computer system.

IF/THEN rules are developed in the form of If *variable* Is *set* Then *action*. Using the home air conditioner example, one of the rules could be: IF temperature IS very cold THEN stop fan. There will be a number of these rules developed into a fuzzy algorithm to be executed by a conventional computer.

Another allocation can be made by using averaging and outputs, where weights are assigned to each sensor’s output based on the sensor’s performance, as well as an averaging and fusing of all the sensor outputs into one output. This output is the command the system uses to adjust itself in response to a change in its environment.

One area of application within fuzzy logic is target tracking. There is ongoing research that examines fuzzy logic and fuzzy inference systems in the use of multiple-sensor integration. During certain operating conditions, one sensor may provide more reliable data than others may. Personnel at the Southern Illinois University, Department of Mechanical Engineering and Energy Processes (Mahajan, Wang, and Ray, 2007) developed a generic model which placed three different sensors on a cantilever beam. The characteristics sensed by these sensors were used as input measurements to a Fuzzy Inference System. The outputs of the Fuzzy Inference System were weights assigned to each sensor measurement. These weights reflected the confidence in the sensors performance. The data from the three sensors were fused together by normalizing with their weights. Each individual sensor error was measured and compared to the error of the fused error. It was found that the model delivered an accurate estimation based on fused data.

To minimize reaction time to an inbound subsonic threat, computational intelligence will have to exist within the architecture. Computational intelligence can dramatically cut down systems response times by eliminating decision pauses in the kill chain. While it is difficult to place a numerical value on the amount of time that an operator reports up the chain of command, it would not be a far stretch to estimate that five to ten seconds may pass from when a positive identification has been made by the sensor operator who tells the supervisor of the confirmation which then verbally goes to the Tactical Action Officer. These technologies are still in their infancy and their adaptive learning and problem-solving applications are not mature enough to defend human life at this time. Further research, testing, and verification of computational intelligence will eventually lead to mostly non-organic self-defense architectures.

4. Information Assurance

Network-based systems are subject to exploitation, theft, viruses, worms, and other network interruptions that can alter data fidelity. This is especially true when other countries attempt to access our classified and tactical information. The current approach to network security is one where the data is protected through a layered defense, an intrusion prevention posture instead of intrusion detection. Network intrusion can be detected along the "outer walls" or perimeter by building various levels of security throughout the data flow in the architecture, and can be defended against prior to any data compromise.

Data integrity must start at the lowest level, coded binary data. The US Government uses the Triple Data Encryption Algorithm (TDEA) to perform this task. Although TDEA is intended for unclassified but sensitive information handling, it can provide a starting point for data protection. This algorithm can be implemented in software, firmware, hardware, or any combination thereof (Barker, 2004). Processing, transmission, and storage components of the architecture will possess the algorithm.

The network will require physical encryption of the data as it leaves classified spaces to provide system security. This can be accomplished with a high speed, CAT 6-supporTable, wideband encryption device. Users throughout the architecture will reside on a distribution list for the key(s); the key will be changed at a predetermined time interval. Some of the source material entering into the architecture may need to be kept separate from other data due to different classification levels or other access restrictions. Higher classified data will require cleansing and downgrading prior to introduction at a lower-security level when combining tactical and ISR data to present a COP for the

Composite Warfare Commander. There is existing government-owned software and hardware programs that when implemented can maintain the required separation and classification downgrade capabilities. The architecture will also contain differing levels of trust assigned to both tactical access and to user accounts.

The network that the equipment resides on will be kept in spaces with limited access. Only those personnel with clearances at or above the classification of the network will be allowed unescorted access to a space containing network interfaces. User access will also be limited to those personnel in performance of their duties by the system administrator and will be required to log on with an issued, restricted common access card, their user name, and a password.

Amongst the data processing, analyzing, and storage nodes, bulk encryption of outgoing and incoming data will occur, providing the first layer of network security. The sensor assets that transmit data may not require bulk encryption since they are a single stream already encrypted. The architecture will employ best of breed IA applications and practices to ensure the availability and confidentiality of system data while providing authentication and verification of system users as in Figure 8.

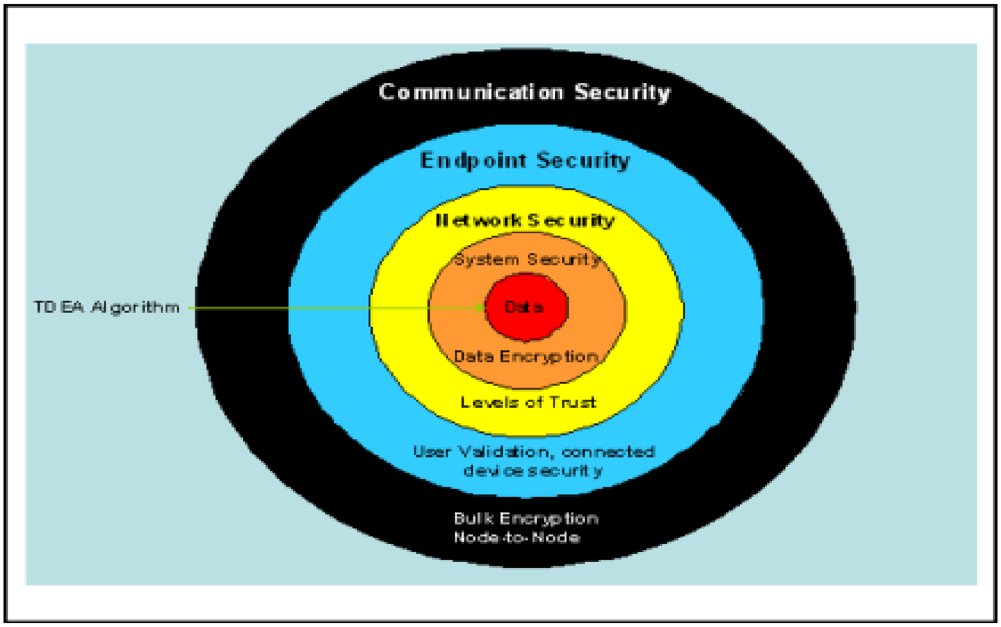


Figure 8. Network Information Assurance applications. Accreditation of the architecture is required in accordance with the Navy Information Assurance Program. (Modified from Defense Science Board, 2000)

5. Allocation of Tasking to People and/or Software

The available resources at the disposal of the system designer for task allocation are hardware, software, people, or combinations of the three. The allocation of some of the functions will be mandatory and predetermined by the stakeholders identified through the requirements analysis process. Task allocation should be determined through the comparison of performance between humans, hardware, and software; what the cost incurred will be; cognitive support of the operators; and knowledge of what pieces of information and decisions must be available to support the function. Knowledge of what resource would be best at executing what functionality at the cheapest cost can be crucial in the selection process to deliver a system with the optimum mix of functionality and resources at a reasonable cost.

Human role strategies will require that certain functions and tasks be performed by people within the system while others, due to performance requirements or stakeholder needs, will be allocated to hardware and software, or both. Given the mandatory allocations, a determination will need to be made of the knowledge, skills, and abilities (KSA) that will be required of the people that will be a part of the system.

After the mandatory functions have been allocated, the design team identifies potential allocations for those functions not yet allocated. These allocations can be static or dynamic in nature. The dynamic allocations will change depending on the mission conditions and/or priorities. During the primary mission phase, mission-critical functions will take priority and will be followed by other functions as well as other primary mission functions. To allocate these tasks effectively among the hardware, software, and people, a study of the operator and maintainer capabilities and limitations, as well as the potential of the hardware and software to perform the systems functions, will need to be identified. Other factors, which influence the allocation of functions and need to be taken into account, include safety, frequency of function occurrence, training requirements, and workload and manning requirements.

Within this phase of development, selection of a set of optimal function allocations based on the system design factors can be made. This effort will include comparing the proposed allocations to accepTable risk of the design, the time required to implement the design, expected performance and system availability, system manning levels, system lifecycle costs, and training requirements. Tradeoff studies will need to be performed, comparing system design factors and stakeholders desires to the proposed allocations to determine the correct mix of allocations as compared to the system requirements.

Possible human-in-the-loop (HIL) optimal allocations within an IFC context of operations are issue firing commands, issue abort commands frequency selection, reset faults, set radar doctrine parameters, selection of automatic modes, monitoring of system status, and monitor engagement resources.

During the problem definition phase of the design process, objectives and measures of effectiveness of the system will be developed by the system design team, and reviewed and approved by the stakeholders. Through the verification, validation, and acceptance phases of system development, the functional allocations will be matched against these requirements as well as the design requirements and specifications to ensure that the system has been designed and built correctly. Verification, starting in the design phase and overlapping the validation period, will determine that the configuration items meet the requirements developed by the stakeholders. The validation phase will determine whether the system capabilities match the operational concept. Acceptance phase is conducted by the stakeholders and will determine if the system satisfies their needs.

B. COMPARISON BETWEEN CURRENT AND PROPOSED ARCHITECTURES

The OA functional domain model, depicted in Figure 9 below, identifies the combat system detect-to-engage (DTE) functionality that is needed by the warfighter to establish a CMD strategy. Some of the OA design principles include the usage of common software that is reusable in part or whole and that can be implemented across many different platforms.

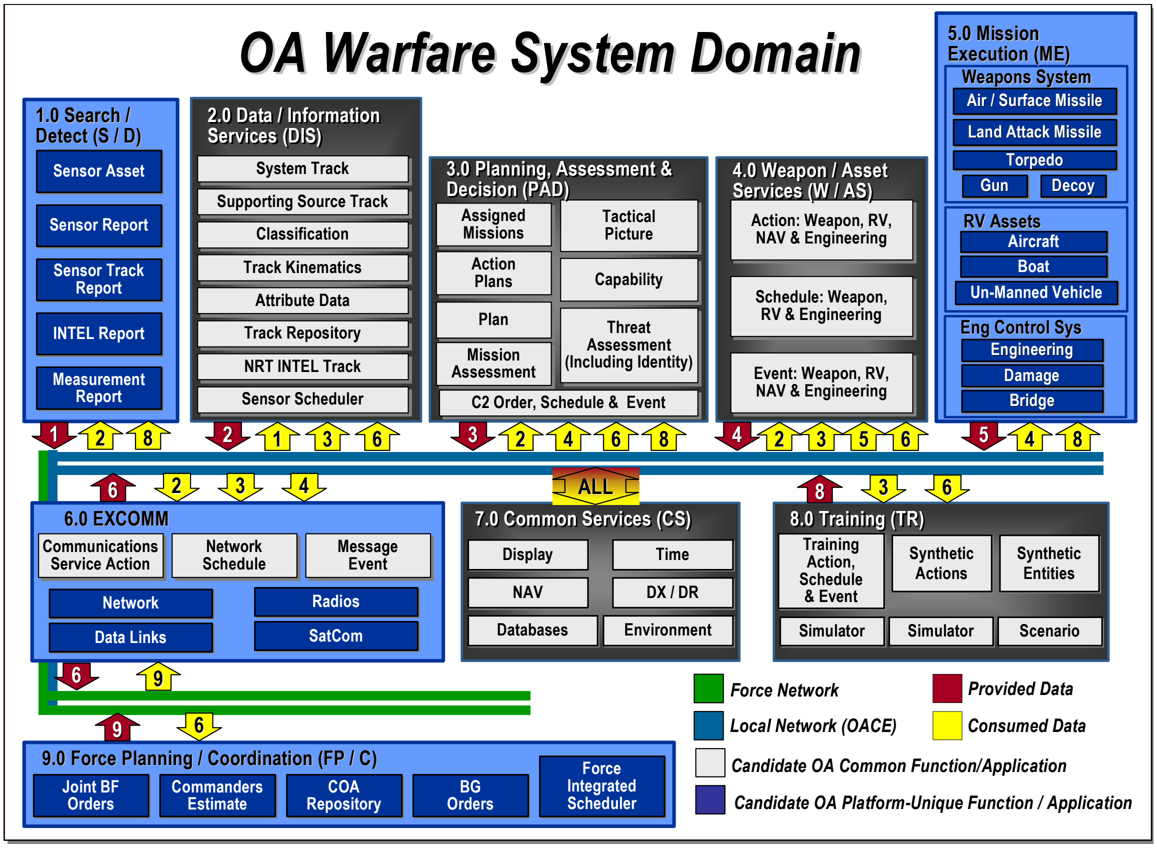


Figure 9. PEO IWS functional architecture (Strei, 2004). This architecture is expected to simplify FORCEnet implementation.

The OA Enterprise approach directly supports the implementation of FORCEnet design concepts and more robust business practices that improve cycle time with respect to technology refresh, simplifies software maintenance and delivery, rapidly enables new technology insertion, and capitalizes on a broader supplier base. These improvements translate into cost savings throughout the lifecycle of a combat or weapon system. The PEO IWS OA functional architecture was evaluated per generally agreed OA criteria. Figure 10 depicts the proposed high-level OA functional architecture.

The fundamental difference between the architectures in Figure 9 and Figure 10 is that the proposed architecture is horizontally integrated, which both greatly simplifies and minimizes the functional interfaces. Horizontal integration refers to the desired end-state where intelligence of all kinds flows rapidly and seamlessly to the warfighter, and enables information dominance warfare (JASON Program Office, 2007). In contrast, the PEO IWS functional idiom is functionally independent and characterized by large, complex, and highly coupled interfaces (Meilir Page-Jones, 1998).

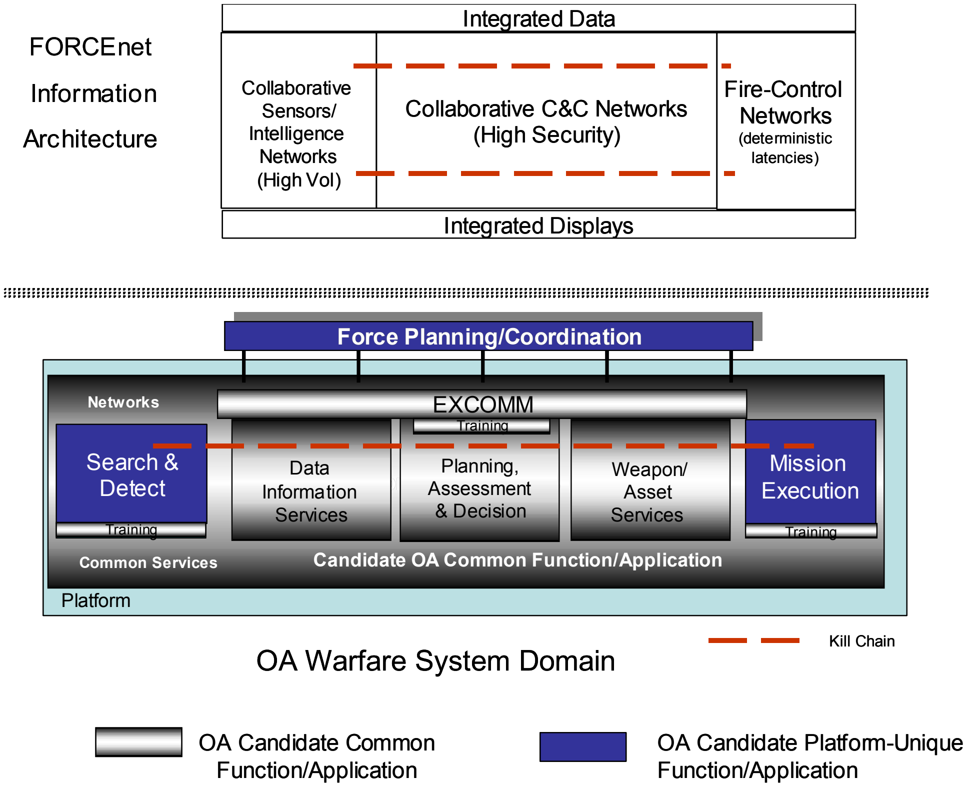

FORCEnet
Information
Architecture
Integrated Data
Collaborative Sensors/ Intelligence Networks (High Vol)
Collaborative C&C Networks (High Security)
Fire-Control Networks (deterministic latencies)
Integrated Displays
Force Planning/Coordination
Networks
EXCOMM
Training
Search & Detect
Training
Data Information Services
Planning, Assessment & Decision
Weapon/ Asset Services
Mission Execution
Training
Common Services
Candidate OA Common Function/Application
Platform
OA Warfare System Domain
Kill Chain
OA Candidate Common Function/Application
OA Candidate Platform-Unique Function/Application

Figure 10. Proposed high-level OA functional architecture. This architecture simplifies the kill chain process by horizontally integrating Search & Detect; Data Information Services; Planning, Assessment, and Decision; Weapon/Asset Services; and Mission Execution.

In computer science terms, this architecture exhibits high coupling and low cohesion and may demonstrate “brittle” behavior when subjected to stressor message transfer rates. While adopting identical functions, the proposed architecture captures an improved balance between cohesion and coupling of functions. While the PEO IWS functional architecture may deliver improved service in certain functionalities, it may perform worse in others. The net structural effect is that when the architecture is stressed during periods of high message flux rates, it may deteriorate or fail completely. The effect is compounded under tactical scenarios where several sources of uncertainty are prevalent. Another critical difference between the architectures is that the external communications (EXCOMM) function needs to become an OA candidate common function/application. Full integration between the EXCOMM and Command and Control systems is critical to the implementation of an IFC.

To further understand the differences, the proposed OA functional architecture is decomposed and compared to the PEO IWS architecture. The first observation is that the proposed architecture is broadly characterized through the parent-child relationship between FORCEnet and open architecture, separated by the Search and Detect, C2, and Engagement functions. That is, the open architecture supports FORCEnet and performs the kill chain functions through C2. Command and Control adheres to the rules of engagement; establishes positive target identification; performs engageability calculations; and preferred weapon selection based on multiple parameters, such as target kinematics, number of threats, and environment prior to issuing a weapons engagement order. These events must be completed quickly, accurately, and all may include several elements of uncertainty such as kinematics and inter-arrival uncertainty. The architecture must be both sTable to efficiently process (service) the arriving messages and be robust during high-stressor states.

The proposed architecture detours from the PEO IWS OA functional architecture as follows. The 7.0 Networks and Common Services, which includes displays, navigation, time, databases, data extraction, and recording functions, broadly aggregates the 1.0 Search & Detect (S&D); 2.0 Data Information Services (DIS); 3.0 Planning, Assessment & Decision (PAD); 4.0 Weapon/Asset Services (W/AS); and 5.0 Mission Execution (ME) under the 6.0 platform External Communications (EXCOMM) function. The EXCOMM function becomes a candidate OA common application. Instead of the PEO IWS independent 8.0 Training function, the proposed architecture integrates training functions (functions 1.0 through 5.0) that are explicitly used to execute simulations of the DTE process for the various combat/weapon systems supporting individual warfare and mission areas. In addition, each individual block within the nine modules will be examined throughout this section to determine if additional functions are needed to prosecute CMD successfully.

The proposed OA architecture retains the following advantages over the PEO IWS OA architecture:

- Horizontal functional integration simplifies and reduces the number of interfaces to balance cohesion with coupling to deliver robust (common) services during high message flux that are characteristic during cruise missile or other tactical engagements.
- The proposed OA architecture improves system stability; the ability to service increased cruise missile threats is greater than their inter-arrival rates.
- By design, the proposed architecture improves the ability to accurately and efficiently process (kinematics and inter-arrival) uncertainty.
- Ability to re-assess and re-engage target after first salvo is fired.
- EXCOMM changes from a candidate OA platform-unique function and application to a candidate OA common function and application.

C. BATTLESPACE DEFINITION

1. Definitions

The battlespace definition was adopted directly from United States Air Force (USAF) doctrine and the first step of the intelligence preparation of the battlespace process. The battlespace is defined as "the commander's conceptual view of the area and factors, which he must understand to apply combat power, protect the force, and complete the mission. It encompasses all applicable aspects of air, sea, space, land, and information operations, as well as the human dimension that the commander must consider in planning and executing military operations. The battlespace dimensions can change over time as the mission expands or contracts, according to operational objectives and force composition. Battlespace provides the commander a mental framework for analyzing and selecting courses of action for employing military forces in relationship to time, tempo, and depth" (Air Force Doctrine Document, 1997; Department of the Army, 1994).

The objective is to expand the battlespace volume. In the context of this paper, it is defined as increasing over-the-horizon surveillance and wide area defense against ASCM's. As defined above, the battlespace is not fixed; it varies in volume as a function of time and depends on wide area and long range combat identification (CID) of CMD threats, degree of interoperability, sensor range, Single Integrated Air Picture (SIAP) accuracy, synchronization, IFC, ABMA, and passive defense (Kaler, Riche, Hassell, 1999-2000). Long range CID of airborne threats increases the battlespace by increasing the composite warfare commander's confidence interval of achieving hard or soft mission kills of stressing CMD threats.

Similarly, improved interoperability among task force elements will increase the battlespace through accurate translation of kill chain events among the task force elements. For example, any lost track will effectively constrain the battlespace volume and reduce the probability of successfully defending task force elements in the event of ASCM attacks. Theater Air Missile Defense (TAMD) 2010 introduced six tenets to defeat aerial stressors by expanding the battlespace, only several of which will be applied to defining the CMD battlespace.

Increased sensor range by itself will not expand the battlespace. Instead, increasing the sensor range in concert with increased interoperability and intelligent signal processing algorithms expands the battlespace volume by increasing the Single Shot Probability of Kill (SSP_k) over single and multiple engagements. The SIAP continuously tracks each target and provides a common operating picture (COP) of overlapping engagement zones that increases the probability of defeating stressing CMD threats better than a singular task force element. The SIAP supports force synchronization, which means that weapons and sensors receive common parametric information from each task force element including weapons inventory and target track data. This information is used in IFC scenarios to determine the preferred sensor, weapon, and shooter. Thus, the SIAP enables the Composite Warfare Commander to capitalize on layered defenses.

IFC, composed of six scenarios, relies on platform-independent sensor fusion and weapons employment to overcome radar horizon or earth curvature effects that effectively constrain the battlespace volume. IFC's ability to increase the battlespace volume through layered defenses was successfully demonstrated during the 1996 Mountain Top Advanced Concept Technology Demonstrator (ACTD). The Mountain Top experiment validated the mid 1970's Forward Pass (FP) IFC concept of increasing the battlespace by extending the engagement range beyond the ship horizon (Krill, 1997). During the 1990's, FP was a type of Cooperative Engagement Capability (CEC) whereas today it is one of six independent IFC scenarios. Also significant is that FP-aggregated Engage on Remote (EOR) entails currently stand-alone IFC scenarios. IFC is defined as the ability of a weapon system to develop fire control solutions from information provided by one or more non-organic sensor sources, conduct engagements based on these fire control solutions, and either provide mid-course guidance to the interceptors based on this externally provided information or in certain cases, have them provided by a warfare unit other than the launching unit. IFC can be executed through several architectures that include human-in-the-loop, semi-automated IFC, or fully automated IFC. Only fully automated IFC is considered with human override capability in the context of this paper and for the purposes of defining the battlespace and designing a FORCEnet architecture capable of defending against CMD threats.

In simple terms, ABMA increases engagement efficiency by optimizing the sensors, weapons, and identification of shooters from multiple geographically-separated task elements. Optimization in this sense refers to the ability of the Composite Warfare Commander to assign quickly and accurately weapons to stressing threats in a dynamic tactical scenario. Quality signal processing algorithms are ABMA inputs that enable expansion of the Composite Warfare Commander's battlespace volume.

Passive defense effectively expands the battlespace through early warning prediction of point of impact and time of intercept based on IFC scenarios that optimize the sensors, weapons, and shooter's ability to maximize P_k.

The model represented by Figure 11 includes selected portions of the TAMD 2010 six tenets that were used in the context of this paper to define the battlespace. In particular, ABMA and CID functionality are responsible for improving the interoperability within the IFC scenarios.

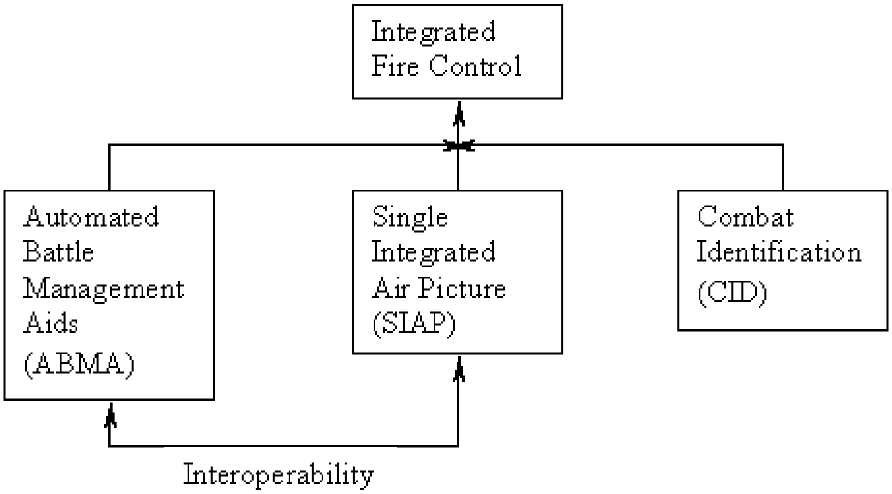

Integrated Fire Control
Automated Battle Management Aids (ABMA)
Single Integrated Air Picture (SIAP)
Combat Identification (CID)
Interoperability

Figure 11. 2010 CMD concept optimizing force employment (Barwis, 2006). The model illustrates the relationship among the SIAP, ABMA, and wide area long range CID in support of the IFC scenarios.

"Defeating modern cruise missiles involves three distinct phases: detection, control, and engagement (GAO, 2000)." While the GAO battlespace model is correct in the above statement, it is too wide in scope and insufficiently granular to support the six kill chain functions. The GAO model aggregates the sensor-to-shooter kill chain functions into the GAO detect function. The GAO model parallels the kill chain track and control functionality but completely ignores the assessment function. The GAO battlespace model functionality is compared to the chain model in Figure 12 below.

IFC is fundamental to improved cruise missile defense. The United States and its allies spend large sums of money over protracted development cycles, frequently measured in decades, to field weapons delivery platforms. In contrast, foreign suppliers are agile in their ability to field low-cost ASCM's and export the tactics needed to deploy them successfully against coalition shipping.

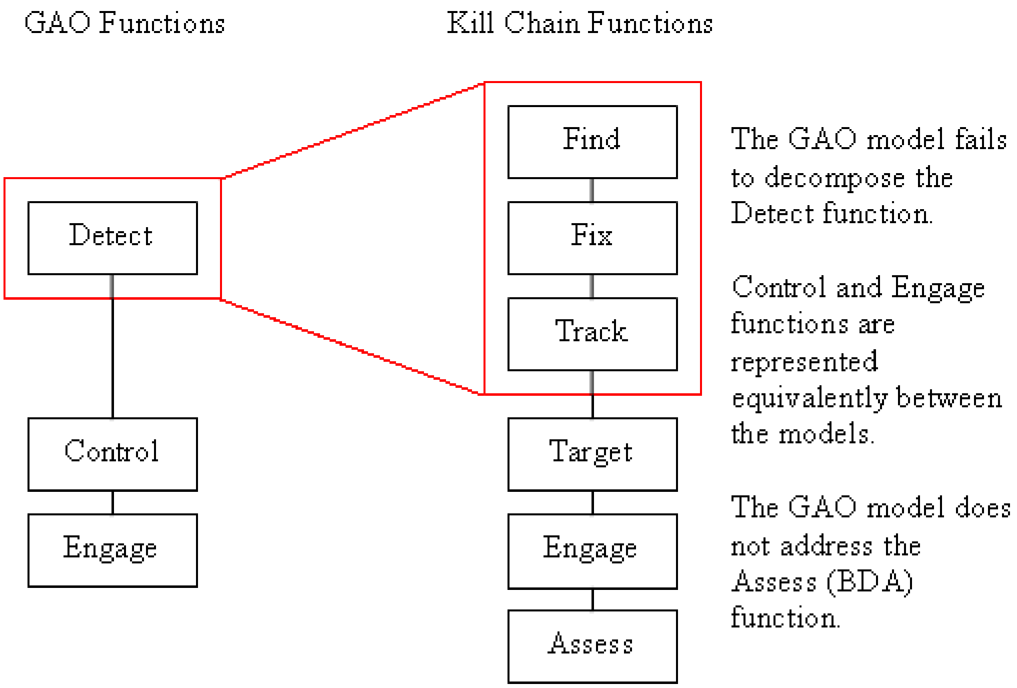

GAO Functions
Kill Chain Functions
Detect
Control
Engage
Find
Fix
Track
Target
Engage
Assess
The GAO model fails to decompose the Detect function.
Control and Engage functions are represented equivalently between the models.
The GAO model does not address the Assess (BDA) function.

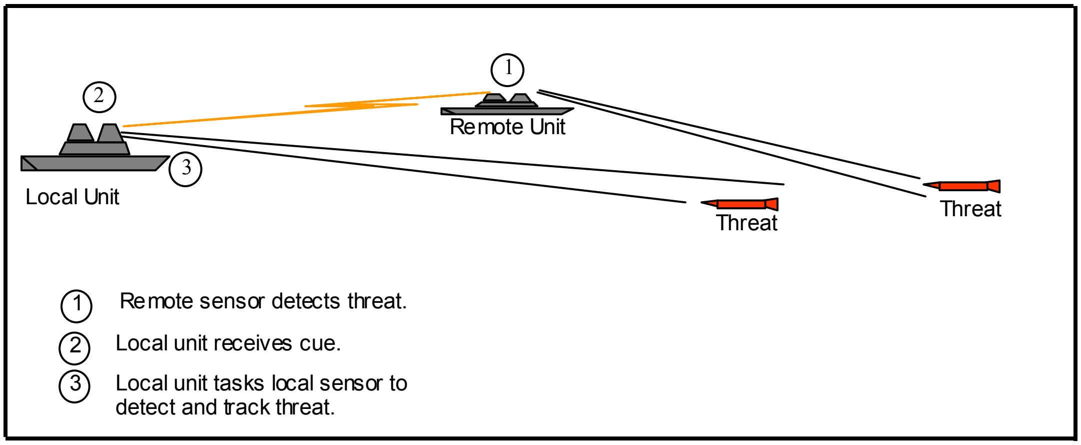
Figure 12. GAO battlespace functionality model versus kill chain model. The GAO model parallels the kill chain track and control functionality.

These ASCM's enable nations of economically modest means, but hostile to the United States, to exercise power in response to perceived coalition threats, further political or regional power agendas, or to promote theater-specific mayhem. IFC expands the battlespace by enabling airborne surveillance platforms, or Joint Land Attack Cruise Missile Defense Elevated Netted Sensor (JLENS), to relay ASCM tracks through FORCEnet to task force elements (Bolkcom, Hichkad, 2005).

2. Battlespace Scenarios

This report focuses on CMD scenarios using IFC capabilities with emphasis on Precision Cue, Launch on Remote, and Preferred Shooter Determination. The CMD scenarios are described below. Figures 13 through 18 are modified from *Future Integrated Fire Control* (Young, 2005).

Precision Cue, shown in Figure 13, is an IFC capability in which a cue is received from a remote source that represents a possible threat and is used to direct local sensors to hold a specific target. The cue is comprised of target information such as a location estimate, target track data, and assessment of the target's identification. The remote sensors can be located on an airborne or surface platform and the local sensors are located on a surface platform. The cue from a remote sensor on an airborne platform is more advantageous than the cue from a surface platform due to radar geometry. The airborne platform extends the range of the surface radars and provides earlier warning to the surface platform before the incoming cruise missile enters the detection range of the local sensors. Early detection from the remote sensor cue will increase the CMD reaction time, allowing for early engagement with a higher probability of kill. The best shooter to engage the threat at this point can be selected using the Preferred Shooter Determination IFC capability.



1
2
Remote Unit
3
Local Unit
Threat
Threat
1 Remote sensor detects threat.
2 Local unit receives cue.
3 Local unit tasks local sensor to detect and track threat.

Figure 13. Depiction of the Integrated Fire Control Precision Cue scenario. A cue is received from a remote source that represents a possible threat and is used to direct local sensors to hold a specific target.

Preferred Shooter Determination, as shown in Figure 14, is an IFC in which the optimum weapons from a group of warfare units is selected to intercept the threat target. The best shooter is selected based on best available engagement geometry and engageability determination. This IFC capability requires extensive collaboration among units. Ship location is another factor that will influence the choice for best shooter. The best shooter can be an airborne or surface platform, or a combination of the two. With Fire Control Quality (FCQ) threat data from a remote sensor, the remote unit can initiate a launch from a local firing unit of the best shooter using the Launch on Remote IFC capability.

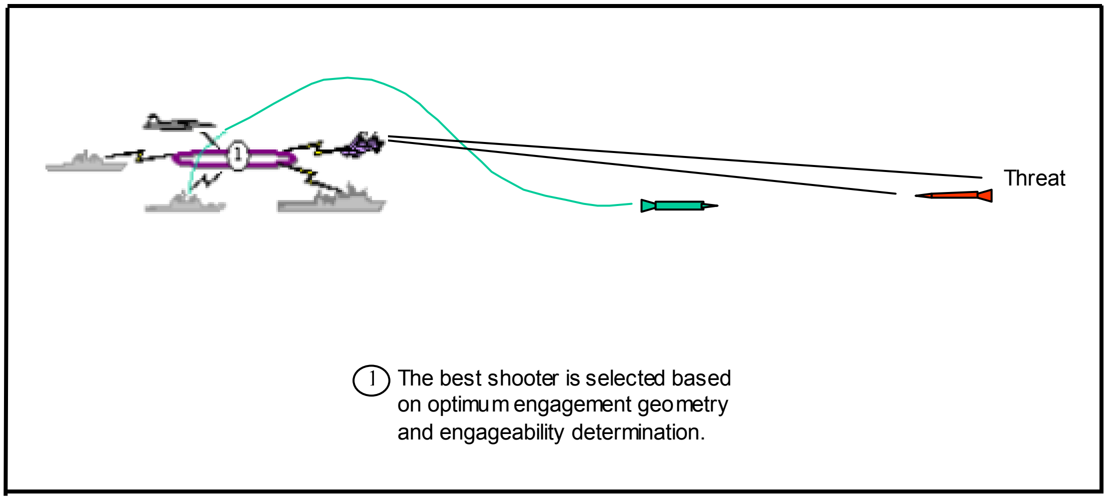

1
Threat
1 The best shooter is selected based on optimum engagement geometry and engageability determination.

Figure 14. The Integrated Fire Control Preferred Shooter Determination scenario. The optimum weapon from a group of warfare units is selected to intercept the threat target.

Launch on Remote, Figure 15, is an IFC capability in which the remote sensor data is used to initiate a missile launch without holding the track locally. The local firing unit uses remote sensor data from a remote airborne or surface platform to track and engage cruise missiles launched from air, land, or sea-based platforms. Since remote sensors on an airborne platform are not limited by line of sight, local firing units have more time to react before the local sensor detects inbound cruise missiles. With early and accurate remote sensor data, the local firing unit on a surface platform can start launching missile as soon as the cruise missile enters the radar detection zone. This early engagement keeps the intercept point as further away as possible from the ship and provides additional re-engagement opportunities.

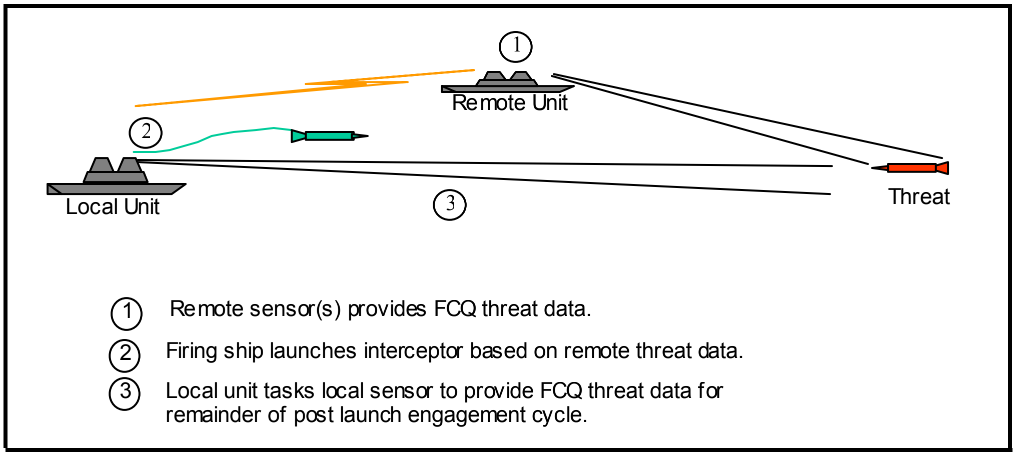

1
Remote Unit
2
Local Unit
3
Threat
1 Remote sensor(s) provides FCQ threat data.
2 Firing ship launches interceptor based on remote threat data.
3 Local unit tasks local sensor to provide FCQ threat data for remainder of post launch engagement cycle.

Figure 15. Depiction of the Integrated Fire Control Launch on Remote scenario. Remote sensor data is used to initiate a missile launch without holding the track locally.

Other IFC capabilities used to engage cruise missiles threats are Engage on Remote, Forward Pass, and Remote Fire. Engage on Remote, Figure 16, is an IFC capability where one or more remote sensor units provide data to conduct an engagement. Engage on Remote uses remote data to initiate a missile launch from a firing unit, and remote sensors to illuminate the threat by relaying guidance to the interceptor. Engage on Remote is Launch on Remote with in-flight support from the remote unit.

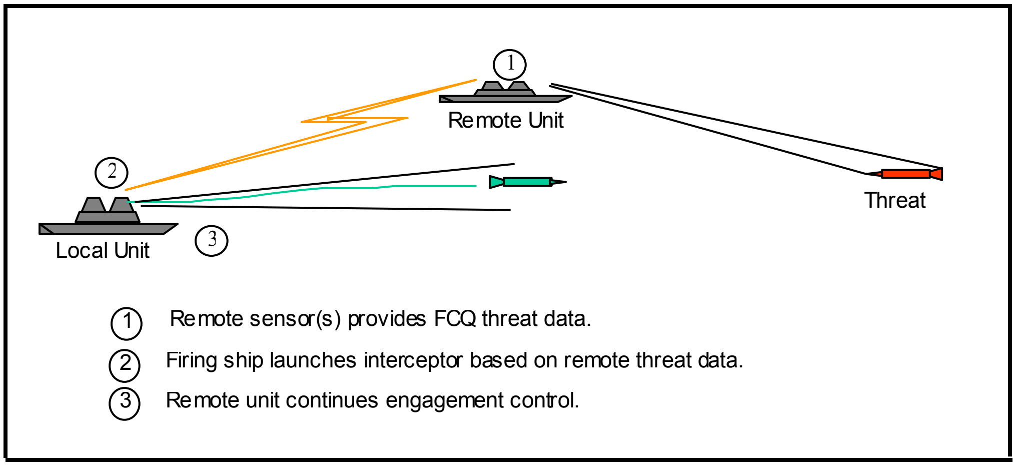
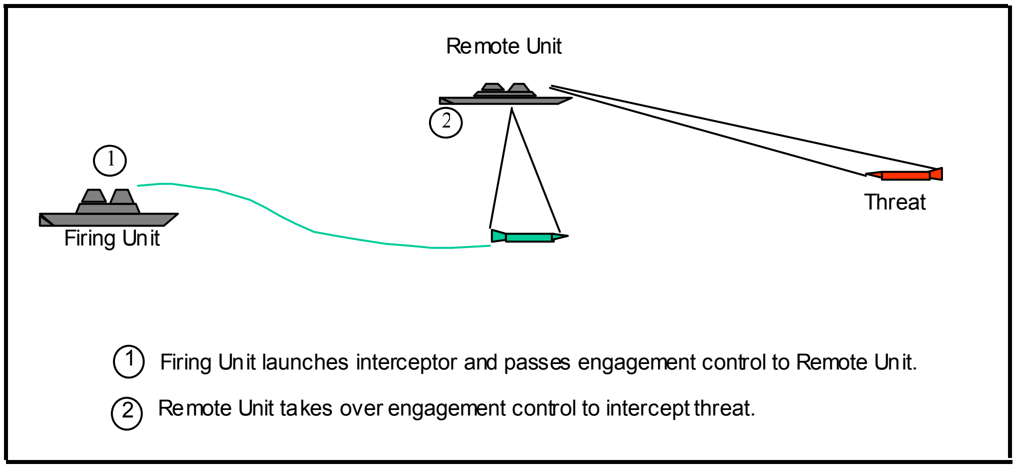

1
Remote Unit
2
Local Unit
3
Threat
1 Remote sensor(s) provides FCQ threat data.
2 Firing ship launches interceptor based on remote threat data.
3 Remote unit continues engagement control.

Figure 16. Depiction of the Integrated Fire Control Engage on Remote scenario. One or more remote sensor units provide data to conduct an engagement.

Forward Pass, Figure 17, occurs when a remote unit takes over the in-flight missile control from the firing unit to complete the engagement. This IFC is effective in battlegroup engagements to defend against a single or multiple cruise missile threats.

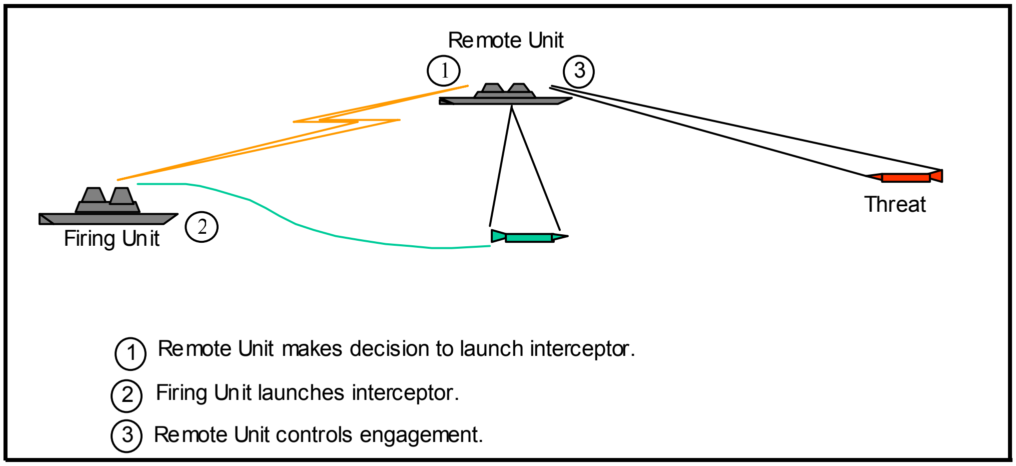


1 Firing Unit launches interceptor and passes engagement control to Remote Unit.

2 Remote Unit takes over engagement control to intercept threat.

Figure 17. Depiction of the Integrated Fire Control Forward Pass scenario. A remote unit takes over the in-flight missile control from the firing unit to complete the engagement.

In the Remote Fire scenario, Figure 18, the launch decision is made by the remote unit and the engagement control can be handled by the remote unit or the firing unit. This IFC provides flexible engagement control between remote and firing units for the most effective engagement.



1 Remote Unit makes decision to launch interceptor.

2 Firing Unit launches interceptor.

3 Remote Unit controls engagement.

Figure 18. Depiction of the Integrated Fire Control Remote Fire scenario. The launch decision is made by the remote unit and the engagement control can be handled by the remote unit or the firing unit.

To better visualize the intended architecture and its capabilities, Navy strike group configurations and CONOPS were developed by the authors to provide a blueprint for wargaming. The following scenarios were created to provide a realistic approach for edification of our architecture and validation of our model. The current layout shows a CSG with its units aligned in defense along the estimated threat axis. Although an ESG will have different aircraft assets, the model for all generally aligns the same and can be used by renaming the main body and minor modification of sensors/weapons input to the model. See Figure 19 below.

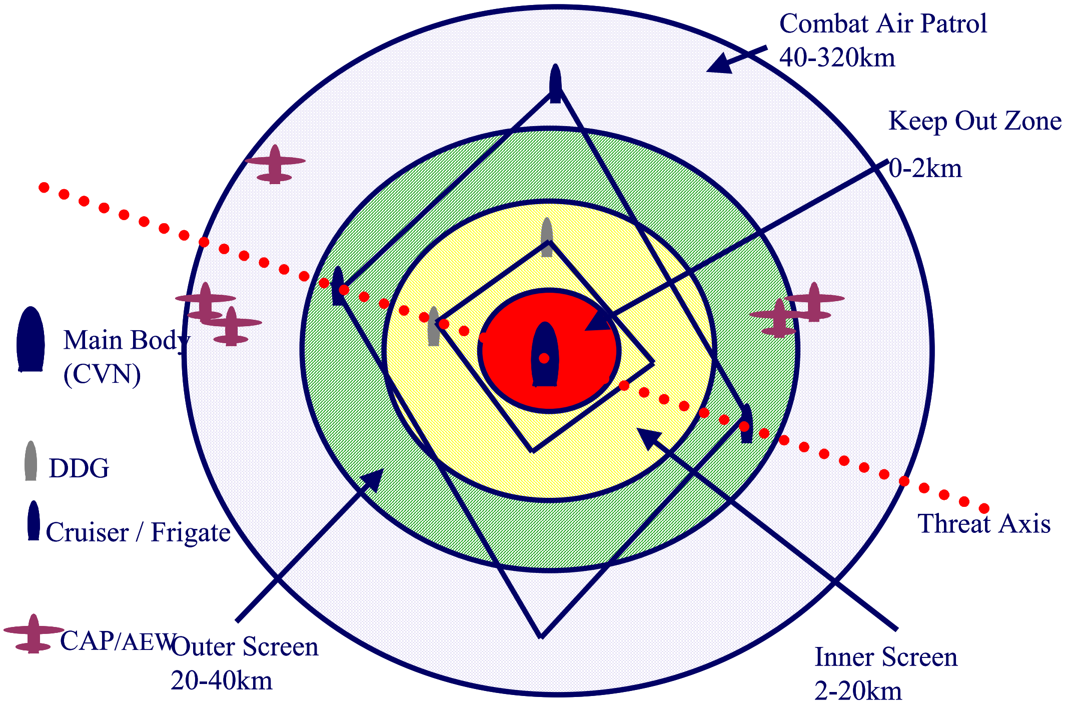


Figure 19. Overall physical layout of battleforces.
This layout is used to visualize scenarios for validating the proposed ASCMD functional architecture.

For the first tactical scenario, Carrier Strike Group CONSTELLATION is underway in the Arabian Gulf. Tensions in the area of operations (AOR) are elevated due to political unrest in the fictional nation of Drmecia. A pro-democracy faction has held demonstrations demanding less government involvement and more individual rights. The United States supports this faction and has sent CONSTELLATION CSG into international waters off of the Drmecia coast. The Drmecia president has vowed to suppress the insurrection in his country and has warned the international community not to interfere with domestic affairs. He has previously stated that he has purchased Sunburn missiles for defense of his country and will not hesitate to launch them if provoked.

An SH-60 helicopter flying off CONSTELLATION has experienced a catastrophic loss of hydraulic fluid and has crash-landed onto an island two kilometers off of the Drmecia coast, land that is claimed by Drmecia. Attempts to rescue the stranded crewmembers are interpreted by the Drmecia Defense Ministry as an attempt to infiltrate American Special Forces. Drmecia defenses go on high alert. Drmecia's President states that he is not afraid of the United States and vows that his will be the first country to sink an American Aircraft Carrier since WWII if the Americans continue their aggressive actions.

CSG Carrier Air Patrols are extended, but the main body pulls farther from the coast. Drmecia naval vessels actively shadow CSG units. During a Maritime Interdiction Operation (MIO) against a Drmecia fishing vessel, a Drmecia ship opens fire on USS CHOSIN, which returns fire destroying the Drmecia vessel. The response from Drmecia is ten inbound missiles coming from Drmecia, each one fired at approximately three-second intervals from shore-based battery. Each missile flies at sum Mach 2.0 with initial launch pop up to altitude of 90 feet, then drop to 15-20 feet above sea level within 15 miles of target.

In the second scenario Country "Orange" leadership has coveted the island nation "Green" for its newfound oil reserves and natural deep industrial harbors. Country Green is located approximately 180 kilometers off country Orange's coastline. Both countries share a similar ancestry but differ in political views. Country Green has provided its citizens with generous royalties derived from a strong economy, angering many of the Orange politicians who believe that Green should share the wealth with their impoverished nation.

Country Green is an ally of country Blue whose strong Navy acts as a deterrent to hostile actions from Orange. Country Orange has recently protested joint naval exercises conducted by Green and Blue CSG's close to their shoreline as acts of provocation and vow to defend their country against all threats.

Country Orange has mined three of the five harbors that Green homeports its surface fleet. A Green destroyer encounters a mine while entering port and the subsequent explosion causes 15 deaths and cripples the ship. Orange has initiated a naval blockade on the western coast of country Green and warns international traffic that it will sink any vessel that enters the vicinity without Green’s escort. Blue CSG returns to the conflict area and takes up station off the Orange coast. Orange then fires 10 missiles, each firing simultaneously at one-second intervals, from aircraft at altitude of 10,000 meters from ten separate bearings. Missile flies at an average of just under Mach 2, dropping to an altitude of 5-7 meters above sea level once it is 32 kilometers from the target.

Queuing theory provided the foundation to model the battlespace in defense against stressor ASCM threats. The model is based on a quadruple serial queue; one arrival and three weapons assignment queues for each layered defense weapon. This model is represented by the event graph model in Figure 20 and Figure 21 below. The engagements represented in Figure 19 were modeled as discrete-events while the software simulation was based on a process view. Still, it is important for the reader to understand that the discrete-event model drove the process view-based simulation.

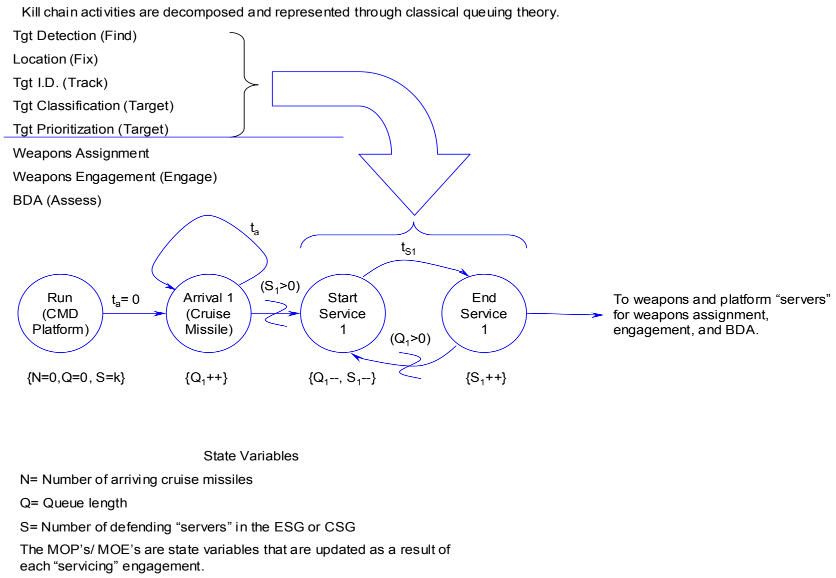

Kill chain activities are decomposed and represented through classical queuing theory.
Tgt Detection (Find)
Location (Fix)
Tgt I.D. (Track)
Tgt Classification (Target)
Tgt Prioritization (Target)
Weapons Assignment
Weapons Engagement (Engage)
BDA (Assess)
t_a
t_{S1}
Run (CMD Platform)
t_a= 0
Arrival 1 (Cruise Missile)
(S_1>0)
Start Service 1
End Service 1
(Q_1>0)
To weapons and platform “servers” for weapons assignment, engagement, and BDA.
{N=0,Q=0, S=k}
{Q_1++}
{Q_1--, S_1--}
{S_1++}
State Variables
N= Number of arriving cruise missiles
Q= Queue length
S= Number of defending “servers” in the ESG or CSG
The MOP’s/ MOE’s are state variables that are updated as a result of each “servicing” engagement.

Figure 20. Event graph representation of ASCM defense.

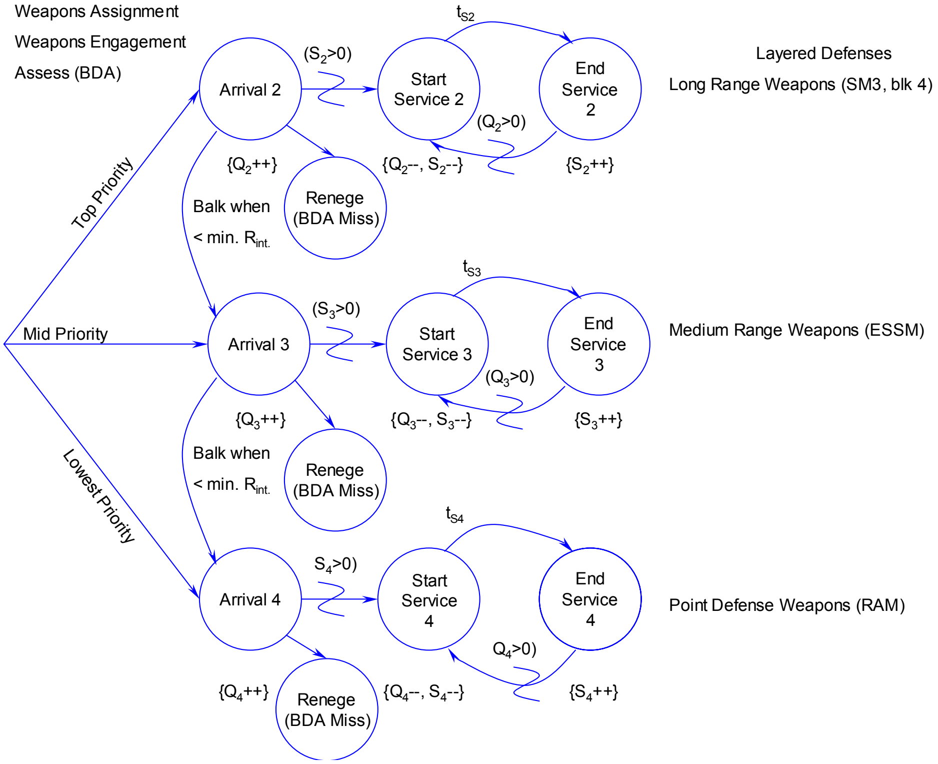


Figure 21. Event graph representation of ASCM defense.
The battlespace model is based on a quadruple serial queue.

ASCM's enter the first queue with an arrival rate, λ, and average time between arrivals or inter-arrival rate, λ^{-1}. The arriving ASCM initializes the queue and the kill chain begins service defensive functions. The queue becomes unsTable and defenses reach saturation, whenever the raid or stream arrival rate exceeds the service time. ASCM service time is complete upon ASCM intercept or declaration of a leaker.

Based on the kill chain functions the threat is first identified, classified, prioritized, weapon-target pairing is completed, and then enters the second queue. The ASCM is serviced or engaged by the shooter using long-range Standard Missile 3 (SM-3) interceptors whose average service rate is μ and average service time for stream or raid engagements is μ^{-1}. If the ASCM penetrated the outer layer of defense, then it is either a leaker or the queue balks because the stressor is within the minimum intercept range. In either event, the ASCM enters the third queue for reengagement by the second layer of defensive capability, the Evolved Sea Sparrow Missile (ESSM). Similarly, if the queue balked due to minimum intercept range limitations or the ASCM penetrated the second defensive layer with sufficient reengagement time remaining, then it enters the fourth and final queue for engagement by the Rolling Airframe Missile (RAM). Figure 21 does not feature a fifth queue for point defense guns such as the Close-In Weapon System (CIWS) because it was assumed that even if the ASCM was successfully engaged it will typically be within the keep-out range where fragments have a high probability of intercepting the shooter.

The following assumptions were used in the initial battlespace modeling:

- The initial queue state is empty and idle.
- First-in, first-out (FIFO) queue discipline is maintained to service stressor threats.
- Perfect IFC self-synchronization; no more than a single shooter engages a single stressor.
- From the shooter's perspective, a leaker that penetrated the innermost layer of defense is considered a miss even if it does not directly or indirectly impact the shooter.
- The IPB process does not give the task force commander reliable knowledge of ASCM inventory.
- Uncertainty in raid or stream arrival distributions and distribution parameters were estimated and discussed below.
- ASCM inter-arrival rates are statistically dependent; the arrival of one stressor threat directly influences the arrival of the next stressing threat. That is, ASCM raid or stream attacks are coordinated and based on the adversaries firing policy.
- The probability of detection, Pd, equals 1.0. The task force calculates with certainty the ASCM's position, velocity, time of intercept (TOI), and Point of Impact (POI).

As mentioned above, uncertainty of arriving ASCM's and the service times were estimated. Several statistical distributions were modeled including the Poisson, Beta, Uniform, and Triangular distribution. There were two sources of top-level uncertainty associated with the analysis; selection of a distribution and estimation of the distribution parameters.

The Poisson distribution is a discrete distribution suiTable for counting events such as counting the arrivals of ASCM's in a raid or streaming attack. At first glance, the Binomial distribution, another discrete distribution, was considered to model service time uncertainty because it is based on success or failure criteria. The success or failure logic was extrapolated to ask whether the stressor ASCM was killed or missed. However, applying the binomial distribution required testing several criteria. One criterion was that the arrival of stressor ASCM's, called trials in statistical terms, must be independent. This criterion was not met because it is contrary to a key assumption of dependence.

The Beta, Uniform, and Triangular continuous distributions were well-suited for modeling ASCM arrival and service times. While ASCM arrivals are a counting process, the Beta distribution is suiTable because the output of its α and β shape parameters define the expected value. The Uniform distribution is a suiTable distribution because any value is equally likely to occur. The Triangular distribution is suiTable because the probability of the random variable of interest is assumed within a given interval.

The following firing policies that had direct bearing on the probability of kill (P_k) and number of leakers, which translates to the probability of survival (P_s), were modeled:

- Shoot
- Shoot-Look-Shoot
- Shoot-Shoot-Look

The reader is advised that dependence in the context of arriving ASCM's must not be confused with the probability of kill (P_k) of shooting down the ASCM. The P_k for successive shots against stressor threats is statistically independent but is influenced by the selected firing policy. In other words, the firing policy influences P_k, P_k does not influence the selected firing policy.

While the application of statistics to queuing theory was not the thrust of this paper, they were fundamental to constructing a reasonable simulation model. Deterministic P_k calculations for the various firing policies are left as an exercise for the reader but were based on simple parallel networks where the P_k for successive shots against stressor threats must be independent.

3. Design Principles

The following design principles were identified in an attempt to define the high-level requirements for the development of system solutions in support of Cruise Missile Defense operations. Extensive research was conducted to identify the most relevant design principles that should be taken into account and are paramount to ensure the effectiveness, suitability, and survivability of our deployed forces. This is not a comprehensive list of all the design principles required, but serves as a departure point for further research and to improve upon.

- Provide robust, reliable, and timely communication to all platforms (nodes), based on mission requirements and inherent capabilities of those platforms/nodes (Clark, Hagee, 2006). Allow for interoperability with C2 and weapon systems of very different types and levels of sophistication. This level of interoperability needs to allow for implementation of requirements such as: engagement control strategy, distributed weapons coordination, battle management, distributed training, and in-flight control of non-organic weapons.
- Provide each decision-maker the ability to depict situational information in a tailorable, user-defined, shareable, primarily visual representation (Clark, Hagee, 2006). This requires reliable, accurate, and timely location, identity and status information on all friendly forces, environmental, neutral, and hostile elements, units, activities, events, sites, and entities/individuals.
- Store, process, analyze, evaluate, synthesize, catalogue, and retrieve all information produced by any platform/node on the network in a comprehensive, standard repository so that the information is readily accessible to all nodes and compatible with the forms required by any nodes, within security restrictions (FORCEnet, 2005). Implement push-pull technologies to allow efficient access, retrieval, sharing, and distribution of critical C2 and integrated fire control data that is accurate

and provided at the right time, and at the right location. As explained in *Enterprise C/S* (Hurwitz, 1997) push technology means that a user states under which conditions information should be sent. The user therefore subscribes to key pieces of data that are then "pushed" or delivered to the user. Pull technology refers to information that is stored on a server and accessed on demand by a user.

- Design IFC into a decentralized architecture (Young, 2005) that allows individual platforms to support individual phases of the Detect to Engage (DTE) process against cruise missile threats.
- Automate DTE functions to be conducted locally or remotely (Young, 2005) such as: ordnance selection, issuance of firing command, re-engagement, engagement initiation, salvo size, rate of fire, guidance control, weapon-target pairing, sensor support for engagement, intercept geometry, preferred shooter, and terminal homing support for interceptors.
- Provide information assurance at the platform and battleforce level (FORCEnet, 2005). Protect the confidentiality, integrity, and availability of data and their delivery systems, in addition to ensuring adequate authentication and non-repudiation. This includes providing for restoration of information systems by incorporating protection, detection, and reaction capabilities.
- Transfer search and detect, command and control, and fire control functions from individual systems methods to CSG/ESG/CAW common processes.
- Allow individual platforms to perform IFC while exercising local Command Authority (Young, 2005) and to function independently while temporarily disconnected from the battleforce network.
- Conduct persistent readiness level monitoring of all combat systems elements such as sensors, command and control, weapons, ordnance, logistics, support systems, networks, communications, computing infrastructure, and networks in order to maintain the highest possible level of material readiness and to identify, diagnose, troubleshoot problems,

make timely repairs, document, and share lessons learned with applicable users as needed. Ensure that readiness levels are maintained and shared between all platforms.

- Provide decision makers the ability to determine the best course of action under conditions of uncertainty, friction, time, pressure, and other stresses (Young, 2005 & FORCEnet, 2005).
- Provide the capability to determine or predict the intent or threat level of an inbound unknown object with high level of accuracy and reliability.
- Implement a battle force resource management capability that is distributed across all platforms/nodes to aid in the determination of best course of action.

Additionally, the following Open Architecture design principles are the foundation for the realization of the above design concepts:

- Develop a common and standards-based computing environment and implement distributed computer processing power to improve survivability.
- Functional capabilities are implemented as medium-grain components (OACE, 2003). A software component is a unit of independent deployment and third-party composition that has no persistent state. They often reside on the server and provide infrastructure for applications such as frameworks, binary programs, and templates (Aitken, 1999).
- Use of object-oriented (OO) programming within components and middleware technologies for interconnection of and interoperation among components (Bruegge, Dutoit, 2004; OACE, 2003).
- Use of design mechanisms such as client-server to maximize isolation of implementation details from publicly visible services and Application Programmer Interfaces (OACE, 2003).
- Build modular designs and disclose data to permit evolutionary designs, technology insertion, competitive innovation, and alternative competitive approaches from multiple qualified sources (OA Strategy).

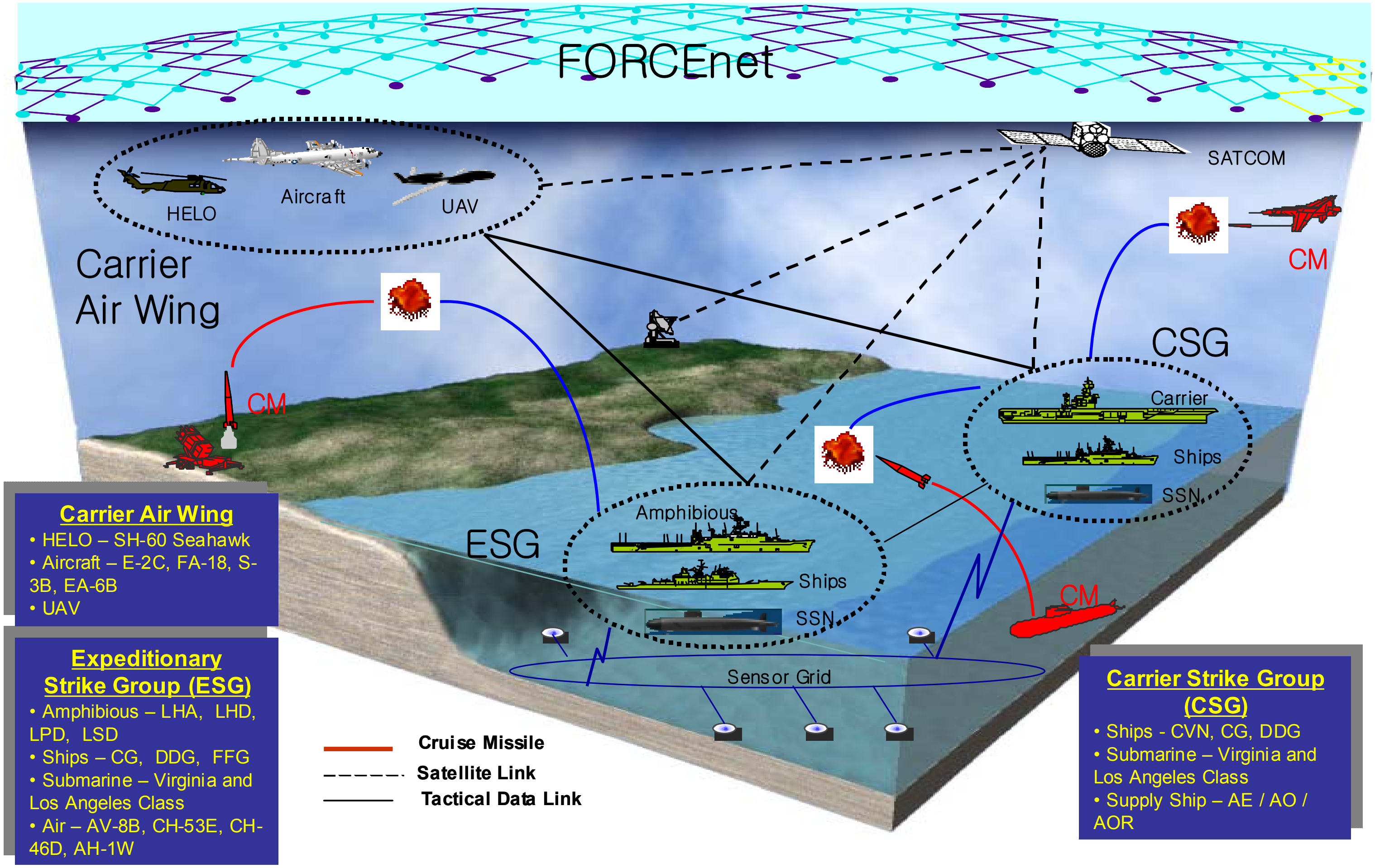
- Provide a common interface using the same standards to establish a plug-and-play connectivity throughout the combat system.

D. CONCEPTUAL DESIGN

1. Conceptual Framework – Operational View

Figure 22, developed by the authors, shows the operational concept of the ASCMD functional architecture in support of CMD for the overview scenario described in this paper. In essence, the Operational View (OV-1) represents a graphical executive summary that describes the missions, high-level operations, organizations, and geographical distribution of assets. Both local and remote OA units in the FORCEnet command and control network consist of Carrier Air Wing (CAW), Expeditionary Strike Group (ESG), and Carrier Strike Group (CSG). The FORCEnet command and control communicates with the integrated CAW, CSG, and/or ESG sensors to provide situational awareness and adequate response with a joint single integrated picture. All participant units are to monitor and assess the current tactical situation of cruise missile threats via FORCEnet's C2 component utilizing DoD Satellite Communications (SATCOM). Each element of the ASCMD system provides the capability of sharing all available resources and information to present successful Integrated Fire Control (IFC) for intercepting potential cruise missile threats from air, land, and undersea.

Note that the scenarios discussed in the Defining the Battlespace section do not reflect cruise missile launches from a submarine. The scope of this project is on above-water cruise missile engagement. However, our architecture has the capability to handle undersea cruise missile attacks.



Carrier Air Wing

- HELO – SH-60 Seahawk
- Aircraft – E-2C, FA-18, S-3B, EA-6B
- UAV

Expeditionary Strike Group (ESG)

- Amphibious – LHA, LHD, LPD, LSD
- Ships – CG, DDG, FFG
- Submarine – Virginia and Los Angeles Class
- Air – AV-8B, CH-53E, CH-46D, AH-1W

Carrier Strike Group (CSG)

- Ships - CVN, CG, DDG
- Submarine – Virginia and Los Angeles Class
- Supply Ship – AE / AO / AOR

Figure 22. Operational View (OV-1) for ASCMD in support of CMD.
The OV-1 provides a graphical executive summary that describe the missions, high-level operations, organizations, and geographical distribution of assets (http://www.vsix.net/other/special/United_States_IPv6_Summit_2005/United_States_IPv6_Summit.htm, 2007).

2. Architecture Diagrams

An Architecture Flow Diagram (AFD) based on *Process for System Architecture and Requirements Engineering* (Hatley, Hruschka, Pirbhai, 2000) was developed to capture the modules and flows that make up the proposed OA functional architecture described in this paper. The AFD, represented in Figure 23, is divided into five regions: user interface processing, main and support functions, and input and output processing. The External Communications (EXCOMM) is common to both the user interface processing and the main functions. The reason for the duality is that once the battleforce commander plans a mission, coordination and communication of mission orders to his battlegroup occur through EXCOMM. The battlegroup in turn uses the EXCOMM to retrieve and access those orders. Each module and flow within the AFD is described below.

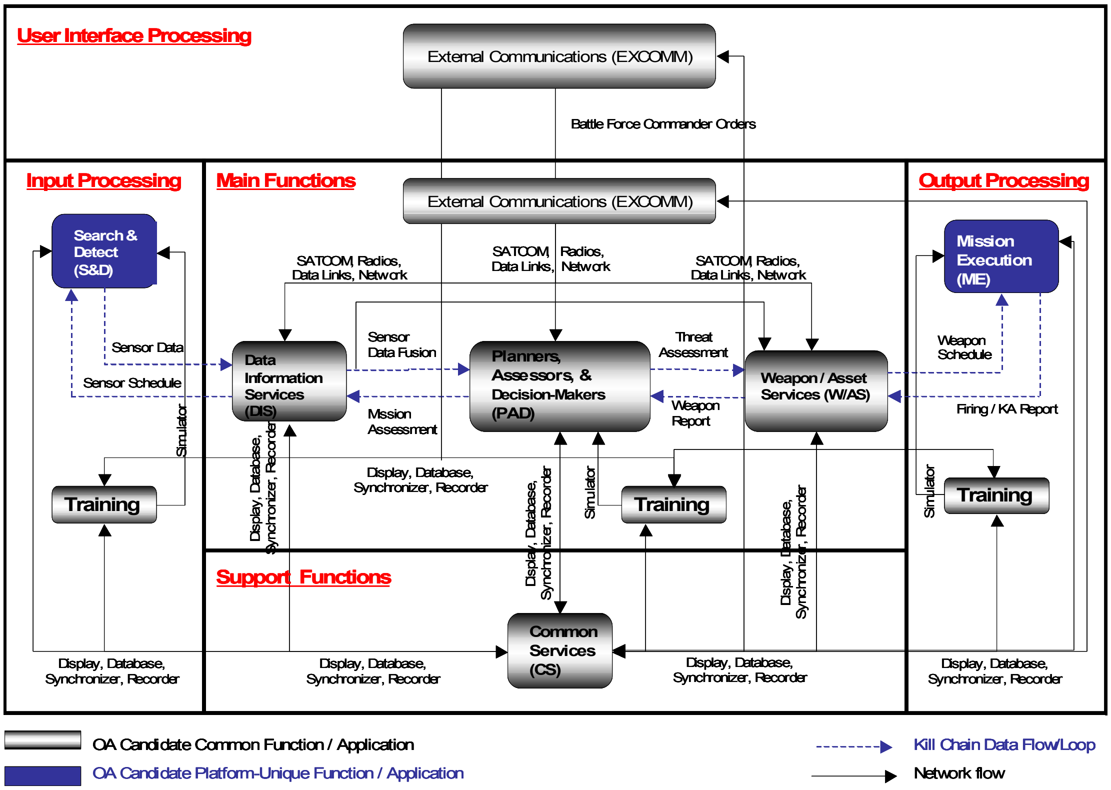

User Interface Processing
External Communications (EXCOMM)
Battle Force Commander Orders
Input Processing
Main Functions
Output Processing
External Communications (EXCOMM)
Search & Detect (S&D)
SATCOM, Radios, Data Links, Network
SATCOM, Radios, Data Links, Network
SATCOM, Radios, Data Links, Network
Mission Execution (ME)
Sensor Data
Sensor Schedule
Sensor Data Fusion
Threat Assessment
Weapon Schedule
Data Information Services (DIS)
Planners, Assessors, & Decision-Makers (PAD)
Weapon / Asset Services (W/AS)
Mission Assessment
Weapon Report
Firing / KA Report
Simulator
Display, Database, Synchronizer, Recorder
Display, Database, Synchronizer, Recorder
Simulator
Simulator
Training
Training
Training
Support Functions
Common Services (CS)
Display, Database, Synchronizer, Recorder
Display, Database, Synchronizer, Recorder
Display, Database, Synchronizer, Recorder
Display, Database, Synchronizer, Recorder
OA Candidate Common Function / Application
OA Candidate Platform-Unique Function / Application
Kill Chain Data Flow/Loop
Network flow

Figure 23. Architecture Flow Diagram (AFD) for proposed OA model. The AFD provides a snapshot of the proposed architecture flows and key functions.

The following is a list and description of the modules shown in Figure 23:

- EXCOMM – includes all methods of communication and communication service actions used by force planners and coordinators to transmit mission orders.
- Search and Detect (S&D) – includes all sensor tasking actions such as sensor availability and tracking reports.
- Data Information Services (DIS) – includes data fusion activities such as compilation, scheduling, and classification.
- Planners, Assessors, and Decision-Makers (PAD) – threat data is analyzed and assessed. Command and Control orders are issued to engage threat.
- Weapon/Asset Services (W/AS) – includes all weapon allocation and scheduling.
- Mission Execution (ME) – includes weapon assignment, threat engagement, and kill report.
- Common Services (CS) – includes services used by modules a through f listed above, such as displays, databases, synchronizers, and recorders.

The following is a description of the flows shown in Figure 23:

- Battle Force Commander Orders – orders issued at the Force Planning/Coordination (FP/C) level to execute a mission. These orders are issued to joint strike forces and strike groups.
- Satellite Communications (SATCOM), Radios, Data Links, and Networks – methods of communication used for mission coordination.
- Simulator – used for training on search, detection, decision-making, and mission execution.

The following is a description of the kill chain loop and flows shown in Figure 23:

- Sensor data – sensor assets provide threat track and intelligence reports to the DIS.

- Sensor data fusion – the DIS compiles and fuses the threat sensor data for analysis by the PAD.
- Threat assessment – after the PAD analyzes the fused threat sensor data, an assessment is made and a Course of Action (COA) determined. The COA is sent to the W/AS for weapon scheduling.
- Weapon schedule – the best weapon to engage the threat is scheduled for mission execution and kill.
- Firing/Kill Assessment (KA) report – after weapon engagement, a report is created to determine if the threat has been eliminated, or if re-engagement is necessary.
- Weapon report – a report is created to determine weapon status. This report covers salvos fired and remaining salvos.
- Mission assessment – mission status is assessed by the Firing/KA report.
- Sensor schedule – if threat re-engagement is necessary, then sensors are scheduled to provide track and intelligence reports.

The Architecture Interconnect Diagram (AID) reflects the channels in which information is transferred between the architecture modules (Hatley, Hruschka, Pirbhai, 2000). Since the focus of this project is on the development of an abstract architecture, specifications for physical channels will not be determined. However, the channels will form a local network that binds the architecture modules together. This local network will be made of wired and wireless connections. See Figure 24. As with the flows depicted on the AFD, all modules will have a common or shared connection leading to the Common Services module.

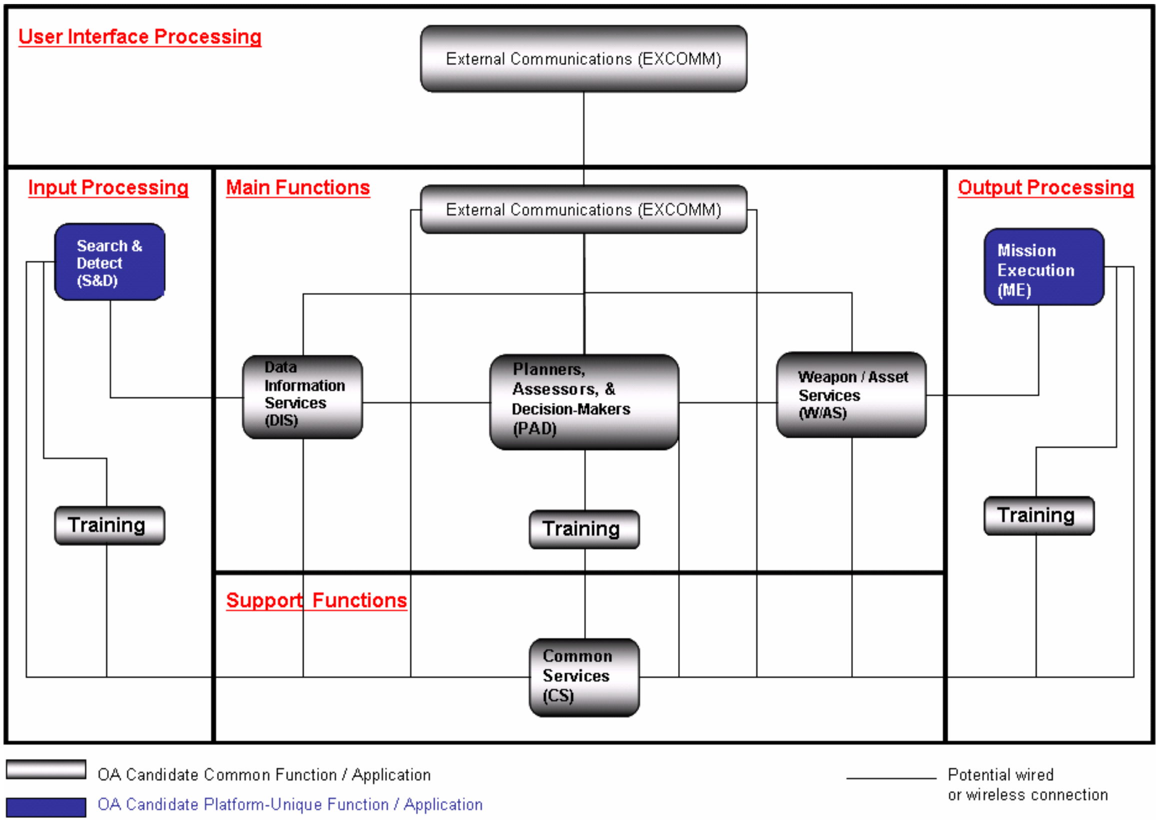

User Interface Processing
External Communications (EXCOMM)
Input Processing
Main Functions
External Communications (EXCOMM)
Output Processing
Search & Detect (S&D)
Mission Execution (ME)
Data Information Services (DIS)
Planners, Assessors, & Decision-Makers (PAD)
Weapon / Asset Services (W/AS)
Training
Training
Training
Support Functions
Common Services (CS)
OA Candidate Common Function / Application
OA Candidate Platform-Unique Function / Application
Potential wired or wireless connection

Figure 24. Architecture Interconnect Diagram (AID) for proposed OA model. The AID provides a snapshot of the proposed architecture interconnections between key functions.

E. FUNCTIONAL DESIGN ANALYSIS

1. Value System Design

The problem definition phase of our Systems Engineering Design Process entails the development of a Value System Design (VSD). The VSD methodology requires the definition of system functions that define what the system must do, objectives that indicate the preferred direction of attainment of an evaluation consideration (*Higher probability of kill*), a goal that shows the threshold of achievement (*Probability of kill ≥ 0.95*) and an evaluation measure that serves as a scale to measure the degree at which we attain an objective.

During the development of this VSD, it was difficult to obtain access to interview senior leaders and key stakeholders. Therefore, several group sessions and interviews were conducted with experienced systems engineers and stakeholder representatives.

The VSD model supports the Open Architecture functional domain model in a FORCEnet environment and its application to Cruise Missile Defense. In addition, this VSD was created at a level of abstraction to systematically support the operational Integrated Fire Control scenarios presented in *Integrated Fire Control for Future Aerospace Warfare* (Young, 2005). Only functional requirements are analyzed; non-functional requirements are not considered.

Modeling efforts commenced with allocating system functionality across the kill chain. Four broad levels of combat system functionality were developed and represent the high-level activities the CMD system will perform to execute the Integrated Fire Control and associated design concepts. The structure is functionally sub-divided into four major functional groups.

The functions and sub-functions represent the refined activities and provide a vehicle to develop the objectives necessary to achieve the result of the mission. In addition, these objectives are the next logical step towards developing the necessary evaluation measures needed to determine the fulfillment of each objective.

The value hierarchy includes functions and sub-functions that are encompassed in two network areas to ensure network communications and data sharing. One set of functions and associated sub-functions are related to the local network. The other set is related to the force network. Open Architecture candidates in both networks are further divided into OA Common functions and OA specific functions.

The VSD process allowed the authors to identify the system functionalities that are required to implement a cruise missile defense combat system using FORCEnet and OA design concepts. By going through the exercise of a VSD, the authors were able to verify and validate the requirements and problem space that would be used in the simulations providing the data for analysis. The VSD further insured that additional redundant or irrelevant requirements and functionality were not placed in the model, which had the potential to add unnecessary complexity and obfuscate the results.

The following discussion will follow a top-down flow of the functionality of the VSD. Figure 25 is the visual depiction of the value system hierarchy evolving from the need statement or the main function into the major functions. The main function of the VSD, the Provide Cruise Missile Defense using ASCMD, is functionally sub-divided into four major functions: Maintain Communications; Perform Search, Detection, and Tracking; Perform Command and Control; and Perform Engagement. These major functions are indexed at the top center from 1.0 to 4.0 respectively for further decomposition and analysis. The value of each major function compared to the value of the Provide Cruise Missile Defense function is shown at the lower right-hand side of each major function as a relative weight in decimal of the whole. The relative weight of each major function is added to one as the total weight at the lower right-hand of the main function. The relative weight of each major function is distributed not equally but accordingly with its role and importance to support or achieve the main function purpose. The Perform Search, Detection, and Tracking is considered the most important role to achieve the purpose of the Provide CMD using ASCMD function; thus, it contributes up to 0.30 or 30% of the main function weight; Maintain Communications and Perform Command & Control functions have the same contribution weight of 0.25 or 25%; and Perform Engagement has 0.20 or 20% weight because its performance needs support of other major functions. All the evaluation measures are considered natural and direct measures. These measures focus on the attainment of each of the stated objectives and can have a common interpretation. They focus on the greatest value of the objective either in hard terms of quantity such as “number of contacts: identified” and/or quality such as “accurate battlegroup (BG) COP.”

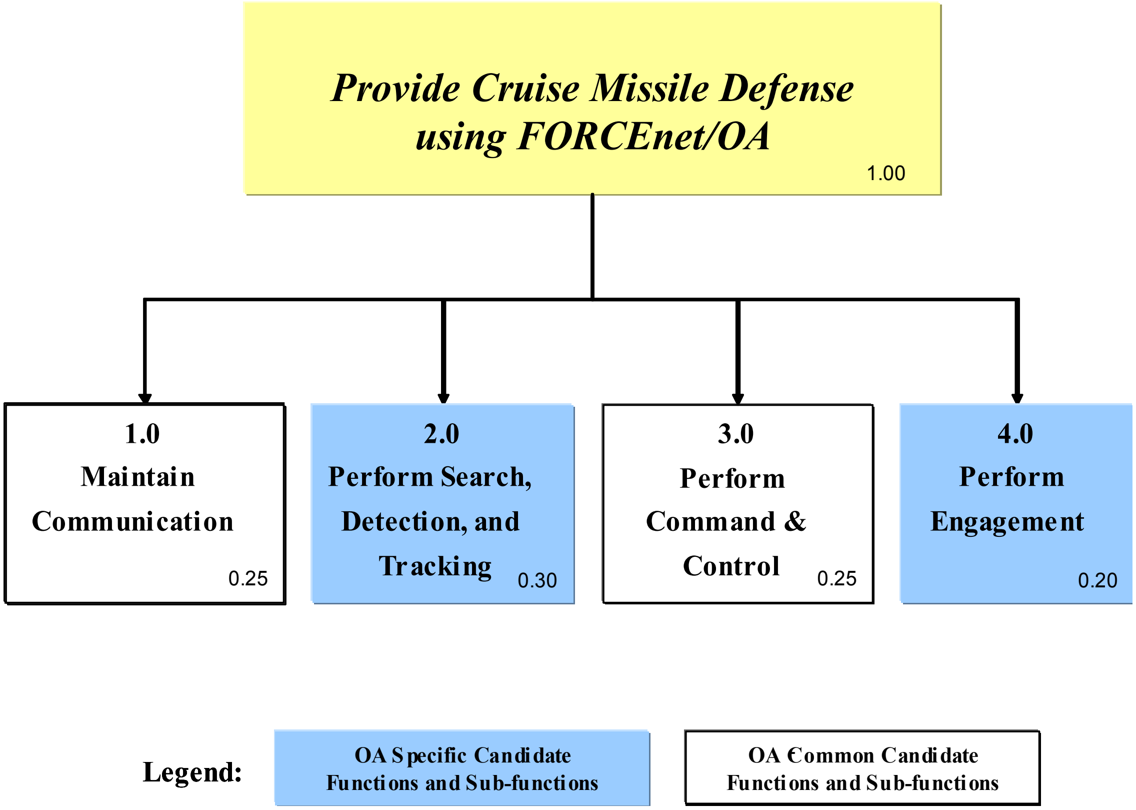

Provide Cruise Missile Defense using FORCEnet/OA
1.00
1.0 Maintain Communication
0.25
2.0 Perform Search, Detection, and Tracking
0.30
3.0 Perform Command & Control
0.25
4.0 Perform Engagement
0.20
Legend:
OA Specific Candidate Functions and Sub-functions
OA Common Candidate Functions and Sub-functions

Figure 25. Value System Design Hierarchy.
The representation of the highest-level required functions derived from the functional need statement.

The first major function, Maintain Communications, supports the communication requirements necessary for the IFC scenarios. It is an OA-specific function and is decomposed into two levels of sub-functions. In Figure 26, the top-level function of Maintain Communications has additional levels showing the sub-functions and their relative importance. The first level of decomposition includes three sub-functions decomposed from the major function: Maintain Local Network, Maintain Force Network, and Maintain IA. Then three sub-functions are further decomposed to one more level into eight lower sub-functions to capture all functionalities needed for the major function. Each level represents a level of abstraction of three decomposed functions or sub-functions. Their relative weights at each level represent their individual value in support of the next higher-level function.

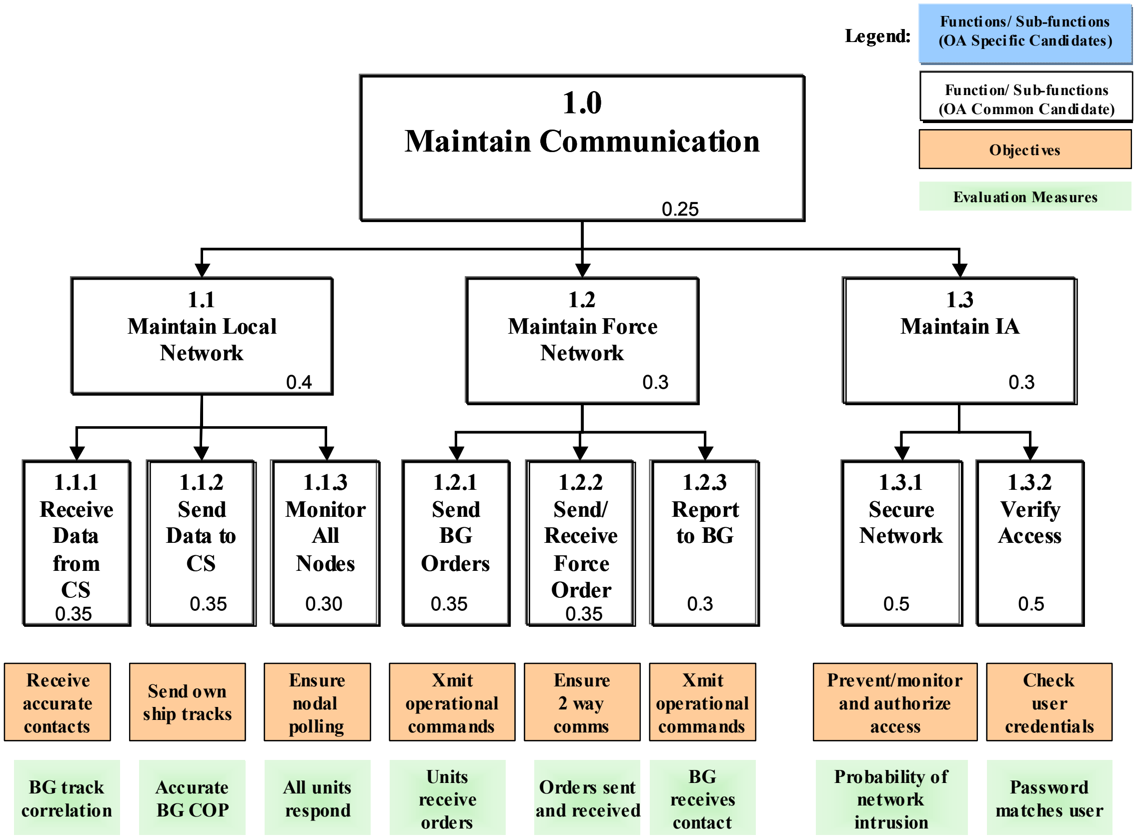

Legend:
Functions/ Sub-functions (OA Specific Candidates)
Function/ Sub-functions (OA Common Candidate)
Objectives
Evaluation Measures
1.0 Maintain Communication 0.25
1.1 Maintain Local Network 0.4
1.2 Maintain Force Network 0.3
1.3 Maintain IA 0.3
1.1.1 Receive Data from CS 0.35
1.1.2 Send Data to CS 0.35
1.1.3 Monitor All Nodes 0.30
1.2.1 Send BG Orders 0.35
1.2.2 Send/ Receive Force Order 0.35
1.2.3 Report to BG 0.3
1.3.1 Secure Network 0.5
1.3.2 Verify Access 0.5
Receive accurate contacts
Send own ship tracks
Ensure nodal polling
Xmit operational commands
Ensure 2 way comms
Xmit operational commands
Prevent/monitor and authorize access
Check user credentials
BG track correlation
Accurate BG COP
All units respond
Units receive orders
Orders sent and received
BG receives contact
Probability of network intrusion
Password matches user

Figure 26. Maintain Communications.
This represents the aggregation of sub-functions that form the Maintain Communications function.

The Maintain Local Network sub-function is responsible for ensuring that local area networks are able to support real-time data transfer of tactical information between organic combat, weapon, and support systems. It provides the functionality that is required for monitoring and troubleshooting data node issues within the platform. Its objectives are to send own ship tracks, receive accurate contact data, and perform nodal polling. Successful completion of these objectives results in BG track correlation, seamless data flow, and an accurate BG COP.

The Maintain Force Network sub-function allows for full interoperability between members of the CSG, ESG, and the CAW. The Composite Warfare Commander uses this functionality to send orders to the battlegroup, send and receive Force Orders, sharing and updating common tactical picture, receive mission status reports, and for communication with high-level headquarter commands.

The Maintain Information Assurance system sub-function ensures that those system operations required to protect and defend information and information systems are executed to ensure information availability, integrity, authentication, confidentiality, and non-repudiation. Proper monitoring and verification of the network provides preventive network security.

The next top-level block, Perform Search, Detection, and Tracking functionality, is an OA-specific function and is decomposed into two levels of sub-functions as shown in Figure 27. The major function is decomposed at the first level into three sub-functions including Perform Search, Perform Detection, and Perform Tracking. Each sub-function is further decomposed into two lower sub-functions. Similar to the decomposition of the Maintain Communication function, the relative weights are expressed at each level of decomposition.

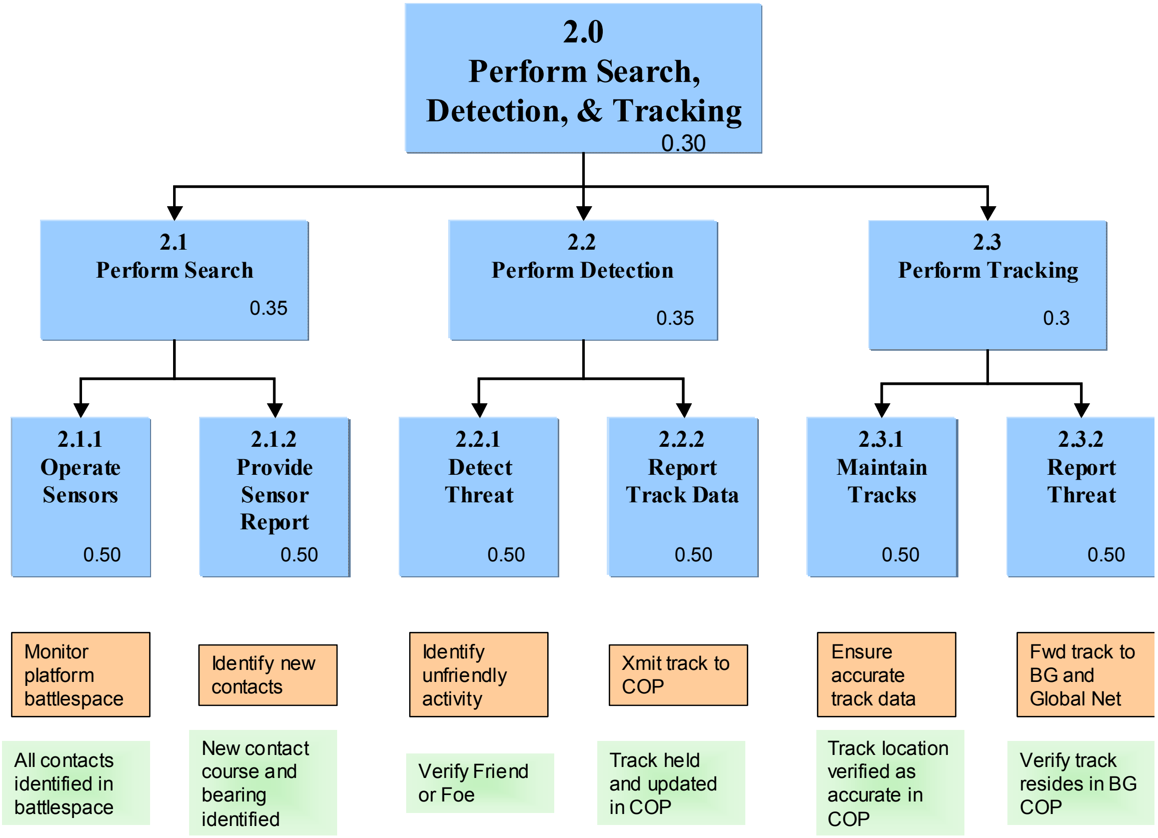


Figure 27. Value system decomposition of Perform Search, Detection, and Tracking.

The sub-function Perform Search is responsible for conducting surveillance using all available shipboard sensors. Sensors include radar systems such as continuous wave, frequency-modulated continuous wave, high-resolution, synthetic aperture radar, electronic scanning, and the phased array. Other sensors include electro-optical thermal imagers, laser radar systems, electronic support systems, and sonar. Sensor reports are provided for all detections within the sensor coverage for further assessment. The Perform Search functionality will respond to a track cueing from local or remote sensors to conduct a targeted sensor search in the battlespace tracks of interest. Knowledge and intention of all platforms within the battlespace are required.

Perform Detection is the sub-function that evaluates the behavior of an object in order to estimate type, quantity, radar cross-section, and identity. Part of this evaluation entails the discrimination of targets in a sea clutter or noise environment. The object must be identified as friend or foe, submitted to, and held by the COP.

The final sub-function in the Perform Search Detection and Tracking block is the Perform Tracking. This function is responsible for maintaining a system track once the primary sensor is selected and scheduled by the combat system. Track data is used for maintaining and updating COP and for conducting threat assessment on possible hostile threats or monitoring targets of interest for follow-up actions. The track data must be verified as accurate prior to inclusion in the global grid.

C2 is the most critical element of a cruise missile defense combat system. The C2 process includes the mission planning, directing, and coordinating with local forces and higher-level commands as well as controlling local forces and operations. The system of command and control includes the personnel, computer programs, equipment, communications, facilities, and procedures employed by the Composite Warfare Commander. This functionality is part of the OA Common functional architecture. Perform Command and Control is decomposed into three levels of sub-functions with associated relative weights as shown in Figure 28.

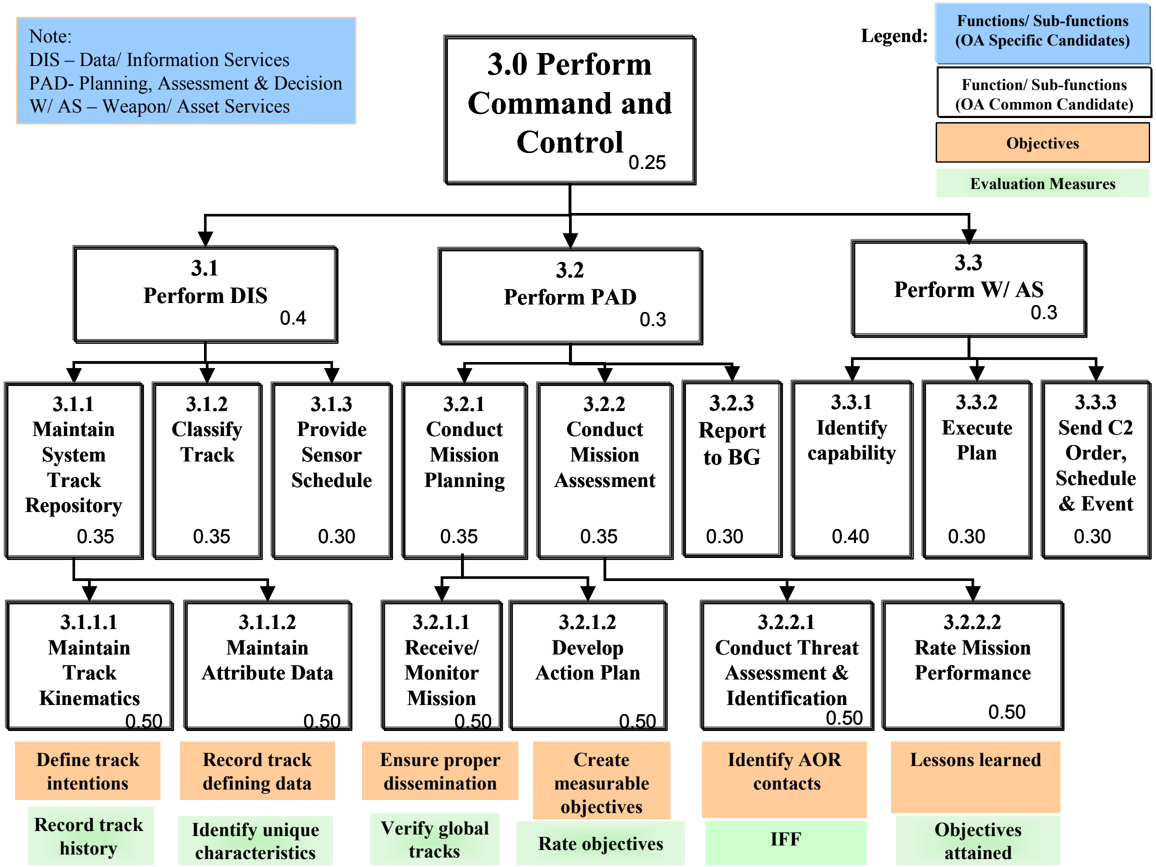

Note:
DIS – Data/ Information Services
PAD- Planning, Assessment & Decision
W/ AS – Weapon/ Asset Services
3.0 Perform Command and Control 0.25
Legend:
Functions/ Sub-functions (OA Specific Candidates)
Function/ Sub-functions (OA Common Candidate)
Objectives
Evaluation Measures
3.1 Perform DIS 0.4
3.2 Perform PAD 0.3
3.3 Perform W/ AS 0.3
3.1.1 Maintain System Track Repository 0.35
3.1.2 Classify Track 0.35
3.1.3 Provide Sensor Schedule 0.30
3.2.1 Conduct Mission Planning 0.35
3.2.2 Conduct Mission Assessment 0.35
3.2.3 Report to BG 0.30
3.3.1 Identify capability 0.40
3.3.2 Execute Plan 0.30
3.3.3 Send C2 Order, Schedule & Event 0.30
3.1.1.1 Maintain Track Kinematics 0.50
3.1.1.2 Maintain Attribute Data 0.50
3.2.1.1 Receive/ Monitor Mission 0.50
3.2.1.2 Develop Action Plan 0.50
3.2.2.1 Conduct Threat Assessment & Identification 0.50
3.2.2.2 Rate Mission Performance 0.50
Define track intentions
Record track defining data
Ensure proper dissemination
Create measurable objectives
Identify AOR contacts
Lessons learned
Record track history
Identify unique characteristics
Verify global tracks
Rate objectives
IFF
Objectives attained

Figure 28. Value system decomposition for Perform Command and Control function. This represents the aggregation of sub-functions that form the Perform Command and Control function.

Perform Data/Information Services (DIS) is the function responsible for maintaining situational awareness at the battleforce level by contributing to the establishment and maintenance of the COP through track data fusion from multiple organic sensors and other sensors within the battleforce. DIS is also responsible for classifying and tracking kinematics of system tracks in order to define track intentions and provide the ability to schedule needed sensors.

The Perform Planning, Assessment, and Decision (PAD) functionality provides the ability to conduct mission planning and assessment (mission and threat) as well as the ability to communicate and report mission plans and status to the battle group. The PAD will also provide firing policy determination (Shoot-Look-Shoot (S-L-S), Shoot-Shoot (S-S), Shoot-Look-Shoot-Shoot (S-L-SS), or Shoot-Shoot-Look-Shoot-Shoot (SS-L-SS)) for the different weapon systems. Proper completion of mission planning can be determined by thorough dissemination of action plan, threat assessment completion, and objectives attained.

Finally, the Perform Weapon/Asset Services ((W/AS) provides the functionality required to control weapons, remote vehicles, and engineering assets and services. This function is responsible for scheduling the required weapon and support assets to counter an inbound threat in support of cruise missile defense missions.

As the major function at the end of the kill chain, the Perform Engagement functionality has only the two sub-functions at the first level of abstraction, which are Engage Threat and Control Engagement.

The Engage Threat function conducts the engageability calculations required to develop a fire control solution, designate preferred weapon scheduling, and support systems based on established ROE's and weapon firing policy. Once the target is verified, engaging the threat will consist of confirmation of engagement orders to verify threat data, selection of interceptor to determine best target solution, and launching interceptor to engage the threat.

The Control Engagement functionality will provide the ability to maintain in-flight local or remote control of the interceptor as well as kill assessment reporting. Evaluation of the hit or miss will be confirmed with sensors and other damage assessors.

The Perform Engagement VSD decomposition is shown below in Figure 29. The major function is decomposed into its component parts to the third level of abstraction. Additionally, the relative value associated with each sub-function is shown and expressed as a decimal percentage. These values may serve as a baseline for weighting relative performance of each sub-function relative to the major function in future evaluations.

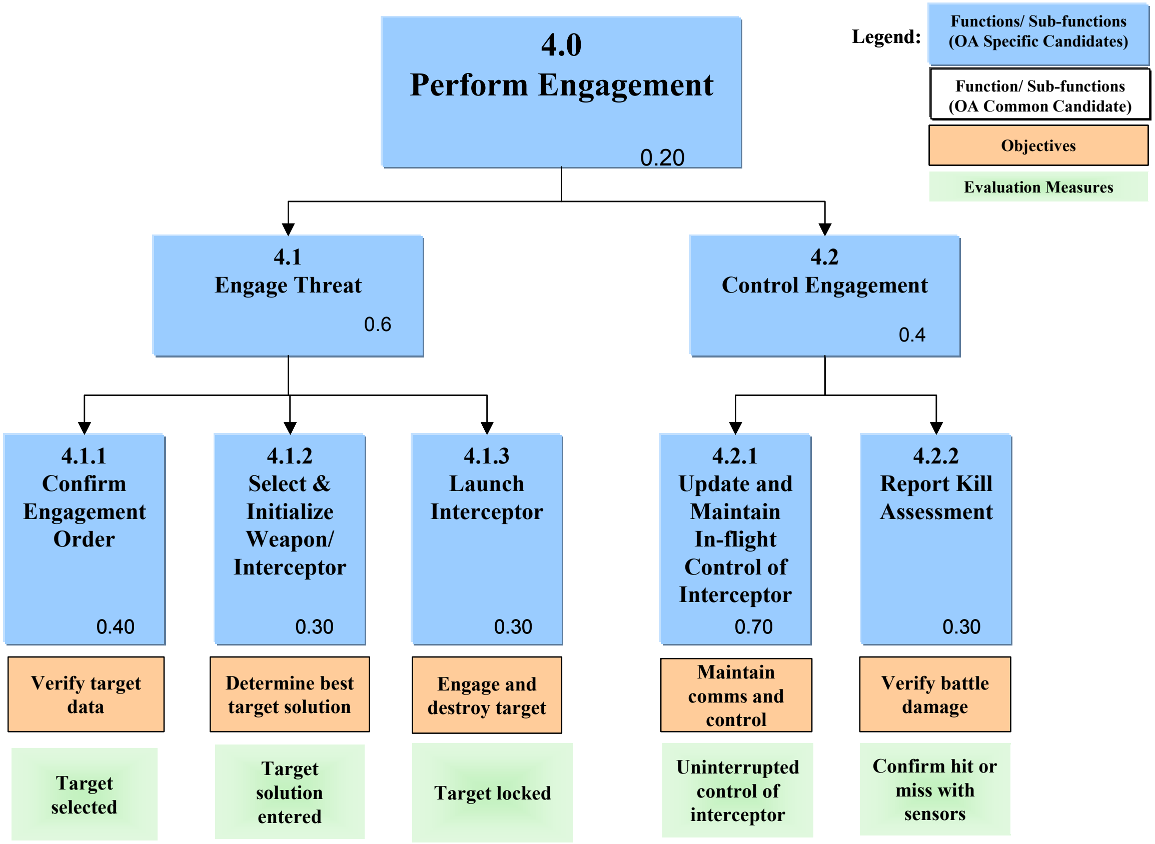

4.0
Perform Engagement
0.20
Legend:
Functions/ Sub-functions (OA Specific Candidates)
Function/ Sub-functions (OA Common Candidate)
Objectives
Evaluation Measures
4.1
Engage Threat
0.6
4.2
Control Engagement
0.4
4.1.1
Confirm Engagement Order
0.40
4.1.2
Select & Initialize Weapon/ Interceptor
0.30
4.1.3
Launch Interceptor
0.30
4.2.1
Update and Maintain In-flight Control of Interceptor
0.70
4.2.2
Report Kill Assessment
0.30
Verify target data
Determine best target solution
Engage and destroy target
Maintain comms and control
Verify battle damage
Target selected
Target solution entered
Target locked
Uninterrupted control of interceptor
Confirm hit or miss with sensors

Figure 29. Value system decomposition for Perform Engagement function. Engage Threat and Control Engagement form the higher-level functions.

2. Functional Flow Block Diagram

To establish how the functions relate and interface within the system, a Functional Flow Block Diagram (FFBD) is developed. FFBD is defined as "a formal technique for defining lower-lever functions and sequencing relationships using a formatted, consistent graphical methodology which includes function blocks, flow connections and directions and various ways of showing linkages between functional events and their traceability to higher-level functions" (Defense Acquisition University, 2007). FFBD's are broken down into first, second, and third functional decomposition levels, identifying the task sequences or order of execution and relationships among the functions. The FFBDs are then employed as a reference when constructing the Arena simulation model. The high-level FFBD is broken down into the main functional blocks depicted on Figure 30.

Constant communication with all nodes must be maintained, therefore the Maintain Communications functionality is in parallel with the rest of the function blocks. Blocks 2.0 through 4.0 encompass concepts from John Boyd's OODA loop.

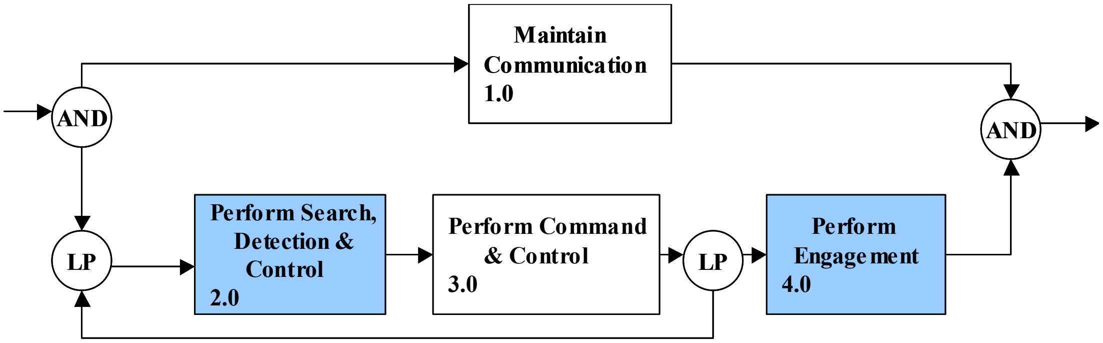

Maintain Communication 1.0
AND
AND
LP
Perform Search, Detection & Control 2.0
Perform Command & Control 3.0
LP
Perform Engagement 4.0

Figure 30. High-level FFBD for Maintain Communications objective. Maintain Communications is in parallel with function blocks 2.0, 3.0, 4.0.

All three sub-functions of the primary function, Maintain Communication (1.1 through 1.3), must be executed in parallel to achieve effective communications within a FORCEnet environment. See Figure 31 below.

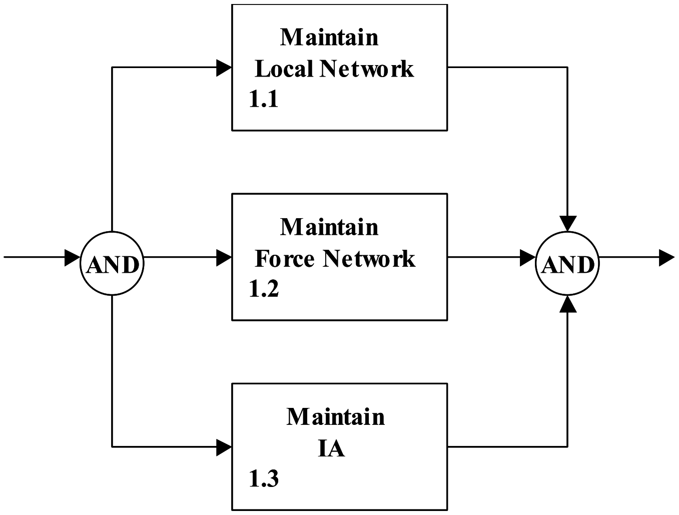


Figure 31. FFBD for Maintain Communications sub-functions. The sub-functions Maintain Local Network, Maintain Force Network, and Maintain IA all occur in parallel.

Similar to an Intranet, the Maintain Local Network, Figure 32, has the functions of either send or receive data to/from the CS, while monitoring all nodes simultaneously within the platform.

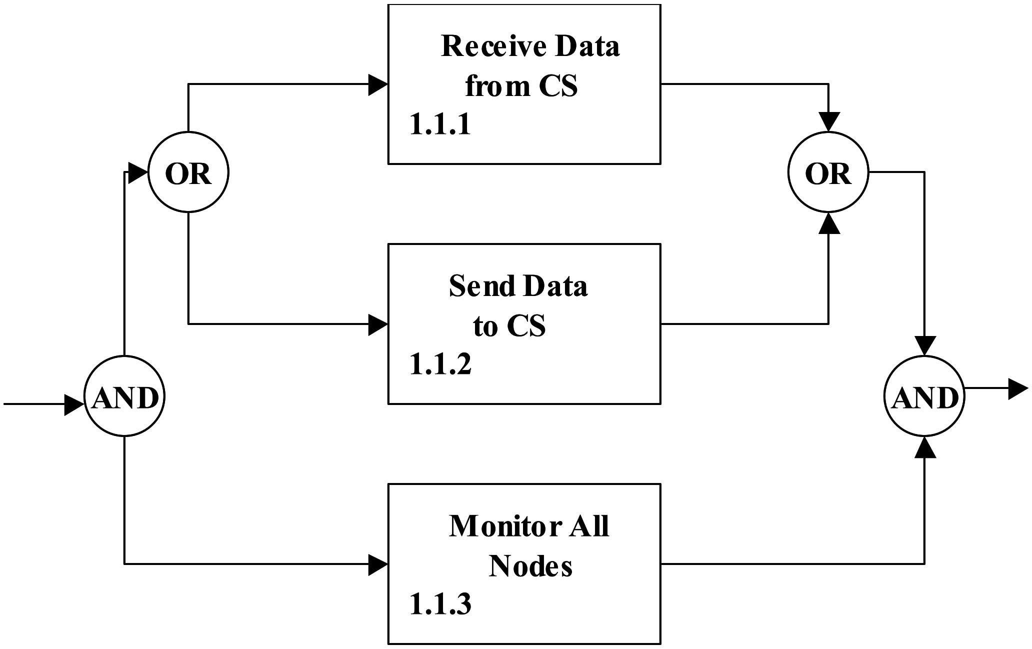


Figure 32. Maintain Local Network FFBD.
Either Receive or Send Data from CS can be performed, which are in parallel to Monitor All Nodes.

Similarly, the Maintain Force Network functions achieve the same concept of communications as illustrated above, but in this case it is concerned with the exchange of data with external entities via EXCOMM. See Figure 33 below.

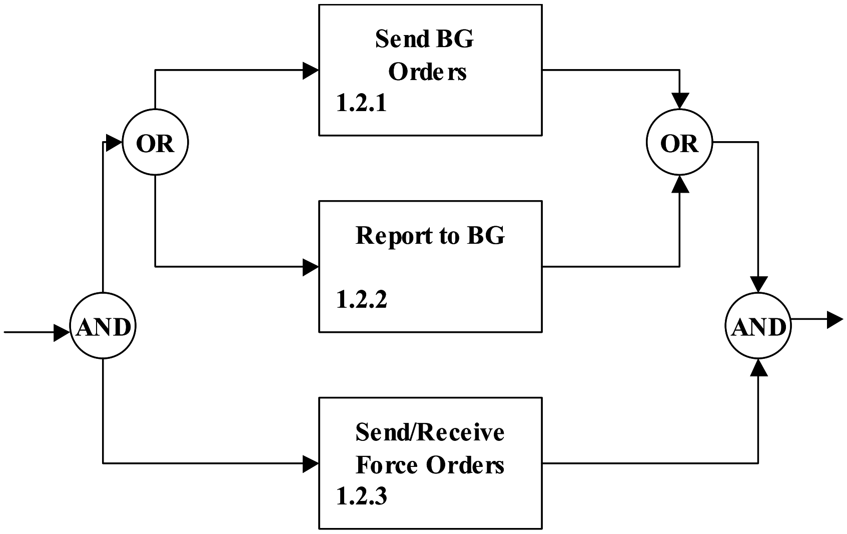


Figure 33. Maintain Force Network FFBD.
Either Send BG Orders or Report to BG can be performed, which operate in parallel to Send/Receive Force Order.

With the exchange of information comes the responsibility of safeguarding and ensuring that the information is not compromised in any way. Figure 34 below reflects the Maintain IA FFBD. These functions must occur at all times and uninterruptedly in order to achieve a secure network.

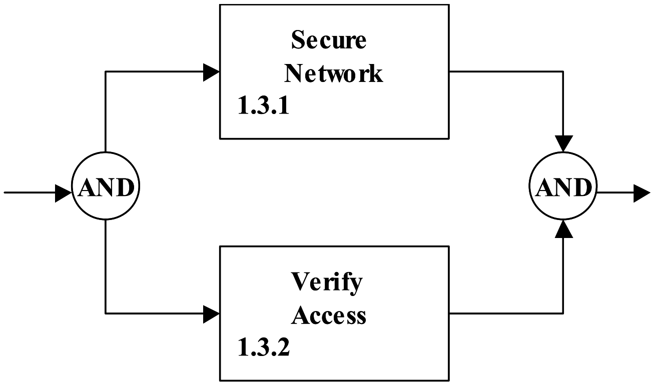


Figure 34. Maintain IA FFBD.
Secure Network and Verify Access operate in parallel to ensure data protection.

Figure 35 shows the major function subgroup that is responsible for the search, detection, and control of threats. The search, detect, and track functions occur in series and then are repeated in a loop.

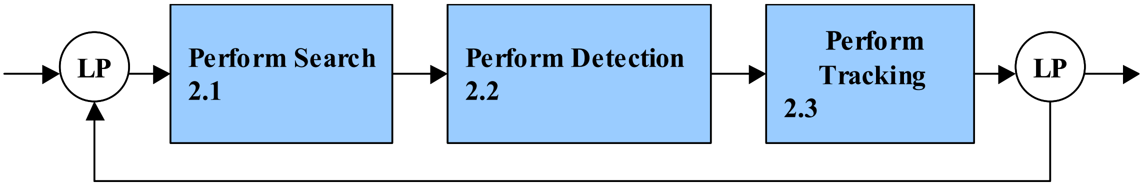


Figure 35. Perform Search Detection & Control FFBD.
These functions occur in series and can loop.

Figure 36 shows the Perform Search FFBD. Surveillance is conducted by multiple sensors, which operate continuously providing information.

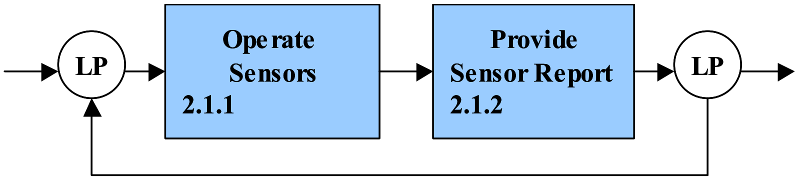


Figure 36. Perform Search FFBD.
These functions occur in series and loop as they continually update information.

The same concept as in Figure 36 above is reflected in Figure 37 below for the Perform Detection FFBD. Once the threat is detected, a track report is generated and updated on a continuous basis. The Perform Tracking FFBD in Figure 38 establishes a continuous process of updating the Common Operational Picture (COP) and maintaining all entities informed.

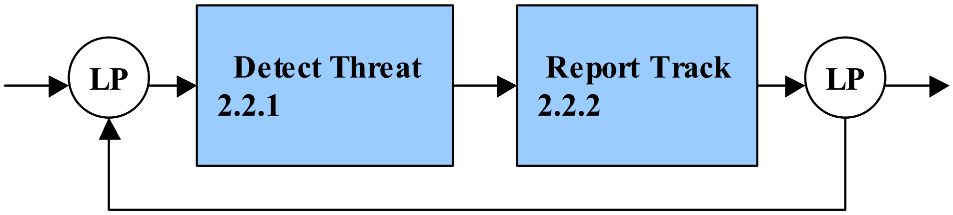


Figure 37. Perform Detection FFBD.
Once a threat is detected a track report is generated and updated continually.

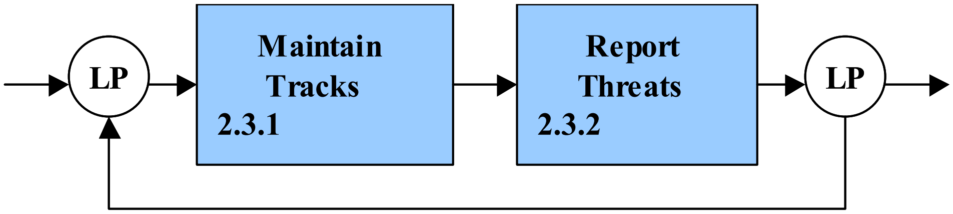


Figure 38. Perform Tracking FFBD.
The COP is updated on a continuing basis.

The C2 major sub-function group handles the DIS, PAD, and W/AS. These functions occur in series. The track picture along with classification is fed to the mission-planning portion of the combat system, and then a weapon is scheduled to defend against the cruise missile threat. All of this happens on a continuous basis as described in Figure 39 below.

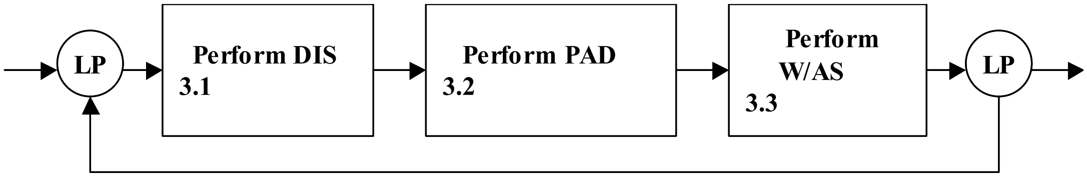


Figure 39. Perform Command & Control FFBD.
These functions occur in series and loop continually.

Figure 40 shows the Perform DIS FFBD. There are two paths. Either the DIS is providing sensor schedule, or it is maintaining and classifying the track database.

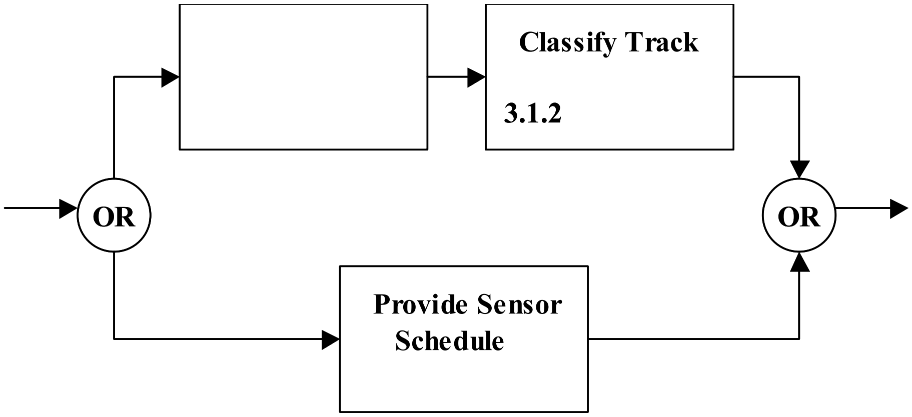


Figure 40. Perform DIS FFBD.
The DIS can perform either of these functions.

With PAD functions, as pictured in Figure 41, the platform will be either planning a mission, or evaluating a mission. Once this is done, the platform is expected to report the status to the BG.

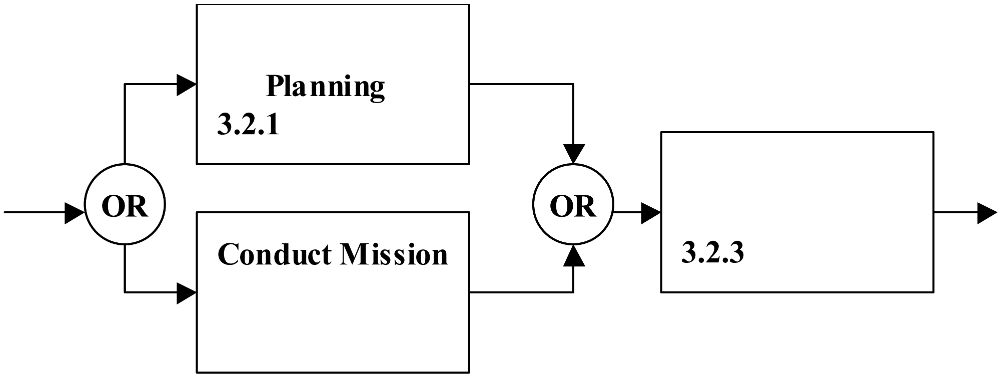


Figure 41. Perform PAD FFBD.
Conduct of Mission Planning or Assessment must be performed before the Report to the Battlegroup.

Figure 42 represents the Perform W/AS FFBD. Functions dealing with the Weapon/Asset Services involve a need either to identify the capability or execute the plan, which triggers the Send C2 Order, Schedule, and Event block.

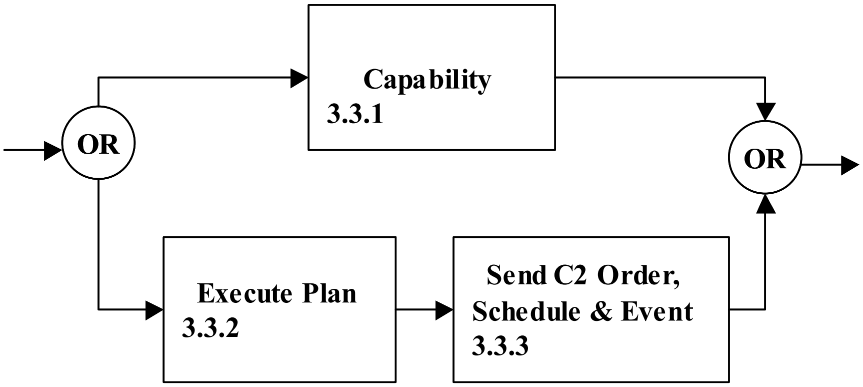


Figure 42. Perform W/AS FFBD.
A capability must be identified or a plan executed and the C2 order given.

Figures 43, 44, and 45 illustrate third functionality levels for additional granularity.

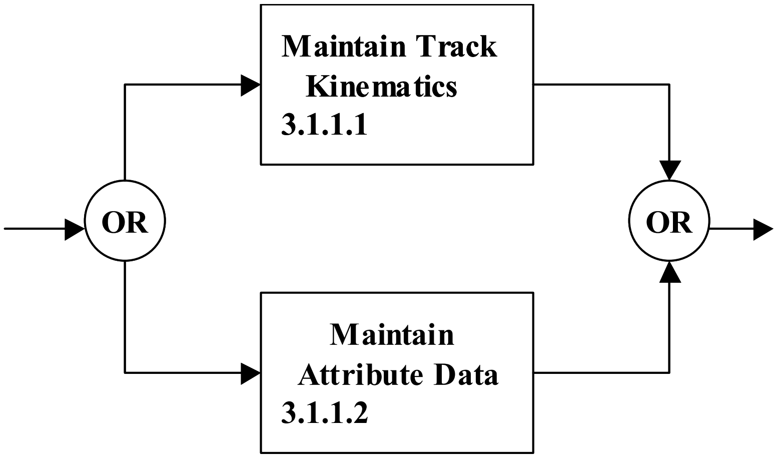


Figure 43. Maintain System Track Repository FFBD.
Either Maintain Track Kinematics or Maintain Attribute Data can be performed.

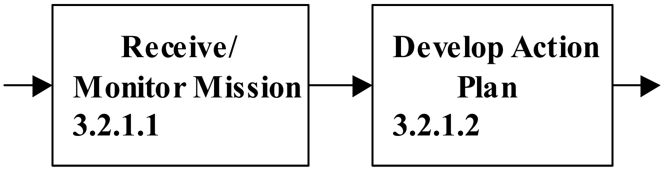
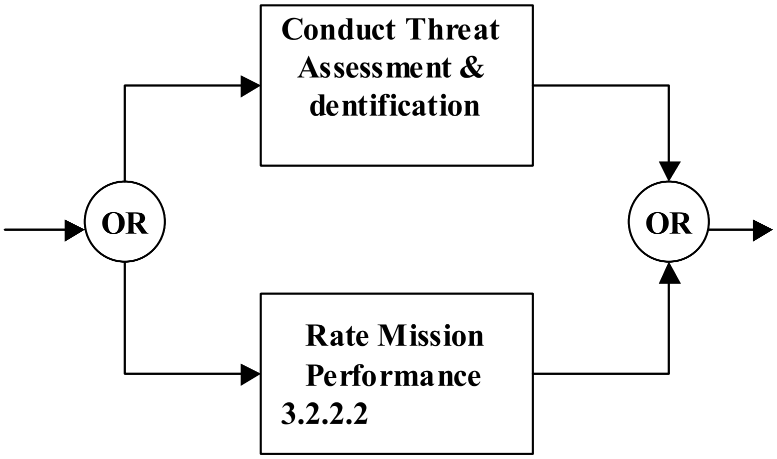


Figure 44. Conduct Mission Planning FFBD.
Mission planning functions occur in series.



Conduct Threat Assessment & dentification
OR
OR
Rate Mission Performance 3.2.2.2

Figure 45. Conduct Mission Assessment FFBD.
Either Conduct Threat Assessment & ID or Rate Mission Performance can be performed.

Once a defensive measure is launched in the form of an intercepting missile, the fire solution is calculated, missile is launched, and then guidance or remote control of the missile takes place in series, which forms the Perform Engagement function. The process is repeated depending on the results of the kill evaluation. See Figure 46.

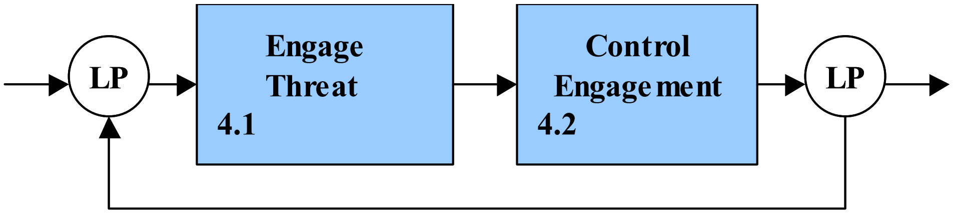


Figure 46. Perform Engagement FFBD.
These functions will occur in a loop if re-engagement is necessary.

Figure 47 illustrates the launching functions for the Engage Threat FFBD in series.

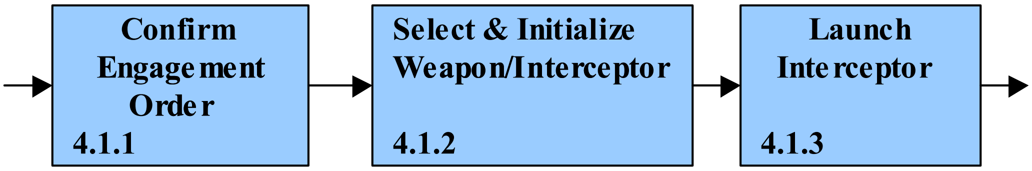


Figure 47. Engage Threat FFBD.
Processes within the threat engagement must occur in series.

While the missile is in the air, the interceptor is guided, and a kill assessment is done at the end of flight. See Figure 48.

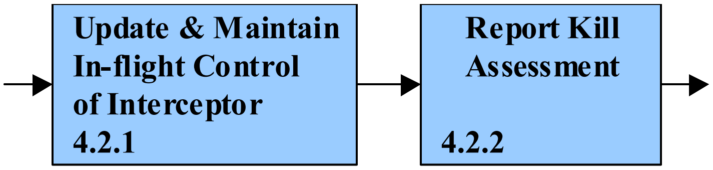


Figure 48. Control Engagement FFBD.
The functions within Control Engagement must occur in series.

3. Use Cases

Eleven use cases are identified and examined through the OA functional architecture for CMD. These use cases are Verify Access, Detect Target, Send Threat Data, Receive Threat Data, Maintain Track, Determine the Preferred Shooter, Make Firing Decision, Fire Weapon, Pass Engagement Control, Receive Engagement Control, and Control Engagement. These use cases are used in combination to perform six IFC scenarios proposed in *Integrated Fire Control for Future Aerospace Warfare* (Young, 2005). These use cases and corresponding sequence diagrams create the path into the ASCMD simulation model. The format used for all use case diagrams, textual representations, and sequence diagrams are in accordance with the textbook *Object-Oriented Software Engineering* (Bruegge, Dutoit, 2004). System responses on the textual representation of the use cases are indented to distinguish from the actor's actions. Figure 49 depicts the high-level ASCMD use case.

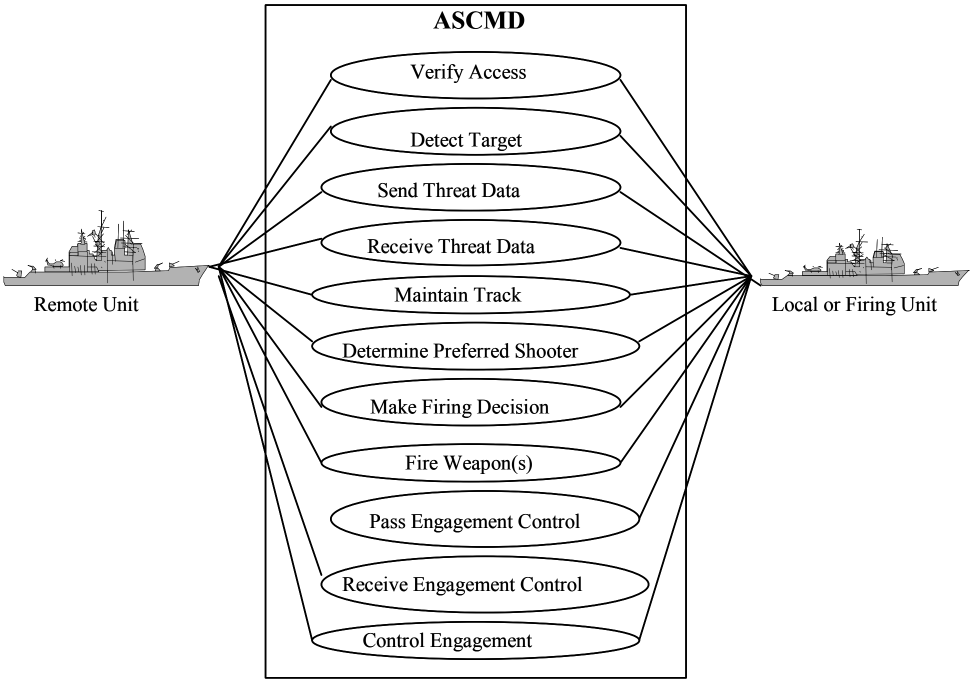


Figure 49. ASCMD use case diagram.
This diagram covers all the functions that will be simulated in the ASCMD simulation model.

The Verify Access use case verifies the unit Identification (ID) for access to the network. When a remote or local unit logs onto the network, the system will verify the unit's identification through Force Planning (FP)/Coordination to deny or grant network access. All units are required to be identified and established as nodes on the network for communication throughout the network. This use case is one of the tools used to maintain information assurance for the network since OA is a web-based architecture that relies heavily on an Internet Protocol (IP)-based environment (Rushton, 2004). The Verify Access use case is required before any other use cases can be performed. Network security is also maintained by terminating operator's access if there is no action detected or in case of power shutdown. Flow of events and entry/exit conditions required for this use case are described in Table 4.

Use Case Name	Verify Access
Participating Actors	Remote or Local Unit
Flow of Events	1. Remote or local unit initiates Network Access through Common Services (CS). 2. CS sends Access Request to Force Planning / Coordination (FP/C) via EXCOMM. 3. FP verifies access request and responds to CS of requesting unit via network of EXCOMM. 4. CS displays/notifies Access Request status (Granted or Denied) to requesting Unit and maintains network access if granted.
Entry Condition	All units are established as nodes on network for the communication within Battlegroup. Interface between OA common function and platform-unique function is available.
Exit Condition	Network access is granted. System automatically terminates access if there is no operator's input in 10 minutes or system power down.
Quality Requirements	Time required for verifying access is no more than 10 seconds.

Table 4. Use Case: Verify Access.
The unit Identification (ID) for access to the network is verified.

Figure 50 describes the sequence of events in the Verify Access use case between the unit and the system. The unit accesses the network through Common Services (CS). CS includes displays, input and output control consoles (IOCC), and other monitoring services as database and time synchronizers. Unit's ID is sent through EXCOMM to FP for verification and denied or granted access, then sent back to unit at the CS display. EXCOMM also disseminates unit's ID to all nodes for sharing data and communication. Figure 51 shows the functional flow block diagram of the events.

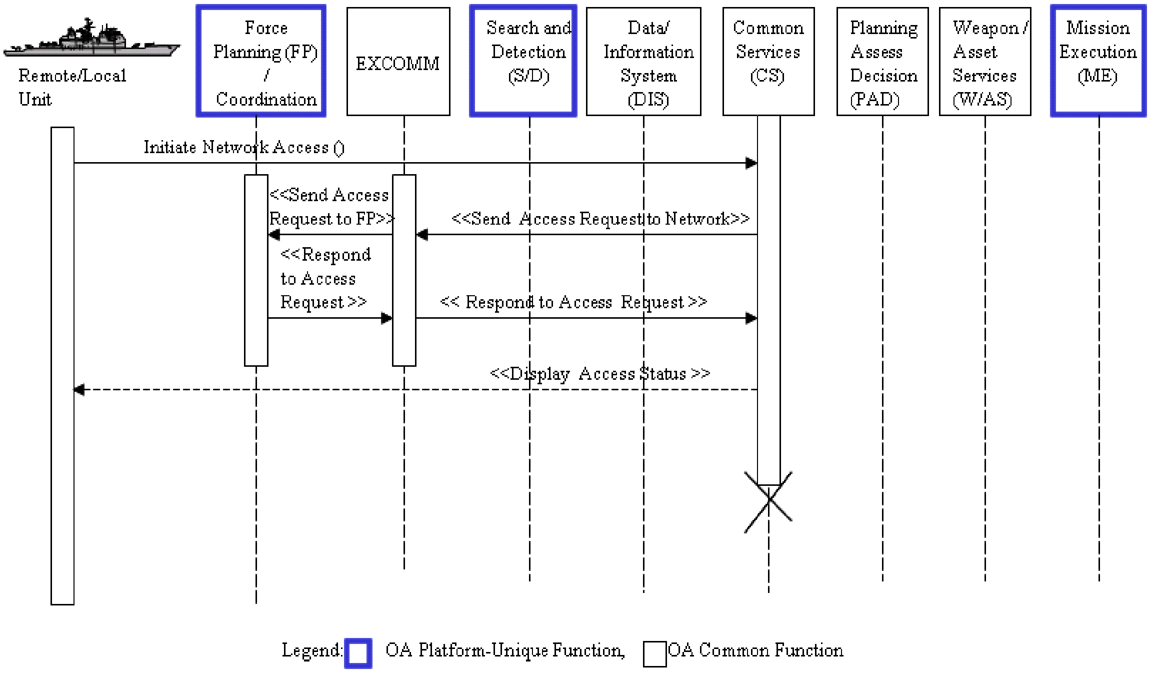

Remote/Local Unit
Force Planning (FP) / Coordination
EXCOMM
Search and Detection (S/D)
Data/ Information System (DIS)
Common Services (CS)
Planning Assess Decision (PAD)
Weapon / Asset Services (W/AS)
Mission Execution (ME)
Initiate Network Access ()
<<Send Access Request to FP>>
<<Send Access Request to Network>>
<<Respond to Access Request >>
<< Respond to Access Request >>
<<Display Access Status >>
Legend: OA Platform-Unique Function, OA Common Function

Figure 50. Verify Access sequence diagram.
Timing and sequence of events are reflected.

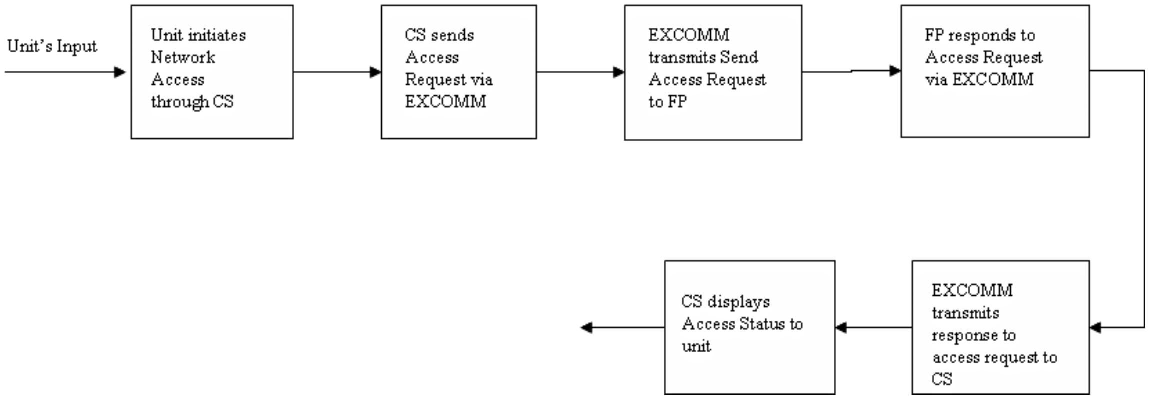

Unit's Input
Unit initiates Network Access through CS
CS sends Access Request via EXCOMM
EXCOMM transmits Send Access Request to FP
FP responds to Access Request via EXCOMM
EXCOMM transmits response to access request to CS
CS displays Access Status to unit

Figure 51. Flow of events for the Verify Access use case.

The next use case explored is Detect Target. Either a remote or local unit can use the Detect Target use case to search and detect threats in the environment, but a remote unit is more favorable for this use case because it can provide an earlier warning increase reaction time. The searching unit initiates a search function at the IOCC of CS and the system will automatically perform sequential actions and responses. CS sends search request to Data Information Services (DIS) and DIS schedules sensors from Search and

Detect (S&D) to search. If a target is detected, S&D reports sensor data to DIS and displays at CS. DIS fuses sensor data and sends to Planning, Assessment, and Decision (PAD) for threat assessment. Weapon/ Asset Services (W/AS) receives threat data from PAD, assesses best weapon response, and sends response to PAD for decision. Threat data is distributed to all nodes of the network through the web-based C2 environment. All units are aware of the battlespace environment, enemy course of action, disposition of own force, and logistics tail (Rushton, 2004). Detection is very critical because it is the first step in the kill chain: Detect, Control, and Engagement. The earlier the warning is received, the higher the probability of survival. Search time has no limits, but time from target detection to threat dissemination through the network can be complete in the first 10 seconds. Flow of events and entry/exit conditions required for this use case are described in Table 5.

Figure 52 describes the sequence of events in the Detect Target use case between the unit and the system and where the information is distributed through multicast. The unit starts its search function through the CS. CS includes displays, IOCC, and other monitoring services as database and time synchronizers. CS sends request to DIS. DIS keeps track repository, system track, and track kinematics for scheduling sensors or data fusion. DIS receives search request, schedules sensors, and receives sensor data from S&D. Sensor data received from S&D is fused at DIS and sent to PAD. PAD includes tactical picture, threat assessment, mission assignment, or C2 order, and aid in making decision. PAD assesses threat and sends to W/AS for weapon assessment. W/AS performs weapon inventory, reports or assesses weapon availability, and schedules weapon for engagement or re-engagement. Weapon assessment from W/AS is sent to PAD to complete tactical picture, assign mission or C2 order, and aid in making decision. All units in the battlespace are informed and share all threat data through EXCOMM. EXCOMM includes data links, satellite, and radio for data transmission and communication. EXCOMM disseminates data to all units on the network for situation awareness and readiness. After detecting and disseminating the threat, the search unit keeps monitoring the target to update track for follow-up action. Figure 53 shows the FFBD of the events.

Use Case Name	Detect Target
Participating Actors	Remote or Local Unit
Flow of Events	1. The searching unit initiates search at Common Services (CS). 2. CS sends search request to Data Information Services (DIS) and informs search-in-progress to other units and Force Planning/Coordination (FP/C) through EXCOMM 3. DIS schedules sensors of Search/Detection (S/D) to search. 4. The searching unit monitors environment through CS. 5. Target detected by sensors from S/D is reported to CS and sent to DIS for track repository. 6. DIS requests sensor data from S/D for fusion. 7. S/D reports sensor data to CS for recording and display, and to DIS for fusion. 8. DIS reports fused data to CS for recording and sends fused data to Planning Assessment Decision (PAD) for threat assessment and to other units including FP/C. 9. PAD assesses threat and reports to detecting unit at CS, to W/AS for capability assessment, to other units and FP through EXCOMM. 10. The searching unit keeps monitoring or tracking the threat for follow up action 11. W/AS reports capability against threat to searching unit through CS and PAD, and disseminates to other units and FP.
Entry Condition	Interface between OA common function and platform-unique function is available and network access is granted.
Exit Condition	Target is detected, reported, and disseminated.
Quality Requirements	Time from detecting to disseminating capability against threat is no more than 10 seconds.

Table 5. Use Case: Detect Target
Threats are searched and detected in the environment.

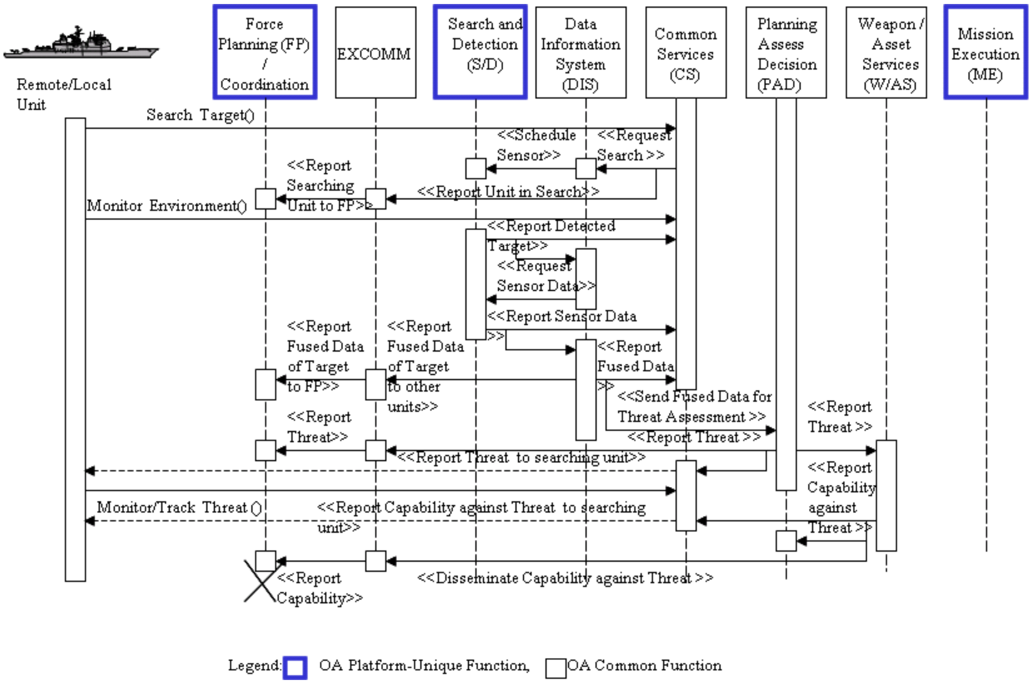

Remote/Local Unit
Force Planning (FP) / Coordination
EXCOMM
Search and Detection (S/D)
Data Information System (DIS)
Common Services (CS)
Planning Assess Decision (PAD)
Weapon / Asset Services (W/AS)
Mission Execution (ME)
Search Target()
<<Schedule Sensor>>
<<Request Search >>
<<Report Searching Unit to FP>>
<<Report Unit in Search>>
Monitor Environment()
<<Report Detected Target>>
<<Request Sensor Data>>
<<Report Sensor Data >>
<<Report Fused Data of Target to FP>>
<<Report Fused Data of Target to other units>>
<<Report Fused Data>>
<<Send Fused Data for Threat Assessment >>
<<Report Threat >>
<<Report Threat>>
<<Report Threat >>
<<Report Threat to searching unit>>
<<Report Capability against Threat >>
Monitor/Track Threat ()
<<Report Capability against Threat to searching unit>>
<<Report Capability>>
<<Disseminate Capability against Threat >>
Legend:
OA Platform-Unique Function,
OA Common Function

Figure 52. Detect Target sequence diagram.
Timing and sequence of events are reflected.

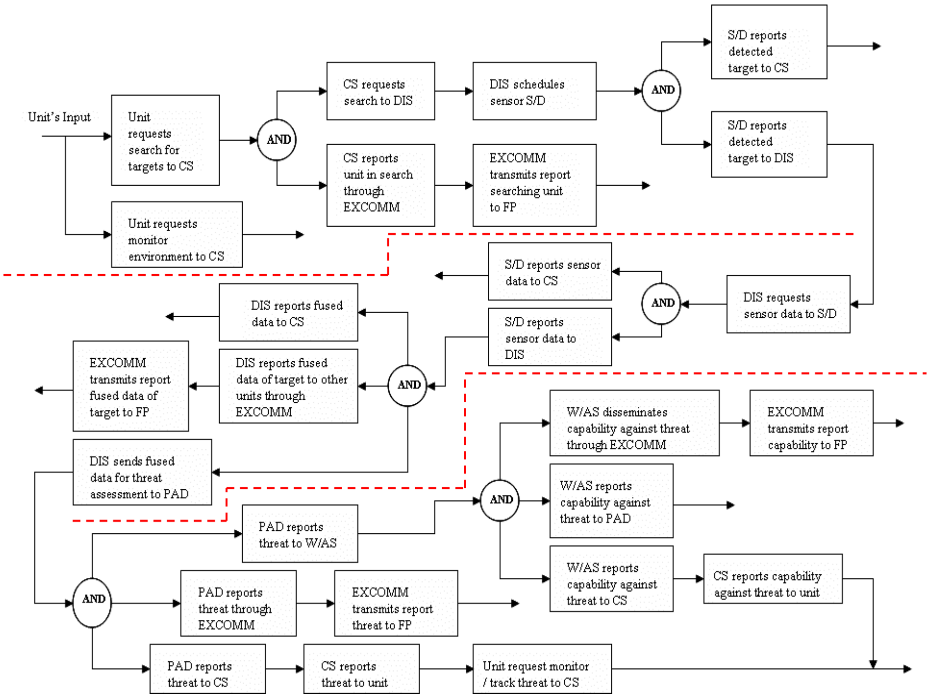

Unit's Input
Unit requests search for targets to CS
AND
CS requests search to DIS
DIS schedules sensor S/D
AND
S/D reports detected target to CS
S/D reports detected target to DIS
CS reports unit in search through EXCOMM
EXCOMM transmits report searching unit to FP
Unit requests monitor environment to CS
S/D reports sensor data to CS
AND
DIS requests sensor data to S/D
S/D reports sensor data to DIS
DIS reports fused data to CS
AND
DIS reports fused data of target to other units through EXCOMM
EXCOMM transmits report fused data of target to FP
DIS sends fused data for threat assessment to PAD
W/AS disseminates capability against threat through EXCOMM
EXCOMM transmits report capability to FP
AND
W/AS reports capability against threat to PAD
PAD reports threat to W/AS
W/AS reports capability against threat to CS
CS reports capability against threat to unit
AND
PAD reports threat through EXCOMM
EXCOMM transmits report threat to FP
PAD reports threat to CS
CS reports threat to unit
Unit request monitor / track threat to CS

Figure 53. Flow of events (FFBD) for Detect Target use case.

In the Send Threat Data use case either a remote or a local unit can send threat data to another unit in the environment. This use case is widely used by detecting units sending threat data to a firing unit or by a local unit to keep tracking a threat for further action. Although threat data is disseminated through the network for sharing, this use case is still needed because a unit in action may be a newly joint unit or temporarily network-disconnected unit. The sending unit identifies the candidate unit and initiates sending the threat data at the IOCC of CS and the system will automatically perform the requisite functions. CS sends notification to receiving unit. When receiving acknowledgement from the receiving unit, CS of sending unit requests DIS to send threat data to receiving unit. The receiving unit responds upon receipt of threat data to the sending unit. This use case should be complete within 3 seconds so the receiving unit has enough time to take follow-up action. Threat data is sent through the web-based C2 environment employing collaborative web tools like Chart or Knowledge Web (KWEB). Flow of events and entry/exit conditions required for this use case are described in Table 6.

Figure 54 describes the sequence of events in the Send Threat Data use case between the unit and the system and where the information is sent. The unit starts sending actions through the CS. CS includes displays, IOCC, and other monitoring services as database and time synchronizers. CS notifies the receiving unit and receives acknowledgement through the EXCOMM. EXCOMM includes data links, satellite, and radio for data transmission and communication. EXCOMM provides the path for communication and data transmission. CS sends threat data request to DIS. DIS keeps track repository, system track, and track kinematics. DIS sends threat data as requested to the receiving unit. EXCOMM sends confirmation of data receipt from the receiving unit to DIS and CS of the sending unit. Figure 55 shows the FFBD of the events.

Use Case Name	Send Threat Data
Participating Actors	Remote or Local Unit
Flow of Events	1. Sending unit initiates sending action at Common Services (CS). 2. CS sends notification to the receiving unit through EXCOMM 3. EXCOMM transmits notification to receiving unit. 4. EXCOMM transmits acknowledgement to CS of sending unit. 5. CS displays acknowledgement to sending unit 6. CS requests DIS to send threat data/track 7. DIS sends threat data/track to the receiving unit through EXCOMM. 8. EXCOMM transmits threat data/track to receiving unit 9. EXCOMM transmits confirmation of receipt to sending unit CS. 10. CS displays confirmation of receipt to sending unit
Entry Condition	Network access is granted. Receiving Unit is identified and threat/track data is available for transmission.
Exit Condition	Send Unit receives a confirmation from EXCOMM.
Quality Requirements	Time from notification to receipt confirmation is no more than 3 seconds.

Table 6. Use Case: Send Threat Data.
Threat data is sent to another unit in the environment.

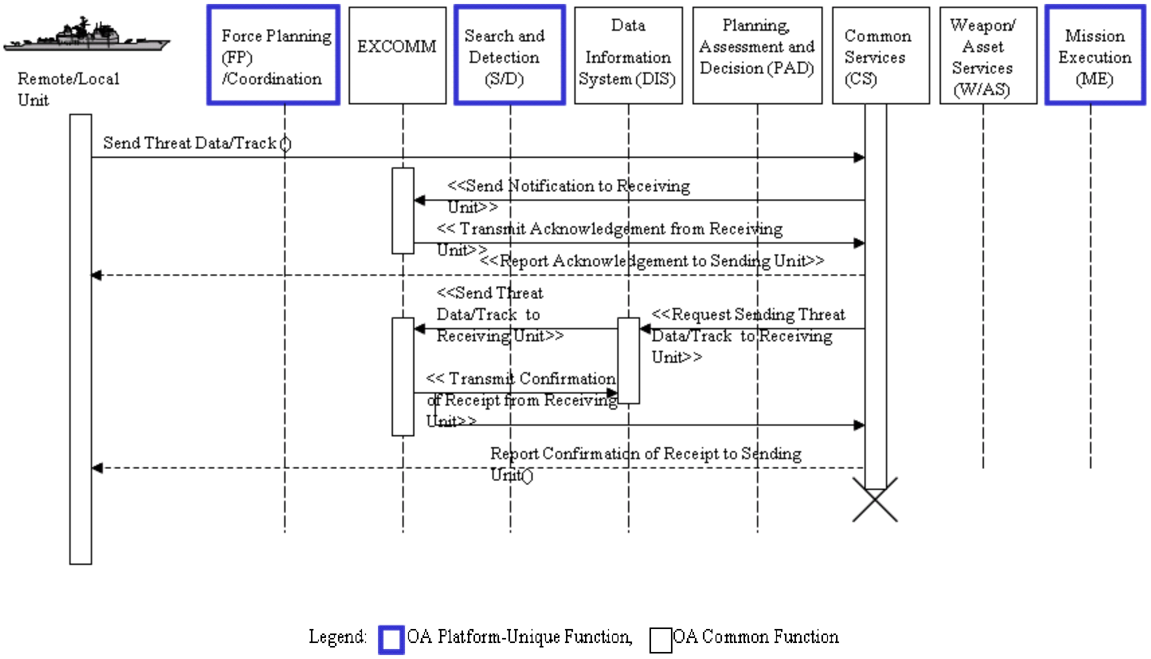

Force Planning (FP) /Coordination
EXCOMM
Search and Detection (S/D)
Data Information System (DIS)
Planning, Assessment and Decision (PAD)
Common Services (CS)
Weapon/ Asset Services (W/AS)
Mission Execution (ME)
Remote/Local Unit
Send Threat Data/Track ()
<<Send Notification to Receiving Unit>>
<< Transmit Acknowledgement from Receiving Unit>>
<<Report Acknowledgement to Sending Unit>>
<<Send Threat Data/Track to Receiving Unit>>
<<Request Sending Threat Data/Track to Receiving Unit>>
<< Transmit Confirmation of Receipt from Receiving Unit>>
Report Confirmation of Receipt to Sending Unit()
Legend: OA Platform-Unique Function, OA Common Function

Figure 54. Send Threat Data sequence diagram.
Timing and sequence of events are reflected.

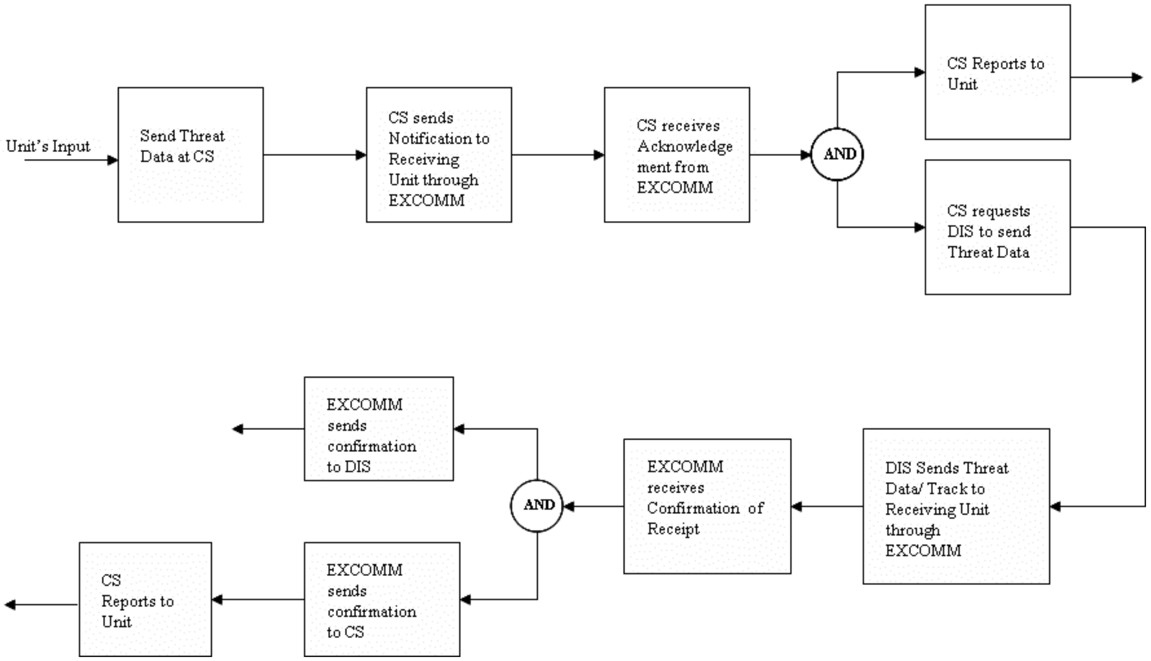

Unit's Input
Send Threat Data at CS
CS sends Notification to Receiving Unit through EXCOMM
CS receives Acknowledgement from EXCOMM
AND
CS Reports to Unit
CS requests DIS to send Threat Data
DIS Sends Threat Data/ Track to Receiving Unit through EXCOMM
EXCOMM receives Confirmation of Receipt
AND
EXCOMM sends confirmation to DIS
EXCOMM sends confirmation to CS
CS Reports to Unit

Figure 55. Flow of events for Send Threat Data use case.

The fourth use case is Receive Threat Data. Either a remote or a local unit can receive threat data from another unit in the environment. This use case is widely used in case a firing unit or a local unit needs threat data from a detecting unit to keep tracking a threat for further action. Although threat data is disseminated through the network for sharing, this use case is still necessary since a unit in action may be a newly joint unit or temporarily network-disconnected unit. The receiving unit, which is active on the network, responds to the notification of the sending unit at the IOCC of CS and the system will automatically perform sequential actions and responses. The threat data from the sending unit is sent to DIS of the receiving unit and the DIS reports to the CS. The receiving unit responds upon receipt of threat data to the sending unit. This use case should be complete within 3 seconds so the receiving unit has enough time to take follow-up action. Threat data is sent through the web-based C2 environment employing collaborative web tools like Chart or KWEB. Flow of events and entry/exit conditions required for this use case are described in Table 7.

Figure 56 describes the sequence of events in the Receive Threat Data use case between the unit and the system and where the information is sent. The EXCOMM sends the notification to the CS of the receiving unit. The receiving unit sends acknowledgement to the sending unit through the CS and the EXCOMM. The EXCOMM sends threat data to the DIS and DIS reports receipt of threat data to the receiving unit via the CS. The receiving unit sends confirmation of data receipt through the CS and EXCOMM to the sending unit. Figure 57 shows the FFBD of the events.

Use Case Name	Receive Threat Data
Participating Actors	Remote or Local Unit
Flow of Events	1. EXCOMM transmits notification to Common Services (CS) of the receiving unit. 2. CS reports notification from the sending unit to receiving unit. 3. Receiving unit sends acknowledgement at CS. 4. CS sends acknowledgement to the sending unit through EXCOMM. 5. EXCOMM transmits acknowledgement to sending unit. 6. EXCOMM transmits threat data/track from sending unit to DIS. 7. DIS reports receipt of threat data to CS. 8. CS reports threat data/track to receiving unit. 9. Receiving unit confirms receipt of threat data/track at CS. 10. CS sends confirmation of receipt to the sending unit through EXCOMM 11. EXCOMM transmits confirmation of receipt to sending unit
Entry Condition	Network access is granted. Receiving unit receives notification through EXCOMM.
Exit Condition	Receiving Unit completes transmission of confirmation.
Quality Requirements	Time from receipt of notification to complete transmission of confirmation is no more than 5 seconds

Table 7. Use Case: Receive Threat Data.
Threat data is received from another unit in the environment.

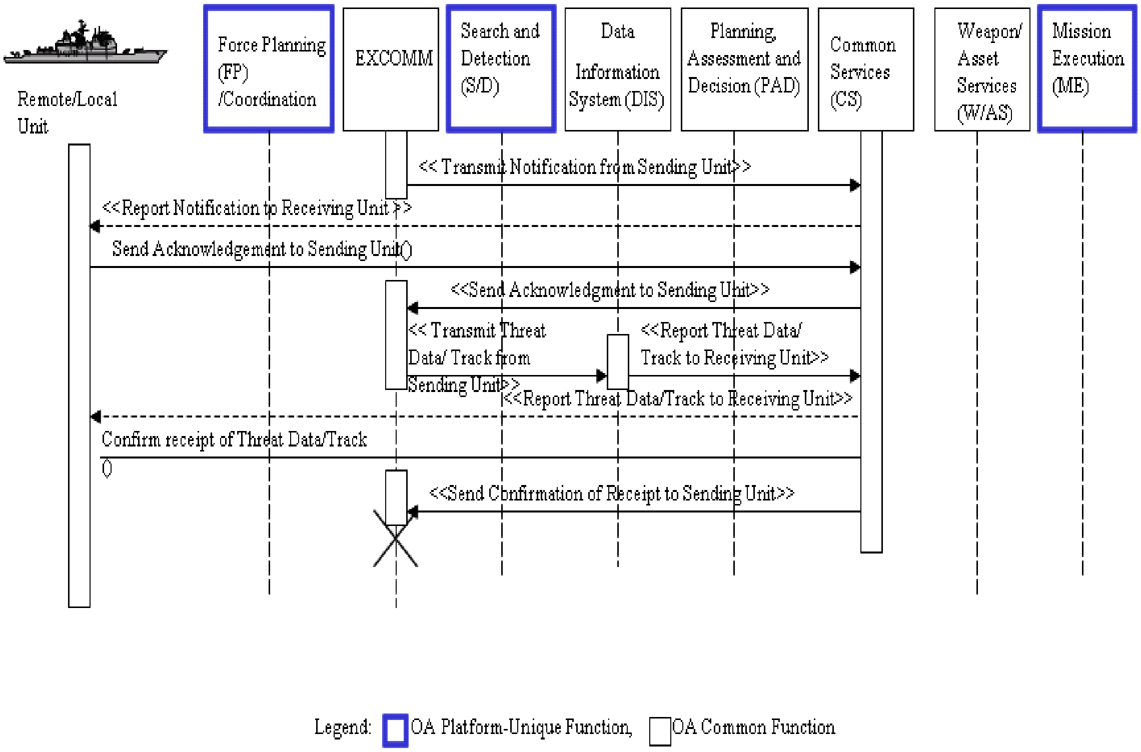

Remote/Local Unit
Force Planning (FP) /Coordination
EXCOMM
Search and Detection (S/D)
Data Information System (DIS)
Planning, Assessment and Decision (PAD)
Common Services (CS)
Weapon/ Asset Services (W/AS)
Mission Execution (ME)
<< Transmit Notification from Sending Unit>>
<<Report Notification to Receiving Unit>>
Send Acknowledgement to Sending Unit()
<<Send Acknowledgment to Sending Unit>>
<< Transmit Threat Data/ Track from Sending Unit>>
<<Report Threat Data/ Track to Receiving Unit>>
<<Report Threat Data/Track to Receiving Unit>>
Confirm receipt of Threat Data/Track ()
<<Send Confirmation of Receipt to Sending Unit>>
Legend: OA Platform-Unique Function, OA Common Function

Figure 56. Receive Threat Data sequence diagram. Timing and sequence of events are reflected.

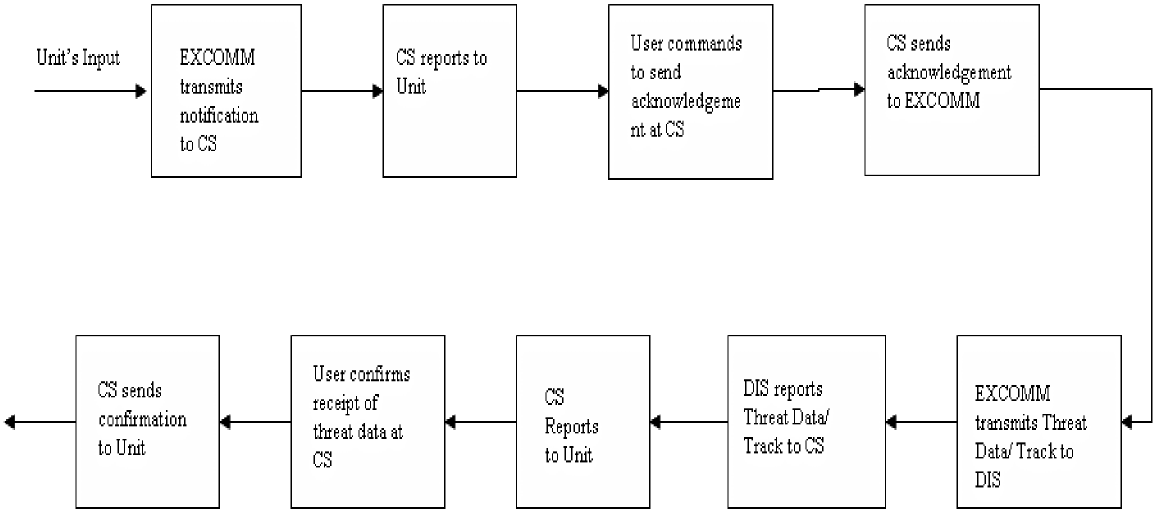

Unit's Input
EXCOMM transmits notification to CS
CS reports to Unit
User commands to send acknowledgement at CS
CS sends acknowledgement to EXCOMM
EXCOMM transmits Threat Data/ Track to DIS
DIS reports Threat Data/ Track to CS
CS Reports to Unit
User confirms receipt of threat data at CS
CS sends confirmation to Unit

Figure 57. Flow of events for Receive Threat Data use case.

Either a remote or local unit can use the Maintain Track use case to monitor and keep track of threats in the environment after a target is detected or following a receipt of threat data. This use case is also used with the Control Engagement use case to provide in-flight guidance. The tracking unit initiates tracking function at the IOCC of CS and the system will automatically perform sequential actions and responses. Sensors of the tracking unit are scheduled to track in accordance with available threat data. Sensor data is sent to DIS for data track updates and displayed at CS. DIS sends threat/track data to PAD for threat assessment and action planning, to W/AS for weapon assessment, and to EXCOMM for dissemination through the network. All units share data to update their situation awareness. Tracking time from tasking sensors to disseminating the updated track should not take more than 10 seconds. Flow of events and entry/exit conditions required for this use case are described in Table 8.

Figure 58 describes the sequence of events in the Maintain Track use case between the unit and the system. The tracking unit starts tracking functions through the IOCC of the CS. CS sends request for tasking sensors to DIS. DIS schedules sensors from S&D to track in accordance with threat data. S&D sends sensor data to DIS for track updates. DIS sends the updated threat/track data to PAD for tactical picture or threat assessment. PAD sends tactical picture and threat assessment to the tracking unit via CS and to the EXCOMM for dissemination. All units in the battlespace are informed and share all threat data through the EXCOMM. The EXCOMM disseminates data to all units on the network for situation awareness and readiness. The tracking unit keeps monitoring the threat/target track until it is no longer in the environment. Figure 59 shows the FFBD of the events.

Use Case Name	Maintain Track
Participating Actors	Remote or Local Unit
Flow of Events	1. Tracking Unit tasks sensors to track in accordance with track data at Common Services (CS). 2. CS sends track data to Data Information Services (DIS) and CS. 3. DIS schedules sensor to track. 4. Search/Detection (S/D) reports sensor data/track to DIS and CS 5. DIS sends target track to Planning Assessment Decision (PAD) for composing a Tactical Picture 6. PAD provides Tactical Picture to tracking unit through CS and disseminates to other units and Force Planning/Coordination (FP/C) through EXCOMM 7. CS displays/reports the Tactical Picture to tracking unit. 8. Tracking unit keeps monitoring target track for follow-up actions.
Entry Condition	Network access is granted and threat / track data is available
Exit Condition	CS displays/reports Tactical Picture to tracking unit.
Quality Requirements	Time from tasking sensors to disseminating Tactical Picture is no more than 10 seconds

Table 8. Use Case: Maintain Track.
Threats in the environment are monitored and tracked after a target is detected.

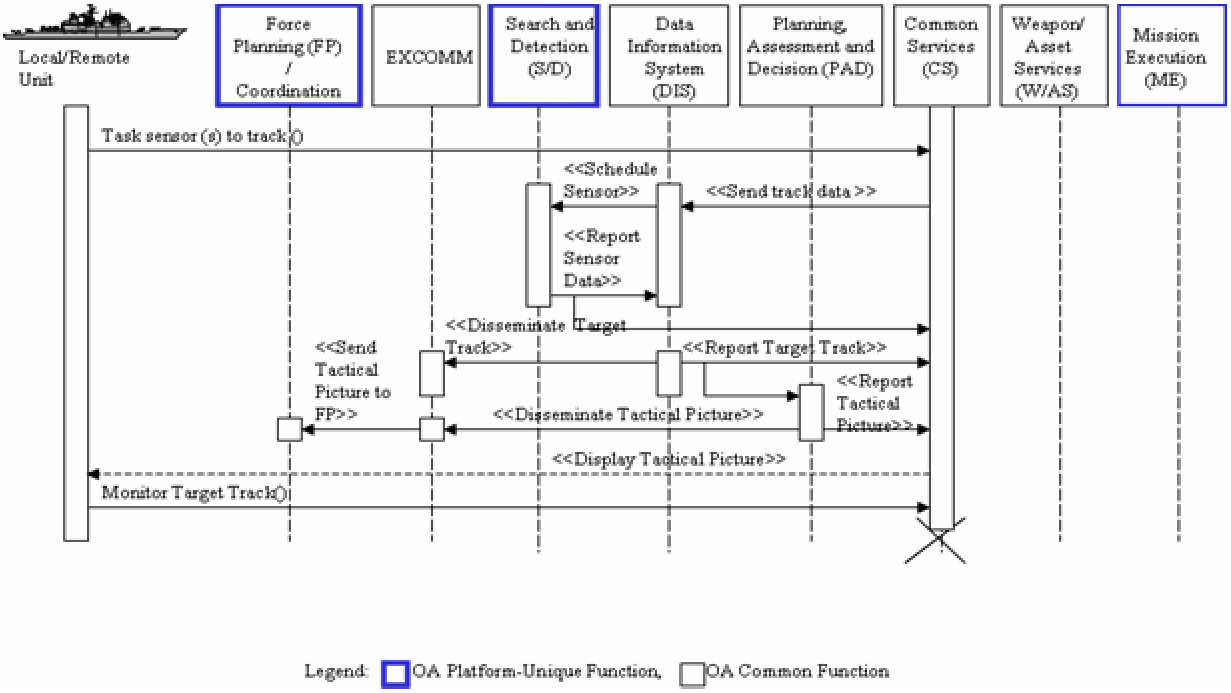

Local/Remote Unit
Force Planning (FP) / Coordination
EXCOMM
Search and Detection (S/D)
Data Information System (DIS)
Planning, Assessment and Decision (PAD)
Common Services (CS)
Weapon/ Asset Services (W/AS)
Mission Execution (ME)
Task sensor (s) to track()
<<Schedule Sensor>>
<<Send track data >>
<<Report Sensor Data>>
<<Disseminate Target Track>>
<<Send Tactical Picture to FP>>
<<Report Target Track>>
<<Report Tactical Picture>>
<<Disseminate Tactical Picture>>
<<Display Tactical Picture>>
Monitor Target Track()
Legend: OA Platform-Unique Function, OA Common Function

Figure 58. Maintain Track sequence diagram.
Timing and sequence of events are reflected.

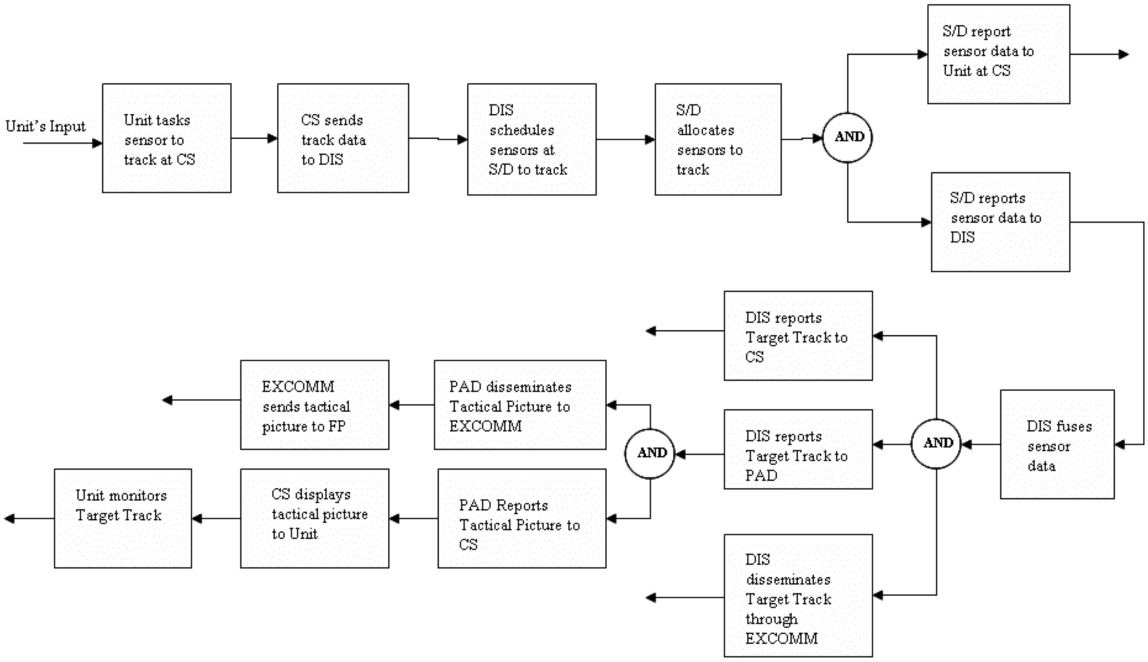

Unit's Input
Unit tasks sensor to track at CS
CS sends track data to DIS
DIS schedules sensors at S/D to track
S/D allocates sensors to track
AND
S/D report sensor data to Unit at CS
S/D reports sensor data to DIS
DIS fuses sensor data
AND
DIS reports Target Track to CS
DIS reports Target Track to PAD
DIS disseminates Target Track through EXCOMM
AND
PAD disseminates Tactical Picture to EXCOMM
PAD Reports Tactical Picture to CS
EXCOMM sends tactical picture to FP
CS displays tactical picture to Unit
Unit monitors Target Track

Figure 59. Flow of events for Maintain Track use case.

Either a remote or a local unit can use the Determine the Preferred Shooter use case to take advantage of the most effective firing unit against the existing threat determined by target geometry and operational capability. This use case is used after a target is detected or following a receipt of threat data. The requesting unit initiates a determination request at the IOCC of the CS and the system will automatically perform sequential actions and responses. The determination request is sent to the PAD for mission assessment. The PAD assesses the capability of the requesting unit and accesses sharing data for engagement geometry and capability of other units through the EXCOMM. After gathering data and using decision-making aids, the PAD determines the best shooter for the current threat and reports to the requesting unit. The PAD also multicasts the Preferred Shooter to all units through the EXCOMM. All units share data to update their situation awareness. Determination time from receiving the request to disseminating the determination should not take more than 5 seconds. Flow of events and entry/exit conditions required for this use case are described in Table 9.

Figure 60 describes the sequence of events in the Determine the Preferred Shooter use case between the unit and the system. The requesting unit initiates the request at the IOCC of the CS. The CS sends the request of engagement determination to the PAD. The PAD acquires the capability and geometry of the requesting unit in the PAD and other units through the EXCOMM. The EXCOMM responds with capability of other units to the PAD and the PAD determines the best shooter for engaging the current threat. The PAD reports the preferred shooter to the requesting unit via CS and disseminates to all units through the EXCOMM. All units in the battlespace are informed of the best shooter to engage the current threat. Figure 61 shows the FFBD of the events.

Use Case Name	Determine the Preferred Shooter
Participating Actors	Remote or Local Unit.
Flow of Events	1. Requesting Unit initiates a request to determine the best shooter in current situation through the Common Services (CS). 2. CS sends a request to the PAD for a determination. 3. PAD assesses the Engagement Geometry and Ability of the Requesting Unit. 4. PAD acquires the Engagement Geometry and Ability of other units through sharing data. 5. PAD determines the best shooter of the current situation. 6. PAD reports the best shooter to requesting unit through CS and disseminates to other units and FP through EXCOMM 7. CS reports the best shooter to requesting unit.
Entry Condition	Network access is granted and threat data is available.
Exit Condition	The dissemination of the determination.
Quality Requirements	Time from receiving the request to completing dissemination of the preferred shooter is no more than 5 seconds.

Table 9. Use Case: Determine the Preferred Shooter.
The most effective firing unit is used against an existing threat based on target geometry and operational capability.

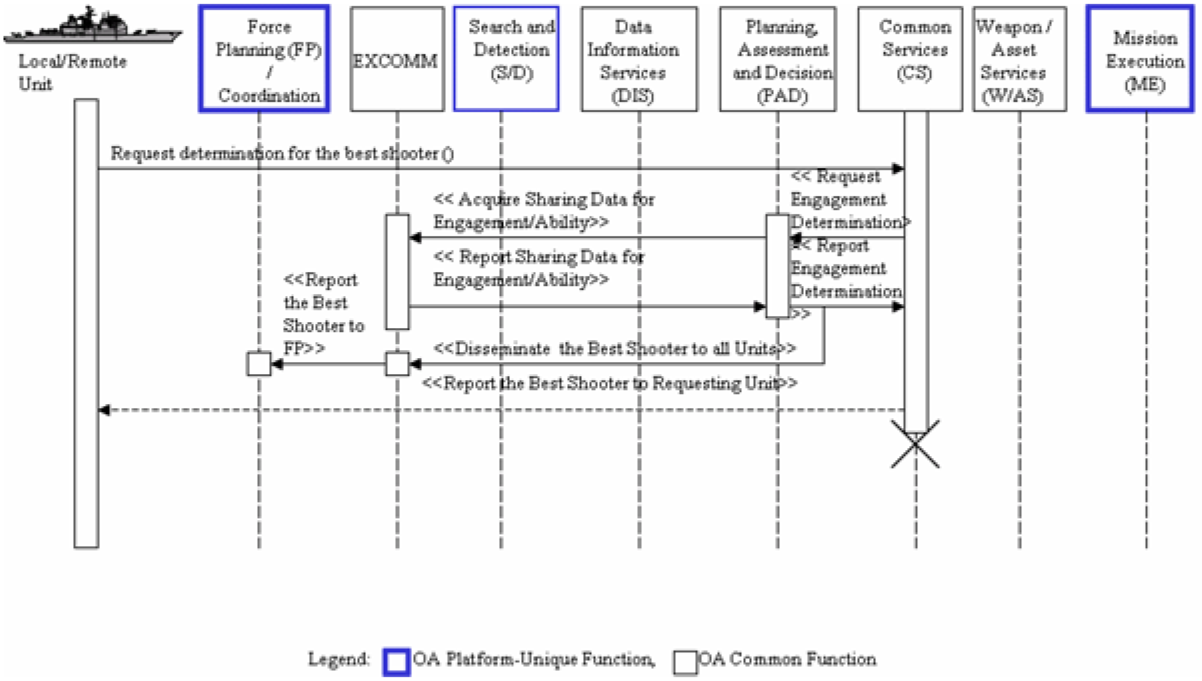

Local/Remote Unit
Force Planning (FP) / Coordination
EXCOMM
Search and Detection (S/D)
Data Information Services (DIS)
Planning, Assessment and Decision (PAD)
Common Services (CS)
Weapon / Asset Services (W/AS)
Mission Execution (ME)
Request determination for the best shooter ()
<< Request Engagement Determination>
<< Acquire Sharing Data for Engagement/Ability>>
<< Report Sharing Data for Engagement/Ability>>
<< Report Engagement Determination >>
<<Report the Best Shooter to FP>>
<<Disseminate the Best Shooter to all Units>>
<<Report the Best Shooter to Requesting Unit>>
Legend: OA Platform-Unique Function, OA Common Function

Figure 60. Determine the Preferred Shooter.
Timing and sequence of events are reflected.

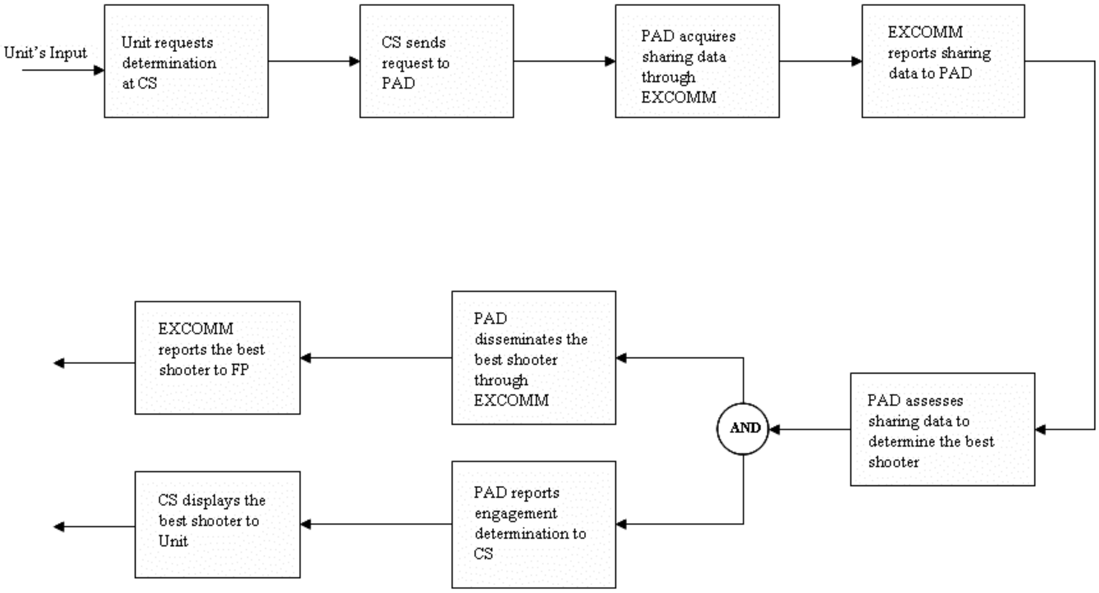

Unit's Input
Unit requests determination at CS
CS sends request to PAD
PAD acquires sharing data through EXCOMM
EXCOMM reports sharing data to PAD
PAD assesses sharing data to determine the best shooter
AND
PAD disseminates the best shooter through EXCOMM
EXCOMM reports the best shooter to FP
PAD reports engagement determination to CS
CS displays the best shooter to Unit

Figure 61. Flow of events for the Determine the Preferred Shooter use case.

A remote or a local unit can use the Make Firing Decision use case to determine unit suitability for firing. This use case is used to determine if the firing should be performed by the remote or a local unit for a certain threat. The requesting unit initiates a determination request at the IOCC of the CS and the system will automatically perform sequential actions and responses. The determination request is sent to the PAD for mission assessment. The PAD queries W/AS for current weapon and accesses the tactical picture in the PAD. After gathering data and using decision-making aids, the PAD determines the suitability for firing against the current threat and reports to the requesting unit. The PAD also disseminates the determination to other units in the network. All units share data to update their situation awareness. Determination time from receiving the request to disseminating the determination should not take more than 3 seconds. Flow of events and entry/exit conditions required for this use case are described in Table 10.

Figure 62 describes the sequence of events in the Make Firing Decision use case between the unit and the system. The requesting unit initiates the request at the IOCC of the CS. The CS sends the request to assess suitability to the PAD. The PAD assesses the threat and capability of the requesting units through the W/AS. The W/AS reports the updated capability to the PAD. The PAD determines and reports the suitability to engage the current threat to the requesting unit. The PAD also disseminates to all units through the EXCOMM for situation awareness. Figure 63 shows the FFBD of the events.

Use Case Name	Making Firing Decision
Participating Actors	Remote or Local Unit
Flow of Events	1. Unit determines suitability for firing through Common Services (CS). 2. CS requests Planning Assessment Decision (PAD) for assessing suitability. 3. PAD assesses capability through Weapon/Asset Services (W/AS). 4. PAD assesses capability against threat and reports to CS and disseminates to other units in the network. 5. CS reports determination of suitability for firing to requesting unit
Entry Condition	Interface between OA common function and platform-unique function is available and network access is granted. Threat data is available.
Exit Condition	Suitability of firing is determined.
Quality Requirements	Time from inquiring determination to receiving determination is no more than 3 seconds. .

Table 10. Use Case: Make Firing Decision.
A unit's suitability for firing is determined.

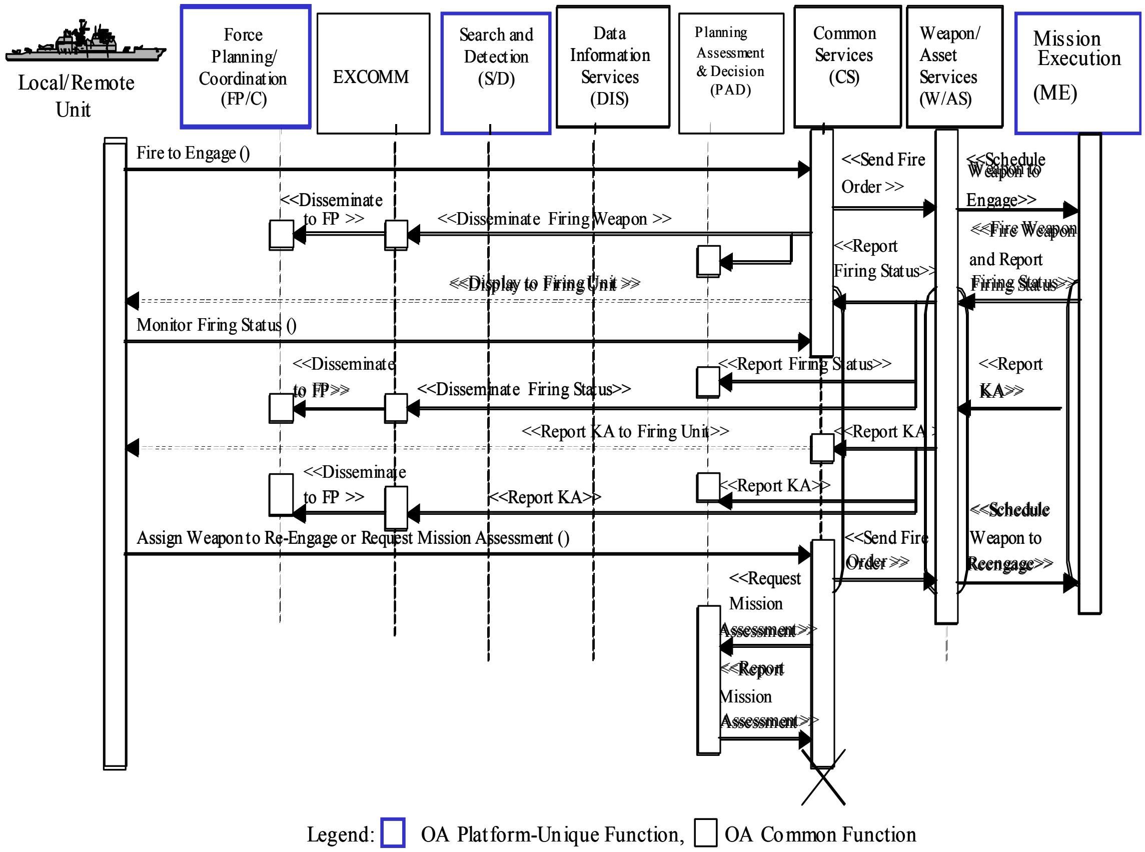

Local/Remote Unit
Force Planning/ Coordination (FP/C)
EXCOMM
Search and Detection (S/D)
Data Information Services (DIS)
Planning Assessment & Decision (PAD)
Common Services (CS)
Weapon/ Asset Services (W/AS)
Mission Execution (ME)
Fire to Engage ()
<<Send Fire Order >>
<<Schedule Weapon to Engage>>
<<Disseminate to FP >>
<<Disseminate Firing Weapon >>
<<Fire Weapon and Report Firing Status>>
<<Report Firing Status>>
<<Display to Firing Unit >>
Monitor Firing Status ()
<<Disseminate to FP>>
<<Disseminate Firing Status>>
<<Report Firing Status>>
<<Report KA>>
<<Report KA to Firing Unit>>
<<Report KA >>
<<Disseminate to FP >>
<<Report KA>>
<<Report KA>>
Assign Weapon to Re-Engage or Request Mission Assessment ()
<<Send Fire Order>>
<<Schedule Weapon to Reengage>>
<<Request Mission Assessment>>
<<Report Mission Assessment>>
Legend: □ OA Platform-Unique Function, □ OA Common Function

Figure 62. Make Firing Decision sequence diagram. Timing and sequence of events are reflected.

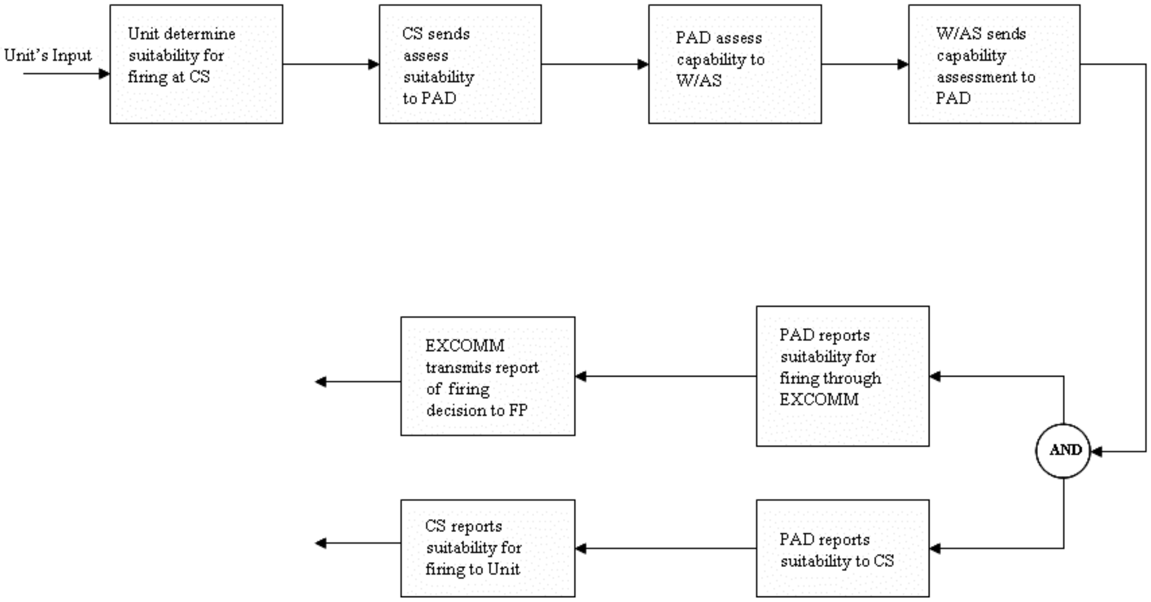

Unit's Input
Unit determine suitability for firing at CS
CS sends assess suitability to PAD
PAD assess capability to W/AS
W/AS sends capability assessment to PAD
AND
PAD reports suitability for firing through EXCOMM
EXCOMM transmits report of firing decision to FP
PAD reports suitability to CS
CS reports suitability for firing to Unit

Figure 63. Flow of events for Make Firing Decision use case.

Either remote or local units can use the Fire Weapon use case to intercept the threat. This use case can be used after the Determine the Preferred Shooter or Make Firing Decision use case. The firing unit sends fire orders at the IOCC of the CS and the system will automatically perform sequential actions and responses. The CS sends the fire order to the W/AS and disseminates the fire order to all units. The W/AS functions as a weapon control system that keeps weapon inventory and schedules weapon for engagement as ordered. The W/AS receives the fire order from the CS and schedules weapon at the Mission Execution (ME) for engagement. The ME includes weapon systems that fire or launch guns or missiles to engage threat. The ME fires interceptor as ordered and reports firing status to the weapon control system at W/AS and displays at the CS. The W/AS sends report of firing to the PAD and disseminates to all units for current situation awareness. The ME also provides in-flight guidance to control the missile engaging the threat and performs Kill Assessment (KA). The firing unit evaluates the KA report from the ME for re-engagement. This process can take place in a loop in accordance with ship doctrine Shoot-Look-Shoot until the threat is killed or leaked. The KA report is disseminated to all units in the battlespace to maintain the situation awareness. Time from sending Fire Order to W/AS until receiving firing status from the ME should be no more than 3 seconds. Flow of events and entry/exit conditions required for this use case are described in Table 11.

Figure 64 describes the sequence of events in the Fire Weapon use case between the unit and the system. The firing unit initiates Fire Order at the IOCC of the CS. The CS sends the Fire Order to the W/AS. The W/AS schedules weapon for firing at the ME. The ME fires weapon as scheduled and reports the firing status to the W/AS and the CS. The firing unit monitors KA reports from the ME to decide for reengagement. The W/AS also disseminates to all units through the EXCOMM for situation awareness. Figure 65 shows the FFBD of the events.

Use Case Name	Fire Weapon(s)
Participating Actors	Remote or Local Unit
Flow of Events	1. Firing Unit executes firing orders by sending Fire Order at Common Services (CS). 2. CS sends request to fire weapon to Weapon / Asset Services (W/AS) and disseminates firing order to other units and Force Planning (FP)/Coordination through EXCOMM 3. W/AS schedules to fire weapon at Mission Execution (ME). 4. ME fires weapon(s) and reports firing status to W/AS. 5. W/AS reports firing status to CS, Planning Assessment Decision (PAD), and disseminates firing status through EXCOMM 6. Firing Unit monitors firing status. 7. ME reports KA to W/AS. 8. W/AS reports KA to firing unit and disseminates through EXCOMM. 9. Firing Unit evaluates KA and assigns weapon to reengage if there is a No Kill and time allows for reengagement (Shoot-Look-Shoot).
Entry Condition	Network access is granted. Firing decision is made.
Exit Condition	A Kill or Leak is reported and disseminated.
Quality Requirements	Time from sending Fire Order to W/AS until receiving firing status from the ME is no more than 3 seconds.

Table 11. Use Case: Fire Weapon.
Either a remote or a local unit can engage an incoming threat.

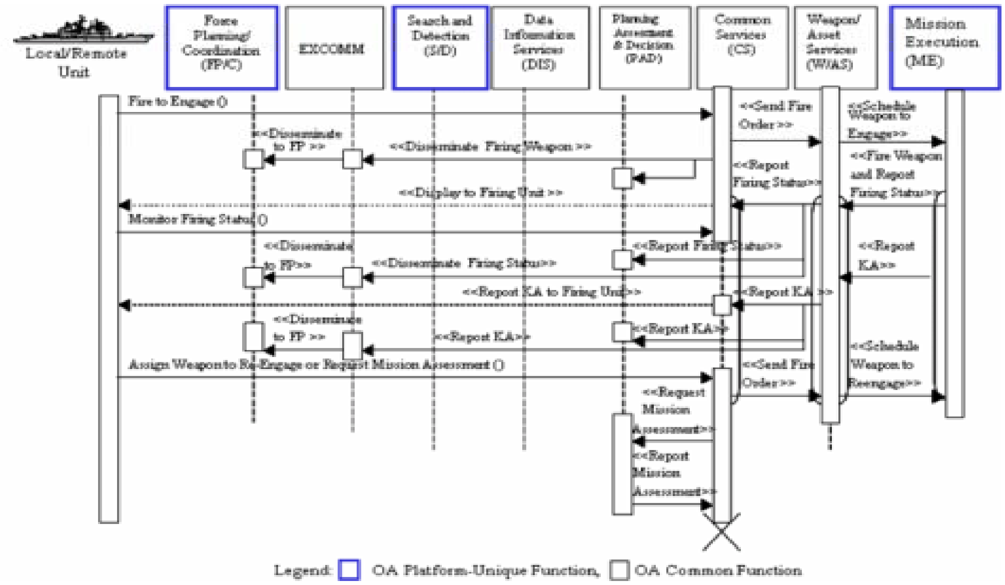

Local/Remote Unit
Force Planning/ Coordination (FP/C)
EXCOMM
Search and Detection (S/D)
Data Information Services (DIS)
Planning Assessment & Decision (PAD)
Common Services (CS)
Weapon/ Asset Services (W/AS)
Mission Execution (ME)
Fire to Engage ()
<<Send Fire Order>>
<<Schedule Weapon to Engage>>
<<Disseminate to FP >>
<<Disseminate Firing Weapon >>
<<Report Firing Status>>
<<Fire Weapon and Report Firing Status>>
<<Display to Firing Unit >>
Monitor Firing Status ()
<<Disseminate to FP>>
<<Disseminate Firing Status>>
<<Report Firing Status>>
<<Report KA>>
<<Report KA to Firing Unit>>
<<Report KA >>
<<Disseminate to FP >>
<<Report KA>>
<<Report KA>>
<<Schedule Weapon to Reengage>>
Assign Weapon to Re-Engage or Request Mission Assessment ()
<<Send Fire Order>>
<<Request Mission Assessment>>
<<Report Mission Assessment>>
Legend: OA Platform-Unique Function, OA Common Function

Figure 64. Fire Weapon sequence diagram.
Timing and sequence of events are reflected.

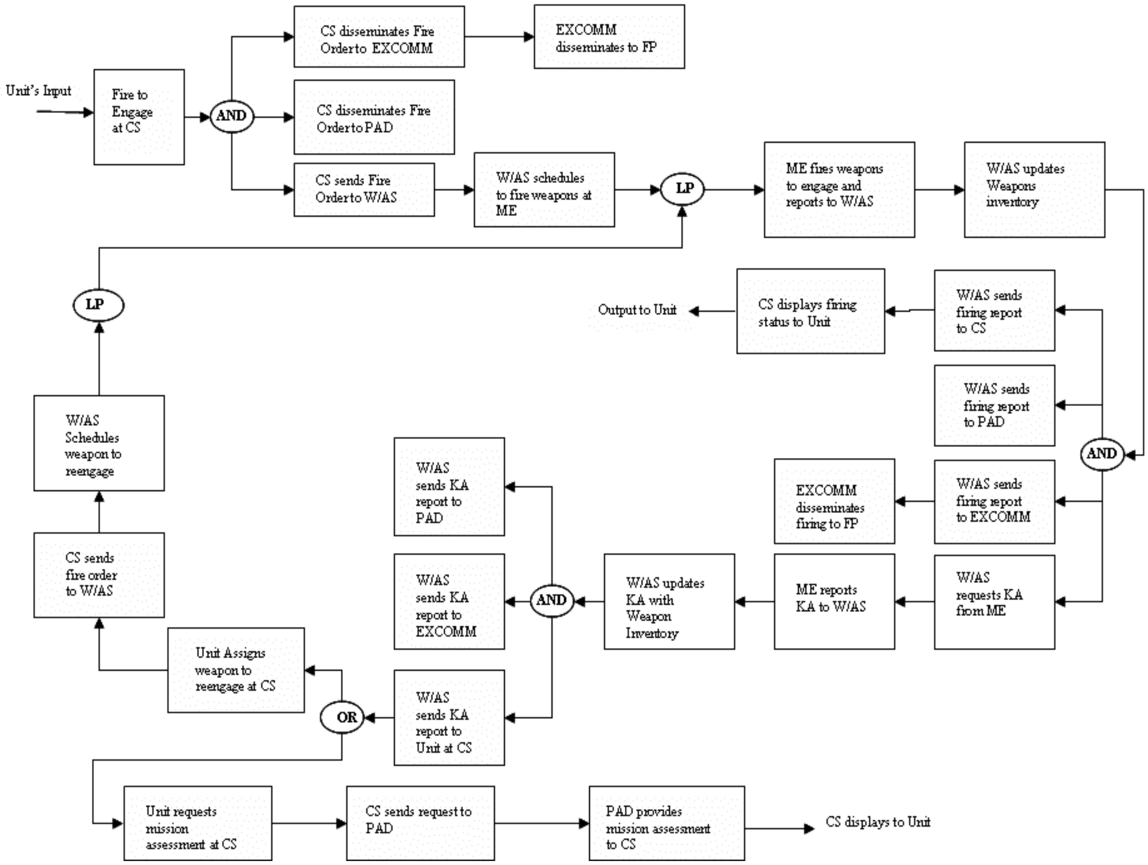

Unit's Input
Fire to Engage at CS
AND
CS disseminates Fire Order to EXCOMM
EXCOMM disseminates to FP
CS disseminates Fire Order to PAD
CS sends Fire Order to W/AS
W/AS schedules to fire weapons at ME
LP
ME fires weapons to engage and reports to W/AS
W/AS updates Weapons inventory
LP
Output to Unit
CS displays firing status to Unit
W/AS sends firing report to CS
W/AS sends firing report to PAD
AND
W/AS Schedules weapon to reengage
W/AS sends KA report to PAD
EXCOMM disseminates firing to FP
W/AS sends firing report to EXCOMM
CS sends fire order to W/AS
W/AS sends KA report to EXCOMM
AND
W/AS updates KA with Weapon Inventory
ME reports KA to W/AS
W/AS requests KA from ME
Unit Assigns weapon to reengage at CS
OR
W/AS sends KA report to Unit at CS
Unit requests mission assessment at CS
CS sends request to PAD
PAD provides mission assessment to CS
CS displays to Unit

Figure 65. Flow of events for Fire Weapon use case.

A local or firing unit can use the Pass Engagement Control use case to pass engagement control to a remote unit. This use case is used in case a firing unit is firing to engage a threat beyond its own sensor range or defending a raid. Although threat data is disseminated through the network for sharing, this use case is still needed because a unit in action may be a newly joint unit or temporarily network-disconnected unit. The sending unit identifies the candidate unit and initiates sending the engagement control data at the IOCC of CS and the system will automatically perform sequential actions and responses. CS sends notification to receiving unit. Once acknowledgement is received from the receiving unit, CS of sending unit requests DIS to send engagement control data to the receiving unit. The receiving unit responds upon receipt of engagement control data to the sending unit. This use case should be complete within 3 seconds so the receiving unit has enough time to take follow-up action. Flow of events and entry/exit conditions required for this use case are described in Table 12.

Figure 66 describes the sequence of events in the Pass Engagement Control use case between the firing unit and the system and where the information is sent. The firing unit starts sending action through the CS. CS notifies the receiving unit and receives through the EXCOMM. When EXCOMM sends the acknowledgement to the CS of the sending unit, the CS requests the DIS to send the engagement control to the receiving unit. This use case concludes when the EXCOMM sends confirmation of data receipt from the receiving unit to DIS and CS of the sending unit. Figure 67 shows the FFBD of the events.

Use Case Name	Pass Engagement Control
Participating Actors	Remote or Local Unit
Flow of Events	1. Sending unit initiates sending engagement control at Common Services (CS). 2. CS sends notification to the receiving unit through EXCOMM. 3. EXCOMM transmits notification to receiving unit. 4. EXCOMM transmits acknowledgement to CS of sending unit 5. CS displays acknowledgement to sending unit. 6. Sending unit sends engagement control data at CS 7. CS requests the DIS to send engagement control data to the receiving unit through EXCOMM 8. EXCOMM transmits engagement control data to receiving unit 9. EXCOMM transmits confirmation of receipt to sending unit CS 10. CS displays confirmation of receipt to sending unit.
Entry Condition	Network access is granted. Firing status is a success launch.
Exit Condition	A confirmation is received from the EXCOMM.
Quality Requirements	Time from sending notification to receiving confirmation is no more than 5 seconds.

Table 12. Use Case: Pass Engagement Control.
Used in case a firing unit is firing to engage a threat beyond its own sensor range or defending a raid.

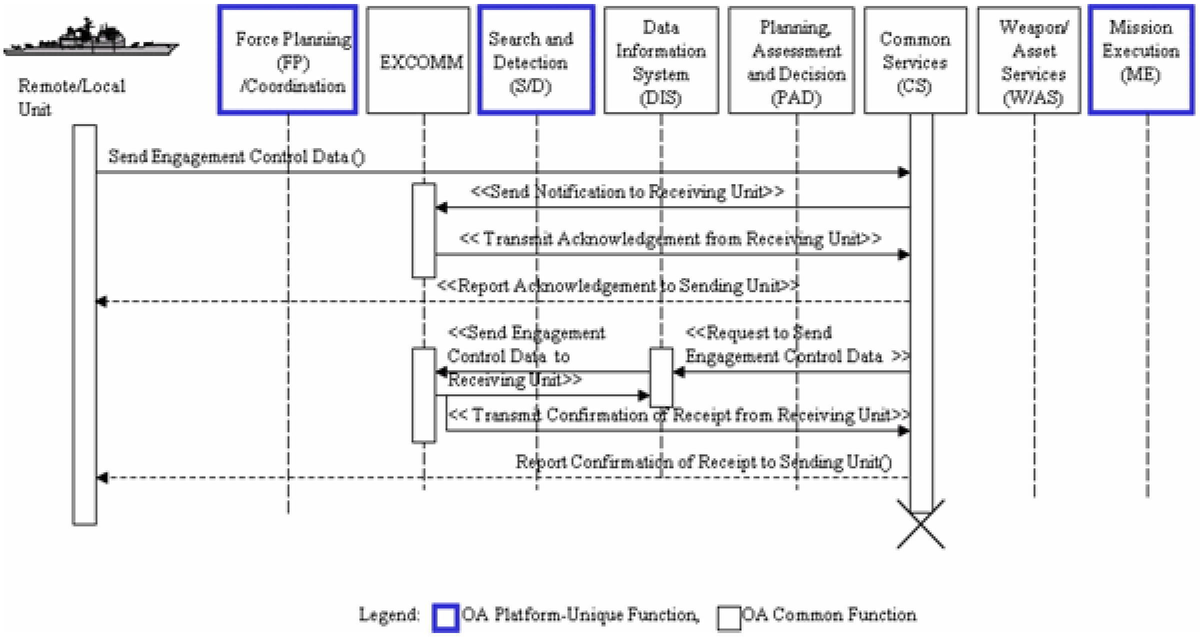

Remote/Local Unit
Force Planning (FP) /Coordination
EXCOMM
Search and Detection (S/D)
Data Information System (DIS)
Planning, Assessment and Decision (PAD)
Common Services (CS)
Weapon/ Asset Services (W/AS)
Mission Execution (ME)
Send Engagement Control Data ()
<<Send Notification to Receiving Unit>>
<< Transmit Acknowledgement from Receiving Unit>>
<<Report Acknowledgement to Sending Unit>>
<<Send Engagement Control Data to Receiving Unit>>
<<Request to Send Engagement Control Data >>
<< Transmit Confirmation of Receipt from Receiving Unit>>
Report Confirmation of Receipt to Sending Unit()
Legend: OA Platform-Unique Function, OA Common Function

Figure 66. Pass Engagement Control sequence diagram.
Sequence and timing of events for Pass Engagement Control are reflected.

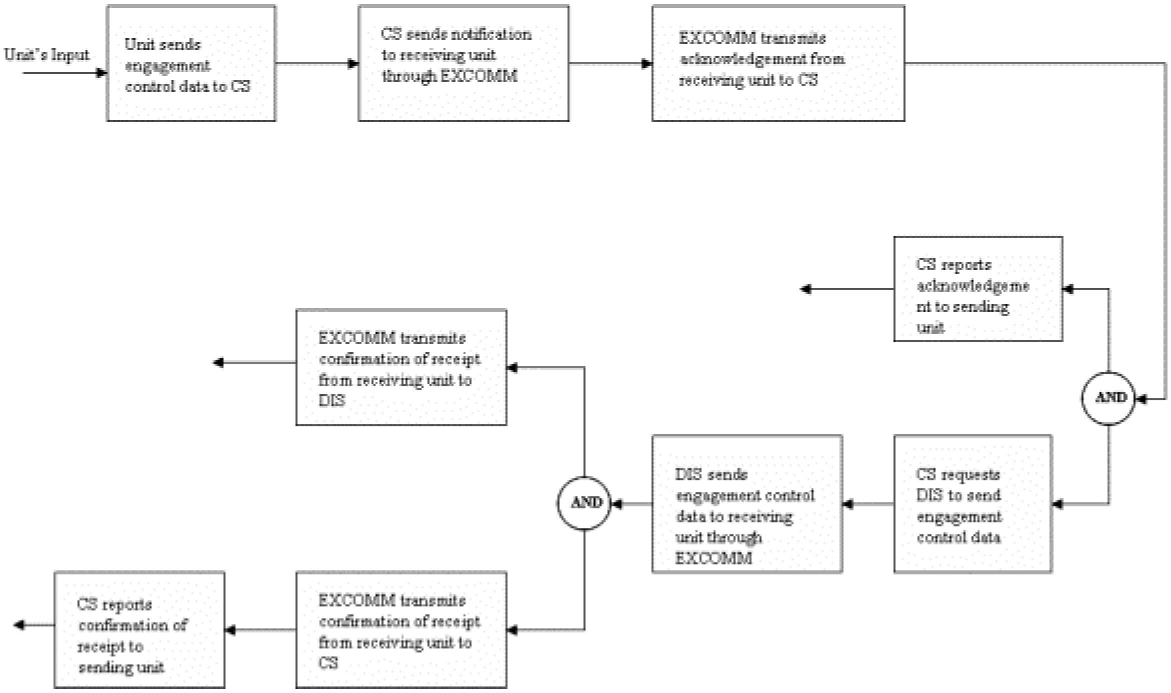

Unit's Input
Unit sends engagement control data to CS
CS sends notification to receiving unit through EXCOMM
EXCOMM transmits acknowledgement from receiving unit to CS
CS reports acknowledgement to sending unit
AND
EXCOMM transmits confirmation of receipt from receiving unit to DIS
AND
DIS sends engagement control data to receiving unit through EXCOMM
CS requests DIS to send engagement control data
CS reports confirmation of receipt to sending unit
EXCOMM transmits confirmation of receipt from receiving unit to CS

Figure 67. Flow of events for Pass Engagement Control use case.

Either a remote or a local unit can use the Receive Engagement Control use case to receive engagement control from a firing unit. This use case is used in case a remote unit has no weapon to engage the threat, which is beyond the sensor range of the firing unit, or a local unit can help a firing unit to engage multiple threats. Although threat data is disseminated through network for sharing data, this use case is still needed because a unit in action may be a newly joint unit or temporarily network-disconnected unit. The receiving unit, which is active on the network, responds to the notification of the sending unit at the IOCC of CS and the system will automatically perform sequential actions and responses. The engagement control data from the sending unit is sent to DIS of the receiving unit and the DIS reports to the CS. The receiving unit responds upon receipt of engagement control data to the sending unit. This use case should be complete within 3 seconds to allow enough time for the receiving unit to take follow-up actions. Flow of events and entry/exit conditions required for this use case are described in Table 13.

Figure 68 describes the sequence of events in the Receive Engagement Control use case between the unit and the system and where the information is sent. The EXCOMM sends the notification to the CS of the receiving unit. The receiving unit sends acknowledgement to the sending unit through the CS and the EXCOMM. The EXCOMM sends engagement control data to the DIS of the receiving unit and DIS reports receipt of engagement control data to the receiving unit via the CS. The receiving unit sends confirmation of data receipt through the CS and EXCOMM to the sending unit. Figure 69 shows the FFBD of the events.

Use Case Name	Receive Engagement Control
Participating Actors	Remote or Local Unit
Flow of Events	1. EXCOMM transmits notification to Common Services (CS) of receiving unit. 2. CS reports notification from the sending unit to receiving unit. 3. Receiving unit sends acknowledgement at CS. 4. CS sends acknowledgement to the sending unit through EXCOMM. 5. EXCOMM transmits acknowledgement to sending unit 6. EXCOMM transmits engagement control data from sending unit to DIS of the receiving unit. 7. CS reports engagement control data to receiving unit 8. Receiving unit confirms receipt of engagement control data at CS. 9. CS sends confirmation of receipt to the sending unit through EXCOMM 10. EXCOMM transmits confirmation of receipt to sending unit.
Entry Condition	Network access is granted. A notification is received through EXCOMM
Exit Condition	A confirmation of receipt is sent to the sending unit.
Quality Requirements	Time from receiving notification to sending confirmation is no more than 5 seconds.

Table 13. Use Case: Receive Engagement Control.
Used in case a remote unit has no weapon to engage a threat that is beyond the sensor range of the firing unit or a local unit.

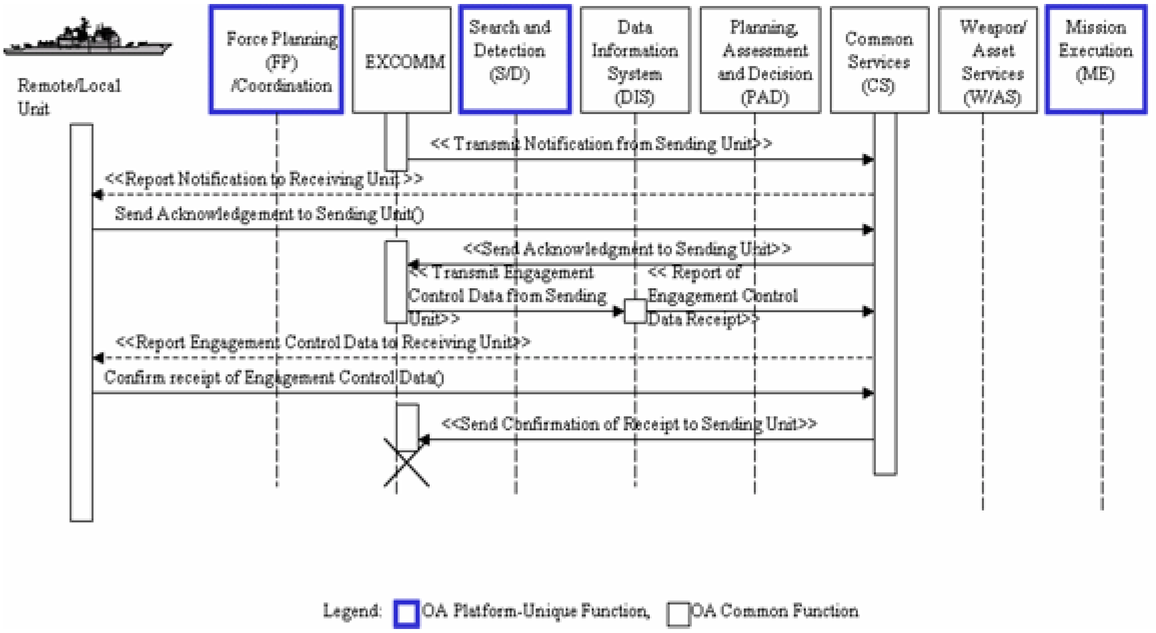

Remote/Local Unit
Force Planning (FP) /Coordination
EXCOMM
Search and Detection (S/D)
Data Information System (DIS)
Planning, Assessment and Decision (PAD)
Common Services (CS)
Weapon/ Asset Services (W/AS)
Mission Execution (ME)
<< Transmit Notification from Sending Unit>>
<<Report Notification to Receiving Unit>>
Send Acknowledgement to Sending Unit()
<<Send Acknowledgment to Sending Unit>>
<< Transmit Engagement Control Data from Sending Unit>>
<< Report of Engagement Control Data Receipt>>
<<Report Engagement Control Data to Receiving Unit>>
Confirm receipt of Engagement Control Data()
<<Send Confirmation of Receipt to Sending Unit>>
Legend: OA Platform-Unique Function, OA Common Function

Figure 68. Receive Engagement Control sequence diagram. Timing and sequence of events are reflected.

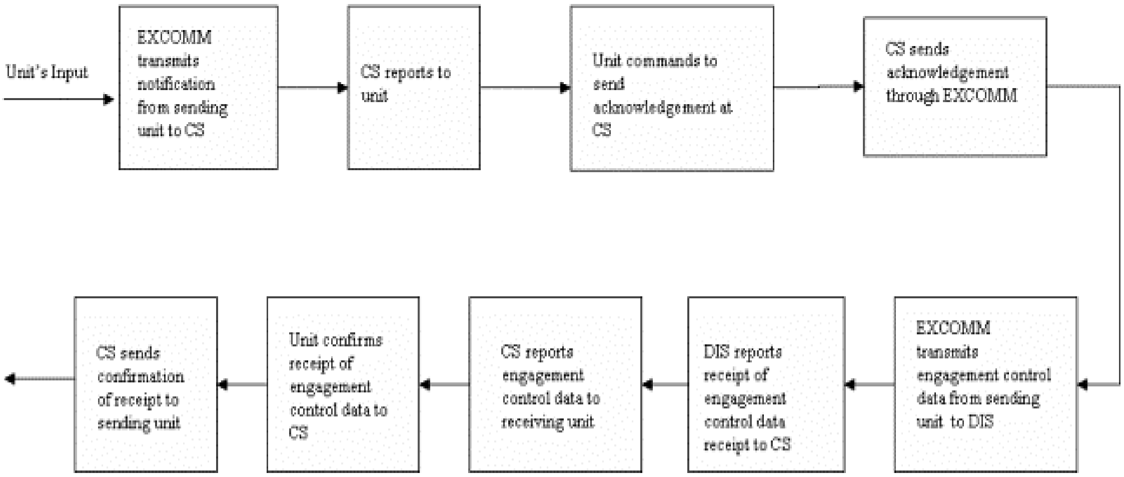

Unit's Input
EXCOMM transmits notification from sending unit to CS
CS reports to unit
Unit commands to send acknowledgement at CS
CS sends acknowledgement through EXCOMM
EXCOMM transmits engagement control data from sending unit to DIS
DIS reports receipt of engagement control data receipt to CS
CS reports engagement control data to receiving unit
Unit confirms receipt of engagement control data to CS
CS sends confirmation of receipt to sending unit

Figure 69. Flow of events for Receive Engagement Control use case.

Either a remote or a local unit can use the Control Engagement use case to provide guidance to the missile for engaging the threat. This use case is used after the missile is fired and after the Fire Weapon or Receive Engagement Control use case. The controlling unit initiates the action to provide the guidance at the IOCC of the CS and the system will automatically perform sequential actions and responses. The CS requests track/threat data from the DIS. The DIS provides the track/threat data to the W/AS and the W/AS provides guidance to the ME. The ME uses this guidance data for weapon control to engage threat and reports KA to the W/AS and CS. Time for uplink to provide in-flight guidance is no more than 5 seconds all of which are described in Table 14.

Figure 70 describes the sequence of events in the Control Engagement use case between the unit and the system. The engagement is controlled by using track/threat data obtained from the Maintain Track use case to provide guidance to the in-flight interceptor. The controlling unit initiates the control action at the IOCC of the CS to provide guidance control. The CS sends the request for track/threat data to the DIS. The DIS sends threat/track data to W/AS and to CS for display, which in turn uses the threat/track data to provide guidance to the ME. The ME uses the guidance to control the weapon for threat engagement. Lastly, the ME sends the KA report to the W/AS and to the CS. The W/AS also disseminates to all units through the EXCOMM for SA.

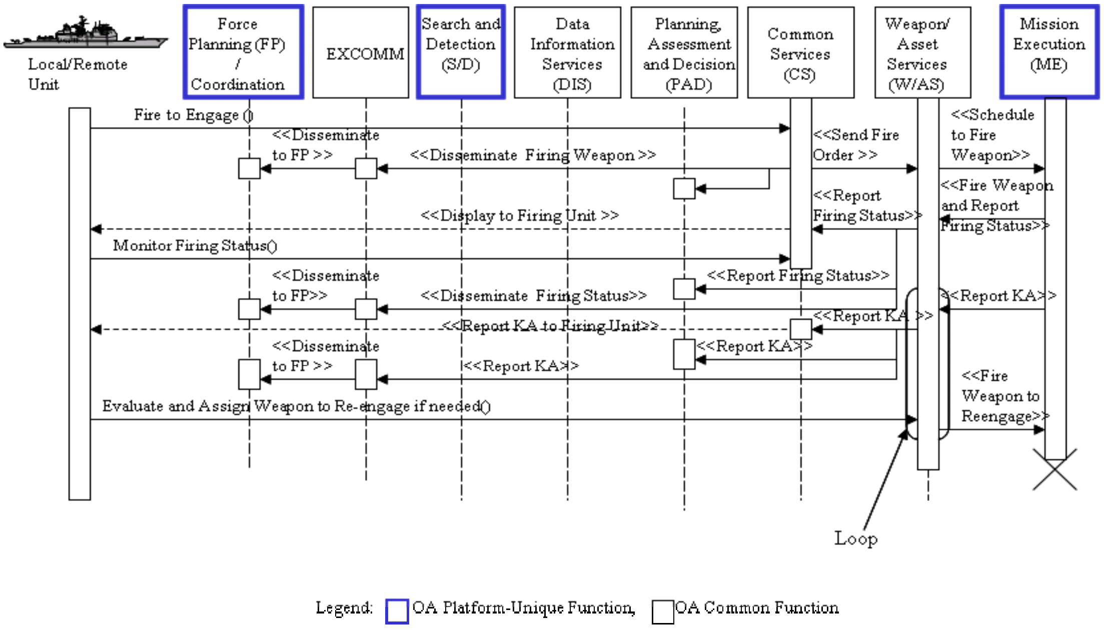

Local/Remote Unit
Force Planning (FP) / Coordination
EXCOMM
Search and Detection (S/D)
Data Information Services (DIS)
Planning, Assessment and Decision (PAD)
Common Services (CS)
Weapon/ Asset Services (W/AS)
Mission Execution (ME)
Fire to Engage ()
<<Disseminate to FP >>
<<Disseminate Firing Weapon >>
<<Send Fire Order >>
<<Schedule to Fire Weapon>>
<<Fire Weapon and Report Firing Status>>
<<Report Firing Status>>
<<Display to Firing Unit >>
Monitor Firing Status()
<<Report Firing Status>>
<<Disseminate to FP>>
<<Disseminate Firing Status>>
<<Report KA>>
<<Report KA >>
<<Report KA to Firing Unit>>
<<Disseminate to FP >>
<<Report KA>>
<<Report KA>>
Evaluate and Assign Weapon to Re-engage if needed()
<<Fire Weapon to Reengage>>
Loop
Legend: OA Platform-Unique Function, OA Common Function

Figure 70. Control Engagement sequence diagram. Timing and sequence of events are reflected.

Use Case Name	Control Engagement
Participating Actors	Remote or Local Unit
Flow of Events	1. Unit controls engagement by providing guidance through Common Services (CS). 2. CS requests Data Information Services (DIS) for threat track/data. 3. DIS sends threat track for in-flight guidance to Weapon/Asset Service (W/AS) 4. W/AS provides guidance to Missile Execution (ME) 5. ME uses guidance to control missile in-flight to engage threat. 6. ME reports Kill Assessment (KA) to W/AS and CS 7. W/AS reports KA to other units and Force Planning (FP)/Coordination through EXCOMM
Entry Condition	Network access is granted. Threat data/Track (through Maintain Track Use Case) and Fired Weapon data link is available,
Exit Condition	A Kill or Miss is reported and disseminated.
Quality Requirements	Time for uplink to provide in-flight guidance is no more than 5 seconds.

Table 14. Use Case: Control Engagement.
A local or remote unit provides guidance to its ordnance (missile) for engaging a threat.

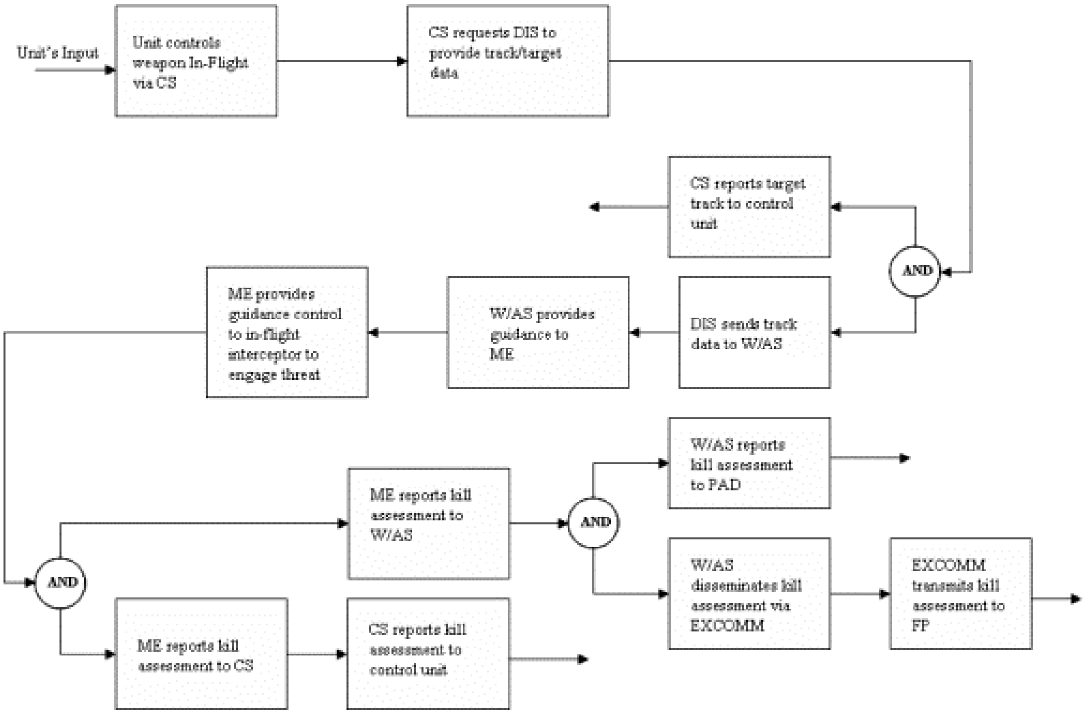

Unit's Input
Unit controls weapon In-Flight via CS
CS requests DIS to provide track/target data
CS reports target track to control unit
AND
ME provides guidance control to in-flight interceptor to engage threat
W/AS provides guidance to ME
DIS sends track data to W/AS
W/AS reports kill assessment to PAD
ME reports kill assessment to W/AS
AND
AND
W/AS disseminates kill assessment via EXCOMM
EXCOMM transmits kill assessment to FP
ME reports kill assessment to CS
CS reports kill assessment to control unit

Figure 71. Flow of events for Control Engagement use case.

The three selected IFC scenarios from *Integrated Fire Control for Future Aerospace Warfare* (Young, 2005) including Precision Cue, Launch on Remote, and Preferred Shooter Determination scenarios are presented as combination of selected use cases in the following use case diagrams (Figures 72 through 74):

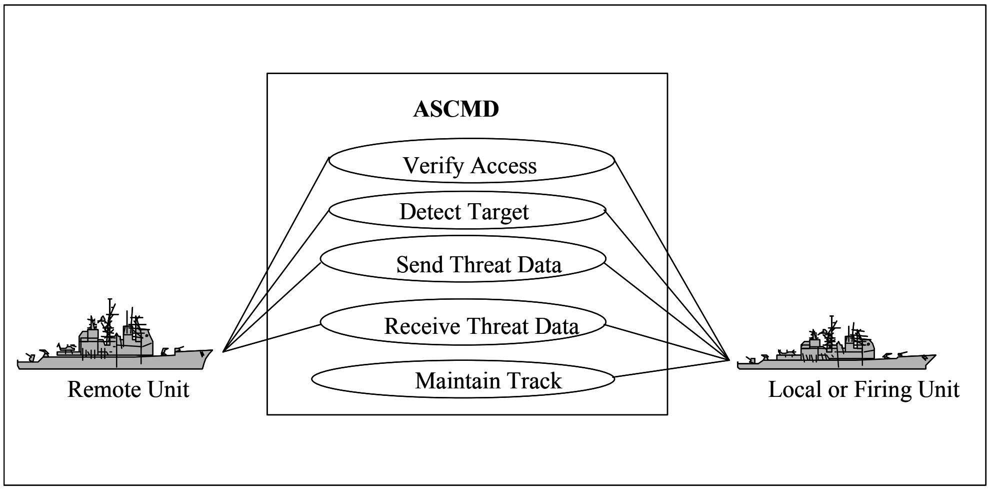


Figure 72. Reflection of the combination of the above use cases between the remote and local units to form the Precision Cue IFC use case.

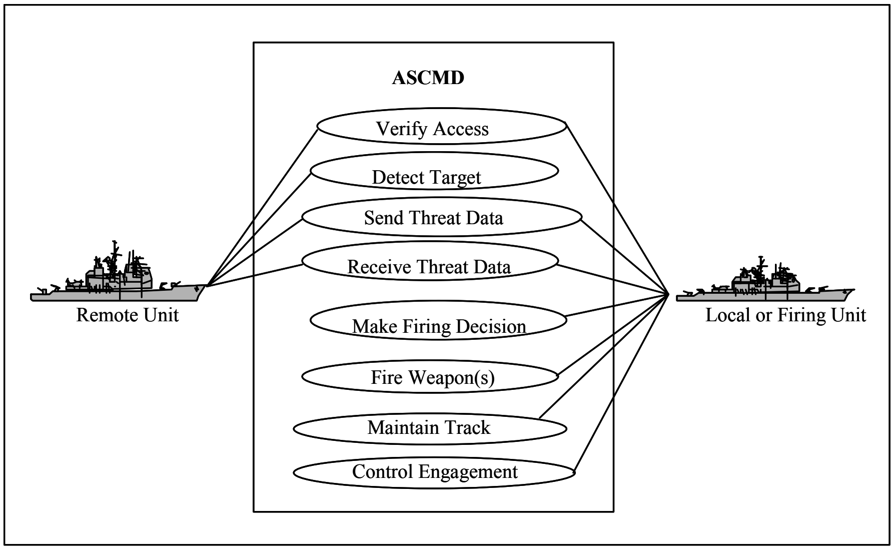


Figure 73. Reflection of the combination of the above use cases between the remote and local units to form the Launch on Remote IFC use case.

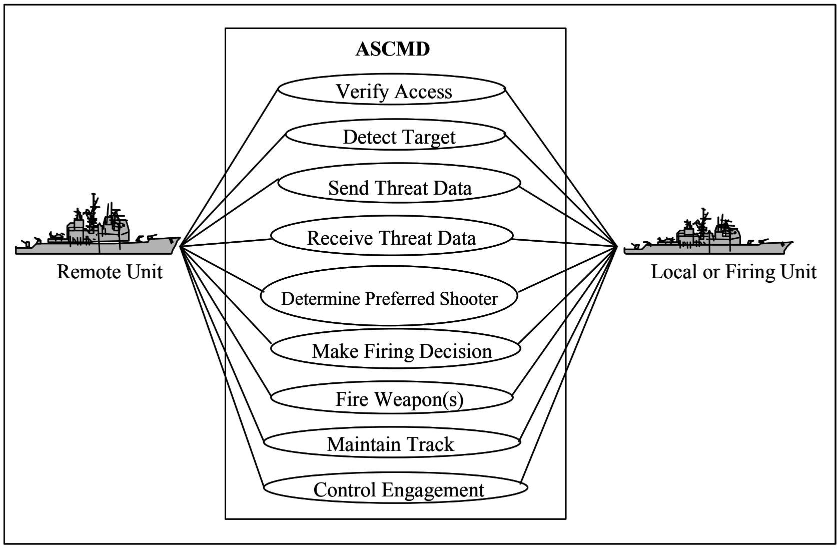


Figure 74. Reflection of the combination of the above use cases between the remote and local units to form the Preferred Shooter Determination use case.

4. Class Diagrams

The class diagrams in Figures 75 and 76 also use the format from the Object-Oriented Software Engineering textbook (Bruegge, Dutoit, 2004). The FORCEnet class diagrams describe the types of objects in the architecture, their static relationships, associations, attributes, and operations. The class diagrams include three main classes of FORCEnet: Composite Warfare Commander (CWC), FIGHTING UNIT, and GEO UNIT connected via association. The CWC includes Force commanders or Command and Control (C2) authorities that plan, coordinate, and control the fighting and geo units via FORCEnet and local networks. The fighting units include all remote and local fighting units such as CG, DDG, or frigates that operate within the battlespace and provide battlegroup defense by detecting and engaging all threats. For the cruise missile defense mission, the fighting units detect and engage a THREAT class that includes subsonic and supersonic cruise threats capable of attacking surface ships. The GEO UNIT class provides FORCEnet and local network links for communication among FIGHTING UNITS and CWC. Three main classes are linked or networked through a subclass called EXCOMM. The associations among the EXCOMM subclass and the GEO UNIT, CWC, and FIGHTING UNIT class are represented by shared aggregation. This means that the GEO UNIT, CWC, and FIGHTING UNIT classes exist independently. The subclass EXCOMM is composed of communication assets from three main classes including network, radios, data links, and satellite communications.

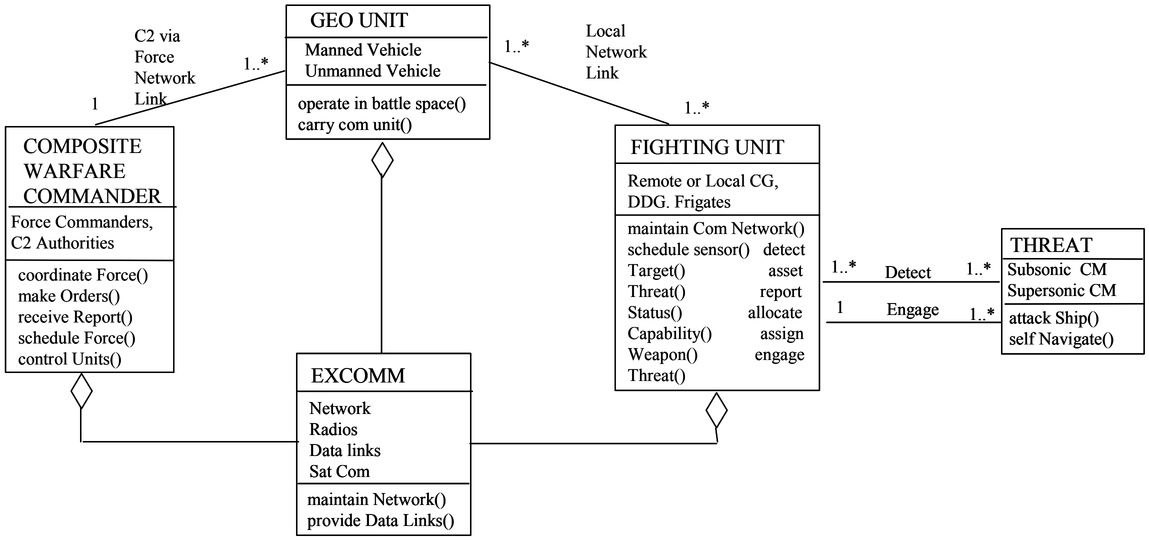


Figure 75. FORCEnet class diagram.
This diagram describes the types of objects in the architecture, their static relationships, attributes, and operations.

The FIGHTING UNIT class includes the remote and local fighting units. Remote fighting units are surface combatants such as CG, DDG, or frigates that are operational beyond the line of sight from the main body. Local fighting units are surface combatants such as CG, DDG, or frigates that are operational within the line of sight from the main body. Each remote or local fighting unit is composed of subclasses that include S&D, DIS, PAD, W/AS, ME, CS, and EXCOMM. These subclasses and their associations with the Local Unit and Remote Unit subclasses are modeled through shared aggregation. The local unit and remote unit subclasses are similarly associated to the FIGHTING UNIT class through shared aggregation. The subclasses originate from the OA Warfare System Domain Model. S&D provides sensor assets to search, detect, and track targets. Target track data is forwarded to DIS for compilation and fusion. DIS supports tracking by scheduling sensors for tracking and by messaging fused data to PAD for analysis and threat assessment. PAD analyzes the fused data for potential threats and assesses the threat as needed. If threats exist, PAD coordinates target prosecution with C2. Targeting orders are forwarded to W/AS for weapons allocation and scheduling. Weapons allocation and schedules are forwarded to ME for weapons assignment. ME controls the engagement and conducts BDA. CS displays the status of the tactical environment, maintains databases, and synchronizes time. Communications are maintained and linked by EXCOMM and provide the networks, radios, data links, and satellite communications needed to execute tactical actions and keep all units in the battlespace informed. The training class is associated with S&D, PAD, ME, CS, and EXCOMM to provide sustaining training for theater operators.

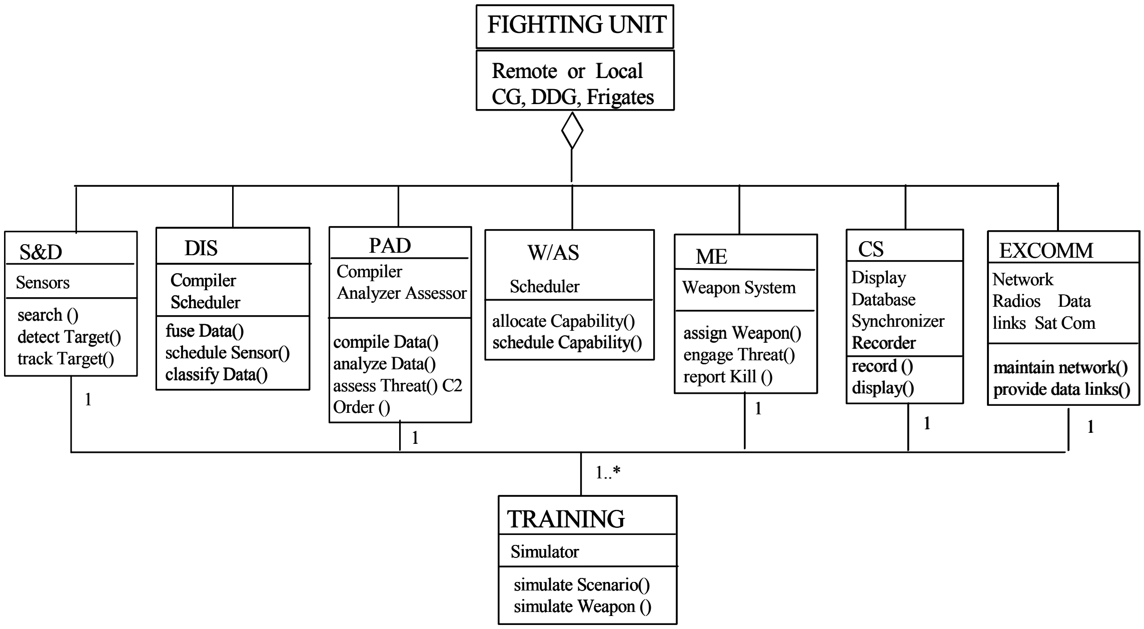
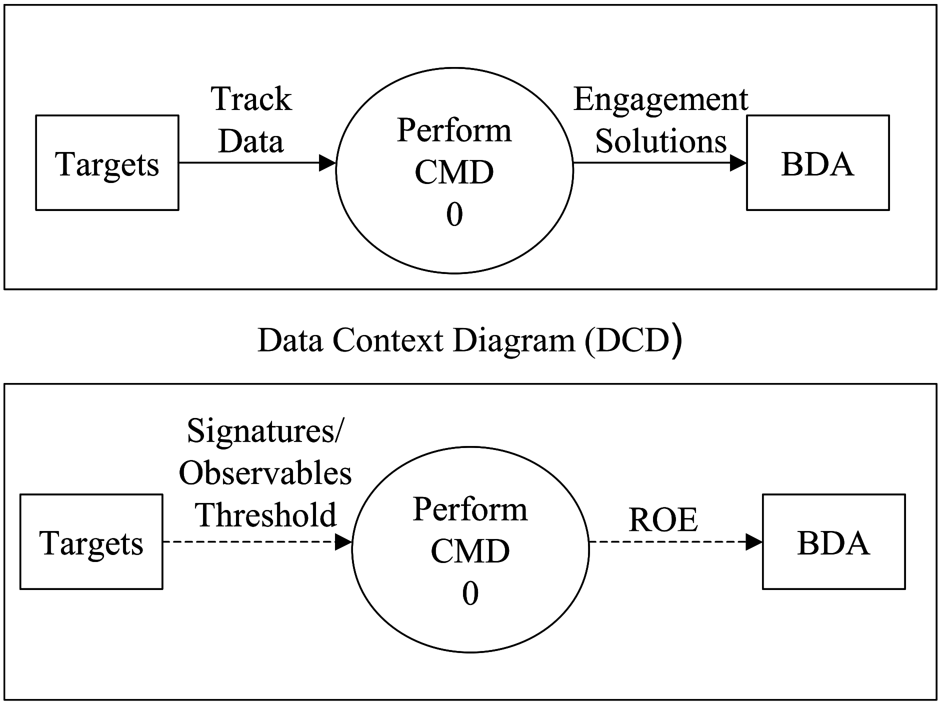


Figure 76. FORCEnet class diagram continued.
This diagram covers the fighting unit class and its functions.

5. Control Flow and Data Flow Diagrams

The Data Context Diagram (DCD) and Control Context Diagram (CCD), represented by Figure 77 below, are the top-level data and control flow diagrams for the Perform CMD architecture that define the boundary between the system and the operational environment. They are represented in parallel to illustrate their differences; the DCD represents the process view while the CCD represents the control flow view. Both models are composed of a single input and output flow process: Perform CMD as the main system function, two repeated terminators represented by (*), one input and output for the DCD, and one input and output flow for the CCD. The target is repeated for both input and output terminators because it is the only object that the system must detect and provide solutions to encounter. Track data as the data flow input is detected by the system on the DCD depending on the control flow input as observables or signature threshold performed by the system on the CCD, and engagement solutions as the data flow output is provided by the system depending on the Rule of Engagement as the control flow output. Neither diagram represents data stores because the intrinsic nature of cruise missile defense requires that data be manipulated into near real-time information for immediate tactical action. Thus, the data and control flow entities passing through the process are perishable and without shelf life, rendering data store representation irrelevant. The diagrams are identical for all IFC and tactical scenarios due to process and functional commonality, thus negating the need for additional IFC or tactical scenario specific models.

Both the DCD and CCD model the kill chain functions in the context of the OODA loop. The difference between the DCD and the CCD is that the CCD provides command and control functions and includes the information assurance function prior to any weapon-target engagement. The second and most significant difference between the DCD and the CCD is that the control flow of the CCD behaves like a switch; it activates and terminates processes. The data flow associated with the DCD represents process inputs and outputs as being activated or deactivated by the control flow of the CCD.



Data Context Diagram (DCD)

Control Context Diagram (CCD)

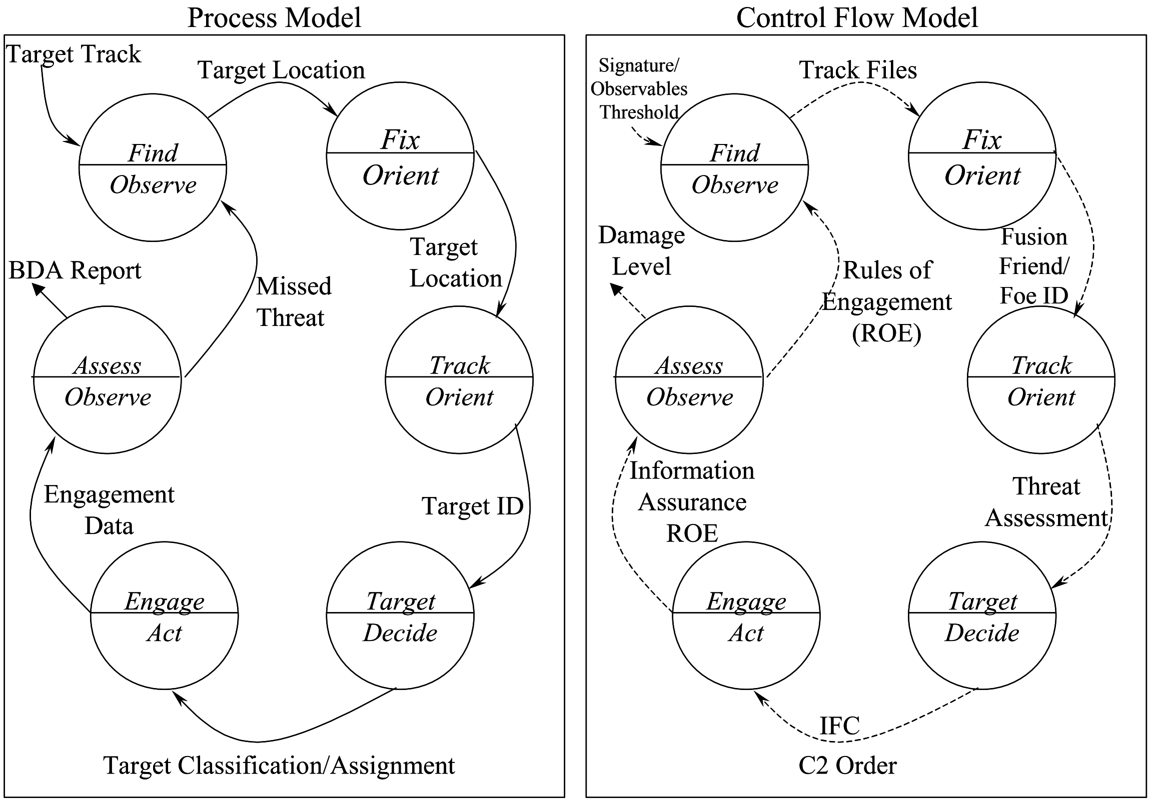
Figure 77. Perform CMD Data Context Diagram (DCD) and Control Context Diagram (CCD).
The DCD and CCD define the boundary between the system and the operational environment.

As mentioned above, the kill chain functions were modeled in the context of the Observe-Orient-Decide-Act (OODA) loop, which requires elaboration. See Table 15 below. As illustrated by the Table, the OODA loop models CMD engagements at a higher level of abstraction than the kill chain. The OODA loop Orient function is represented by two kill chain functions, Fix and Track. The OODA loop Observe function is represented twice to parallel the kill chain Find and Assess functions. The kill chain targeting functions mirrors the OODA loop Decide function. Each representation describes a serial-continuous process versus a one-step concerted process as will be discussed by the following Data Flow Diagram (DFD) and Control Flow Diagram (CFD). The kill chain and OODA loop functions share identical process terminators (*) located at the beginning and end of an engagement at the Find*-Observe* and Assess*-Observe* function pairing.

Function	Kill Chain Function	OODA Loop Function
Target Detection	Find*	Observe*
Target Location (kinematics and position)	Fix	Orient
Target Identification	Track	Orient
Target Classification	Target	Decide
Target Prioritization	Target	Decide
Weapon-Target Assignment	Target	Decide
Weapons Engagement	Engage	Act
Battle Damage Assessment (BDA)	Assess*	Observe*

Table 15. Kill chain versus OODA loop functionality.
The Table identifies the kill chain functions and places them alongside the equivalent OODA loop functions.

The Data Flow Diagram (DFD) and Control Flow Diagram (CFD), illustrated by Figure 78 below, represent the CMD architecture processing capability and are the next lower level of functional decomposition or level 0. They are one level down from the DCD and CCD, the context level, to describe the data or control flows as entities within the main system function (Perform CMD). In contrast, the DFD and CFD illustrate the data or control flows exchanged among all processes associated with and common to the kill chain and OODA loop functions. As with the DCD and CCD representations, the diagrams are identical for all IFC and tactical scenarios due to process and functional commonality, thus negating the need for additional IFC or tactical scenario specific models. The kill chain processes are also modeled in the context of the OODA loop, making it convenient and revealing to represent them both within each process bubble. Similarly, the DFD and CFD are rendered in parallel to illustrate how data flows of input and output from each process bubble of the DFD are associated with control flows of input and output from each process bubble of the CFD.



Data Flow Diagram (DFD) 0
Perform FORCEnet for CMD

Control Flow Diagram (CFD) 0
Perform FORCEnet for CMD

Figure 78. Data Flow Diagram (DFD 0) and Control Flow Diagram (CFD 0) supporting Perform CMD.
These diagrams represent the architecture processing capability**.**

The DFD0 and CFD0 models do not represent the FORCEnet architecture nor do they represent the architecture's procedures or its implementation. Instead, the DFD0 and CFD0 models represent the functional requirements of the kill chain and OODA loop. The fact that the models do not reflect the FORCEnet architecture is important because from the representation the processes appear procedural. The apparent modeling dichotomy is resolved because the kill chain functions, in the context of the OODA loop, are completed serially and continuously.

The power of the DFD0 and CFD0 models lies in the fact that, beyond their serial and continuous implementation, in this instance, they operate near-instantaneously given the limits of the queue length as a function of the stressor threat inter-arrival time and distribution of the CMD service time. Outside of those statistical constraints, data or control flow occurs instantaneously for both models. The near-instantaneous data and control flow response is precisely the feature that the DFD and CFD was designed to represent.

Another difference between the DFD and the CFD is identical to that of the DCD and CCD; the CFD provides for control flows between processes that explicitly support the command and control functions of integrated sensor fusion over all IFC scenarios, IA query prior to each target engagement, and BDA through the Find-Observe and Assess-Observe kill chain and OODA loop pairing. The BDA function can only be revealed by the CFD through decomposition of the CCD.

F. ASCMD SIMULATION MODEL

1. Simulation Method

The ASCMD simulation model is based on the proposed Open Architecture for FORCEnet-enabled cruise missile defense discussed herein. This simulation analyzes the following capabilities: data fusion techniques and algorithms; resource management scheduling and optimization methods; weapon and sensor management; and engagement functionality, initialization, and control.

The simulation applies the sensor-to-shooter kill chain, Observe-Orient-Decide-Act (OODA) loop, functional flow block diagrams, and the sequence diagrams discussed in previous chapters herein. The software used for creating the simulation was Arena version 10.0.

Before constructing the simulation, the physical geometry of the scenario had to be determined. Previously seen Figure 79 is the representation of the assumed layout of the fleet at the time of the cruise missile attack (also included below). The layout shows a fleet consisting of a main body (CVN, LHA, or LHD), two guided missile destroyers, two cruisers, a frigate, and the combat air patrol (CAP). The CAP is assumed to be part of the sensor grid, but is not assumed to be a launching platform for engagements. The threat axis and layered zones were used to create the fleet layout, but are not used for any other purposes in the simulation.

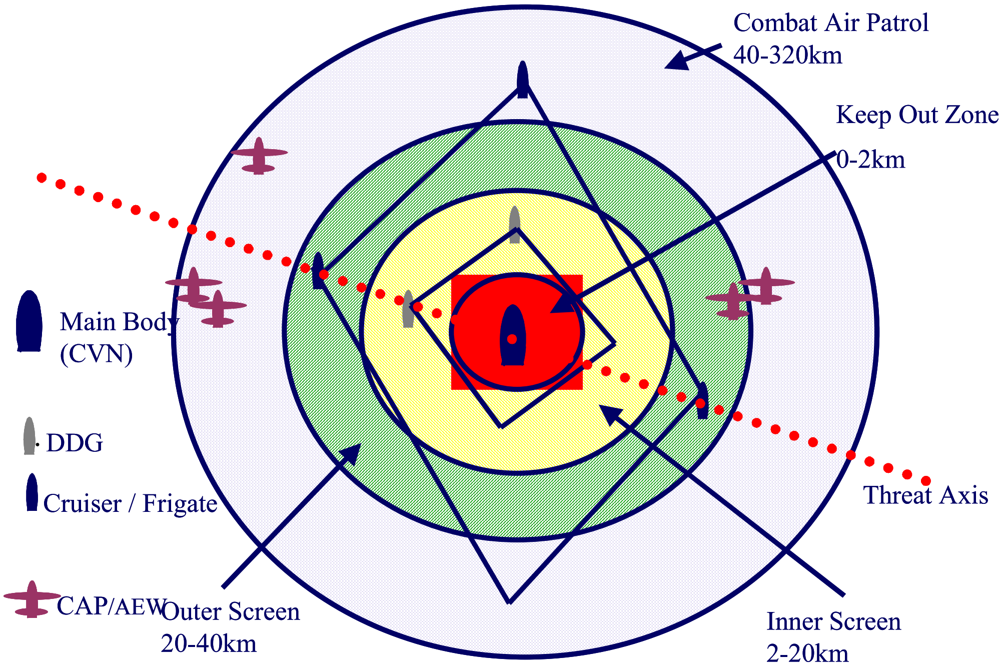


Figure 79. Overall physical layout of battlegroup.
This layout is used to visualize scenarios for validating the proposed ASCMD functional architecture.

The coordinate system used in the simulation is assumed to have the main body at the origin with unrestricted steaming set in the line of 0 degrees relative bearing. All subsequent angles process in a counterclockwise fashion from that point as shown in Table 16.

Ship	Range	Angle	Angle	X	Y
(class)	(km)	(degrees)	(radians)	(km)	(km)
Frigate	200	0	0	0	200
DDG2	15	0	0	0	15
Cruiser 2	30	135	2.3561925	-21	21
DDG1	10	135	2.3561925	-7	7
Cruiser 1	25	315	5.4977825	17	-17
CVN	0	0	0	0	0

Table 16. Locations of the members of the battlegroup.
The Table details the unit location and the distance in x and y coordinates from the main body and the angle from the main body.

Figure 80 shows four sub-models from a top-level perspective labeled as the four basic elements of the OODA loop, plus a Re-observe sub-model. The order of the top-level simulation represents the sensor-to-shooter kill chain. The simulation flows first from the first detection of the incoming threat to an Observe sub-model, then to the Orient, to the Decide, next to the Act, and finally to the Re-observe sub-model. The Re-observe sub-model is the addition used to create the loop portion. This loop behavior is critical to the success of the FORCEnet-enabled cruise missile defense schema and is one of the major changes to the overall architecture that was not within the PEO IWS model after the first salvo is fired. The Re-observe sub-model in the architecture is a critical addition; the previous model provided a one-time shot expecting 100% skin on skin elimination of an incoming ASCM. This was not a realistic assumption because properly conducted threat doctrines created by own ship personnel will invariably contain some variant Shoot-Look-Shoot policy. The structure of the following discussion will explore each of the major sub-models shown in Figure 80 moving from left to right through the kill chain.

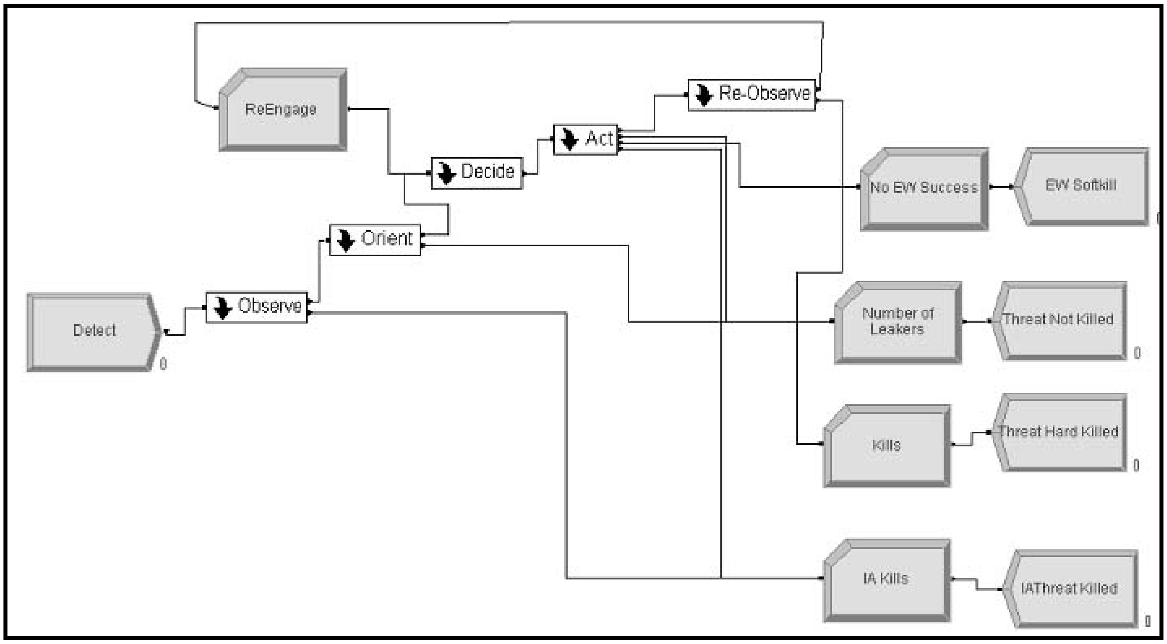


Figure 80. The top-level view of the ASCMD simulation model. This reflects the OODA loop and Re-Observe sub-models.

For the purposes of the simulation, the threat information is the tracked entity within Arena and is created in the process labeled “Detect.” The concern here is the defense of the capital ship, not the neutralization or retaliation against the aggressor. Thus, we leave parts of the cruise missile’s flight before detection to other explorations. The Detect process allows the creation of the two major scenarios to be explored. First, the raid scenario was developed where ten missiles attack the main body, capital ship, in a coordinated fashion from ten bearings with only a maximum of ten seconds from the first missile to the last. The second scenario is one where all ten missiles are inbound from a single shooter so that they are all from the same bearing, but are separated by some inter-arrival time. In the second scenario, the time was a triangular distribution with a minimum of ten seconds between, a maximum of one, and a mean of thirty seconds. A third and trivial scenario was where one missile was used for troubleshooting purposes, the results of which were not evaluated. All threats are considered to be of the same missile type and moving at a rate of 0.6 km/s. Arena would crash every time uncertainty was introduced with respect to the threat speed although no errors were ever detected. Therefore, the authors used a constant value of 0.6 km/s. Figure 81 below provides a visual representation legend of the main types of processes used in the ASCMD simulation model.

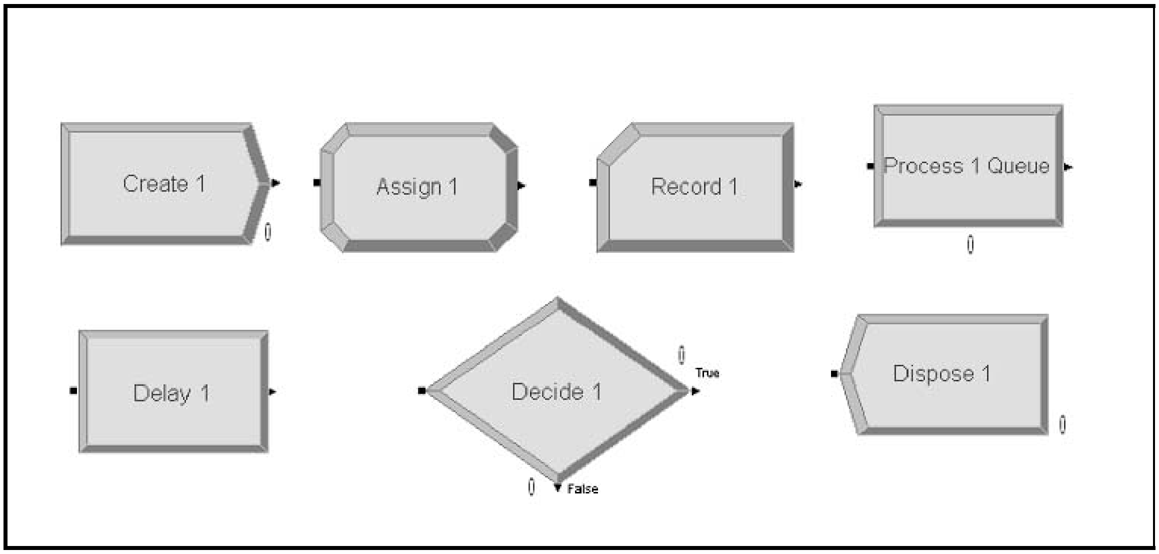


Figure 81. Arena icons.
These icons represent the main processes used within the Arena software to run simulations.

Several tasks occur in the Observe sub-model as the entities interact with the simulation as shown in Figure 82. First, the attributes that characterize the threat and its responses are assigned to the "entities," which are the threat missiles themselves, in the initialize blocks. These assigned values are used throughout the simulation as inputs for various calculations. Some values, such as the current location of the threat, are used in subsequent calculations; therefore, several initialize processes require Arena to perform the calculations in the prescribed order. Later updates to these variables will likewise be broken into multiple blocks, which can be viewed as collectively updating Arena's stored values at that instant. When the detection takes place, one of the assigned properties is the distance at which the threat is located from the main body. This property allowed the authors to account for a host of conditions including but not limited to weather, electronic attack, sensor locations, and abilities. The probability that the threat will be able to penetrate further without being detected increases as these conditions worsen.

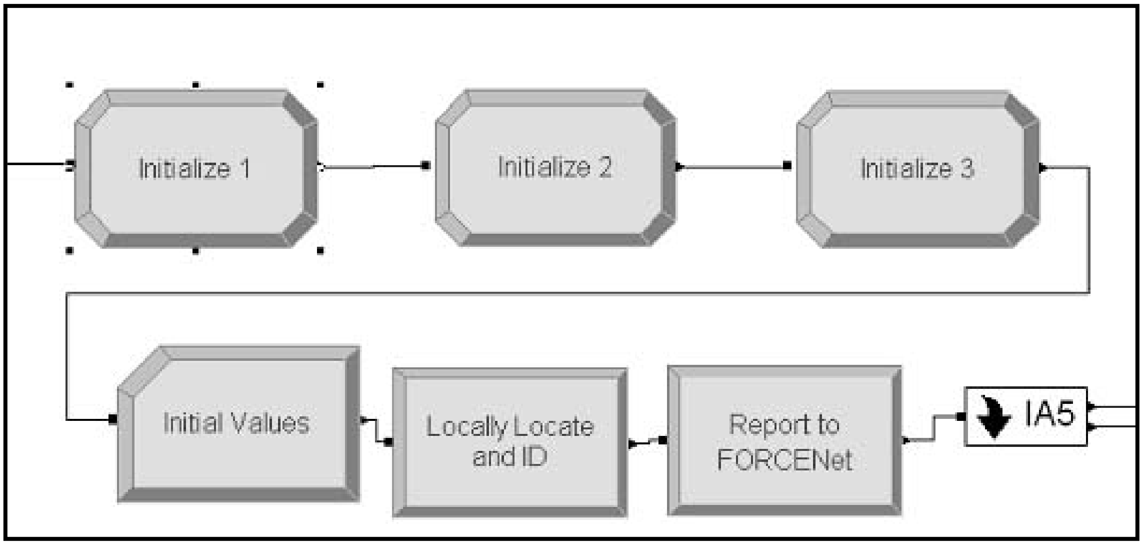
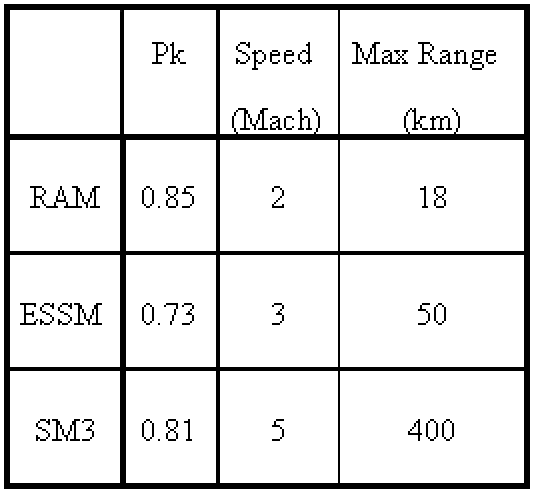


Figure 82. Layout of the Observe sub-model.
The Initialize blocks contain threat missile data characterized by threat attributes and responses.

All times in the simulation were measured in seconds, distance in kilometers, and angular measure in radians. Table 17 lists the interceptors and their attributes. Processes and delays used through the simulation are listed in Appendix B along with their associated distribution.



	Pk	Speed (Mach)	Max Range (km)
RAM	0.85	2	18
ESSM	0.73	3	50
SM3	0.81	5	400

Table 17. List of own ship missile interceptors and their assumed properties: Rolling Airframe (RAM), Evolved Sea Sparrow Missile (ESSM), and Standard Missile (SM) 3.

Once the initial values have been established, the Arena-assigned properties such as entity creation time are recorded in the "Initial Values" block. After the initial mechanics of setting up the mathematics have been satisfied, the system then moves into the process of tracking the entity. First, the entity encounters a delay as a generic sensor has detected it but still must track, localize, and provide a preliminary assessment. The next block in the Observe sub-model was where the sensor's command and control function alerts the FORCEnet through an external communication link. The communication and decision steps that are used throughout the simulation are not only used to simulate that function, but to also simulate allocation of network resources and account for network's capacities and capabilities.

As information leaves the bounded area local to the sensor and is sent to the larger community, a parallel background network process takes place to ensure that the integrity of the data is maintained, thus concluding the Observe sub-model. This Information Assurance step has two major components and the sub-model associated with it is shown in Figure 83. First, the network must resist attack by detecting and isolating information that is part of an attack or corrupted through the course of "normal" operations before it is inserted into the larger operating picture. If no distortion of the data is detected, then the system allows the information to continue through the simulation. When a corruption is detected, then the system places the data in the "IA Penalty Box" where the data is reevaluated to determine if the threat data is legitimate or can be repaired. This delay can involve everything from bit and frame synchronization to requests for re-transmission among other troubleshooting processes. Based on the success of the repair function the threat is either passed on to the larger FORCEnet community or terminated as a false alarm in the "IA Threat Killed" disposal block in the main model. This background network process allows the system to resist information attack and is robust enough to recover from attack.

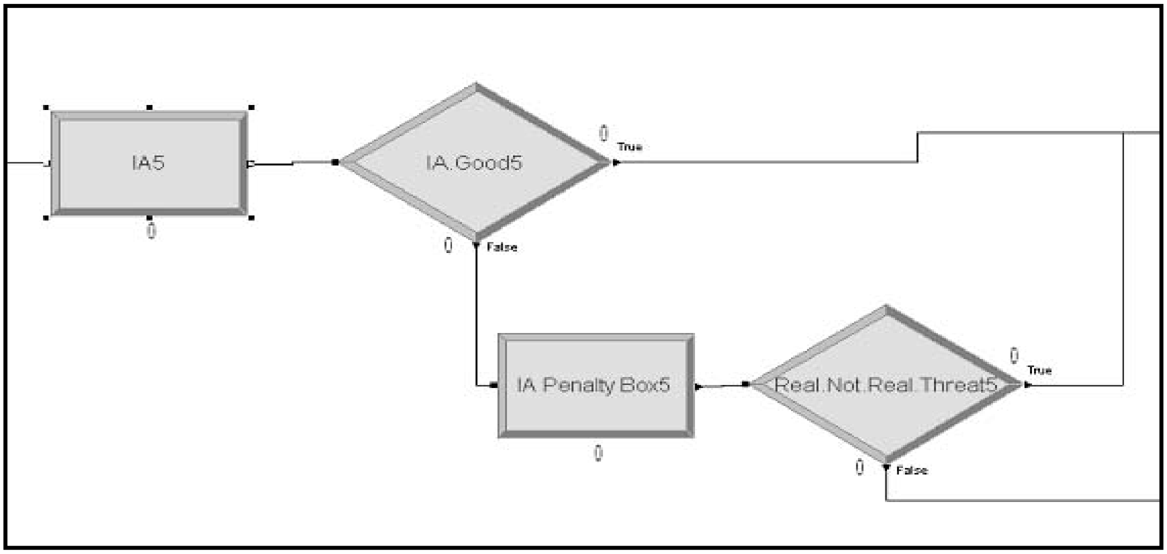


Figure 83. Layout of the IA sub-model.
This sub-model is run as a background process to ensure data integrity.

Moving from the Observe sub-model to the Orient sub-model, as shown in Figure 84, a block titled "Update Common Control Picture" simulates the tasking, network allocations, and delays as the overall FORCEnet picture is updated with the threat information. The stored values for the variables that have a time-based component are then updated in the simulation. Prior to introducing humans into the loop or making decisions based on the current threat information the simulation determines the threat missile location. If at any point in the simulation the threat enters the keep out range of 2 km, the threat is considered to have completely penetrated all defenses and is disposed of in the main model in the "Threat Not Killed Block." The evaluation takes place in the decision process, "PROX." The times associated with the updating and decision-making are part of updating the FORCEnet function and form a portion of the model mechanics that do not have a directly associated delay. At various points in the remainder of the simulation, checks are made based on updated information; the process immediately preceding them accounts for their delay.

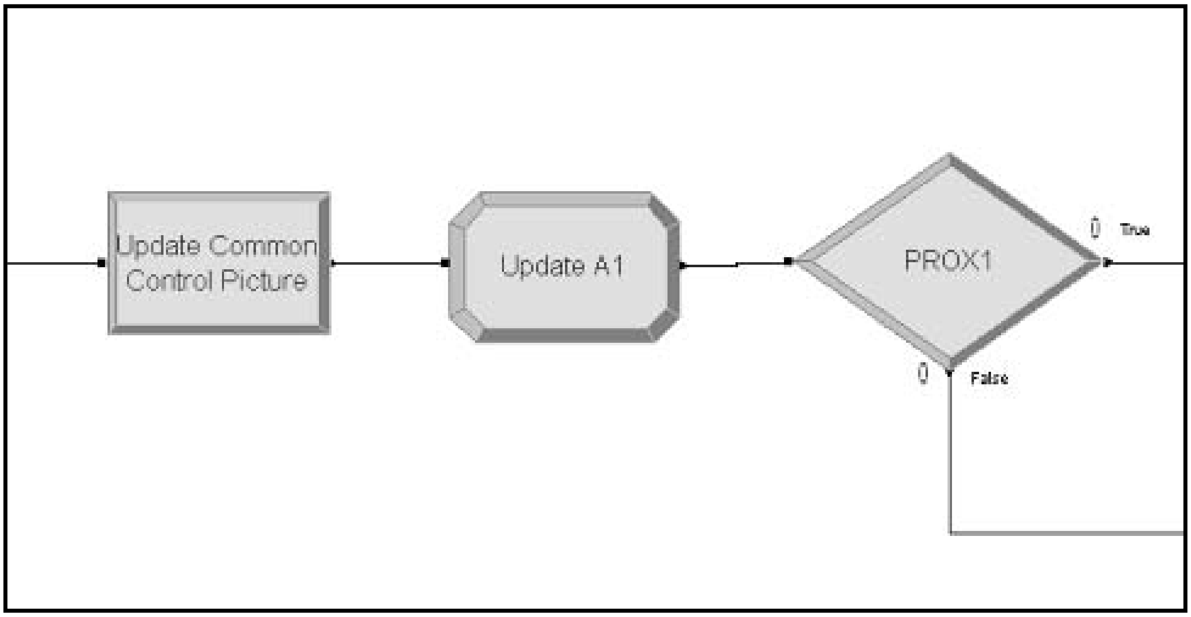


Figure 84. Layout of the Orient sub-model.

Moving out of the proximity check and concluding the Orient sub-model the threat is then processed in the Decide sub-model. This sub-model contains only one block, the “IFC” portion. Here the tactical action officer or other human in the loop is designated as the decision authority. This process is modeled as a queue and requires time to make a decision and order its execution. Although this process was modeled as a single Arena selected random distribution, this block allowed the investigator to adjust for experience, training, damage, rules of engagement, and overall level of data integration.

Next, the simulation moves into the most complex Act sub-model. While some of this simulation would occur in concert with the CWC’s decision-making process, the order was chosen to allow Arena to represent the time delays and functional allocations. Therefore, the authors acknowledge that the weapon and platform selection are within the scope of the CWC’s responsibilities even though they are shown in the Act sub-model. Figure 85 shows the four sub-models that comprise the Act sub-model. The IA6 sub-model provided defense of the FORCEnet information and is an exact replication of the previous Information Assurance sub-model in Figure 83.

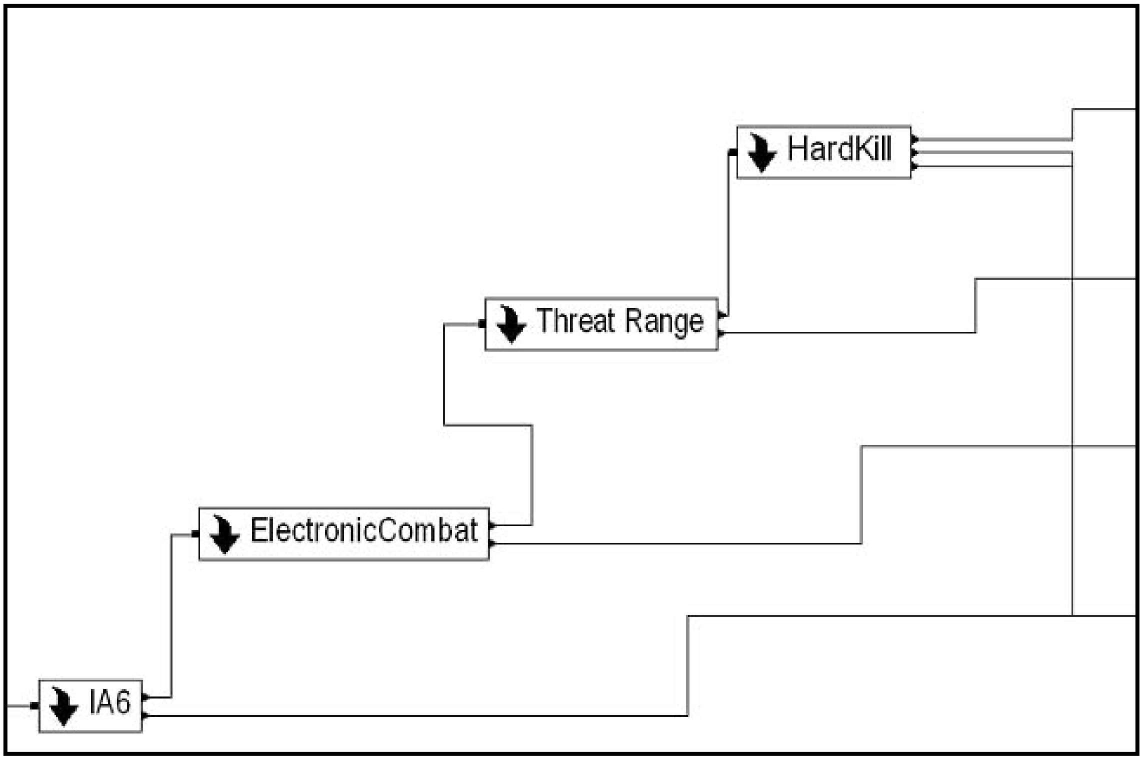


Figure 85. The Act sub-model.
This sub-model is run as a background process to ensure data integrity.

The simulation next considers the battlegroup's use of electronic countermeasures (ECM) that is assumed to be continuous throughout the engagement and occur in parallel to all of the other processes. However, Arena is a discrete-event based simulation and to simulate the parallel process the authors simply reversed the sequence of events in the Electronic Combat sub-model shown in Figure 86. The decision is first made to determine if the ECM will be successful. If the ECM is not successful, electronic combat occurs for the remainder of the engagement but does not affect the outcome. When ECM successfully defended the main body against attack, the simulation adds the appropriate delay and then disposes of the threat as a successful "EW Softkill."

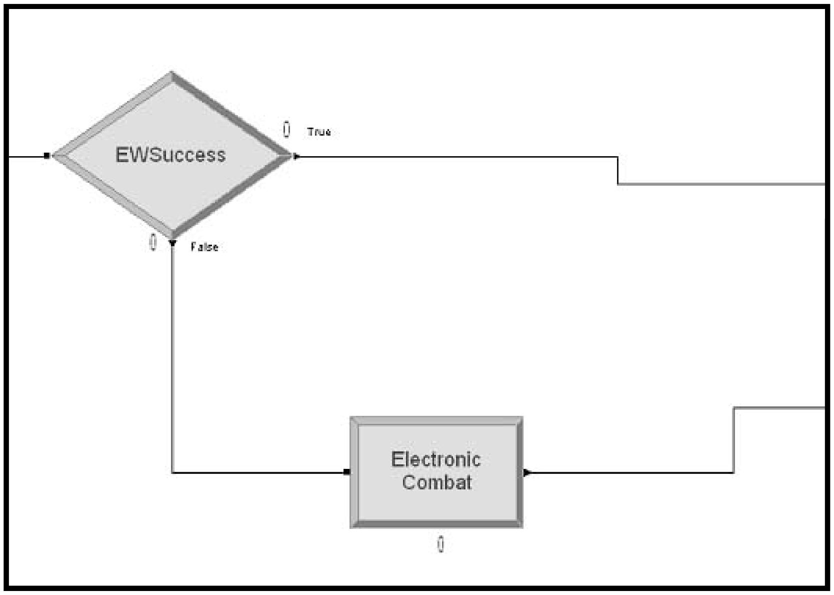


Figure 86. The layout of the electronic combat sub-model.

Prior to engaging with missiles, the simulation updates the threat location and verifies that the threat is not within the keep out range. The simulation segues into the Hardkill sub-model where the weapon-platform pair is selected and the missile engages the threat. The Hardkill sub-model is shown in Figure 87 and consists of the individual platforms and the platform selection process.

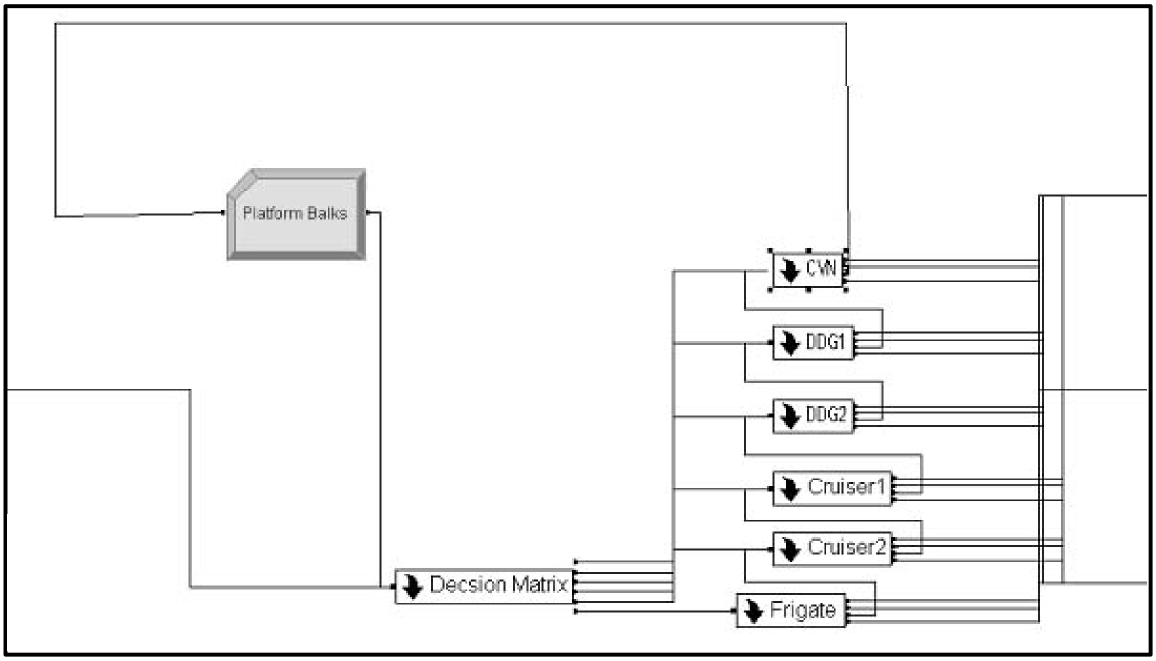


Figure 87. The layout of the Hardkill sub-model.

The selection process used is a bubble sort comparison whereby the closest platform is chosen to attack the threat. The process begins by updating the threat location and compares the distances of the threat to each of the battlegroup platforms against the distance of the threat to the main body. A comparison is made of each platform against the distance to the platform closest to the main body. These comparisons continue in order until the frigate furthest from the main body is used as the basis for comparison. The simulation preserves the layered defense concept by selecting the platform closest to the threat to ensure that the engagement is kept as far from the main body as possible. In addition to measuring the threat-to-platform range, the simulation predicts a 10-second future location to ensure that the target is not past the closest point of approach (CPA) and opening. By verifying that the target is closing, the authors ensured that the interceptor missile would close and engage the threat. Although some of the interceptors could close and engage the threat the time required would negate the tiered defense structure. Figure 88 shows the platform comparison order, while Figure 89 shows the comparisons and decision logic for the main body. Each additional platform has the same logic pattern but uses the threat distance to that platform as the basis for comparison.

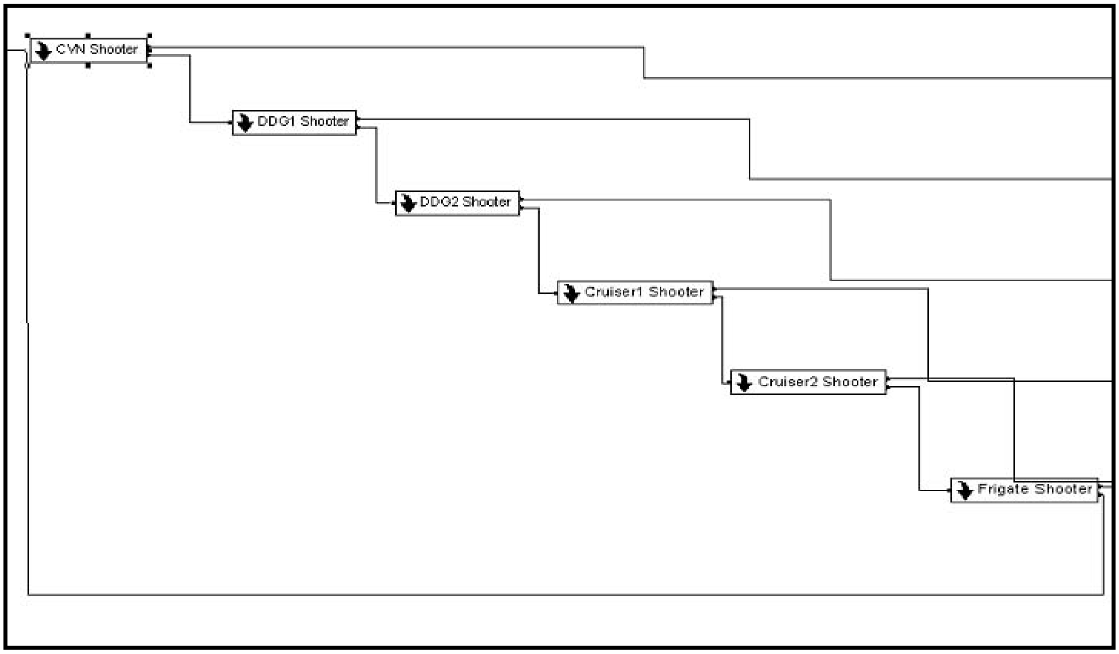


Figure 88. The layout of the Decision Matrix sub-model.
The order of platform basis used for comparison.

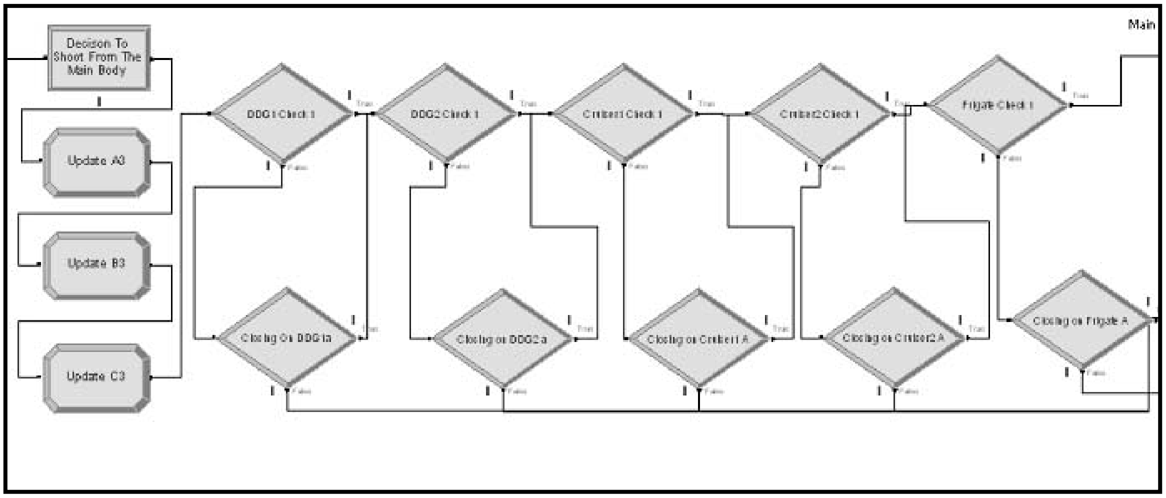


Figure 89. The diagram shows the layout of the logic used when the main body is the basis for comparison in the CVN Shooter sub-model.

The simulation selects the best weapon-platform pair, orders missile launch, and moves into the platform-specific engagement. As the system moves from the CWC's console to the platform an IA step was again performed as in Figure 83. The platform's combat information center (CIC) receives firing orders and passes the engagement information to the weapons system in the block labeled with the platform's name, which in the case of the first guided missile destroyer (DDG) is "DDG1." Arena updates its location, verifies that the weapon is not within the keep out range, and begins the firing process in the Fox2 sub-models. Figure 90 shows the receipt of the firing solution by the engaging platform through the pass to the Fox2 sub-model.

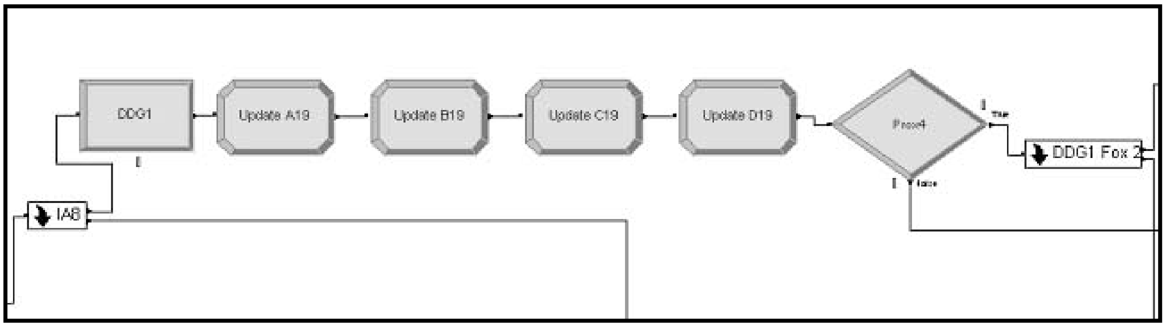


Figure 90. The layout of the DDG1 sub-model.

Table 17 shows the interceptor properties assumed for each interceptor. The three interceptors used are the RAM, SM-3, and ESSM. Once the engaging platform has been chosen the interceptor must be selected. The decision block chooses the weapon with the smallest range capable of engaging the threat. Thus, long-range weapons are reserved for long-range engagements that extend the layered defense range. Finally, having selected the appropriate weapon, the simulation begins the process of engaging the threat.

Figure 91 provides an example of the sub-model containing DDG1's ESSM fly-out. First, the simulation checks weapons availability, updates the time-based simulation parameters, verifies the threat is within interceptor's engagement range, and holds fire until it is. It then spins up the guidance and firing systems of the interceptor.

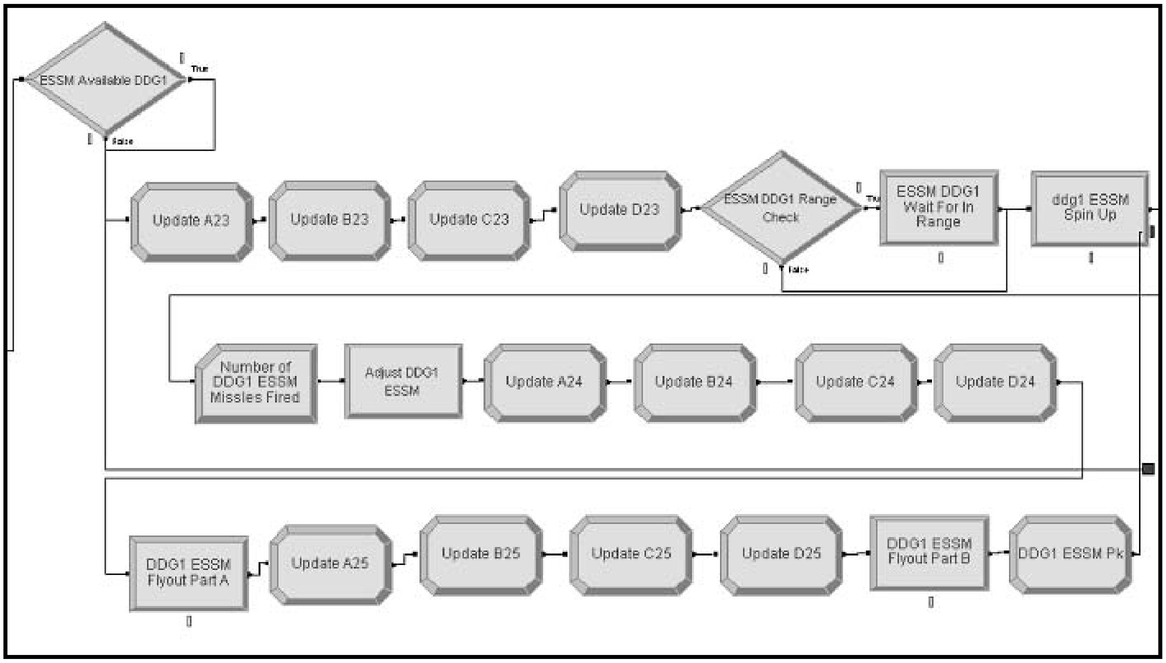


Figure 91. The layout of the DDG1 ESSM sub-model.

At this point, the simulation began the interceptor launch sequence, recorded the number of missiles fired, and decremented inventory. With the launch sequence started the inventories are adjusted to prevent the system requesting fire from more interceptors than are remaining in the launchers. The time-based values are updated and part one of the fly-out is executed. Positions are again updated and the second part of the fly-out is conducted. Finally, a score is assigned to the fly-out in the block "DDG1 ESSM Pk." This score is a random triangular distribution centered about the Pk values shown in Table 17.

The fly-out values were calculated as a two-part engagement. First, the interceptor flies at a right angle to the threat missile's line of bearing until it is on a line directly between the threat and the main body. The interceptor then executes a maneuver to go into a head-to-head engagement with the threat. This algebraically simulates a worst case of the proportional guidance solution. This allowed the authors to obtain a discrete solution for the time to engage the threat from an arbitrary platform, with an arbitrary interceptor, against an arbitrary threat with arbitrary properties.

Having assigned a random number based on the probability of kill associated with that weapon system in that firing doctrine, the simulation then undergoes a process whereby a sensor is tasked to see if there is still an inbound threat. All values are given a final update and an evaluation takes place where if the random number is greater than 0.6, then the threat has been neutralized; if not, then the encounter is termed unsuccessful. The value of 0.6 was chosen to bias conservatively the simulation to ensure a kill had taken place. Figure 92, the Re-Observe sub-model, shows these steps. If successful, the threat is disposed of in the "Threats Killed Hard" block. If unsuccessful, the threat then is looped back to the Observe, the Orient, the Decide, or the Act sub-model, or if no loop is used it is disposed of as a "Threat Not Killed."

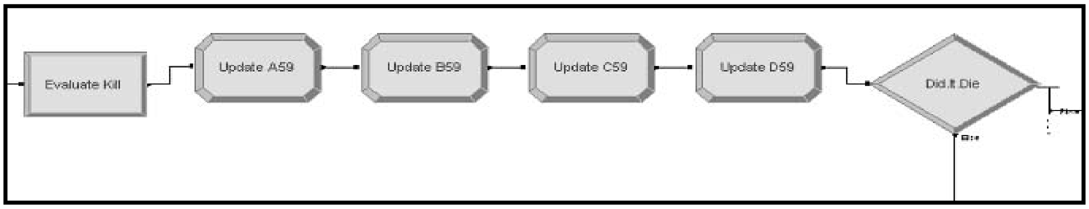


Figure 92. The layout of the Re-Observe sub-model.

The simulation allowed the authors to begin to evaluate the FORCEnet-enabled concept of cruise missile defense from the sensor-to-shooter when applying either the PEO IWS architecture or the proposed architecture. In the model we have implemented the following open architecture design principles: search and detect, data information services (IA), planning and assessment and decision, weapon / asset services, and mission execution. Furthermore, the integrated fire control scenarios described earlier are simultaneously integrated into the simulation. The precision cue scenario involves networking, tasking, and integrating multiple sensors, while the forward pass scenario involves the communication of the threat data, and the launch on remote takes the first two scenarios and uses them to order the defense of the ship.

2. Simulation Data and Results

The authors originally looked at the mean time interval to accomplish several tasks. However, the authors were also faced with the problem that the geometry of the threat and the battlegroup significantly confounded the time-based results. To provide a high-fidelity simulation, the attackers' detection range and direction were allowed to vary as they would during an actual attack. Further, the times assumed for communication, network processes, and interceptor spin-up were arbitrary to preserve the unclassified nature of the simulation and the report. Therefore, the authors recognized the value of using time as an MOP, but were forced to remove that MOP from consideration as it would not have the same fidelity and any conclusions based on time MOP's would be suspect.

The MOP's used in the evaluation were the means of the following: the number of IA attacks; the number of threat missiles killed by electronic combat; the number of threat missiles killed by interceptor missiles; the number of reengagements; and the number of threat missiles that leaked, or were unsuccessfully addressed by defensive countermeasures. The MOP's used, came directly from the value system design. The top-level function requires defending against a cruise missile attack; therefore, each measure was chosen to measure directly how well the system performs this function. The key measure that addresses this function is the number of leakers. All other measures simply add clarity to this one measure of performance.

All values are measured at the 95% confidence level. The values in Table 18 presents the raw data collected, and are shown as the mean and tolerance associated with a 95% confidence interval. The PEO IWS architecture simulation results were held as the control group in both the raid and the stream case. Throughout this discussion, comparisons limited to evaluations within the scenarios.

Group	Scenario	IA Kills	EW Success	Kills	Total Kills	Leakers	Reengagements
r p	**R p**	**0.048±0.010**	**1.921±**	**9**	**0 0**	**5 7**	**0 ± 000**
r p	**t am op**	**0.047±0.010**	**1.978±**		**0 0**	**070**	**0 ± 000**
Test Group	Raid Observe	0.056±0.010	2.041±0.080	6.650±0.090	8.290±0.190	1.064±0.060	1.710±0.080
Test Group	Raid Orient	0.048±0.010	2.207±0.080	6.714±0.090	8.969±0.180	0.838±0.050	1.937±0.100
Test Group	Raid Decide	0.048±0.010	2.330±0.080	6.738±0.090	8.747±0.180	0.679±0.050	1.993±0.100
Test Group	Raid Act	0.048±0.010	2.308±0.080	6.839±0.090	9.116±0.180	0.599±0.050	1.975±0.100
Test Group	Stream Observe	0.056±0.010	2.095±0.080	6.616±0.100	8.767±0.190	1.044±0.060	1.671±0.080
Test Group	Stream Orient	0.047±0.010	2.237±0.080	6.719±0.100	9.195±0.180	0.806±0.050	1.869±0.090
Test Group	Stream Decide	0.047±0.010	2.330±0.080	6.757±0.100	9.253±0.190	0.664±0.050	1.911±0.100
Test Group	Stream Act	0.047±0.010	2.421±0.090	6.785±0.100	9.134±0.190	0.554±0.040	1.940±0.100

Table 18. The raw data collected from the ASCMD simulation model showing the mean with tolerances at 95% confidence.

3. Simulation Analysis

First, the numbers of reengagements were plotted in Figure 93. The plot shows that the loops in the architecture were allowing statistically significant opportunities at the p = 0.05 level for the system to subsequently engage the threat. Further, the plot shows that entities are not looping indefinitely through the system. Statistically, there is a difference in the means within the test groups based on loop location; however, this is due primarily to large sample size. The overlapping confidence intervals create a conflict where the statistical significance is not enough for us to conclude that difference is substantial enough to assign a best location for the loop.

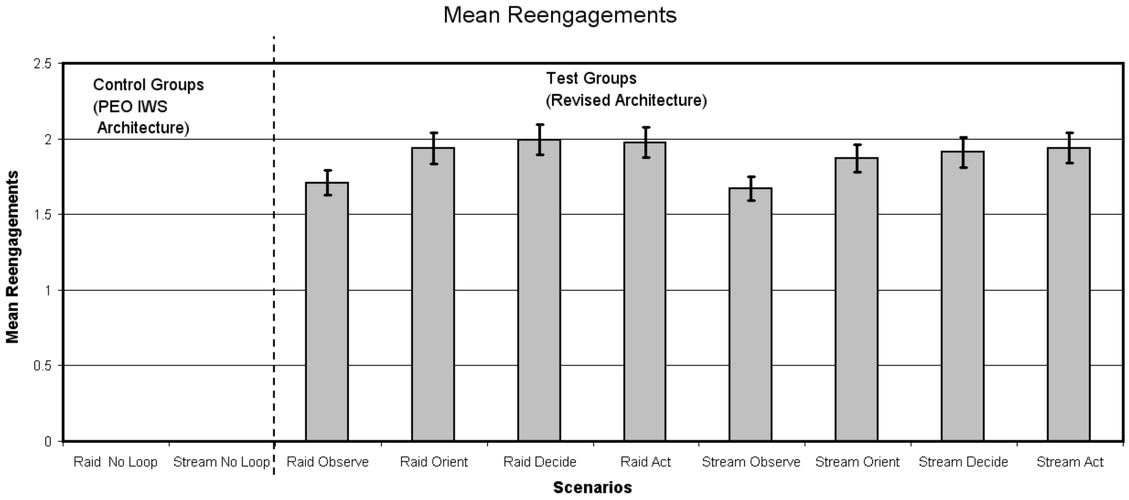


Figure 93. The mean number of reengagements plotted at the 95% confidence level.

The Electronic Warfare (EW) success and the Information Assurance (IA) kills were plotted in Figure 94 and Figure 95 respectively. In both cases, the test groups show an increase over the control group. The increases were statistically significant at the p = 0.05 levels, but the overlapping confidence intervals preclude the authors from drawing any conclusions. This was an expected result caused by an artifact in the design of the simulation. The information assurance and electronic warfare portions have decision blocks that are “2-way by chance.” Therefore, there is an additional opportunity for the simulation to process randomly an IA or EW success when the looping occurs. However,

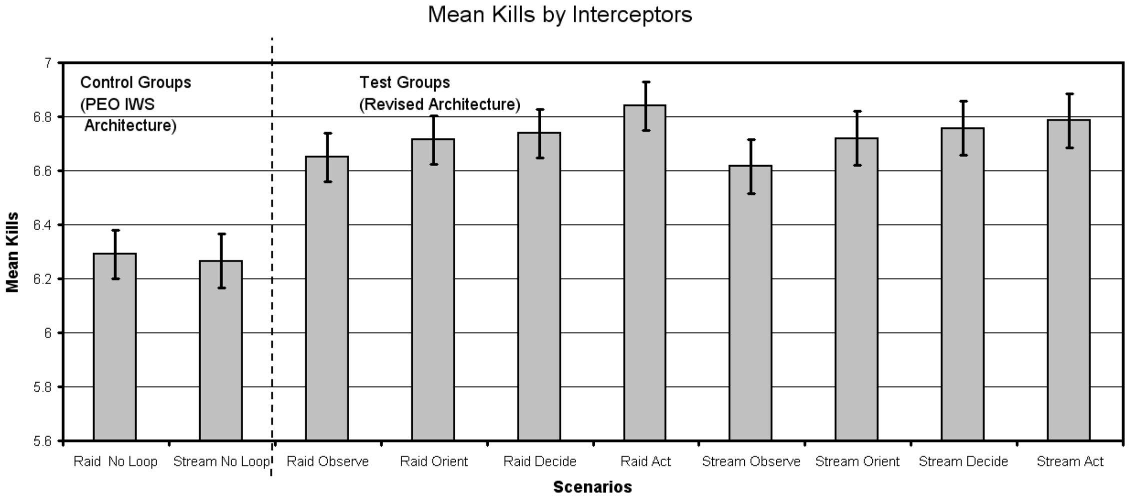


Figure 96. The mean number of kills by interceptors at the 95% confidence level. The test group shows an increase over the control group.

Further, the confidence intervals do not overlap between the control group and the two test groups. The improvement was measured by increased number of interceptor kills and there is a statically significant improvement as the Observe-Orient-Decision-Act (OODA) loop is truncated and the loop portion is moved closer to weapon release in the Act sub-model. This is an expected result as the response time is improved by moving eliminating steps. However, the overlapping confidence intervals within the test groups preclude drawing any conclusions with respect to the loop location. In this case, the authors were able to state that there is a distinct and measurable difference between the two architectures when the stimuli are a raid or a stream.

The number threats that leaked through the defenses were plotted in Figure 97 below.

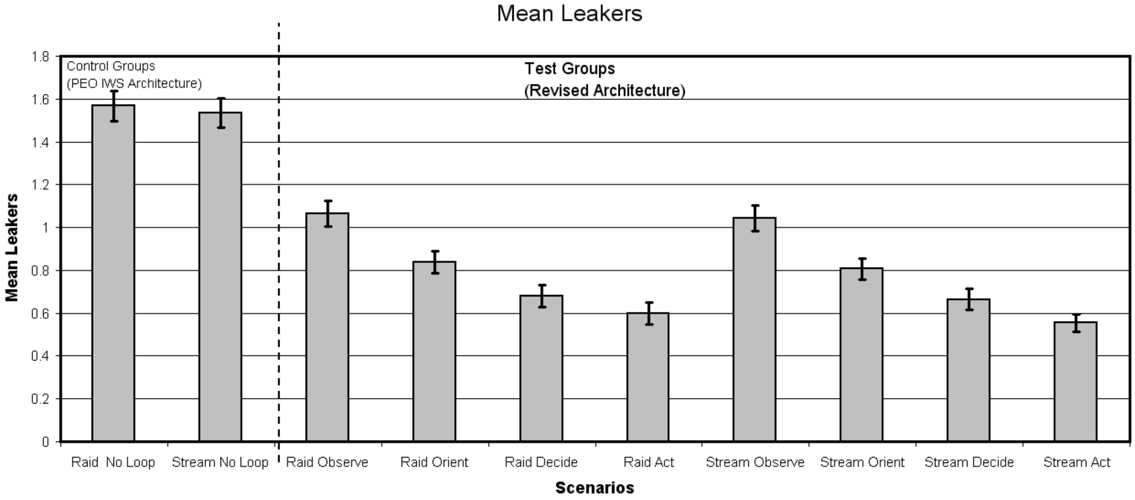


Figure 97. The mean number of leakers at the 95% confidence interval. The plot shows that the test groups performed significantly better than the control groups by allowing fewer leakers to get through.

For the leakers MOP, the better number was the lower number. The number of leakers MOP confirmed the previous result that the revised architecture was an improvement. Further, it allowed the authors to discriminate which architectures were better among the test groups. By moving the loop back to the Decide sub-model, the architectures performed significantly and distinguishably better at the previously described levels. This seems logical as many of the steps performed in the initial observe and orient steps would be redundant once initial decision-making and re-observation had taken place. The addition of the integrated fire control scenarios would also tend to argue for the loop location at the Decide or Act sub-model. The overlapping confidence intervals again prevented a distinction between the Act and the Decide sub-models.

The Findings and Recommendations section draws the conclusions available from the simulation results, provides a final overview of the proposed architecture, and discusses outstanding issues for this project's stakeholders and decision-makers.

THIS PAGE INTENTIONALLY LEFT BLANK

V. FINDINGS AND RECOMMENDATIONS

The research and analysis efforts reflected on the previous sections of this report yielded a proposed ASCMD functional architecture that was compared to PEO IWS's existing architecture. The authors decomposed the proposed architecture from the highest level, the conceptual design, to the lowest level, the ASCMD simulation model. This section closes the report by discussing the comparison findings and conclusions, final overview of the proposed architecture, and outstanding issues to be reviewed by the project's stakeholders and decision-makers.

A. SIMULATION CONCLUSIONS

Threat missiles were able to leak through in both attack scenarios. The architecture changes alone will not solve the Navy's need to improve its defenses against cruise missile attack. There is a statistically significant improvement in the revised architecture's performance when compared to PEO IWS's architecture in the simulation. Furthermore, the models whose loops returned to the Decide or Act sub-model performed substantially better in both attack scenarios. Although no conclusion could easily be drawn between the Decide and Act sub-models based on the measures of performance used in this simulation, some control aspects of the engagement are necessarily lost if the loop bypasses the Decide sub-model. Since there is not a distinguishable difference between the two locations based on the measures of performance, the authors would not recommend the removal of the human in the loop at the Decide sub-model.

The authors conclude that the revised architecture should formally include a loop and that the loop should bring the information back to the Composite Warfare Commander in the decide portion of the OODA loop.

B. OUTSTANDING ISSUES

The focus of this paper is the development of a conceptual Anti-Ship Cruise Missile Defense (ASCMD) system that adheres to and integrates the FORCEnet information architectural framework with the technical requirements of the PEO IWS OA functional domain model. Research and analysis efforts substantiated that Open Architecture provides the right venue for the development and implementation of FORCEnet design concepts. These design concepts make the implementation of a CMD Integrated Fire Control and advanced Command and Control command structure a near-term reality. PEO IWS, chair of the open architecture enterprise team, continues to promulgate OA policies and standards, as well as planning and implementation of OA into the next generation of cruisers, destroyers, aircraft carriers, and submarines. There are many challenges to overcome, many risks that need to be identified, managed, and mitigated as early as possible during the system acquisition phase. To realize the full potential of this new architecture, FORCEnet will need to be an operational construct supporting all U. S. Navy commands prior to implementation.

The goal of the conceptual ASCMD architecture is to fuse time-dependent information from different systems seamlessly, with minimal erroneous data, and be able to distribute the information in real-time to the decision makers and to the Strike Group participating units using push-pull technology. This means that information will be distributed at the right time and to the right participating units. Open architecture design leverages ease of technology insertion and compatibility with other members of the same distributed network.

There are areas that need further consideration in the application of OA, IFC and FORCEnet design concepts. An effective CMD system design requires the achievement of the smallest possible reaction time from threat detection to weapons firing. FORCEnet and OA will expedite data flow due to support common services and reduce human interaction in the kill chain. The sensor-to-shooter kill chain can be hastened by introducing automated processes and computational intelligence, using fuzzy logic and neural networks, which in turn will curtail time lost due to organic intervention. Unfortunately, the neural network technology is not sufficiently mature, but recent research and development with neural networks show promise for the design as well as other adaptive technologies, which can increase system automation and reduce reaction time.

The FORCEnet communications infrastructure requires sTable data links that are geographically distributed and fully interoperable across different platforms. These data links support the implementation of a CMD and IFC system that addresses the challenges associated with the various cruise missile threats. Battlegroup network connectivity needs to be adapTable to the capability inherent within their participating units. Levels of connectivity need to integrate current communications and data exchange networks.

The PEO IWS 7 OA functional architecture model requires a level of integration that is not currently available in US Navy platforms. A new level of networks and combat system interoperability is essential to the implementation of all the Integrated Fire Control operational concepts previously addressed in this paper. Creation of CONOPS for this architecture will require inputs from DoD stakeholders, joint and coalition forces, and private industry partners. Deployment of assets within the Strike Group along and near the threat axis, command structure, and other battle considerations will need to be addressed to use the inherent capability of this new architecture. The CONOPS that are currently in place for ASCMD can be modified to include the integrated fire control scenarios. It is also recommended that an Integrated Product Team (IPT) be formed, consisting of naval combat system subject matter experts, who will integrate the PEO IWS 7 OA functional architecture EXCOMM and Command and Control functions.

The combat system architecture addressed by this paper was scoped to USN assets alone. There are numerous applications that can include joint forces, which will extend both the sensor net, best weapon, and shooter selection. The Army has researched over the horizon cruise missile defense systems and conducted successful cruise missile defense experiments like Mountain Top (Zinger, Krill, 1997) with both Army and Navy units. Allied considerations must also take place, especially with countries that employ similar detection and weapons systems. Integrating these global assets will be a force multiplier enabling the strike group commander to coordinate multiple operations simultaneously.

OA allows for rapid insertion of the latest technology with minimal impact or redesign of the system. Provisions should be made for over-the-air computer program upgrades to a deployed strike group, where all units will be upgradeable while still operating in the FORCEnet construct. This will help avoid backwards compatibility issues between units having received upgrades and those without them. There needs to be a strategic deployment process where the installations and upgrades are minimal but sufficient for the unit and strike group to perform their mission and current operations.

Risk is a component of any design implementation and the architecture's main concern addresses the current lack of an operating FORCEnet. Granted, the groundwork has been laid and processes and common services are being developed, but FORCEnet has not deployed and there is no firm operational start date. Navy information and data flow superiority requires a proven and reliable network, which FORCEnet will achieve, once implemented. When considering possible future technology and capabilities, risk is an inherent part of the design phase. Assumptions have been made regarding bandwidth both overhead and off ship. There is also risk in software development that integrates existing systems and the timely delivery, installation, and testing of this software to the strike group prior to deployment. Additional risks remain in the following areas:

- Ability to demonstrate joint interoperability.
- Achievement of high track data rates via secure networks to address all the IFC scenarios.
- Configuration control of Commercial Off The Shelf (COTS) equipment.
- Constant evolution of computer programs programming languages.
- Constant evolution of standards.
- Establishment of an organic and strike group training capability to maintain a high- level of operational readiness. Training capability needs to be part of the original design, not added as a separate system.
- Evolution of the threat
- Fusion of sensor data for air, surface, and undersea situational awareness
- How to implement push-pull technology for sharing information within the strike group to alleviate network message traffic and improve delivery of critical information to the end user.
- Human system integration (HSI) certification done as early as possible.
- Information assurance: user and information authentication.
- Integration of systems that are not fully OA compliant.
- Integration of EXCOMM and Command and Control systems.

- Introduction of disruptive technologies.
- Method of delivering critical information: multicast, unicast, anycast, and broadcast.
- Timely implementation of IPv-6 and associated capabilities.
- Creation of a single integrated picture for air, surface, and undersea assets.
- The FORCEnet vision is dependent on technology developing at least at its present rate.
- GIG and FORCEnet architectures are somewhat contingent on what the other services and allied forces bring to the Table. Their commitment in terms of present and future funding and political involvement as well as their rate of technology development will shape these architectures.

Future research efforts to be considered include the following:

- A CONOPS vision based on degraded network architecture from external intrusion.
- Parallel research and development in the areas of neural network and fuzzy logic, as well as neural fuzzy networks.
- A phased development program for FORCEnet, possibly spiral in nature, which takes into consideration advanced technology with concrete deliverable functionality in phased or block increments.
- Continued investment in research and development programs to ensure the required technology advancements needed for FORCEnet development are successful**.**

C. FINAL OVERVIEW OF PROPOSED ARCHITECTURE

1. Requirements Overview Model

The functional decomposition of the proposed ASCMD functional architecture is depicted in the Requirements Overview model in Figure 98. The Requirements Overview model summarizes all of the proposed architecture’s capabilities and performance (Hatley, Hruschka, Pirbhai, 2000). The decomposition began with a high-level conceptual framework, followed by a high-level Value System Design and Functional Flow Block Diagram of the architecture. These high-level steps were followed by a detailed breakdown of the architecture that included class diagrams, use cases, sequence diagrams, control flow and data flow diagrams, and finally a software simulation. The creation of the class diagrams led into the use cases and sequence diagrams, which in turn led into the development of the control flow and data flow diagrams. The process was not serialized; all diagrams were created iteratively and in parallel in some cases.

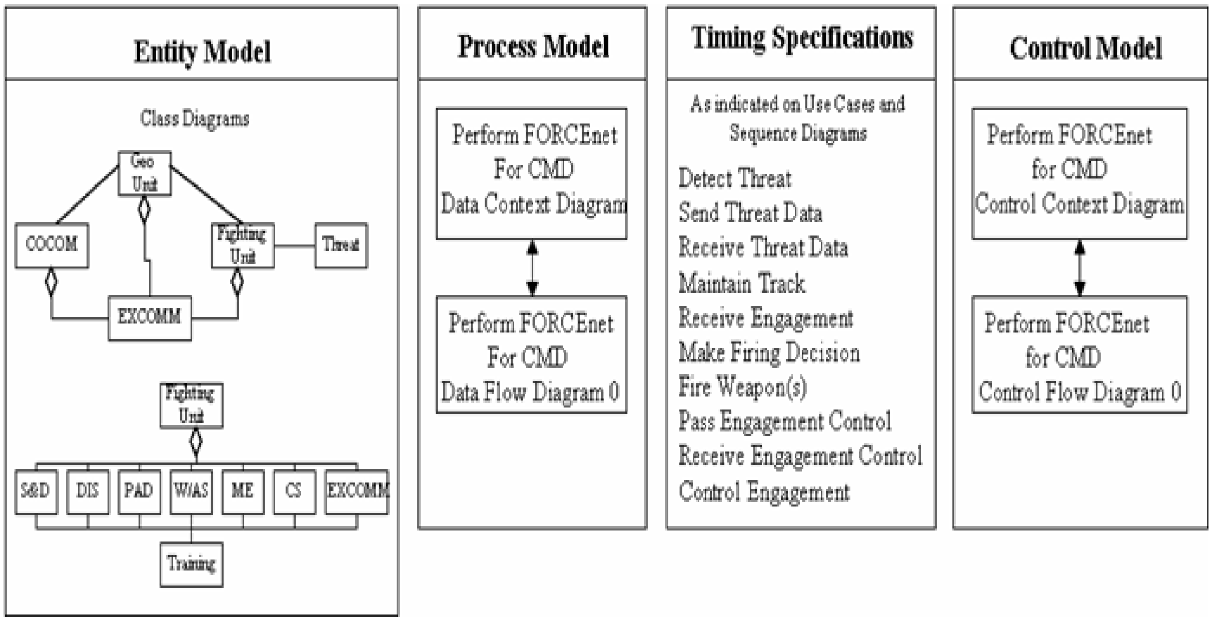


Figure 98. Requirements Overview model of proposed ASCMD functional architecture.
This model captures the architecture’s required capabilities and performance.

2. Changes to Module Functionalities

Figures 99 through 107 capture the changes made to each individual module's functionalities. Each module within the existing PEO IWS 7 architecture was reviewed and changes made in accordance with the analysis of key capabilities and functional decomposition performed in the Design and Analysis section. It is important to note that the flows of the "Provided Data" and "Consumed Data," depicted as red and yellow arrows within PEO IWS's model, were not changed. The authors agree with the way these data exchanges are represented. Changes to the Candidate OA Platform-Unique Function/Application modules are in yellow font; changes to the Candidate OA Common Function/Application modules are in blue font.

Additionally, the proposed architecture is tightly coupled to the simulation and the modeling efforts undertaken. The OODA loop functions used in the simulation represent the proposed architecture functions depicted in Figure 10. Table 19 shows the correlation of the simulation modules to the proposed architecture functions. The function *Search and Detect* is represented in the simulation by the Observe module. The function *Data Information Services* is represented by the Orient module of the simulation. In the simulation the Decide module represents the function *Planning, Assessment, and Decision*. The Act part of the simulation is distributed among the functions *Weapon Asset Services* and *Mission Execution.* Finally, the lines connecting each block in the simulation together are represented by the functions *EXCOMM* and *Common services*. In this way, the proposed changes capture all of the aspects of the simulation and the modeling efforts.

Simulation Module name	**Architecture Function Name**
Observe / Re-observe	Search and Detect
Orient	Data Information Services
Decide	Planning, Assessment, and Decision
Act	Weapon Asset Services
Act	Mission Execution
Connecting Lines	EXCOMM
Connecting Lines	Common Services

Table 19. Mapping the simulation to the architecture function
This shows the correlation between the simulation and the architecture

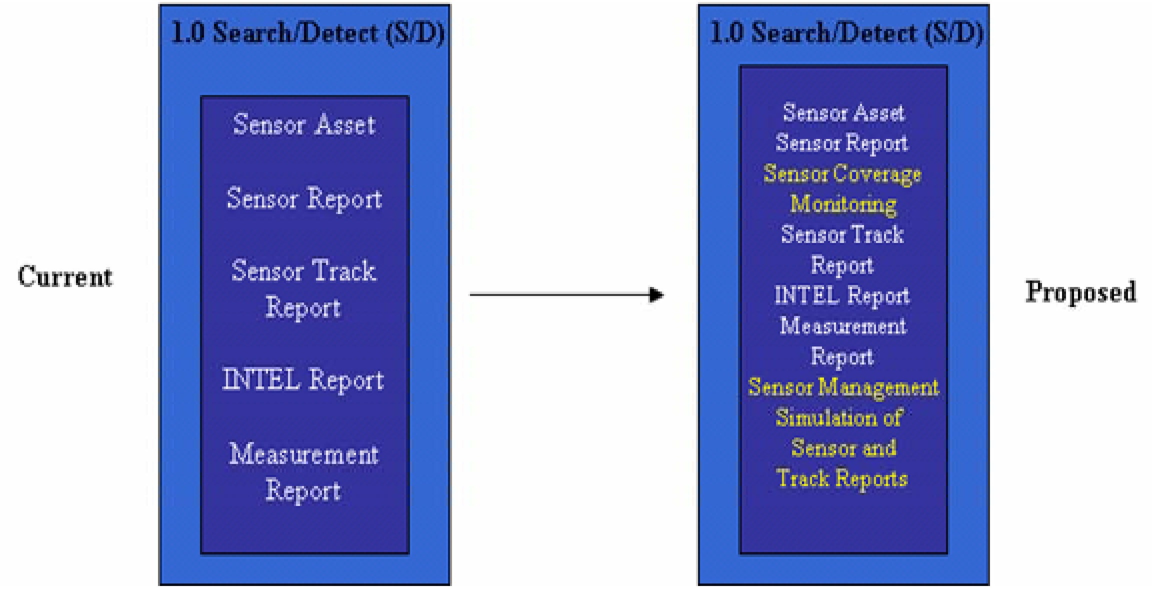


Figure 99. Function additions to Search/Detect module. Added functions are Sensor Coverage Monitoring, Sensor Management, and Simulation of Sensor and Track Reports.

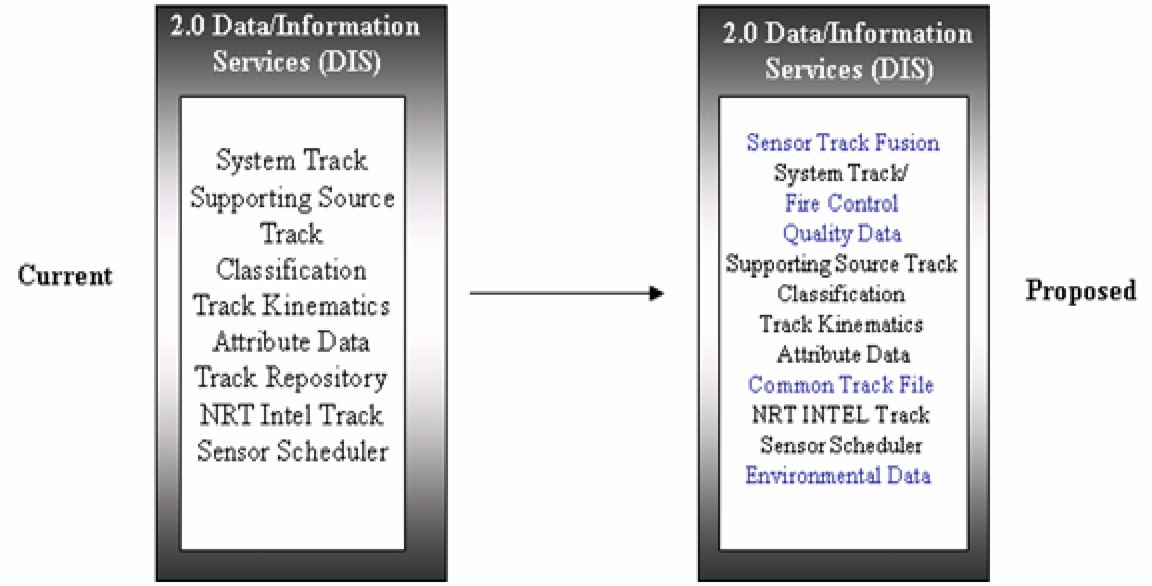


Figure 100. Function additions to Data/Information Services module. Added functions are Sensor Track Fusion, Fire Control Quality Data, Common Track File, and Environmental Data.

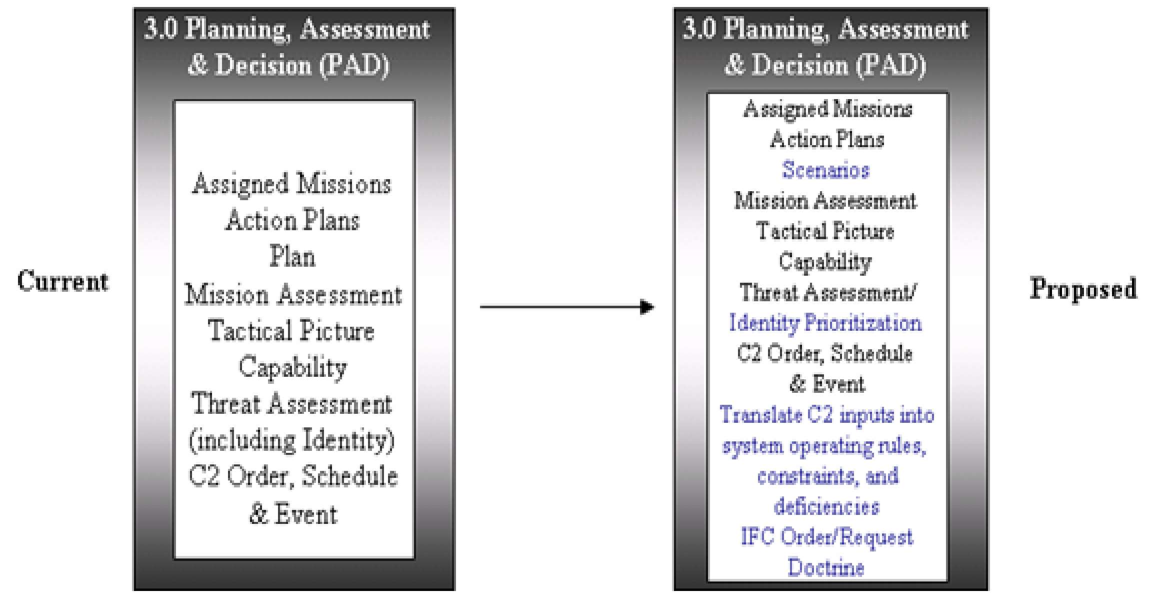

Current
3.0 Planning, Assessment & Decision (PAD)
Assigned Missions
Action Plans
Plan
Mission Assessment
Tactical Picture
Capability
Threat Assessment (including Identity)
C2 Order, Schedule & Event
3.0 Planning, Assessment & Decision (PAD)
Assigned Missions
Action Plans
Scenarios
Mission Assessment
Tactical Picture
Capability
Threat Assessment/
Identity Prioritization
C2 Order, Schedule & Event
Translate C2 inputs into system operating rules, constraints, and deficiencies
IFC Order/Request
Doctrine
Proposed

Figure 101. Function additions to Planning, Assessment & Decision module. Added functions are Scenarios; Identity Prioritization; Translate C2 Inputs into System Operating Rules, Constraints, and Deficiencies; and IFC/Request Doctrine.

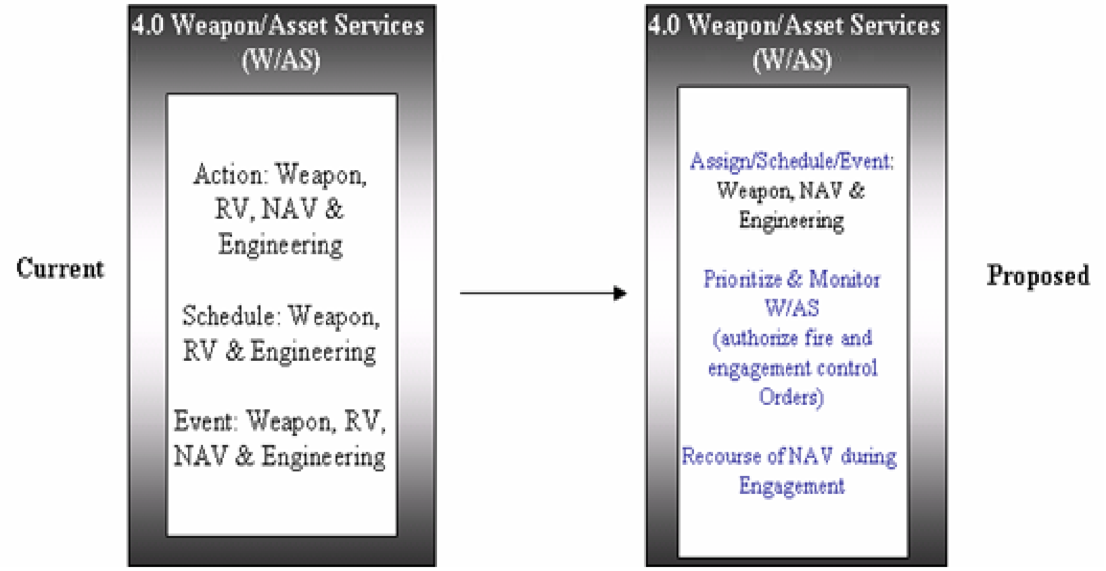

Current
4.0 Weapon/Asset Services (W/AS)
Action: Weapon, RV, NAV & Engineering
Schedule: Weapon, RV & Engineering
Event: Weapon, RV, NAV & Engineering
4.0 Weapon/Asset Services (W/AS)
Assign/Schedule/Event: Weapon, NAV & Engineering
Prioritize & Monitor W/AS (authorize fire and engagement control Orders)
Recourse of NAV during Engagement
Proposed

Figure 102. Additions to Weapon/Asset Services module. Added functions are combination of Assign/Schedule/Event for Weapons, Navigation, and Engineering; Prioritize and Monitor W/AS through Authorize Fire and Engagement Control Orders; and Recourse of Navigation during Engagement.

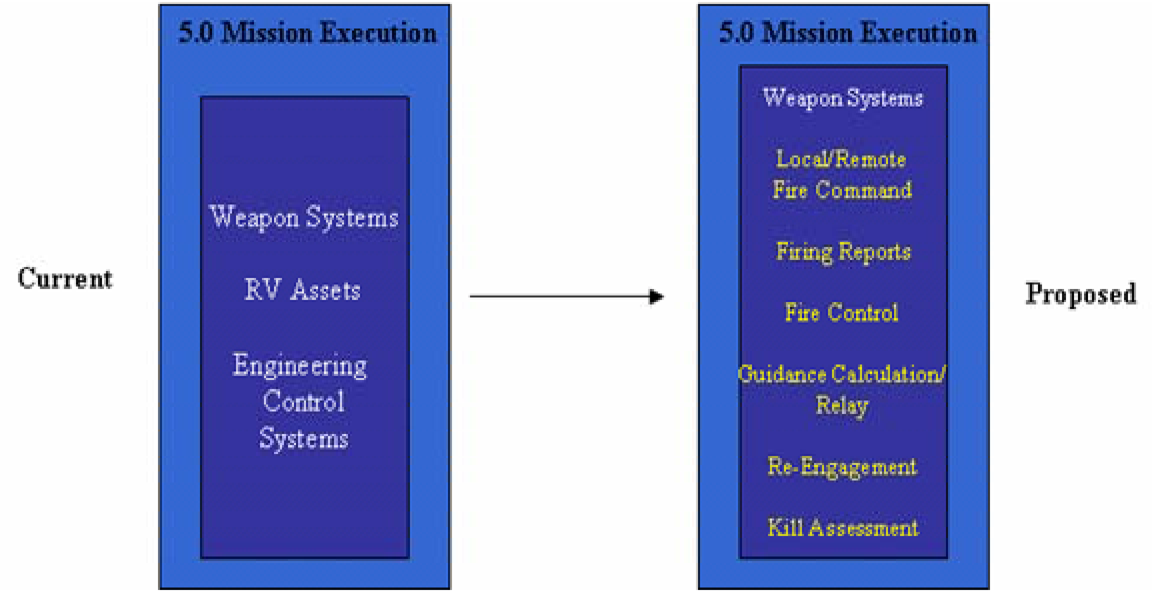


Figure 103. Function additions to Mission Execution module.
Added functions are Local/Remote Fire Command, Firing Reports, Fire Control, Guidance Calculation/Relay, Re-Engagement, and Kill Assessment.

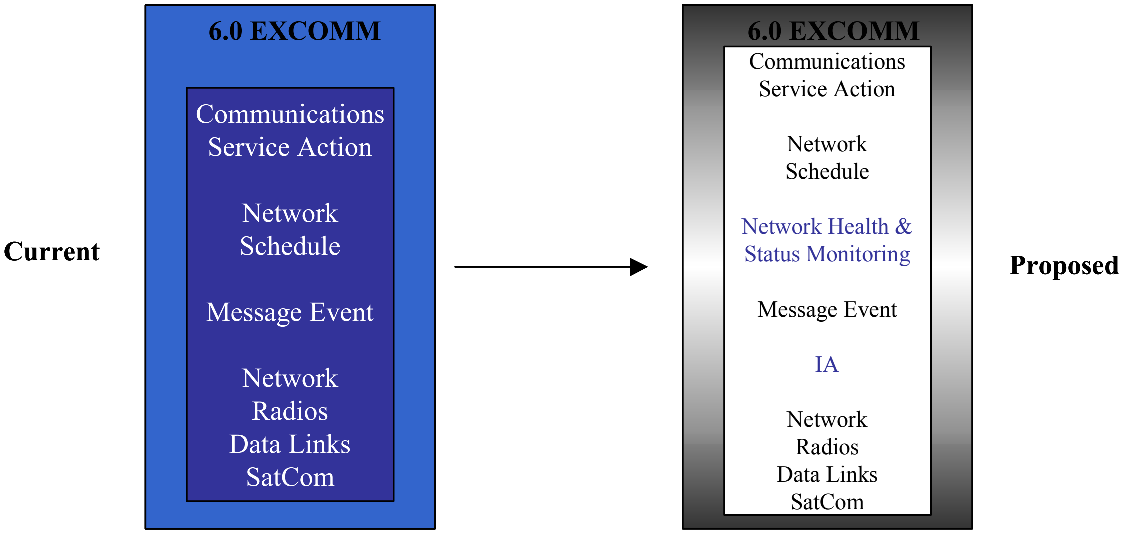

Current
6.0 EXCOMM
Communications Service Action
Network Schedule
Message Event
Network Radios Data Links SatCom
6.0 EXCOMM
Communications Service Action
Network Schedule
Network Health & Status Monitoring
Message Event
IA
Network Radios Data Links SatCom
Proposed

Figure 104. Function additions to EXCOMM module.
Added functions are Network Health/Status Monitoring and IA.

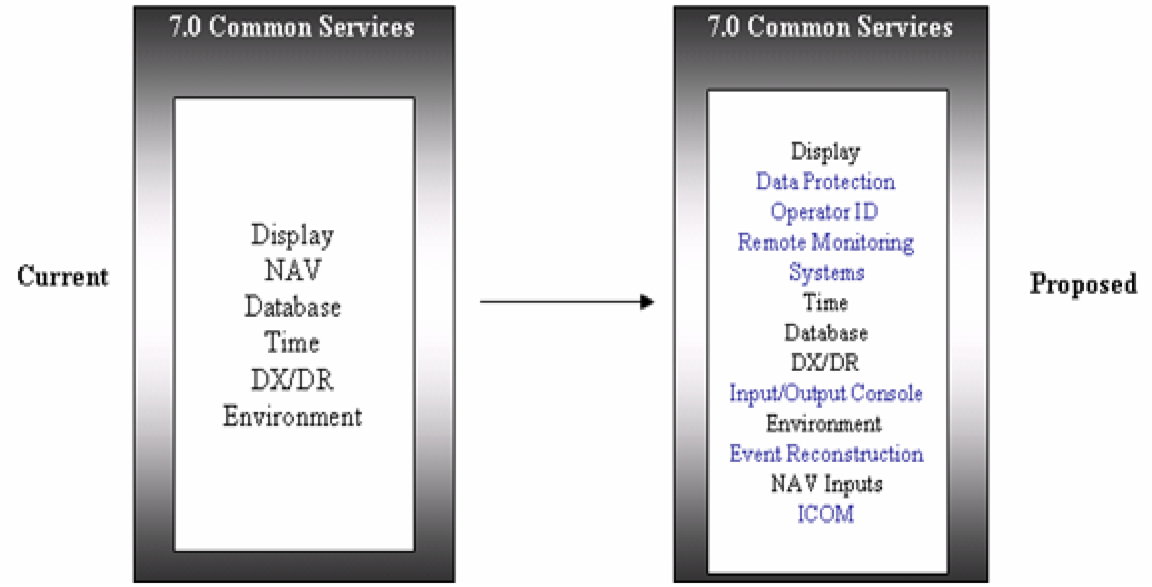


Figure 105. Function additions to Common Services module.
Added functions are Data Protection, Operator ID, Remote Monitoring Systems, Input/Output Control Console, Event Reconstruction, and Internal Communications (ICOM).

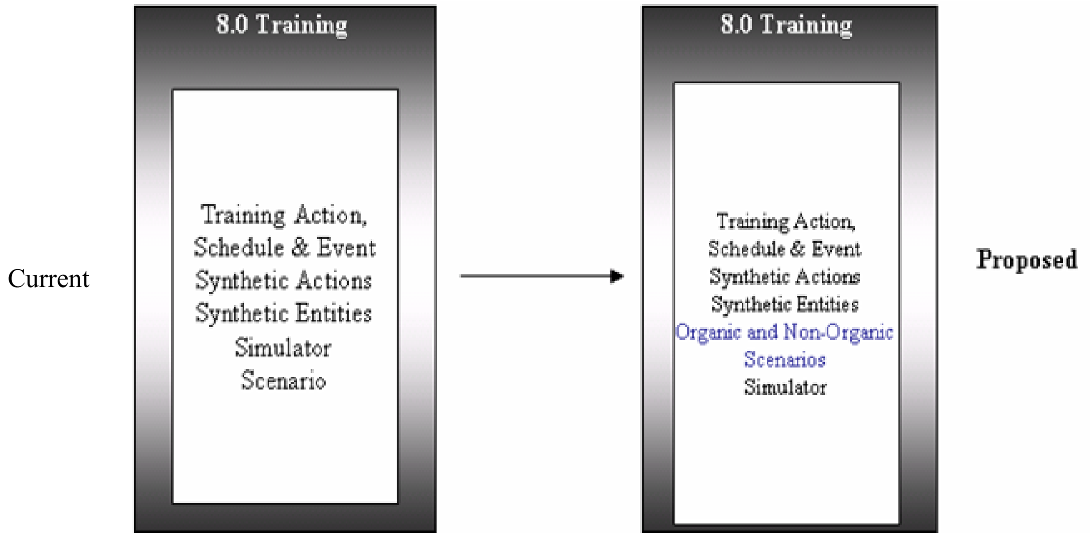


Figure 106. Function changes to Training module.
Scenarios were modified to cover organic and non-organic events.

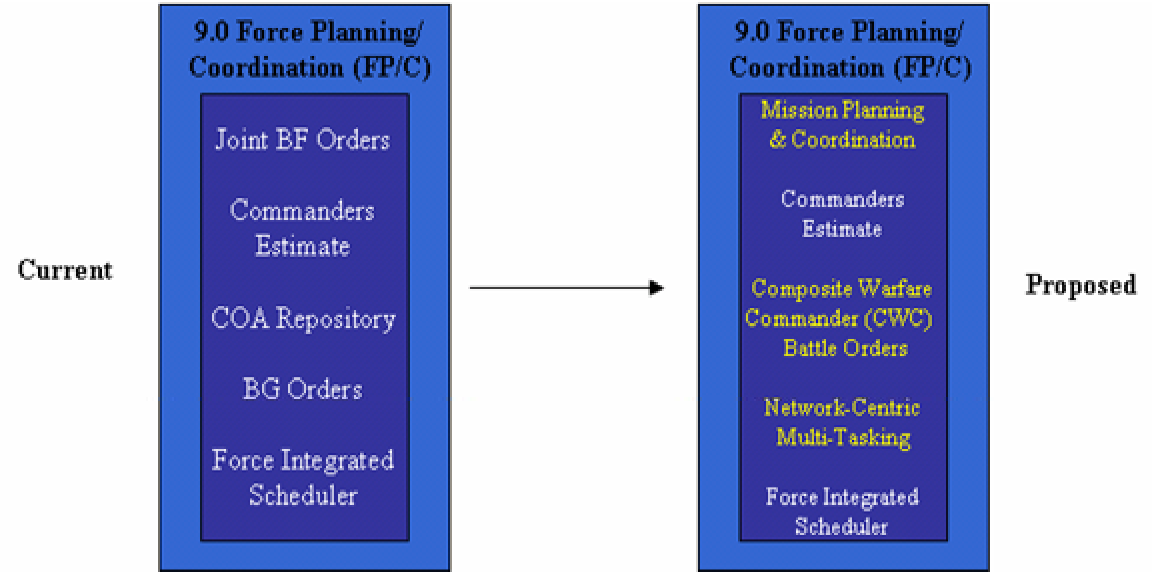

Current
9.0 Force Planning/ Coordination (FP/C)
Joint BF Orders
Commanders Estimate
COA Repository
BG Orders
Force Integrated Scheduler
9.0 Force Planning/ Coordination (FP/C)
Mission Planning & Coordination
Commanders Estimate
Composite Warfare Commander (CWC) Battle Orders
Network-Centric Multi-Tasking
Force Integrated Scheduler
Proposed

Figure 107. Function additions to Training module.
Added functions are Mission Planning and Coordination; Distributed Resource Management and Readiness Assessment; Health and Status of Units, Sensors and Weapons; Network-Centric Multi-Tasking; Processor Health and Status Assessment; and Integrated Fire Control.

The modeling, simulation, and analysis reflected in this study show that the revised open architecture model when implemented will provide the Navy with a capability that will reduce its vulnerability to cruise missile attacks. Additional and continuing research will reduce risk while improving effectiveness and performance.

APPENDIX A

STATEMENT OF WORK

Open Architecture as an Enabler for FORCEnet

Task 1: Cruise Missile Defense

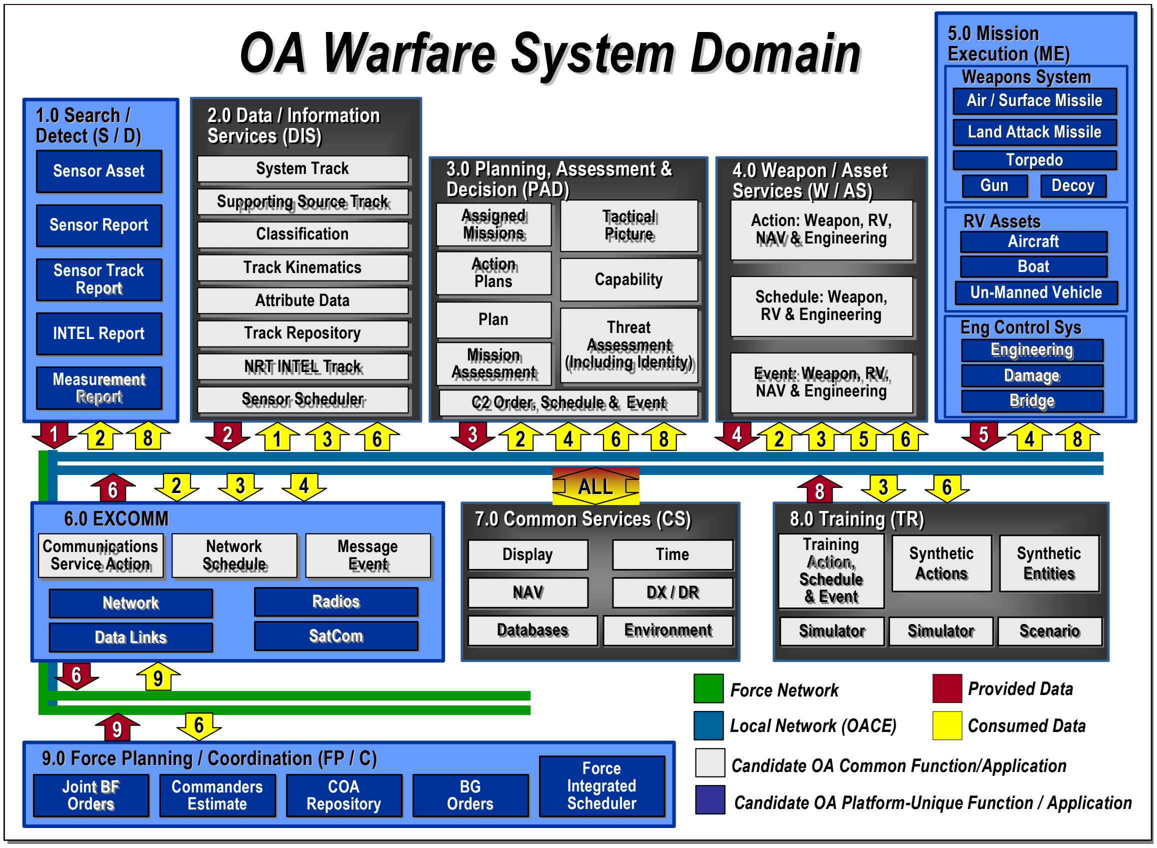
Scope:

This task investigates the role of the Open Architecture (OA) Functional Domain Model in FORCEnet (Fn) and its application to cruise missile defense. The specific focus will be on integrated fire control and the shipboard component.

The following extract from Chapter 5 of FORCEnet Implementation Strategy provides key insights into the issues of FORCEnet Architecture. The material in this paper is extracted from Chapter 5 of FORCEnet Implementation Strategy (http://www.nap.edu/catalog/11456.htm).

If FORCEnet is to be the architectural framework for naval warfare in the information age, it must deliver performance, information assurance, and quality of-service guarantees unprecedented in a system with the nodal diversity evidenced in the joint force. This challenge is best met incrementally so that existing capability is not degraded nor information security ever compromised. The design and implementation of complex systems for purposes of warfighting require a dedicated core of warfighters and system engineers trained in the art of operations analysis. Together, warfighters and engineers make decisions about when and how to introduce new capabilities as technologies and operational concepts evolve in independent but integrated spirals.

The FORCEnet information architecture should be thought of as a boundary between layers of functionality that is held invariant (over long periods), thus allowing developments to proceed independently on all sides of the boundary. In the committee's view, architecting FORCEnet is the process of defining thin waists, or boundaries, that are invariant and, when coupled with selected industrial standards and throttled with a network control system, would enable FORCEnet to evolve with advances in technology. The boundaries standardize the interfaces between the functions common to all warfare



Extracted from: **Open Architecture, The Critical Network Centric Warfare Enabler**

First Edition, March 18, 2004

Captain Richard T. Rushton, USN, et al.

FORCEnet and the fighting units and command-and-control structure that it supports are all subsystems of a joint battle force. Systems engineering is a process for allocating functionality to subsystems that are bounded by system architecture so that the probability of mission success is increased within available resources. A battle force performs three major functions: it manages battle, dominates battlespace, and sustains control over the battlespace over time. FORCEnet functionality is a subset of battle force functionality that can contribute to battle management, battlespace dominance, and sustainability. FORCEnet cost and contribution to battle management, battlespace dominance, and sustainability should provide a basis for implementation decisions. As a subsystem, FORCEnet must interface seamlessly with the remainder of the force while increasing the probability of mission success more than alternative investments. Understanding and defining the interfaces between what is in the FORCEnet subsystem and what is outside of it will be an ongoing process. This top-down view of FORCEnet, together with the bottom-up work that is being done at the information architecture boundaries, is necessary to explain and quantify the warfighting value.

Selected Issues:

The following issues extracted from Chapter 5 capture some of the underlying research goals of this project:

- The process and tools for translating FORCEnet operational concepts into products, services, and warfighting capabilities have yet to be fully developed. Systems engineering is a process for allocating functionality to subsystems that are bounded by a system architecture.

- The number of unique interfaces that must be maintained need to be carefully selected and kept to an absolute minimum, or evolution will be hindered by expensive and lengthy integration and testing. One way to do this is to require that systems must partition common functions in a common way.

- There has been little attempt to characterize how FORCEnet will function in terms of network management, data flow, traffic control, nodal performance, or data access. This information is required to engineer the FORCEnet network management system.

Technical Requirements:

The FORCEnet functional architecture is based on two operationally oriented scenarios selected to validate the FORCEnet architecture: (1) time-critical targeting employing persistent sensors and (2) cruise missile defense Mission Capability Packages (MCP's)(*these are not the same MCP's that the Warfare Integration Unit under the DCNO for Warfare Requirements and Programs (N70) uses for program assessment*). The focus of this engineering and analysis effort is on cruise missile defense with the goal to explore and develop a conceptual model that marries the operational and system

Fn architecture requirements with the technical requirements of the OA Functional Domain Model as required to support the concept. This work will be based upon the use of the Integrated Fire Control scenarios from references 7 and 8 to elaborate upon the basic mission capability requirements of cruise missile defense.

STATEMENT OF WORK:

Characterization of the Problem Space: the identification of current system and legacy deficiencies as well as constraints inherent in the operational environment in order to characterize, understand and bound the problem space. The project team will translate relevant operational imperatives into system engineering structures (concepts, functions, requirements, solutions) necessary to develop the concept. A key step in this process is to evaluate the "correctness of the OA Warfare System Domain Model shown on page 2.

Design Principles: the formulation of principles for the design and architecting of OA and Fn (IFC) capabilities. The project team will formulate design principles to serve as guidelines for the development of system solutions. Design principles will consider known limitations and constraints of the operational environment such as communication challenges, unreliability, ad hoc mobile networks, limited bandwidth, and operator interaction.

Conceptual Design: the development of a vision, architecture, and conceptual framework that addresses the problem space and is based on the design principles for a distributed system of automated decision aids for managing warfare resources for collaborative operations. The project team will formulate a conceptual design of the required system within the boundaries of Fn and OA.

Functional Representation And Decomposition: the representation of system concepts through functional description and decomposition as well as system architecting and simulation. Develop representations, models, and methods to express automated resource collaboration concepts and solutions in the context of the Fn/OA architecture and domains. The project team will develop a system model to evaluate the performance of the proposed architecture.

Analysis of Key Capabilities: the identification and evaluation of technologies and research areas key to the Fn/OA concept. Technology areas that will be researched and analyzed include:

- Data fusion techniques and algorithms
- Resource management scheduling and optimization methods
- Weapon and sensor management
- Engagement functionality, initialization, and control
- Situation prediction and wargaming
- Tactical planning and battle management
- Opportunities for application of fuzzy logic and neural networks
- Allocation of tasking to people and/or software

Documentation: The results of tasks 1-5 will be documented in accordance with the NPS MSSE (DL) Project Guide Requirements as modified by agreement with the project advisor.

APPENDIX B

SIMULATION PARAMETERS AND DISTRIBUTIONS AND CALCULATIONS

Number	NAME	Value	Type
1	Cruiser 2 ESSM Part B Flyout	(Current.Location - (30/COS(ABS(Threat.Angle.-2.356)))) / (Threat.Speed +1.18)	Variable
2	Cruiser1 ESSM Available	8	Variable
3	Cruiser1 ESSM Part A Flyout	25 * (TAN(ABS(Threat.Angle.-5.5))) /1.18	Variable
4	Cruiser1 ESSM Part B Flyout	(Current.Location - (25/COS(ABS(Threat.Angle.-5.5)))) / (Threat.Speed +1.18)	Variable
5	Cruiser1 RAM Available	42	Variable
7	Cruiser1 RAM Part B Flyout	(Current.Location - (25/COS(ABS(Threat.Angle.-5.5)))) / (Threat.Speed +.680)	Variable
8	Cruiser1 SM3 Available	8	Variable
9	Cruiser1 SM3 Part A Flyout	25 * (TAN(ABS(Threat.Angle.-5.5))) /2.67	Variable
10	Cruiser1 SM3 Part B Flyout	(Current.Location - (25/COS(ABS(Threat.Angle.-5.5)))) / (Threat.Speed +2.67)	Variable
11	Cruiser2 ESSM Available	8	Variable
12	Cruiser2 ESSM Part A Flyout	30 * (TAN(ABS(Threat.Angle.-2.356))) /1.18	Variable
13	Cruiser2 RAM Available	42	Variable
14	Cruiser2 RAM Part A Flyout	30 * (TAN(ABS(Threat.Angle.-2.356))) /.680	Variable
15	Cruiser2 RAM Part B Flyout	(Current.Location - (30/COS(ABS(Threat.Angle.-2.356)))) / (Threat.Speed +.680)	Variable
16	Cruiser2 SM3 Available	8	Variable
17	Cruiser2 SM3 Part B Flyout	(Current.Location - (30/COS(ABS(Threat.Angle.-2.356)))) / (Threat.Speed +2.67)	Variable
18	Cruiser2 SM3 Part A Flyout	30* (TAN(ABS(Threat.Angle.-2.356))) /2.67	Variable
19	Current.Location	Set originally as 50 to prevent an undefinded number	Variable
20	CVN Ram Available	42	Variable
21	DDG 1 ESSM Part A Flyout	10 * (TAN(ABS(Threat.Angle.-2.356))) /1.18	Variable
22	DDG1 ESSM Available	16	Variable
23	DDG1 ESSM Part B Flyout	(Current.Location - (10/COS(ABS(Threat.Angle.-2.356)))) / (Threat.Speed +1.18)	Variable
24	DDG1 RAM Available	42	Variable
25	DDG1 RAM Part A Flyout Time	10 * (TAN(ABS(Threat.Angle.-2.356))) /.680	Variable

Number	NAME	Value	Type
27	DDG1 SM3 Available	16	Variable
28	DDG1 SM3 Part A Flyout	10 * (TAN(ABS(Threat.Angle.-2.356))) /2.67	Variable
29	DDG1 SM3 Part B Flyout	(Current.Location - (10/COS(ABS(Threat.Angle.-2.356)))) / (Threat.Speed +2.67)	Variable
30	DDG2 ESSM Avaiable	16	Variable
31	DDG2 ESSM Part A Flyout	15 * (TAN(ABS(Threat.Angle.))) /1.18	Variable
32	DDG2 ESSM Part B Fllyout	(Current.Location - (15/COS(ABS(Threat.Angle.)))) / (Threat.Speed +1.18)	Variable
33	DDG2 RAM Available	42	Variable
34	DDG2 RAM Part A Flyout	15 * (TAN(ABS(Threat.Angle.))) /.680	Variable
35	DDG2 RAM Part B Flyout	(Current.Location - (15/COS(ABS(Threat.Angle.)))) / (Threat.Speed +.680)	Variable
36	DDG2 SM3 Available	16	Variable
37	DDG2 SM3 Part A Flyout	15* (TAN(ABS(Threat.Angle.))) /2.67	Variable
38	DDG2 SM3 Part B Flyout	(Current.Location - (15/COS(ABS(Threat.Angle.)))) / (Threat.Speed +2.67)	Variable
39	ESSM Max Launch Range	56+(Threat.Speed * 40)	Variable
40	Frigate ESSM Available	8	Variable
41	Frigate ESSM Part A Flyout	200 * (TAN(ABS(Threat.Angle.))) /1.18	Variable
42	Frigate ESSM Part B Flyout	(Current.Location - (200/COS(ABS(Threat.Angle.)))) / (Threat.Speed +1.18)	Variable
43	Frigate RAM Available	42	Variable
44	Frigate RAM Part A Flyout	200 * (TAN(ABS(Threat.Angle.))) /.680	Variable
45	Frigate RAM Part B Flyout	(Current.Location - (200/COS(ABS(Threat.Angle.)))) / (Threat.Speed +.680)	Variable
46	Frigate SM3 Part A Flyout	200 * (TAN(ABS(Threat.Angle.))) /2.67	Variable
47	Frigate SM3 Part B Flyout	(Current.Location - (200/COS(ABS(Threat.Angle.)))) / (Threat.Speed +2.67)	Variable
48	Frigatge SM3 Available	0	Variable
49	Future Threat To Cruiser 1	SQRT(((Future X Location-17)*(Future X Location-17))+((Future Y Location+17)*(Future Y Location+17)))	Variable
50	Future Threat To Cruiser 2	SQRT(((Future X Location+21)*(Future X Location+21))+((Future Y Location-21)*(Future Y Location-21)))	Variable
51	Future Threat to DDG1	SQRT(((Future X Location+7)*(Future X Location+7))+((Future Y Location-7)*(Future Y Location-7)))	Variable
52	Future Threat to DDG2	SQRT(((Future X Location-0)*(Future X Location-0))+((Future Y Location-15)*(Future Y Location-15)))	Variable
53	Future Threat to Frigate	SQRT(((Future X Location-0)*(Future X Location-0))+((Future Y Location-200)*(Future Y Location-200)))	Variable
54	Future X Location	COS(Threat.Angle.) * Future.Location.Of.Threat	Variable
55	Future Y Location	SIN(Threat.Angle.) * Current.Location	Variable

Number	NAME	Value	Type
56	Future.Location.Of.Threat	Initial.Range-(Threat.Speed * (TNOW-Entity.CreateTime+10))	Variable
57	Ram Max Launch Range	17+(Threat.Speed * 45)	Variable
58	SM3 Max Launch Range	500+(Threat.Speed * 175)	Variable
59	Threat.To.Cruiser1	SQRT(((X.Location-17)*(X.Location-17))+((Y.Location+17)*(Y.Location+17)))	Variable
60	Threat.To.Cruiser2	SQRT(((X.Location+21)*(X.Location+21))+((Y.Location-21)*(Y.Location-21)))	Variable
61	Threat.To.DDG1	SQRT(((X.Location+7)*(X.Location+7))+((Y.Location-7)*(Y.Location-7)))	Variable
62	Threat.To.DDG2	SQRT(((X.Location-0)*(X.Location-0))+((Y.Location-15)*(Y.Location-15)))	Variable
63	Threat.To.Frigate	SQRT(((X.Location-0)*(X.Location-0))+((Y.Location-200)*(Y.Location-200)))	Variable
64	X.Location	COS(Threat.Angle.) * Current.Location	Variable
65	Y.Location	SIN(Threat.Angle.) * Current.Location	Variable
66	Threat.Speed	0.6	Attribute
67	Initial.Range	UNIF(250,1000,102)	Attribute
68	Threat.Angle	UNIF(0, 6.28,103)	Attribute
69	Detect.Time	Entity.CreateTime	Attribute
70	Locally Locate and ID	TRIA(1,5,60,100)	Delay
71	Report to FORCENet	TRIA(15 , 20 , 200,101)	Delay
72	IA5	TRIA(1 , 3 , 30)	Delay
73	IA Penalty Box5	TRIA(10 , 15 , 30)	Delay
74	Update Common Control Picture	TRIA(3,5,15,104)	Delay
75	IFC	TRIA(2,6,30)	Queue
76	IA	TRIA(1 , 3 , 30)	Delay
77	IA Penalty Box	TRIA(10 , 15 , 30)	Delay
78	Electronic Combat	TRIA(20 , 60 , 240)	Delay
79	Decison To Shoot From The Each Platform	TRIA(0.5 , 1 , 2)	Delay
80	Platform	TRIA(0.1 , .5 , 2)	Delay
81	RAM Wait For In Range	(Current.Location-Ram Max Launch Range) / Threat.Speed	Delay
82	RAM Spin Up	TRIA(1 , 3 , 10)	Delay
83	ESSM Wait For In Range	(Threat.To.Cruiser1-ESSM Max Launch Range) / Threat.Speed	Delay
84	ESSM Spin Up	TRIA(3 , 4 , 10)	Delay
85	SM3 Wait For In Range	(Threat.To.Cruiser1-SM3 Max Launch Range) / Threat.Speed	Delay
86	SM3 Spin Up	TRIA(3 , 4 , 10)	Delay
87	Evaluate Kill	TRIA(2,5,10)	Delay
88	IA.Good	99% True	Decision
89	Real.Not.Real.Threat	50% True	Decision
90	EWSuccess	80% True	Decision
91	Prox	Current.Location > 2 True	Decision

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX C

LIST OF ACRONYMS

ABMA	Automated Battle Management Aids
AC	Air Conditioning
ACTD	Advanced Concept Technology Demonstrator
ADC	Air Defense Commander
AEW	Air Expeditionary Wing
AFD	Architecture Flow Diagram
AID	Architecture Interconnect Diagram
ANSI	American National Standards Institute
A_o	Operational Availability
AOI	Area of Interest
AOR	Area of Operation
ASCM	Anti-Ship Cruise Missile
ASCMD	Anti-Ship Cruise Missile Defense
BDA	Battle Damage Assessment
BG	Battlegroup
C2	Command and Control
C4I	Command, Control, Communications, Computers, & Intelligence
C&C	Command and Control
CAC	Common Access Card
CAP	Combat Air Patrol
CASREP	Casualty Report
CAT	Category
CAW	Carrier Air Wing
CCD	Control Context Diagram
CEC	Cooperative Engagement Capability
CENTRIXS	Combined ENTerprise Regional Information eXchange System
CFD	Control Flow Diagram

CG	Guided Missile Cruiser
CI	Configuration Item
CIC	Combat Information Center
CID	Combat Identification
CIWS	Close-In Weapon System
CM	Cruise Missile
CMD	Cruise Missile Defense
COA	Course of Action
CONOPS	Concept of Operations
COP	Common/Composite Operating Picture
CORBA	Common Object Request Broker Architecture
COTS	Commercial off the Shelf
CPA	Closest Point of Approach
CRS	Congressional Research Service
CS	Common Services
CSG	Carrier Strike Group
CVN	Carrier Vessel Nuclear
CWC	Composite Warfare Commander
DCA	Defensive Counter-Air
DCD	Data Context Diagram
DDG	Guided Missile Destroyer
DFD	Data Flow Diagram
DIS	Data/Information Services
DoD	Department of Defense
DoDD	Department of Defense Directive
ECM	Electronic Countermeasures
EDM	Engineering Design Model
EOR	Engage on Remote
ESG	Expeditionary Strike Group
ESSM	Evolved Sea Sparrow Missile
EW	Electronic Warfare

EXCOMM	External Communications
FCQ	Fire Control Quality
FFBD	Functional Flow Block Diagram
FFG	Guided Missile Frigate
FIFO	First-In, First-Out
FP	Forward Pass
FP/C	Force Planning/Coordination
FTC	Force Track Coordinator
GAO	General Accounting Office
GCCS – M	Global Command and Control System – Maritime
GIG	Global Information Grid
HELO	Helicopter
HIL	Human-in-the-Loop
HSI	Human System Integration
IA	Information Assurance
ID	Identification
IEEE	Institute of Electrical and Electronic Engineers
IFC	Integrated Fire Control
IFF	Identification Friend or Foe
INTEL	Intelligence
IOCC	Input and Output Control Consoles
IP	Internet Protocol
IPB	Intelligence Preparation of the Battlespace
IPT	Integrated Product Team
IPv-6	Internet Protocol version 6
ISR	Intelligence, Surveillance, and Reconnaissance
JLENS	Joint Land Attack Cruise Missile Defense Elevated Netted Sensor
JEZ	Joint Engagement Zone
JTAMDO	Joint Theater Air and Missile Defense Organization
JTRS	Joint Tactical Radio System
JTT	Joint Tactical Terminal

KA	Kill Assessment
KSA	Knowledge, Skills, and Abilities
KWEB	Knowledge Web
LHA/LHD	Amphibious Assault Ship
LP	Loop
LOR	Launch on Remote
LOS	Line of Sight
MCP	Mission Capability Packages
ME	Mission Execution
MIO	Maritime Interdiction Operation
MOE	Measures of Effectiveness
MOP	Measures of Performance
MOSA	Modular Open Systems Approach
NPS	Naval Postgraduate School
NRT	Navy Review Team
NSWC	Naval Surface Warfare Center
OA	Open Architecture
OACE	Open Architecture Computing Environment
OAET	Open Architecture Enterprise Team
OMC	Operations Management Center
OO	Object-Oriented
OODA	Observe-Orient-Decide-Act
OOPDA	Observe-Orient-Predict-Decide-Act
OSJTF	Open Systems Joint Task Force
OTH	Over-the-Horizon
OV	Operational View
PAD	Planning, Assessment &Decision
PC	Precision Cue
Pd	Probability of Detection
PEO IWS	Program Executive Office of Integrated Warfare Systems
Pk	Probability of Kill

POI	Point of Impact
POMC	Platform Operations Management Center
POSIX	PorTable Operating System Interface
PROX	Proximity
Ps	Probability of Survival
PSD	Preferred Shooter Determination
RAM	Rolling Airframe Missile
RF	Remote Fire
ROE	Rules of Engagement
RV	Radar View
SA	Situational Awareness
S&D	Search and Detect
SATCOM	Satellite Communications
SCAN	Strategic Creative Analysis
SEDP	System Engineering Design Process
SIAP	Single Integrated Air Picture
S-L-S	Shoot-Look-Shoot
SM	Standard Missile
SoS	System of Systems
SRA	Self Referencing Acronym
S-S	Shoot-Shoot
SSC San Diego	Space and Naval Warfare Systems Center San Diego
SS-L-SS	Shoot-Shoot-Look-Shoot-Shoot
SOW	Statement of Work
SSDS	Ship Self Defense System
SSN	Submersible Ship Nuclear
SSPk	Single Shot Probability of Kill
SWC	Surface Warfare Commander
SWOT	Strengths, Weaknesses, Opportunities, and Threats
TADIL	Tactical Data Links
TAMD	Theater Air Missile Defense

TDEA	Triple Data Encryption Algorithm
TOI	Time of Intercept
TRO	Top Rank Objective
UAV	Unmanned Air Vehicle
US	United States
USAF	United States Air Force
VSD	Value System Design
W/AS	Weapon/Asset Services
WRT	With regard to
Xmit	Transmit

REFERENCES

Air Force Doctrine Document 1, “Air Force Basic Doctrine,” September 1997, 79.

Aitken, Ashley, “Component-Based Software: A Business Perspective.” http://72.14.253.104/search?q=cache:UDqafLUu1q8J:www.cbs.curtin.edu.au/files/cbsstaffpublications/Component-Based_Software_A_business_perspective.doc+medium-grain+components&hl=en&ct=clnk&cd=3&gl=us (accessed June 30, 2007).

American National Standards Institute (ANSI)/Institute of Electrical and Electronics Engineers (IEEE), Standard 1471, 2000.

Assistant Secretary of the Navy (Research, Development & Acquisition) (ASN [RD&A]), Policy Statement, “Naval Open Architecture Scope and Responsibilities,” August 2004.

Augelli, Vince, Dave Samara, and George Haw, “Coalition Interoperability Reaches New Heights in RIMPAC 2006,” CHIPS, 2007, 44-45.

Axe, David, *Israel and Hezbollah: Their Weapons of War*, World Politics Review Exclusive, http://www.worldpoliticsreview.com/article.aspx?id=62 (accessed on August 19, 2007).

Barker, William C., “Recommendation for the Triple Data Encryption Algorithm (TDEA) Block Cipher,” National Institute of Standards and Technology Special Publication 800-67, Version 1, National Institute of Standards and Technology Computer Security Division, Gaithersburg, MD, May 2004.

Barwis, Robert, “The Roadmap to Cruise Missile Defense”, white paper, Joint Theater Air and Missile Defense Organization (JTAMDO), March 2006.

Bruegge, Bernd and Allen H. Dutoit, *Object-Oriented Software Engineering Using UML, Patterns, and Java*. Upper Saddle River: Pearson Prentice Hall, 2004.

Cisco Networks, “Quality of Service Overview.” http://www.cisco.com/univercd/cc/td/doc/product/software/ios124/124cg/hqos_c/qchintro.pdf (accessed June 30, 2007).

Clark, Vern and Michael W. Hagee, “FORCEnet, A Functional Concept for the 21st Century.” http://www.nwdc.navy.mil/Conops/Files/FnFuncCon.pdf (accessed May 23, 2007).

Committee on the FORCEnet Implementation Strategy, Naval Studies Board, Division on Engineering and Physical Sciences, *FORCEnet Implementation Strategy*, National Academies Press, Washington, D.C., 2005.

Defense Acquisition University. “Systems Engineering Key Terms And Acronyms.” https://acc.dau.mil/GetAttachment.aspx?id=111025&pname=file&aid=24274 (accessed June 20, 2007).

Defense Science Board. “Protecting the Homeland -2000 Summer Study.” http://all.net/books/iw/cryptome.org-dio/cryptome.org/dio/dio.htm (accessed June 20, 2007).

Defense Threat Information Group, “Russian/Soviet Sea-Based Anti-Ship Missiles”. http://www.dtig.org/docs/Russian-Soviet%20Naval%20Missiles.pdf (accessed June 10, 2007).

Department of the Army, *Field Manual 34-130: Intelligence Preparation of the Battlefield*. Washington, D.C.: U.S. Government Printing Office, 1994.

England, Gordon, Vern Clark, and James L. Jones, *Naval Transformation Roadmap, Power and Access. From the Sea*, http://www.oft.osd.mil/library/library_files/document_202_naval_transformation.pdf (accessed May 1, 2007).

Eshel, David, *IAF Investigates Cause of Israeli AH-64 Helicopter Losses*, Defense Update, http://www.defense-update.com/2006_07_01_defense-update_archive.html (accessed August 19, 2007).

Eshel, David, *INS Hanit Suffers Iranian Missile Attack*, Defense Update, http://www.defense-update.com/2006/07/ins-hanit-suffers-iranian-missile.html (accessed August 19, 2007).

Feickert, Andrew “Report of the Congressional Research Service (CRS) for Congress: Cruise Missile Proliferation”, Technical Report, Specialist in National Foreign Affairs, Defense, and Trade Division: Order Code RS21252, July 28, 2005.

FORCEnet High-Level Operational View (OV-1) Diagram. Modified by Naval Surface Warfare Center (NSWC) Port Hueneme Division (PHD) MSSE Cohort #5, Team Bravo. http://www.vsix.net/other/special/United_States_IPv6_Summit_2005/United_States_IPv6_Summit.htm (accessed April 22, 2007).

Gavel, Donald T., “Multisensor Data Fusion System.” http://www.llnl.gov/sensor_technology/STR25.html (accessed June 22, 2007).

Gillis, Matt, “Open Systems Joint Task Force Gets the Word Out, PMs Now Expected to Consider Using Open Systems.” http://www.acq.osd.mil/osjtf/whatisos.html (accessed March 6, 2006).

Goebel, Greg, “An Introduction to Fuzzy Control Systems.” http://www.faqs.org/docs/fuzzy/ (accessed June 25, 2007).

Hatley, Derek, Peter Hruschka, and Imtiaz Pirbhai, *Process for System Architecture and Requirements Engineering*. New York: Dorset House Publishing, 2000.

Hichkad, Ravi R. and Christopher Bolkom, "Report of the Congressional Research Service (CRS) for Congress: Cruise Missile Defense," May 2, 2005.

Hooker, Richard, "A Baseline Definition of Architecture." http://www.wsu.edu/~dee/ARCHI/ARCHI.HTM (accessed April 15, 2007).

Hurwitz, Judith, "Enterprise C/S." http://www.dbmsmag.com/9708d04.html (accessed June 22, 2007).

JASON Program Office, MITRE Corporation, "Horizontal Integration: Broader Access Models for Realizing Information Dominance." http://www.fas.org/irp/agency/dod/jason/index.html (accessed June 23, 2007).

Kaehler, Steven D., "Fuzzy Logic – An Introduction." http://www.seattlerobotics.org/encoder/mar98/fuz/flindex.html (accessed June 23, 2007).

Kaler, Herbert C., Robert Riche, and Timothy B. Hassell, "A Vision for Joint Theater Air and Missile Defense," *Joint Force Quarterly*, 23 (1999-2000): 65-70.

Long, James "Relationships between Common Graphical Representations in System Engineering." Paper originally presented at the International Council of Systems Engineering (INCOSE), Saint Louis, MO, June 12-14, 1995. Updated July 2002.

Luessesn, Lawrence H., "A Self-Consistent Context for Unit and Force-Level Tactical Decision Making," *Naval Engineers Journal* 115 (2003): 67-78.

Mahajan, Ajay, Kaihong Wang, Probir Kumar Ray, "Multisensor Integration and Fusion Model that uses a Fuzzy Inference System." http://www.engr.siu.edu/asl/j11.pdf (accessed June 24, 2007).

Mallory, J.A., *Global Command and Control System-Maritime (GCCS-M) Navy Training System Plan (NTSP) E-70-9804*. http://www.fas.org/man/dod-101/sys/ship/weaps/docs/gccs-m-ntsp/1_cover.htm#N6-NTSP-E-70-9804 (accessed July 28, 2007).

Mullen, Michael, "Naval OA Strategy." https://acc.dau.mil/GetAttachment.aspx?id=129676&pname=file&aid=26477 (accessed March 6, 2006).

Naval Surface Warfare Center Division, "Open Architecture (OA) Computing Environment Technologies and Standards," Version 1.0, dated 23 August 2004.

http://www.nswc.navy.mil/wwwDL/B/OACE/docs/OACE_Tech_Stds_v1dot0_final.pdf (accessed March 3, 2007).

OA Technical Architecture Integrated Product Team (IPT), *Open Architecture Computing Environment Design Guidance 1.0*. http://www.nswc.navy.mil/wwwDL/B/OACE/docs/OACE_Design_Guidance_v1dot0_final.pdf (accessed March 10, 2007).

Owen, Mark W., "An Artificial-Neural-Network Multiple-Model Tracker." http://www.spawar.navy.mil/sti/publications/pubs/td/3155/4a_S3papers/ANNMMT.pdf (accessed June 24, 2007).
Page-Jones, Meilir, *The Practical Guide to Structured Systems Design*, chap. 6, http://www.waysys.com/ws_content_bl_pgssd_ch06.html (accessed June 23, 2007).

Pal, Nikhil R. and Rajani K. Mudi, "Computational Intelligence for Decision-Making Systems," *International Journal of Intelligent Systems* 18 (2003): 483-486.

Pendall, David W., "Persistent Surveillance and Its Implications for the Common Operating Picture," *Military Review*, November-December 2005, 41.

Roche, Pat, "Architecture: The Foundation for FORCEnet," presented at the FORCEnet Engineering Conference, Virginia, June 2005.

Rushton, Richard T, Michael McCrave, Mark N. Klett, and Timothy J. Sorber, "Open Architecture, The Critical Network Centric Warfare Enabler," Technical Report, Network Systems and Integration Branch, Surface Warfare Directorate (N76), Chief of Naval Operations Staff (OPNAV), Anteon International Corporation, and Klett Consulting Group, Inc., March 18, 2004.

"SC-21/ONR S&T Manning Affordability Initiative." http://www.manningaffordability.com/S&tweb/Index_hse.htm (accessed June 22, 2007).

Sage, Andrew P. and James E. Armstrong Jr., *Introduction to Systems Engineering*. New York: John Wiley & Sons, Inc., 2000.

Schekkerman, J., "Adopting & Developing an Effective Enterprise Architecture for Network Centric Capability, Information Dominance, The New Doctrine," Part 1, Workshop, Institute for Enterprise Architecture Developments, Verdonck, Klooster, and Associates, presented at Dominant Battlespace Knowledge, Asia, 2005.

Sibbald, Robert, Joe Zuliani, and Stephan Lapic, "Subnet Relay – Enhancing Multinational Connectivity," CHIPS, 2007, 24-26.

Simonoff, Adam J., "A Conceptual Architecture for Naval Effects-Based Operations," white paper, Dahlgren, VA, 2006.

Sowell, Thomas, "Fuzzy Logic Tutorial," http://www.fuzzy-logic.com (accessed June 23, 2003).

Stergiou, Christos and Dimitrios Siganos, "Neural Networks." http://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol4/cs11/report.html (accessed June 22, 2007).

Strei, Thomas J., "Open Architecture in Naval Combat System Computing of the 21st Century," white paper, Washington, D.C., April 2003.

Strei, Thomas J., "Open Architecture, An Enterprise Approach to Introducing Open Architectures into navy Combat Systems… and Beyond." Presentation given at the Modular Open Systems Approach Review Team (MOSART) meeting, Alexandria, VA, February 27, 2004.

The Rand Corporation, *Cruise Missile and Ballistic Missile Defense*, chap. 3, http://www.rand.org/pubs/monograph_reports/MR1449/MR1449.ch3.pdf (accessed June 29, 2007).

Under Secretary of Defense (Acquisition, Technology & Logistics) Memorandum, "Amplifying DoDD 5000.1 Guidance Regarding Modular Open Systems Approach (MOSA) Implementation," April 2004.

United States General Accounting Office (GAO), "Defense Acquisitions: Comprehensive Strategy Needed to Improve Ship Cruise Missile Defense," Technical Report No. GAO/NSIAD-00-149, Washington, D.C., July 2000.

United States Naval Reserve Intelligence Program, "Ready-for-Sea Modular Course & Handbook." http://www.fas.org/irp/doddir/navy/rfs/part03.htm (accessed May 4, 2007).

Winer, Leon, *MBA Tool Box*, chap. 1, http://mbatoolbox.org/stories/storyReader$19 (accessed June 30, 2007).

Young, Bonnie W., "Integrated Fire Control for Future Aerospace Warfare," presented at the 10th International Command and Control Research and Technology Symposium, June 2005.

Zinger, William H. and Jerry A. Krill, "Mountain Top: Beyond-the-Horizon Cruise Missile Defense," *Johns Hopkins Applied Physics Laboratory (APL) Technical Digest* 18 (1997): 501-520.

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California

3. John M. Green
   Senior Lecturer
   Naval Postgraduate School
   Monterey, California

4. Paul V. Shebalin D.Sc.
   Senior Lecturer
   Naval Postgraduate School
   Monterey, California

5. Department of the Navy, Program Executive Office
   Integrated Warfare Systems 7, Naval Open Architecture
   Washington DC

6. Wesley Holser
   Naval Surface Warfare Center
   Port Hueneme, California

7. Adam J. Simonoff
   Naval Surface Warfare Center
   Dahlgren, Virginia

8. Juan Camacho
   Naval Surface Warfare Center
   Port Hueneme, California

9. Lawerence Guest
   Naval Air Warfare Center
   China Lake, California

10. Belen Hernandez
    Naval Surface Warfare Center
    Port Hueneme, California

11. Thomas Johnson
Naval Surface Warfare Center
Port Hueneme, California

12. Alan Kang
Naval Surface Warfare Center
Port Hueneme, California

13. Giang Le
Naval Surface Warfare Center
Port Hueneme, California

14. Brian Macgillivray
Naval Surface Warfare Center
Port Hueneme, California

15. Tu Ngo
Naval Surface Warfare Center
Port Hueneme, California

16. Kyle Norman
Naval Air Warfare Center
China Lake, California

17. Franklin Tomei
Naval Surface Warfare Center
Port Hueneme, California

THIS PAGE INTENTIONALLY LEFT BLANK