2007

AEGIS platforms: using KVA analysis to assess Open Architecture in sustaining engineering

Adler, Jameson R.

Monterey, California. Naval Postgraduate School

NPS-AM-07-049

ACQUISITION RESEARCH SPONSORED REPORT SERIES

AEGIS Platforms: Using KVA Analysis to Assess Open Architecture in Sustaining Engineering

30 June 2007

by

Jameson R. Adler, ENS, USN

Jennifer L. Ahart, ENS, USN

Advisors: Thomas Housel, Professor, and

Glenn Cook, Lecturer,

Graduate School of Operational & Information Sciences

Naval Postgraduate School

Prepared for: Naval Postgraduate School, Monterey, California 93943

The research presented in this report was supported by the Acquisition Chair of the Graduate School of Business & Public Policy at the Naval Postgraduate School.

To request Defense Acquisition Research or to become a research sponsor, please contact:

NPS Acquisition Research Program
Attn: James B. Greene, RADM, USN, (Ret)
Acquisition Chair
Graduate School of Business and Public Policy
Naval Postgraduate School
555 Dyer Road, Room 332
Monterey, CA 93943-5103
Tel: (831) 656-2092
Fax: (831) 656-2253
e-mail: jbgreene@nps.edu

Copies of the Acquisition Sponsored Research Reports may be printed from our website www.acquisitionresearch.org

Abstract

The purpose of this thesis is to estimate the potential performance improvement in sustaining engineering (SE) when an Open Architecture (OA) approach to system development is used. Its basis is that in Integrated Warfare Systems (IWS) acquisition, 80% of total lifecycle costs occur during the Operation and Support phase. This statistic demonstrates the necessity of measuring how the OA approach will affect software upgrades and maintenance processes for the AEGIS IWS lifecycle. Using the OA approach, advances in distance support and monitoring and maintenance-free operating periods are possible; these advances are significant in supporting the need to reduce costs and manpower while improving performance. To estimate the potential (Return on Investment) ROI that an OA approach might enable SE in the form of software maintenance and upgrades, this thesis will apply the Knowledge Value Added (KVA) methodology to establish the baseline, "As Is," configuration of the current solutions in AEGIS. The KVA analysis will yield the ROI's and the current models for the approach to software maintenance and upgrades. Based on the assumptions of OA design for original system development, new approaches to distance and maintenance and monitoring will be explored in "To Be" solutions, and the ROIs will be estimated. The "To Be" solutions are rooted in the assumptions of MFOP and ARCI, and the results indicate that these solutions yield a potential improvement of 720% and a cost savings of $365,104.63 over the current methodology for just one ship. For all ships using AEGIS, ROI improves by 71,967%—with a cost savings of $26,543,824.56. The conclusion is that OA enables extension of these best practice approaches to AEGIS maintenance and upgrade solutions.

Keywords: AEGIS Platforms, KVA, KVA+RO, Sustaining Engineering, Distance Support

THIS PAGE INTENTIONALLY LEFT BLANK

Acknowledgements

We would like to thank RADM Jim Greene for his generous support of our research.

We would like to thank Mr. Mark Wessman for his innovative vision in seeing the possibilities of the future.

We would like to thank Dr. Thomas Housel and Mr. Glenn Cook for dedicating their time and energies to guiding us through the thesis process.

Additionally, we'd like to thank Mr. Eric Tarantino, Ms. Penny Graninger and Mr. Joel Timm, our families and our friends.

THIS PAGE INTENTIONALLY LEFT BLANK

NPS-AM-07-049

ACQUISITION RESEARCH
SPONSORED REPORT SERIES

AEGIS Platforms: Using KVA Analysis to Assess Open Architecture in Sustaining Engineering

30 June 2007

by

Jameson R. Adler, ENS, USN

Jennifer L. Ahart, ENS, USN

Advisors: Thomas Housel, Professor, and

Glenn Cook, Lecturer,

Graduate School of Operational & Information Sciences

Naval Postgraduate School

THIS PAGE INTENTIONALLY LEFT BLANK

Table of Contents

List of Abbreviations, Acronyms and Symbols

ACCESS/STARSY:	AEGIS Software Database
AEGIS:	Advanced Electronic Guided Interceptor System
CASREP:	Casualty Report
CBM	Component Business Model
CDR:	Critical Design Review
CPCR:	Computer Program Change Request
CPDD:	Computer Program Design Document
DoD:	Department of Defense
ECP:	Engineering Change Proposal
GIG:	Global Information Grid
ICR:	Interface Change Request
IWG:	Integration Working Group
IWS:	Integrated Warfare Sytems
JCRB:	Joint Change Review Board
KVA:	Knowledge Value Added
KVA+RO:	Knowledge Value Added and Real Options
MOE:	Measure of Effectiveness
NMCI:	Navy Marine Corps Intranet
NSWC:	Naval Surface Warfare Center
OA:	Open Architecture
OACE:	Open Architecture Computing Environment
PAT:	Preliminary Acceptance Test
PDR:	Preliminary Design Review
PDS:	Preliminary Design Specification

PEO IWS:	Program Executive Office, Integrated Warfare Systems
RO:	Real Options
ROI:	Return on Investment
ROK:	Return on Knowledge
SCCB:	Software Configuration Change Board
SCP:	Software Change Proposal
SDD:	Software Design Document
SE:	Sustaining Engineering
SHIPALT:	Ship Alterations
SME:	Subject-matter Experts
SOA:	Service-oriented Architecture
SSDS:	Ship Self Defense System
STR:	Software Trouble Reports
TLT:	Total Learning Time

I. Introduction

A. Purpose

The need has always existed aboard ship to maintain operations of specific organic assets while at sea, and as the United States Navy rapidly advances towards reduced manning, Integrated Warfare Systems (IWS) will require a new approach to Sustaining Engineering (SE) for software maintenance and upgrade solutions if a smaller crew is to perform at the same caliber. The value of sustaining engineering and creating more efficient software upgrades can be realized as time spent on mission efforts increases, and time spent on solving IWS anomalies decreases. The more efficient system design of the future can adapt to the requirements through open architecture (OA); and, when designers use an OA approach, it enables innovation without major efforts on the part of the ship's crew.

A previous study conducted at the Naval Postgraduate School by Capt. Joseph Uchytil entitled, "Assessing the Operational Value of Situational Awareness of AEGIS and Ship Self-Defense System (SSDS) Platforms through the Application of the Knowledge Value Added (KVA) Methodology," demonstrated that KVA could be used to estimate the performance of an OA implementation in terms of a Return on Investment (ROI). While Capt. Uchytil's research focused on benefits derived from the warfighters' perspective, the purpose of this research is to generate and assess the impact of an OA development and acquisition approach to SE in IWS.

By extending the focus of this OA study from the warfighter to the acquirer and system developer, the analysis provides a more complete view of the whole system development lifecycle and the potential of OA to improve the performance of the cycle in the SE processes. Due to the scope of this study, it is more likely for the SE portion of the lifecycle to achieve the highest potential productivity; this, in turn, helps to ensure the benefits of OA are exploited in this part of the lifecycle, which is developer- and acquirer-intensive.

B. Background

The benefit of open architecture—from the developer and acquirer point-of-view—is that it creates greater flexibility by introducing additional technologies and capabilities to the Fleet, which closed systems of the past have failed to introduce after procurement. This is because closed systems are, typically, not as amenable to rapid upgrades as open systems. Current systems need to be fluid and dynamic to respond to and anticipate the anomalies encountered on ships. Ergo, it is no longer practical for software maintenance support and upgrades to operate within closed systems, because they are difficult and slow to upgrade, have limited interoperability, and upgrades must be done with the same vendor. Additionally, it is hard to maintain proprietary systems because of their interdependencies in code and software. Due to lifecycle time and cost constraints, OA, which offers independent coding and faster business and system models to the acquirer and developer, should be effectively used in order to promote the future view of a Navy which will operate within the Global Information Grid (GIG). The Global Information Grid (GIG) will create an informative and integrated environment in which to pass the information (Clark, 2002). This will require integration of several parts, and it will require those parts to be fully operational, usable, and easily upgraded under reduced manning and joint operational conditions. One of the enablers of GIG will be the effectiveness of open architecture, which will allow for faster integration when applied to sustaining engineering—i.e., the software maintenance support and upgrade process.

C. Research Objectives

The objective of this research is to analyze the potential benefits of open architecture from the developers' and acquirers' perspective in the SE process for the AEGIS weapons system. This will be achieved through the KVA approach that will provide the analytical framework to assess the added value of the open architecture approach to software maintenance support and upgrade solutions to the developer and acquirer.

The KVA approach provides the static ROI analysis models which serve as the input for Real Options (which will be conducted in further research). By placing both related and unrelated elements in a single unit, the value of both objects can be compared in one lump figure. Please refer to the Appendix for a more detailed description of the KVA methodology.

D. Research Questions

Since the measure of effectiveness of a ship in the Fleet does not rely solely on monetary cost savings, but rather on the productivity and mission accomplishment, the knowledge value-added approach can be applied to produce a return on investment (ROI) that will serve as a measure of productivity. Possible models that would increase the productivity of software maintenance support and upgrade solutions in the AEGIS weapons system can be explored after a baseline of the current system is established.

This thesis will provide “To Be” scenarios for using an open architecture approach to meet the demands of the future Navy with regards to sustaining engineering. The secondary research will explore the Department of Defense and Department of the Navy initiatives for Open Architecture, Open Architecture Computing Environments, Services-oriented Architecture Solutions, Distance-support Solutions, and current “best practices” examples in software upgrades in military environments.

Our analysis will answer the following research questions:

- Using IBM's Component Business Modeling (CBM), what are the areas of highest concern and cost in the AEGIS weapons system as they relate to sustaining engineering?
- What are the "best practice" examples of sustaining engineering in the commercial or military environment, and how do they improve the processes of system development and acquisition lifecycles? What are the best examples of SOA and distance support systems?
- What is the benefit of OA in the process of sustaining engineering? We will apply the KVA methodology to the areas of highest concern as identified by IBM in its CBM to address these questions.
- What overall numerical effect does OA have on Sustaining Engineering, and is it appreciable and low-risk enough to system development and to the DoD acquisition lifecycle?

This research will provide decision-makers with a structured analysis of employing open architecture to improve productivity within sustaining engineering and software upgrades for Integrated Warfare Systems (IWS).

E. Methodology

We will employ the case example method when conducting our research. We will focus on exploring various avenues of improvement associated with sustaining engineering, specifically the software maintenance and upgrade processes for the AEGIS system. A KVA analysis will be conducted on the system in the "As Is" configuration. This will serve as a static baseline analysis and will be used to generate the "To Be" models. The KVA and follow-on RO analysis in future research will then be conducted for the software maintenance and upgrade process that employs an OA framework. Both analyses will be conducted with the help of AEGIS subject-matter experts. The KVA analysis associated with the "As Is" and the "To Be" (employing an OA framework) systems will produce an ROI for each process model. The ROI associated with each process model will be compared in further research to determine the impact of OA as a viable solution to improving sustaining engineering and the software maintenance and upgrade processes associated with AEGIS. The follow-on RO analysis will identify options associated with each process model—including valuation options, cost options and risk options. The results will be compared and evaluated, and the benefit of OA to sustaining engineering and the software upgrade processes for Integrated Weapons Systems (IWS) will be determined. The research will offer recommendations on how to improve sustaining engineering and the software upgrade processes in the context of OA systems.

F. Scope

The scope of this thesis will be to specifically prescribe an operational value to the improvement of the software maintenance and upgrade procedures of AEGIS using OA. This research highlights the inherent value of knowledge capital in a system by using KVA methodologies; and it emphasizes the need to introduce OA solutions to many more sustaining engineering processes aboard ships which will undergo reduced manning and be expected to achieve "decision superiority" in the future.

G. Thesis Organization

The chapter organization will be as follows: Chapter I will give a general overview of the purpose and intended methods and scope of the thesis. Chapter II will provide secondary research and background information on open architecture aims, OACE, SOA, distance support solutions, and best practice examples. Chapter III will consist of the KVA analysis and the resulting figures and charts. Chapter IV will discuss the results from the KVA analysis and the implications of the current "As Is" state of AEGIS maintenance and then explore the "To Be" results. Chapter V will present conclusions from the research that was conducted. Chapter VI will recommend further research that can be conducted to continue the process of refining sustaining engineering and software upgrades within the context of open architecture in the Fleet.

THIS PAGE INTENTIONALLY LEFT BLANK

II. Literature Review: Software Upgrades and Maintenance Solutions in the Open Architecture Environment

A. CBM Theory and Sustaining Engineering

The driving factor that has promoted the need for improvement of Sustaining Engineering (SE) in software maintenance and upgrades is the Component Business Model (CBM) process conducted by IBM in June of 2006. Component Business Modeling is an "IBM-developed technique for representing an enterprise as non-overlapping business components in order to identify opportunities for innovation and/or improvement" (Pavlick, 2005, p. 7). In a CBM, a business component is a group of resources, people, and technology that have the necessary information to deliver value from functional performance (p. 7). The final result of the grouping of business components is visually represented in a component business map, which hones in on the essential foundational blocks of the organization. Figure 1 is the final component business map for the breakdown of AEGIS in Program Executive Office Integrated Warfare Systems (PEO IWS) for Fiscal Year 2007.

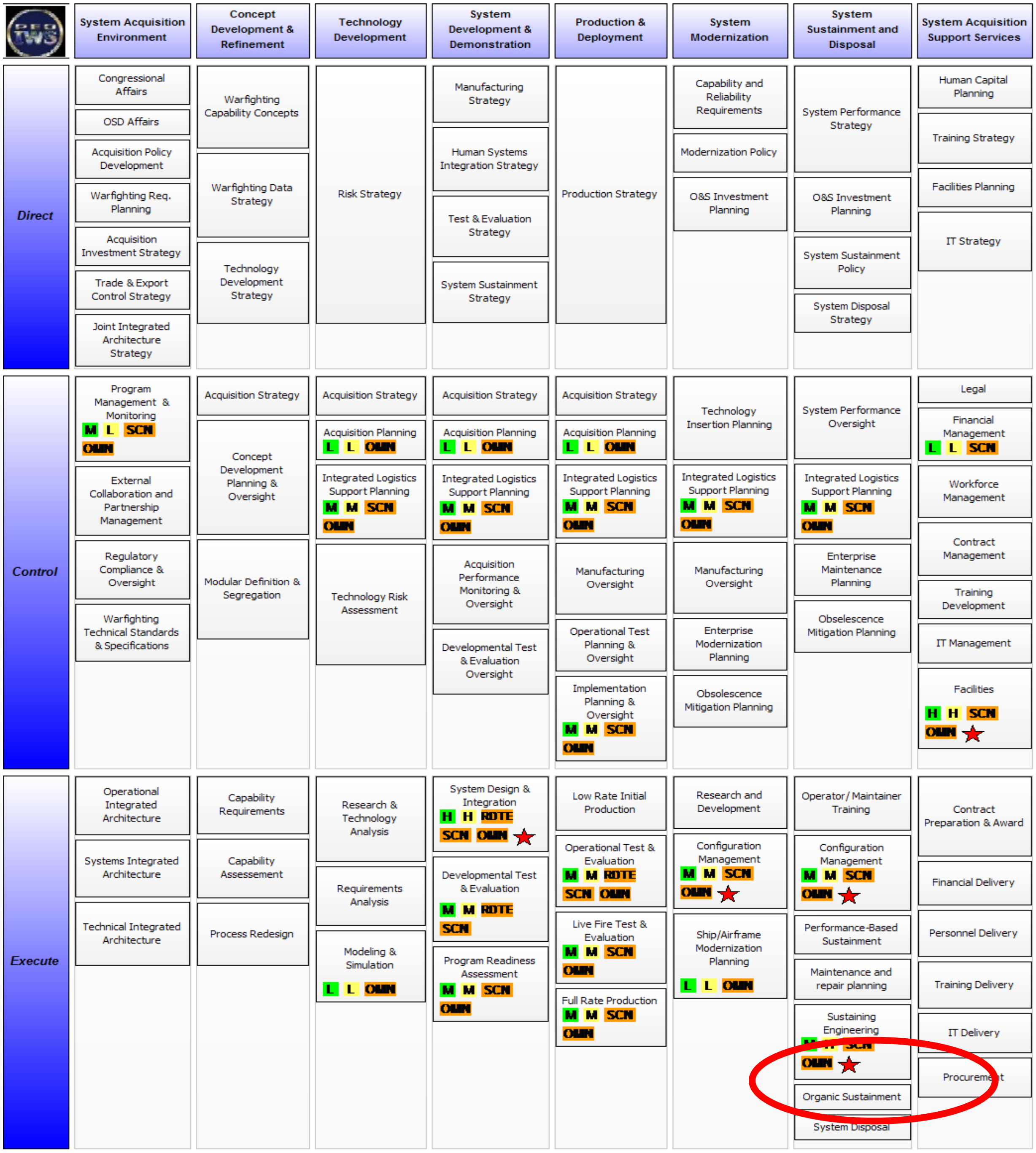


Figure 1. CBM Map Identifying Sustaining Engineering as “Hot Component”
(Shannon, 2006, p. 14)

“Hot components,” or components that are worth further examination, are represented by a star. They are identified as hot components based on attributes selected as important to the organization being assisted through the analysis. In the case of this CBM effort, there were three criteria selected: investment of total budget (green), number of efforts required for the task (yellow), and color of money (orange) (Shannon, 2006, p. 11). The sustaining engineering category has a medium percentage of the PEO IWS budget, a high number of efforts (greater than six), and two colors of money involved. The colors of money, or the money which is procured and used for specific acquisitions, are in the areas of Operation and Maintenance (OMN) and Ship Building and Conversion (SCN). The horizontal axis represents a key competency (one which requires similar skills and capabilities), while the vertical axis represents accountability levels. SE is "System Sustainment and Disposal" competency, in which the "executing" branch, the branch that does the work, is accountable.

In addition to SE being identified as a "hot component," the Operations and Support (O&S) phase of a system's lifecycle is often represented to incur 80% of the total lifecycle costs of a system. According to an article published in *Program Managers Magazine*, weapons system sustainment consumes "about 80 percent of logistics resources, or approximately 64B per year" (Kratz, Fowler & Cothran, 2002, p. 2). With such a large factor of the total lifecycle costs being focused in this lifecycle phase, along with the CBM results, it is reasonable to examine Sustaining Engineering for ways to make it more efficient. IBM also anticipates that a large cost component within O&S is SE (Shannon, 2006, p. 16). In Figure 2, this is evident. In fact, this is the only starred area on the CBM diagram in which all colors of money will increase spending. KVA will seek to give decision-makers a tool-set for making the vast amount of spending on SE more efficient through the use of OA.

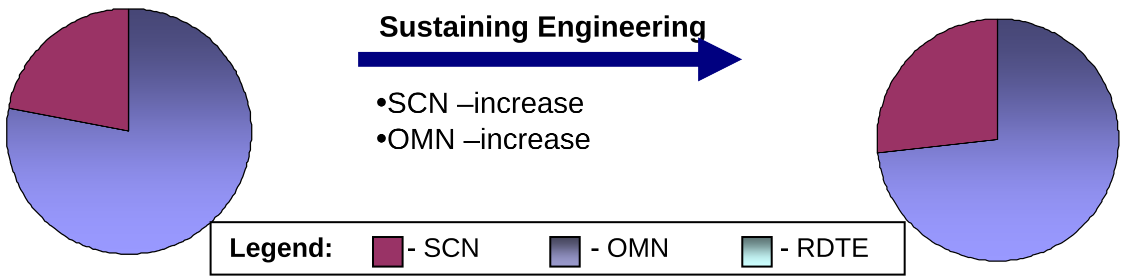


Figure 2. Sustaining Engineering Color of Money Cost Increase
(Shannon, 2006, p. 16)

B. Open Architecture Environment and Tenets

To achieve efficiency in software upgrade and maintenance, it is necessary to eliminate current inadequacies in implementing open architecture. Department of Defense systems, according to a report released in 2006 by the Government Accountability Office, continue to lag behind in interoperability, even though the Program Executive Office, Integrated Warfare Systems (PEO IWS), was established in 2002 and was in charge of executing OA (GAO, 2006). Perhaps this is a result of the well-known and frequently addressed security concerns involved in implementing OA in weapons systems, such as malicious code, computer viruses or system latency; however, the civilian sector grasped the concept quickly, and they also have a need for security. Even banks, for the most part, operate within the framework of OA. Some banks even operate with Service-oriented Architecture, which will be defined and discussed at the conclusion of the literature review.

To implement OA, it is necessary to understand some basic concepts. In a general sense, OA is realized through rapid change and fluid upgrades and solutions. According to the Deputy Chief of Naval Operations, the requirements for OA implementation are as follows: modular design and design disclosure, reusable application software, interoperable joint warfighting applications and secure information exchange, lifecycle affordability, encouraged competition and collaboration, scalability and portability (Chief of Naval Operations, 2005).

1. Modular Design and Design Disclosure

Modularity is the concept of decomposing a system into transparent subcomponents (Coronado Mondragon, Coronado Mondragon & Miller, 2006, p. 247). These subcomponents are operable without relying on another aspect of the system; hence, they can rapidly change and allow for interactions with numerous systems. The underlying goal of decomposition, in the case of modularity, is to allow for the independent upgrade of each of the smallest subcomponents while leaving the complete system operable. With modular design and design disclosure, multiple competitors can participate; thus, innovation flourishes as each subcomponent is independently tried and tested.

2. Reusable Application Software

Reuse allows a system to use the same components and code that have been used across other platforms (Chief of Naval Operations, 2005). In the case of application software, a database of segments of code that worked for the tracking device of one platform can be shared when creating other tracking devices. This database would be continually updated with components and segments of code that could have potential use in other areas. These components can be used interchangeably with other components without affecting the system in its entirety. This idea is revolutionary for coding and software upgrade in much the same way that “interchangeable parts” revolutionized the assembly lines of the 1920’s with increased output and increased revenue. Disclosure of the design of application software would also be necessary for evolutionary improvement in future upgrades (2005).

3. Interoperable Joint Warfighting Applications and Secure Information Exchange

This particular tenet ensures that across a wide variety of systems, the same information and applications can be shared. It involves commonality of services, warfighting applications, and information assurance and requires these commonalities to be essential for the basic design elements of any new system (2005).

4. Lifecycle Affordability

This tenet includes all phases of the lifecycle—from design and requirements gathering to delivery and testing. Since the primary concern of this thesis is the sustaining engineering portion, and since SE is such a large portion of the lifecycle costs, the results could directly benefit the implementation of OA with respect to lifecycle affordability.

5. Encouraging Competition and Collaboration

OA naturally encourages competition and collaboration, because unlike closed systems, many different systems can be integrated to complete upgrades or create a new system. That is not to say that proprietary systems did not contain many different parts that required different companies to collaborate, but they were less likely to constantly create an environment of competition and innovation because some of the contracts were sole-source. Sole-source contracts are those which restrict Full Open Competition; the sole-source method is a non-competitive procurement process in which solicitation is only with one source (DoD, 2003).

6. Scalability

Scalability encompasses the introduction of new functionalities into a system without procuring a whole new system to do the same job. An example of scalability is the method of increasing bandwidth during the holiday season to allow for faster transactions during a season of heightened traffic.

7. Portability

Portability is the ability of the software or hardware and its users to easily integrate into different platforms. It requires source code to make transitions between hardware and software and requires the switch to be rapidly and smoothly accomplished.

C. Legacy Systems and Their Effect on Software Upgrades and Maintenance

For the most part, closed systems of the past contain software which is designed for the purpose of supporting the computing hardware. When proprietary systems do need upgrades, computer code must change as well; but their unique design sometimes makes software upgrades financially and technically prohibitive. Programs such as these delay the time to introduce new technologies to the Fleet and increase the total lifecycle cost of the system. Table 1, shown below, lists the contrasting characteristics between closed systems and open systems. Most important to this research are three points:

- That expansion and upgrade of closed systems requires more time, effort, and money than the open systems.
- That closed systems are less adaptable to changes in threats and new technologies.
- That closed systems are focused mostly on development cost and on meeting the present mission, while open systems focus on the total costs of ownership, sustainment, and growth of the system.
-

Closed System Characteristics	**Open System Characteristics**
Use of closely held, private interfaces, languages, data formats and protocols (government- or vendor-unique standards)	Use of publicly available and widely used interfaces, languages, data formats and protocols
Critical importance is given to unique design and implementation	Critical importance is given to interface management, and widely used conventions
Less emphasis on modularity	Heavy emphasis on modularity
Vendor and technology dependency	Vendor and technology independence
Minimization of the number of implementations	Minimization of the number of types of interfaces

Difficult and more costly integration	High degree of portability, connectivity, interoperability, and scalability
Use of sole-source vendor	Use of multiple vendors
Expansion and upgrading usually requires considerable time, money and effort	Easier, quicker and less expensive expansion and upgrading
Higher total ownership cost	Lower total ownership cost
Slower and more costly technology to transfer	Technology transfer is faster and less costly
Components, interfaces, standards, and implementations are selected sequentially	Components, interfaces, standards, and implementations are selected interactively
Systems with shorter life expectancy	Systems with longer life expectancy
Use of individual company preferences to set and maintain specifications	Use of group consensus process to maintain interface specifications
Less adaptable to change in threats and technologies	More adaptable to evolving threats and technologies
Focusing mostly on development cost and meeting present mission	Focusing on total costs of ownership, sustainment and growth
User as the producer of system	User as the consumer of components
Rigid and slow system of influence and control	Real-time and cybernetic system of influence and control
Adversarial relationship with prime contractors/supplier/vendors	Symbiotic relationship with prime contractors/suppliers/vendors
Mostly confined to traditional suppliers	Non-traditional suppliers can compete
Simple conformance testing	Very challenging conformance testing

Table 1. Open Systems v. Closed Systems

(Azani, 2001, p. 1)

Legacy systems also have a specific computational power limitation. Systems like the AEGIS 6Ph3 radar processing has software which relies on the military standard computer, UYK-43, which was sole-source contracted to Lockheed Martin in 1980 (FAS Military Network Analysis, 1998). Such systems cannot keep up with the steadily increasing computational power in the commercial sector. The negative effect on current, closed systems is magnified because they are fast becoming obsolete while the benefits of OA are not realized. This research seeks to shift the focus of the AEGIS system software upgrades to increase the overall value of the system both monetarily, operationally, and from a user's perspective by proving the worth of the implementation of open architecture in software maintenance and development—where it is most amenable.

D. Open Architecture Computing Environment

As stated previously, the high costs of computer program maintenance and development are attributed to obsolescence, frequently needed changes, and proprietary systems which contain software applications that are closely linked to the backbone of system operation. Maintenance and development of such software could adversely affect the system as a whole, thus making it less amenable to any type of change which, hypothetically, is avoidable in an age with rapid development of commercial off-the-shelf (COTS) technology.

According to the directive for Network Centric Warfare, the open architecture concept will be applied not only to hardware, but also to software and the computing environment. An Open Architecture Computing Environment (OACE) is, at its essence, the application of open architecture to computing systems—so that over the life of the platform, changes can be made with commercial technologies that will rapidly meet the changing demands of reduced manning source. Closed systems of the past reduced the ability of developers to modernize the system and to provide maintenance solutions for underlying problems. They also robbed the acquisition field of competitive contracts, as their field of suppliers was limited. In existing proprietary systems, obsolescence and the inability to introduce upgrades has decreased the overall value of acquiring the system. Vice Admiral John Nathan said to the House Armed Services Committee in 2003:

> By pursuing an Open Architecture and an Open Architecture Computing Environment based on mainstream COTS technologies, systems and

> standards, we can avoid the high cost of maintaining and upgrading multiple legacy computing systems that quickly become obsolete and are not responsive to changes in war-fighting requirements. (Nathan, 2003)

Additionally, *DoD Directive 5000.1* states, "a modular, open-systems approach shall be employed where feasible"; and a memorandum in April 2004 from the Under Secretary of Defense for Acquisition, Technology, and Logistics, expands on that language by requiring that all programs are subject to a milestone review brief of their modular open-systems approach (DoD, 2003).

1. OACE Shift

The move to the OACE is designed to be incremental and is currently implemented in four categories, as Table 2 below demonstrates.

Category 1 Hardware Adapter	*Category 2* OACE Interface	*Category 3* OACE Standards	*Category 4* OA Common Functions
• Legacy Application	• Legacy Application or Requirements-based Innovative Application	• Applications Running on OACE Standards (e.g., Operating System, Middleware, etc.)	• Common Applications and Services Across Platforms
• Legacy Hardware • Legacy Operating System, Middleware, etc. • Physical Interface Adapter • Little Reuse	• Legacy Middleware and Operating System APIs • "Wrapper" Layer Makes Application Code Portable • OACE Middleware for External Interfaces • Systems-level Reuse	• OACE Standards Used Internally • OACE Physical Infrastructure • Minimal Change to Application Software Design • Supports Common Function Reuse • Distributed Computing Resource Managememt – Next Generation Approach to Survivability and Extensibility* ➢Location Transparency ➢Shared Resources	• Applications Built on OACE Standards • Use of OA Common Applications and Services Across the Force • Applications Adhere to OA Functional Architecture and APIs • Applications Use Common Design Patterns (e.g., Fault Tolerance)
Non-OACE Application Non-OACE Environment Hardware Adapter OACE App	Non-OACE Application Adaptation layer OACE App	OACE-Based Application OACE	Common Apps OA Apps OA Services OACE *selected acquisitions only

Table 2. OACE Incremental Compliance
(DoD, NAVSEA, PEO IWS, 2004)

2. OACE Category 3

Open Architecture Computing Environment category 3 describes complete compliance with all OACE standards to include, physical media, networks, operating systems, middleware and programming languages. This is critical in reuse of components and allows for the interoperability between different computing infrastructures. The goal for the full integration of Category 3 is the year 2008, with the main component being the standard middleware (DoD, NAVSEA, PEO IWS, 2004). Middleware is the use of software which allows interoperability between two different closed systems. Rather than proprietary-based middleware, standards-based solutions will be used instead to meet the ever-changing requirements of the system. Standards-based solutions are those which meet the industry regulated norm, such as the “user-friendly” norm which Microsoft created when it released Windows on an international level.

E. Security Questions

One central concern of the Department of Defense with regards to the shift to OA and OACE is the need to maintain security in military systems. Some have speculated that to let open architecture be the prevailing architecture for Fleet-releasable software maintenance and upgrades would be to let the proverbial "wolf in sheep's clothing" penetrate our defenses. These concerns stem from the fear of malicious code causing a whole defense system to malfunction. Supporters of open architecture state that because newer systems are so open to review by so many different sources, the possibility of malicious code passing under the eyes of so many is slim.

F. Service-oriented Architecture (SOA) Solutions

Service-oriented architecture is a permutation of OA in that it is the ultimate in the OA Tenets of “modularity” and “reuse.” SOA seeks to combine many different services that communicate through XML messages across a common web so they can be interchangeably used to complete a task. It makes software-based changes rather than hardware-based changes. Figure 3 is a before and after example of the implementation of SOA, and it includes both pros and cons.

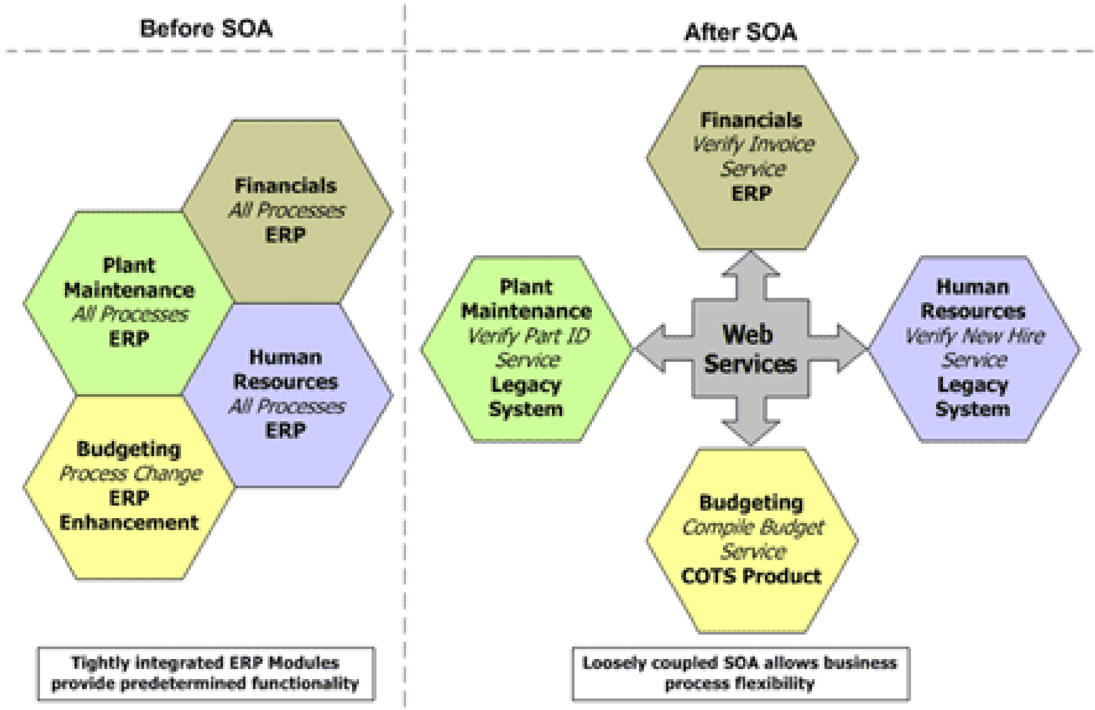


Figure 3. Before & After SOA
(US Army PEO IES, 2006)

1. SOA Benefits

With the introduction of SOA, any service that requires examining, such as a broken pipe or a computer that has "crashed," can be evaluated by meshing together whichever portions of an organization need to be combined to address the specific problem at the time of an anomaly. Some other benefits include a higher potential ROI, visibility or enterprise-level business processes, reduced redundancy and ease of interoperability internally and between third parties (US Army PEO IES, 2006). SOA could be extremely beneficial in this case by bypassing organizational "stove-pipes" and bridging legacy systems that otherwise would be non-interoperable in the environment.

2. SOA Drawbacks

There are several drawbacks to SOA. If there is a large volume of transactions within the system (for instance, an online bank), then SOA would require a massive amount of time and dedicated resources to fully realize its potential. Additionally, security is as much of an issue in SOA as it is in OA. At times, it may be easier to have tightly grouped interfaces with a history of collaboration rather than loose interfaces. This is especially sensitive where AEGIS is concerned because if the interface is not tightly coupled, then any latency present could cause enough delay to give an enemy superiority over our assets. As an air defense platform, any increase in latency would be unacceptable.

G. Distance Support Maintenance Solutions

According to the 2006 *Distance Support Policy* released by the Chief of Naval Operations, distance support is rapidly becoming "the Fleet's principal web-based readiness enabler" (Chief of Naval Operations, 2006b). At a minimum, the current distance support system "combines people, processes, and technology into a collaborative infrastructure without regard to geographic location" (2006b). In other words, ships can be underway for several months and communicate with shore-based sites to fix software and hardware problems that occur onboard and, hopefully, resolve them without pulling into a foreign port or returning to the shipyards.

The future of distance support will also include shore-based monitoring of systems, in much the same way that cars sold in 2006 and 2007 can communicate with central databases and give a report of their technical status, which is then emailed to the owner of the vehicle. This form of distance support, called remote monitoring and notification, in a possible form of procedure for shipboard operations, is displayed in Figure 4 below.

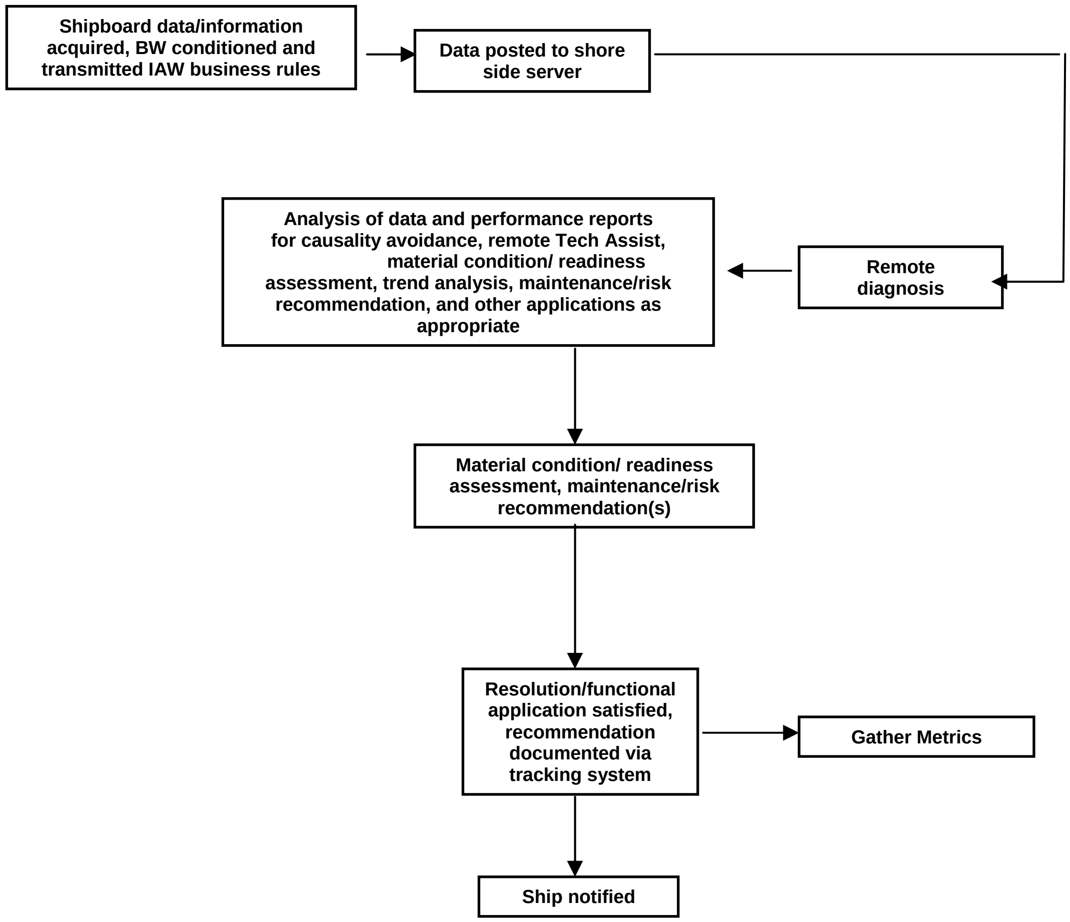


Figure 4. Remote Monitoring and Notification

As illustrated in the figure, the shipboard information is constantly monitored in the "data/information acquired phase," which then relays the information to the shore-side server for diagnostics and assessments of trends and material readiness. If there is a problem with the systems, maintenance and risk recommendations are made, documented, and then analyzed in metrics; the ship is notified. If there is no detected issue, then the monitoring cycle will repeat and send a clean report to the ship.

When distance support is used correctly, it complements the tenets of OA quite effectively—if upgrades and the necessary changes can be made to an open system. It allows for modularity and design reuse that can be distantly monitored and repaired rather than requiring a visit to port for the problem to be fixed. Additionally, in the figure above, remote monitoring can provide automation in the process of upgrades and repairs; and, automation (in most cases) leads to a decrease in number of employees, hence, an increase in a systems' Return on Investment.

H. Best Practices

1. Distance Support

a. Maintenance Free Operating Period (MFOP) and Acoustic Rapid Commercial Off-the-shelf (COTS) Insertion (ARCI)

In 2005, Naval Sea Systems Command (NAVSEA) completed a pilot program to test the feasibility of a Maintenance Free Operating Period (MFOP) on the Acoustic Rapid COTS Insertion (ACRI) System. The ARCI system was designed to replace the AN/BSY-1 and the AN/BQQ-5 on the Fleet's in-service submarines (688/688I/Trident/Seawolf) (Lockheed Martin, 2005). ARCI was a success in its own right in that it effectively demonstrated the use of OA with COTS technology on a large scale in the Fleet and allowed for technology insertion and refreshment (2005).

The ARCI MFOP program was conducted over a one-year time span; it tested the use of COTS technology and the COTS support provided to design ARCI in such a way to enable MFOP. Four platforms participated in the testing, and over the course of one year, no maintenance was required in any of the four. One resulting benefit of this test, for the purposes of this research, was that they implemented distance-support capabilities into the ARCI system before they conducted the test (NAVSEA Surface Warfare Logistics and Maintenance, 2005).

Most particularly, the following results are applicable for the formulation of "To Be" models for AEGIS software maintenance and upgrade:

> A database of maintenance related data was built into the ARCI system which provides the capability to perform statistical analysis of system performance and improve availability. An availability correlation function was developed to monitor system parameters and make recovery recommendations to system operators. […] An additional benefit of the MFOP Pilot Program was to develop and implement functionality in the ARCI system which further enables the system to be supported via Distance Support initiatives. (2005)

Using the advances outlined above, the "To Be" models were formulated, and the basis for those changes was grounded in research that has proven its MFOP reliability over the course of an entire year.

THIS PAGE INTENTIONALLY LEFT BLANK

III. Process Diagrams and "As Is" Models

A. Introduction

The Program Executive Office, Integrated Warfare Systems (PEO IWS), with its creation in 2002, began an initiative to implement open architecture (OA) throughout the Navy's Integrated Warfare Systems. One of the current initiatives is to implement OA into the sustaining engineering process associated with the AEGIS system.

Sustaining engineering in the AEGIS system was identified as an area of concern by a Component Business Model (CBM) analysis completed by IBM. The AEGIS software maintenance and upgrade process was further identified by the research team as an area in which the most improvements could be made by implementing an open architecture framework. In order to accomplish the implementation of open architecture into the AEGIS software maintenance and upgrade process, metrics must be determined to discover which areas of the software maintenance and upgrade process would be the best candidates for open architecture.

This proof of concept/case study will utilize information gathered from subject-matter experts (SMEs) in both the Surface Warfare Fleet and the Naval Surface Warfare Center (NSWC). The process information gathered from these subject-matter experts will be utilized to provide a Return on Investment (ROI) analysis using the Knowledge Value Added (KVA) methodology. This will be an analysis of the "As Is" system configuration or the baseline case. The ROI discovered through the KVA methodology will be analyzed to determine if open architecture could improve the sustaining engineering process. A Real Options (RO) analysis will be conducted in future research on the "To Be" software maintenance and upgrade process model in order to provide PEO IWS with options and risks for future courses of action.

B. Research Questions

Measures of effectiveness (MOE) for open architecture systems have been accurately derived through the Knowledge Value Added Methodology. This proof of concept was conducted in previous research by Capt. Joseph Uchytil, USMC, in his thesis, “Assessing the Operational Value of Situational Awareness for AEGIS and Ship Self Defense System (SSDS) Platforms through the Application of the Knowledge Value Added (KVA) Methodology.” This study is intended to draw on Capt. Uchytil’s proof of concept in order to answer the following research questions:

- Using IBM’s Component Business Modeling (CBM), what are the areas of highest concern and cost in the AEGIS weapons system as they relate to sustaining engineering?
- What are the “best practice” examples of sustaining engineering in the commercial or military environment, and how do they improve the processes of system development and acquisition lifecycles? What are the best examples of “design for maintenance” systems?
- What is the benefit of OA in the process of sustaining engineering? We will apply the KVA methodology to the areas of highest concern as identified by IBM in its CBM to address these questions.
- What overall numerical effect does OA have on Sustaining Engineering, and is it appreciable and low-risk enough to system development and to the DoD acquisition lifecycle?

C. Analysis and Data Collection

1. The Software Maintenance and Upgrade Process

The AEGIS software maintenance and upgrade process is a very complex method encapsulating a large number of processes in four main phases. The phases are the requirements definition phase, the design phase, the test phase, and the implementation or installation phase. These are the basic phases in any software maintenance/upgrade process—whether commercial, government, or non-profit. Depending on the type of upgrade required, the severity of the problem it is intended to fix, and the timeliness in which it is installed, the process can be slightly tailored to produce more immediate results.

One example of this process being tailored to fit a certain Fleet need occurs when an Emergent Update is required. The Emergent Update process is utilized for problems with software that are considered “showstoppers,” or a Priority 1A problem. Pending approval by both the project manager and the customer, the software maintenance and update process will be tailored to address only the Priority 1A process and will be completed in approximately one month. Emergent Updates happen on rare occasions; approximately 95% of AEGIS software upgrades go through the entire software maintenance and upgrade process.

The entire AEGIS software upgrade lifecycle is intended to take eighteen months, but typically takes closer to twenty-four months due to problems that are found during the testing phase or to failure of certifications. This software maintenance and upgrade process involves many sub-processes in each one of its main processes. These sub-processes may or may not have a bearing on the rest of the processes in the software maintenance and upgrade process. The fact that some of these sub-processes may be able to function in a stand-alone capacity makes the analysis of the software maintenance and upgrade process very difficult.

2. Data Collection Challenge

Due to the complex nature of the AEGIS software maintenance and upgrade process and the large number of people involved, collecting accurate data to be used by the KVA methodology proved to be a challenge. There were only a few SMEs who understood the complexity of the process. Also, outputs and learning time associated with each process and sub-process are not documented. This is coupled with the confusion that occurs between learning time and time spent in a Navy training course to learn the job. The Navy training courses are often of a uniform length of time (no matter the complexity of job), and subject-matter experts often confuse these training times with actual learning time. This leads to a slow data-collection process because the data needed for the KVA analysis is not readily available. There is also a need to separate the time spent in a Navy training course or school learning the specific process and time spent learning other skills. Due to these concerns, data collected for this analysis was collected through conversations and surveys given to subject-matter experts (SMEs). The data was then aggregated to capture the AEGIS software maintenance and upgrade process as a whole.

D. Defining the AEGIS Software Maintenance and Update Process

1. AEGIS Software Maintenance and Update Process Overview

The AEGIS software maintenance and update process takes place in two primary areas: on-ship and off-ship. The on-ship portion of the process takes place aboard an AEGIS-equipped US Naval vessel and is conducted by Surface Warfare Fleet personnel and various support personnel, including contractors. The on-ship portion of the software maintenance and update process deals with identifying problems that were not found in the testing phase of the process and also deals with installation and on-ship testing of the fielded AEGIS software update. The off-ship portion of the process takes place at Naval Surface Warfare Center, Dahlgren, VA. This primarily deals with the requirements definition phase, the design phase and the testing phase.

2. Defined AEGIS Software Maintenance and Update Processes

Since the AEGIS software maintenance and update process is a complex procedure involving may processes and sub-processes, it was necessary to breakdown the process into each of its individual processes and sub-processes. This breakdown allows for a more detailed analysis of each process contained within the software maintenance and update procedure.

The aggregate processes associated with the AEGIS software maintenance and update process are depicted in Figure 5. These processes were developed as a result of communication with several subject-matter experts. Although the process could differ for high-priority updates, such as an emergent update, the process depicted encompasses almost all AEGIS software maintenance and updates. Some sub-processes were captured within their larger process to provide a level of decomposition that was sufficient to produce accurate results for the KVA methodology.

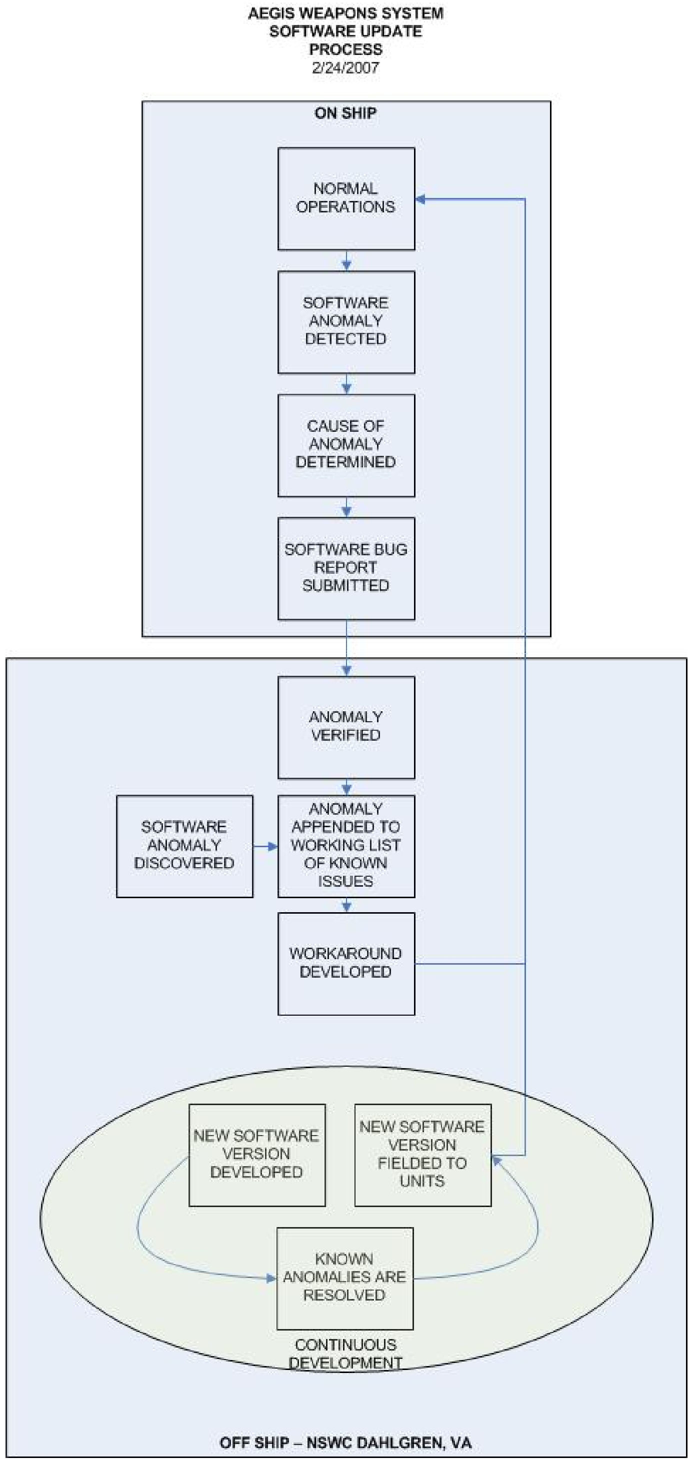


Figure 5. AEGIS Software Maintenance and Update Process (Aggregate Level)

E. On-ship Processes

The on-ship process reflects all of the processes in the AEGIS software maintenance and update process that would take place aboard a US Naval vessel. The detection of equipment and software failures and the effect of these failures on the mission capability of the AEGIS system are made by two departments. The anomaly identification and repair function works with the detection, reporting and resolution of software anomalies function. Casualty Report (CASREP) procedures, under normal ship operations, are explained below.

1. Software Anomaly Detected

Software anomalies can be detected through many methods. In most cases, the software and test engineers in the test phase of software maintenance and update process detect these anomalies. Software anomalies can also be detected during the combat systems integration testing after shipboard delivery. This is the first time that the software is in its shipboard configuration and allows for fully fielded software updates to be operationally tested. On rare occasions, software anomalies can also be detected during normal shipboard operation by Surface Warfare Fleet operators.

2. Cause of Anomaly Determined

Personnel who have observed the anomaly when the software is in its shipboard configuration attempt to collect data regarding the anomaly and also attempt to trace the anomaly to its source.

3. Software Bug Report Submitted

All data and information surrounding the anomaly and its source are collected. The configuration and status of the AEGIS system, as well as any environmental data, is also gathered and reported to the Program Manager.

F. Off-ship Aggregate Processes

The following represents an aggregate software maintenance and update procedure taken at the NSWC, Dahlgren, VA, to process software anomalies when an anomaly has been detected on software in its shipboard configuration. The aggregate process executed at NSWC, Dahlgren, VA, will be further decomposed and explained in the next section.

1. Software Anomaly Verified

The Project Manager and the project team work to recreate the conditions in a lab that were documented in the Software Bug Report in an effort to recreate the software anomaly. If the software anomaly is verified, the software maintenance and update process will continue.

2. Anomaly Appended to Working List of Known Issues

The software anomaly is documented in the Computer Program Change Request (CPCR). Software anomalies are tracked in the ACCESS/STARSY database. The CPCR is also assessed by the Joint Change Review Board (JCRB) and the Software Configuration Change Board (SCCB) for inclusion in the baseline.

3. Workaround Developed

If the CPCR is not corrected, then if a workaround exists to allow for avoidance of the anomaly, it is documented in the database and included in the Computer Program Design Document (CPDD)/Crew Brief.

4. New Software Version Developed

Teams of software programmers develop new versions of the software, which not only serve to fix anomalies, but also implement new functionality and exploit new technologies.

5. Directed Software Anomalies are Resolved

Anomalies that are evident in the new software and are found through certification testing in labs are resolved, and the required changes are made to the new software.

6. New Software Fielded to Units

Teams of contractors and support personnel arrive aboard ship to deliver new software, install the new software and conduct crew briefs and training.

G. Off-ship Process Decomposition

After the initial process model depicted above in Figure 5 was developed, it became apparent after numerous conversations with subject-matter experts that the most significant improvements to the software maintenance and update process would result in restructuring the off-ship component. Due to this, a higher level of decomposition was needed for the off-ship processes occurring at NSWC, Dahlgren, VA. Subject-matter experts were again consulted, and a more detailed process model for the AEGIS software maintenance and update process that occurred at NSWC, Dahlgren, VA, was developed. The more detailed off-ship AEGIS software maintenance and update process is depicted in Figure 6 and explained below.

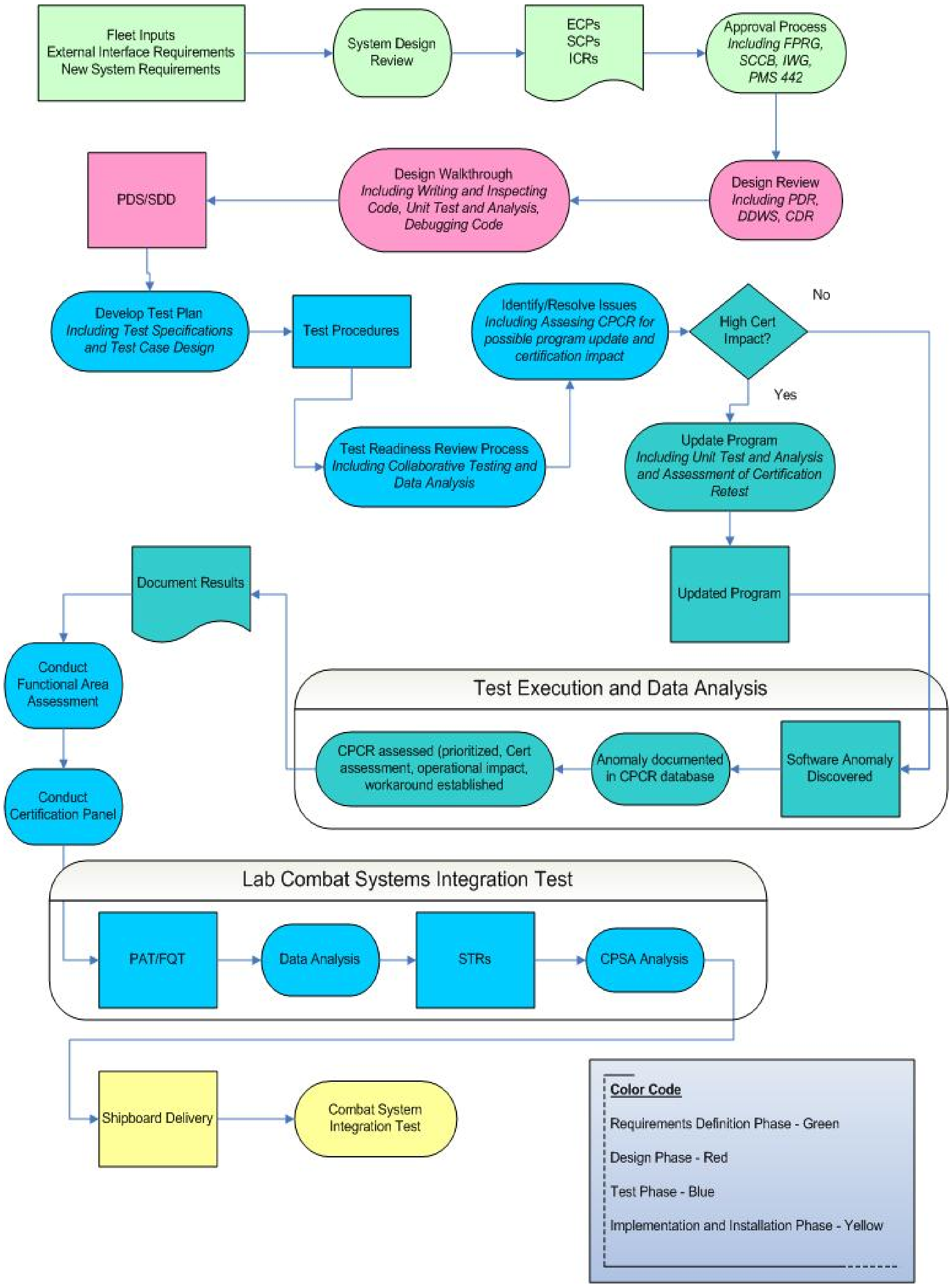


Figure 6. AEGIS Software Maintenance and Update Process (Off-ship)

1. Inputs and Requirements Gathered

Fleet inputs, external interface requirements and new system requirements are all gathered during this process.

2. System Design Review

This is a scheduled review that is technical in nature. Specifically:

> This review shall be conducted to evaluate the optimization, correlation, completeness, and risks associated with the allocated technical requirements. Also included is a summary review of the system engineering process which produced the allocated technical requirements of the engineering planning for the next phase of effort. (DoD, 1985)

3. ECPs, SCPs, ICRs

This process documents any changes that need to be made to the software after it has undergone the system design review. These are documented through an Engineering Change Proposal (ECP), Software Change Proposal (SCP) and an Interface Change Request (ICR).

4. Approval Process

The change proposals and requests are then sent through an approval process—in which the SCP awaits Software Configuration Change Board (SCCB) approval, and the ICR undergoes Integration Working Group (IWG) approval. The aggregate approvals are then sent to PMS 442 for approval. PMS 442 is the acquisition organization responsible for that product; in this case, it would be NAVSEA program management.

5. Design Review

This is the first step in the design phase of the AEGIS software maintenance and update process. The design review process includes a preliminary design review (PDR) and a Critical Design Review (CDR).

6. Design Walkthrough

The design walkthrough process includes writing and inspecting code, unit test and analysis and the debugging of code.

7. PDS/SDD

The previous design walkthrough process produces the Preliminary Design Specification (PDS) and the Software Design Document (SDD).

8. Develop Test Plan

This is the first process in the test phase of the software maintenance and update process. In this process, a test plan is developed using test specifications and the test case design process. For the purposes of this thesis, it is not necessary to describe the test case design process.

9. Test Procedures

The test procedures are the output of the previous test plan development process. These procedures will later be utilized in the test execution and data analysis process.

10. Test Readiness Review Process

The test plan and test procedures are reviewed in order to ensure that the test process will be most effective. In order to achieve maximum efficiency, collaborative testing and data analysis are included.

11. Identify/Resolve Issues

This process includes an assessment of any CPCRs for possible program update and also the certification impact of any CPCRs.

12. Certification Impact Decided

If any of the CPCRs are determined to have the potential for a high-certification impact, then the program must be updated before it can be sent to the test execution and data analysis portion of the software maintenance and update process. If the CPCRs are not determined to have a high-certification impact, the software is then sent directly to test execution and data analysis.

13. Update Program (High-certification Impact)

For software that contains CPCRs that are determined to have a high-certification impact, the software must be updated. This includes another unit test and analysis and an assessment of the certification retest.

14. Updated Program

Once the program is updated, and the unit test and analysis and an assessment of the certification retest have been conducted, the software is then sent to test execution and data analysis.

15. Test Execution and Data Analysis

The software is tested in lab conditions in order to detect any potential problems that might arise before the software is fully fielded. It consists of three sub-processes:

a. Software Anomaly Discovered

A software anomaly is found under lab conditions.

b. Anomaly Documented in CPCR Database

A CPCR is generated for the anomaly and is then entered in the ACCESS/STARSY database.

c. CPCR Assessed

The CPCR is prioritized, and its certification impact is assessed. The CPCR's operational impact is also assessed, and, if possible, a workaround is established.

16. Document Results

The results from the test execution and data analysis portion of the software maintenance and update process are documented.

17. Conduct Functional Area Assessment

This process is an analysis conducted at a higher level in order to prepare the software for the certification panel review.

18. Conduct Certification Panel

A certification panel assesses the software's results from the test execution and data analysis process and certifies the software for fielding.

19. PAT/FQT

A Preliminary Acceptance Test (PAT) and Functional Quality Testing (FQT) is then conducted on the software.

20. Data Analysis

The data collected from the PAT and the FQT is assessed and analyzed in preparation for the Lab Combat Systems Integration Test.

21. Lab Combat System Integration Test

This process includes any final testing that occurs in the lab environment, including any software trouble reports (STR) that are collected, a CPSA Analysis and a CPSA report.

22. Shipboard Delivery

The new software is fielded to operational units and installed by teams of contractors and support personnel. Crew briefs and training are also conducted.

23. Shipboard Combat System Integration Test

The software is tested in its fully fielded shipboard configuration. Due to the fact that this is the first time the software is in the shipboard configuration, it must be tested for functionality. This also ensures that it is interoperable with all combat systems already in place on the ship.

H. "As Is" KVA Analysis—On-ship and Off-ship Aggregate Process Model

An analysis of each sub-process in the AEGIS software maintenance and upgrade process for the on-ship and off-ship aggregate process model is provided in the following tables. The information provided for each analysis was produced through discussions with subject-matter experts. Each category for the KVA analysis is defined below.

1. Title of Head Process Executer

The "Title of Head Process Executer" category represents the job title of the person executing or overseeing the execution of the specific process or sub-process. The process executors' pay grade is indicated next to their job title. For purposes of this thesis, we used pay grades that erred on the high side to be conservative. If several executors with different pay grades were executing the same process, then the highest pay grade was used as a baseline for that process executor. This produces the most conservative KVA results. Some other basic assumptions for this category were:

- Each pay grade was to be at a Step 6 level within the respective pay grade.

- Base pay was location-adjusted for San Diego, CA, as a baseline.
- Civilian software contractor salary market comparables were estimated to be 175% of government salary.

2. Number of Employees

The "Number of Employees" category represents the number of government employees or contractors which are involved in the specific sub-process. If more than one person was involved in both the parent and specific sub-processes, that person is documented separately for each sub-process.

3. Rank Order of Difficulty

An ordinal ranking of the relative difficulty of learning each of the processes is collected and used to ensure that the "Relative Learning Time" and "Actual Average Training Period" estimates are reliable. By allowing the subject-matter experts to rank each of the sub-processes (1 being the least complex) outside of the context of time units, a correlation can be made between the "Rank Order of Difficulty" and the "Relative Learning Time." If a correlation of 80% is achieved, the results appear to be reliable, and the "Relative Learning Time" can be considered an accurate description of the relative difficulty of the sub-processes. If a correlation of 80% is not achieved, the results must be closely scrutinized, and the subject-matter experts must be resurveyed and possibly given a more in-depth explanation of the concept of "Relative Learning Time."

4. Relative Learning Time

The "Relative Learning Time" category represents a distributed relative amount of 100 hours of learning time among the processes. "Relative Learning Time" assumes an "average person" will learn all he/she needs to know to successfully complete all the tasks in each process. This learning time estimate includes the time it would take to learn how to produce the same output that any automation (e.g., information systems) currently produces. The 100-hour learning period is distributed according to how difficult and complex the processes are for the "average person" to learn. The purpose is to determine "Relative Learning Times" for each process given the 100-hour total. This helps identify the most complex processes and can be used as another internal reliability measure.

5. Actual Average Training Period

The "Actual Average Learning Time" is what the actual average training time in hours is for the "average person" for each process. This would be for a new employee with no background who would be required to learn everything to produce the outputs of the given processes. Learning time includes both formal training and on-the-job training.

The results from "Relative Learning Time" and "Actual Average Learning Time" are also correlated. If a correlation of 80% is achieved, the results appear to be reliable, and the "Actual Average Learning Time" can be considered an accurate description of the "Relative Learning Time" of the sub-processes. If a correlation of 80% is not achieved, the results must be closely scrutinized; in addition, the subject-matter experts must be resurveyed and possibly given a more in-depth explanation of the concept of "Relative Learning Time" along with "Actual Average Learning Time." In some cases, subject-matter experts may associate "Actual Average Learning Time" with a school or training period associated with the process. These schools and training periods are generally conducted over a uniform length of time and do not accurately reflect the "Actual Average Learning Time." Assumptions include:

- On-the-job training is estimated to be 50% of the time learning the task and 50% of the time actually performing the task.
- On-the-job training was conducted 8 hours a day, 5 days a week, and 50 weeks per year.
- On-the-job training occurs over a one-year period.

6. Percentage Automation

Each sub-process has a "Percentage Automation" associated with it between 0 and 100. This number captures the knowledge that is embedded in any information technology so it can be accounted for in later calculations. This number represents the percentage of information technology that it utilized so a process executor would not have to accomplish the task. For example, a process that has 100% automation would not require any process executors and would be accomplished fully by the automation tools listed for that process. If a process has 0% automation, no automation tools are utilized, and the process is totally executed by the process executors. These numbers are estimates based on subject-matter experts' observations and experience. One basic assumption associated with this:

- "Replacement Technology" is automation that will reduce the number of process executors associated with the process without increasing the output of the process.

7. Times Performed in a Year

The "Times Performed in a Year" category represents the number of times each sub-process is acted upon by a head process executor in a given year. The values were obtained by asking subject-matter experts for their inputs to determine a valid estimate for the year-long period.

8. Average Time to Complete

Each time a sub-process is acted upon (as indicated in the "Times Performed in a Year" category) there is a specific amount of time that it takes for each sub-process to be satisfactorily completed. This category represents the number of hours it takes a person trained in each process/sub-process to complete each task.

9. Automation Tools

The “Automation Tools” category represents any tools—such as MS Office, Visio, SIPR Database or Belvoir Paperless Office. This is used as a baseline for any automation tools that are already in use for the process and may provide insight for the implementation of other automation tools.

10. Total Learning Time (TLT)

This category is produced by dividing the “Actual Average Learning Time” by the “Percent Automation.” Because we assume “replacement technology,” the formula used to determine TLT is “Actual Average Learning Time”/(1-“Percent Automation”). This provides a total time, in hours, for each process, to include the learning time that is present in the automation tools. For example, if it takes one hour to learn a system that is 50% automated, then the total learning time associated with the process is two hours: one hour associated with the process executors and one hour associated with the automation tools.

11. Total Knowledge

This category is a representation, in hours, of all of the knowledge for each process that occurs over the one-year timeframe encompassed by the survey. The “Number of Employees” category is multiplied by the “Times Performed in a Year,” and the “Total Learning Time” categories.

12. Personnel Cost

This number represents the costs that are associated with the government employees associated with each process. This number is calculated by multiplying an employee’s hourly wage with the “Average Time to Complete,” the “Number of Employees,” and the “Times Preformed in a Year” categories. This number shows the cost of process only associated with personnel over the course of a year. Even though personnel may be involved with other tasks during their employment, this number shows the wages paid only based on their work with the specified process. Assumptions include:

- An average employee works 250 days per year.
- Wages are adjusted for the location of San Diego, CA.

13. Other Costs

The “Other Costs” category represents the cost of the process executors utilizing workstations that can access the Navy Marine Corps Intranet (NMCI). This category was calculated by taking the average cost per seat associated with NMCI and multiplying it by the “Number of Employees” category. Assumptions include (Navy Marine Corps Intranet):

- NMCI cost per seat is approximately $3000.
- Assumes a “Red Seat”—Pentium 800MHz. This provides performance for use with 2-D and light 3-D graphics or engineering-related applications, applications that require additional processing capability.

14. Total Costs

The “Total Costs” category represents the total costs of the process. This category was calculated by adding the “Personnel Costs” category with the “Other Costs” category.

15. Price

This category represents the price that it would cost if the process was executed by civilian contractors rather than government employees. The “Price” category was calculated much in the same way that the “Personnel Cost” category was calculated, except using contractor hourly wages in lieu of government employee hourly wages. This category was calculated by multiplying the contractor wage per hour by the “Number of Employees,” the

"Average Time to Complete," and the "Times Performed in a Year" categories. Assumptions include:

- Civilian software contractor market comparables were estimated to be 175% of government salary.

16. Denominator

This category shows the cost associated with producing the output of each process. It is the same as the "Total Costs" category.

17. Numerator

The "Numerator" category is the "percentage of the revenue or sales dollar allocated to the amount of knowledge required to obtain the outputs of a given process in proportion to the total amount of knowledge required to generate the corporation's salable outputs" (Housel & Bell, 2001, p. 45). For the purposes of this thesis, the revenue allocated to the amount of knowledge can be compared to the amount of knowledge that is present in each process or sub-process. This can also be thought of as the total knowledge multiplied by the price of each common unit. This value was calculated by first finding the price of each common unit. The price per common unit was calculated by dividing the total knowledge into the total price. The formula is: (price of the entire process)/(total knowledge of the entire process). The "Numerator" was then calculated by multiplying the total knowledge associated with each sub-process with the price of each common unit. An established price per common unit is important when developing the "To Be" model which will be discussed in Chapter IV.

I. ROK

With each process or sub-process, there is both a cost and a revenue associated with producing an output. The Return on Knowledge (ROK) provides a representation of how well the assets within a process are distributed in relation to one another by utilizing the costs and revenues associated with each sub-process. The ROK is calculated by dividing the “Numerator” by the “Denominator.” ROKs can be compared within a process to determine which processes are utilizing assets in an efficient manner and which processes need to be changed, perhaps by the utilization of automation tools, in order to improve efficiency. Although the ROK is a very valuable tool, a low ROK does not dictate that a process is in need of increased automation, but serves as an indicator that the process should be analyzed more closely to discover if process efficiency can be improved.

J. ROI

“ROI,” or return on investment, is a common accounting term that is widely understood by the financial community. For this reason, it is a slightly more meaningful number than ROK. Essentially, it is a very similar number to ROK, just derived by a different unit of measure. In financial terms, ROI is the profit or loss resulting from an investment transaction, usually expressed as an annual percentage return. ROI is a return ratio that contrasts the net benefits of a project to its total costs. In financial terms, ROI is calculated by profit minus investment all divided by investment. For the purposes of KVA, ROI is calculated by the “Numerator” minus the “Denominator” all dived by “Denominator.”   Much like ROKs, ROIs can be compared within a process to determine which processes are utilizing assets in an efficient manner and which processes need to be changed, perhaps by the utilization of automation tools, in order to improve efficiency.

K. “As Is” Process Data—On-ship and Off-ship Aggregate Process Model

Each of the AEGIS software maintenance and upgrade processes and sub-processes will be presented for evaluation. The ROKs and ROIs that were calculated for each process and sub-process will be utilized to find the differences in efficiency for each of the processes and sub-processes. The analysis of the “As Is” process data will provide insight on the amount and location of knowledge assets throughout the AEGIS software maintenance and upgrade procedure.

1. On- and Off-ship Aggregate Process

Tables 3 and 4 depict the on- and off-ship aggregate processes included in the KVA analysis for the AEGIS software maintenance and upgrade process that occurs for one AEGIS ship.

Process	Title of Head Process Executer	Number of Employees	Rank Order of Difficulty	Relative Learning Time	Actual Average Learning Time	Percentage Automation	Times Performed In a Year	Average Time to Complete	Automation Tools
Software Anomaly Detected	Project Manager (GS-11)	2	4	10	300	0%	10	20	N/A
Cause of Anomaly Determined	Technology Director (GS-12/13)	5	8	20	500	30%	25	40	Advanced Software
Software Bug Report Submitted	Project Manager (GS-11)	1	3	10	300	10%	10	4	MS Word
Anomaly Verified	Project Manager (GS-11/12)	2	5	10	300	0%	10	20	N/A
Anomaly Appended to Working List of Known Issues	Fleet Support (GS-9/10)	1	2	5	160	10%	10	4	Excel
Workaround Developed	Project Manager (GS-11/12)	2	6	10	160	0%	10	20	N/A
New Software Version Developed	Lead Programmer (GS-13/14)	10	9	20	640	20%	2	200	Compiler
Known Anomalies are Resolved	Lead Programmer (GS-12/13)	5	7	10	500	20%	2	200	N/A
New Software Version Fielded to Units	Fleet Support (GS-9/10)	3	1	5	160	10%	20	80	Tracking
		Correlation (Ordinal to RLT)	0.877032523						
		Correlation (RLT to AATP)	0.845793835						

Process (continued)	TLT	Total Knowledge	Personel Cost	Other Costs	Total Costs	Price	Denominator	Numerator	ROK	ROI
Software Anomaly Detected	**300**	6000	$ 13,701.00	$ 600.00	$ 14,301.00	$ 23,976.75	14301	56826.27304	397%	**297%**
Cause of Anomaly Determined	**714**	89286	$ 244,096.88	$ 7,500.00	$ 251,596.88	$ 427,169.53	251596.875	845629.0632	336%	**236%**
Software Bug Report Submitted	**333**	3333	$ 1,370.10	$ 60.00	$ 1,430.10	$ 2,397.68	1430.1	31570.15169	2208%	**2108%**
Anomaly Verified	**300**	6000	$ 16,421.50	$ 600.00	$ 17,021.50	$ 28,737.63	17021.5	56826.27304	334%	**234%**
Anomaly Appended to Working List of Known Issues	**178**	1778	$ 1,247.00	$ 60.00	$ 1,307.00	$ 2,182.25	1307	16837.41424	1288%	**1188%**
Workaround Developed	**160**	3200	$ 16,421.50	$ 600.00	$ 17,021.50	$ 28,737.63	17021.5	30307.34562	178%	**78%**
New Software Version Developed	**800**	16000	$ 230,750.00	$ 6,000.00	$ 236,750.00	$ 403,812.50	236750	151536.7281	64%	**-36%**
Known Anomalies are Resolved	**625**	6250	$ 97,638.75	$ 3,000.00	$ 100,638.75	$ 170,867.81	100638.75	59194.03442	59%	**-41%**
New Software Version Fielded to Units	**178**	10667	$ 149,640.00	$ 7,200.00	$ 156,840.00	$ 261,870.00	156840	101024.4854	64%	**-36%**
Totals		142513	$ 771,286.73	$ 25,620.00	$ 796,906.73	$ 1,349,751.77	796906.725	1349751.769	169%	**69%**

Personel Costs	Base Pay	Location Adjusted	**Hourly Wage**	**Contractor Wage**
GS 9	$ 45,294.00	$ 56,617.50	$ 28.31	$ 49.54
GS 10	$ 49,880.00	$ 62,350.00	$ 31.18	$ 54.56
GS 11	$ 54,804.00	$ 68,505.00	$ 34.25	$ 59.94
GS 12	$ 65,686.00	$ 82,107.50	$ 41.05	$ 71.84
GS 13	$ 78,111.00	$ 97,638.75	$ 48.82	$ 85.43
GS 14	$ 92,300.00	$ 115,375.00	$ 57.69	$ 100.95

Price Per Common Unit $ 9.47

Table 3. Off-ship Aggregate Process (One ship)

2. On- and Off-ship Aggregate Process for Entire AEGIS Fleet

Table 4 depicts the further decomposed off-ship process included in the KVA analysis for the AEGIS software maintenance and upgrade procedure that is scaled to include all AEGIS ships in the US Fleet. Assumptions for scaling the process data include:

- Updates are delivered to each ship twice a year.
- Each update contains five anomaly fixes.
- The “New Software Version Fielded to Units” occurs 168 times per year based on 84 AEGIS ships and two updates per year.

Process	Title of Head Process Executer	Number of Employees	Rank Order of Difficulty	Relative Learning Time	Actual Average Learning Time	Percentage Automation	Times Performed In a Year	Average Time to Complete	Automation Tools
Software Anomaly Detected	Project Manager (GS-11)	2	4	10	300	0%	10	20	N/A
Cause of Anomaly Determined	Technology Director (GS-12/13)	5	8	20	500	30%	25	40	Advanced Software
Software Bug Report Submitted	Project Manager (GS-11)	1	3	10	300	10%	10	4	MS Word
Anomaly Verified	Project Manager (GS-11/12)	2	5	10	300	0%	10	20	N/A
Anomaly Appended to Working List of Known Issues	Fleet Support (GS-9/10)	1	2	5	160	10%	10	4	Excel
Workaround Developed	Project Manager (GS-11/12)	2	6	10	160	0%	10	20	N/A
New Software Version Developed	Lead Programmer (GS-13/14)	10	9	20	640	20%	2	200	Compiler
Known Anomalies are Resolved	Lead Programmer (GS-12/13)	5	7	10	500	20%	2	200	N/A
New Software Version Fielded to Units	Fleet Support (GS-9/10)	3	1	5	160	10%	20	80	Tracking
		Correlation (Ordinal to RLT)	0.877032523						
		Correlation (RLT to AATP)	0.845793835						

Process (continued)	TLT	Total Knowledge	Personel Cost	Other Costs	Total Costs	Price	Denominator	Numerator	ROK	ROI
Software Anomaly Detected	**300**	6000	$ 13,701.00	$ 600.00	$ 14,301.00	$ 23,976.75	14301	4773406.936	33378%	**33278%**
Cause of Anomaly Determined	**714**	89286	$ 244,096.88	$ 7,500.00	$ 251,596.88	$ 427,169.53	251596.875	71032841.31	28233%	**28133%**
Software Bug Report Submitted	**333**	3333	$ 1,370.10	$ 60.00	$ 1,430.10	$ 2,397.68	1430.1	2651892.742	185434%	**185334%**
Anomaly Verified	**300**	6000	$ 16,421.50	$ 600.00	$ 17,021.50	$ 28,737.63	17021.5	4773406.936	28043%	**27943%**
Anomaly Appended to Working List of Known Issues	**178**	1778	$ 1,247.00	$ 60.00	$ 1,307.00	$ 2,182.25	1307	1414342.796	108213%	**108113%**
Workaround Developed	**160**	3200	$ 16,421.50	$ 600.00	$ 17,021.50	$ 28,737.63	17021.5	2545817.032	14956%	**14856%**
New Software Version Developed	**800**	16000	$ 230,750.00	$ 6,000.00	$ 236,750.00	$ 403,812.50	236750	12729085.16	5377%	**5277%**
Known Anomalies are Resolved	**625**	6250	$ 97,638.75	$ 3,000.00	$ 100,638.75	$ 170,867.81	100638.75	4972298.891	4941%	**4841%**
New Software Version Fielded to Units	**178**	10667	$ 149,640.00	$ 7,200.00	$ 26,349,120.00	$ 261,870.00	26349120	8486056.775	32%	**-68%**
Totals		142513	$ 771,286.73	$ 25,620.00	$ 26,989,186.73	$ 1,349,751.77	26989186.73	113379148.6	420%	**320%**

Personel Costs	Base Pay	Location Adjusted	**Hourly Wage**	**Contractor Wage**
GS 9	$ 45,294.00	$ 56,617.50	$ 28.31	$ 49.54
GS 10	$ 49,880.00	$ 62,350.00	$ 31.18	$ 54.56
GS 11	$ 54,804.00	$ 68,505.00	$ 34.25	$ 59.94
GS 12	$ 65,686.00	$ 82,107.50	$ 41.05	$ 71.84
GS 13	$ 78,111.00	$ 97,638.75	$ 48.82	$ 85.43
GS 14	$ 92,300.00	$ 115,375.00	$ 57.69	$ 100.95

Price Per Common Unit $ 9.47

Table 4. Off-ship Aggregate Process (All Ships)

3. Off-ship Process Decomposition

Table 5 depicts the further decomposed off-ship process included in the KVA analysis for the AEGIS software maintenance and upgrade procedure that occurs for one AEGIS ship.

Process	Title of Head Process Executer	Number of Employees	Rank Order of Difficulty	Relative Learning Time	Actual Average Learning Time	Percentage Automation	Times Performed In a Year	Average Time to Complete	Automation Tools	Cost of IT
Fleet Inputs/External Interface Requirements/New System Resources	Program Manager (ND-IV)	15	2	1	500	15%	3	335	Databases to ID issues for inclusion/fleet issues/etc	10000
System Design Review	Systems Engineer (ND-IV)	50	4	2	500	5%	1	335	database/Adjudication	10000
ECP's/SCP's/ICR's	Systems Engineer (ND-IV)	25	1	1	250	5%	25	60	ACSIS	50000
Approval Process (Including FPRG, SCCB, IWG, PMS 422)	PM , PMS 422, LSEA (ND-V)	50	8	8	835	15%	50	8	ACSIS	45000
Design Review (Including PDR, DDWS, CDR)	PM (ND-IV)	50	5	5	500	5%	3	120	Comment database/Adjudication tracker	10000
Design Walkthrough (Including Writing and Inspecting Code, Unit Test and Analysis, Debugging Code)	Devloper (ND-IV)	8	7	7	1000	15%	50	80	MS Word	0
Develop Test Plan (Including Test Specifications and Test Case Design)	Test Engineer (ND-IV)	2	12	10	1000	5%	50	1000	MS Word	0
Test Procedures	Test Engineer (ND-IV)	2	13	10	1000	5%	50	1000	MS Word	0
Test Readiness Review	PM (ND-IV)	30	3	1	500	5%	3	16	MS Powerpoint	0
st Execution and Data Analysis (Software anomaly detected, anomaly documented in CPCR, CPCR assess	Engineers (ND-IV)	50	14	10	1000	0%	50	665	Test Director and TERMS are used as respository	0
Document Results	Engineers (ND-IV)	25	6	5	500	10%	50	665	Test Director and TERMS are used as respository	0
Identify/Resolve Issues (Including assessing CPCR for possible program update and certification impact)	Engineers (ND-IV)	25	15	10	1000	0%	50	40	ACSIS	50000
High Cert Impact (Yes/No branch, No 95% of time)	Engineers (ND-IV)	30	10	10	1000	10%	50	8	ACSIS	50000
Conduct Functional Area Assessment (NO BRANCH)	Test IPT Lead (ND-IV)	15	11	10	1000	0%	50	120	MS Excel	0
Conduct Certification Panel (NO BRANCH)	PM (ND-V)	30	9	10	1000	0%	1	16	MS Powerpoint	0
		Correlation (Ordinal to RLT)	0.930399929	100						
		Correlation (RLT to AATP)	0.931962825							

Process (continued)	TLT	Total Knowledge	Personel Cost	Other Costs	Total Costs	Price	Denominator	Numerator	ROK	ROI
Fleet Inputs/External Interface Requirements/New System Resources	**588**	26471	$ 729,385.03	$ 32,612.50	$ 761,997.53	$ 1,276,423.80	$ 761,997.53	$ 647,266.17	85%	-15%
System Design Review	**526**	26316	$ 810,427.81	$ 35,125.00	$ 845,552.81	$ 1,418,248.67	$ 845,552.81	$ 643,480.99	76%	-24%
ECP's/SCP's/ICR's	**263**	164474	$ 1,814,390.63	$ 106,250.00	$ 1,920,640.63	$ 3,175,183.59	$ 1,920,640.63	$ 4,021,756.20	209%	109%
Approval Process (Including FPRG, SCCB, IWG, PMS 422)	**982**	2455882	$ 1,357,050.00	$ 75,000.00	$ 1,432,050.00	$ 2,374,837.50	$ 1,432,050.00	$ 60,051,917.22	4193%	4093%
Design Review (Including PDR, DDWS, CDR)	**526**	78947	$ 870,907.50	$ 37,000.00	$ 907,907.50	$ 1,524,088.13	$ 907,907.50	$ 1,930,442.97	213%	113%
Design Walkthrough (Including Writing and Inspecting Code, Unit Test and Analysis, Debugging Code)	**1176**	470588	$ 1,548,280.00	$ 48,000.00	$ 1,596,280.00	$ 2,709,490.00	$ 1,596,280.00	$ 11,506,954.20	721%	621%
Develop Test Plan (Including Test Specifications and Test Case Design)	**1053**	105263	$ 4,838,375.00	$ 150,000.00	$ 4,988,375.00	$ 8,467,156.25	$ 4,988,375.00	$ 2,573,923.97	52%	-48%
Test Procedures	**1053**	105263	$ 4,838,375.00	$ 150,000.00	$ 4,988,375.00	$ 8,467,156.25	$ 4,988,375.00	$ 2,573,923.97	52%	-48%
Test Readiness Review	**526**	47368	$ 69,672.60	$ 2,160.00	$ 71,832.60	$ 121,927.05	$ 71,832.60	$ 1,158,265.78	1612%	1512%
st Execution and Data Analysis (Software anomaly detected, anomaly documented in CPCR, CPCR assess	**1000**	2500000	$ 80,437,984.38	$ 2,493,750.00	$ 82,931,734.38	$ 140,766,472.66	$ 82,931,734.38	$ 61,130,694.18	74%	-26%
Document Results	**556**	694444	$ 40,218,992.19	$ 1,246,875.00	$ 41,465,867.19	$ 70,383,236.33	$ 41,465,867.19	$ 16,980,748.38	41%	-59%
Identify/Resolve Issues (Including assessing CPCR for possible program update and certification impact)	**1000**	1250000	$ 2,419,187.50	$ 125,000.00	$ 2,544,187.50	$ 4,233,578.13	$ 2,544,187.50	$ 30,565,347.09	1201%	1101%
High Cert Impact (Yes/No branch, No 95% of time)	**1111**	1666667	$ 580,605.00	$ 68,000.00	$ 648,605.00	$ 1,016,058.75	$ 648,605.00	$ 40,753,796.12	6283%	6183%
Conduct Functional Area Assessment (NO BRANCH)	**1000**	750000	$ 4,354,537.50	$ 135,000.00	$ 4,489,537.50	$ 7,620,440.63	$ 4,489,537.50	$ 18,339,208.25	408%	308%
Conduct Certification Panel (NO BRANCH)	**1000**	30000	$ 32,569.20	$ 720.00	$ 33,289.20	$ 56,996.10	$ 33,289.20	$ 733,568.33	2204%	2104%
Totals		10371684	$ 144,920,739.33	$ 4,705,492.50	$ 149,626,231.83	$ 253,611,293.83	$ 149,626,231.83	$ 253,611,293.83	169%	69%

Personel Costs	Base Pay	Location Adjusted	**Hourly Wage**	**Contractor Wage**
ND-III	$ 55,585.00	$ 69,481.25	$ 34.74	$ 60.80
ND-IV	$ 77,414.00	$ 96,767.50	$ 48.38	$ 84.67
ND-V	$ 108,564.00	$ 135,705.00	$ 67.85	$ 118.74

Price Per Common Unit $ 24.45

Table 5. Decomposed Off-ship Process (One Ship)

IV. “To Be” Results

A. “To Be” KVA Analysis—On-ship and Off-ship Aggregate Process Model

The “To Be” analysis is a hypothetical improved process model of the possible effects of a future Open Architecture (OA)-enabled AEGIS software maintenance and upgrade process. The processes and sub-processes that have been identified as having the potential for using OA and Distance Support have been modified to reflect the improvements. The potential improvements are described in detail in the following sections. The OA-enabled processes and sub-processes were developed using current distance-support policy, Maintenance Free Operating Period (MFOP) research, and suggestions from subject-matter experts (SMEs).

B. Open Architecture Reengineering

The Naval Surface Warfare Center (NWSC) Port Hueneme, CA, has been developing the concept of remote maintenance of Cooperative Engagement Capability (CEC) on ships. Through the implementation of an OA-based system, the concept of remote maintenance could greatly improve the AEGIS software maintenance and upgrade process. The NWSC Port Hueneme (2006) explains:

> Remote maintenance enables the Navy to provide distance support with fielded systems and enhances the ability to meet requirements of reduced manpower, lower cost, and faster more efficient troubleshooting and repair. It works within the initiatives to return ships to the Strike Group faster and provides safe, effective and affordable combat systems into future designs.

The AEGIS weapons system is already equipped with a link to provide distance support. “Currently installed as a fielded Ship Alteration (SHIPALT), Operational Readiness Test System Tech Assist Remote Support (ORTSTARS) allows AEGIS ships to establish a secure link between the Operational

Readiness Test System (ORTS) and any shore facility equipped with SIPERNET" (2006). This concept was used in the development of the "To Be" model for the AEGIS software maintenance and upgrade process.

The Undersea Warfare community has also provided valuable assistance through its research with the MFOP concept that has been incorporated into the Acoustic Rapid COTS Insertion (ARCI) program. The ARCI program incorporated hardware, software and COTS logistics support and technology to achieve a MFOP for 90 days. "An additional benefit of the MFOP Pilot Program was to develop and implement functionality in the ARCI system which further enables the system to be supported via Distance Support initiatives" (NAVSEA Surface Warfare Logistics and Maintenance, 2005). These important principles helped guide the reengineering of the "To Be" model for the AEGIS software maintenance and upgrade process.

Through an analysis of the "As Is" process data, it was determined that the most advantageous improvements to the AEGIS software maintenance and upgrade process could be achieved through implementing the OA tenets of scalability and portability. The analysis was concentrated on the Return on Investment (ROI) findings from the "As Is" process data, but also heavily considered the concepts of distance support and, in turn, of a Maintenance Free Operating Period (MFOP). Reengineering the process provided increases in the ROI for the sub-processes that were affected and also produced an increase in total ROI. Assumptions for the process reengineering include:

- The use of middleware was necessary until Category 4 OACE level could be reached.
- No process would become fully 100% automated.
- One employee would always be on hand as a supervisor to even a mostly automated process.

- The "Average Time to Complete" for the "New Software Version Fielded to Units" was estimated to be 15 minutes using the distance-support concept.
- "Replacement Technology" will be used instead of "Additive Technology."
- Development costs were not included because they are distributed throughout the lifecycle of the system.

C. "To Be" Process Data

In the following process model and Knowledge Value Added (KVA) analysis, it was necessary to estimate changes in the "Percentage Automation" category due to the fact that the "To Be" model is hypothetical. It was also considered that at least one human employee would oversee each of the sub-processes even though the potential for the sub-processes to be totally automated exists.

In an OA-enabled AEGIS software maintenance and upgrade process, software updates can be made available by a push or pull method. In the pull method, the user would download updates and then install them. In the push method, the software would be pushed to the network node remotely, thereby reducing the need for onsite personnel. These software updates would take place through the secure link provided by ORTSTARS. The ability of updates to be made readily available is an inherent benefit of open architecture systems and serves to reduce the number of personnel hours spent on the AEGIS software maintenance and upgrade process. This would also serve to decrease the number of on-site personnel and to increase the speed of upgrade.

The processes "Software Anomaly Detected," "Cause of Anomaly Determined," "Software Bug Report Submitted," and "New Software Version Fielded to Units" were determined to be the most amenable to change in the OA-enabled method. In an OA-enabled AEGIS environment, diagnostics from a single or multiple locations can be used if there is a problem at the first tier of support prior to dispatching personnel. Eliminating on-site requirements and the need for multiple on-site maintenance personnel will drive down costs and improve operational availability. These three sub-processes would change into two sub-processes: “Remote Diagnostics Detect/Fix Anomaly” and “Remote Diagnostics Submit Software Bug Report for Anomaly.” The aggregate processes that were changed in the OA-enabled AEGIS software maintenance and update process are depicted in Figure 7.

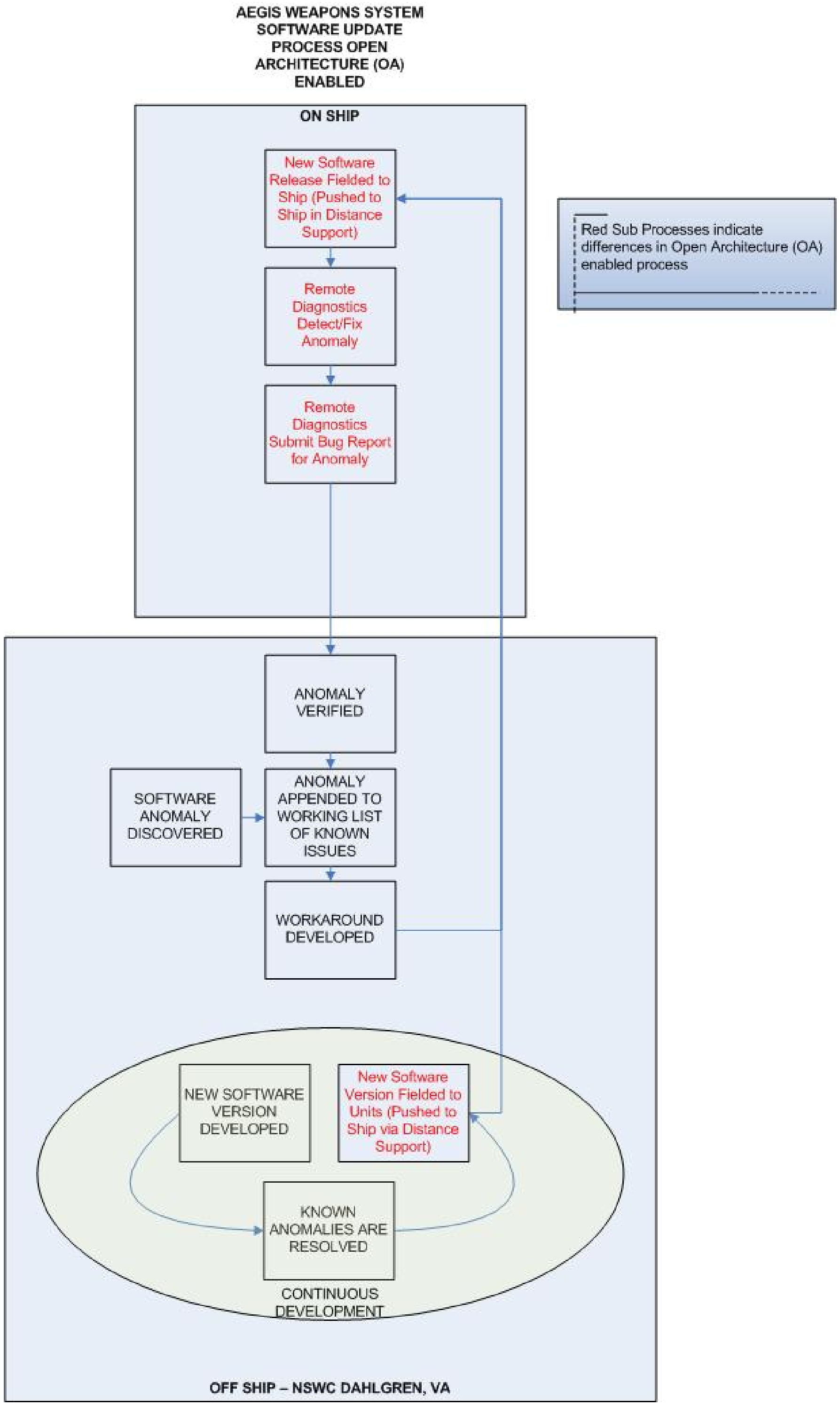


Figure 7. OA-enabled AEGIS Software Maintenance and Upgrade Process (Aggregate Level)

1. Remote Diagnostics Detect/Fix Anomaly

The transfer to OA in new systems and converting to OA in old, closed systems enables the use of remote diagnostics in the "To Be" process model. Through ORTSTARS, a remote diagnostic can potentially identify a software anomaly before an operator on the ship can identify the same anomaly. The remote diagnostics, then, could record the circumstances surrounding the anomaly and compare them to similar Computer Program Change Requests (CPCRs) managed in the ACCESS/STARSY database. If a CPCR is found that closely matches the anomaly detected, the remote diagnostics could then take appropriate actions already listed in the ACCESS/STARSY database to fix the anomaly.

The increase in remote diagnostics capabilities through the implementation of OA would drastically reduce the number of personnel required to complete the process. In the "As Is" model, the "Remote Diagnostics Detect/Fix Anomaly" was two separate processes: "Software Anomaly Detected" and "Cause of Anomaly Determined." Distance Support allowed these two processes to be combined into one process and reduced the number of personnel required to complete the processes from seven employees to one employee. This allows for the process to become 90% automated while still preserving some human intervention.

2. Remote Diagnostics Submit Software Bug Report for Anomaly

The usage of OA allows for the use of remote diagnostics to submit a software bug report. Again through ORTSTARS, the software bug report could be submitted through a secure link in real time; in addition, the software bug report has potential to be a more thorough representation of the circumstances surrounding the anomaly than those reported by personnel, as no human interpretation would be required. The process still retains some human intervention as one process executor would still oversee the process. This combination of OA and remote diagnostics could also drastically reduce cycle-time—normally delayed by submission of the software bug report.

3. New Software Version Fielded to Units (Pushed to Ship via Distance Support)

In the OA-enabled model for the AEGIS software maintenance and upgrade process, new software updates could be fielded to the ship through ORTSTARS in either the push method or the pull method. This would allow for greatly reduced cycle-time in the fielding of new software in its shipboard configuration. Remote diagnostics could also perform the functions involved in the "Combat System Integration Test," further reducing cycle-time. Software fielding through distance support and the push/pull method would also reduce the number of personnel required to field the software to the unit from three employees to one employee. The one process executor would still remain available to oversee the process and resolve any issues, via distance support, that the ship may encounter once the software has been fielded in its shipboard configuration.

4. OA-enabled On- and Off-ship Aggregate Process

Table 6 depicts the OA-enabled on- and off-ship process included in the KVA analysis for the OA-enabled AEGIS software maintenance and upgrade process that occurs for one AEGIS ship.

Process	Title of Head Process Executer	Number of Employees	Rank Order of Difficulty	Relative Learning Time	Actual Average Learning Time	Percentage Automation	Times Performed In a Year	Average Time to Complete	Automation Tools
New Release Fielded (Pushed to Ship via Distance Support)	Fleet Support (GS-9/10)	1	4	10	300	90%	10	20	N/A
Remote Diagnostics Detect/Fix Anomaly	Technology Director (GS-12/13)	1	8	20	500	95%	25	40	Advanced Software
Remote Diagnostics Submit Software Bug Report for Anomaly	Project Manager (GS-11)	1	3	10	300	95%	10	4	MS Word
Anomaly Verified	Project Manager (GS-11/12)	2	5	10	300	0%	10	20	N/A
Anomaly Appended to Working List of Known Issues	Fleet Support (GS-9/10)	1	2	5	160	10%	10	4	Excel
Workaround Developed	Project Manager (GS-11/12)	2	6	10	160	0%	10	20	N/A
New Software Version Developed	Lead Programmer (GS-13/14)	10	9	20	640	20%	2	200	Compiler
Known Anomalies are Resolved	Lead Programmer (GS-12/13)	5	7	10	500	20%	2	200	N/A
New Software Version Fielded to Units (Pushed to Ship via Distance Support)	Fleet Support (GS-9/10)	1	1	5	160	90%	20	0.25	Tracking
		Correlation (Ordinal to RLT)	0.877032523						
		Correlation (RLT to AATP)	0.845793835						

Process (continued)	TLT	Total Knowledge	Personel Cost	Other Costs	Total Costs	Price	Denominator	Numerator	ROK	ROI
New Release Fielded (Push to Ship via Distance Support)	**3000**	30000	$ 6,850.50	$ 300.00	$ 7,150.50	$ 10,911.25	7150.5	284131.3652	3974%	**3874%**
Remote Diagnostics Detect/Fix Anomaly	**10000**	250000	$ 48,819.38	$ 1,500.00	$ 50,319.38	$ 85,433.91	50319.375	2367761.377	4705%	**4605%**
Remote Diagnostics Submit Software Bug Report for Anomaly	**6000**	60000	$ 1,370.10	$ 60.00	$ 1,430.10	$ 2,397.68	1430.1	568262.7304	39736%	**39636%**
Anomaly Verified	**300**	6000	$ 16,421.50	$ 600.00	$ 17,021.50	$ 28,737.63	17021.5	56826.27304	334%	**234%**
Anomaly Appended to Working List of Known Issues	**178**	1778	$ 1,247.00	$ 60.00	$ 1,307.00	$ 2,182.25	1307	16837.41424	1288%	**1188%**
Workaround Developed	**160**	3200	$ 16,421.50	$ 600.00	$ 17,021.50	$ 28,737.63	17021.5	30307.34562	178%	**78%**
New Software Version Developed	**800**	16000	$ 230,750.00	$ 6,000.00	$ 236,750.00	$ 403,812.50	236750	151536.7281	64%	**-36%**
Known Anomalies are Resolved	**625**	6250	$ 97,638.75	$ 3,000.00	$ 100,638.75	$ 170,867.81	100638.75	59194.03442	59%	**-41%**
New Software Version Fielded to Units (Pushed to Ship via Distance Support)	**1600**	32000	$ 155.88	$ 7.50	$ 163.38	$ 272.78	163.375	303073.4562	185508%	**185408%**
Totals		405228	$ 419,674.60	$ 12,127.50	$ 431,802.10	$ 733,353.43	431802.1	3837930.724	889%	**789%**

Personel Costs	Base Pay	Location Adjusted	**Hourly Wage**	**Contractor Wage**
GS 9	$ 45,294.00	$ 56,617.50	$ 28.31	$ 49.54
GS 10	$ 49,880.00	$ 62,350.00	$ 31.18	$ 54.56
GS 11	$ 54,804.00	$ 68,505.00	$ 34.25	$ 59.94
GS 12	$ 65,686.00	$ 82,107.50	$ 41.05	$ 71.84
GS 13	$ 78,111.00	$ 97,638.75	$ 48.82	$ 85.43
GS 14	$ 92,300.00	$ 115,375.00	$ 57.69	$ 100.95

Price Per Common Unit $ 9.47

Table 6. KVA OA-enabled (One Ship)

5. OA-enabled On- and Off-ship Aggregate Process for Entire AEGIS Fleet

Table 7 depicts the further decomposed off-ship process included in the KVA analysis for the AEGIS software maintenance and upgrade process that is scaled to include all AEGIS ships in the US Fleet. Assumptions for scaling the process data include:

- Each software update is fielded to each of the 84 AEGIS ships on a case-by-case basis.
- Anomalies are resolved on a case-by-case basis.

Process	Title of Head Process Executer	Number of Employees	Rank Order of Difficulty	Relative Learning Time	Actual Average Learning Time	Percentage Automation	Times Performed In a Year	Average Time to Complete	Automation Tools
New Release Fielded (Pushed to Ship via Distance Support)	Fleet Support (GS-9/10)	1	4	10	300	90%	10	20	N/A
Remote Diagnostics Detect/Fix Anomaly	Technology Director (GS-12/13)	1	8	20	500	95%	25	40	Advanced Software
Remote Diagnostics Submit Software Bug Report for Anomaly	Project Manager (GS-11)	1	3	10	300	95%	10	4	MS Word
Anomaly Verified	Project Manager (GS-11/12)	2	5	10	300	0%	10	20	N/A
Anomaly Appended to Working List of Known Issues	Fleet Support (GS-9/10)	1	2	5	160	10%	10	4	Excel
Workaround Developed	Project Manager (GS-11/12)	2	6	10	160	0%	10	20	N/A
New Software Version Developed	Lead Programmer (GS-13/14)	10	9	20	640	20%	2	200	Compiler
Known Anomalies are Resolved	Lead Programmer (GS-12/13)	5	7	10	500	20%	2	200	N/A
New Software Version Fielded to Units (Pushed to Ship via Distance Support)	Fleet Support (GS-9/10)	1	1	5	160	90%	20	0.25	Tracking
		Correlation (Ordinal to RLT)	0.877032523						
		Correlation (RLT to AATP)	0.845793835						

Process (continued)	TLT	Total Knowledge	Personel Cost	Other Costs	Total Costs	Price	Denominator	Numerator	ROK	ROI
New Release Fielded (Push to Ship via Distance Support)	**3000**	30000	$ 6,850.50	$ 300.00	$ 7,150.50	$ 10,911.25	7150.5	23867034.68	333781%	**333681%**
Remote Diagnostics Detect/Fix Anomaly	**10000**	250000	$ 48,819.38	$ 1,500.00	$ 50,319.38	$ 85,433.91	50319.375	198891955.7	395259%	**395159%**
Remote Diagnostics Submit Software Bug Report for Anomaly	**6000**	60000	$ 1,370.10	$ 60.00	$ 1,430.10	$ 2,397.68	1430.1	47734069.36	3337813%	**3337713%**
Anomaly Verified	**300**	6000	$ 16,421.50	$ 600.00	$ 17,021.50	$ 28,737.63	17021.5	4773406.936	28043%	**27943%**
Anomaly Appended to Working List of Known Issues	**178**	1778	$ 1,247.00	$ 60.00	$ 1,307.00	$ 2,182.25	1307	1414342.796	108213%	**108113%**
Workaround Developed	**160**	3200	$ 16,421.50	$ 600.00	$ 17,021.50	$ 28,737.63	17021.5	2545817.032	14956%	**14856%**
New Software Version Developed	**800**	16000	$ 230,750.00	$ 6,000.00	$ 236,750.00	$ 403,812.50	236750	12729085.16	5377%	**5277%**
Known Anomalies are Resolved	**625**	6250	$ 97,638.75	$ 3,000.00	$ 100,638.75	$ 170,867.81	100638.75	4972298.891	4941%	**4841%**
New Software Version Fielded to Units (Pushed to Ship via Distance Support)	**1600**	32000	$ 155.88	$ 7.50	$ 13,723.50	$ 272.78	13723.5	25458170.32	185508%	**185408%**
Totals		405228	$ 419,674.60	$ 12,127.50	$ 445,362.23	$ 733,353.43	445362.225	322386180.8	72387%	**72287%**

Personel Costs	Base Pay	Location Adjusted	**Hourly Wage**	**Contractor Wage**
GS 9	$ 45,294.00	$ 56,617.50	$ 28.31	$ 49.54
GS 10	$ 49,880.00	$ 62,350.00	$ 31.18	$ 54.56
GS 11	$ 54,804.00	$ 68,505.00	$ 34.25	$ 59.94
GS 12	$ 65,686.00	$ 82,107.50	$ 41.05	$ 71.84
GS 13	$ 78,111.00	$ 97,638.75	$ 48.82	$ 85.43
GS 14	$ 92,300.00	$ 115,375.00	$ 57.69	$ 100.95

Price Per Common Unit $ 9.47

Table 7. KVA OA-enabled (All Ships)

6. "To Be" Process Data Analysis

The "To Be" OA-enabled model and the implementation of Distance Support and Monitoring for the AEGIS software maintenance and upgrade process has produced appreciable increases in the ROI for each of the sub-processes. Each of the sub-processes that were changed through the OA transformation experienced an increase in the categories "Total Knowledge" and the "Numerator." The "Numerator" category represents revenue for each of the sub-processes.

The increase in "Total Knowledge" was due to several factors. The OA-enabled AEGIS software maintenance and upgrade process allows for easier anomaly identification. Once the anomaly is identified, the OA-enabled process allows for a more complete representation of the circumstances surrounding the anomaly. Both of these factors allow for an increase in the knowledge, in hours, over a given year using the OA-enabled AEGIS software maintenance and upgrade process. The increase in "Total Knowledge" was also affected by a remote diagnostics network that could make the anomaly information available to a subject-matter expert rather than being assessed solely by ship personnel. The remote diagnostics allow for easier collaboration between SMEs and personnel on the ship.

The category "Numerator" was also substantially increased. This increase in revenue was primarily due to more units of knowledge being made available at the same price per unit of knowledge. The increase in revenue was due to a more efficient OA-enabled AEGIS software maintenance and upgrade process. The move to OA facilitated the use of distance support and allowed for improvements to key sub-processes that changed the current "As Is" process to an entirely different procedure. The "To Be" OA-enabled AEGIS software maintenance and upgrade process provides a more efficient, highly automated alternative to the current "As Is" process.

The increases in the category "Numerator" or revenue, "Total Knowledge," "ROK" and "ROI" in the OA-enabled AEGIS software maintenance and upgrade process were estimated using a conservative method. The potential for further increases in each of the categories mentioned above exists, but is very difficult to properly document. For the purposes of this research, the category “Average Time to Complete” was left unchanged, except for the process “New Software Version Fielded to Units.” The “Average Time to Complete” this process was estimated to be 15 minutes in the “To Be” model. This estimation was based on SME inputs, along with precedents that were established in distance-support policy. The other processes in the AEGIS software maintenance and upgrade process have the potential for decreased completion times. Implementing remote station monitoring allows employees to become more efficient and potentially decrease the “Average Time to Complete” in the execution of each of their processes. Since there is no way to accurately project or estimate this increase of efficiency, this was not taken into account in this research.

Due to the fact that the sub-processes changed in the OA-enabled AEGIS software maintenance and upgrade procedure, the process executors would need to relearn how to execute each of the sub-processes. Even though training can facilitate this learning, a substantial amount of learning still takes place on the job, or “learning by doing.” It would be expected that over time, the sub-processes would become even more efficient than represented in this analysis. Further increased efficiency would serve to both decrease cycle-time and also decrease the “Average Time to Complete” for each of the sub-processes. This could not be projected in this work due to data-collection challenges, as the amount of efficiency increase cannot be accurately predicted.

D. Comparative Analysis

Now that both the “As Is” and “To Be” process models and data have been presented, it is valuable to present them in a side-by-side comparison. Each of the sub-processes are presented below with their corresponding ROI’s for both the “As Is” and “To Be” configurations.

"As Is"

Process	Revenue	Total Costs	ROI
Software Anomaly Detected	$ 56,826.27	$ 14,301.00	297%
Cause of Anomaly Determined	$ 845,629.06	$ 251,596.88	236%
Software Bug Report Submitted	$ 31,570.15	$ 1,430.10	2108%
Anomaly Verified	$ 56,826.27	$ 17,021.50	234%
Anomaly Appended to Working List of Known Issues	$ 16,837.41	$ 1,307.00	1188%
Workaround Developed	$ 30,307.35	$ 17,021.50	78%
New Software Version Developed	$ 151,536.73	$ 236,750.00	-36%
Known Anomalies are Resolved	$ 59,194.03	$ 100,638.75	-41%
New Software Version Fielded to Units	$ 101,024.49	$ 156,840.00	-36%
Totals	$1,349,751.77	$ 796,906.73	69%

"To Be"

Process	Revenue	Total Costs	ROI
New Release Fielded (Push to Ship via Distance Support)	$ 284,131.37	$ 7,150.50	3874%
Remote Diagnostics Detect/Fix Anomaly	$2,367,761.38	$ 50,319.38	4605%
Remote Diagnostics Submit Software Bug Report for Anomaly	$ 568,262.73	$ 1,430.10	39636%
Anomaly Verified	$ 56,826.27	$ 17,021.50	234%
Anomaly Appended to Working List of Known Issues	$ 16,837.41	$ 1,307.00	1188%
Workaround Developed	$ 30,307.35	$ 17,021.50	78%
New Software Version Developed	$ 151,536.73	$ 236,750.00	-36%
Known Anomalies are Resolved	$ 59,194.03	$ 100,638.75	-41%
New Software Version Fielded to Units (Pushed to Ship via Distance Support)	$ 303,073.46	$ 163.38	185408%
Totals	$3,837,930.72	$ 431,802.10	789%

Table 8. Side-by-side Comparison (One Ship)

The side-by-side comparison of the “As Is” and “To Be” models shown above demonstrate the dramatic effect that OA and distance support initiatives could have on the AEGIS software maintenance and upgrade process. The increase in revenue by $2,488,178.96, the cost savings of $365,104.63, and the increase in ROI from 69% to 720% after reengineering the AEGIS software maintenance and upgrade process using an OA approach depicts a substantial increase in efficiency of each of the affected sub-processes and also of the process as a whole. This side-by-side comparison represents the efficiency improvements for one ship in the current AEGIS Fleet. It is also of value to present the side-by-side comparisons for the entire AEGIS Fleet.

"As Is"

Process	Revenue	Total Costs	ROI
Software Anomaly Detected	$ 4,773,406.94	$ 14,301.00	33278%
Cause of Anomaly Determined	$ 71,032,841.31	$ 251,596.88	28133%
Software Bug Report Submitted	$ 2,651,892.74	$ 1,430.10	185334%
Anomaly Verified	$ 4,773,406.94	$ 17,021.50	27943%
Anomaly Appended to Working List of Known Issues	$ 1,414,342.80	$ 1,307.00	108113%
Workaround Developed	$ 2,545,817.03	$ 17,021.50	14856%
New Software Version Developed	$ 12,729,085.16	$ 236,750.00	5277%
Known Anomalies are Resolved	$ 4,972,298.89	$ 100,638.75	4841%
New Software Version Fielded to Units	$ 8,486,056.77	$ 26,349,120.00	-68%
Totals	$ 113,379,148.58	$ 26,989,186.73	320%

"To Be"

Process	Revenue	Total Costs	ROI
New Release Fielded (Push to Ship via Distance Support)	$ 23,867,034.68	$ 7,150.50	333681%
Remote Diagnostics Detect/Fix Anomaly	$ 198,891,955.66	$ 50,319.38	395159%
Remote Diagnostics Submit Software Bug Report for Anomaly	$ 47,734,069.36	$ 1,430.10	3337713%
Anomaly Verified	$ 4,773,406.94	$ 17,021.50	27943%
Anomaly Appended to Working List of Known Issues	$ 1,414,342.80	$ 1,307.00	108113%
Workaround Developed	$ 2,545,817.03	$ 17,021.50	14856%
New Software Version Developed	$ 12,729,085.16	$ 236,750.00	5277%
Known Anomalies are Resolved	$ 4,972,298.89	$ 100,638.75	4841%
New Software Version Fielded to Units (Pushed to Ship via Distance Support)	$ 25,458,170.32	$ 13,723.50	185408%
Totals	$ 322,386,180.83	$ 445,362.23	72287%

Table 9. Side-by-side Comparison (All Ships)

The side-by-side comparison of the “As Is” and “To Be” models shown above for all AEGIS ships in the US Fleet demonstrate the dramatic effect that OA and distance support initiatives could have on the AEGIS software maintenance and upgrade process. The increase in revenue by $209,007,032.26, the cost savings of $26,543,824.50, and the increase in ROI by 71987% represent the substantial benefits that can be achieved when the OA-enabled AEGIS software maintenance and upgrade process is applied to all AEGIS ships.

V. Conclusions and Recommendations

A. Conclusions and Recommendations

Proprietary closed architecture systems, such as the AEGIS system, have been effective systems that have provided the Navy with important operational capabilities in the past. As these systems age, there becomes a need for increased Sustaining Engineering (SE) support for these systems. The recent Component Business Model (CBM) conducted by IBM and the large investment in SE reaffirmed the importance and need for efficiency. This is especially evident in the AEGIS software maintenance and upgrade process. The current proprietary, closed architecture design of the AEGIS system makes the AEGIS software maintenance and upgrade process a very costly and time-intensive process requiring a great deal of personnel. Incompatibility and missed opportunities for new technologies are a considerable problem for the AEGIS software maintenance and upgrade process and for acquirers and developers.

The incorporation of Open Architecture (OA) would allow current proprietary systems to leverage new technologies in an effort to increase efficiency and realize the full potential of the Navy's systems and processes. Current programs and policies, such as distance support and Maintenance Free Operating Period (MFOP), could be easily integrated into the AEGIS software maintenance and upgrade process through a move to OA. This thesis provided insight into the operational value that can be achieved by using an OA framework in the AEGIS software maintenance and upgrade process through an increase in Return on Investment (ROI).

The current AEGIS software maintenance and upgrade process is a fairly efficient process. The total ROI associated with the "As Is" AEGIS software maintenance and upgrade process is 69%. This indicates that the process returns more revenue than the cost of the aggregate process; however, even though the "As

Is" process produces a positive ROI, the potential for increased ROI exists through the transition to an OA-enabled AEGIS software maintenance and upgrade process.

The "To Be" OA-enabled AEGIS software maintenance and upgrade process incorporates the OA tenets of scalability and portability. The OA framework allows for several of the sub-processes to be changed to allow for the use of distance support and MFOP concepts. These concepts enabled decreased personnel and increased automation in the processes "New Software Fielded to Units," "Remote Diagnostics Detect/Fix Anomaly," and "Remote Diagnostics Submit Software Bug Report for Anomaly." The improvements made in the "To Be" OA-enabled AEGIS software maintenance and upgrade process greatly increased the ROI for each of the improved sub-processes and also for the entire process as a whole. The total ROI and costs saving associated with the "To Be" OA-enabled AEGIS software maintenance and upgrade process is 720% and $365,104.10. This represents a sizable improvement in efficiency with the incorporation of OA in the AEGIS software maintenance and upgrade process. The improvement in efficiency is even more apparent when it is applied to all AEGIS ships in the current US Navy Fleet. This can be seen below in Table 10.

"As Is"

Process	Revenue	Total Costs	ROI
Software Anomaly Detected	$ 56,826.27	$ 14,301.00	297%
Cause of Anomaly Determined	$ 845,629.06	$ 251,596.88	236%
Software Bug Report Submitted	$ 31,570.15	$ 1,430.10	2108%
Anomaly Verified	$ 56,826.27	$ 17,021.50	234%
Anomaly Appended to Working List of Known Issues	$ 16,837.41	$ 1,307.00	1188%
Workaround Developed	$ 30,307.35	$ 17,021.50	78%
New Software Version Developed	$ 151,536.73	$ 236,750.00	-36%
Known Anomalies are Resolved	$ 59,194.03	$ 100,638.75	-41%
New Software Version Fielded to Units	$ 101,024.49	$ 156,840.00	-36%
Totals	$1,349,751.77	$ 796,906.73	69%

"To Be"

Process	Revenue	Total Costs	ROI
New Release Fielded (Push to Ship via Distance Support)	$ 284,131.37	$ 7,150.50	3874%
Remote Diagnostics Detect/Fix Anomaly	$2,367,761.38	$ 50,319.38	4605%
Remote Diagnostics Submit Software Bug Report for Anomaly	$ 568,262.73	$ 1,430.10	39636%
Anomaly Verified	$ 56,826.27	$ 17,021.50	234%
Anomaly Appended to Working List of Known Issues	$ 16,837.41	$ 1,307.00	1188%
Workaround Developed	$ 30,307.35	$ 17,021.50	78%
New Software Version Developed	$ 151,536.73	$ 236,750.00	-36%
Known Anomalies are Resolved	$ 59,194.03	$ 100,638.75	-41%
New Software Version Fielded to Units (Pushed to Ship via Distance Support)	$ 303,073.46	$ 163.38	185408%
Totals	$3,837,930.72	$ 431,802.10	789%

Table 10. Side-by-side Comparison (One Ship)

There are several potential benefits that are not captured in the KVA analysis performed in this research. The increase in capabilities that occurs from the remote maintenance and upgrade ability of the OA-enabled AEGIS software maintenance and upgrade process was not captured in the KVA analysis. This would allow for increased capabilities to be delivered to the warfighter. These increased capabilities could allow the warfighter to maintain a more complete operational picture and lead to increased operational effectiveness.

The OA-enabled AEGIS software maintenance and upgrade process allows for the software to be delivered to operational units much sooner than the "As Is" configuration allows. Due to this, it is important to consider the concept of the time value of money. The time value of money concept suggests that a dollar delivered today would be worth more than a dollar delivered in a month from today. This same principle can be applied to software updates and operational capabilities. The sooner that a software update can be fielded, the greater benefit it can provide to the operational unit and to the warfighter.

B. Research Limitations

The data collected and analyzed in this research was provided by subject-matter experts (SMEs). These SMEs each have a different background and current level of experience. These factors have an effect on the data and suggestions provided by each SME. This gives some level of subjectivity to the data that was collected. Until data-capturing methods are in place to collect historical data associated with the AEGIS software maintenance and upgrade process, SME inputs will continue to be the appropriate method for data collection in this type of analysis.

The "To Be" analysis was based on inputs from SMEs as to the best and most feasible areas in which to implement OA and distance-support initiatives. Inputs from programs implemented in other communities, such as the Maintenance Free Operating Period utilized by the Undersea Warfare Community, were carefully considered; but, ultimately, SME inputs determined what was feasible for the AEGIS software maintenance and upgrade process. The “To Be” process provides a conceptual framework for the process reengineering of the AEGIS software maintenance and upgrade procedure without taking into account technical and legal aspects associated with the reengineering.

The KVA analysis performed in this research estimated the reduction in costs that would occur through transformation to an OA-enabled AEGIS software maintenance and upgrade process. Due to data-collection challenges, costs were estimated based on the number of employees and the cost of information technology associated with each sub-process in the AEGIS software maintenance and upgrade process. The reduction in cost that occurred in the transformation to the OA-enabled AEGIS software maintenance and upgrade process was most likely underestimated. Remote monitoring enables reductions in personnel and work time that were accounted for in the KVA analysis. The KVA analysis did not account for the cost savings to Sustaining Engineering (SE) activities not specifically associated with the AEGIS software maintenance and upgrade process. Processes outside the AEGIS software maintenance and upgrade process, but still included in SE (such as configuration management, verification and validation), could represent additional cost savings through the transformation to OA.

VI. Future Research

A. Future Research

This thesis explored the possibility of implementing an Open Architecture (OA) approach to the AEGIS software maintenance and upgrade process. The reengineering of the AEGIS software maintenance and upgrade process using OA illustrated that there are definite benefits to implementing the principles of OA into system and process design. The move to an Open Architecture framework is a large undertaking that will require a considerable change in thinking—especially when designing new system interfaces and ways to bridge existing proprietary closed architecture systems to open systems. The Navy has recognized the benefits of Open Architecture, and through the work of the Project Executive Office Integrated Warfare Systems (PEO IWS) will continue to improve existing legacy systems.

A baseline has been set in this research for the value of integrating OA into the AEGIS software maintenance and upgrade process. There is still much research that can be conducted to evaluate the benefit of OA in the AEGIS software maintenance and upgrade process. For instance, there is the great possibility of exploring the impact of OA on ship logistics. This research reinforced the fact that OA enables personnel reductions and decreased cycle-time. These benefits free-up time for operators on the ship to focus on more mission-critical areas rather than on Sustaining Engineering (SE) processes. The benefits of this increase on operational mission areas could positively affect operational efficiency and could lead to additional research areas. There is also the potential of including increased capabilities into each software upgrade that is fielded. The decreased cycle-time that is produced by the OA-enabled AEGIS software maintenance and upgrade process would allow developers to incorporate increased capabilities into software updates sooner, possibly leading to improved mission effectiveness. This topic could also present an area for future research.

The potential for OA reengineering exists in the decomposed off-ship AEGIS software maintenance and upgrade process model. Many of the processes involved in the off-ship portion could be refined, but due to data-collection challenges, this research was forced to focus on the aggregate on-ship and off-ship process model. The implementation of OA, along with distance-support policies and the Maintenance Free Operating Period (MFOP) concept, could drastically alter the decomposed off-ship AEGIS software maintenance and upgrade process model. Data-capturing methods could be put into place to provide historical data inputs along with subject-matter expert (SME) inputs. The potential for increased efficiency in this process should not be overlooked and would provide an area for further research. Another area for future research could employ business process reengineering (BPR) to the decomposed off-ship AEGIS software maintenance and upgrade process in order to reduce redundancies that occur between the software contractor and government agencies.

In addition, the implementation of an OA-enabled AEGIS software maintenance and upgrade process presents an interesting challenge in the area of training shipboard operators. When a new software version is fielded via distance support in real-time, no time period currently exists for familiarization and training with the new software version. This could potentially have adverse affects on mission effectiveness. An area for future research exists in the potential for distance-support training or other training methods with OA-enabled systems.

The potential for a move to Service-oriented Architecture (SOA) also exists for the AEGIS software maintenance and upgrade process. This move would require a radical reengineering of the current processes and sub-processes and could have the potential to greatly improve efficiency by loosely coupling AEGIS modules. A Knowledge Value Added (KVA) analysis could be conducted in a future study to examine the potential benefits of incorporating SOA into the AEGIS software maintenance and upgrade process.

Lastly, the researchers plan to conduct a Real Options (RO) analysis in future research. This analysis will serve to project potential benefits and risks to different options that could be implemented when reengineering the AEGIS software maintenance and upgrade process. Real Options analysis will provide investment options through careful analysis of the KVA and will take into account risk identification, quantification, valuation, mitigation and diversification.

THIS PAGE INTENTIONALLY LEFT BLANK

List of References

Azani, C. (2001, September-October). The test and evaluation challenges of following an open system strategy. *ITEA Journal,* 22(3), 1. Retrieved March 15, 2007.

Chief of Naval Operations. (2006a). *Memorandum, distance support policy.* Washington, DC: author. Retrieved November 10, 2006.

Chief of Naval Operations. (2006b). *Navy distance support policy*. (Vol. Navy distance support policy clarification). Washington, DC: Office of the Chief of Naval Operations. Retrieved January 18, 2007.

Chief of Naval Operations. (2005). *Reference 9010 Ser N6N7/5u916276*. Washington, DC: Office of the Chief of Naval Operations. Retrieved March 15, 2007.

Clark, V. (2002, October). Sea Power 21: Projecting decisive joint capabilities. *Proceedings*, 1. Retrieved November 10, 2006.

Commander, Naval Sea Systems Command. (1999). *NAVSEA INSTRUCTION 5400.57B.* (Vol. Engineering agent assignment and technical authority.) Arlington, VA: NAVSEA. Retrieved January 18, 2007.

Coronado Mondragon, C. E., Coronado Mondragon, A. E., & Miller, R. (2006). Modularity, open architecture and innovation: An automotive perspective. *International Journal of Automotive Technology and Management*, 6(3), 346-363. Retrieved November 10, 2006.

Department of Defense. (2003). *DoD Directive 5000.1*. Washington, DC: author.

Department of Defense. (1985). *Military standard technical reviews and audits for systems, equipments, and computer software.* Washington, DC: author. Retrieved November 10, 2006, from http://sparc.airtime.co.uk/users/wysywig/1521b.htm.

Department of Defense, NAVSEA, PEO IWS. (2004). *Open architecture computing environment technologies and standards* (Vers. 1.0). Washington, DC: authors. Retrieved March 15, 2007.

Department of the Navy, NAVSEA, PEO IWS. (2004). Open architecture. *Open architecture computing environment technologies and standards* (Vers. 1.0). Washington, DC: NAVSEA. Retrieved March 15, 2007.

FAS Military Network Analysis. (n.d.). *AN/SQQ-89 ASW Combat System [ASWCS].* Retrieved March 15, 2007, from www.fas.org, http://www.fas.org/man/dod-101/sys/ship/weaps/an-sqq-89.htm.

Government Accountability Office. (2006). *Defense acquisitions: DoD management approach and processes not well-suited to support development of information grid*. Washington, DC: author. Retrieved March 15, 2007.

Housel, T. & Bell, A. (2001). Measuring and managing knowledge (1st ed.). New York City: McGraw Hill.

Komoroski, C.L., Housel, T., Mun, J., & Hom, S. (n.d.). *Improving the shipyard planning process: The KVA + real options approach*. Executive Summary. Monterey, CA: Naval Postgraduate School. Retrieved October 10, 2006.

Kratz, L.A., Fowler, R.T., & Cothran, J.D. (2002). Achieving defense transformation: Through total life cycle systems management—Sustainment capabilities. *Program Manager Magazine*, Sept-Oct. Retrieved March 15, 2007, from http://findarticles.com/p/articles/mi_m0KAA/is_5_31/ai_94771281.

Lapham, M.A. (2005). *Sustaining software-intensive systems—A conundrum*. Pittsburgh, PA: Carnegie Mellon Software Engineering Institute. Retrieved January 17, 2007.

Madu, C. N. (2005, January). Strategic value of reliability and maintainability management. *International Journal of Quality & Reliability Maintenance*, 22(3), 317-328. Retrieved January 18, 2007, from www.emeraldinsight.com/0265-671X.htm.

Mathaisel, D. F. X., Cathcart, T. P., & Comm, C.L. (2004). A framework for benchmarking, classifying, and implementing best sustainment practices. *Benchmarking: An International Journal*, 11(4:1463-5771), 403-417. Retrieved from www.emeraldinsight.com/1463-5771.htm.

Nathan, J. (2003). *Concerning Navy transformation and future naval capabilities.* Statement to Congress before the Subcommittee on Projection Forces, House

Armed Service Committee, United States House of Representatives. Washington, DC: US Congress. Retrieved January 10, 2007.

NAVSEA Surface Warfare Logistics and Maintenance. (n.d.). *Distance support remote monitoring & remote diagnostics concept of operations*. Washington, DC: CONOPS, NAVSEA. Retrieved January 10, 2007.

NAVSEA Surface Warfare Logistics and Maintenance and PMS401. (2005). *ARCI maintenance free operating period (MFOP) pilot program final report.* Washington, DC: NAVSEA. Retrieved November 10, 2006.

NAVSURFWARCENDIV Port Hueneme, CA. *Distance support—Navy information application product COMNAVSURFOR San Diego, CA, UNCLAS.* Memorandum, Port Hueneme, CA. Retrieved November 10, 2006.

Navy Marine Corps Intranet. Retrieved March 15, 2006, from, http://www.msc.navy.mil/n6/nmci/.

Pavlick,T. (2005). *Component business modeling. Assessing the Navy's business of warfighting and business of acquisition.* IBM: IBM Software Group, Strategy. Retrieved January 18, 2007.

Shannon, J. (2006). *Naval open architecture: Component business modeling for PEO-IWS.* IBM: PEO-IWS 7.0. Retrieved January 18, 2007.

US Army PEO IES. (2006). *Services oriented architecture*. Fort Belvoir, VA: author. Retrieved March 15, 2007, from http://www.army.mil/escc/erp/soa.htm#6.

Uchytil, J. (2006). *Assessing the operational value of situational awareness for AEGIS and ship self-defense system (SSDS) platforms through the application of the knowledge value-added (KVA) methodology*. (Master of Science in Information Technology Management). Monterey, CA: Naval Postgraduate School. Retrieved October 10, 2006.

Uchytil, J., Housel, T. Hom, S., Mun, J., & Tarantino, E. *AEGIS and ship self-defense system (SSDS) platforms: Using KVA analysis, risk simulation and strategic real options to assess operational effectiveness.* Monterey, CA: Naval Postgraduate School. Retrieved September 10, 2006.

THIS PAGE INTENTIONALLY LEFT BLANK

Appendix 1: (Komoroski, Housel, Mun & Hom, 2006, p. 4-6)

KVA+RO Framework

KVA+RO measures operating performance, cost-effectiveness, return on investments (ROI), risk, real options (capturing strategic flexibility), and portfolio optimization. KVA+RO results empower decision-makers and support IT acquisition business cases by providing performance-based data and scenario analysis. Analyses like the ROI on individual projects, programs, processes and sub-processes within a portfolio of IT acquisitions can be derived through the KVA methodology.

Figure A-1. Valuation Framework

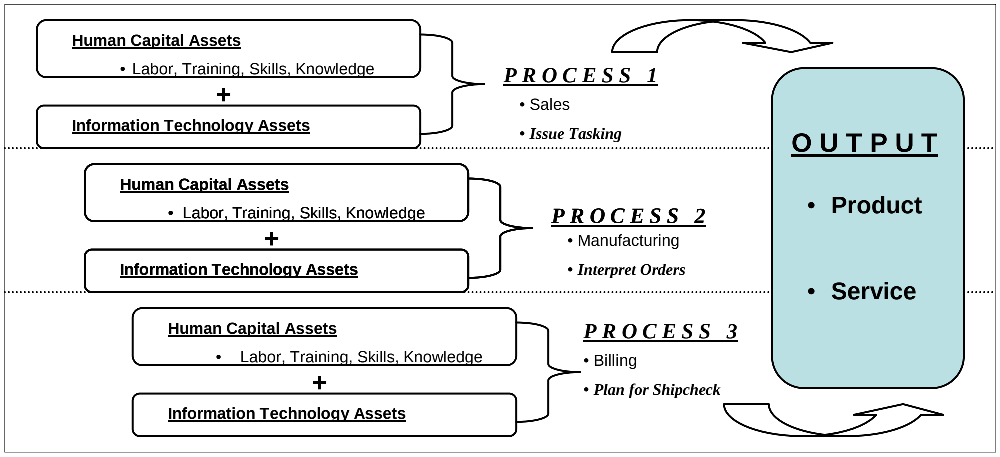
Data Collection + KVA Methodology + Real Options Analysis = Historical, Performance-based Data and Analyses

Providing ROI and total strategic options values along with the risk measurements for each option.

Knowledge Value Added (KVA)

KVA measures the value provided by human capital assets and IT assets for an organization, process or function at any level of analysis. It monetizes the outputs of all assets, including intangible knowledge assets, using a market-comparables-valuation technique. Capturing the value embedded in an organization's core processes, employees and IT enables the actual cost and revenue of a product or service to be calculated.

Figure A-2. Measuring Output



Total value is captured in the key metric measurement of ROI. This ratio has comparable revenue – investment cost in the numerator and investment cost in the denominator.

Table A-1. KVA Metrics

Metric	Description	Type	Calculation
Return on Investment (ROI)	Same as ROI at the sub-corporate, process level	Traditional investment finance ratio	(Revenue-Investment Cost) / Investment cost

Real Options (RO)

Potential strategic investment options can then be evaluated with real options analysis using historical data provided by KVA. The analysis applied is a robust and analytical process incorporating risk identification (applying various sensitivity techniques), risk quantification (applying Monte Carlo simulation), risk valuation (real options analysis), risk mitigation (real options framing), and risk diversification (analytical portfolio optimization).

Initial Distribution List

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California

3. Navy Representative
   Naval Postgraduate School
   Monterey, California

4. NAVSEA
   PEO IWS (IA)
   Open Architecture
   Washington Navy Yard, Washington D.C.

5. Dr. Dan Boger
   Naval Postgraduate School
   Monterey, California

6. Dr. Thomas Housel
   Naval Postgraduate School
   Monterey, California

7. Mr. Glenn Cook
   Naval Postgraduate School
   Monterey, California

8. Karey L. Shaffer
   Program Manager, Acquisition Research Program, GB
   555 Dyer Road, Naval Postgraduate School, Monterey, CA 93943-5000

Copies of the Acquisition Sponsored Research Reports may be printed from our website www.acquisitionresearch.org

THIS PAGE INTENTIONALLY LEFT BLANK

2003 - 2006 Sponsored Acquisition Research Topics

Acquisition Management

- Software Requirements for OA
- Managing Services Supply Chain
- Acquiring Combat Capability via Public-Private Partnerships (PPPs)
- Knowledge Value Added (KVA) + Real Options (RO) Applied to Shipyard Planning Processes
- Portfolio Optimization via KVA + RO
- MOSA Contracting Implications
- Strategy for Defense Acquisition Research
- Spiral Development
- BCA: Contractor vs. Organic Growth

Contract Management

- USAF IT Commodity Council
- Contractors in 21st Century Combat Zone
- Joint Contingency Contracting
- Navy Contract Writing Guide
- Commodity Sourcing Strategies
- Past Performance in Source Selection
- USMC Contingency Contracting
- Transforming DoD Contract Closeout
- Model for Optimizing Contingency Contracting Planning and Execution

Financial Management

- PPPs and Government Financing
- Energy Saving Contracts/DoD Mobile Assets
- Capital Budgeting for DoD
- Financing DoD Budget via PPPs
- ROI of Information Warfare Systems

- Acquisitions via leasing: MPS case
- Special Termination Liability in MDAPs

Logistics Management

- R-TOC Aegis Microwave Power Tubes
- Privatization-NOSL/NAWCI
- Army LOG MOD
- PBL (4)
- Contractors Supporting Military Operations
- RFID (4)
- Strategic Sourcing
- ASDS Product Support Analysis
- Analysis of LAV Depot Maintenance
- Diffusion/Variability on Vendor Performance Evaluation
- Optimizing CIWS Life Cycle Support (LCS)

Program Management

- Building Collaborative Capacity
- Knowledge, Responsibilities and Decision Rights in MDAPs
- KVA Applied to Aegis and SSDS
- Business Process Reengineering (BPR) for LCS Mission Module Acquisition
- Terminating Your Own Program
- Collaborative IT Tools Leveraging Competence

A complete listing and electronic copies of published research within the Acquisition Research Program are available on our website: www.acquisitionresearch.org

ACQUISITION RESEARCH PROGRAM
GRADUATE SCHOOL OF BUSINESS & PUBLIC POLICY
NAVAL POSTGRADUATE SCHOOL
555 DYER ROAD, INGERSOLL HALL
MONTEREY, CALIFORNIA 93943

www.acquisitionresearch.org