NPS-AM-10-022

PROCEEDINGS

OF THE

SEVENTH ANNUAL ACQUISITION RESEARCH SYMPOSIUM WEDNESDAY SESSIONS VOLUME I

Acquisition Research

Creating Synergy for Informed Change

May 12 - 13, 2010

Published: 30 April 2010

Prepared for: Naval Postgraduate School, Monterey, California 93943

ACQUISITION RESEARCH PROGRAM
GRADUATE SCHOOL OF BUSINESS & PUBLIC POLICY
NAVAL POSTGRADUATE SCHOOL

The research presented at the symposium 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.net

Preface and Acknowledgements

One needs no powers of prophecy to predict that perilous economic times lie ahead for our nation and the rest of the world. While uncertainty surrounds the global economic crisis—how long it will last, for example, and whether conditions will worsen further before they improve—there can be little doubt that national budgets for defense acquisition will experience considerable pressure and quite possibly decline in the coming years.

Under such conditions, we may be tempted either to accept with fatalism some reduced level of expectations for acquisition outcomes or, alternatively, to embrace trite exhortations to do more with less. Neither of these is, of course, a tenable position for serious scholars of acquisition. Rather, we ought to continue to seek the best possible understanding of acquisition that will lead to the best possible outcomes given available resources. This entails continued work on what Don Kettl has termed the "smart buyer" problem: knowing what to buy, who to buy it from, and how to assess its quality.

Nor can the "how to buy" problem be neglected, as new laws, regulations, and other policies continue to subject acquisition processes to what many see as excessively "bureaucratic" requirements. While these new structures often have worthy goals to increase accountability, transparency, and social equity, their economic costs can't be ignored. Acquisition scholars should find ways to, paraphrasing Aaron Wildavsky, "speak scientific truth to power" so that policy-makers' decisions are informed by the best possible research on their costs and benefits.

Currently, DoD investments in acquisition research represent about one one-thousandth of one percent of the total defense budget, yet we believe they have the potential to lead to considerable savings, both in the near and long term. We see strong evidence that this potential is being realized through the products of the Naval Postgraduate School's Acquisition Research Program.

Our goals remain the same, with recent highlights noted below:

1. Position the ARP in a leadership role to continue to develop the body of knowledge in defense acquisition research
    - Over 450 published works since inception,
    - All completed research is published in full text on the ARP website, www.acquisitionresearch.net, allowing ready access by any and all parties interested in the DoD acquisition process,
    - Sponsoring our 7th Annual Acquisition Research Symposium, the first of which was held in May 2004, draws thought leaders of the DoD acquisition community, academia and industry.
2. Establish acquisition research as an integral part of policy-making for DoD officials. Some processes informed by this research include:
    - Open Architecture implementation practices and policies to include software/hardware reuse repository characteristics such as ontology, search engines, licensing issues and testing requirements; Creation of the concept of Integration Readiness Levels paired with technology readiness levels to create a System Readiness Level scale;

- Development of logistics resource strategies for selecting among Contracted Logistics Support, Organic Logistics Support or Blended Logistics Support;
- Identification of the scope and causes of bid protests and recommendations for reducing same; and
- Creation of a database of key contracting workforce demographics for the Army Contracting Command.

3. Create a stream of relevant information concerning the performance of DoD acquisition policies with viable recommendations for continuous process improvement.
    - The body of knowledge on the DoD acquisition process continues to increase by over 140 research products a year.
    - Faculty researchers routinely give multiple presentations, in both national and international fora, featuring their research work—thereby increasing exposure to a broader audience. Typical audiences include the London School of Economics, the Federal Reserve, the Center for Strategic and International Studies, the Western Economics Association International Conference, the International Procurement Conference and the National Contracts Management Association. At a minimum, over 90 presentations of sponsored research results were made in 2009.
    - The *International Journal of Defense Acquisition Management* is the ARP's initiative to promote defense acquisition research in the peer-reviewed literature. All published articles can be freely downloaded from the journal's website. During 2009, three more articles were published in the journal from authors in the US, Australia, and the UK, bringing the total to six since the journal was founded in May 2008. There are also seven articles currently under review. The journal substantially increases the "reach" of our research products.
4. Prepare the DoD workforce to participate in the continued evolution of the defense acquisition process.
    - The ARP plays a major role in providing a DoD-relevant graduate education program to future DoD officials. Synergy between research conducted and course content delivered enhances both the teaching and learning processes.
    - The number of students engaged in focused acquisition research for their MBA projects continues to grow. These students have the benefit of being able to immediately apply their newly acquired acquisition skills to real-world issues.
    - Student projects on the economics of the shipbuilding industrial base and services contracting are expected to contribute significantly to the body of knowledge and decision-making process for these two very important and timely subjects.
5. Collaboration among universities, think tanks, industry and government in acquisition research.

- Over 60 universities/think tanks have participated in the annual Acquisition Research Symposium or the Acquisition Research Program as a result of a focused effort to create a Virtual University Consortium.
- Emerging collaborative research efforts continue to bring new scholar and practitioner thought to the business issues facing the DoD, as was demonstrated by the large response to our fourth Broad Area Announcement (BAA) in support of the OSD-sponsored acquisition research program. As we write this, our fifth BAA is being prepared for release.
- The International Journal of Defense Acquisition is attracting scholars from the United Kingdom, Canada, Nigeria, Singapore, The Netherlands, the United States and Australia.

We gratefully acknowledge the ongoing support and leadership of our sponsors, whose foresight and vision have assured the continuing success of the Acquisition Research Program:

- Office of the Under Secretary of Defense (Acquisition, Technology & Logistics)
- Program Executive Officer SHIPS
- Commander, Naval Sea Systems Command
- Army Contracting Command, US Army Materiel Command
- Program Manager, Airborne, Maritime and Fixed Station Joint Tactical Radio System
- Program Executive Officer Integrated Warfare Systems
- Office of the Assistant Secretary of the Air Force (Acquisition)
- Office of the Assistant Secretary of the Army (Acquisition, Logistics, & Technology)
- Office of Naval Air Systems Command PMA-290
- Deputy Assistant Secretary of the Navy (Acquisition & Logistics Management)
- Director, Strategic Systems Programs Office
- Director, Defense Business Systems Acquisition Executive, Business Transformation Agency
- Deputy Director, Acquisition Career Management, US Army

We also thank the Naval Postgraduate School Foundation and acknowledge its generous contributions in support of this Symposium.

James B. Greene, Jr.
Rear Admiral, US Navy (Ret.)

Keith F. Snider, PhD
Associate Professor

THIS PAGE INTENTIONALLY LEFT BLANK

The Acquisition Research Program Team

Rear Admiral James B. Greene, Jr. USN (Ret.)—Acquisition Chair, Naval Postgraduate School. RADM Greene develops, implements and oversees the Acquisition Research Program in the Graduate School of Business and Public Policy. He interfaces with DoD, industry and government leaders in acquisition, facilitates graduate student research and conducts guest lectures and seminars. Before serving at NPS, RADM Greene was an independent consultant focusing on Defense Industry business development strategy and execution (for both the public and private sectors), minimizing lifecycle costs through technology applications, alternative financing arrangements for capital-asset procurement, and "red-teaming" corporate proposals for major government procurements.

RADM Greene served as the Assistant Deputy Chief of Naval Operations (Logistics) in the Pentagon from 1991-1995. As Assistant Deputy, he provided oversight, direction and budget development for worldwide US Navy logistics operations. He facilitated depot maintenance, supply chain management, base/station management, environmental programs and logistic advice, and support to the Chief of Naval Operations. Some of his focuses during this time were leading Navy-wide efforts to digitize all technical data (and, therefore, reduce cycle-time) and to develop and implement strategy for procurement of eleven Sealift ships for the rapid deployment forces. He also served as the Senior Military Assistant to the Under Secretary of Defense (Acquisition) from 1987-1990; as such, he advised and counseled the Under Secretary in directing the DoD procurement process.

From 1984-1987, RADM Greene was the Project Manager for the AEGIS project. This was the DoD's largest acquisition project, with an annual budget in excess of $5 billion/year. The project provided oversight and management of research, development, design, production, fleet introduction and full lifecycle support of the entire fleet of AEGIS cruisers, destroyers, and weapons systems through more than 2500 industry contracts. From 1980-1984, RADM Greene served as Director, Committee Liaison, Office of Legislative Affairs followed by a tour as the Executive Assistant, to the Assistant Secretary of the Navy (Shipbuilding and Logistics). From 1964-1980, RADM Greene served as a Surface Warfare Officer in various duties, culminating in Command-at-Sea. His assignments included numerous wartime deployments to Vietnam, as well as the Indian Ocean and the Persian Gulf.

RADM Greene received a BS in Electrical Engineering from Brown University in 1964; he earned an MS in Electrical Engineering and an MS in Business Administration from the Naval Postgraduate School in 1973.

RADM Greene received the **2009 Richard W. Hamming Annual Faculty Award for Achievement in Interdisciplinary Activities.** The selection is based on his work in leading and administering the Naval Postgraduate School's Acquisition Research Program.

Dr. Keith F. Snider—Associate Professor of Public Administration and Management in the Graduate School of Business & Public Policy at the Naval Postgraduate School in Monterey, California, where he teaches courses related to defense acquisition management. He also serves as Principal Investigator for the NPS Acquisition Research Program and as Chair of the Acquisition Academic Area.

Snider has a PhD in Public Administration and Public Affairs from Virginia Polytechnic Institute and State University, a Master of Science degree in Operations Research from the Naval Postgraduate School, and a Bachelor of Science degree from the United States Military Academy at West Point. He served as a field artillery officer in the US

Army for twenty years, retiring at the rank of Lieutenant Colonel. He is a former member of the Army Acquisition Corps and a graduate of the Program Manager's Course at the Defense Systems Management College.

Professor Snider's recent publications appear in *American Review of Public Administration*, *Administration and Society*, *Administrative Theory & Praxis*, *Journal of Public Procurement*, *Acquisition Review Quarterly*, and *Project Management Journal.*

Dr. Snider received the **2009 Richard W. Hamming Annual Faculty Award for Achievement in Interdisciplinary Activities.** The selection is based on his work in leading and administering the Naval Postgraduate School's Acquisition Research Program.

Karey L. Shaffer—Program Manager, General Dynamics Information Technology, supporting the Acquisition Research Program at the Graduate School of Business & Public Policy, Naval Postgraduate School. As PM, Shaffer is responsible for operations and publications in conjunction with the Acquisition Chair and the Principal Investigator. She has also catalyzed, organized and managed the Acquisition Research Symposiums hosted by NPS.

Shaffer served as an independent Project Manager and Marketing Consultant on various projects. Her experiences as such were focused on creating marketing materials, initiating web development, assembling technical teams, managing project lifecycles, processes and cost-savings strategies. As a Resource Specialist at Watson Wyatt Worldwide in Minneapolis, Shaffer developed and implemented template plans to address continuity and functionality in corporate documents; in this same position, she introduced process improvements to increase efficiency in presentation and proposal production in order to reduce the instances of corruption and loss of vital technical information.

Shaffer has also served as the Project Manager for Imagicast, Inc., and as the Operations Manager for the Montana World Trade Center. At Imagicast, she was asked to take over the project management of four failing pilots for Levi Strauss in the San Francisco office. Within four months, the pilots were released; the project lifecycle was shortened; and the production process was refined. In this latter capacity at the MWTC, Shaffer developed operating procedures, policies and processes in compliance with state and federal grant law. Concurrently, she managed $1.25 million in federal appropriations, developed budgeting systems and helped secure a $400,000 federal technology grant. As the Operations Manager, she also launched the MWTC's Conference site, managed various marketing conferences, and taught student practicum programs and seminars.

Shaffer holds an MBA from San Francisco State University and earned her BA in Business Administration (focus on International Business, Marketing and Management) from the University of Montana.

A special thanks to our editors, Adrianne Malan, Jessica Moon, Steve Williams, and Lyndsee Cordes, for all that they have done to make this publication a success, to Shellee Dooley and Tera Yoder for production support, and to the staff at the Graduate School of Business & Public Policy for their administrative support. Our program success is directly related to the combined efforts of many.

Announcement and Call for Symposium Proposals

The Graduate School of Business & Public Policy at the Naval Postgraduate School announces the **8th Annual Acquisition Research Symposium** to be held **May 11-12, 2011 in Monterey, California.**

This symposium serves as a forum for the presentation of acquisition research and the exchange of ideas among scholars and practitioners of public-sector acquisition. We seek a diverse audience of influential attendees from academe, government, and industry who are well placed to shape and promote future research in acquisition.

The Symposium Program Committee solicits proposals for panels and/or papers from academicians, practitioners, students and others with interests in the study of acquisition. The following list of topics is provided to indicate the range of potential research areas of interest for this symposium: **acquisition and procurement policy, supply chain management, public budgeting and finance, cost management, project management, logistics management, engineering management, outsourcing, performance measurement, and organization studies.**

Proposals must be submitted by **November 5, 2010**. The Program Committee will make notifications of accepted proposals by **December 10, 2010**. Final papers must be submitted by **April 1, 2011**.

Proposals for papers should include an abstract along with identification, affiliation, contact information and short bio for the author(s). Proposals for papers plan for a 20 minute presentation. Proposals for panels (plan for 90 minute duration) should include the same information as above as well as a description of the panel subject and format, along with participants' names, qualifications and the specific contributions each participant will make to the panel.

Submit paper and panel proposals to www.researchsymposium.org

THIS PAGE INTENTIONALLY LEFT BLANK

GRANTS.GOV -- NPSBAA10-002

The Acquisition Research Program
Open until 5:00 p.m. PDST 18 June 2010

Primary objective is to attract outstanding researchers and scholars to investigate topics of interest to the defense acquisition community. The program solicits innovative proposals for defense acquisition management and policy research to be conducted during fiscal year (FY) 2011 (1 Oct 2010 - 30 Sep 2011).

Defense acquisition management and policy research refers to investigations in all disciplines, fields, and domains that (1) are involved in the acquisition of products and/or services for national defense, or (2) could potentially be brought to bear to improve defense acquisition. It includes but is not limited to economics, finance, financial management, information systems, organization theory, operations management, human resources management, and marketing, as well as the "traditional" acquisition areas such as contracting, program/project management, logistics, and systems engineering management.

This program is targeted in particular to U.S. universities (including U.S. government schools of higher education) or other research institutions outside the Department of Defense.

The Government anticipates making multiple awards up to $120,000 each for a basic research period of twelve months. NPS plans to complete proposal evaluations and notify awardees in mid-August 2010. The actual date of grant award will depend on availability of funds and the capabilities of the grants office. Prior year awards occurred in the August – January timeframe. Awardees may request approval of pre-award costs (up to three months), or they may request adjustments in the grant period of performance.

Full Text can be found at www.grants.gov

To locate the call quickly:

1) **Go to www.grants.gov**

2) Use **Quick Links** on the far right hand corner under **FOR APPLICANTS**, **Grant Search**.

3) Type in **NPSBAA10-002** under **Search by Funding Opportunity Number**.

THIS PAGE INTENTIONALLY LEFT BLANK

NPS-AM-10-022

PROCEEDINGS

OF THE

SEVENTH ANNUAL ACQUISITION RESEARCH SYMPOSIUM WEDNESDAY SESSIONS VOLUME I

Acquisition Research
Creating Synergy for Informed Change
May 12 - 13, 2010

Published: 30 April 2010

Prepared for: Naval Postgraduate School, Monterey, California 93943

THIS PAGE INTENTIONALLY LEFT BLANK

Table of Contents

Keynote: **The Honorable Robert O. Work, Under Secretary of the Navy**

Robert O. Work was confirmed as the Under Secretary of the Navy on May 19, 2009. In this capacity, Work serves as the deputy and principal assistant to the secretary of the Navy and acts with full authority of the secretary in the day-to-day management of the Department of the Navy. Work was a distinguished graduate of the Naval Reserve Officers Training Course at the University of Illinois, and was commissioned a second lieutenant in the US Marine Corps in August 1974. During his 27-year career, Work held a wide range of command, leadership, and management positions. He commanded an artillery battery and artillery battalion, and was the base commander at Camp Fuji, Japan. His last assignment was as Military Assistant and Senior Aide to the Honorable Richard J. Danzig, 71st secretary of the Navy.

After retiring from the Marine Corps, Work joined the Center for Strategic and Budgetary Assessments (CSBA), first as the senior fellow for maritime affairs, and later as the vice president for strategic studies. In these positions, he focused on defense strategy and programs, revolutions in war, Department of Defense transformation, and maritime affairs. He wrote and spoke extensively on US Navy and Marine Corps strategies and programs; directed and analyzed war games for the Office of Net Assessment and Office of the Secretary of Defense; contributed to Department of Defense studies on global basing and emerging military missions; and provided support for the 2006 *Quadrennial Defense Review*.

In addition, he studied and prepared several reports on future defense challenges, including the changing nature of undersea warfare, power projection against regional nuclear powers, and power projection against future anti-access/area denial networks. During this time, Work was also an adjunct professor at George Washington University, where he taught defense analysis and roles and missions of the armed forces.

In late 2008, Work served on President Barack Obama's Department of Defense Transition Team. In this role, he was the leader of the Department of the Navy issue team, and served on the defense policy, acquisition, and budget teams.

Work earned a Bachelor of Science degree in Biology from the University of Illinois; a Master of Science in Systems Management from the University of Southern California; a Master of Science in Space System Operations from the Naval Postgraduate School; and a Master in International Public Policy from the Johns Hopkins School of Advanced International Studies. He is a member of the International Institute for Strategic Studies (IISS).

THIS PAGE INTENTIONALLY LEFT BLANK

Plenary Panel – National Security Acquisition Challenges

Wednesday, May 12, 2010	
9:30 a.m. – 11:00 a.m.	**Chair: Dr. Jacques S. Gansler**, Director, Center for Public Policy & Private Enterprise, University of Maryland; former Under Secretary of Defense for Acquisition, Technology & Logistics **Panelists:** **Dr. Robert H. Trice**, Senior Vice President, Lockheed Martin Corporation **Dr. Steven J. Kelman**, Weatherhead Professor of Public Management, John F. Kennedy School of Government

Jacques Gansler—The Honorable Jacques S. Gansler, former Under Secretary of Defense for Acquisition, Technology, and Logistics, is a Professor and holds the Roger C. Lipitz Chair in Public Policy and Private Enterprise in the School of Public Policy, at the University of Maryland. He is the Director of both the Center for Public Policy and Private Enterprise and the Sloan Biotechnology Industry Center. As the third-ranking civilian at the Pentagon from 1997 to 2001, Professor Gansler was responsible for all research and development, acquisition reform, logistics, advance technology, environmental security, defense industry, and numerous other security programs.

Before joining the Clinton Administration, Dr. Gansler held a variety of positions in government and the private sector, including Deputy Assistant Secretary of Defense (Material Acquisition), Assistant Director of Defense Research and Engineering (electronics), Executive Vice President at TASC, Vice President of ITT, and engineering and management positions with Singer and Raytheon Corporations.

Throughout his career, Dr. Gansler has written, published, and taught on subjects related to his work. Gansler recently served as the Chair of the Secretary of the Army's "Commission on Contracting and Program Management for Army Expeditionary Forces." He is a member of the Defense Science Board and also a member of the National Academy of Engineering and a Fellow of the National Academy of Public Administration. Additionally, he is the Glenn L. Martin Institute Fellow of Engineering at the A. James Clarke School of Engineering, an Affiliate Faculty member at the Robert H. Smith School of Business and a Senior Fellow at the James MacGregor Burns Academy of Leadership (all at the University of Maryland). For 2003–2004, he served as Interim Dean of the School of Public Policy. For 2004–2006, Dr. Gansler served as the Vice President for Research at the University of Maryland.

Jacques Gansler
jgansler@udm.edu
Center for Public Policy & Private Enterprise, School of Public Policy, University of Maryland
2101 Van Munching Hall
Room 2211B
College Park, MD 20742

THIS PAGE INTENTIONALLY LEFT BLANK

Panel #2 – Assessment and Oversight of Major Defense Acquisition Programs

Wednesday, May 12, 2010	
11:15 a.m. – 12:45 p.m.	**Chair: Dr. Nancy Spruill**, Director, Acquisition Resources & Analysis, Office of the Under Secretary of Defense for Acquisition, Technology & Logistics ***The Effect of the Nunn-McCurdy Amendment on Unit Cost Growth of Defense Acquisition Projects*** **Jacques Gansler**, William Lucyshyn, and Adam Spiers, University of Maryland ***An Assessment of Acquisition Outcomes and Potential Impact of New Legislation and Policy Changes*** **Michael Sullivan**, Government Accountability Office ***Cost and Time Overruns in Major Defense Acquisition Programs*** **David Berteau**, Joachim Hofbauer, Gregory Sanders, and Guy Ben-Ari, Center for Strategic & International Studies

Dr. Nancy Spruill, Director, Acquisition Resources & Analysis, Office of the Under Secretary of Defense for Acquisition, Technology & Logistics

Dr. Nancy Spruill received Bachelor of Science degree in Mathematics, in 1971. From 1971 to 1983, she held a variety of positions with the Center for Naval Analyses, including Technical Staff Analyst, Professional Staff Analyst and Project Director. She earned her Master of Arts in Mathematical Statistics in 1975 followed by her Doctorate in 1980.

Dr. Spruill served on the staff of the Office of the Secretary of Defense from 1983 to 1993. Initially, she was the Senior Planning, Programming, and Budget Analyst in the Manpower, Reserve Affairs and Logistics Secretariat. Later, she served as the Director for Support and Liaison for the Assistant Secretary of Defense for Force Management and Personnel. Then she served as the Senior Operations Research Analyst in the Office of the Assistant Secretary of Defense for Program Analysis and Evaluation.

In 1993, she joined the staff of the Defense Mapping Agency (DMA), serving as the Chief of Programs and Analysis Division for the DMA Comptroller. Subsequently, she served as Acting Deputy Comptroller and was a member of the Reinvention Task Force for the Vice President's National Performance Review.

In March 1995, she was selected as the Deputy Director for Acquisition Resources for the Under Secretary of Defense for Acquisition Technology and Logistics (USD(AT&L)). In February 1999, she was appointed Director, Acquisition Resources & Analysis (ARA) for USD(AT&L). In this capacity, she is responsible for all aspects of AT&L'S participation in the Planning, Programming and Budgeting System (PPBS); the Congressional process; and the Defense Acquisition System. She serves as the Executive Secretary to the Defense

Acquisition Board and is responsible for the timely and accurate submission to Congress of Selected Acquisition Reports and Unit Cost Reports for Major Defense Acquisition Programs. She manages the Defense Acquisition Execution Summary monthly review of programs; monitors cost and schedule status of high interest programs; and conducts analyses of contract and program cost performance including analysis of the effective use of Integrated Program Management principles through the use of Earned Value Management. She leads the Department in developing plans to manage Property, Plant and Equipment, Inventory, Operating Materials and Supplies/Deferred Maintenance and Environmental Liabilities. She proposes modifications to, or acquisition of, new DoD feeder systems, in support of achieving an unqualified audit opinion on DoD Financial Statements as mandated by the Chief Financial Officers (CFO) Act. She also manages the studies program for OSD, oversees USD(AT&L)'s office automation system and manages its information system network. She serves as the focal point for DoD-wide software-intensive systems program initiatives to improve mechanisms for the management of defense acquisition programs; manages software intensive systems assessment initiatives; performs systemic analysis from independent expert program reviews to improve acquisition policy and education, and conducts special analyses for the Under Secretary.

Dr. Spruill has been a member of the Senior Executive Service since 1995. She is a certified Acquisition Professional and an active member of the American Statistical Association. Her many honors and awards include the Defense Medal for Exceptional Civilian Service, the Defense Medal for Meritorious Civilian Service, the Hammer Award and the Presidential Rank Award. She has contributed papers in publications of the statistics and defense analyses communities and authored articles in the general press on how politicians use--and abuse--statistics

The Effect of the Nunn-McCurdy Amendment on Unit Cost Growth of Defense Acquisition Projects

Jacques Gansler—The Honorable Jacques S. Gansler, former Under Secretary of Defense for Acquisition, Technology, and Logistics, is a Professor and holds the Roger C. Lipitz Chair in Public Policy and Private Enterprise in the School of Public Policy, at the University of Maryland. He is the Director of both the Center for Public Policy and Private Enterprise and the Sloan Biotechnology Industry Center. As the third-ranking civilian at the Pentagon from 1997 to 2001, Professor Gansler was responsible for all research and development, acquisition reform, logistics, advance technology, environmental security, defense industry, and numerous other security programs.

Before joining the Clinton Administration, Dr. Gansler held a variety of positions in government and the private sector, including Deputy Assistant Secretary of Defense (Material Acquisition), Assistant Director of Defense Research and Engineering (electronics), Executive Vice President at TASC, Vice President of ITT, and engineering and management positions with Singer and Raytheon Corporations.

Throughout his career, Dr. Gansler has written, published, and taught on subjects related to his work. Gansler recently served as the Chair of the Secretary of the Army's "Commission on Contracting and Program Management for Army Expeditionary Forces." He is a member of the Defense Science Board and also a member of the National Academy of Engineering and a Fellow of the National Academy of Public Administration. Additionally, he is the Glenn L. Martin Institute Fellow of Engineering at the A. James Clarke School of Engineering, an Affiliate Faculty member at the Robert H. Smith School of Business and a Senior Fellow at the James MacGregor Burns Academy of Leadership (all at the University of Maryland). For 2003–2004, he served as Interim Dean of the School of Public Policy. For 2004–2006, Dr. Gansler served as the Vice President for Research at the University of Maryland.

William Lucyshyn—William Lucyshyn is the Director of Research and Senior Research Scholar at the Center for Public Policy and Private Enterprise, in the School of Public Policy, at the University of Maryland. In this position, he directs research on critical policy issues related to the increasingly complex problems associated with improving public sector management and operations, and how government works with private enterprise.

Current projects include: modernizing government supply chain management, identifying government sourcing and acquisition best practices, and Department of Defense business modernization and transformation. Previously, Mr. Lucyshyn served as a program manager and the principal technical advisor to the Director of the Defense Advanced Research Projects Agency (DARPA) on the identification, selection, research, development, and prototype production of advanced technology projects.

Prior to joining DARPA, Mr. Lucyshyn completed a 25-year career in the US Air Force. Mr. Lucyshyn received his Bachelor's Degree in Engineering Science from the City University of New York, and earned his Master's Degree in Nuclear Engineering from the Air Force Institute of Technology. He has authored numerous reports, book chapters, and journal articles.

William Lucyshyn
lucyshyn@umd.edu
Center for Public Policy & Private Enterprise, School of Public Policy, University of Maryland
2101 Van Munching Hall
Room 2211B
College Park, MD 20742
301-405-8257 - Phone

Adam Spiers—Adam Spiers is a Faculty Research Assistant at the Center for Public Policy and Private Enterprise, in the School of Public Policy at the University of Maryland. In this position, he researches and writes draft versions of final reports on selected defense acquisition topics. As a Graduate Research Assistant at the think tank, he coauthored "Using Spiral Development to Reduce Acquisition Cycle Times" (September 2008) and "The Role of Lead System Integrator" (January 2009), both submitted to the Naval Postgraduate School.

Mr. Spiers received a Master's in Public Policy from the University of Maryland in May 2009. He previously graduated *summa cum laude* from the University of Maryland, College Park, with dual Bachelor of Arts degrees in Economics and History. Mr. Spiers currently plans to further his education by pursuing either a doctorate or a law degree.

Adam Spiers
aspiers@umd.edu
Center for Public Policy & Private Enterprise, School of Public Policy, University of Maryland
2101 Van Munching Hall
Room 2211B
College Park, MD 20742
301-405-9752 – Phone

Abstract

The Department of Defense's acquisition projects have experienced close to 50% unit cost growth since at least the 1950s. The most direct policy to curtail unit cost growth was the Nunn-McCurdy Amendment (NM), which Congress implemented in 1982 and later revised in 2006 and 2009. NM requires the DoD to report when the unit cost growth of any major defense acquisition program is known, expected, or anticipated by a program manager to exceed certain cost growth thresholds. Implementation of NM does not appear to have significantly impacted acquisition outcomes, although the long-term impact of recent revisions is indeterminate at this point in time. The authors performed several data analyses. The authors concluded that (1) the DoD's current metrics are not useful for determining the root cause of unit cost growth in acquisition programs and (2) programs that experience high, unit-cost growth are not randomly distributed. NM breach is highly correlated with a project's value size and the amount of estimating cost growth the program experiences. Two relevant cases, the Space-Based Infrared System (SBIRS) High program, and the Virginia-class Submarine (SSN 774), are analyzed. Finally, the authors make recommendations to improve NM and reduce unit cost growth.

Executive Summary

The Department of Defense (DoD) has faced significant acquisition problems over an extended period of time. As noted by one GAO report, the "DoD's major weapon system programs continue to take longer, cost more, and deliver fewer quantities and capabilities than originally planned" (Sullivan, 2008). For example, the programs that comprise the DoD's Major Defense Acquisition Projects (MDAPs)[1] for 2007 had an average program cost growth of 26% when compared to initial estimates, which collectively culminated in $295 billion dollars in additional costs (Sullivan, 2008). Given other pressing financial obligations,

[1] The DoD's largest programs, which represent roughly 80% of the DoD's acquisition budget in a given year (Younossi, Arena, Leonard, Roll & Sollinger, 2007).

the DoD cannot afford to incur similar development problems in the future that it has experienced in the past.

Cost growth is defined as the positive difference between actual cost and budgeted costs. Due to its relative ease of measurement, cost growth provides a simple barometer to determine if the acquisition process is achieving its stated goals or not. Since the 1950s, numerous reports have found that, in general, the DoD's acquisition process experiences high cost growth at both the program and unit levels.

Congress has made several attempts to implement reforms that would control program and unit cost growth, but these have not achieved their intended results. The most direct policy that attempted to curtail unit cost growth was the Nunn-McCurdy Amendment (NM), which Congress implemented in 1982. The law was significantly modified in 2006 and 2009 (as described below).

NM requires the DoD to report when unit cost growth of any major defense acquisition program was "known, expected, or anticipated" by a program manager to exceed certain cost growth thresholds (US Congress, 2007). More specifically, NM stipulates two levels of unit cost growth breach, the significant level and critical level. A significant unit cost breach occurred if a program experienced cost growth over 15% of the *current* baseline estimate, whereas a critical unit cost breach occurred if a program experiences cost growth of 25% over the *current* baseline estimate. A unit cost breach occurs if a program experiences unit cost growth above specified thresholds as measured by either program acquisition unit cost[2] (PAUC) or average procurement unit cost[3] (APUC).

The NM law requires a program manager to fulfill specific criteria when a program breaches. For a significant unit cost breach, the "Service Secretary must notify Congress within 45 days after the report (normally program deviation report) upon which the determination is based… [and] submit a Selected Acquisition Report (SAR) with the required additional unit cost breach information" (Axtell & Irby, 2007).. For a critical unit cost breach, the Under Secretary of Defense for Acquisition, Technology and Logistics (USD(AT&L)) must fulfill and complete all 'significant' breach requirements and must additionally certify to Congress within 60 days of the SAR that the program meets four criteria: (1) the system is essential to national security; (2) there are no alternatives to such system which will provide equal or greater military capability at less cost; (3) the new estimates of the unit cost are reasonable; and (4) the management structure for such major defense system is adequate to manage and control unit cost (US Congress, 2007).

Implementation of NM did not seem to have any significant impact on acquisition outcomes. The most consistent criticism of NM was that the measure was ineffective because programs would avoid incurring a NM breach by rebaselining a program (i.e., establishing a new "current" baseline)—a procedure that did not require Congressional notification (Axtell, 2006).

The NM statute was amended in 2006 to close the rebaselining loophole. The new provision included language specifying a second condition for incurring a NM breach: unit cost growth over the *original* baseline estimate. A significant unit cost breach occurs when cost growth exceeds 30% of the *original* baseline and a critical unit cost breach occurs when

---

[2] (Total Development cost + Procurement cost + Construction cost)/(Total program quantity)

[3] (Total Procurement cost)/(Procurement Quantity)

cost growth exceeds 50% of the *original* baseline estimate. The revision did not change the reporting requirements for either the significant or critical unit cost breach.

Soon after the implementation of the 2006 NM revision, the DoD reported 40 of the 85 current MDAP programs were experiencing unit cost growth high enough to warrant a Nunn-McCurdy breach. Although 25 of these programs experienced unit cost growth of over 50% relative to their original baseline, the DoD did not report programs as having incurred a Nunn-McCurdy breach as the *National Defense Authorization Act* permitted the "original baseline estimate to be revised to the current baseline estimate as of January 6, 2006" (Office of the Under Secretary of Defense, 2006). Between 2006 and 2007, 16 additional programs experienced unit cost growth high enough to incur a NM breach. Despite the impact of the new legislation on the number of programs that breached, it is too soon to determine the long-term impact of the legislation on current acquisition performance, although the immediate short-term impact has been to place greater visibility, as well as a great deal more emphasis, on the unit cost growth, relative to the original program baseline.

Congress again amended NM with *The Major Weapons Systems Acquisition Reform Act of 2009*. This law added two requirements to the process of recertifying programs that incur a NM breach. A program with a NM unit cost breach now must (a) rescind the most recent Milestone approval and (b) receive a new Milestone approval before any actions regarding the contract may continue. The new Milestone approval requires a certification that the costs of the program are reasonable, and the certification must be supported by an independent cost estimate that includes a confidence level for the estimate (US Congress, 2009). This statute was implemented too recently to evaluate its impact upon the defense acquisition process.

The authors performed several data analyses, based on the limited, publicly available information, to determine if any reported variables were correlated in a statistically significant way with NM unit cost breach. The data analysis computed several tests of independence, using Fisher's "exact test." This analysis produced two conclusions. First, the DoD's current metrics are not useful for determining the root cause of unit cost growth in acquisition programs. Second, despite data limitations, it appears that programs that experience high, unit-cost growth are not randomly distributed. Going further, programs that experience a NM unit cost breach appear to have the strongest relationship with two factors—value size of project and the estimating cost category. Programs appear much more likely to breach if the total program has a large value (above $7.95 billion) and positive cost growth from the estimating category. Conversely, programs with small total program value (below $3.5 billion) appear to rarely breach.

The report analyzed two relevant case studies. The Space-Based Infrared System (SBIRS) High program highlights how the threat of a NM breach does not necessarily lead to improved acquisition outcomes, and as a result the program subsequently breached. The Virginia-class Submarine (SSN 774) program highlights how programs that experience high unit cost growth can implement policies to achieve substantial cost reductions (i.e., the desired action, to avoid a NM breach).

This study resulted in 8 findings: (1) unit cost growth has remained high; (2) few programs incurred a NM breach until the recent 2006 revision of the law that requires programs to consider unit cost growth above a program's original baseline; (3) the DoD's data collection has been inconsistent (regarding: definitions, moving baselines, quantities, etc.); (4) the DoD often has not conducted systematic analysis of root-cause problems; (5) limited and inconsistent data undermines an effective analysis; (6) NM may identify acquisition problems too late in the development process to allow program reforms to be effective; (7) NM's effectiveness may be limited by its focus on the development and procurement of assets, as opposed to the entire life-cycle of the program; and (8) recent legislation has not been implemented long enough to evaluate its impact on DoD acquisition processes.

The authors developed 8 recommendations. Regarding NM, the DoD should: (1) develop a system to determine and distribute lessons-learned from a NM breach, throughout the DoD and (2) develop leading indicators. In order to control cost growth, the DoD should (1) fully embrace and implement the *Weapon Systems Acquisition Reform Act of 2009* legislation as prior attempts to reform DoD acquisitions have been ineffective in large part due to the DoD's institutional resistance; (2) implement a more complete acquisition data information system; (3) consider lifecycle costs when rendering acquisition decisions; (4) directly address the lack of incentives that allow current underlying problems to persist; (5) work with Congress to increase funding flexibility (e.g., being able to use production money to increase development costs so as to save far more significant unit production costs); and (6) provide programs with greater requirements flexibility (e.g., allowing cost/performance trade-offs, especially for "block I" of the deployed system, so that the last 5 to 10% of performance "requirements" doesn't double the unit costs).

References

Axtell, L. (2006, May 10). Nunn-McCurdy unit cost reporting. Paper read at 2006 Business Managers' Conference.

Axtell, L., & Irby, W. (2007, October 30-31). APB, SAR, and Nunn-McCurdy status. Paper read at DAMIR Conference, Fort Belvoir, VA.

OUSD (Acquisition Resources and Analysis). (2006). *Selected Acquisition Report (SAR) summary tables as of date: December 31, 2005*. Retrieved from http://www.acq.osd.mil/ara/am/sar/2005-DEC-SARSUMTAB.pdf

Sullivan, M.J. (2008). *Defense acquisitions: Fundamental changes are needed to improve weapon program outcomes* (GAO-08-1159T). Washington, DC: GAO.

US Congress. (2007). *Unit cost reporting*. Title 10, Subtitle A, Part IV, Chapter 144, Section 2433. Washington, DC: US GPO.

US Congress. (2009). *Weapon systems acquisition reform act of 2009* (Public Law 111-23). Washington, DC: US GPO.

Younossi, O., Arena, M.V., Leonard, R.S., Roll, C.R., Jr., Jain, A., & Sollinger, J.M. (2007). *Is weapon system cost growth increasing? A quantitative assessment of completed and ongoing programs* (RAND Report MG-588-AF).

Cost and Time Overruns in Major Defense Acquisition Programs

David Berteau—David J. Berteau is a Senior Adviser and Director of the CSIS Defense-Industrial Initiatives Group, covering defense management, programs, contracting, and acquisition. His group also assesses national security economics and the industrial base supporting defense. Mr. Berteau is an adjunct professor at Georgetown University, a member of the Defense Acquisition University Board of Visitors, a director of the Procurement Round Table, and a fellow of the National Academy of Public Administration. He also serves on the Secretary of the Army's Commission on Army Acquisition and Program Management in Expeditionary Operations.

Joachim Hofbauer—Joachim Hofbauer is a research associate with the Defense-Industrial Initiatives Group at CSIS, where he specializes in acquisition policy and defense industrial base issues. His current work focuses on US defense acquisition reform, the European defense market, and root cause analysis for cost and schedule overruns in major defense acquisition programs.

Before joining CSIS, Mr. Hofbauer worked as a defense analyst in Germany and the United Kingdom. Mr. Hofbauer holds an MA with honors in security studies from Georgetown University and a BA in European studies from the University of Passau.

Gregory Sanders—Gregory Sanders is a research associate with the Defense-Industrial Initiatives Group at CSIS, where he gathers and analyzes data on US defense acquisition and contract spending. Past projects include studying US government professional services contracting, US defense export controls, and European defense trends. His specialty is data collection, analysis, and visualization.

Mr. Sanders holds an MA in international relations from the University of Denver and a BA in government and politics and a BS in computer science from the University of Maryland.

Guy Ben-Ari—Guy Ben-Ari is Deputy Director of the Defense-Industrial Initiatives Group at the Center for Strategic International Studies, where he works on projects related to the US technology and industrial bases supporting defense. His current research efforts involve defense R&D policies, defense economics, and managing complex defense acquisition programs.

Mr. Ben-Ari holds a Bachelor's degree in political science from Tel Aviv University, a Master's degree in international science and technology policy from the George Washington University, and is currently a PhD candidate (ABD) at the George Washington University.

Abstract

Cost and time overruns in Major Defense Acquisition Programs (MDAPs) have become a high-profile problem attracting the interest of Congress, government and watchdog groups. According to the GAO, the 96 MDAPs from FY2008 collectively ran $296 billion over budget and were an average of 22 months behind schedule. President Obama's memo on government contracting of 4 March 2009 also highlighted this issue.

This paper presents interim findings of research on the root causes of cost and schedule delays for MDAPs. This research is ongoing and will incorporate the 2010 SAR data. The final findings and policy recommendations will be presented at the May 2011 Naval Post Graduate School annual Acquisition Symposium.

Introduction

Cost and time overruns in Major Defense Acquisition Programs (MDAPs) have become a high-profile problem attracting the interest of Congress, government and watchdog groups. According to the GAO, the 96 MDAPs from FY2008 collectively ran $296 billion over budget and were an average of 22 months behind schedule. President Obama's memo on government contracting of 4 March 2009 also highlighted this issue.

This paper presents interim findings of research on the root causes of cost and schedule delays for MDAPs. This research is ongoing and will incorporate the 2010 SAR data. The final findings and policy recommendations will be presented at the May 2011 Naval Post Graduate School annual Acquisition Symposium.

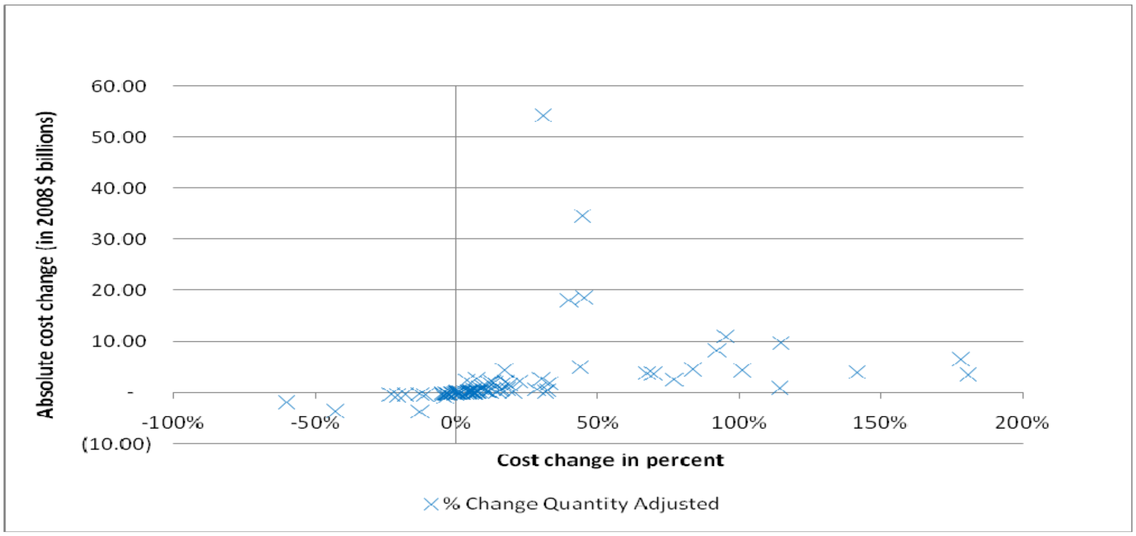


Figure 1. Relative Cost Growth Versus Absolute Cost Growth for FY2008 MDAPs

Note: Only FY2008 MDAPs with a baseline estimate beyond Milestone B in the September 2008 SAR were included

(Source: September 2008 SAR; analysis by CSIS Defense-Industrial Initiatives Group)

Problem Definition

Past studies on this topic either have not offered rigorous data analysis or were focused on a narrow aspect of the problem, such as technical maturity. As a result, acquisition reform efforts, most recently the Weapon Systems Acquisition Reform Act of 2009, are hampered by an insufficient analytical basis.

For instance, in its annual assessment of selected weapon systems, the GAO primarily focuses on technology maturity and associated program decisions as causes for these problems. Former Under Secretary of Defense for Acquisition Technology & Logistics John Young claimed in a memorandum on March 31, 2009, that many of the allegations of the GAO are based on inadequate analytical methods and that consequently many of the results are misleading.

This disagreement is exemplary of the diverging set of opinions that exists regarding the root causes of MDAPs cost overruns and schedule delays. The result amplifies disagreement regarding potential fixes. On the government side, Senator McCain identified the usage of cost plus contracts as a major source for cost increases and Secretary Gates pointed towards the contract structures as a key source of cost and schedule overruns in some MDAPs. Defense contractors, on the other hand, regularly cite the altering of requirements in advanced program stages as an important factor for cost increases.

The currently ongoing process of reforming and fixing defense acquisition system still lacks the foundation of a detailed evaluation of the causality chain of cost overruns and program delays of MDAPs. This lack of understanding of underlying mechanism makes the design of adequate solutions inherently difficult and renders them potentially ineffective. This study directly aims at developing the urgently needed knowledge base that will better guide efforts to correct the growing trends of cost increases and schedule overruns.

Methodology

This brief analysis a series of variables—namely realism of baseline program cost estimates, government management and oversight, the role of contractors and lead military services, levels of competition, and contract structures—to determine what factors might contribute to the observed cost overruns in the execution of MDAPs.

The research draws on four primary data sources:

Selected Acquisition Reports (SARs): The SARs track Major Defense Acquisition Programs reporting on their schedule, unit counts, total spending, and progress through milestones. The unit of analysis is the programs themselves, making it the ideal source for top level analysis.

Federal Procurement Data System (FPDS): The FPDS is a database of every government contract, with millions of entries each year. Each entry has extensive data on the contractors, contract type, competition, place of performance, and a variety of other topics as mandated by Congress. Cross-referencing individual contracts with MDAPs is possible using the system equipment codes (which match up with those of MDAPs). This source provides the most in-depth data on the government contracting process.

Department of Defense Budget Documents: In addition to budget data, these documents provide topical information on each MDAP and its subcomponents. They will primarily be used to categorize projects as well as to support and double check spending figures from the other two sources.

The initial analysis phase focuses on MDAPs from the FY2008 MDAP list. Within this sample group, the analysis is limited to 87 MDAPs with cost estimates set at Milestone B or beyond. That gate is meant to be a hurdle that requires programs to reach a certain level of technological maturity. As a result Milestone B "is normally the initiation of an acquisition program." This common starting point ensures that only programs in a relatively mature acquisition phase are compared. Unfortunately, full data is not available on all 87 MDAPs when examining contract type and competition because only a majority of the programs have at least 50% of the SARs contract value accounted for in 2004-2008 FPDS data. The "unclear" category is used to signify this missing data in competition and contract type findings. In addition, FPDS totals for program spending are sometimes higher than the funding status according to the SARs. In those cases, the SAR totals are treated as the more reliable figure.

This preliminary snapshot provides an adequate starting point for detecting correlations between a series of potentially relevant factors and cost growth. Subsequent analysis will examine multiple factors at the same time, expand the breadth of the sample group and will also test for correlations with regard to schedule overruns.

Analysis

The initial analysis focuses on examining the impact of baseline cost estimates, quantity and schedule changes, as well as engineering problems; the extent of competition in contracts (full and open, partial, none)[4]; contract structure; lead branch of military service; and identity of prime contractor on the cost performance on MDAPs.

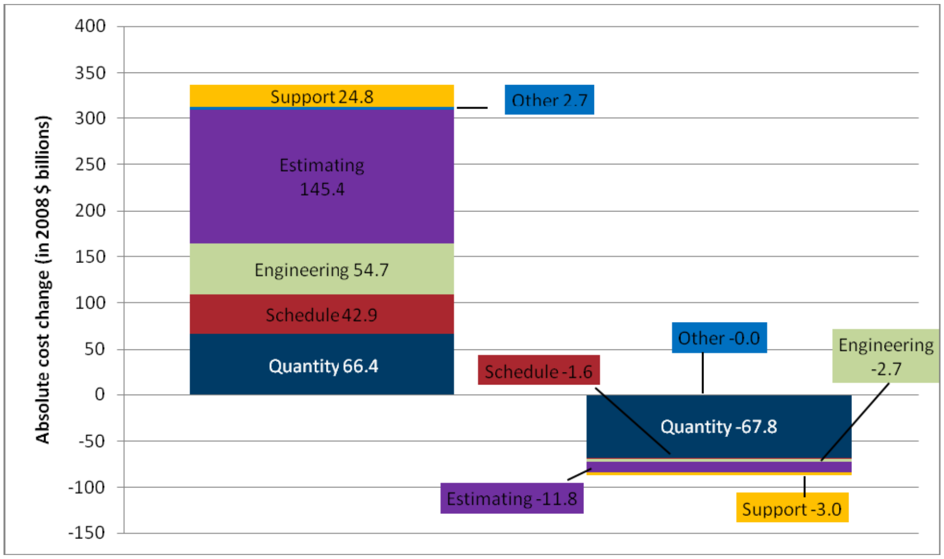


Figure 2. Functional Reasons for Cost Growth
(Source: September 2008 SAR; analysis by CSIS Defense-Industrial Initiatives Group)

Breaking down cost growth by functional areas as provided in the SARs identifies mistakes in the estimating process as the primary driver for cost growth, being responsible for $145.5 billion in cost growth for the 87 MDAPs analyzed.

Another noteworthy observation is the fact that the cost savings achieved through quantity changes almost equals the cost growth originating from changes in unit numbers. Quantity based changes are unlike the five other types of changes as the SARs adjust the top-line cost overrun figures to remove the impact of quantity changes. The key distinction is that for programs with upfront research and development costs, reducing the number of units lowers the overall cost but increases the unit cost. In turn, cost increases deriving from increases in the number of units require a higher overall budget but lower the price per unit. Similarly, Nunn-McCurdy breaches are based on the growth in the acquisition unit cost and not the overall cost.

[4] Full competition refers to programs competed under full and open competition with at least two bidders. Partial competition includes all other competed contracts such as follow-ons to competed contracts, competitions where the number of bidders is legally limited, and full and open competition with only a single bidder.

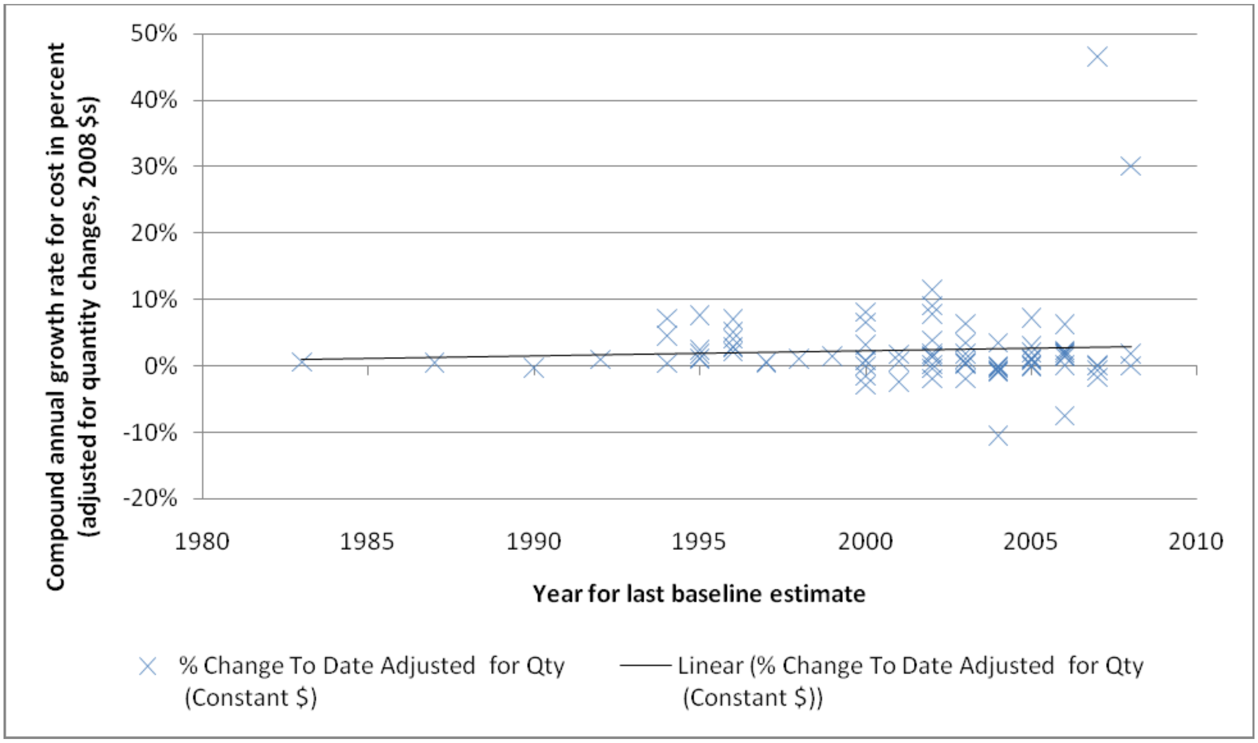


Figure 3. Time-cost Correlation
(Source: September 2008 SAR; analysis by CSIS Defense-Industrial Initiatives Group)

The next explanatory variable examined for its impact on program performance is the time-cost growth correlation. If cost increases accrue over time then programs with an older baseline estimate would tend to accumulate relatively higher cost increases. The data for the analyzed programs shows that older programs indeed experience larger overruns.

When measured in compound annual growth rate[5] rather than aggregate relative cost growth, <u>the time-cost growth correlation is almost constant</u>. This does not only provide further evidence for the assertion that cost growth occurs steadily throughout the program lifespan, but it also suggests that younger programs are not performing better than older programs. On the other hand, this sample does not include older programs that were cancelled. Future research with a broadened sample set will be better able to avoid this confounding factor and thus provide more insight into the successes and failures of past reform efforts.

---

[5] The compound annual growth rate describes the average year-to-year cost growth of a program spending since its baseline. Thus if comparing two programs with same percentage of cost growth since their baseline estimate, the program with an earlier baseline year would have a smaller compound annual growth rate.

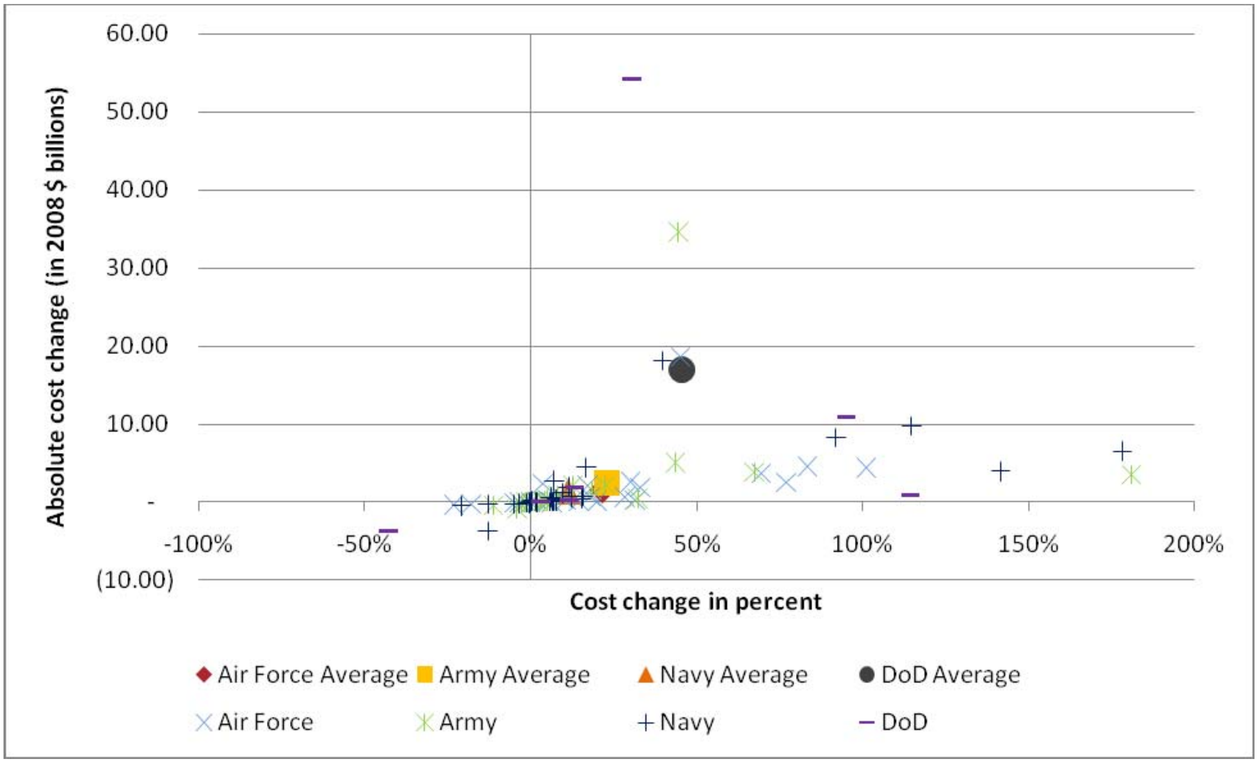


Figure 4. Cost Overruns by Lead Service (I)
(Source: September 2008 SAR; analysis by CSIS Defense-Industrial Initiatives Group)

The analysis on the correlation between the lead branch of military service responsible for MDAPs and cost growth patterns reveals that programs led by the Navy appear to perform best, followed by the Air Force and Army, while DoD-wide programs tend to accrue significant higher cost overruns. As broken down by the SARs, DoD-wide refers not just to programs managed by DoD agencies but also joint programs such as the Joint Strike Fighter. The outcome of this data analysis might be skewed based on the relatively small sample group utilized in this preliminary analysis. For instance, it appears that the DoD-wide category might be heavily influenced by the negative developments for the Joint Strike Fighter program. As for the other components, further analysis with larger sample groups are required to validate observed trends.

Any conclusions identifying superior program management of existing programs by the better performing services as means of avoiding cost growth would be premature, even if additional data and analysis will confirm this variation in cost performance based on lead service. A number of other factors occurring before Milestone B may explain the differences, such as a tendency toward less risk-prone MDAPs or better cost estimating at the outset of programs. Further research will be needed to analyze the underlying causality and detect the true root causes for these trends.

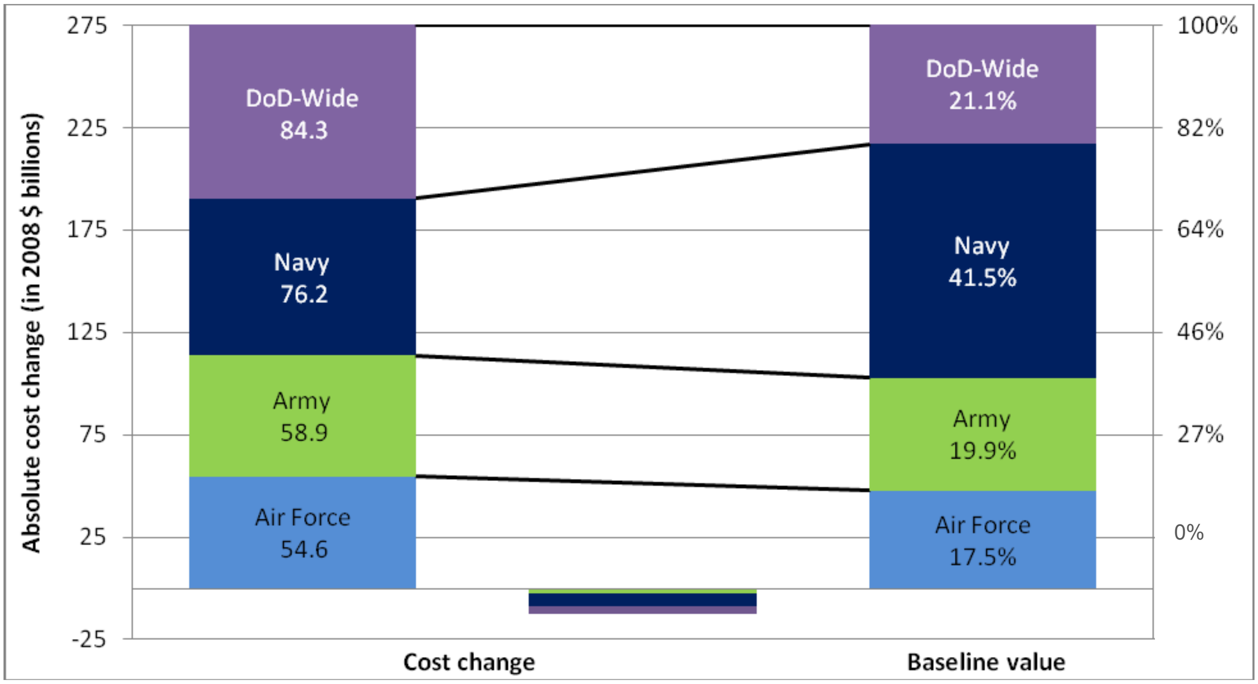


Figure 5. Cost Overruns by Lead Service (II)
(Source: September 2008 SAR; analysis by CSIS Defense-Industrial Initiatives Group)

Comparing the share of cost growth for which each service is responsible with the share of contract value, based on baseline estimates, which each service is managing supports the poor cost performance of DoD-wide managed MDAPs. DoD-wide led programs are responsible for an over-proportional large share of absolute cost overruns. The picture is reversed for the Navy, meanwhile for the Air Force and the Army the share of cost overruns and is slightly larger than the share of baseline value. This comparison provides further support for the assertion that Navy managed MDAPs over-perform, while DoD-wide managed MDAPs underperform. However, the level of analysis conducted so far does not allow for any firm conclusions on the actual role of the Navy program management skills in these trends.

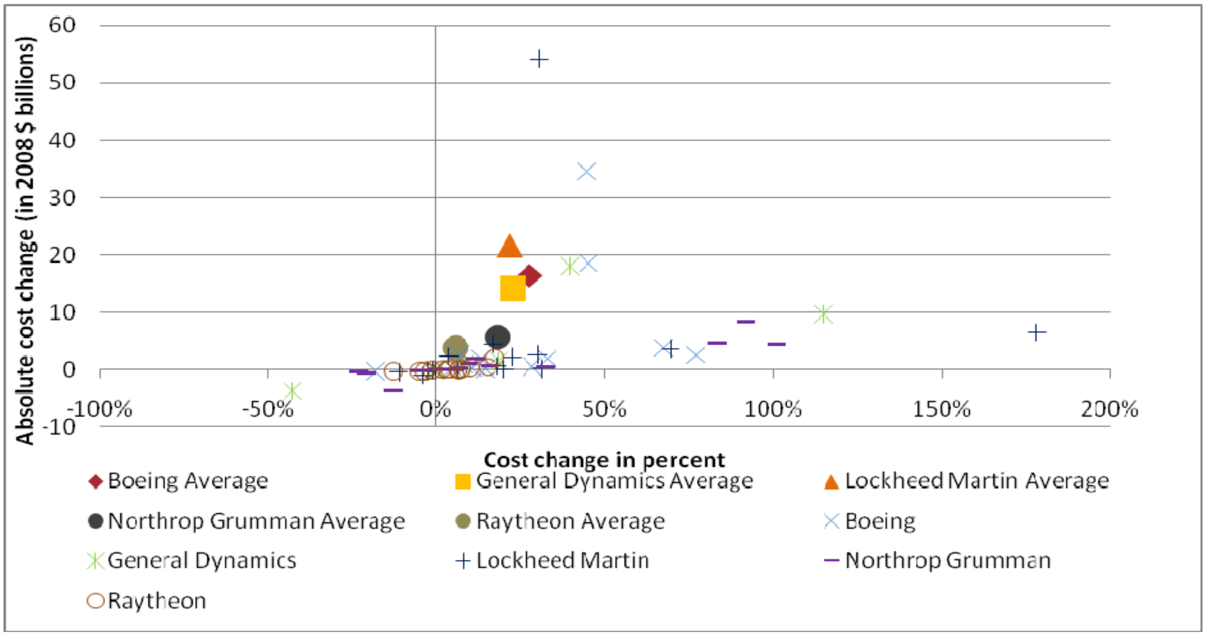


Figure 6. Figure 6. Cost Overruns by Prime Contractor (I)
(Source: September 2008 SAR; analysis by CSIS Defense-Industrial Initiatives Group)

Another predictor for program performance could be the identity of the prime contractor executing a given program. The picture becomes a lot more complex, based on the amount of actors involved. One striking trend that is visible for the “big five” US defense companies is the fact that Raytheon on average appears to deliver significantly better cost performance outcomes than the other four defense companies.

Again, the preliminary character of the analysis does not fully validate these findings. In addition, even if confirmed, it would be premature to start praising Raytheon for superior program execution, as other factors such as specialization in technologically more mature program areas might be the true drivers behind this trend. As was the case for the breakdown by lead service, further research will be needed to analyze the underlying causality.

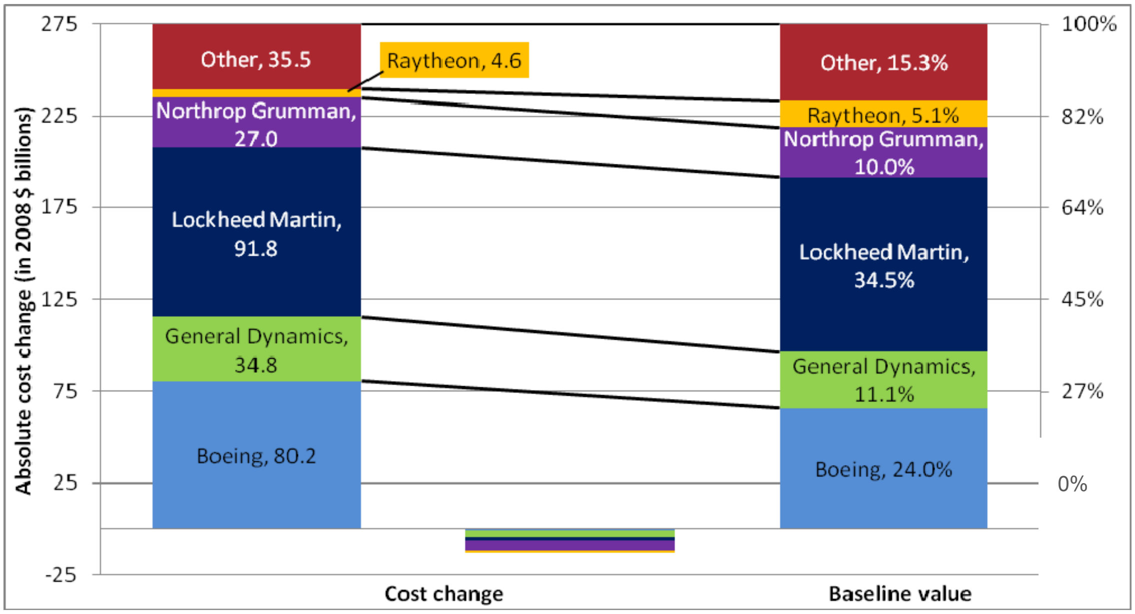


Figure 7. Cost Overruns by Prime Contractor (II)
(Source: September 2008 SAR; analysis by CSIS Defense-Industrial Initiatives Group)

The comparison between the share of cost growth and the share of contract value for MDAPs, aggregated by prime contractor correlates with the finding that Raytheon managed MDAPs appear to exhibit the best cost performance amongst the "big five" defense companies. However, the above graph also shows that the respectable performance of Raytheon contracted MDAPs has only a marginal impact on aggregate cost growth bottom line due to the small contract value share of Raytheon led programs.

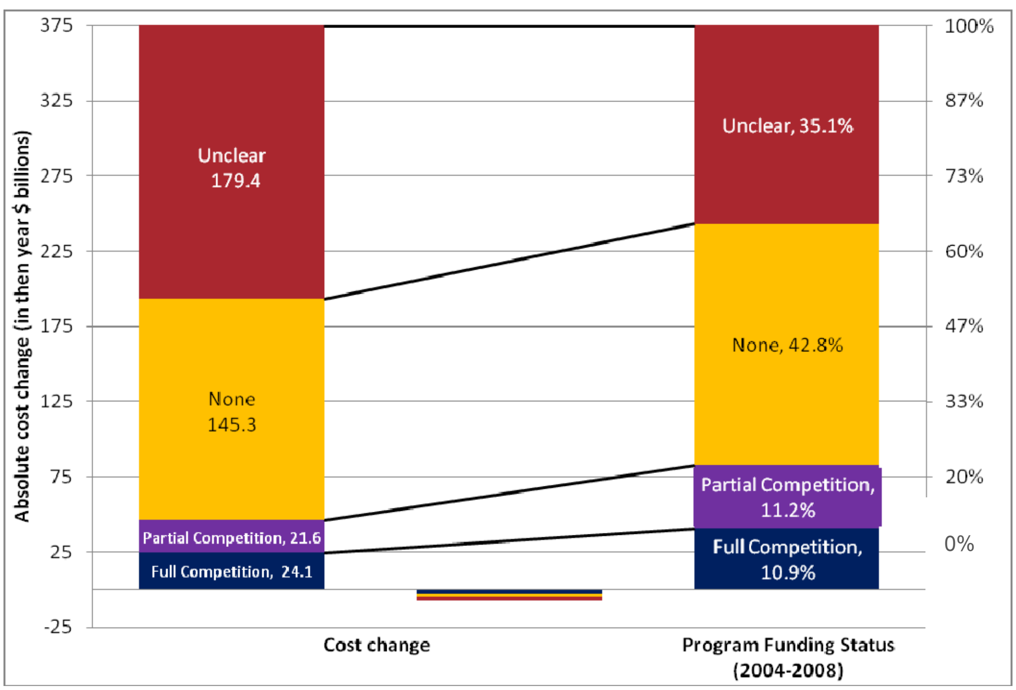


Figure 8. Cost Overruns by Type of Competition
(Source: September 2008 SAR; analysis by CSIS Defense-Industrial Initiatives Group)

Contract awarding mechanisms could potentially also have an impact on cost performance of MDAPs. Competitive contracts[6] are outperforming contracts awarded with no competition or under unclear circumstances with regard to cost performance. This might indicate that competition either results in more realistic bids or that winning companies have more incentives to keep costs under control. This advantage holds for the category of partial competition, which includes cases open for competition but with only a single bidder, follow-ons to competed contracts, and competitions with a legally limited pool of applicants. In fact, somewhat surprisingly, partially competed MDAPs appear to have lower overruns than fully competed ones. However, without further study, it is impossible to say whether this is due to benefits of partially competitive structures or is fully competitive procedures are deemed to be a necessary for high risk programs.

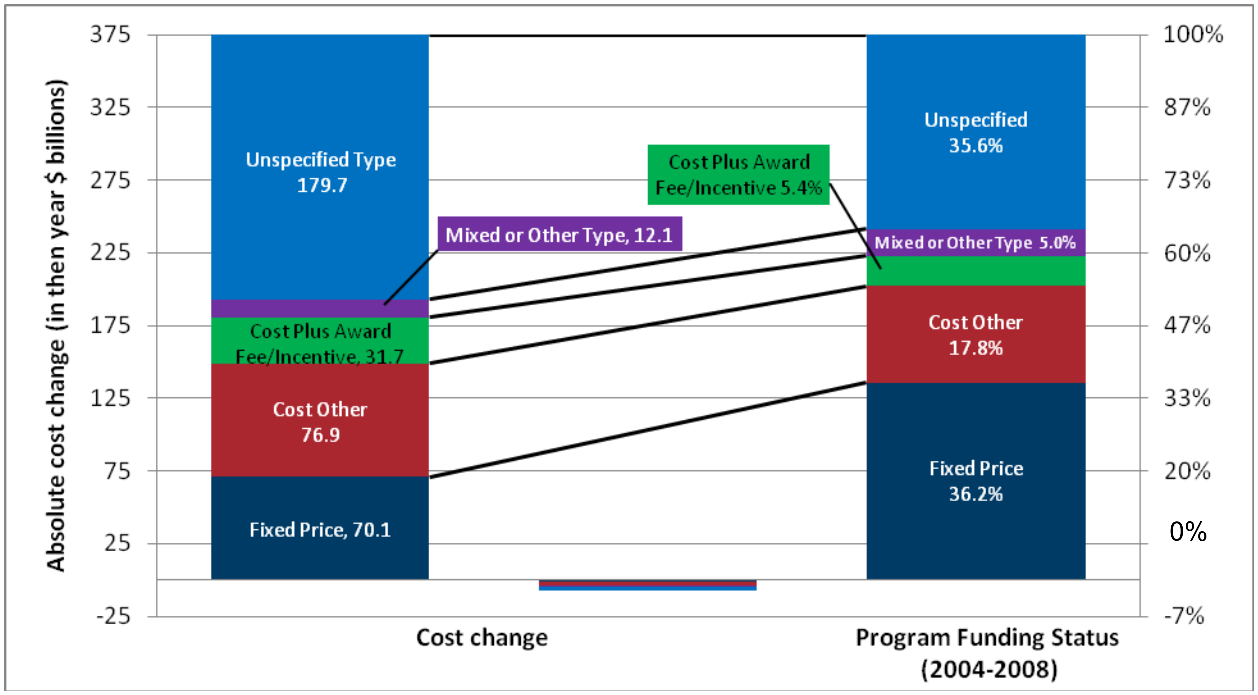


Figure 9. Cost Overruns by Contract Type

(Source: September 2008 SAR; analysis by CSIS Defense-Industrial Initiatives Group)

Contract structure provides another possible determining factor for the performance of MDAPs. One key observation is that fixed price contracts appear to over-perform and unspecified contract types appear to under-perform when comparing the share of cost growth and the share of contract value for MDAPs.

Acquisition reformers often point towards over- and misuse of cost plus contracts as a factor driving cost overruns. Yet as the comparison reveals the use of cost plus contracts, award fee and incentive as well as conventional, seem not to be responsible for dramatically over-proportional cost growth. In addition, fixed price contracts are often the vehicle of choice for mature technology in full rate production, which are generally considered low risk.

---

[6] Full competition refers to programs competed under full and open competition with at least two bidders. Partial competition includes all other competed contracts such as follow-ons to competed contracts, competitions where the number of bidders is legally limited, and full and open competition with only a single bidder.

Preliminary Findings

The initial analysis yielded a number of preliminary trends for determining the sources of cost overruns in MDAPs. One key finding from the SAR data is that overly optimistic cost estimating is responsible for almost half of the accumulated cost overruns. In fact, of all the services the Army is the only one with cost estimating not being the primary reason for cost growth.

Of the tested variables, only the time-cost correlation appears to have no impact on cost overrun developments once accounted for on a compound annual growth rate. This suggests that program performance might not have been improving in recent times. If this trend is further validated it hints toward the concerning conclusion than any acquisition reform efforts prior to 2008 have so far failed to create any improvements for cost performance. In this context, it must, however, be noted that cost performance as measured in the SARs constitutes clearly a lagging indicator for the impact of any acquisition reform. In addition, some of the worst performing older programs of the past have already been cancelled and if included would increase the overrun compound annual growth rate for earlier years.

The examination of all of the other examined variables reveals patterns that suggest that each of them, or associated secondary or tertiary factors, could play a role in explaining the occurrence of MDAP cost overruns. While one service may appear to have the best cost performance; or one company may seem to deliver fewer cost-overrun programs of the "big five" defense contractors; fixed price contracts appear to constitute the contract vehicle with the best cost growth control; and awarding contracts in a competitive fashion seems to ensure better cost performance than alternative awarding mechanism, all of these trends need to be further validated through additional analysis, incorporating larger sample groups. Afterwards, more rigorous quantitative and qualitative analysis is required to identify the actual root causes for cost overruns, which might be only masked by the examined variables. Ongoing research will lead to better answers about these root causes.

Panel #3 – Cost Analysis Support to Acquisition

Wednesday, May 12, 2010	
11:15 a.m. – 12:45 p.m.	**Chair: Dr. Daniel Nussbaum**, Naval Postgraduate School; former Director, Naval Center for Cost Analysis
	Cost-Benefit Study of a Project to Lower Cost and Improve Fleet Readiness Through Integrating the Management of Technical Information
	Dan Levine and **Stan Horowitz**, Institute for Defense Analyses, **Wayne Gafford**, Naval Surface Warfare Center, Port Hueneme
	Queues in Acquisition
	William Wiltschko, Deloitte Consulting LLP
	Modeling, Simulation, and Analysis: Enabling Early Acquisition Decisions
	Fred Hartman, Institute for Defense Analyses

Cost-Benefit Study of a Project to Lower Cost and Improve Fleet Readiness through Integrating the Management of Technical Information

Dan Levine— Dr. Dan Levine holds a PhD in physics and an MA in economics. After a time at Harry Diamond Labs and the National Bureau of Standards, he has been working since 1962 on economics and cost-effectiveness studies at the Center for Naval Analyses and the Institute for Defense Analyses. His areas of research have been in Navy and Army logistics, and a variety of areas in OSD planning. Levine gives the lecture on cost effectiveness analysis for the course in cost analysis that the IDA Cost Analysis and Research Division gives annually to graduate students in operations research at George Mason University.

Abstract

This paper describes a cost-benefit analysis by the Institute for Defense Analyses of the "Bridge Project" that ADL (Advanced Distributed Learning) is conducting for the Office of Secretary of Defense for Acquisition, Technology and Logistics (OSD(AT&L) to improve the management of Integrated Logistics Support (ILS). The Project is part of the OSD RTOC program (Reduction in Total Ownership Cost). The Bridge Project focuses on integrating (Bridging) the management and production of technical manuals and training courses. The benefits would be lower cost to produce these manuals and courses in the future, and improved readiness through insuring the delivery of consistent and up-to-date logistics support to the Fleet.

Manuals and courses are currently produced by entirely separate processes. Tech writers and course developers obtain contractor data on systems and equipments in parallel, they express the information in different formats, they organize the data in different structures, and they store the data in different repositories. Cost is therefore higher because of duplication of resources and the difficulties in re-using data. The lack of integration can also reduce readiness, since it opens up the possibility that the tech manuals and training courses present disparate information, thus depriving ship operators and maintainers of the most effective support.

The Bridge Project seeks to relieve these problems by designing new software, technical and business processes to integrate the production of technical manuals and training courses. All technical and learning content would be expressed by the same digital specification (the S1000D industry specification), they would employ the same structure (Data Modules), and the data would all be stored in the same repositories (Common Source Data Bases, or CSDBs). The project is developing an API (Application Programming Interface) to enable course developers to exchange data with any CSDB, and a Web Service to more quickly update tech manuals and training courses in response to Engineering Change Proposals (ECPs).

The analysis finds that the Bridge would achieve net benefits (benefits less costs) of approximately $87 million over 10 years, far more than enough to cover the $8.7 million 10-year cost of producing all Navy HM&E (Hull, Mechanical and Electrical) technical manuals and training courses delivered by Navy e-Learning, a part of the Naval Education and Training Command (NETC). A sensitivity analysis of the most uncertain inputs yields a range of results (net benefits) from $32 million to $166 million.

The Bridge could also contribute to shipboard readiness by insuring the Navy's policy of providing up-to-date and consistent information to the Fleet upon installation of new systems and equipment. A parametric analysis indicates that by increasing the availability of the electronic, ordnance and HM&E components of a single new DDG 1000 destroyer for only a single day would increase effectiveness the Navy values at $2 million.

Executive Summary

The Institute for Defense Analyses (IDA) is conducting a cost-benefit analysis of the "Bridge Project" that the Advanced Distributed Learning (ADL) office is conducting for the Office of the Secretary of Defense for Acquisition, Technology and Logistics (OSD(AT&L)). The project is part of OSD's RTOC program (Reduction in Total Ownership Cost). The project's focus is on improving how the Navy manages several aspects of Integrated Logistics Support (ILS) in order to reduce the cost of producing technical manuals and training courses. A related benefit is increasing readiness through insuring the Navy's policy of having the appropriate logistics support on hand when new systems and equipment upgrades are fielded.

Under current ILS management, manuals and courses are produced by entirely separate processes. Tech writers and course developers obtain contractor data on new systems and equipment upgrades in parallel. They express the data in different formats, they organize the data in different structures, and they store the data in different repositories. The cost of producing logistics support is therefore higher because of the duplication of resources and the difficulties in re-using data. The lack of integration can also reduce readiness, since it opens up the possibility that the tech manuals and training courses present disparate information. And there may be delays in updating the information in response to Engineering Change Proposals (ECPs). These disparities and delays can deprive ship operators and maintainers of the most up-to-date information on their systems and equipment.

The Bridge project is designing a new process—new software, technical and business processes to integrate ("Bridge") the production of technical manuals and training courses. The initial beneficiary of the funding is the Littoral Combat Ship (LCS) Mission Modules Program (PMS 420), which is integrating the Mission Modules into the LCS. Under the Bridge Project, all technical and learning content would be expressed by the same digital specification (the S1000D industry specification), organized by the same structure (Data Modules), and stored in the same repositories (Common Source Data Bases). The project is also developing an Application Programming Interface (API) to grant all course developers access to any CSDB, and a Web Service to quickly identify the technical and learning content that must be reviewed for updating in response to Engineering Change Proposals (ECPs).

The cost-benefit study estimates the investment and implementation costs of designing and implementing the integrated approach. Investment is measured by the personnel and related expenses of the project during the second, or coming year of this 2-year project. (The first-year costs are sunk, and no longer relevant.) Implementation involves training technical writers and course developers in using the Bridge, and the licenses and user fees to cover the additional costs of maintaining the networks and the data repositories. The Bridge Project's benefits are the anticipated reduction in the costs of producing future technical manuals and training courses, and possible improvements in shipboard readiness.

The cost savings were estimated by first listing the dozens of tasks (38 for the manual and 80 for the courses) to produce a nominal 500-page technical manual and a

nominal one-content-hour training course. Project personnel estimated the number of staff hours to produce these tasks under both current and Bridge processes, and thus the staff hour savings from using the Bridge. Pay rates are used to convert the staff hour savings to cost savings. The final step is scaling up the results to the sample of yearly production of technical manuals and training courses. Costs and benefits were expressed in 10-year present values calculated using the 2.4% annual discount rate mandated by OMB for 10-year studies.

Two different samples were chosen for analysis, reflecting the different perspectives of OSD and the PMS 420 Program Office. The first analysis recognizes OSD's interest in seeing whether the new software and technical and business processes that comprise the Bridge would lead to positive net benefits to DoD overall—would the benefits cover the costs if implemented by the Navy and other Services as a whole. This analysis is therefore conducted for a substantial number of the Navy's yearly production of technical manuals and training courses: all Hull, Mechanical, and Electrical (HM&E) technical manuals produced by the Naval Ship System Engineering Station (NAVSSES) in Philadelphia (over 45,000 pages annually), and 50% of the Computer-Based Training (CBT) courses delivered by Navy e-Learning (NeL), a part of the Naval Education and Training Command (NETC) (approximately 3,300 content hours annually, in total).

Only 50% of the NeL courses were considered because our analysis of the course titles indicated that only half of the courses trained "hard skills." These are the training courses that deal with equipment and other technical content, and whose cost would therefore be reduced by integrating the production of training courses and technical manuals. (Although courses that train "soft skills" such as leadership and personal advancement would not be directly affected by integration, they might benefit from the information-organizing features of the other Bridge innovations.)

The second analysis, reflecting the Program Office's perspective, is a test case of the results of the aggregate analysis in which the Bridge is applied to a particular system, the LCS's AN/AQS-20A mine hunting sonar. The focus here is on the benefits alone—whether the Bridge would lead to reductions in future cost of producing future technical manuals and training courses for this system. It would not be reasonable to expect the benefits for a single program to cover the full investment and implementation costs of the Bridge, which could lead to savings across DoD.

The first analysis finds that the Bridge would save approximately $87 million in 10 year cost in producing the Navy HM&E manuals and 50% of NeL delivered courses—far more than enough to cover the $8.7 million investment and implementation costs of the program. The second analysis finds that the Bridge would produce substantial savings of almost $306 thousand in producing technical manuals and training courses for the LCS AN/AQS-20A.

Dealing with uncertainty was a major analytical problem. Although much of the analysis used historical data from NAVSSES, NETC and the AN/AQS-20A program, the cost savings are also based on several uncertain inputs relating to the new Bridge process. A sensitivity analysis was conducted of five inputs: 1) the future pay rate for technical writers and course developers who are trained in using the Bridge, 2) the percentage of training hours that would be benefited by the Bridge, 3) the investment cost (to hedge against unanticipated cost of developing the Bridge), 4) the implementation cost (to hedge against problems caused by the cultural changes in Navy programming), and 5) the percentage saving in course developer staff hours from using the Bridge. Considering these changes in combination yielded a full range of net benefits (benefits less costs) for the aggregate case varying from a minimum of $32 million to the Base Case of $78 million ($87 million less $8.7 million) to a maximum of $160 million. Efforts will be taken in the second year of this study to further refine the inputs.

There is every reason to expect that net benefits would be significantly larger if the Bridge were applied to all Navy manuals and courses, and those of the other Services as well.

The final benefit for analysis was the improvement in Fleet readiness. Integrated production of technical manuals and training courses would increase the likelihood of providing the Fleet with up-to-date and consistent information, thus providing some insurance to the Navy's policy of fielding new systems and equipment upgrades only when the appropriate logistics support is available. A parametric analysis indicates that increasing the availability of the electronic, ordnance and HM&E components of a single new DDG 1000 destroyer by a single month would yield a gain in effectiveness the Navy values at $2 million.

Queues in Acquisition

William Wiltschko

Abstract

Acquisition programs have inherent variability in their task durations, which often results in unforecasted completion delay. Using concepts from Lean Production and Lean Product Development, queues that are at the heart of these delays can be made visible and can be managed. Observing queues in acquisition programs can give early warning of project problems. Several techniques can be used to manage queues.

Keywords: Queues, queueing theory, acquisition, product development, lean product development, cost of delay, utilization

Introduction

This paper is intended to be an introduction to a portion of a large subject. Conferences, scores of books, hundreds of papers, and uncounted consultants have been devoted to product development in the public and private sectors. Issues such as configuration management tools, quality of the IMS (Integrated Management Schedule), domain-specific considerations, and people management—all important issues—will not be treated here. This paper focuses on a topic that may not be as well known as other product development topics, but I believe has great potential to better manage acquisition programs.

Queues are generally unrecognized entities in acquisition programs, yet they are valuable information sources and useful handles for controlling them. Lean manufacturing experts have long viewed queues as near-evils to be managed in a production environment, but only relatively recently have they viewed them as either problems or opportunities in product development. There is now a rich literature on lean production (Ohno, 2008) and lean product development (Morgan & Liker, 2006) that relies on insights gained from queueing theory.

Queues are easy to recognize in a production environment—piles of physical product in front of a workstation or machine make them obvious. What is a queue in acquisition, where the thing being manufactured, at least in the early stages, is only information? In this paper, I take the point of view of the Program Manager (PM) or PM leadership and focus on those acquisition phases having a project orientation.

Recognizing a Queue

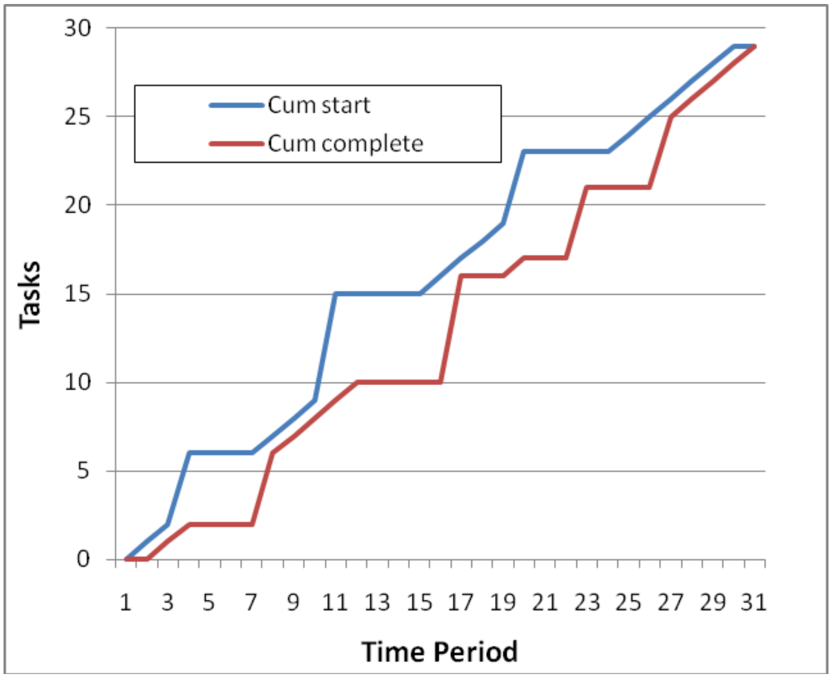


Figure 1 shows a project overview. The top line shows how many tasks have been *started* since the beginning of the project as of the date on the *x* axis. The bottom line shows how many tasks have been *finished* since beginning of the project as of the date on the *x* axis. Thus, in period 13, there are 15 projects that have been started since the beginning and 10 projects that have been finished since the beginning, leaving 5 projects in the queue.

Figure 1. Where is the Queue?

Queues arise whenever there are unfinished tasks. Thus, some amount of queueing is inevitable. In the figure, the only points where the lines meet (where the queue has disappeared) are at the beginning and at the end. However, when the number of unfinished tasks is large, queues are large. In the figure, the gap between cumulative started and cumulative finished tasks is the queue size. Note that both the vertical (quantity) and horizontal gaps (time) grow for increasing queues. The key fact to note is that queue size will increase well before the task completion dates *prove* that the schedule is slipping. Thus, queue size is a leading indicator of schedule slips.

This graph can be created using the program management tool to create a scatter plot of numbered task actual start dates and actual finish dates, sorted by actual start date.

In the graph, there are a few points where the queues are dramatically reduced. This occurs when the cumulative complete line jumps up after going horizontal for some time (approximately periods 8, 17, and 24). These points correspond to authorization points such as milestone decisions. Here the queue arises not only because it takes some time to complete several tasks—in synchrony—but also because the milestone meeting may not occur immediately after the tasks are complete. The milestone decision meeting may be delayed. While teams will not completely stop work while they wait for the milestone decision, the milestone decision may render speculative work irrelevant.

What Makes Queues Large?

The factors that make queues larger are longer task durations, the number of tasks being worked on simultaneously, waiting for completion of other dependent tasks, and waiting for task or project reviews. Going a level deeper, these factors are caused by not breaking down tasks into small-enough chunks, multi-tasking key people (thus spreading them too thin), poor metrics that do not allow queues to be better managed, insufficient parallelization of tasks when staff is adequate to support more simultaneous tasks, and infrequent review meetings that increase team wait time.

A pernicious kind of queue is created by rework. Rework is usually not visible in common project management software. In particularly risky acquisitions, where new science and engineering knowledge are being developed, rework is inevitable, and is sometimes represented as a finite number (i.e., a guess) of iterations of a set of tasks. Other reasons that rework occurs is when it is due to team directions that are either under- or over-specified, when testing is delayed, or when authorization reviews do not take place regularly.

A more fundamental cause of large queues arises from the variable nature of work that dominates a typical acquisition. Task durations can only be approximately estimated, and duration varies widely among different tasks. Variability in both estimated and actual durations produces unexpected, non-obvious task duration (cycle time) increases, which in turn increases queues. Figure 2 illustrates this phenomenon The two curves result from two different values for coefficient of variation. The more variation there is in task duration, the more that cycle time tends to “blow up” with increasing utilization.

This phenomenon is well known to practitioners of lean *production*. Their normal response, as opposed to the response of lean *product development* practitioners is to aggressively reduce variability. This is often not an option in acquisitions that require knowledge work, such as science and engineering. Variability is inherent in knowledge work, so other approaches must be used to make an impact on project cycle time and queues.

How Do You Measure Queues?

Unlike estimated schedule completion, queues are measured with actual data. Their size is the accumulated person-hours actually spent on *started* but *unfinished* tasks. This number can be calculated from most project management software if incurred person-hours are entered into the tool.

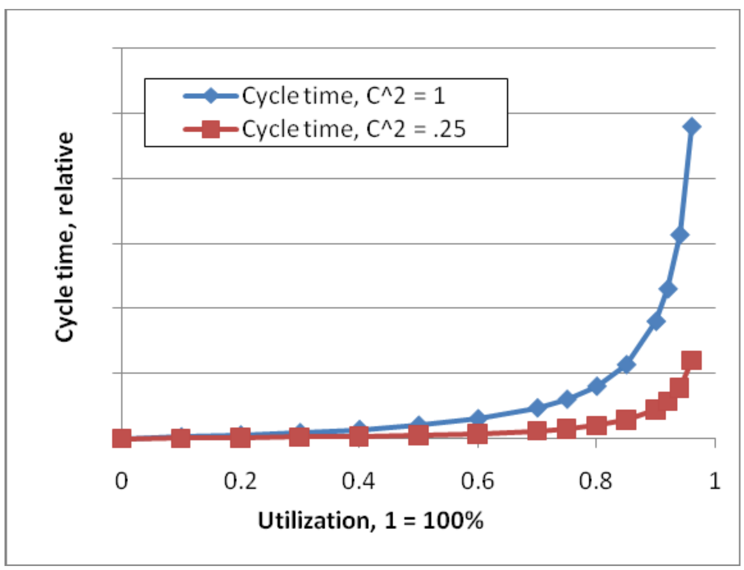


Figure 3. C

Cycle Time versus Utilization

(Hopp & Spearman, 1996)

Task	Start	Actual Finish	Projected Finish	WIP
Program Budget Cycle 2010 (Apr '08-Sept '09)	4/15/2008	9/30/2009	9/30/2009	0
Budget Execution Cycle 2010 (Jul '09 - Nov '10)	6/29/2009		12/1/2010	270
NCCA to review the draft CARD	3/19/2010		6/25/2010	7
NCCA Develop ICE	3/30/2010		7/7/2010	0

Figure 4. Measuring Queues from IMS

Above is a simple example taken from a typical IMS. For simplicity, only two started-but-unfinished tasks are shown, with 270 and 7 workdays invested in the two tasks. Workdays from any additional started-but-unfinished tasks would simply be added to 277. This value would be valid only on the day that this data is recorded. By recording this data every week and graphing it, queue size and trends would become apparent.

However, for very large programs, there may be scores of open tasks, and not very timely accounting for actual hours spent. Lack of timely data entry defeats the purpose of providing early warning, but there is an easier way to providing nearly the same information. Tracking only workdays (without regard to how many people are working on each task) spent on started-but-unfinished tasks provides a good substitute.

Figure 4 is a graph of queue task-periods for the graph shown in Figure 1. In other words, they have been calculated for every period in the project rather than just one period as in Figure 3. Figure 4 shows the queue in period 13 growing above the previous maximum. This is early warning that work may not be completed as scheduled. While the cause may be long duration tasks and not late tasks, the graph provides triggers to ask questions about what is going on. The height of the curve in Figure 4 represents the *area* between the two lines in Figure 1.

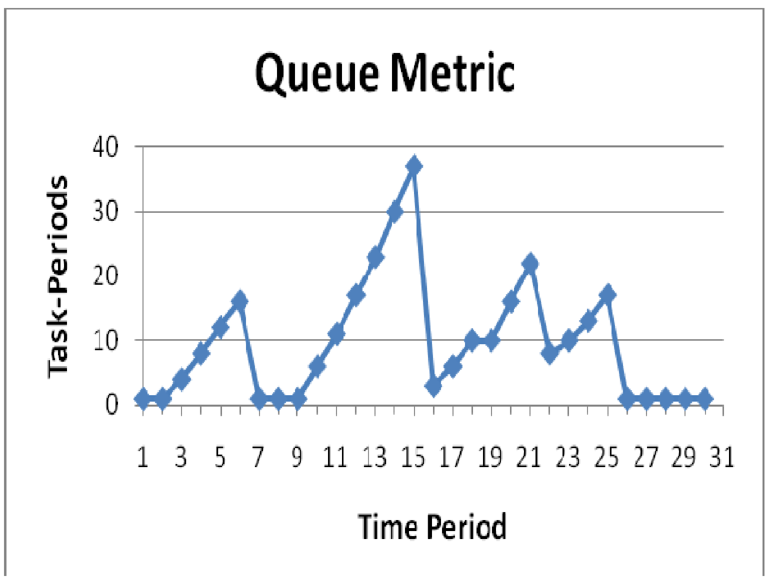


Figure 5. Size of Queue in Task-Periods

The beauty of this metric is that it can be calculated from PMO (Program Management Office) IMS data with little extra work. In other words, no matter the size of the program, this metric is easily calculated and tracked.

How Can We Reduce Queues?

Three general measures can be taken to reduce queues:

- Manage demand,
- Increase capacity, and
- Project management.

Demand can be managed via requirements management. Most requirements' development processes bucket requirements into "must haves" versus "nice to haves." This can be expanded to ranking (possibly by dollarizing) requirements so that when a schedule slip with a given set of requirements looks likely, there is a list of "nice to have" requirements that can be jettisoned in rank order. The key is to *rank order* requirements. The program would then have a requirements relief valve.

Increasing capacity will reduce utilization, thereby reducing queues and cycle time (see Figure 2). Capacity can be increased by staff additions or staff adaptation—that is, intelligently and dynamically allocating staff. Many programs assume that only IMS people do IMS, only acquisition people do acquisition planning, only manpower people do manpower planning, etc. Using people with multiple domain capabilities can help increase utilization and decrease cycle time. They can either be teams of senior people who move from function to function as problems crop up, or teams of junior people who may not need as much domain-specific knowledge to change function and still perform adequately in a reduced role. These teams are sometimes called SWAT teams or tiger teams.

Understanding queues gives the program manager extra tools. First, as mentioned above, queues are a leading indicator of program health. Second, developing expectations of where specific queues are likely to occur makes it possible to prevent them, not just fix them. For example, task parallelization can be concentrated on the potentially longest (riskiest) queues, and efforts can be made to move these potential queues from the critical path. Third, as mentioned in the paragraph above, SWAT-like teams can be constructed with the right skill sets for expected queues. Fourth, with the right economic guidelines, which we will discuss next, the program manager can respond quickly to rapidly developing queue problems. Finally, the program manager and his or her leadership can schedule reviews of the program both internally and externally on a frequent, regular basis (a cadence) so that queues don't build while waiting for a decision.

Quick Response Based on Solid Economic Guidelines

Controlling queues means getting the right resources put on solving real-time design and planning problems. This requires knowing what level of resources is reasonable to apply to the problem. This can only be done well if tradeoff guidelines *based on data* are created at the beginning.

Figure 5 shows these tradeoff guidelines. There are three tradeoff ratios:

1. Between development cost and procurement cost (or cost to field)
2. Between procurement cost and cost of delay
3. Between development cost and cost of delay

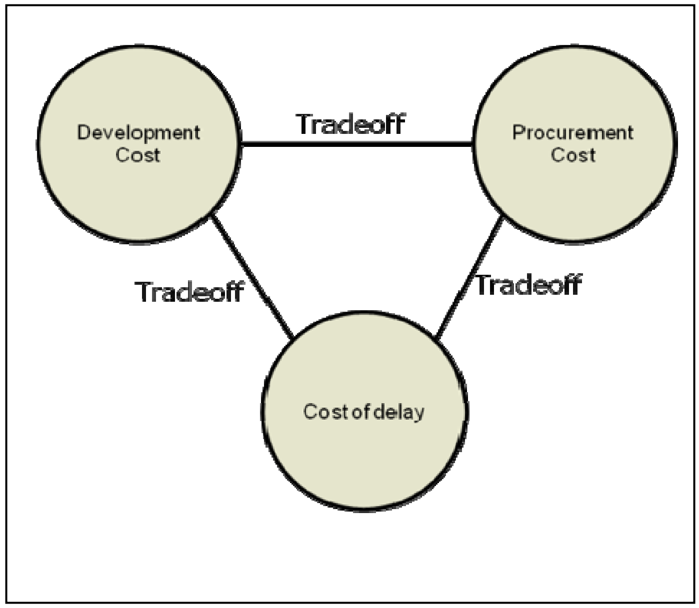


Figure 6. Economic Tradeoffs

The most difficult metric to calculate is the cost of delay. This is not often done in either government or private industry, since the cost of delay has a high subjective component. Nonetheless, an intelligent guess, especially if there is buy-in at every level of leadership, is better than none at all. For, if there is not even a guess, many decisions that may affect queues and cycle time—and thus the program being on schedule—may have to be made above the PMO or after the program slips. Or to put it another way, having upfront guidelines to make these tradeoffs makes it possible to push many decisions down to the PMO's teams where quick response at the most detailed level may help prevent schedule slips.

An example of a tradeoff guideline is, "you are authorized to spend up to $100 to save $200 cost of delay, without asking for permission." One dollar value can be given to the PMO, who may give smaller limits to the teams below depending on the degree of oversight desired. As Reinertsen has pointed out regarding product development (2009), this kind of guideline can become a core part of mission-type orders (Lind, 1985) to the PMO. Figure 6 below shows a notional way to transmit guidelines to the PMO.

Metric	To achieve savings:	Team leads may authorize spending up to:	Functional managers may authorize spending up to:	PM may authorize spending up to:
Development cost		$50	$100	$150
Procurement cost	$1,000			
Cost of Delay	$200			

Figure 7. Sample Economic Guidelines

Summary

Queues in acquisition are good leading indicators of future schedule slips. Queues can be managed by ranking requirements, controlling task starts, staffing adaptively, setting up "SWAT teams" of acquisition experts, parallelizing tasks, reviewing cadences, and establishing guidelines for tradeoffs. A list of references is provided to direct the reader to the latest writing I found on the subject.

References

Hopp, W.J., & Spearman, M.L. (1996). *Factory physics: Foundations of manufacturing management*. Chicago: Irwin.

Lind, W.S. (1985). *Maneuver warfare handbook*. Boulder, CO: Westview Press.

Morgan, J.M., & Liker, J.K. (2006). *The Toyota product development system: Integrating people, process, and technology*. New York: Productivity Press.

Ohno, T., & Bodek, N. (2008). *Toyota production system: Beyond large-scale production*. New York: Productivity Press.

Reinertsen, D.G. (2009). *The principles of product development flow: Second generation lean product development*. Redondo Beach, CA: Celeritas.

Womack, J.P., & Jones, D.T. (2003). *Lean thinking: Banish waste and create wealth in your corporation*. New York: Free Press.

Modeling, Simulation, and Analysis: Enabling Early Acquisition Decisions

Fred Hartman—Fred Hartman, Research Staff Member with the Institute for Defense Analyses, Science and Technology Division, has a broad background in Defense-related management and analysis positions in industry and government. Specializing in modeling and simulation (M&S) applications for training, analysis, and acquisition, Mr. Hartman has over 30 years of experience in providing technical management and oversight for Defense and industry programs. He served from 2003 to 2007 in the Office of the Under Secretary of Defense (Personnel and Readiness) as Director, Training Transformation Joint Assessment and Enabling Capability (JAEC), and as Deputy Director, Readiness and Training Policy and Programs (RTPP). In this capacity, he was study director for the Training Transformation Block Assessments and the Training Capabilities Analysis of Alternatives, a major training systems cost-benefit analysis for future M&S applications in joint training. Mr. Hartman currently is supporting the DoD M&S Coordination Office to align corporate and crosscutting capabilities for future development.

Frederick E. Hartman
Institute for Defense Analyses
4850 Mark Center Drive
Alexandria, VA 22311
Contact Information: 703-578-2776 (W), 703-473-3382 (C), 202-298-7166 (H),
Fax: 703-931-7792

Abstract

The use of systems engineering early in the acquisition cycle is being advocated for programs as a means to add analytic rigor prior to Milestone A. Modeling and simulation (M&S) coupled with early requirements and effectiveness analyses can shape programs and test alternatives prior to costly program commitments. Conceptual modeling and early cost effectiveness analyses are key to revitalizing development planning and early systems engineering, which will enable more-informed decisions by acquisition leadership. Early systems prototyping, coupled with continuous program support and assessment, will enable better acquisition decisions through the series of milestone decisions.

Keywords: modeling, simulation, analysis, early systems engineering, prototyping

Research Issue

In the current DDR&E organization, the Systems Engineering Directorate includes modeling and simulation as part of the Systems Analysis Division. This new organization places powerful assessment capabilities and access to modeling and simulation for systems engineering early in the acquisition program lifecycle.

Background

During the last several decades, we have witnessed incredible progress improving underlying modeling and simulation (M&S) technologies. Dr. Anita Jones (1988) led a Defense Science Board Study (DSB) published in 1988 that recommended improving our simulations to allow for more home station training of commanders and staffs, facilitate the sharing of our simulation data and arrive at more simulation-based training with less developmental redundancy. The 1988 DSB study got to the heart of many of the fundamental issues in M&S that we are still working to improve today—some two decades later. A few years after the DBS study, the Defense Modeling and Simulation Office (DMSO) was formed, and Dr. Jones took over as the Director, Defense Research and Engineering (DDR&E). During the 1990s, a progressive series of architectures and standards were developed to improve the DoD's ability to form distributed, interoperable simulation environments with reusable scenario data and content. Each of the three M&S communities—Analysis, Training, and Acquisition—kicked off major joint programs during that time. The Acquisition Community formed the Joint Modeling and Simulation System (JMASS) to expand on simulation-based acquisition (SBA)[7] and to leverage simulation capabilities across the acquisition lifecycle. The main thrust of both of these programs was to incorporate models and simulations as very integral components in each phase of the Acquisition process. Although there was not widespread follow-up support, the concepts are still relevant today.

The Under Secretary of Defense for Acquisition, Technology, and Logistics (USD(AT&L)) is the single focal point for the coordination of all matters related to DoD M&S (USD(AT&L), 2007). The *DoDD 5000.59* provides for the management of M&S via an executive-level DoD M&S Steering Committee, comprised of key agencies in the OSD, the Joint Staff, Services and Combatant Commands. Chair of the M&S SC is delegated to the DDR&E. With the publication of the M&S Steering Committee's *Strategic Vision for Modeling and Simulation* (2007) and *The 2008 Modeling and Simulation Corporate and Crosscutting Business Plan* (DoD, 2009), the DoD M&S community has moved forward on a series of high-level tasks (HLTs) aimed at improving the M&S tools, data, functional representations and enterprise services across the Department. The HLTs are consistent with the five M&S goals contained in the *Strategic Vision*.

For example, the instantiation of Live Virtual and Constructive (LVC) environments for training, experimentation and testing applications show that we can today achieve many of the interoperability goals discussed in the late 1980s. The training community can establish persistent networks dedicated to distributed simulations to link together nodes that are located all over the globe. The technology that has been assembled at the US Joint Forces Command (JFCOM) in their LVC training environment is supported by Joint Training and Experimentation Network (JTEN) and is used by three of our Communities enabled by M&S as well as the Services. The JFCOM training environment has also served the Information Operation (IO) Range to examine cyber network issues for the future. The success of these LVC environments today provides us only a glimpse of the opportunities likely available to enjoy in the net-enabled world of the future. Another area that has seen a tremendous increase in capability is the use and reuse of data for common scenarios and other wide-ranging applications. A number of existing programs in the Services as well as for joint applications have collaborated to solve many of the hard issues that have precluded meaningful data reuse over the last decade. We have learned over time that in M&S, both the user needs and the enabling technologies are continuously evolving. Information technology now supports an environment that allows the creation of more realistic, more capable, and more powerful simulation tools. Significant reductions in program

[7] Overview from Chapter 11.13, *Defense Acquisition Guidebook*: SBA is the robust and interactive use of M&S throughout the product lifecycle. The program manager should employ SBA and M&S during system design, test and evaluation, and modification and upgrade. The program manager should collaborate with operational users and consider industry inputs during SBA/M&S program planning.

development times, lifecycle costs, and improved systems performance can be realized through use of M&S in acquisition.

Acquisition Reform

It is widely perceived that there are problems with the DoD acquisition process. Several of the common complaints from the user communities are as follows:

- Too slow,
- Requires significant labor investments just to satisfy and document the process,
- Capabilities frequently reach concept decision and enter Milestone A without sufficient concept refinement and contact with the users, and
- Too many requests from senior management for more rigorous analysis to drive decisions for program start up and/or no go early in the process.

The acquisition reform goals and policies of the Obama Administration outline actions that impact government procurement, acquisition programs and contractors in a wide variety of areas. The convergence of new administration priorities, burgeoning costs, and outdated procurement processes has prompted major contracting and policy initiatives designed to:

- Develop more agile acquisition processes to increase the speed of technology deployment,
- Increase transparency of the acquisition process,
- Institute stricter risk and performance parameters, and
- Reduce costs through cuts in contractor spending and use of “high-risk” contract types.

This paper proposes that M&S can assist the USD(AT&L) in meeting the new administrations’ acquisition reform initiatives. Key to reform is the ability to both compress timelines and add more analytic rigor to the acquisition process through the use of modeling and simulation. Especially in the early stages of the acquisition process, the use of M&S for rapid prototyping and to support the analyses stages prior to Milestone A is useful to influence the early concepts, design and recommendations for major systems procurements. Although extremely important in the early stages of acquisition, the use of M&S applications at every stage of the process provides efficiencies and improvements in a wide variety of uses from requirements to technical aspects of design and development to sustainment of a given system. M&S is more than a single tool or set of tools used at critical points in the process; it is rather a way of doing business that impacts every aspect of a system’s lifecycle. In July 2009, the DDR&E introduced four Imperatives to focus the organization in support of the immediate and future needs of the Department of Defense: 1) accelerate delivery of technical capabilities to win the current fight; 2) prepare for an uncertain future; 3) reduce the cost, acquisition time and risk of our major defense acquisition programs; and 4) develop world class science, technology, engineering, and mathematics capabilities for the DoD and the nation. The use of M&S is a clearly an enabler to achieve Imperative 3 above.

Simulation-Based Acquisition (SBA)

The concepts of the SBA program formed a decade ago are still viable—but largely unachieved today. The SBA vision encompasses an acquisition process in which the DoD and industry partners are enabled by the robust, collaborative use of simulation technology integrated across all acquisition phases and programs. The goals of SBA are very consistent with the current administration's acquisition reform policy initiatives:

- Reduce time, resources, and risk associated with the entire acquisition process;
- Increase the quality, military worth and supportability of fielded systems;
- Reduce total ownership costs throughout the system lifecycle; and
- Enable integrated product and process development across the entire acquisition lifecycle.

In keeping with the SBA vision and goals, the Department can provide a systems engineering environment that emphasizes M&S as a primary analysis tool and fosters the use and reuse of data and M&S content across programs and phases. It is envisioned that use of models can refine the needs and provide the underpinning for more rigorous analyses prior to Milestone A, while transitioning critical content to guide systems design and later development and production processes. As far back as 1997, Dr. Pat Sanders, the then-Director, Test, System Engineering and Evaluation, OUSD (A&T), was writing magazine articles on SBA as an effective, affordable mechanism for fielding complex technologies. Even almost 13 years ago, it was believed that the extensive use of constructive models for system-of-systems evaluations would provide significant benefits—particularly as they would enhance virtual prototypes that could be operated on future synthetic battlefields. One can believe the future as regards these simulation environments is very close at hand today.

Early Prototyping

From the early requirements and conceptualization stages, the use of M&S and in particular system prototyping provides a powerful analytic capability to meet user needs. It has been argued that prototypes are platforms for productive participation, as well as for perfecting products and performance (Schrage, 2010). The power of producing systems prototypes early in the process serves as a way to iterate with the end user to arrive at better systems and solutions for the operational needs. The more obvious use of prototypes is to guide the engineering analysis in the development planning stage of the acquisition. Any number of firms can be found through the internet proposing services to industry in the area of model making and prototyping. Many of these firms are highly successful, providing rapid prototyping services that encompass proof of concept and proof of design with functional working simulations and models. The use of prototyping can encompass constructive simulations, virtual environments or physical mock ups of the end system or product. With the use of such tools as 3D visualization, one can progress to "model making" to influence the construct of actual 3D models. The area of rapid prototyping uses state-of-the-art CAD/CAM (computer-aided design and computer-aided machining or modeling) techniques. Significant advances in the area of M&S make it now more important than ever that we incorporate oversight policies and directives to include contracting language that requires the use of simulations, models and prototypes in all phases of the acquisition process.

Research Result

M&S can provide a combination of live, virtual and constructive acquisition environments to impact policies and acquisition decisions early in program development and throughout the acquisition process to facilitate efficiencies and avoid costly program errors.

References

Jones, A. (1988, May). *Defense Science Board (DSB) task force on computer applications to training and wargaming*.

M&S Steering Committee. (2007, August). *Strategic vision for modeling and simulation*.

Sanders, P. (1997, September-October). Simulation based acquisition: An effective, affordable mechanism for fielding complex technologies. *Program Manager (PM)*, 26(5), 72-77.

Schrage, M. (2010, February 1). Keynote address delivered at the M&S Leadership Summit, Sloan School of Management, MIT.

DoD. (2009, February 23). *The 2008 modeling and simulation corporate and crosscutting business plan*. Washington, DC: Author.

USD(AT&L). (2007, August 8). *DoD modeling and simulation management* (DoD Directive 5000.59). Washington, DC: Author.

THIS PAGE INTENTIONALLY LEFT BLANK

Panel #4 – Program Management and Contracting in a Performance Based Environment

Wednesday, May 12, 2010	
11:15 a.m. – 12:45 p.m.	**Chair: Timothy J. Troske,** Division Technical Director, Port Hueneme Division, Naval Surface Warfare Center
	A Model for Determining Optimal Governance Structure in DoD Acquisition Projects in a Performance-Based Environment
	David Berkowitz, James Simpson, Tom Kallam, Joshua Jones, and,University of Alabama in Huntsville; and Gregory Gundlach, University of North Florida
	A Simulation Model for Setting Terms for Performance Based Contracts
	Betty Jester, Marie Bussiere, James Ferguson, Naval Undersea Warfare Center, Newport and Manbir Sodhi, University of Rhode Island
	Contractor Incentives for Success in Implementing Performance-Based Logistics: A Progress Report
	Jatinder (Jeet) N.D. Gupta, Michael C. Eagan, Joshua Jones and James Platt, University of Alabama in Huntsville

A Model for Determining Optimal Governance Structure in DoD Acquisition Projects in a Performance-Based Environment

David Berkowitz

Introduction

Product acquisition and sustainment have traditionally been separate and not necessarily equal concerns in defense acquisition. To reconcile this deficiency, the 2001 *Quadrennial Defense Review (QDR)* proposed a modernization of the defense acquisition process that resulted in the adoption of Performance-Based Logistics (PBL), which integrates a performance-based environment for both acquisition and sustainment. The basic tenets of PBL suggest that the governance structure used must address the potential long-term nature of the relationship between the government and suppliers by integrating more collaboration and adaptability into the contractual mechanism. Knowing this, the ultimate challenge for a contractor is being able to understand the relationship they have with the government and be able to evaluate whether the governance structure chosen permits, inhibits, or prohibits the government and contractor from achieving desired outcomes.

The purpose of this paper is to present a conceptual model that describes the conditions under which defense acquisitions should be structured as either being more short-term, transactional exchanges; long-term relational exchanges; or plural form (which recognizes the complementary nature of contracts and cooperative norms). Using this conceptual model coupled with the logic provided by Transaction Cost Theory (TCE), we should be able to better explain whether the government-contractor relationship has a significant impact on the outcome of the contract. For those contracts that fail as a result of endogenous conditions, we realign those programs with alternative contract types and alternative governance structures that are more suitable for the conditions of those programs. We conclude this study with a discussion of how managers should match contract type with optimal governance structure and a preliminary empirical examination of the conceptual model.

Background and Concepts

Performance-Based Contracts

Geary and Vitasek (2008) argue that longer term contracts encourage long-term investments to improve product or process inefficiencies. Their logic is that long-term (greater than one year) contracts justify higher up-front investments on the part of the contractor, while short-term (one year or less) contracts generally discourage up-front investment on the part of the contractor and are therefore less effective at obtaining a higher degree of performance. Keeping this in mind, we recognize that because the preferred PBL contracting approach is long-term contracts (USD/ATL Policy Memo, 2005), the DoD is not only choosing to invest in the acquisition of a technology or system, it is also investing in a relationship with that contractor.

Formal Contracts

There are different schools of thought concerning the impact of formal contracts on the relationship between the parties involved. Ghoshal and Moran (1996) and Fehr and Gachter (2000) argue that formal contracts may signal distrust which could encourage one or all of the parties to exhibit opportunistic behavior. Poppo and Zenger (2002) argue that when relational governance exists, formal contracts are an unnecessary expense and could potentially be counter-productive. Other scholars seem to think that because transactional uncertainty is inherent in long-term contracts, having formal agreements are necessary for combating market dynamism (Aldrich, 1979; Child, 1972), which is a result of evolving technology, shifting prices, or variance in product availability (Cannon et al., 2000).

Cooperative Norms

We define the term cooperative norms as being the relational norms that exist outside of the formal contract. In other words, if a formal contract establishes a set of legal conditions, in theory, the relational norms that exist between the parties involved are the means by which those conditions are satisfied. Williamson (1993) argues that contractual incompleteness notwithstanding, an ex post maladaptation problem will not arise if (1) the parties promise to disclose all relevant information and behave cooperatively during contract execution and renewals, and (2) these promises are self-enforcing. We view cooperative norms as being complementary to formal contracts, which agrees with Gundlach, 1999; Gulati, 1995; Ring and Van De Ven, 1994; Allen and Lueck, 1992.

Transaction Cost Theory

When it comes to understanding how managers construct governance arrangements, transaction cost theory has become a common supposition for explaining the rationale behind these arrangements. Understanding the impact of transaction costs will allow contractors in the defense industry to better articulate and account for the hazards associated with multi-party, multi-year procurement and sustainment contracts.

The theory of Transaction Cost Economics (TCE) is centered on two basic principles: (1) human beings are *bounded rationally*, and (2), as a result of being rationally bound, will always choose to further their own self-interest (i.e., *opportunism*) (Williamson, 1985). Within the context of TCE, scholars define three categories of exchange hazards that require contractual safeguards: (1) asset specificity, (2) difficulty of measurement, and (3) uncertainty. *Asset specificity* arises as sourcing relationships require significant relationship-specific investments in physical and/or human assets (Poppo & Zenger, 2002). *Difficulty of measurement* arises when the rewards given to a contractor cannot be objectively linked to a set of performance parameters. Lastly, *uncertainty* arises because of one's inability to know and account for all hazards that occur as a result of seen and/or unforeseen changes.

Several variables give rise to transaction cost issues in the defense industry. Some of the most commonly recognized are the defense budget cycle, rapidly evolving technology, a bimodal distribution in the age of government employees, and a giant gap between first and third-tier suppliers (Chao, 2005). Although the degree of significance may vary greatly amid these and other variables, we assume that their collective impact on the government-contractor relationship is significant. As a result of their collective significance, we believe that both the government and the contractor construct contractual agreements that: (a) reduce the level of risk assumed by the contractor, and (b) provide a product or service that meets the government's needs at a reasonable price.

Governance. Over the past 30-40 years, several scholars have contended that interorganizational exchanges are driven by variables outside of the formal contract. Governance emerges from the values agreed-upon processes found in social relationships (Macneil, 1978; 1980; Noordewier et al., 1990; Heide & John, 1992; Poppo & Zenger, 2002). Tubig and Abetti (1990) found that both exogenous (external) and endogenous (internal) variables influence contractual performance. Their research found that endogenous variables such as type of R&D, type of solicitation, and type of contract, all had an effect on contractual performance.

When we think about specific types of governance structures we see governance as existing along a spectrum that moves from transactional to relational (see Conceptual Model). Transactional governance implies that there are fewer *hazards to exchange* (i.e., environmental uncertainty, transaction-specific investments, or difficulty in measurement); therefore, continual interaction between the government and the contractor may be unnecessary. Relational governance, on the other hand, implies that there are *greater* hazards to exchange; therefore, continual interaction would be needed between the government and the contractor to ensure that both players are acting in ways that reflect their mutual interests and not in ways that exhibit opportunism.

We hypothesize that for a large majority of Major Defense Acquisition Programs (MDAPs), contractual success is permitted when there is a strong mix of both legal and social conventions. This plural form governance structure, however, does have both pros and cons. According to Dyer (1996) and Dyer and Singh (1998), social governance may lead to a reduction in transaction costs when compared to formal contracts. Gundlach (2000), however, takes the view that the institution of social norms requires a history of interaction and reinforcement, whereas the absence of such a history could lead to conflict, distrust, and opportunism.

Conceptual Model

In government contracting, formal contracts serve as the primary governing mechanism for acquiring and supplying organizations. Yet studies consistently report that performance is typically higher among organizations that use non-legal principles to govern the relationship among the buyers and suppliers. Our conceptual model aligns the alternative governance structures derived from transaction cost economics, normative structures derived from relational exchange theory, and plural forms derived from the joining of these two frameworks to explain the three possible mechanisms for governing DoD contractual relationships. The model also describes the hazards of exchange and moderating variables that suggest a shift from more traditional transactional exchanges to more relational exchanges. Finally, the model provides a framework for aligning alternative contract mechanisms with the optimal governance structures and accessing the impact of alternative contacting arrangements on the DoD's perception of performance.

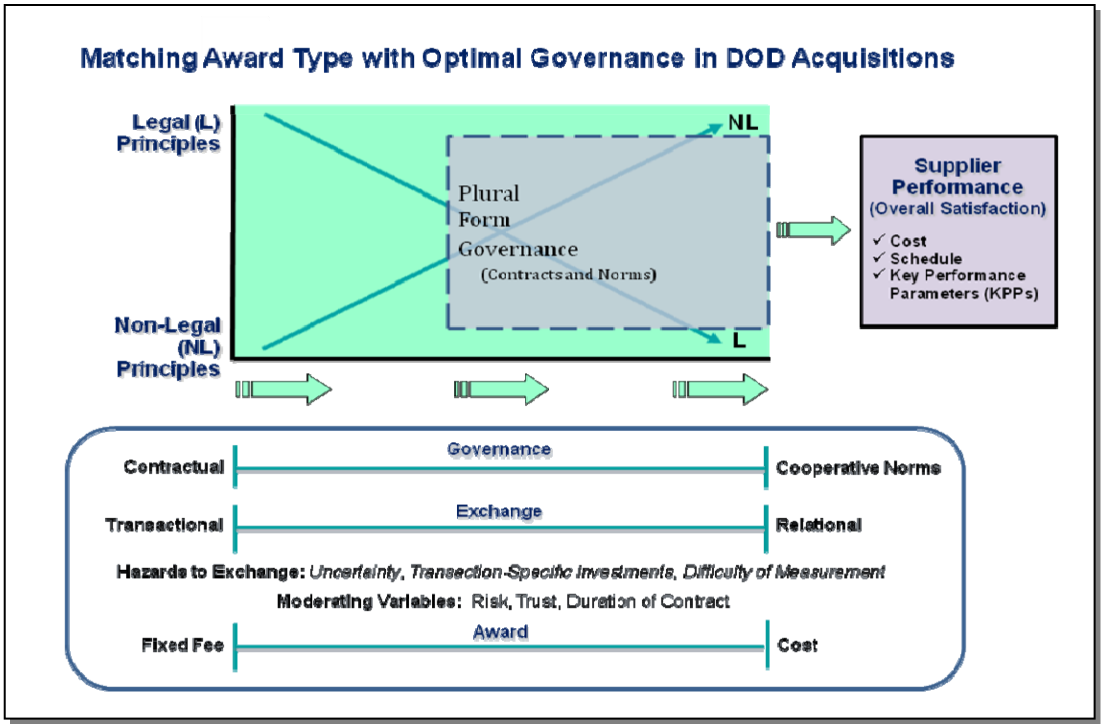


Figure 1. Matching Award Type with Optimal Governance in DoD Acquisitions

Type of Contract

FAR 16.101(b) states the following: "contract types are grouped into two broad categories: fixed-price contracts and cost-reimbursement contracts." On one end of the contractual spectrum you have the *Firm-Fixed-Price* (FFP) contract where there is no mitigation of the cost risk associated with producing an end item by the government; therefore, the contractor assumes all of the cost risk associated with that end item. On the other end of that spectrum you have the *Cost-Plus-Award-Fee* (CPAF) contract where objective incentive targets are not feasible for critical aspects of performance; therefore, the government's objectives are more broad, giving the contractor flexibility to interpret how to achieve those objectives. As a result of those broad objectives, the government chooses to share in the risk associated with creating that end item.

The contractual spectrum reveals certain proclivities about the types of relationships one would find given certain types of contracts. As an example for major weapon systems (MWS), under an FFP contract, the government is not investing in any of the current developmental risk associated with that product; therefore, the type of relationship the government has with the contractor may not be a critical issue. On the other hand, under a CPAF contract, the government is investing in the development of a product that may be currently immature, or perhaps, does not even exist; therefore, we assume that the success of that contract will be dependent upon the type of relationship that the government has with that contractor.

Preliminary Analysis

Using contract data housed by the Federal Procurement Data System (FPDS) coupled with performance data found in the Selected Acquisition Reports (SAR) housed by the Defense Acquisition Management Information Retrieval (DAMIR) system, we evaluated 16 Major Defense Acquisition Programs (MDAPs) that spanned across the different service branches. Three programs were selected from the US Army, 3 from the US Air Force, 5 from the US Navy, and 5 programs were classified as Joint Service Products (see Appendix A). The programs selected were based upon a predetermined set of criteria that allowed the analysis done to be well-balanced.

Matching Contract Type with the Appropriate Governance Structure

When one considers the type of contractual mechanism and governing structure that should be applied to a particular program or project, it is important to first evaluate the types of variables that would, or could potentially, have the most significant impact on the overall success of the project. In the defense industry, some of the variables to consider would be relational history (contractor-government and/or contractor-contractor), duration of the contract, level of investment risk, wartime verses peacetime, state of the economy, rate of technological change for the item being procured, and complexity of development.

As a contractor, it is vital to understand the role the firm plays in the defense industry. This will allow the firm to better predict which variables could have the greatest impact on the firm's ability to achieve desired outcomes. Once those variables have been identified and a suitable governance structure has been selected for dealing with those potential hazards, ceteris paribus, there should be greater degrees of contractual success.

A Simulation Model for Setting Terms for Performance Based Contract Terms

Betty Jester—Betty C. Jester is employed by the Naval Undersea Warfare Center (NUWC) Division, Newport. She has extensive experience in torpedo systems, having worked on torpedo programs since graduating from the University of Rhode Island with a BS in Electronic Computer Engineering. Ms. Jester also earned an MBA from Salve Regina University. She derives her technical expertise from working on all aspects of torpedo life cycle development, from requirements definition through development and test, production and support, and maintenance for both lightweight and heavyweight torpedoes. Ms. Jester is currently the Technical Project Manager for Torpedo Production.

Betty C. Jester
1176 Howell Street
Code 8191, Building 990/3
Newport, Rhode Island 02841
Phone: 401-832-2303
Fax: 401-832-6859
Email: betty.jester@navy.mil

James C. Ferguson—James C. Ferguson is employed as an engineer by the Naval Undersea Warfare Center (NUWC) Division, Newport in the Production, Logistics, and Quality Assurance branch. He was hired in June 2009. He received a BS in Industrial and Systems Engineering from the University of Rhode Island in May 2009. He is currently pursuing an MS in Systems Engineering at the University of Rhode Island.

James C. Ferguson
1176 Howell Street
Code 8113, Building 148/1
Newport, Rhode Island 02841
Phone: 401-832-4809
Fax: 401-832-4194
Email: james.c.ferguson1@navy.mil

Marie E. Bussiere—Marie E. Bussiere is employed by the Naval Undersea Warfare Center (NUWC) Division, Newport. She is a graduate of the University of Rhode Island, where she earned a BS in Electrical Engineering. Ms. Bussiere also earned an MBA from Salve Regina University and received a diploma from the Naval War College for coursework in the areas of Strategy and Policy, National Security Decision Making, and Joint Maritime Operations. Ms. Bussiere is currently the Head of the Undersea Weapons Acquisition and Life Cycle Engineering Division in NUWC Division Newport's Torpedo Systems Department. Ms. Bussiere is responsible for Torpedo Test Equipment; Propulsion and Mechanical Systems; Production, Logistics, and Quality Assurance; and Fleet Engineering Support.

Marie E. Bussiere
1176 Howell Street
Code 811, Building 148/1
Newport, Rhode Island 02841
Phone: 401-832-1420
Fax: 401-832-4194
Email: marie.bussiere@navy.mil

Manbir Sodhi—Dr. Manbir Sodhi is a Professor of Systems and Industrial Engineering in the Department of Mechanical, Industrial and Systems Engineering at the University of Rhode Island. He obtained his graduate degrees from the University of Arizona, and has taught courses in Systems Design, Systems Simulation, Deterministic and Stochastic Optimization, etc. In addition to consulting for several companies, he has also worked as a visiting scientist at the Naval Undersea Warfare Center (NUWC) Division, Newport and at the NATO Undersea Research Center in La Spezia, Italy. His recent work has appeared in professional journals that include the Journal of Scheduling, International Journal of Production Research, and IIE Transactions. He is currently exploring decision models supporting supply chain planning in defense operations and is developing tools and concepts of operations for the use of unmanned undersea vehicles (UUVs) for a variety of search operations.

Dr. Manbir Sodhi
2 East Alumni Drive
Industrial and Systems Engineering
University of Rhode Island
Kingston, RI 02881
Phone: 401-874-5189
Fax: 401-874-5540
Email: sodhi@egr.uri.edu

Abstract

This paper sets forth a model-based approach for selecting terms applicable to a heavyweight torpedoes (HWT) Performance Based Logistics (PBL) contract capable of addressing both near- and long-term support considerations over the torpedo life cycle. Several performance measures commonly used in PBL contracts are described, and a model is presented that is based on an "availability" metric. This metric is calculated using the number of times a required part is not available at field maintenance sites. The metric is computed at the Functional Item Replacement (FIR) or modular level of replacements. The contractor is made responsible for maintaining an inventory of parts on the shelf at the maintenance locations. This is referred to as the Coordinated Shipboard Allowance List (COSAL), which is not to exceed a negotiated maximum. Terms of the contract are not specified with regard to lead times such as logistical delays and manufacturing and restocking lags. These times are assumed to be under the contractor's control as are production quantities, quality, and responsiveness. A newsvendor approach for determining optimal shelf inventory levels is first developed. An augmented model is evaluated using a simulation to determine the performance sensitivity to changes in product quality, demand rates, and various supply chain related lead times. Practical collateral issues such as obsolescence, reliability, and cost are also discussed. This concept is being evaluated as a possible go-forward supportability strategy for the MK48 Common Broadband Advanced Sonar System (CBASS) Torpedo.

Keywords: Supportability, Operational Availability, Performance Based Logistics, and Torpedo Enterprise.

Introduction

The history of modern HWT production and the procurement of associated spares can be divided into four major phases starting in the early 1970s. Each phase was self sustaining for that period; however, with a typical torpedo life cycle of 25-40 years and a philosophy of upgrading existing inventories versus funding the production of new all-up-round (AUR) torpedoes, new approaches are needed to address the spares question. Current factors that combine to challenge those responsible for the continued maintenance of the entire torpedo inventory include limited yearly upgrade kit production quantities, implementation of torpedo acquisition reform requirements (starting in 1995), and the need to maintain a high state of readiness across the entire inventory at minimal cost.

The first modern US HWT, the MK 48 Mod 1, entered production in 1972 after an intensive “shootout” between competitors. This first HWT production contract was a sole source, high production quantity effort. It employed a fully documented “build to print” data package, which consisted of hundreds of military specifications and standards, as well as source/specification control drawings and detailed weapon specification packages. This type of production lasted for over 14 years. Thousands of torpedoes were produced during this period and spares were easy to produce concurrently with production. Since there was a well-documented data package, the Navy Supply System obtained all the spare assemblies and parts needed. This was a technically low risk approach since having proven product disclosure documentation essentially eliminated the risk. During this time period, torpedoes were not only produced but several upgrades were implemented and the configuration advanced to a MK 48 Mod 4 version.

As the enemy threat changed during the height of the Cold War with the emergence of quieter, faster, and deeper-diving nuclear submarines in the Soviet fleet, the US Navy initiated development of a more advanced and capable HWT. The areas of greatest technology improvement included the lowering of torpedo self-noise and the use of ruggedized, embedded, digital micro-processors. The latter capability made it possible for digitally controlled torpedoes to be upgraded with new software as threats and countermeasures evolved. The U.S. Navy initiated an advanced torpedo acquisition program to capitalize on these improvements and to counter quiet coated threat submarines capable of employing sophisticated acoustic countermeasures. The MK 48 Mod 5 Advanced Capability (ADCAP) submarine-launched HWT for anti-submarine warfare (ASW) and anti-surface warfare (ASUW) applications was the follow-on to the older MK 48 Mod 3/4. The MK 48 Mod 5 ADCAP entered pilot production in 1986. Production was at the AUR level and evolved into a series of dual-source competitive production contracts. This approach sustained the selected contractors by issuing various production quantities to each until 1992. At this time, a winner-take-all production contract was implemented due to reduced quantity requirements. During this time, torpedoes were still procured using a build-to-print fully disclosed documentation package in competitive contracts. As system requirements evolved, new torpedo variants were procured. Again, spares were readily available via competition and could be procured by the US Navy Supply system using fully documented disclosure packages at low risk. The ability to procure spares by the US Navy Supply system provided significant savings to the program’s logistics/acquisition disciplines. Separate funding sources ensured that sufficient spares would be available as the need arose. In the event torpedo spares were not transitioned into the US Navy Supply system due to a lack of adequate re-procurement documentation, the spares were procured using program funds. Therefore, it was important that the spares be well documented and transitioned into the supply system as soon as possible.

Starting in 1986 with the Packard Commission Report (1986), “A quest for excellence” and continuing into 2002, a number of acquisition reform initiatives were issued that changed the way the US Navy and other organizations acquired new systems. Perry (1994) started to shift the focus of the acquisition world from processes to outcomes. Since that time there has been a wholesale embracing of Acquisition Reform (AR) initiatives and cancellation of military specifications and standards. The US Navy torpedo program embraced the AR initiative and in 1995 was one of the first to issue a contract under AR guidance. The Torpedo MK 48 Modification Program low rate initial production (LRIP) Contract N00024-95-6190 eliminated the build-to-print technical disclosure package of the

Guidance and Control (G&C) Section and reduced the use of military specifications and standards to five. Detailed weapon specifications were also removed. The major thrust was to replace the proscriptive build-to-print and military specification requirements in the ADCAP production technical data package (TDP) with performance specifications and appropriate commercial specifications. The ADCAP propulsion section remained build-to-print because of its largely mechanical (versus electrical) design, maintenance/replacement complexities, and the effort required to validate/qualify any change in the design. The supply system retains support of the afterbody to this day, but it became the program's responsibility to support the forebody.

It was during this phase of torpedo production that modification kits instead of AURs were procured. The combination of AR requirements, kit procurement, and hardware complexities began to have an impact on forebody spares availability, as well as how spares could be procured. The HWT production contracts had transitioned from a technical "risk avoidance" construct based on the use of a proven detailed TDP that could be built by many qualified vendors to a "risk management" construct based on high level performance specifications. This transition requires vigilant management to avoid problems. Under AR the Navy cannot tell the vendor how to build the item being procured; it can only define the item's performance and interface requirements. Since each vendor has the latitude to build the end item in a different manner, the risk of compatibility within the system as well as across systems became more complex and presents a number of challenges related to production and logistics.

The Supply System Construct

Supply support for the HWT program has been provided by the Navy Supply

(NAVSUP) system. The Navy Inventory Control Point (NAVICP) manages supply support for what is referred to as Depot Level Repairables (DLR) (i.e., unique torpedo items) and the Defense Logistics Agency (DLA) manages supply support for consumable items. For the HWT, NAVICP (Mechanicsburg, PA) is the Program Support Inventory Control Point (PSICP). They receive the TDP for the torpedo-unique hardware from the MK48 In-Service Engineering Agent and initiate the provisioning process for the required items. The PSICP assigns National Stock Numbers (NSN) to the items and updates the COSALs for the various Intermediate Maintenance Activities (IMA). This supply system construct is depicted in Figure 1.

Figure 1. Torpedo Supply Chain Construct

Supply system stocking levels are forecasted based on past demand. A rolling average for the demand from the past eight quarters is used to trigger procurement or repairs. Procurement lead times are adversely impacted by diminishing manufacturing sources, obsolescence, rejected deliverables, and contract defaults. At times, the program must directly compensate for these deficiencies in availability by making their own procurements. Demand for items filled from outside the supply system (i.e., procurements made by the program) is provided to the PSICP via an unfunded "Demand Requisition Only" document, so this data may be factored into overall demand for the item. Attempts have been made to utilize alternative methods to forecast future demand such as "anticipated workload." These types of forecasts can be submitted using "Special Program Requirements" (SPR) to NAVICP and "Demand Data Exchange" (DDE) to DLA. It has proven difficult to identify long term workload requirements and fluctuations, and as a result the program has had limited success using these methodologies. Another attempt to compensate for extensive procurement lead times was the establishment of a program-funded Centralized Logistics Support (CLS) in 2005. The idea was to improve parts availability utilizing central procurement and management for all IMAs. This organization's charter was to overcome shortages and improve availability at the IMAs. CLS was disbanded in 2009 as it was deemed too costly. In parallel with CLS efforts, the enterprise attempted various methods to contract with the prime HWT OEM to provide total commercial supply support responsibility for the HWT program without success; the Request for Proposal (RFP) was never issued.

For support of the major FIR hardware items, which are still being produced (CBASS kits), there exists Contract Line Item Numbers (CLIN) on the production contract to buy spare FIRs and repair FIRs. At this time, the quantity of spares procured is based on known failure rates (calculated by the Government) and limited by the available spares budget. The repair CLINs have a small amount of funding available for a "pay-as-you-go," best effort type arrangement with very few requirements and no contractual obligation. The contractor is not responsible to repair the item if he encounters obsolescence issues. As a result, availability suffers as spares are consumed and/or when failure rates exceed anticipated levels.

As the follow-on production contract was being established, a supportability CLIN was proposed, which would have implemented a PBL-like methodology that established a FIR availability requirement at the IMA as the contractor's responsibility. The contractor would negotiate a firm fixed price to provide a defined percentage of availability for the FIRs he produces over a given performance period. This CLIN was not added to the RFP due to the perception that it was not affordable, and that it might limit competition. As a result, the enterprise decided to continue with the established methodology of procuring spares on the production contract in conjunction with repair CLINs.

Several recent papers have highlighted the reluctance of Program Managers to implement the PBL construct within the DoD. In Fowler (2009), a comparison is made between performance-based logistics (also called performance-based life cycle product support) and the fictional superhero Batman. Like Batman, PBL has received a poor reputation because of the unconventional techniques it employs. PBL is usually accused by critics of "contracting out" logistical support through the use of a Product-Support Integrator

(PSI). The author points out that the PSI only integrates the product support, and does not eliminate the need for logistical services within the DoD. He further points out that the PSI does not have to be the OEM but can be either a government or industry entity. However, because in most cases the OEM is the PSI, the misconception has developed that the PSI must be the OEM. The author also provides figures showing recent cost and time savings within the DoD, which can be directly attributed to a program's use of PBL strategies.

Kim, Cohen and Netessine (2007) also recognize the difficulties encountered when seeking to implement PBL contracts. This paper provides guidance with respect to what type of contract should be used in certain contractual situations. In this paper, the authors present a PBL strategy of purchasing the "results of a product" as opposed to buying the actual repair parts, spares, and maintenance activities. Due to its success in the private sector, PBL was implemented in the DoD as the preferred method for purchasing product life cycle support. The PBL approach does not specify how a contractor must support the product, only the required level of support. However, very few contractors have embraced PBL, and the Government Accountability Office stated that savings related to the implementation of PBL could not be demonstrated. With this background, the authors seek to show how a PBL-type contract can be successfully executed based on the participants' risk strategies. The authors also seek to show which type of contract (fixed-price, cost-plus, or PBL) or combination of contracts is best suited for certain contractual settings. Their results show that if a contractor's decisions are able to be observed and defined, a fixed-price/cost-plus contract is preferable. However, if the contractor's services are unobservable and all parties are risk neutral, a fixed-price contract with PBL incentives is best. Lastly, the authors determine that if any of the parties is risk averse, an optimal contract cannot be executed. In this case, the best contract combines elements of each of the evaluated contract types. The models used in this paper to analyze the different contracting environments are an inventory allocation model and the moral hazard model.

The importance of optimizing how we buy and implement supportability is a vital part of the acquisition process. Critical factors impacting procurement of supportability are limited funding from disparate sources coupled with an uncertain time frame in which the OEM is available to provide the spares/repair capability associated with modern torpedoes and for which the OEM holds the product design and repair know-how. As a result, it is more important than ever to implement a sound methodology that can quickly and accurately address our spares requirements. The following section compares and contrasts PBL versus traditional life cycle support.

Pay Me Now, Pay Me Later

When considering the trade-offs between the performance-based contracting approach (pay me now) and the standard contracting approach (pay me later), it is important to analyze several factors associated with the product or system to be acquired. These factors include, but are not limited to, the overall life-cycle cost, the expected future service and spares' costs, the acquisition's complexity, and the life-cycle length.

In the standard approach to contract writing, services and spares are purchased post-production as needed. This offers several distinct advantages and disadvantages. Because the costs for future services and parts are not added into the contract's overall cost, the starting contract cost is reduced. This in turn lowers the budget allocated to the contract, and allows the unused money to be used for other program needs. However, even though the costs of services and spares are not seen in the original contract cost, they are expected to be purchased as needed in the future. This can cause several problems for the customer. First, in the case of products with a long life cycle (let us assume a life cycle greater than 10 years), if replacement parts or spares are needed for the product after manufacturing has ended, the customer has limited (and often expensive) options for obtaining the needed parts. The customer could approach the original contractor about restarting production, which is likely to be more expensive than the original production cost. This is because the part may be obsolete at this point and unable to be sold or used for any purpose other than as a spare. The customer could also approach a new contractor about recreating the original part. This approach can face problems due to lack of know-how, incomplete documentation on the original part, and testing time and money needed to integrate the new part into the original system. The last option would be to design an entirely new part, which could boost the functionality of the system but would most likely be time consuming and expensive to build and test.

In the PBL approach, spares and services based on a performance measure are purchased up-front and included in the cost of the production contract. This approach also has several advantages and disadvantages. The main hurdle to this approach is the early planning to reprogram out-year supportability funding into the current contract year (aka transition year). The transition year necessitates auxiliary funding to pay for the PBL CLIN. This contributes to a perceived increase in the overall contract cost at contract inception. If the negotiated costs associated with the PBL CLIN were equal to or less than the cost of spares, there would be no increased cost to the enterprise. Purchasing services and spares based on a performance measure, such as operational availability, can save money in the long term. The source of auxiliary funding could be the funding currently used to procure spares; this also requires reprogramming money intended for hardware spares procurement to purchase “supportability” services. An additional challenge for the TE is the inconsistency between the production contract period of performance and torpedo life-cycle. The production contract has a period of performance of six years (i.e., one base year, four option years, one warranty year), whereas the torpedo’s life-cycle is 25 to 40 years (although its maintenance due date is significantly less than that). If a contractor is obligated to support and provide a system’s spares for the full life cycle, the disadvantages for the standard contracting approach become the advantages of the PBL approach. The money (and perhaps the time) that would have to be spent in the future is eliminated. It becomes the contractor’s responsibility to determine how the system will be supported. The contractor can manufacture a large surplus of spares and stock-pile them for the future, maintain (or mothball) a small production line to satisfy future demand, and/or build a highly reliable product that minimizes (or eliminates) the need for the first two options.

In conclusion, when determining which contracting method to employ, it is important to determine the complexity, life-cycle length, and expected costs associated with the product being acquired. Simple products that should not require extensive or unique sparing and servicing in the future might be better suited to the standard approach. Likewise, short life cycle products that are not expected to outlive the manufacturing processes producing them might also be better suited to the standard approach. However, complex and extended life cycle products would most likely be better supported and maintained using a PBL contract. The final factor when determining which contract to use is the expected life cycle cost of the system. If the future costs for sparing and services of the system are expected to exceed the extra cost associated with a PBL contract, then a PBL contract should be used. Cost estimates need to consider the future cost of money in this process. After deciding to utilize the PBL contract methodology, contract requirements in the form of metrics must be selected and defined.

A Short Discussion of Common Inventory Metrics

To better understand the status of an inventory's current state and level of effectiveness, an abbreviated list of relevant and commonly used inventory metrics are identified below. The metrics selected (DAU, 2010) are separated into two categories: "Enterprise" and "Source." Enterprise metrics measure the variables determined by the customer, while Source metrics measure the variables determined by the contractor.

First we will discuss the Enterprise inventory metrics. These include:

- Inventory turns,
- Perfect order fulfillment rate,
- Supply chain response time, and
- Weapon non-mission-capable (NMC) rate.

The *inventory turns* metric measures how much inventory is being used compared to the amount of inventory that is on hand (average) over a certain time period. It can be defined as how much of a certain measure of inventory (i.e., monetary worth, amount, or number of assemblies) is removed from the inventory divided by the average of that measure over the time period being analyzed. In the case of the HWT spares inventory being discussed, the spares stored are used to replace parts (FIRs) internal to the product (HWTs). For this reason, the optimal value for the spares *inventory turns* metric is zero, which correlates to an organization that never needs to replace parts internal to its products.

Perfect order fulfillment rate, when related to the organization's inventory, is defined as the ratio of perfectly satisfied orders and total orders filled from the organization's inventory. A perfectly satisfied order is defined as an order delivered with all of the ordered parts in perfect condition, on time and with all of the necessary documentation.

The *supply chain response time* of an enterprise is defined as the average amount of time it takes from recognizing the need for a certain part to the time the part arrives at the organization and is ready for use. This metric can be broken down into more discrete segments such as the time it takes to plan an order, the time it takes to source the part, and the amount of time it takes for the part to be delivered to the organization.

The metric referred to as the *weapon Non-Mission-Capable (NMC) rate* is the ratio of weapons in the fleet that cannot be used to complete their specified mission and the total amount of weapons in the fleet. This is a very important metric for the TE because it helps define the mission readiness of the larger submarine enterprise. If the submarine's primary weapon is not mission ready at an acceptable rate, then the mission readiness of the submarine will be greatly decreased and therefore the mission readiness of the Navy will be adversely impacted.

We will now proceed to discuss some common Source inventory metrics. The following metrics are mostly concerned with the quality of the delivered order and the time it takes for an order to be delivered. They are:

- Percent of perfect order fulfillment,
- Percent of correct quantity deliveries,
- Percent of defect-free deliveries,
- Percent of deliveries with correct documentation,
- Percent of on-time deliveries,

- Total source lead-time,
- Handling lead times,
- Receiving lead time, and
- Supplier lead time.

Percent of perfect order fulfillment if shown in a Venn diagram would be the unity of the percent of correct quantity deliveries, percent of defect-free deliveries, percent of deliveries with correct documentation, and percent of on-time deliveries metrics. These metrics are relatively straight forward to measure and are self defining. The importance of the percent of perfect order fulfillment is that it gives a high-level view of the a contractor's actual order fulfillment capability, while the metrics that make up a perfect order are more granular and point to actual problems the contractor might be experiencing in their order filling process. These insights can then lead to correction strategies for these problems.

The metric *total source lead time* is very similar to the Enterprise metric *supply chain response time*, except that this lead time is calculated from the contractor's point of view. *Total source lead time* can be viewed as the amount of contractor time elapsed, from the time they become aware of an order being placed to the time that order becomes available to the customer. This is equal to the *supply chain response time* minus the time it takes for the customer to recognize its need to order a part and the order being placed.

Handling lead time refers to the amount of time it takes from receipt of a shipment until the individual parts are put in their first official storage positions at the customer's facility. In the case of an order of office supplies, this lead time would be the amount of time elapsed between the shipment being recognized as arriving at the office and the supplies being placed in the supply buffer area.

Receiving lead time is slightly different than *handling lead time*. *Receiving lead time* is the time immediately before *handling lead time*. *Receiving lead time* is the amount of time that elapses between delivery to the customer's facility and the time when the ordering facility recognizes the shipment as being received. Using the office supplies example again, let's suppose that the shipment were delivered to the office building after hours and the box was first found by the secretary the next morning. The time between delivery and the secretary finding the box would be the *receiving lead time*.

Supplier lead time is defined as the amount of time it takes from order confirmation to the time the order arrives at the ordering facility. Again using the office supplies example, if the secretary ordered the office supplies online, this would be the amount of time from when the secretary received the order confirmation e-mail to the time the shipment was left at the office building by the delivery company.

Several other metrics commonly associated with inventories are:

- System Reliability,
- Product Reliability,
- Operational Availability,
- Mean Time to Repair (MTTR),
- Mean Time to Failure (MTTF),
- Mean Logistics Delay Time (MLDT),
- Mean Supply Response Time (MSRT), and

- Mean Accumulated Down Time (MADT).

System reliability refers to the ability of a system to achieve its specified goals and is measured as a percentage value. For the purpose of this discussion, it is assumed that a system is comprised of many products. In our case, the system we are considering is the torpedo and the products are the FIRs. The torpedo's reliability can be calculated by dividing the number of in-water runs in which there are no failures by the total number of in-water runs. It is important to remember that torpedo reliability is determined by the reliability of the torpedo components.

Product reliability is calculated by dividing, at the FIR level, the number of times a product performs its task correctly by the number of times the product is asked to perform its specified task. As stated in the previous paragraph, *product reliability* determines *system reliability*. Therefore, *system reliability* cannot be greater than *product reliability*.

Operational Availability (A_o) is determined by a number of factors, including *system reliability*. *Operational availability* is defined as the percentage of time that a group of products or systems is available to be used for its intended purpose or the percentage of the group's up-time.

Mean time to repair is the expected amount of time it takes from the time a product or system fails until that product or system is available for use again.

Mean time to failure is the expected amount of time a product is available for use after a repair or purchase until the product or system experiences its next major (or debilitating) failure.

Mean logistics delay time is the sum of the two logistical activities at the beginning and end of the *mean time to repair* metric. The first logistical activity is the amount of time from when the product or system fails to the time it arrives at the repair facility and is available for the needed reparatory action. The second logistical activity is the amount of time it takes from the time the repair is completed to the time the product or system is again able to be used by the product's (or system's) owner.

Mean supply response time is the expected amount of time it takes for a product's or system's supply system to respond to, repair or replace, and return the working product/system to the user.

Mean accumulated down time is the time that a group of systems or products is not operational and can be seen as an inverse metric to operational availability.

Metrics are Not a Two-way Street

The relationship between the metrics is illustrated in Table 1. When viewing Table 1 please note that while the metrics on the horizontal X- and vertical Y-axis are the same, the variables on the X-axis are the independent variables and the variables on the Y-axis are the dependent variables. This means that while variable "a" might influence variable "b," variable "b" does not therefore have an influence on variable "a."

This can be understood in Table 1 by considering the metrics "system reliability" and "product reliability" with the assumption that multiple products make up a system. In this case, the reliability of the individual products influences the reliability of the system. However, the system's reliability does not influence the individual products' reliabilities.

Table 2.

			Independent Metrics																			
	Optimal Values	Optimal Values		%	00%	00%	00%	00%	00%						00%	00%	00%					
			Inventory Turns	Weapon System NMC Rates	Perfect Order Fulfillment Rate	Percent of Correct uantity Deliveries	Percent of Defect-Free Deliveries	Percent of Deliveries with ect Documentation	Percent of On-Time Deliveries	Supply Chain Response Time	Total Source Lead-Time	Handling Lead Times	Receiving Lead Time	Supplier Lead Time	System Reliability	Product Reliability	Operational Availability	Mean Time To Repair (MTTR)	Mean Time To Failure (MTTF)	Mean Logistics Delay Time (MLDT)	Mean Supply Response Time (MSRT)	Mean Accumulated Down Time (MADT)
Dependent Metrics		Inventory Turns																				
	%	Weapon System NMC Rates																				
	00%	Perfect Order Fulfillment Rate																				
	00%	% of Correct Quantity Deliveries																				
	00%	% of Defect-Free Deliveries																				
	00%	% Deliveries with Correct Documentation																				
	00%	% of On-Time Deliveries																				
		Supply Chain Response Time																				
		Total Source Lead-Time																				
		Handling Lead Times																				
		Receiving Lead Time																				
		Supplier Lead Time																				
	00%	System Reliability																				
	00%	Product Reliability																				
	00%	Operational Availability																				
		Mean Time To Repair (MTTR)																				
		Mean Time To Failure (MTTF)																				
		Mean Logistics Delay Time (MLDT)																				
		Mean Supply Response Time (MSRT)																				
		Mean Accumulated Down Time (MADT)																				

As shown in this matrix, the "availability" metric is affected by many of the other metrics and may serve as a good indicator of the contractor's performance on a PBL contract.

Newsvendor-based Approaches for Designing PBL Contracts

The newsvendor problem is a single period mathematical model used to determine optimal inventory levels when the demand is uncertain (Porteus, 1991). The model assumes that a decision to procure a certain number of items (***Q***) is made at the start of a period. Subsequently, the random demand (***D***) for the item is revealed. The distribution of ***D*** is assumed to be ***F(D)***, with a mean μ. An ordering/restocking cost of *C* is charged per unit. If the number of items procured exceeds the realized demand, a per unit effective disposal cost of C_H is charged for the period. However, if the demand exceeds the amount procured, a per unit shortage cost of C_P is assessed for the period. An assumption is made that $F(x) = 0$ for $x < 0$. In this scenario, the cost function for one period is:

$$g(y) = Cy + \int_0^y C_H(y-\zeta)dF(\zeta) + \int_y^\infty C_P(\zeta - y)dF(\zeta)$$

The optimal order quantity that minimizes the cost is then computed as:

$$F(q^*) = \left(\frac{C_P - C}{C_P + C_H}\right).$$

Or

$$q^* = F^{-1}\left(\frac{C_P - C}{C_P + C_H}\right)$$

Here, F^{-1} is the inverse of the distribution function. The quantity $(C_P - C)/(C_P + C_H)$ is the critical fractile and is the optimal probability of not stocking out (Porteus, 1991).

The newsvendor problem has been used as a starting point for analyzing many scenarios. A review of some extensions can be found in Khouja (1999). Among the cases that can be related to the analysis of contractor performance are Dada, Petruzzi, and Schwarz (2006); Bensoussan, Feng, and Sethi (2004); Kim et al. (2007); and others. In Dada et al., a newsvendor model is used to structure a scenario when a single newsvendor is served by several suppliers, some or all of whom may be unreliable. This can be used for modeling operations in PBL when several vendors are contracted to maintain a supply of either weapon assemblies or subsystems (FIRS). In Bensoussan (2004), a vendor commits to an initial purchase, following which some estimate of the demand is revealed. Additional purchases can be made for a higher cost, subsequent to which the final demand is realized. An overall service constraint is also satisfied in determining the solutions to the two stages for ordering. In the context of PBL, each stage can represent the ordering decision at the IMA and the manufacturing facility for the vendor, while the service constraint can guarantee the availability. However, as noted by the authors, when there is private forecast information, the mechanism for coordination of the fleet and vendor's decisions remains to be determined.

As mentioned earlier, Kim et al. (2007) evaluated PBL as a strategy for purchasing the "results of a product" as opposed to buying the actual repair parts, spares and maintenance activities. One of the significant factors identified by the authors when designing incentives for PBL is the observability of contractor performance and the tolerance for risk by the parties involved in the contract.

In Kang, Doerr, and Sanchez (2006), it was noted that PBL specifies outcomes, not numbers of spare parts or hours of maintenance. The emphasis of the contract is on metrics to be achieved by the contractor (in this paper the metrics are operational availability and readiness risk) not the way in which the contractor must achieve the specified metrics. The authors use a simulation to show which alternatives customers should specify to increase operational availability and reduce readiness risk. Their simulation then helps estimate which alternatives will best improve the specified metrics for a given contractual environment. The model shows that transportation/administrative delay is a main determining factor for operational availability, whereas number of spares on the shelf is not.

In the context of torpedo production, under PBL contracts, the interaction between the contractor and the IMA is shown in Figure 2.

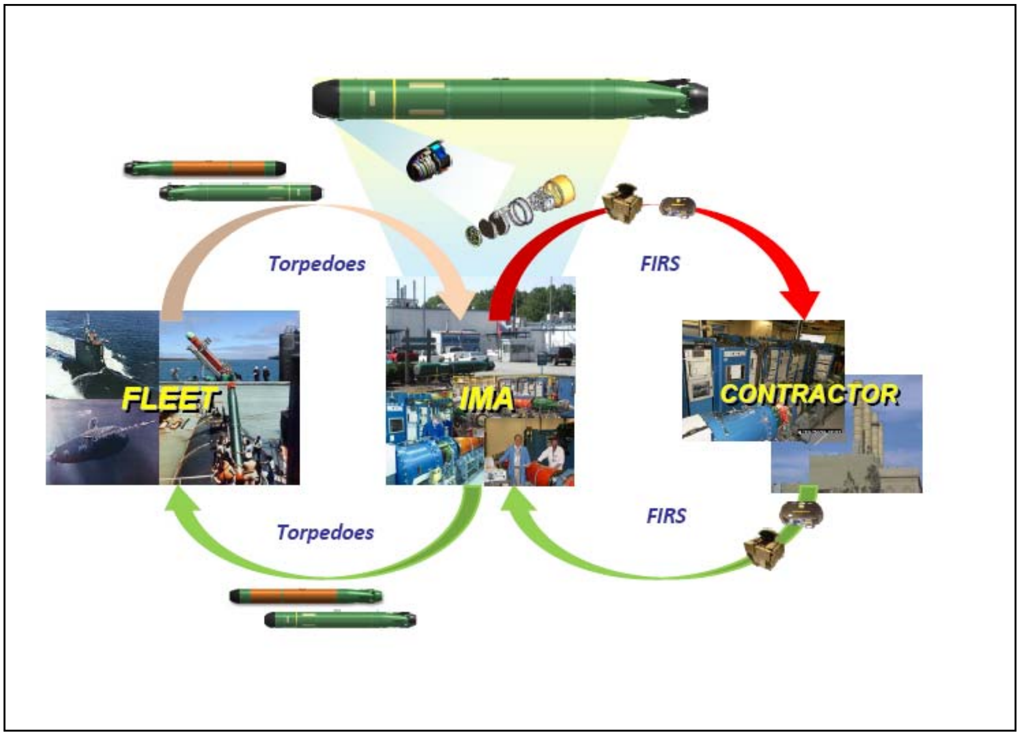


Figure 2. Contractor's Role in the Spares Support Process

The COSAL is the safety stock, and the random demand is generated by fleet usage. The cost of understocking is the total time spent by the IMA waiting for a particular FIR. The cost of overstocking can be assumed to be related to the average amount that it costs a single FIR to be shipped and the cost of managing and maintaining the inventory. Typically, the overstocking cost is low relative to the understocking cost, which implies that the vendor will have an incentive to maintain a large shelf inventory. However, in the context of PBL, the contractor must incentivize lower inventory levels so that the ultimate thrust is on reducing the need for an inventory (i.e. reducing the number of failures during fleet usage).

Simulation-Based Models of PBL Operations

Based on the discussion above, the following protocol for operating a PBL has been proposed: The contractor is made responsible for maintaining spares for the FIRs at the IMA. The maximum number of FIRs is specified in the COSAL for each IMA, and modifications to the COSAL to meet the required availability can be negotiated as part of this contract. When an incoming torpedo needs a replacement FIR, the inventory status of the FIR is determined by the current availability number in the system (**a**). If **a** is zero or negative, a request for immediate replenishment will be issued to the contractor. If **a** is positive, the spare FIR is removed from the container and issued to the IMA floor. The failed FIR is placed in the empty container and returned to the contractor. The contractor has visibility into the inventory level at each IMA at all times.

Clearly, the COSAL should relate to the failure level of a FIR. If the FIR never fails in service, then the corresponding COSAL value can be set to zero. However, since a zero failure rate is unlikely to be achieved, the COSAL must be set to some positive value. Based on analytical and simulation models and using specified reliability numbers for the FIRs, appropriate COSAL levels can be determined that will achieve desired supportability levels. If the contractor cannot meet availability numbers using the COSAL levels in the contract, this is an indicator that the reliability for the FIR has slipped below the expected reliability, and appropriate action must be taken to address this.

The following measure of performance has been developed for availability for an initial analysis:

$$\text{Availability} = 1 - \frac{\sum_{i=1}^{T} \text{number of units short}}{\text{total demand in the quarter}}.$$

The performance of this measure is a function of the failure rate, the stock level on shelf, and the variation in failure rates. Based on a hypothetical usage rate in excess of five hundred per year, Table 2, below, shows the results of a simulation exploring the relationship between the failure rate, the variation in failure rate (which would also represent the variation in the demand or number of torpedoes needed per week), and the COSAL values.

Table 3. Simulation Results for Evaluating Interaction of COSAL and Failure Rates

Availability (OPTEMPO = above 500 per year, Logistic Delay = 1 week)												
	Failure Rate Variation											
FIR Failure Rate	25%			50%			75%			100%		
↓	Common Shipboard Allowance Level (COSAL)											
	1	2	3	1	2	3	1	2	3	1	2	3
0.05	96.1%	100.0%	100.0%	96.1%	100.0%	100.0%	87.9%	100.0%	100.0%	76.5%	100.0%	100.0%
0.06	89.7%	100.0%	100.0%	83.9%	100.0%	100.0%	74.3%	100.0%	100.0%	65.8%	100.0%	100.0%
0.07	73.2%	100.0%	100.0%	67.5%	100.0%	100.0%	63.4%	97.8%	100.0%	57.1%	95.7%	100.0%
0.08	63.4%	100.0%	100.0%	59.1%	97.9%	100.0%	54.2%	96.0%	100.0%	48.1%	88.3%	100.0%
0.09	56.5%	100.0%	100.0%	51.5%	97.0%	100.0%	46.4%	94.4%	100.0%	43.7%	85.0%	99.2%
0.1	50.0%	100.0%	100.0%	47.3%	93.6%	100.0%	42.6%	84.4%	100.0%	39.1%	75.4%	98.5%
0.11	47.7%	93.6%	100.0%	41.3%	85.2%	100.0%	40.3%	74.5%	99.3%	35.4%	71.7%	94.6%
0.12	40.9%	86.7%	100.0%	38.5%	77.6%	100.0%	35.4%	70.7%	97.3%	32.7%	65.8%	92.0%
0.13	38.8%	78.8%	100.0%	35.9%	69.8%	98.6%	32.7%	64.6%	93.7%	30.6%	61.5%	88.9%
0.14	36.9%	73.8%	100.0%	34.2%	65.0%	95.5%	30.4%	62.3%	88.8%	27.7%	55.6%	82.0%
0.15	33.5%	70.3%	98.7%	30.4%	62.7%	90.1%	28.4%	58.1%	84.2%	26.1%	52.0%	78.3%

The entries in the table are the average (over 1,000 runs) of the Availability metric for a given failure rate (row label), and a random variation (for now, uniformly distributed–

column group header) and different COSAL levels. This simulation, implemented in a spreadsheet, verifies that as failure rates drop, the COSAL required to support fleet operations is smaller. The entries in this sheet could have been computed using a newsvendor approach directly—this did not require simulation. However, the actual nature of variation is somewhat more complicated. The simulation is designed to take variations in exercise rates typically encountered throughout the year and changes in the logistic delay to determine the optimal COSAL required to support the fleet. Furthermore, this simulation can also be used when negotiating with the contractors prior to the award of contract to determine what the contractors' estimates of their own failure rates are and to work with them to set mutually satisfactory expectations.

As mentioned in Kang et al. (2006), the transportation delay correlates most significantly with the operational availability. This is also borne out by the simulations performed above. Because of this, the responsibility for delivery to the shelf is best delegated to the contractor in a PBL setting.

An extension of this simulation allows an optimization of the COSAL required to achieve a given service level. This is not dissimilar to the approaches developed in Schneider (1978) and Shang and Song (2004), but the advantage of the simulation/optimization is that it dispenses with the assumptions of independence of failure rates that are often necessary for analytical solutions and the distributional assumptions that go along as well.

Conclusion

This paper discusses the application of Performance Based Logistics (PBL) contracts for supporting the Torpedo Enterprise. Several performance measures commonly used in PBL contracts are described, and a model is presented that uses an "availability" metric for observing and measuring contractor performance. This metric is calculated using the number of times a required part is not available to field maintenance sites. Terms of the contract are not specified with regard to lead times such as logistical delays and manufacturing and restocking lags. These times are assumed to be under the contractor's control, as are production quantities, quality, and responsiveness. A newsvendor approach for determining optimal shelf inventory levels is developed. An augmented model is evaluated using a simulation to determine the performance sensitivity to changes in product quality, demand rates, and various supply chain-related lead times. Practical collateral issues such as obsolescence, reliability, and cost are also discussed.

References

Bensoussan, A., Feng, Q., & Sethi, S.P. (2004). A two-stage newsvendor problem with a service constraint. Working Paper. University of Texas at Dallas, Richardson, TX.

Fowler, A. (2009). *Misunderstood superheroes*. Retrieved March 28, 2010 from www.dau.mil/pubscats/PubsCats/atl/fow_jf09.pdf

Dada, M., Petruzzi, N., & Schwarz, L.B. (2007). A newsvendor's procurement problem when suppliers are unreliable. *Operations Management*, *9*(1), 9-32.

Performance measure definitions. (n.d.). Retrieved March 25, 2010 from https://acc.dau.mil/CommunityBrowser.aspx?id=22646

Kang, K., Doerr, K.H., & Sanchez, S. (2006). A design of experiments approach to readiness risk analysis. In *Proceedings of the 2006 Winter Simulation Conference*.

Khouja, M. (1999). The single-period news-vendor problem: Literature review and suggestions for future research. *Omega*, *27*, 537-553.

Kim, S., Cohen, M.A., & Netessine, S. (2007). Performance contracting in after-sales service supply chains. *Management Science*, *53*(12), 1843-1858.

Packard, D. (1986). *A quest for excellence: Final report to the president by the president's Blue Ribbon Commission on Defense Management*. Washington, DC: US GPO.

Perry, W.J. (1994). *Acquisition reform, a mandate for change*. Testimony before the Senate Committee on Armed Services and Senate Committee on Governmental Affairs. Washington, DC: US GPO.

Porteus, E. (1991). Stochastic inventory theory. In D.P. Heyman & M.J. Sobel (Eds.), *Handbooks in operations research and management science* (Vol. 2). Amsterdam: Elsevier Science.

Schneider, H. (1978). Methods for determining the re-order point of an (s, S) ordering policy when a service level is specified. *Operational Research Society*, *29*, 1181-1193.

Shang, K., & Song, J. (2004). Analysis of serial supply chains with a service constraint. Working Paper. Duke University, Durham, NC.

Contractor Incentives for Success in Implementing Performance-Based Logistics: A Progress Report

Jatinder (Jeet) N.D. Gupta—Jatinder (Jeet) N.D. Gupta is Eminent Scholar of Management of Technology, Professor of Information Systems, Industrial and Systems Engineering and Engineering Management at the University of Alabama in Huntsville. Recipient of the Outstanding Faculty, Teacher and Researcher awards, he has published extensively in national and international journals. He serves on editorial boards of journals and edited several special issues, including *European Journal of Operational Research*, and books, including the *Handbook of Research on Enterprise Systems*. His current research interests include e-Commerce, Supply Chain Management and Logistics, Information and Decision Technologies, Scheduling, Planning and Control, Knowledge Management, and Enterprise Integration.

Michael C. Eagan—Michael C. Eagan is a graduate student at the University of Alabama in Huntsville. He will graduate in May 2010 with an MBA and has an undergraduate degree in History from Hampden-Sydney College. He served as a summer intern in the office of Strategic Analysis and Communication of the Marshall Space Flight Center, Huntsville, AL.

Joshua N. Jones—Joshua N. Jones received his MBA in Finance from UAHuntsville in December of 2009. He was a member of the UAH AMCOM Supply Chain Research Team for the 2009 calendar year and is currently working on another Naval Postgraduate project under the direction of Dr. David Berkowitz that deals with the impact of relationships and contractual mechanisms on the overall success of a Major Defense Acquisition Program. Joshua hopes to begin his PhD in the fall of 2011 in Strategic Management.

James C. Platt—James C. Platt is a graduate student in Information Systems at the University of Alabama in Hutnsville, where he completed an MBA in December of 2008. He graduated with a BS in Finance from the University of North Alabama. His career consists of 25 years in business, eighteen of which were in electronic purchasing. The last seven years in purchasing were in the Defense and Aerospace division as a Senior Buyer. His hobbies include vocal performance and playing four different instruments.

Abstract

In implementing performance-based logistics (PBL), the need for several resources like inventory investment decreases. Therefore, the contractor's profit, which was based on the level of these resources, may decrease. Therefore, a contractor may have disincentives to implement and use PBL. One way to handle such a situation is to develop financial models to assist with profit/cost sharing during the implementation of PBL. Another way is to study the broader topic of contractor incentives in PBL to find appropriate ways to motivate the contractors to enhance their performance. While most literature in PBL mentions the importance of contractor incentives, not much research has been conducted in this topic. With such situations in mind, this research program proposes a framework to study and develop appropriate possible contractor incentives to succeed in the PBL environment. Our proposed framework considers the possibility of financial and non-financial contractor incentives to ensure PBL success. We anticipate that the final results of this research program will be useful in the defense and related public- and private-sector organizations to maximize the overall benefits of PBL projects. This paper provides a progress report in developing our proposed framework and outlines the remaining work to be completed.

Introduction

PBL is the desired product support strategy in the DoD, and it is being integrated into legacy programs as well as into new contracts (Berkowitz, Gupta, Simpson & McWilliams, 2005; Vitasek et al., 2008; Vitasek & Geary, 2008). It is used by all services at the component, system, and subsystem levels of procurement, sustainment and support (Geary & Vitasek, 2008). PBL came to the forefront of government procurement because a better method was needed. Therefore, the implementation of PBL was included in September 2001 in "the Quadrennial Defense Review and initial guidance was issued by the Office of the Secretary of Defense (OSD)" (Aguilar, Estrada & Myers, 2005, p. 13). The implementation of PBL is important because it gives the program manager (PM) the ability to improve reliability, reduce the logistical burden, and save money on Total Life Cycle Costs (Kim, Cohen & Netessine, 2007).

Since PBL is relatively new, it had to be successfully integrated into thousands of legacy contracts and new programs. To effectively do this, the idea of Total Life Cycle Systems Management was propagated around the service branches (Edwards & Nash, 2007; Kratz & Buckingham, 2010). "Total Life Cycle Systems Management emphasizes an early focus on sustainment in the program management office, making the PM responsible for all activities associated with the acquisition, development, production, fielding, sustainment, and disposal of a weapon system across its life cycle" (Aguilar et al., 2005, p. 13). This is the main difference from older procurement methods because they focused solely on the early phases. PBL is a major jump forward in how government procurement and sustainment is done and contract types and incentives need to readapt so they can successfully support the contract (Barber, 2008).

The Department of Defense and the Military Services are transforming from traditional methods of logistics support to PBL as a methodology of product support for the 21st century (Fowler, 2010; Kratz & Buckingham, 2010). It makes the program managers responsible for total life cycle costs (DAU, 2005). Traditionally, support for MWSs in the DoD centered around 10 logistics elements, split between acquisition-related activities at the front end of the life cycle and sustainment-related activities at the back end. Metrics focused on the logistics elements themselves and on internal processes often having little direct relationship to warfighter requirements. The shift toward Integrated Logistics Support attempted to combine distinct logistics elements into a coordinated approach, but there was still the disjointed acquisition versus sustainment-support issue and the lack of a linkage between supportability measures and warfighter needs (Vitasek & Murray, 2009). The advent of Total Life Cycle Systems Management (TLCSM) and performance-based logistics (PBL) addressed all of these issues (DeVries, 2005).

TLCSM mandated a new focus by PMs toward the entire life cycle, firmly linking acquisition and sustainment activities into an integrated process. This was a significant paradigm shift from PMs traditional focus on the early stages (acquisition, development, fielding) of the life cycle. To measure success, PBL required that supportability metrics be directly related to performance outcomes for the warfighter. PBL also offered a choice of organic and commercial support providers for picking the right combination in achieving best value for the program (DeVries, 2005).

The 2001 *Quadrennial Defense Review (QDR)* identified logistics transformation as a key transformation pillar. Specifically, the *QDR* directed logistics enterprise integration, a reduction in logistics demand, and a reduction in the cost of logistics. The 2003 update to the *5000.02* is actually the directive that matters. That is the first time the Components were actually put on the hook to do PBL. One tool the Department of Defense (DoD) identified for use in achieving these goals is performance-based logistics (PBL). This tool is an innovative acquisition approach that represents a cultural shift away from buying parts to buying performance. In practice, application of PBL can be at the system, subsystem, or major assembly level. Executed through long-term incentive based contracts, PBL is a means of system sustainment that integrates supplier support and warfighter requirements with the objective of improving operational readiness while reducing costs. In today's environment of constrained budgets and reduced manpower, PBL represents a potentially cost-effective and efficient method for system sustainment (Lewis, 2005; Mahon, 2007).

It has been said, "it simply makes good business sense to provide the proper contract motivations to encourage high-quality contractor performance." It is this notion of "good business sense" that we would like to examine further. Within the construct of performance-based logistics (PBL), contracts have been written to try and motivate contractors to meet the expectations of the government by constructing incentives that greater serve the needs of the contractor. This implies that if a contractor is performing well, then the proper incentives must be in place. Assuming that is the case, we want to ask the following: what were those incentives, what was the methodology (i.e., "best practices") for selecting those incentives, and would a consistent pattern between the types of incentives and levels of performance indicate the use (or lack) of "best practices," when developing these incentives had a hand in a firm's level of performance? (Gilbreth & Hubbard, 2008; Graham, 2003; Hildebrandt, 1998).

The incentives given to a contractor can be either monetary, non-monetary, or both. Several metrics exist that allow government contracting officers to objectively evaluate the performance of a contractor (Doerr, Lewis & Eaton, 2005). Monetary incentives could be based upon, but are not limited to, the following: material availability, material reliability, and life cycle cost, all at the system level. Delivery schedule incentives focus on getting a contractor to meet or exceed minimum delivery requirements. Under a performance-based construct, the parties involved have relative autonomy in negotiating the terms and conditions for meeting the target delivery dates. Performance standards are defined in a PBC and the incentives are typically given on the basis of whether the contractor met the performance criteria and to what extent they exceeded the standard (Tremaine, 2008). In other words, performance incentives are designed to relate profit to the contractor's achieved results.

Several types of performance incentives are fee structures, bonuses, and/or shared savings, and they can be applied in many different ways for many different reasons; some examples of these are as follows:

- First, an award fee can be given if the government feels that the contractor meets or exceeds specified outcomes.
- Second, an incentive can be given if the contractor appropriately controls costs in a cost plus-incentive-fee contract.
- Third, reliability-based profits allow for increased profits (as in FFP contracts), if the contractor can lower their operating costs by meeting higher product reliability standards. So, they can retain at least a substantial portion of the profits by making a better product.
- Fourth, shared savings is another unique type of bonus because both the contractor and the government share in the savings that result

> from performance enhancements, design improvements, and other efficiency improvements (Kirk & DePalma, 2005, p. 40).

Since PBL is now the preferred procurement and system support program, it is important to know how to incentivize contractors to perform consistently at a high level. However, this is a unique problem considering that PBL is new and little is known about what best practices and incentives should be suggested. As a result, it is important to look at the different types of contracts the government can issue before discussing how incentives can be given under PBLs oversight.

Contract Types

Fixed-price and cost-plus contracts will elicit different contractor responses based upon the inherent nature of the two types. When a contractor is awarded a fixed-price contract, we feel that the contractor is motivated to reduce product support costs because the awarded amount is fixed; therefore, every dollar saved through cost reduction is an additional dollar of profit contribution. Knowing this, we argue that firms operating under fixed-price contracts should be inherently motivated to reduce costs because the cost of not doing so will reduce that firm's profit potential. And when firms that operating under an FP construct experience cost overruns, we have to consider whether the incentives being offered were consistent with performance goals.

Under a fixed-price construct, there are essentially two widely used incentive-type contracts: (1) fixed-price-incentive-fee (FPIF) and (2) fixed-price-award-fee (FPAF). FPIF contracts are based upon a formula that relates final negotiated cost to target cost—these targets could be either firm target or successive targets. The formula used is made up of variables that can be objectively determined (i.e., cost, schedule, performance). Sometimes, Award Fee elements are in fact things that can be objectively measured, but an Award Fee approach is chosen. Award Fees are easier to administer and allow more flexibility on the part of the government.

Cost reimbursement (or cost-plus) contracts are appropriate and largely used in the developmental stages of the product/project life cycle, where costs are essentially unpredictable. When a contractor is awarded a cost-plus contract, there is typically too much ambiguity in the project to assign a fixed price to the end product; therefore, the cost risk for the government is usually greater when cost-plus contracts are used rather than fixed-price. This ambiguity is the result of many things, including technological maturity, political uncertainty, etc. Contractors typically enjoy the freedoms associated with cost-plus contracts because the cost risk associated with a particular project or program is shifted to the government. It is noteworthy that development costs often exceed production costs associated with a product ready for use, even if the product is produced well beyond maturity, when per unit product costs drastically decline.

Taking these thoughts into consideration, we argue that because contractors run a greater risk of financial loss (due largely to the uncertainty associated with cost-plus contracts), the choosing of incentives should be seen as a much more sensitive and delicate process in the eyes of the contractor. If this proves to be true, then the types of incentives used by the government for a particular product could have a significant impact on how motivated a contractor is to meet or exceed the predetermined performance targets.

Incentives for the Government

The Firm Fixed-Price (FFP) contract is the desired contract for the government, as it firmly fixes pricing parameters, shifting the risk of cost overruns to the contractor. When the product life cycle inevitably requires conceptual development and prototyping, the government seeks to accelerate the product to maturity so that FFP contracts become appropriate. The government is subject to congressional funding and the government must show performance to justify funding, which also creates an added incentive to perform responsibly.

Cost-plus contracts are appropriate and largely used in the early developmental stages of the product or project life cycle when the costs are unpredictable. This is not the desired contact type for the government, but it is necessary to reimburse the contractor for unpredictable and volatile costs. The cost-plus contracts shift the risk to the government as they are obliged to cover unpredictable developmental costs within contractual guidelines. It is noteworthy that the developmental costs often exceed the productions costs associated with a product ready for use, even if the product is produced well beyond maturity, when per unit product costs drastically decline.

This brings to mind the "S-curve," prominently known in marketing, where the product begins as a concept and is then brought to the market seeking adoption; then, the product or service is improved and adopted by a large portion of the market; finally, it reaches maturity, pending obsolescence. As shown in Figure 1, the maturity stage essentially predicts obsolescence and prompts replacement with a new product (Visitask, 2010).

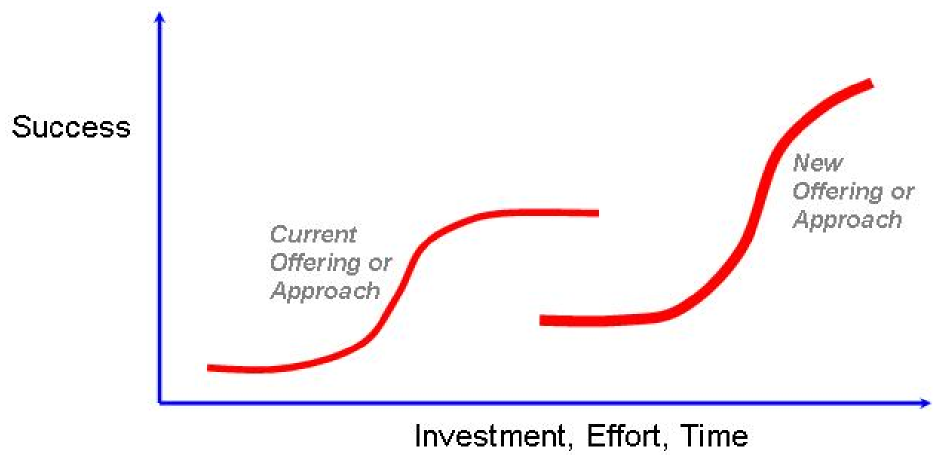


Figure 1.

The government still reserves the right to "terminate for convenience," which absolves the government from future obligation to the contract (GSA, DoD & NASA, 2010). It should also be noted that it is paramount to the government to specify parameters of contract performance, which shifts the risk to the contractor, because a nebulous contract would give too much discretion to the contractor. This situation creates somewhat of a dilemma considering that one of the basic tenants of PBL is that the government is to basically specify what it wants, while allowing the contractor to determine the means of fulfillment (Defense, 2010).

Incentives for Contractors

Under the PBL strategy, the contractor assumes a greater amount of risk; this, however, gives the contractor more latitude in determining and applying its methods (KMC/OPI, 2010). The general consensus, identified through research and personal interviews, is that the continuation of a contract is the main incentive. Continuation simply allows more time for the contractor to recoup capital investment expenses and provides added stability and continuity of staff, expertise, and equipment (P. Cushman, personal communication, October 2009). The general idea derived from the interviews was that a contract term should last at least five years to allow time for the contractor to recoup its capital investment (D. Wilson, personal communication, October 2009). Incentives can be tied to the performance of metrics as they relate to cost, quality, or delivery. For instance, under an FFP contract, a contractor may keep at least a portion of dollars saved when below budget, and it may receive bonuses for reaching certain metrics stated in the contract.

This is apparent in cost-plus contracts because the contractor is incentivized to keep costs and timeframe to a minimum so that it may win continuation of the contract. Even if the contractor is reimbursed for cost overruns, it runs the risk of congressional scrutiny and termination either by convenience or in favor of another contractor, so corporate reputation is at stake on every contract (GSA et al, 2010, p. 1). The reputation of a company may be the most important factor in contract rewards, and it's dependent on how they perform in every contract (Defense, 2010, p. 1). For example, Boeing's poor performance during the competition for the Joint Strike Fighter, contrasted by a relatively better performance by, undoubtedly influenced other contract awards, as evidenced by the recent proliferation of Lockheed contracts (*High Stakes*, 2010, p. 1).

Types of PBL Incentives

In order for a PBL contract to be successful, the contractor has to meet standards established in the contract. To ensure the standards are met, the cost to provide the required level of service is estimated to the best of the government's and the contractor's abilities. However, complications can arise and levels of achievement can be met that warrant additional compensation like bonuses, shared savings, or other forms. Examples of complications are project risks, adjusted product usage, and increases in the price of used resources or components. Aside from that, superior performance levels can be outlined in the contract that also warrant an incentive when met. It is important to have incentives and bonuses as a part of PBL contracts because they can increase the probability that the contract is fulfilled and the warfighter receives the necessary support on time (*High Stakes*, 2010, p. 40). While the topic of contractor incentives is mentioned in various research studies (Beggs, Ertel & Jones, 2005), it is not explored to the extent of developing a framework for determining such incentives in specific situations.

Proposed Framework

Before moving on, it is important to understand that incentives can be given in many different forms, but they fall into three categories: cost-based incentives, time-based incentives, and scope-base incentives. ***Cost-based incentives*** focus on contractor profits, so monetary awards are a good example of this type of incentives. ***Time-based incentives*** are changes in the length of the contract, so the life of the contract is extended for the contractor. ***Scope-based incentives*** are changes in the contract that give the contractor more responsibility and, as a result, larger incentives. For identifying which incentives have the most impact on contractor performance, it is important to separate these incentives because different types of incentives work better with different types of contracts.

Cost-based Incentives

Of all of the incentives that are considered, the most important type of incentive considered is the contract type. This is because contract types vary in their treatment and allocation of cost, schedule, and performance risk. *FAR* Part 16 defines contracts as being one of two types: (1) fixed-price or (2) cost-reimbursement. When deciding how to match incentives with the contractual mechanism being used, the contracting officer needs to look at where most of the responsibilities and risk lie. Under a fixed-price (FP) construct, the contractor assumes all responsibility and risk of fulfilling the contract and providing a product, whereas under a cost-reimbursement construct (i.e., cost-plus contracts), the government reimburses certain allowable and allocable costs, and pays the contractor a fee that is in line with the contractual agreement. The inherent downside in using cost-reimbursement contracts is the lack of motivation on the part of the contractor to reduce costs. As one might suspect, the type of contract that is appropriate for a particular task depends upon several variables, but for the purpose of our research, we want to evaluate the types of incentives being used and determine their overall ability to influence the behavior of the contractor.

Because PBL is focused on system availability and performance metrics, it is important for the contractor to understand what is expected of them. An example of metrics being linked to profits is how “the TOW-ITAS contract directly links profitability to availability—the higher the availability the greater the profit the supplier can earn” (DAU, 2005, pp. 3-23). In this example, in order to link availability to profitability, the acceptable level is determined by the program office and is included in the performance-based agreement (PBA). The first step in meeting the objectives stated in a PBL contract is deciding on a cost that will cover the desired support. Once the PM decides on a cost, it needs to be accepted by stakeholders before the PM can enter into a formal agreement with the contractor. Finally, the level of support provided needs to be appropriately covered by compensation in order for the contract to be appealing to competent companies.

The written agreement between the contractor and the government becomes a part of the PBL support strategy, and the expectations of the contractor are outlined in the user agreement section of the PBA. The user agreement part of the PBA contains the ranges and objectives that the contractor has to meet. “Typically, the agreement identifies ranges of outcome performance with thresholds and objectives, and the target price (cost to the user) for each level of PBL capability” (DAU, 2005, pp. 3-17). Since the contract is based on ranges and objectives, resources will be allocated to the contractor on the basis of what is expected from the weapon system or component in a particular year. It is important to note that unique conditions may require performance above normal operations, and the policy will adapt to meet these conditions. For instance, “PBL agreements should be flexible enough to address a range of support requirements, so as to accommodate changes in OPTEMPO or execution year funding, including surge or contingency requirements to the extent that they can be defined” (DAU, 2005, pp. 3-18). An example of this is the Shadow Unmanned Aerial Vehicle contract, which “procures performance using measurable metrics instead of buying spares and repairs in the traditional manner. This example demonstrates the establishment of a schedule for the transition from Contractor Logistics Support (CLS) to PBL based on lessons learned from operational usage in the user environment” (DAU, 2005, pp. 3-24). As a result, user agreements are important because they outline the requirements and ranges that need to be met in the contract and how to deal with wear and tear changes, like the Shadow UAV contract.

When getting involved in a contract with the government, it is important for the contractor to understand the level of risk they are being asked to assume. One of the main components of PBLs is that it moves risk away from the government and to the contractor. “While DoD can never completely delegate risk for system operational performance, PBL strategies move the level of risk away from DoD to the support provider, commensurate with the scope of support for which the support provider is responsible” (DAU, 2005, pp. 3-20). Risk is an important part of performance-based agreements because it affects how the PM defines an acceptable level of performance, cost, and incentives. This means that the government will design its PBL contracts in a way that makes assuming risks appealing to the contractor. Properly compensating parties for taking risks is an important part of successful PBL contracts, and if it is done correctly, then risk to the government will be significantly reduced.

Time-based Incentives

PBL incentives are generally tied to the contract type and overall performance, so additional compensation is typically given for exceeding the standards as stipulated by the contract. For example, “in most cases, providing incentives for PBL contracts is difficult considering the many different types of contracts that may be used” (DAU, 2005, pp. 3-19). Giving incentives in respect to PBAs means that incentives are given based on meeting the metrics set in the contract, but, in general, the PBL wants to tie incentives to overall performance (D. Ioasco, personal communication, October 2009). “The preferred PBL contracting approach is the use of long-term contracts with the incentives tied to performance” (DAU, 2005, p. 3-19). This will help the contractor make technological investments to improve the system performance, with the hope of making relatively more profit in the long-term. Klevan (2008) reports that most Navy PBL contracts are long-term agreements and address availability, obsolescence, reliability, and cost. He further reports that the use of such time-based incentives creates a win-win strategy, incorporates surge capability, mitigates risks and ensures an exit strategy. Such long-term agreements create government-contractor partnerships that result in significant improvements in the key performance parameters specified in the PBL agreements (Klevan 2008). It enables the government to procure the “end-state” and not the “how-to.” Thus, using overall performance as a basis for rewarding long-term contracts incentivizes the best companies in the industry to apply for contracts in hopes that they will receive future business (S. Kowerduck, personal communication, October 2009).

Scope-based Incentives

While cost-based and time-based incentives are in use, scope-based incentives are not much in use, but may provide motivation to the contractor to significantly improve its performance under the PBL. This incentive is based on the assumption that a contractor wishes to expand its business with the government. If the contractor’s performance under a PBL contractor exceeds the government’s expectation, then the contractor can be given work beyond the scope of the original contract. For example, if the original PBL contract required a contractor to maintain an aircraft engine, then based on a superior performance over time, this contractor can be given the additional task of maintaining tires. Ultimately, this contactor may become the system integrator, as it will become responsible for the entire aircraft. Using overall performance to create scope-based incentives is the best way to provide incentives because the metrics are difficult to define effectively. However, because of current legal restrictions and the need for competitive procurement in government, implementation of scope-based incentives is quite difficult, if not impossible.

Private Sector

The PBL strategy is formulated with the intent to improve upon older contracting methods and provide incentives similar to the private sector. As a result, "it is not uncommon for contractors engaged in PBL contracts to have the majority—or even all—of their profit tied to performance-based metrics and dependent on earning the contractual incentives included in the contract" (DAU, 2005, pp. 3-21). Using incentives to encourage better contractor performance is a useful tool, but finding the right incentives and metrics is difficult. For example, a commercial company is going to require different incentives than a depot. One is a for profit and one is trying to breakeven; so, a depot may want the incentives to, among other things, reduce operating costs and encourage savings, while a commercial company wants profits to please their stockholders and board members. Applying commercial style incentives to government contracting or partnerships with the depots is a way to encourage the best level of performance, but the optimal incentives are needed in order for it to be successful.

The private sector provides incentives based on performance, and this may improve the quality of the product and service provided. In order for the government to get a better product, the government needs to develop a commercial mentality to incentivizing. The *Federal Acquisition Regulation (FAR)* Part 12 is designed to encourage the government to move toward using commercial processes and practices. Incorporating commerciality into government procurement under *FAR* Part 12 can be done at multiple levels in the government contractor relationship. For example, "justification for commerciality does not have to be made at the item level; it can be made at the repair process level or at the support concept level" (DAU, 2005, pp. 3-24). So, incentives should be provided at more than just the item level. Under this regulation, the government can incentivize by methods known as the "Power by the Hour (PBH)" concept, a company's exceptional repair capabilities, or a product support system.

PBL is an important step in the right direction because it allows the government to incentivize like a private company by using a pricing arrangement to encourage the contractor to reduce costs and increase reliability to make a profit. One way the government can incentivize contractors this way is by PBH. For instance,

> under PBH, an hourly rate is negotiated and the contractor is paid in advance based on the forecasted operational hours for the system. Actual hours are reconciled with projected hours, and overages and shortfalls are either added to or credited from the next period's forecasted amounts. Since the contractor receives funding independent of failures it is then incentivized to overhaul the asset the first time it fails so that it stays in operation as long as possible. (DAU, 2005, pp. 3-24)

This basically encourages the contractor to touch the product as little as possible because the less they touch it, the more money they make. This means that they need to have more support structures to reduce their defective product percents. So, the contractor will develop support processes like:

- repair/replace/overhaul,
- material management,

– engineering and logistics support,
– packaging and shipping, and
– configuration management. (DAU, 2005, pp. 3-26)

All of these activities are designed to improve the production and the product's life cycle support. The private sector incentives its contractors by encouraging them to develop processes that will help them meet the goal and metrics in their contracts. PBH is just one example of how the government can use private-company-style incentives to prompt the contractor to meet the metrics in the contracts. In conclusion, *FAR* Part 12 is the government's guide to applying commercial incentives to its procurement so that the DoD's primary objective can be achieved.

Contractors Creating Their Own Incentives

The possibility of contractors creating their own incentives should be taken seriously, but it is likely to require fundamental reform of existing contractual vehicles to make it a reality. The contractor may propose that if its performance is exemplary that it should be considered for future business. The reputation of an exemplary performer often positively influences prospects for follow-on business, but the award process on a government contract must adhere to the *FAR* guidelines for competitive bidding. The creation of incentives is more likely to occur in subcontracting, where the dollar thresholds are lower and subject to less government scrutiny. However, for optimal performance to occur, contractors should be more proactive in proposing creative incentives because this is likely to leverage organizational competencies to achieve higher performance.

It should also be noted that a contractor may benefit greatly from using the technology gained from the development of one product to produce other related products. This is evident in thousands of examples of "spin-off" products. A program called TOCNET (Tactical Operation Centre Intercommunication System), which is a wireless encrypted LAN, was initially used by the Marines for base communications. Substantial numbers of additional, related products were spun-off from this technology, including a commercially viable product called the Coal Miner's Phone, which allows miners to communicate wirelessly (Jane's, 2010).

Within legal guidelines, a firm is permitted to determine its own cost (managerial) accounting methods. This is an evolving area in which improved metrics and evaluation are being developed as a result of pressure from the government and the marketplace. For instance, a firm may find some means to show fewer inventories than actually in the system. While such a practice is a direct misrepresentation to the investor, taxpayer, and the government, it may be advantageous to the contractor. Additionally, it may lead to lost stock, delayed payments, distortion of stock levels, and the added administrative expense associated with reordering. This is an example of a metric that can be reformed. A company's discretionary ability to determine its cost accounting methods is endangered to some degree, especially given the corporate scandals involving "cooking the books." Recently, a department within a major government contractor came in $35,000 under budget for the approaching end of fiscal year. To discourage the possibility of the government (Congress) interpreting the under-budget situation as an over allocation, the contractor spent the money on office supplies. This is an example of a situation that requires reform on the part of government.

Conclusions

PBL is the latest procurement and sustainment method, and it should be noted that it is a result of an evolutionary process. It has, at the very least, articulated a basic strategy for contract performance and has brought about increased analysis and scrutiny of contractual performance. The improvement of metrics resulting from advances in technology is perhaps the greatest operational mechanism for the contractor to achieve the goals laid out by PBL. The contract itself still dictates the performance and it must adhere to the *FAR*, *DFARS*, and *TINA*. Creative methods beyond the scope of the *FAR* are a highly risky proposition for the contractor and would require reform of contractual vehicles and regulations. Applying the appropriate contractual type to the given program is essential under the current conditions, and it should reflect the applicable point in the product life cycle (Kratz & Buckingham, 2010).

The deployment of PBL may put the government in the uncomfortable position of relinquishing control to the contractor, while still being ultimately responsible for the performance of the contract. Relinquished control on the part of the government leads to higher government-born risk:

> Despite its apparent success, there is an inherent conflict that DoD implementers of PBL often face: the PBL goal of developing long-term partnerships that encourage investment from commercial partners is best achieved through lengthy, guaranteed contracts—but such contracts increase the DoD's risk in an environment that is intended to transfer more risk to the contractor. (Gardner, 2008)

As a result, in order for PBL to be successful, the government needs to appropriately balance risk with the right level of compensation and incentives.

While we have provided a progress report of the work done thus far on this project, a lot more work needs to be done to complete this framework of contractor incentives in PBL environment. For example, the proposed framework needs to be verified through empirical means. This requires field work to include interviews with DoD and contractor personnel, using PBL and focused group discussions to assess the viability and desirability of various types of incentives. Such field work is also essential to identify the behavioral factors that result in the success or failure of various contractor incentives in the PBL environment. This information can then be used to develop strategies and tactics to implement appropriate incentive schemes. Subsequently, models need to be developed to identify exact mechanisms and algorithms to use in determining the specific levels of contractor incentives to use in PBL. Thus, the topic of performance-based (or outcomes-based) logistics and life cycle management and the need to find and use appropriate incentives to maximize performance is a fruitful area of future research, both from an academic and practical viewpoint.

Acknowledgements

This research is partially supported by Grant No. N00244-09-1-0083 from the Fleet Industrial and Supply Center of the US Department of Navy through the Naval Postgraduate School. Constructive comments on an earlier draft of this paper by Steve Geary of Supply Chain Visions significantly improved the quality of this paper.

References

Aguilar, M., Estrada, G.C., & Myers, J.J. (2005). *Decision-support quantitative models for valuing incentives in performance based contracts*. Monterey, CA: Naval Postgraduate School.

Barber, E. (2008). Incentive option for performance-based logistic contracts. *Australian Army Journal*, *5*(1), 115-130.

Beggs, J., Ertel, B., & Jones, M. (2005). *Performance-based logistics perspective*. Booz, Allen and Hamilton.

Berkowitz, D., Gupta, J.N.D., Simpson, J.T., & McWilliams, J.B. (2005). Defining and implementing performance-based logistics in government. *Defense Acquisition Review Journal*, *11*, 255-267.

DAU. (2005). *Performance based logistics: A program manager's product support guide*.

Defense Contract Management. (2010, January 19). *Termination for convenience*. Retrieved from http://guidebook.dcma.mil/25/guidebook_process.htm

DeVries, H.J. (2005). Performance-based logistics: Barriers and enablers to effective implementation. *Defense Acquisition Review Journal*, *11*, 243-253.

Doerr, K., Lewis, I., & Eaton, D.R. (2005). Measurement issues in performance-based logistics. *Journal of Public Procurement*, *5*(2), 164-186.

Edwards, V.J., & Nash, R.C. (2007). A Proposal for a new approach to performance-based services acquisition. *Defense Acquisition Review Journal*, *14*(2), 353-367.

Fowler, R. (2010, March-April). The future of product support. *Defense AT&T*, 16-20.

Gardner, C.P. (2008). *Balancing government risks with contractor incentives in performance-based logistics contracts* (Report No. AFIT/GLM/ENS/08-2). Wright Patterson Base, OH: Air Force Institute of Technology.

Geary, S., & Vitasek, K. (2008). *Performance-based logistics: A contractor's guide to life cycle support management*. Supply Chain Visions.

Gilbreth, A.S., & Hubbard, S. (2008). How to make incentives and award fees work. *Defense Acquisition Review Journal*, *16*(2), 133-145.

Graham, R. (2003, summer). The transformation of contract incentive structures. *Acquisition Review Quarterly*, 235-259.

GSA, DoD, NASA. (2010, January 19). *Federal acquisition regulation (FAR)*. Retrieved from https://www.acquisition.gov/far/html/Subpart%2049_5.html

High stakes in the battle for the world's biggest defence contract. (2010, January 19). The Independent Business. Retrieved from http://www.independent.co.uk/news/business/analysis-and-features/high-stakes-in-the-battle-for-the-worlds-biggest-defence-contract-632715.html

Hildebrandt, G.G. (1998, spring). The use of performance incentives in DoD contracting. *Acquisition Review Quarterly*, 217-234.

Jane's Military Communications. (2010, January 21). *TOCNET*. Retrieved from http://www.janes.com/articles/Janes-Military-Communications/TOCNET-TM-tactical-operations-centre-intercommunication-system-United-States.html

Kim, S-H., Cohen, M.A., & Netessine, S. (2007). Performance contracting in after-sales service supply chains. *Management Science*, *53*(12), 1843-1858.

Kirk, R.L., & DePalma, T.J. (2005). *Performance-based logistics contracts: A basic overview*. Alexandria: CNA.

Klevan, P. (2008, October 27). *Navy success with PBL*. Presentation at the DoD Maintenance Symposium.

KMC/OPI. (2010, January 19). *Acquisition reform initiatives that support PBL*. Retrieved from http://www.pblprograms.com/pdf/2008_AcquisitionReform.pdf

Kratz, L., & Buckingham, B.A. (2010). Achieving outcomes-based life cycle management. *Defense Acquisition Review Journal*, *18*, 45-66.

Lewis, I. (2005). Public management and performance-based logistics in the U.S. Department of Defense. *International Public Management Review*, *6*(2), 116-127.

Mahon, D. (2007, summer). Performance-based logistics: Transforming sustainment. *Journal of Contract Management*, 53-71.

Tremaine, R.L. (2008). Incentive contracts: The attributes that matter most in driving favorable outcomes. *Defense Acquisition Review Journal*, *17*, 217-238.

Visitask. (2010, January 19). *S-curve*. Retrieved from http://www.visitask.com/img/s-curve.jpg

Vitasek, K., & Geary, S. (2008, June). Performance-based logistics redefines Department of Defense procurement. *World Trade*, 62-65.

Vitasek, K., & Murray, S. (2009). *Hitting the target in performance-based logistics*. White Paper. Center for Medical Education, The University of Tennessee.

Vitasek, K., Cothran, J., Geary, S., & Rutner, S. (2006). *Performance-based logistics: The changing landscape in support of contracting*. White Paper. Center for Medical Education, The University of Tennessee.

THIS PAGE INTENTIONALLY LEFT BLANK

Panel #5 – Software Acquisition Strategies and Challenges

Wednesday, May 12, 2010	
11:15 a.m. – 12:45 p.m.	**Chair: Rear Admiral Jerry K. Burroughs, US Navy**, Program Executive Officer, Command, Control, Communications, Computers and Intelligence
	Comparing Acquisition Strategies: Open Architecture versus Product Lines
	Nicholas Guertin, PEO IWS and Paul Clements, Carnegie Mellon University
	The Challenge of Heterogeneously Licensed Systems in Open Architecture Software Ecosystems
	Walt Scacchi, Thomas Alsaugh and Hazel Asuncion, University of California, Irvine
	Exploring Acquisition Strategies for Adopting a Software Product Line
	Lawrence Jones and John Bergey, Carnegie Mellon University

Comparing Acquisition Strategies: Open Architecture versus Product Lines

Nicholas Guertin—Nickolas H. Guertin, PE, received a BS in Mechanical Engineering from the University of Washington and an MBA from Bryant University. He is certified in Program Management and Engineering. Mr. Guertin has worked across a wide range of naval mission areas, including nuclear and conventional ship propulsion, torpedo engineering, and sonar and combat systems development. Mr. Guertin has fifteen years of experience in using open architecture business and technical practices for National Security Systems. Now in PEO for Integrated Warfare Systems, Mr. Guertin leads the transformation to change the business, technical, and cultural practices for how the Navy and Marine Corps develops and fields systems as a coordinated enterprise effort.

Paul Clements—Dr. Paul Clements is a senior member of the technical staff at Carnegie Mellon University's Software Engineering Institute, where he has worked since 1994 in software product line engineering and software architecture documentation and analysis. Clements is the co-author of three practitioner-oriented books about software architecture: *Software Architecture in Practice* (1998, second edition 2003), *Evaluating Software Architectures* (2001), and *Documenting Software Architectures* (2002, second edition 2010). He also co-wrote *Software Product Lines: Practices and Patterns* (2001), and was co-author and editor of *Constructing Superior Software* (1999). Before joining the SEI, he was a senior software engineer at the US Naval Research Laboratory in Washington, DC.

Introduction

An open architecture is a development methodology that employs published, widely accepted standards for defining key interfaces within a system. Systems that are "open" have components that can be provided by different vendors, allowing performance improvements and technology refreshments at a faster pace than "closed" systems. This "open" approach for constructing systems can be augmented by acquisition practices that leverage these "open" technical attributes to facilitate competition. This paper gives an overview of open architecture acquisition approaches and investigates whether open architecture by itself is sufficient to provide the stated goals of rapid fielding, reduced cost, and interoperability among systems. After that, we compare the open architecture approach to another acquisition approach for systems, namely the product line approach. A product line is a set of systems that share a common, managed set of features that satisfy the specific needs of a particular market segment or mission and that are developed from a common set of core assets in a prescribed way (*Software Product Lines*, n.d.). Several US DoD systems acquisitions are currently taking the product line approach. We provide an overview of a various product-line-based acquisition strategies and discuss the relative advantages and disadvantages of the product line approach. We argue that open architecture principles are an essential ingredient of the product line approach for the DoD. Furthermore, the product line methodology consists of a robust set of practices that will generally yield more repeatable results of increased performance and lower risk at minimal cost. The combination of the two approaches will deliver more benefits to the acquisition organization than either approach alone. Finally, we highlight the challenges associated with management of an open product line across multiple providers.

Open Architecture

An open architecture is an architecture that employs open standards for key interfaces within a system (*Open Systems Defined*, n.d.). Because the interfaces conform to publicly documented, consensus-based standards, any competent supplier can provide conforming implementations for any module, allowing the owner of the system to take advantage of competitive bids among suppliers who compete to provide each module.

The following principles characterize a set of business and technical practices that will lead to delivery of increased capabilities in a shorter time-to-field at reduced costs:

- Modular designs with loose coupling and high cohesion that allow for independent acquisition of system components, i.e., composability;
- Continuous design disclosure and appropriate use of intellectual property rights, allowing greater visibility into an unfolding design and flexibility in acquisition alternatives;
- Enterprise investment strategies that maximize reuse of system designs and reduce total ownership costs;
- Enhanced transparency of system design through open peer reviews;
- Competition and collaboration through development of alternative solutions and sources; and
- Analysis to determine which components will provide the best return on investment to open, i.e., which components will change most often due to technology upgrades or parts obsolescence and have the highest associated cost over the lifecycle.

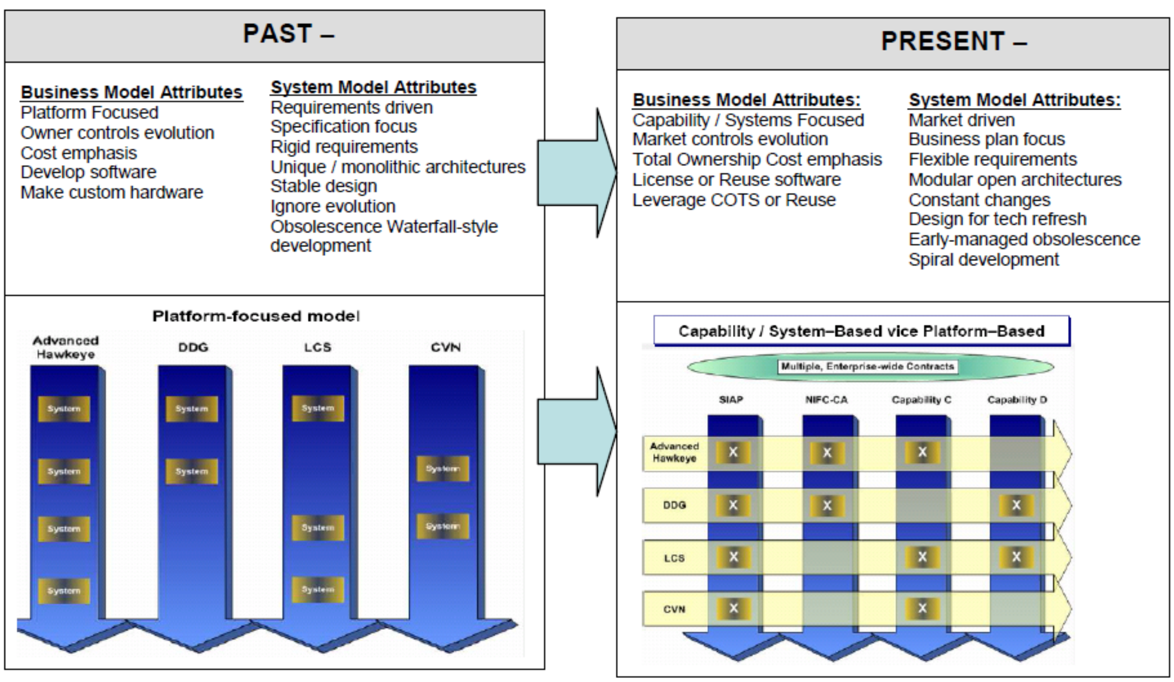


Figure 1. Traditional versus Open Architecture Development Approaches

The need to change the business environment must be the primary factor that drives the technical approach. Accordingly, there are business case decisions to be made about how much investment each principle warrants:

- The use of open standards for key interfaces is a critical aspect of insulating a program from many future cost risks associated with upgrading and establishing some degree of vendor independence. The most important business decisions lie in identifying the "key" interfaces. These typically involve architectural elements encapsulating the most important system behaviors and/or business segments. This principle is highly correlated to the practices of modular design with loose coupling and high cohesion; these help ensure that upgrades and system maintenance can be performed with low cost and schedule risk. Economic benefit is accrued on a system with a multi-year lifespan (i.e., not prototypes or limited production run systems), and components that need to be upgraded or migrated to updated hardware over its lifecycle.

- Continuous design disclosure is especially important for Government acquisitions, even though this was, at one time in the past, looked on as a source of development overhead cost challenges. There are two aspects of design disclosure: contract deliverables and access to the evolving design and development products. This allows the program office to review the evolution of the critical design elements as they evolve, and the ability to exercise data rights on all design related information, even if not a formal deliverable. One of the most common "lessons learned" we have heard is failure to get all the artifacts that are needed to support competition. Formal deliverables should be limited to those things that require a review-comment process or collaboration to ensure design synthesis will yield a result that can be validated against the requirements. All other elements of a system design should be made available to the customer to observe throughout the design process. Electronic access to the design environment and publishing of design artifacts is very low in cost and should not be a cause for cost growth by the developer. This is especially true for systems that will have a long acquisition life, and the design information will need to be made available to subsequent bidders, if system upgrades or maintenance will be competed on a recurring basis (e.g., every five years).
- Strategic reuse is fundamental to a product line approach. Enterprise investment strategies need to be formulated to determine the business basis for those reuse elements. It will likely cost more to make something reusable (additional documentation, commenting, provision for different boundary conditions, etc.) and governing the process of managing collaboratively developed and co-dependent designs is challenging. The current state of practice in many DoD acquisition domains is to build products in which all design elements are tailor-made to specific solutions and few, if any, of the associated products are required to be built for strategic reuse. This business practice is based on minimum emphasis on enterprise reuse, from sponsoring organizations all the way to user communities.
- The *Naval Open Architecture Contract Guidebook* defines a "peer review" as "a refereed, open process used to assess technical approaches proposed by or being used by vendors. An 'independent peer review' includes external membership and is structured to achieve a balanced perspective in which no one organization is dominant." This assessment process normally results in findings or recommendations presented to the decision-maker with the authority and responsibility to select or make the final course of action or decision. These kind of open peer reviews are a technical management construct that has been hard to replicate across a broad continuum and requires lots of communication, purposeful governance, and oversight. Exposing peer competitors to the inner workings of each other's products may require creative intellectual property rights

negotiations in order to get the benefits of peer reviews and create the most innovative and capable products and producers while sustaining a robust marketplace for innovative solutions.

- Development of alternative solutions and sources is a noted weakness of the DoD's acquisition pattern of behavior. A pattern of continuous competition has been proven to establish better pricing and performance. In a recent interview, Dr. Jacques Gansler, former Under Secretary of Defense for Acquisition, Technology and Logistics, stated, "By contrast, whenever we've had competitive sourcing, we get more than a 30 percent cost savings, on average, with higher performance, no matter who wins—and the government most often wins. Competition really pays" (Gansler, n.d.). In order to address this, Congress made specific provisions for requiring competitive prototyping as a major aspect of the *Weapon System Acquisition Reform Act of 2009* (*Summary*, n.d.). In addition, some programs have been able to use a collection of contracting vehicles to establish a framework for continuous competition that gives the program manager additional acquisition choices. There are historical cost references that can be used to justify establishing a second source, especially at the early stages of system development. Having healthy competitive tension at a more granular level throughout the design and integration process has some additional positive, but intangible effects on developer behavior. Most program managers get their best cooperation from their incumbents when there is a full-and-open solicitation on the street.

The value proposition on the OA principles discussed here includes an analysis of how much change will be needed throughout a system lifecycle. Underlying technologies may change faster than others, depending on the market-space from which they come, and the potential demand signal for capability changes by the warfighter or customer need to be addressed. These two dimensions of change need to drive a technology refresh strategy and a capability evolution strategy. These are two sides of the same coin and need to be woven together to form a coherent program plan. However, many programs bent on executing requirements for initial capability fail to address these dynamics. They must also address how their business goals are aligned to the technical architecture, system modularity/coupling/cohesion, design disclosure and data rights analysis, strategic reuse strategies, transparency of system design, the need for a variety of alternative sources, and lifecycle cost models.

Open Architecture and Acquisition

The Navy has extended the work of the DoD Open Systems Joint Task Force (OSJTF) to more comprehensively achieve the desired goals of open architecture as a part of the Naval Open Architecture (NOA) effort. NOA is defined as the confluence of business and technical practices yielding modular, interoperable systems that adhere to open standards with published interfaces. It is the goal of the Naval Open Architecture effort to "field common, interoperable capabilities more rapidly at reduced costs" (*Updated Naval*, n.d.).

The Naval acquisition community is working to adopt these principles. Fully doing so will require a change in technical approach, but that is the easy part. Much harder is to change the business practices, particularly in cross-stakeholder governance, across a wide range of organizations. Government-to-industry relationships can be most effectively changed through new competitively awarded contracts. Changing internal government-to-government business behavior is harder, in that the contract between parties is implied or weak, sometimes in a Memo of Understanding.

The number of programs adopting these principles has been based on two things: cultural barriers and the practical limits of programmatic and technical constraints. The level of adoption has been highly dependent on the drive by individual senior acquisition leaders to change business relationships through steps that break from the long-held pattern of behavior that has been employed in the DoD for many years. Adopting OA principles is a transformational challenge of the highest order.

The Navy and Marine Corps are incorporating OA into selected new-start acquisition or upgrades to existing programs. These programs are implementing open architecture for either new-start acquisitions or upgrades to existing programs where there is a clear business case for opening up the system acquisition and technical characteristics to gain better value and warfighter performance. For new-start acquisitions, there are compelling business cases for ensuring that the design boundaries of the system modules are fully disclosed and work to standards-based methods.

Many programs have adopted aspects of OA behavior, but few have taken a full OA plunge. The Navy Submarine Program has achieved the most compelling example of cost improvements and warfighting performance across the DoD . PEO Subs has spearheaded the practices of OA, specifically the Acoustic Rapid Commercial-off-the-Shelf (COTS) Insertion (A-RCI) and incorporated those methodologies into several other warfighting acquisitions for combat control, including imaging, radar, and others.

Product Lines

A software product line is "a set of software-intensive systems sharing a common, managed set of features that satisfy the specific needs of a particular market segment or mission and that are developed from a common set of core assets in a prescribed way" (*Software Product Lines*, (n.d.).

Software product line practice is a proven and practical approach for software system development, including DoD systems. There are dozens of well-documented cases showing the significant, even order-of-magnitude improvements achieved in terms of cost, time to deployment, and quality (*Catalog*, n.d.). In addition, the international Software Product Line Conference maintains a "Software Product Line Hall of Fame," a collection of exemplary software product line examples that other organizations can emulate; currently, 18 members have been inducted (*Product Line*, n.d.).

Product lines result when builders and acquirers recognize that few systems are unique. This is true for systems acquired by the DoD, systems built by DoD contractors and suppliers, and systems built by industry for private-sector use. Building these systems individually is not good technical or business practice, and in the DoD, it results in expensive rework, unnecessary system duplication, failure to achieve interoperability, and delayed and diminished operational capability. A product line approach exploits the commonality among similar systems, and tremendous cost and schedule improvements and decreased technical risk have also resulted.

At its essence, fielding a product line involves

1. *development or acquisition of core assets*, which are software, document, process, and management artifacts engineered to be re-used;

1. *development or acquisition of products* using those re-usable core assets; and
2. *management* for planning and coordinating core asset and product development.

The development activities can occur in either order (new products are built from core assets, or core assets are extracted from existing products). Often, products and core assets are built in concert with each other. Core asset development has been traditionally called domain engineering. Product development from core assets is often called application engineering. The entire effort is staffed, orchestrated, tracked, and coordinated by management. **Error! Reference source not found.** illustrates this triad of essential activities. The interactions among the symbols indicates not only that core assets are used to develop products, but that revisions to or even new core assets might, and most often do, evolve out of product development. The diagram is neutral about which part of the effort is launched first. In some contexts, already existing products are mined for generic assets—a requirements specification, an architecture, software components, etc.—that are then migrated into the product line's asset base. In other cases, the core assets may be developed or procured for later use in production of products.

Figure 2. The Essential Activities of a Software Product Line

Product lines employ planned, strategic reuse across a family of products to produce savings in the following areas each time a product is ordered:

- Requirements. Most of the requirements are common with earlier systems, and so can be used. Requirements analysis is saved. Feasibility is assured.
- Architectural design. An architecture for a software system represents a large investment in the form of time from the organization's most talented engineers. The quality goals for a system—its performance, reliability, modifiability, etc.—are largely allowed or precluded once the architecture is in place. For a new product birthed from the product line, this most important design step is already done and need not be repeated.
- Components. Not only code can be reused, but also the internal designs for the architectural components are reused from system to system, as is the

documentation of those designs. Data structures and algorithms are saved and need not be reinvented.

- Modeling and analysis. One product line organization reports that one of the major headaches associated with the kinds of systems they build—namely, real-time distributed—has all but vanished. When they field a new product in their product line, they have extremely high confidence that the timing problems have been worked out, and the bugs associated with distributed computing—synchronization, network loading, absence of deadlock—have been eliminated because their performance models have been validated across the entire family (Bergey & Jones, 2010).
- Testing. Test plans, test processes, test cases, test data, test harnesses, and the communication paths required to report and fix problems are already available.
- Planning. Budgets and schedules can be informed or reused from previous projects, and they're much more reliable.
- Processes. Configuration control boards, configuration management tools and procedures, management processes, and the overall software development process are in place, have been used before, and are robust, reliable, and responsive to the organization's special needs.
- People. Because of the commonality of the systems, personnel can be fluidly transferred among projects as required. Their expertise is applicable across the entire line. When operational needs call for a rapid deployment of a system, the right supplier personnel can be brought to bear immediately.
- Training materials. Since systems in a product line have a common look and feel, training is simplified and training materials apply across the family.

These reuse opportunities lead to the advantages touted for a product line approach to software system development, which include:

- Reduced time to deployment. Turning out a new product in the product line is more akin to generation and integration, rather than ground-up coding. Cummins, Inc., reports that systems that used to take a year to complete now can be turned out in about a week (Clements & Northrop, 2003).
- Reduced cost. For example, products in the National Reconnaissance Office’s Control Channel Toolkit product line cost approximately 10% of what they otherwise would have (Clements, Cohen, Donohoe & Northrop, 2001).
- Increased productivity. For example, Cummins estimates that they are now turning out *fourteen times* the number of products they were before, while using only two thirds the software resources, for a productivity gain of 2,100% (McGregor & Clements, n.d.).
- Higher quality. Product lines enhance quality. Each new system takes advantage of all of the defect elimination in its forebears; developer and customer confidence both rise with each new instantiation. The more complicated the system, the higher the payoff for solving the vexing performance, security, and availability problems.
- Simplified training. Users competent in one member of the product line are generally competent to use others.

Product Lines and Acquisition

Product line practice is gaining more and more traction every year in the DoD, gaining a foothold and proving its merits in small systems to high-visibility systems of systems. DoD organizations that have adopted the software product line approach include:

- the Navy's Program Executive Office for Integrated Warfare Systems (PEO IWS) (**Error! Reference source not found.**) (Emery, n.d.),
- the National Reconnaissance Office (Clements et al., 2001),
- the Naval Undersea Warfare Center (NUWC) (Cohen, Dunn & Soule, 2002),
- the Army's Technical Applications Program Office (TAPO) (Clements & Bergey, 2005),
- the Army's Live Training Transformation effort (*Live Training*, n.d.),
- The Navy's PEO for Submarine's products from the Submarine Warfare Federated Tactical System family of systems (**Error! Reference source not found.**)

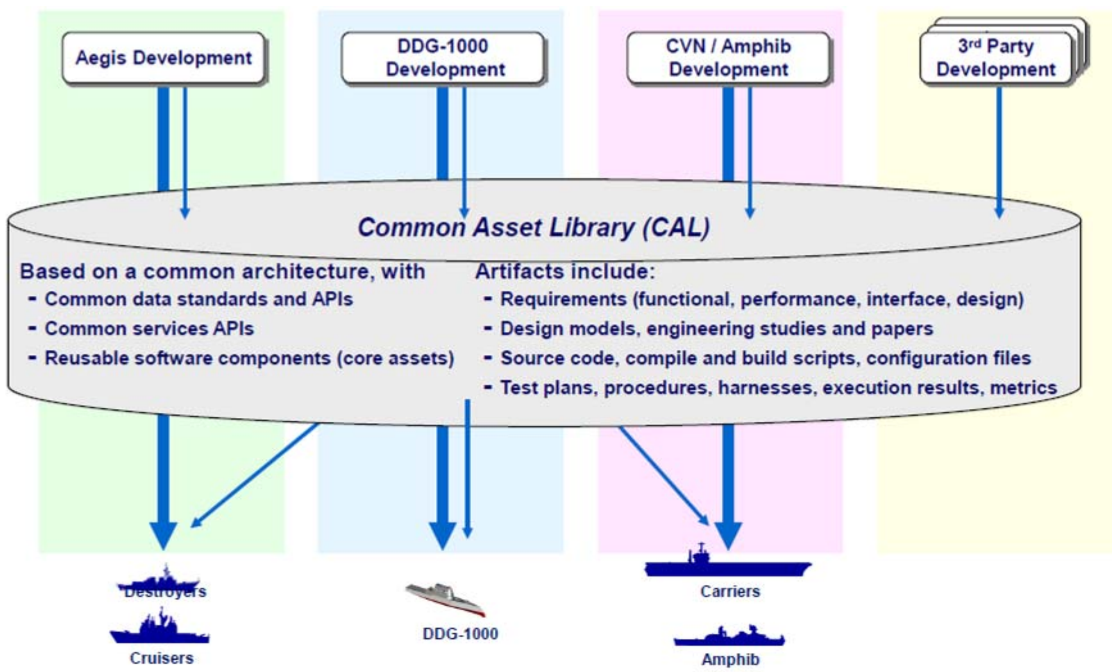


Figure 3. PEO IWS Product Line Approach for Surface Combat Systems

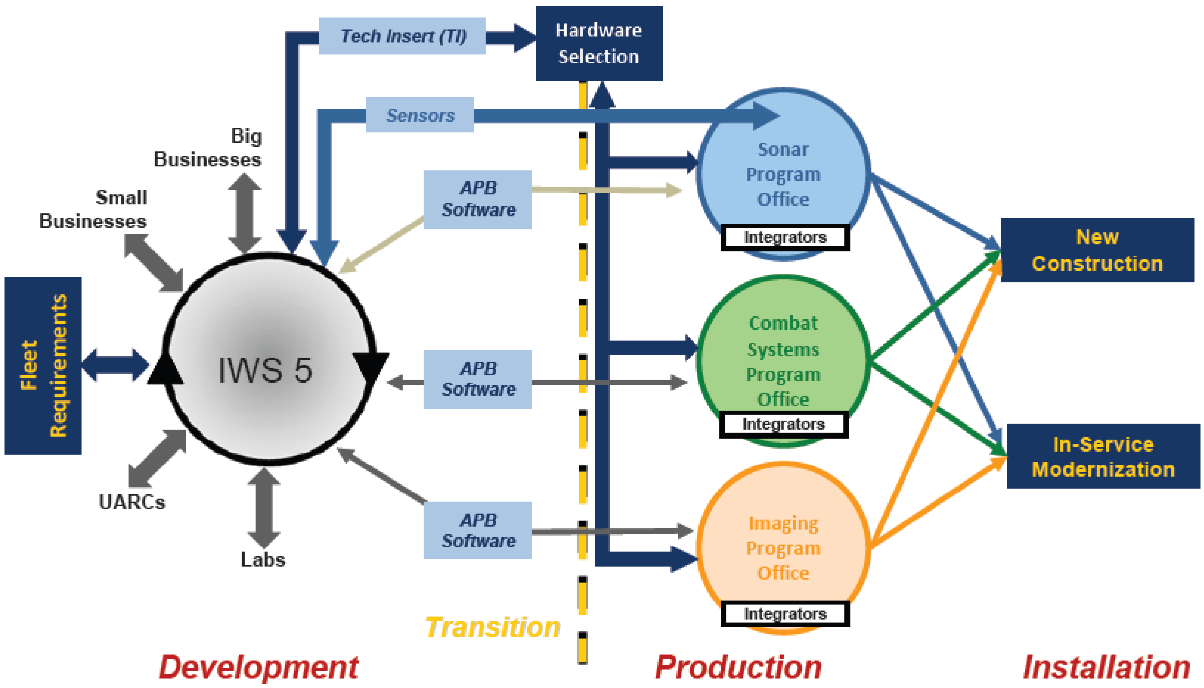


Figure 4. PEO Submarines SWFTS Model for Cross-platform Product Commonality

In addition a growing number of commercial DoD contractors are gravitating to software product lines. The Software Engineering Institute maintains a catalog of software product line experience reports published in the open literature; that catalog currently includes 54 examples (*Catalog*, n.d.).

There are three overall product line acquisition approaches (Bergey & Jones, 2010):

1. The government can commission a supplier to develop a specific product (or products) using the supplier's own proprietary product line. This strategy involves acquiring products directly from a supplier who has an existing product line and a demonstrated capability to build products in the domain of interest. An example of this approach is found in Jensen (2007).

3. The government can commission a government organization to develop a product line production capability and build specific products. This strategy involves acquiring a completely government-owned product line using the in-house capabilities of a designated government acquisition organization. An example of this approach is found in *Live Training Transformation (LT2)* (n.d.).

4. The government can commission a supplier to develop a product line production capability and perform integration of products from other vendors into the production line. This strategy involves acquiring a complete product line production capability and developing derivative products through contracting with one or more suppliers. An example of this approach is found in Clements et al. (2001).

Major challenges include the fact that the DoD's acquisition policies and infrastructure are still largely predicated on acquiring "one-of-a-kind" stove-piped systems, and no institutionalized means exist for funding the development and sustainment of a product line across multiple programs. Nevertheless, successful DoD product lines are being created by acquisition authorities with vision and foresight enough to overcome the difficulties and reap the benefits.

Comparing Acquisition Approaches

A product line approach can only be fruitfully applied in the context of building a family of systems, whereas an open architecture approach works for even a single system that evolves over time.   In a context in which *both* are applicable, how do they compare?

Cost. Both approaches promise lower cost. Open architecture achieves its cost savings by engendering and facilitating competition among suppliers. However, crafting of a competitive market out of a closed and vendor-locked set of business relationships has been a major challenge in the past. Designing an architecture to put into place separately acquirable elements requires thorough systems engineering and marketplace awareness. The goal is to foment a true competition in a situation in which there is a high likelihood that the incumbent could be the only possible winner by dint of long involvement with the legacy system. Meeting this goal is a business and engineering challenge, but failure amounts to leaving in place an unassailable barrier to entry by new suppliers, who may not be able to provide the right technical products or (even if they are) not be able to undercut the price at all. The product line approach achieves its cost savings by amortizing the cost of the core assets across all of the products that use them. Product line approaches have demonstrated repeatable per-product cost savings of 50% (Cohen et al., 2002) to 67% (Clements & Bergey, 2005) to 90% (Clements et al., 2001). The more general open architecture approaches have demonstrated savings up at this level, but with lower consistency. For example, the A-RCI program achieved a 5:1 estimated cost savings over a ten-year period (Boudreau, 2006). Savings in an open architecture approach remain roughly constant over the number of products, whereas savings in the product line approach increase with the number of products. In product line development, one source of cost savings is higher productivity among the developers. Developer productivity in a product line context has been shown to increase by 400% (Toft, Coleman & Ohta, 2000) to 500% (*Catalog*, n.d.) to 2,100% (McGregor & Clements, n.d.).

Time to delivery. Open architecture approaches achieve reduced time to delivery by fostering enterprise reuse and competition among vendors to bring greater innovation in product development methodologies. Product line approaches achieve reduced time to delivery by pre-positioning the core assets required to produce the next product (or next version of a product). The A-RCI project, the ability to take robust solutions from the science and technology community and integrate them into tactical sonar system in two years or less, a process that would have taken five years or more in the legacy framework. Product line approaches have been shown to reduce time to delivery by 50% (McGregor & Clements, n.d.) to 60% (Jensen, 2007) to 67% (Toft et al., 2000) to over 90% (Clements & Northrop, 2003; *Catalog*, n.d.).

Elimination of duplicate effort. The DoD suffers from a plethora of almost-alike systems, developed in isolation from each other. In the US alone, over 80 companies, universities, and government organizations are actively developing one or more of some 200 UAV designs (*UAV Forum*, n.d.). In 2004, the General Accounting Office was able to identify 2,274 separate DoD business systems (but nobody knows the true number), a waste of billions of dollars (FedSmith.com, n.d.). In the vast majority of cases, such systems are all developed and maintained separately, with poor or no acquisition interoperability among them. There is no repeatable or systematic means to take advantage of the commonality of these systems and apply common reusable components or features as a standard practice. Building and maintaining one system at a time, compared to a proven product line approach, is a process laden with systemic inefficiency, stretching development and sustainment budgets to the limit and leaving little left over to work on imaginative new solutions. New software development reuse efforts, where attempted, are ad hoc, repository based, and often devolve into a clone-and-own effort. Open architecture approaches do not directly address the problem of duplication (there may be several open but duplicate implementations that are not strategically or financially aligned), whereas the product line approach gains its benefits by exploiting situations in which duplication would otherwise occur.

Higher quality. Higher quality results from an OA approach through technical practices such as hardware/software independence, modularity with loose coupling and high cohesion, integrability, upgradability and business practices such as strategic reuse, especially the healthy pressure of competition for component development as well as for system integration. Higher quality results from the PL approach because errors wrung out of one system are automatically wrung out of other systems in the same product line. In product line development, defects have been shown to drop by 50% (Pronk, 1999) 90% (Clements et al., 2001) to 96% (Toft et al., 2000).

Open Architecture and Product Lines Together

While the two approaches differ in some fundamental ways, happily there is no reason why they cannot work together. In fact, the two in combination might represent a "perfect storm" of acquisition leverage that can systematically reduce cost, increase performance, and drive down risk. The ideal acquisition occurs when both product lines and open approaches are applicable in the same acquisition context. The focus of combining the two approaches lies in the architecture, but the challenge to achieving it lies in the governance of the DoD acquisition community.

The architecture of a product line is one of its most important core assets, providing the blueprint for how every product will be assembled and the parts (software components and supporting artifacts) it will comprise. Interfaces of those parts are critical to the success of the product line's architecture, for only by mixing and matching instances of components suitable for different products can the product line strategy work. Hence, product line architectures *are* open architectures, in a strict technical sense: they have "published, accepted interfaces to components "that can be provided by different vendors." Whether a product line architecture is an open architecture in the business sense (in other words, whether the components for core assets and products really do come from different vendors) is a matter of business policy within the organization that owns the product line. Some certainly are. For example, Nokia's product line for mobile phones is open outside Nokia, allowing external companies to use Nokia's core asset base to build their own phone products (Van der Linden, Schmid & Rommes, 2007). Hewlett Packard's product line for computer peripheral devices is open across widely disparate organizations within Hewlett Packard (Toft et al., 2000).

An acquisition combining the two approaches could employ strategy #3 in Section □□, overlaid with a requirement that the architecture be open with publicly defined interfaces for the key elements. Here, the government commissions one or more suppliers to develop a product line production capability and build specific products. The production capability would include the architecture, openly defined; populating the architecture with components applicable across the defined scope of the product line would be awarded on the basis of open competition.

Neither approach embodies unsolved technical challenges. The main hurdle for both is in the domain of management and changing the way that organizations (government and private) do business. As Machiavelli said, "There is nothing more difficult to take in hand, more perilous to conduct, or more uncertain in its success, than to take the lead in the introduction of a new order of things." The Defense Research and Engineering "imperatives" (*DDR&E Imperatives*, n.d.) are as follows:

- Accelerate delivery of technical capabilities to win the current fight,
- Prepare for an uncertain future,
- Reduce the cost, acquisition time and risk of our major defense acquisition programs, and
- Develop world-class science, technology, engineering, and mathematics capabilities for the DoD and the nation.

These imperatives speak to a critical need for bold new ways to acquire and field systems for the warfighter. Product line engineering and open architecture together promise the kind of outcomes necessary to address DoD needs.

Product lines, together with open architecture methodologies have great potential in the DoD to unlock opportunities for innovation, reduced risk, improve response to warfighter needs, and reduce costs. However, this combined approach will require fundamental change in program office behavior, acquisition leadership, resource community communication, warfighter interaction, and, most importantly, in business practices. Moving out of vendor-locked expensive business relationships to bring access to affordable innovation and flexibility requires a fundamentally different technical and business approach. The best method to change government-industry business relationships is by writing the desired behavior into the contract—a gradual, but achievable change process. Changing internal government to government business behavior is harder, in that the contract between parties is implied or weak. Program officers that do strategic reuse and combine forces with another program to improve enterprise business value are making a bold move. The reward mechanisms for acting on the best value for the Enterprise are not well established. Coordinating budgets and aligning schedules across different resource sponsor offices is a daunting challenge that needs further exploration, new methods, bold leadership, and sustained and steady hard work.

References

Bergey, J., & Jones, L. (2010). Exploring acquisition strategies for adopting a software product line approach. In *Proceedings of the Seventh Annual Acquisition Research Symposium*. Monterey, CA: Naval Postgraduate School.

Boudreau, M. (2006, October30). *Acoustic rapid COTS insertion: A case study in spiral development*. Naval Postgraduate School. Retrieved from http://www.acquisitionresearch.org/_files/FY2006/NPS-PM-06-041.pdf

Brownsword, L., & Clements, P.A. (1996, October). *Case study in successful product line development* (Technical Report CMU/SEI-96-TR-016/ESC-TR-96-016).

Catalog of software product lines. (n.d.). Retrieved from http://www.sei.cmu.edu/productlines/casestudies/catalog/

Clements, P., & Bergey, J. (2005, September). *The U.S. Army's common avionics architecture system (CAAS) product line: A case study* (Technical Report CMU/SEI-2005-TR-019).

Clements, P., & Northrop, L. (2003). *Software product lines: Practices and patterns*. Boston: Addison Wesley.

Clements, P., Cohen, S., Donohoe, P., & Northrop, L. (2001, October). *Control channel toolkit: A software product line case study* (Technical Report CMU/SEI-2001-TR-030).

Cohen, S., Dunn, E., & Soule, A. (2002, September). *Successful product line development and sustainment: A DoD case study* (CMU/SEI-2002-TN-018).

DDR&E imperatives. (n.d.). Retrieved from http://www.dod.mil/ddre/index.html

Emery, K. (n.d.). *Surface Navy combat systems engineering strategy*. Retrieved from http://www.nps.edu/Academics/Institutes/Meyer/docs/IWS%20OA%20Briefing%20to%20NPS%20March0410.pdf

FedSmith.com. (n.d.). *Billions wasted…* Retrieved from http://www.fedsmith.com/article/313/billions-wasted-dod-because-duplicate-business-systems.html

Gansler, J. (n.d.). *"Global war" on contractors must stop*. Retrieved from http://blog.executivebiz.com/jacques-gansler-global-war-on-contractors-must-stop/7105

Jensen, P. (2007, September). Experiences with product line development of multi-discipline analysis software at Overwatch Textron Systems. In *Proceedings, SPLC 2007*. Kyoto: IEEE Computer Society.

Live training transformation (LT2). (n.d.). Retrieved from https://www.fbo.gov/index?s=opportunity&mode=form&tab=core&id=ba606e9dbff7c6d266cf16f2fd2a1bee

McGregor, J., & Clements, P. (n.d.). Better, faster, cheaper—pick any three. *MIT Sloan Management Review*, submitted.

Open systems defined—terms and defiitions. (n.d.). Retrieved from http://www.acq.osd.mil/osjtf/termsdef.html

Product line hall of fame. (n.d.). Retrieved from http://www.splc.net/fame.html

Pronk, B. (1999). Medical product line architectures. In *Proceedings of the TC2 First Working IFIP Conference on Software Architecture (WICSA1)* (pp. 357-367).

Software product lines. (n.d.). Retrieved from http://www.sei.cmu.edu/productlines/start/index.cfm

Summary of the Weapon Systems Acquisition Reform Act of 2009. (n.d.). Retrieved from http://levin.senate.gov/newsroom/release.cfm?id=308525

Toft, P., Coleman, C., & Ohta, J. (2000). A cooperative model for cross-divisional product development for a software product line. In P. Donohoe (Ed.), *Proceedings SPLC1*. Kluwer Academic Publishers.

UAV forum, librarian's desk. (n.d.). Retrieved from http://www.uavforum.com/library/librarian.htm

Updated naval OA strategy for FY 2008. (n.d.). Retrieved from https://acc.dau.mil

Van der Linden, F., Schmid, K., & Rommes, E. (2007). *Software product lines in action*. Springer, Ch. 12.

The Challenge of Heterogeneously Licensed Systems in Open Architecture Software Ecosystems

Walt Scacchi—Walt Scacchi is a senior research scientist and research faculty member at the Institute for Software Research, University of California, Irvine. He received a PhD in information and computer science from UC Irvine in 1981. From 1981 to 1998, he was on the faculty at the University of Southern California. In 1999, he joined the Institute for Software Research at UC Irvine. He has published more than 150 research papers and has directed 45 externally funded research projects. In 2007, he served as General Chair of the 3rd IFIP International Conference on Open Source Systems (OSS2007), Limerick, IE. In 2010, he chaired the Workshop on the Future of Research in Free and Open Source Software, Newport Beach, CA, for the Computing Community Consortium and the National Science Foundation. He also serves as Co-chair of the Doctoral Consortium at the 6th IFIP International Conference on Open Source Systems (OSS2010), Notre Dame, IN.

Thomas Alspaugh—Thomas Alspaugh is adjunct professor of computer science at Georgetown University and visiting researcher at the Institute for Software Research at UC Irvine. His research interests are in software engineering and software requirements. Before completing his PhD, he worked as a software developer, team lead, and manager in industry and as a computer scientist at the Naval Research Laboratory on the Software Cost Reduction project, also known as the A-7E project.

Hazel Asuncion—Hazel Asuncion is a postdoctoral scholar at the Institute for Software Research. Starting Fall 2010, she will be an assistant professor in the Department of Computer Science at the University of Washington, Bothell. Her research interests focus on traceability, process workflows, software system architectures, and their interrelationships.

Abstract

The role of software ecosystems in the development and evolution of open architecture systems has received insufficient consideration. Such systems are composed of heterogeneously licensed components, open source or proprietary or both, in an architecture in which evolution can occur by evolving existing components or by replacing them. But this may result in possible license conflicts and organizational liability for failure to fulfill license obligations. We have developed an approach for understanding and modeling software licenses, as well as for analyzing conflicts among groups of licenses in realistic system contexts and for guiding the acquisition, integration, or development of systems with open-source components in such an environment. This work is based on empirical analysis of representative software licenses and heterogeneously licensed systems, and collaboration with researchers in the legal world. Our approach provides guidance for achieving a “best-of-breed” component strategy while obtaining desired license rights in exchange for acceptable obligations.

Introduction

A substantial number of development organizations are adopting a strategy in which a software-intensive system is developed with an *open architecture* (OA) (Oreizy, 2000), whose components may be open source software (OSS) or proprietary with open application programming interfaces (APIs). Such systems evolve not only through the evolution of their individual components, but also through replacement of one component by another, possibly from a different producer or under a different license. With this approach, the organization becomes an integrator of components largely produced elsewhere that are interconnected through open APIs as necessary to achieve the desired result. An OA development process results in an ecosystem in which the integrator is influenced from one direction by the goals, interfaces, license choices, and release cycles of the component producers, and in another direction by the needs of its consumers. As a result, the software components are reused more widely, and the resulting OA systems can achieve reuse benefits such as reduced costs, increased reliability, and potentially increased agility in evolving to meet changing needs. An emerging challenge is to realize the benefits of this approach when the individual components are *heterogeneously licensed*, each potentially with a different license rather than a single OSS license as in uniformly licensed OSS projects, or a single proprietary license when acquired from a vendor employing a proprietary development scheme.

This challenge is inevitably entwined with the software ecosystems that arise for OA systems. We find that an OA software ecosystem involves not only organizations and individuals producing and consuming components, and supply paths from producer to consumer, but also:

- the OA of the system(s) in question,
- the open interfaces met by the components,
- the degree of coupling in the evolution of related components, and
- the rights and obligations resulting from the software licenses under which various components are released, that propagate from producers to consumers.

An example software ecosystem is portrayed in Figure 1.

In order to most effectively use an OA approach in developing and evolving a system, it is essential to consider this OA ecosystem. An OA system draws on components from proprietary vendors and open source projects. Its architecture is made possible by the existing general ecosystem of producers, from which the initial components are chosen. The choice of a specific OA begins a specialized software ecosystem involving components that meet (or can be shimmed to meet) the open interfaces used in the architecture. We do not claim this is the best or the only way to reuse components or produce systems, but it is an ever more widespread way. In this paper, we build on previous work on heterogeneously licensed systems (HLSs) (German & Hassan, 2009; Alspaugh & Scacchi, 2008; Alspaugh, Asuncion & Scacchi, 2009a, May) by examining the role of component licenses in OA software ecosystems and how OA development affects and is affected by software ecosystems.

A motivating example of this approach is the Unity game development tool, produced by Unity Technologies (Unity Technologies, 2008), which can be used to create game-based virtual worlds for training applications. Its license agreement, which we quote below, lists eleven distinct licenses and indicates the tool is produced, apparently using an OA approach, using at least 18 externally produced components or groups of components:

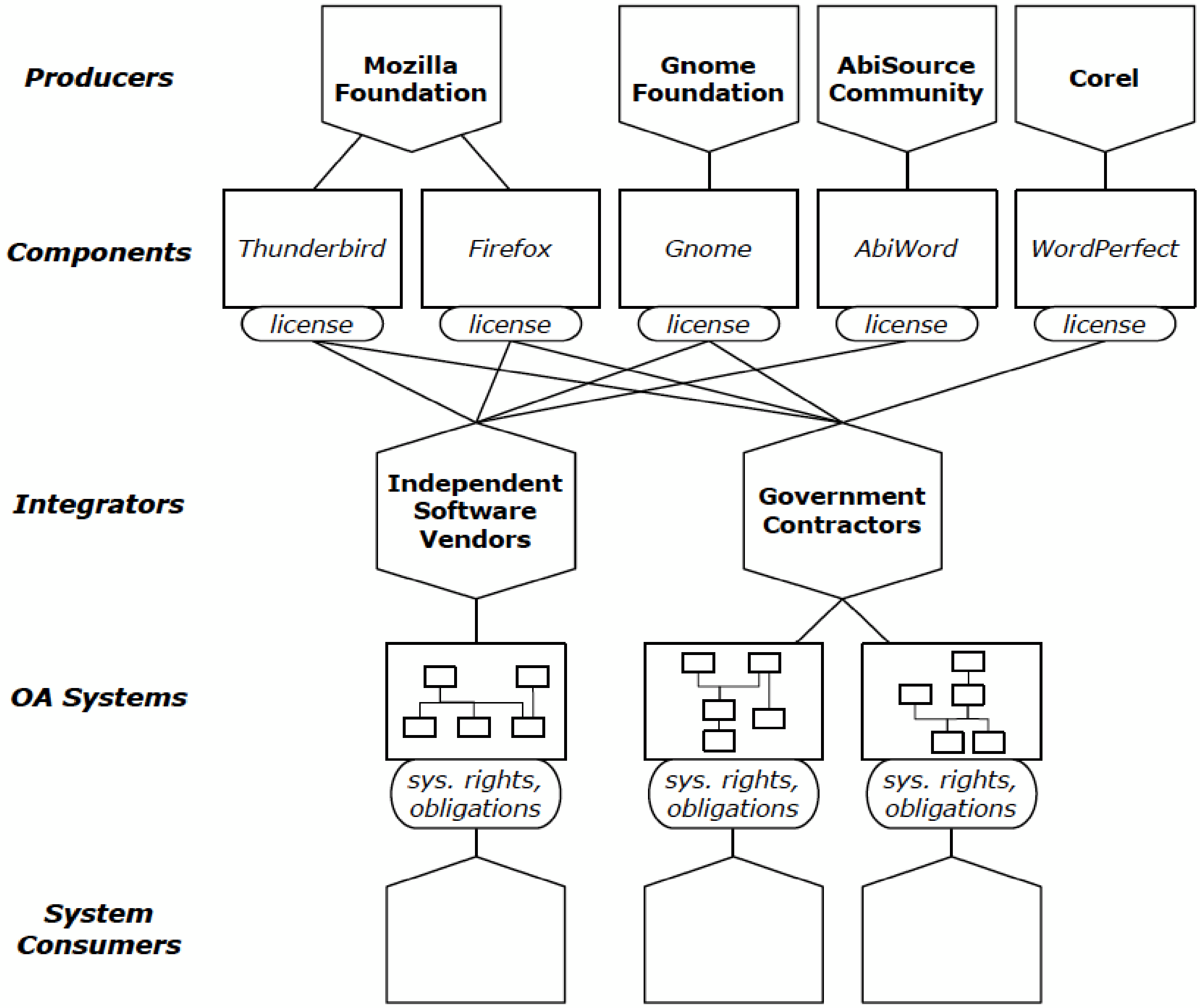


Figure 1. An Example of a Software Ecosystem in which OA Systems Are Developed

1. The Mono Class Library, Copyright 2005-2008 Novell, Inc.
5. The Mono Runtime Libraries, Copyright 2005-2008 Novell, Inc.
6. Boo, Copyright 2003-2008 Rodrigo B. Oliveira.
7. UnityScript, Copyright 2005-2008 Rodrigo B. Oliveira.
8. OpenAL cross platform audio library, Copyright 1999-2006 by authors.
9. PhysX physics library. Copyright 2003-2008 by Ageia Technologies, Inc.
10. libvorbis. Copyright (c) 2002-2007 Xiph.org Foundation.
11. libtheora. Copyright (c) 2002-2007 Xiph.org Foundation.
12. zlib general purpose compression library. Copyright (c) 1995-2005 Jean-loup Gailly and Mark Adler.
13. libpng PNG reference library.

14. jpeglib JPEG library. Copyright (C) 1991-1998, Thomas G. Lane.
15. Twilight Prophecy SDK, a multi-platform development system for virtual reality and multimedia. Copyright 1997-2003 Twilight 3D Finland Oy Ltd.
16. dynamic bitset, Copyright Chuck Allison and Jeremy Siek 2001-2002.
17. The Mono C# Compiler and Tools, Copyright 2005-2008 Novell, Inc.
18. libcurl. Copyright (c) 1996-2008, Daniel Stenberg <daniel@haxx.se>.
19. PostgreSQL Database Management System.
20. FreeType. Copyright (c) 2007 The FreeType Project (www.freetype.org).
21. NVIDIA Cg. Copyright (c) 2002-2008 NVIDIA Corp.

An OA system can evolve by a number of distinct mechanisms, some of which are common to all systems and others of which are a result of heterogeneous component licenses in a single system.

By component evolution—One or more components can evolve, altering the overall system's characteristics (for example, upgrading and replacing the Firefox Web browser from version 3.5 to 3.6).

By component replacement—One or more components may be replaced by others with different behaviors but the same interface, or with a different interface and the addition of shim code to make it match (for example, replacing the AbiWord word processor with either Open Office or MS Word).

By architecture evolution—The OA can evolve, using the same components but in a different configuration, altering the system characteristics. For example, as discussed in Section 4, changing the configuration in which a component is connected can change how its license affects the rights and obligations for the overall system. This could arise when replacing email and word processing applications with web services like Google Mail and Google Docs.

By component license evolution—The license under which a component is available may change (for example, when the license for the Mozilla core components was changed from the Mozilla Public License (MPL) to the current Mozilla Disjunctive Tri-License), or the component may be made available under a new version of the same license (for example, when the GNU General Public License (GPL) version 3 was released).

By a change to the desired rights or acceptable obligations—The OA system's integrator or consumers may desire additional license rights (for example, the right to sublicense in addition to the right to distribute) or no longer desire specific rights or the set of license obligations they find acceptable may change. In either case, the OA system evolves, whether by changing components, evolving the architecture, or by other means, to provide the desired rights within the scope of the acceptable obligations. For example, they may no longer be willing or able to provide the source code for components within the reciprocality scope of a GPL-licensed module.

The interdependence of integrators and producers results in a co-evolution of software within an OA ecosystem. Closely coupled components from different producers must evolve in parallel in order for each to provide its services, as evolution in one will typically require a matching evolution in the other. Producers may manage their evolution with a loose coordination among releases, for example like that between the Gnome and Mozilla organizations. Each release of a producer component creates a tension through the ecosystem relationships with consumers and their releases of OA systems using those components, as integrators accommodate the choices of available, supported components with their own goals and needs. As discussed in our previous work (Alspaugh et al., 2009a, May), license rights and obligations are manifested at each component interface then mediated through the OA of the system to entail the rights and corresponding obligations for the system as a whole. As a result, integrators must frequently re-evaluate the OA system rights and obligations. In contrast to homogeneously licensed systems, *license change across versions is a characteristic of OA ecosystems*, and architects of OA systems require tool support for managing the ongoing licensing changes.

We propose that such support must have several characteristics. It must rest on a license structure of rights and obligations (section 5), focusing on obligations that are enactable (it can be put into practice) and testable. For example, many OSS licenses include an obligation to make a component's modified code public, and whether a specific version of the code is public at a specified Web address is both enactable and testable. In contrast, the GPL v.3 provision "No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty" is not enactable in any obvious way, nor is it testable—how can one verify what others deem?

- It must take account of the distinctions between the design-time, build-time, and distribution-time architectures (sections 4, 5, 6) and the rights and obligations that come into play for each of them.
- It must distinguish the architectural constructs significant for software licenses and embody their effects on rights and obligations (section 4).
- It must define license architectures (section 6).
- It must provide an automated environment for creating and managing license architectures. We are developing a prototype that manages a license's architecture as a view of its system architecture (Alspaugh et al., 2009a, May).
- Finally, it must automate calculations on system rights and obligations so that they may be done easily and frequently, whenever any of the factors affecting rights and obligations may have changed (Section 7).

In the remainder of this paper, we survey some related work (section 2), provide an overview of OSS licenses and projects (section 3), define and examine characteristics of open architectures (section 4), introduce a structure for licenses (section 5), outline license architectures (section 6), and sketch our approach for license analysis (section 7). We then close with a discussion addressing how our software license and analysis scheme relates to software products lines and to specification of software system security requirements (section 8) before stating our conclusions (section 9).

Related Work

It has been typical until recently for each software or information system to be designed, built, and distributed under the terms of a single proprietary or OSS license, with all its components homogeneously covered by that same license. The system is distributed with sources or executables bearing copyright and license notices, and the license gives specific rights while imposing corresponding obligations that system consumers (whether external developers or users) must honor, subject to the provisions of contract and commercial law. Consequently, there has been some very interesting study of the choice of OSS license for use in an OSS development project, and its consequences in determining the likely success of such a project.

Brown and Booch (2002) discuss issues that arise in the reuse of OSS components, such as interdependence (via component interconnection at design-time or linkage at build-time or run-time) causing functional changes to propagate and versions of the components evolving asynchronously, giving rise to co-evolution of interrelated code in the OSS-based systems. If the components evolve, the OA system itself is evolving. The evolution can also include changes to the licenses, and the licenses can change from component version to version (cf. Footnote 1).

Legal scholars have examined OSS licenses and how they interact in the legal domain but not in the context of HLSs (Fontana et al., 2008; Rosen, 2005; St. Laurent, 2004). For example, Rosen surveys eight OSS licenses and creates two new ones written to professional legal standards. He examines interactions primarily in terms of the general categories of reciprocal and non-reciprocal licenses, rather than in terms of specific licenses. However, common to this legal scholarship is an approach that analyzes the interaction among licenses on a pair-wise or interlinked components basis. This analysis scheme means that if system A has OSS license of type X, system B has a licenses of type Y, and system C has license of type Z, then license interaction (matching, subsumption, or conflicting constraints) is determined by how A interacts with B, B with C, and A with C. This follows from related legal scholarship (e.g., Burk, 1998) that brought attention to problems of whether or not intellectual property rights apply depending on how the systems (or components) are interlinked (cf., German and Hassan, 2009). We similarly adopt this approach in our analysis efforts.

Stewart, Ammeter, and Maruping (2006) conducted an empirical study to examine whether license choice is related to OSS project success, finding a positive association following from the selection of business- friendly licenses. Sen, Subramaniam, and Nelson in a series of studies (2007 & 2009) similarly find positive relationships between the choice of a OSS license and the likelihood of both successful OSS development and adoption of corresponding OSS systems within enterprises. These studies direct attention to OSS projects that adopt and identify their development efforts through use of a single OSS license. However, there has been little explicit guidance on how best to develop, deploy, and sustain complex software systems when heterogeneously licensed components are involved, and thus multiple OSS and proprietary licenses may be involved. Ven and Mannaert (2008); Tuunanen, Koskinen, and Karkkainen (2009); and German and Hassan (2009) are recent exceptions.

Ven and Mannaert discuss the challenges faced by independent software vendors developing an HLS. They focus on the evolution and maintenance of modified OSS components. Tuunanen, Koskinen, and Karkkainen exemplify most work to date on HLSs, in that they focus on reverse engineering and recovery of individual component licenses on existing systems, rather than on guiding HLS design to achieve and verify desired license outcomes. Their approach does not support the calculation of HLS virtual licenses. German and Hassan model a license as a set of grants, each of which has a set of conjoined conditions necessary for the grant to be given. They analyze interactions between pairs of licenses in the context of five types of component connection. They also identify twelve patterns for avoiding license mismatches, found in an empirical study of a large group of OSS projects, and characterize the patterns using their model. Our license model extends German and Hassan's to address semantic connections between obligations and rights we find through a textual analysis of OSS licenses.

Other previous work examined how best to align acquisition, system requirements, architectures, and OSS components across different software license regimes to achieve the goal of combining OSS with proprietary software that provide open APIs when developing a composite "system of systems" (Scacchi & Alspaugh, 2008). This is particularly an issue for the US Federal Government in its acquisition of complex software systems subject to *Federal Acquisition Regulations (FARs)* and military service- specific regulations. HLSs give rise to new functional and non-functional requirements that further constrain what kinds of systems can be built and deployed, as well as recognizing that acquisition policies can effectively exclude certain OA configurations while accommodating others, based on how different licensed components may be interconnected.

Open Source Software

Traditional proprietary licenses allow a company to retain control of software it produces and to restrict the access and rights that outsiders can have. OSS licenses, on the other hand, are designed to encourage sharing and reuse of software, and grant access and as many rights as possible. OSS licenses are classified as academic or reciprocal. *Academic OSS licenses* such as the Berkeley Software Distribution (BSD) license, the Massachusetts Institute of Technology license, the Apache Software License, and the Artistic License grant nearly all rights to components and their source code and impose few obligations. Anyone can use the software, create derivative works from it, or include it in proprietary projects. Typical academic obligations are simply to not remove the copyright and license notices.

Reciprocal OSS licenses take a more active stance towards sharing and reusing software by imposing the obligation that reciprocally licensed software not be combined (for various definitions of "combined") with any software that is not in turn also released under the reciprocal license. The goals are to increase the domain of OSS by encouraging developers to bring more components under its aegis and to prevent improvements to OSS components from vanishing behind proprietary licenses. Example reciprocal licenses are GPL, the Mozilla Public License (MPL), and the Common Public License.

Both proprietary and OSS licenses typically disclaim liability, assert no warranty is implied, and obligate licensees to not use the licensor's name or trademarks. Newer licenses often cover patent issues as well, either giving a restricted patent license or explicitly excluding patent rights.

The Mozilla Disjunctive Tri-License licenses the core Mozilla components under any one of three licenses (MPL, GPL, or the GNU Lesser General Public License LGPL). OSS developers can choose the one that best suits their needs for a particular project and component.

The Open Source Initiative (OSI) maintains a widely respected definition of "open source" and gives its approval to licenses that meet it (Open Source Initiative, 2008). OSI maintains and publishes a repository of approximately 70 approved OSS licenses.

Common practice has been for an OSS project to choose a single license under which all its products are released and to require developers to contribute their work only under conditions compatible with that license. For example, the Apache Contributor License Agreement grants enough of each author's rights to the Apache Software Foundation for the foundation to license the resulting systems under the Apache Software License. This sort of rights regime, in which the rights to a system's components are homogeneously granted and the system has a single well-defined OSS license, was the norm in the early days of OSS and continues to be practiced.

Open Architecture

Open architecture (OA) software is a customization technique introduced by Oreizy (2000) that enables third parties to modify a software system through its exposed architecture, evolving the system by replacing its components. Increasingly more software-intensive systems are developed using an OA strategy, not only with OSS components but also proprietary components with open APIs (Unity Technologies, 2008). Using this approach can lower development costs and increase reliability and function (Scacchi & Alspaugh, 2008). Composing a system with HLS components, however, increases the likelihood of conflicts, liabilities, and no-rights stemming from incompatible licenses. Thus, in our work we define an OA system as a software system consisting of components that are either open source or proprietary with open API, whose overall system rights at a minimum allow its use and redistribution in full or in part.

It may appear that using a system's architecture that incorporate OSS components and uses open APIs will result in an OA system. But not all such architectures will produce an OA, since the (possibly empty) set of available license rights for an OA system depends on (a) how and why OSS and open APIs are located within the system architecture, (b) how OSS and open APIs are implemented, embedded, or interconnected, and (c) the degree to which the licenses of different OSS components encumber all or part of a software system's architecture into which they are integrated (Alspaugh & Anton, n.d.; Scacchi & Alspaugh, 2008).

The following kinds of software elements appearing in common software architectures can affect whether the resulting systems are open or closed (Bass, Clements & Kazman, 2003).

Software source code components—These can be either (a) standalone programs, (b) libraries, frameworks, or middleware, (c) inter-application script code such as C shell scripts, or (d) intra-application script code, as for creating Rich Internet Applications using domain-specific languages such as XUL for the Firefox Web browser (Feldt, 2007) or "mashups" (Nelson & Churchill, 2006). Their source code is available and they can be rebuilt. Each may have its own distinct license.

Executable components—These components are in binary form, and the source code may not be open for access, review, modification, or possible redistribution (Rosen, 2005). If proprietary, they often cannot be redistributed, and so such components will be present in the design- and run-time architectures but not in the distribution-time architecture.

Software services—An appropriate software service can replace a source code or executable component.

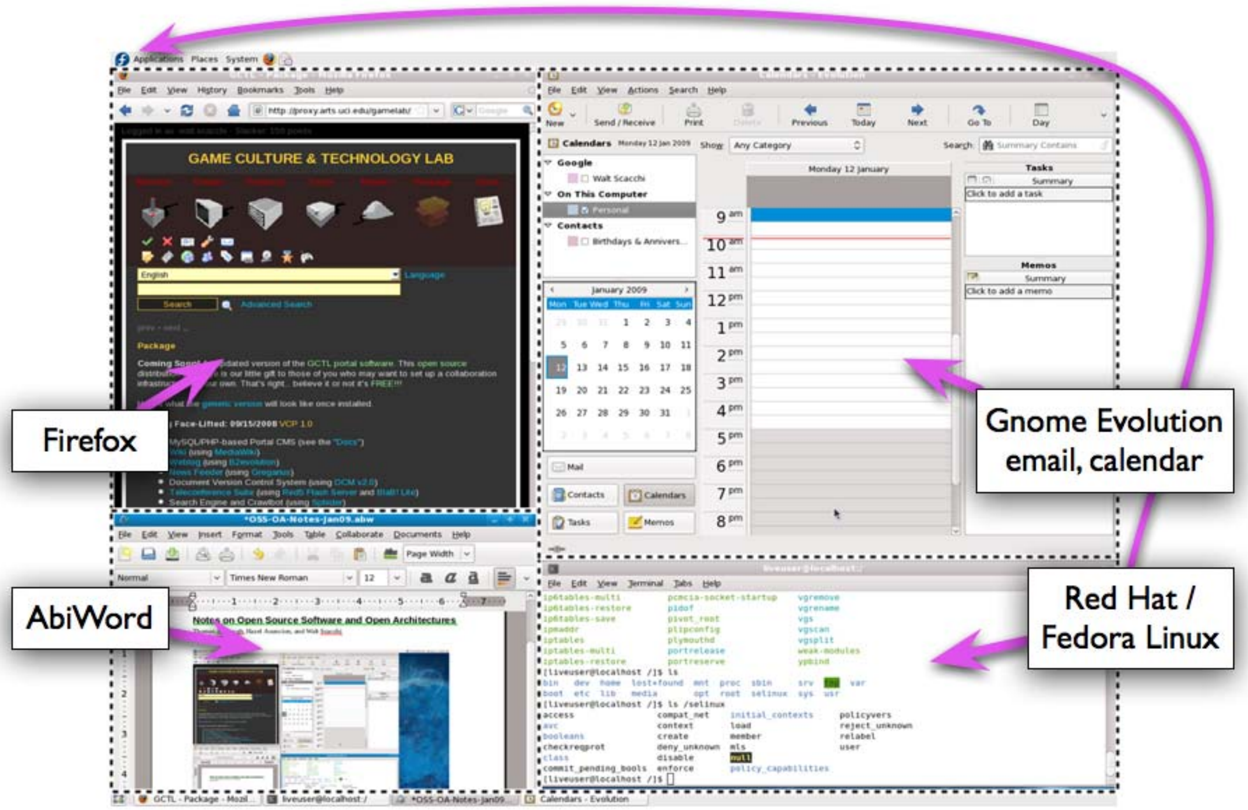


Figure 2. A Heterogeneously Licensed System Composed from Multiple Systems

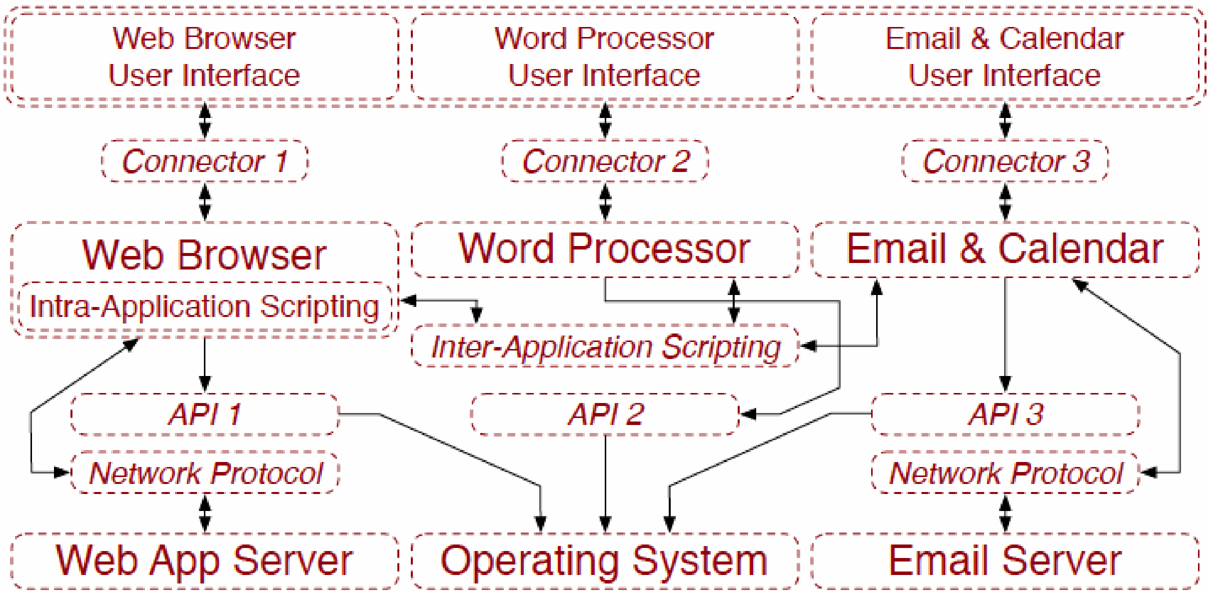
Application programming interfaces/APIs—Availability of externally visible and accessible APIs is the minimum requirement for an open system (Meyers & Oberndorf, 2001). APIs are not and cannot be licensed and can limit the propagation of license obligations.

Software connectors—Software whose intended purpose is to provide a standard or reusable way of communication through common interfaces, e.g., High Level Architecture (Kuhl, Weatherly & Dahmann, 1999), CORBA, MS.NET, Enterprise Java Beans, and GNU Lesser General Public License (LGPL) libraries. Connectors can also limit the propagation of license obligations.

Methods of connection—These include linking as part of a configured subsystem, dynamic linking, and client-server connections. Methods of connection affect license obligation propagation, with different methods affecting different licenses.

Configured system or subsystem architectures—These are software systems that are used as atomic components of a larger system and whose internal architecture may comprise components with different licenses, affecting the overall system license. To minimize license interaction, a configured system or sub-architecture may be surrounded by what we term a license firewall, namely a layer of dynamic links, client-server connections, license shims, or other connectors that block the propagation of reciprocal obligations.

Figure 3 shows a high-level view of a reference architecture that includes all the kinds of software elements listed above. This reference architecture has been instantiated in a number of configured systems that combine OSS and closed source components. One such system handles time sheets and payroll at our university; another implements the web portal for a university computer game research laboratory (the updated version now at



http://cgvw.ics.uci.edu).

Figure 3. The Design-time Architecture of the System of Figure 2

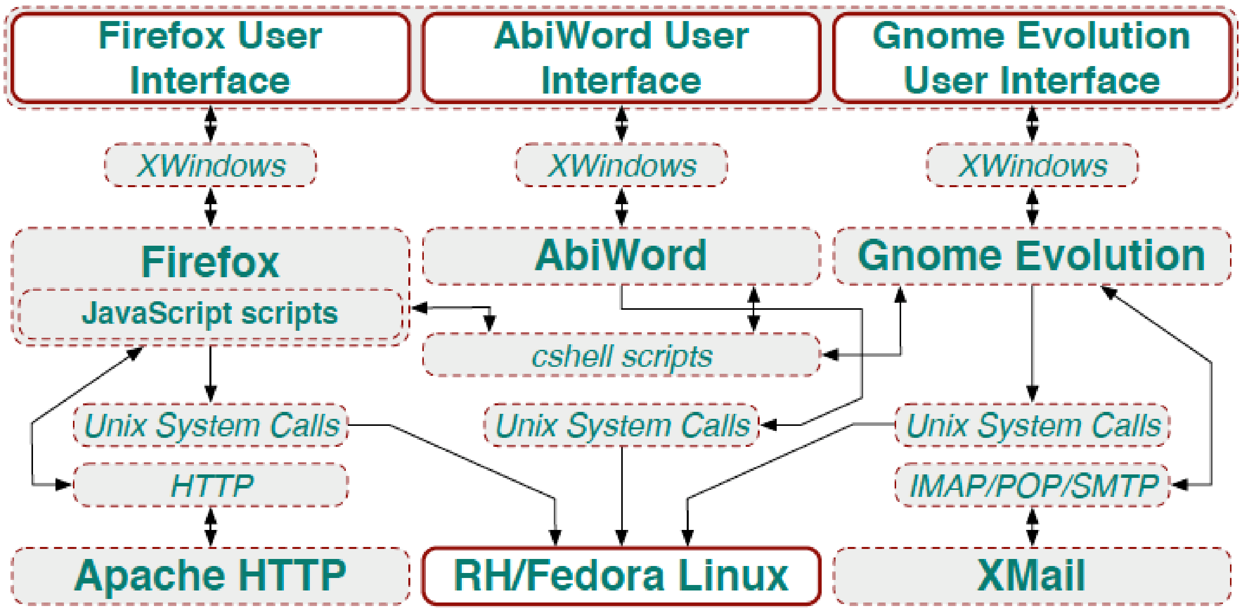


Figure 4. A Build-time Architecture Describing the Version Running in Figure 2

(Note: Components or connectors not visible in Figure 2 are shown in gray.)

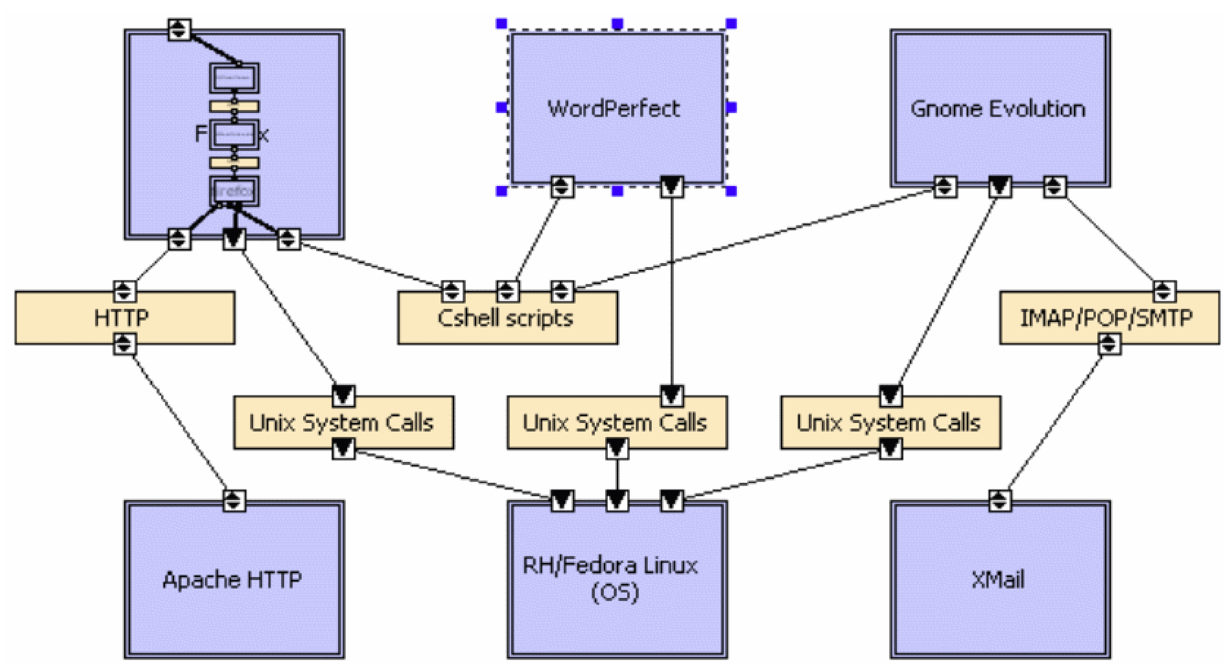


Figure 5. Instantiated Build-time Architecture (Figure 4) within ArchStudio

The configured systems consist of software components such as a Mozilla Web browser, Gnome Evolution email client, and AbiWord word processor (similar to MS Word), all running on a RedHat Fedora Linux operating system accessing file, print, and other remote networked servers such as an Apache Web server. Components are interconnected through a set of software connectors that bridges the interfaces of components and combines the provided functionality into the system's services.

Software Licenses

Copyright law is the common basis for software licenses and gives the original author of a work certain exclusive rights: the rights to use, copy, modify, merge, publish, distribute, sub-license, and sell copies. The author may license these rights, individually or in groups, to others; the license may give a right either exclusively or non-exclusively. After a period of years, copyright rights enter the public domain. Until then, copyright may only be obtained through licensing.

Licenses typically impose obligations that must be met in order for the licensee to realize the assigned rights. Common obligations include the obligation to publish at no cost any source code you modify (MPL) or the reciprocal obligation to publish all source code included at build-time or statically linked (GPL). The obligations may conflict, as when a GPL'd component's reciprocal obligation to publish source code of other components is combined with a proprietary component's license prohibition of publishing its source code. In this case, no rights may be available for the system as a whole, not even the right of use, because the two obligations cannot simultaneously be met and thus neither component can be used as part of the system.

The basic relationship between software license rights and obligations can be summarized as follows: if the specified obligations are met, then the corresponding rights are granted. For example, if you publish your modified source code and sub-licensed derived works under MPL, then you get all the MPL rights for both the original and the modified code. However, license details are complex, subtle, and difficult to comprehend and track. So it is easy to become confused or make mistakes. The challenge is multiplied when dealing with configured system architectures that compose a large number of components with heterogeneous licenses, so the need for legal counsel begins to seem inevitable (Rosen, 2005; Fontana et al., 2008).

We have developed an approach for expressing software licenses that is more formal and less ambiguous than natural language and that allows us to calculate and identify conflicts arising from the rights and obligations of two or more component's licenses. Our approach is based on Hohfeld's classic group of eight fundamental jural relations (1913), of which we use *right*, *duty*, *no-right*, and *privilege*. We start with a tuple <**actor, operation, action, object**> for expressing a right or obligation. The actor is the "licensee" for all the licenses we have examined. The operation is one of the following: "may," "must," "must not," or "need not," with "may" and "need not" expressing rights and "must" and "must not" expressing obligations. Because copyright rights are only available to entities that have been granted a sublicense, only the listed rights are available, and the absence of a right means that it is not available. The action is a verb or verb phrase describing what may, must, must not, or need not be done, with the object completing the description. A license may be expressed as a set of rights, with each right associated with zero or more obligations that must be fulfilled in order to enjoy that right. Figures 6, 7, and 8 show the meta-model we use to express licenses, with the allowed combinations of modality, object, and license shown in Figure 6.

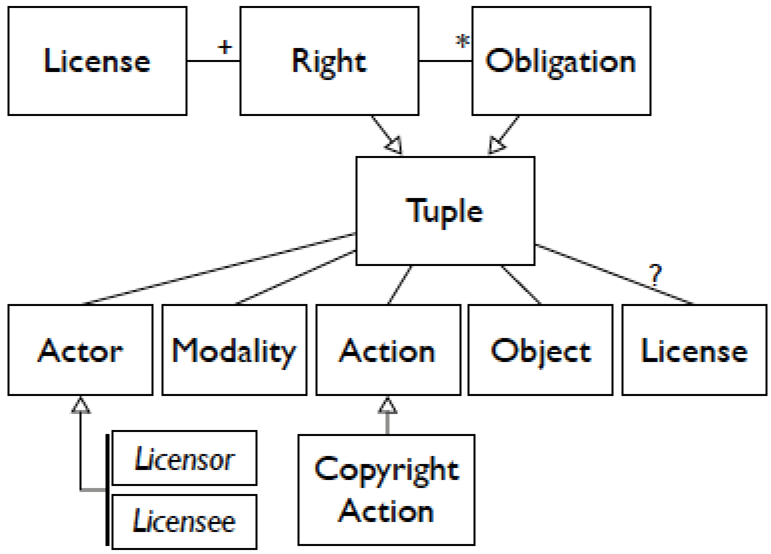


Figure 6. License Meta-model

HLS designers have developed a number heuristics to guide architectural design while avoiding some license conflicts. First, it is possible to use a reciprocally licensed component through a *license firewall* that limits the scope of reciprocal obligations. Rather than connecting conflicting components directly through static or other build-time links, the connection is made through a dynamic link, client-server protocol, license shim (such as a Limited General Public License connector), or run-time plug-ins. A second approach used by a number of large organizations is simply to avoid using any components with reciprocal licenses. A third approach is to meet the license obligations (if that is possible) by for example retaining copyright and license notices in the source and publishing the source code. However, even using design heuristics such as these (and there are many), keeping track of license rights and obligations across components that are interconnected in complex OAs quickly becomes too cumbersome. Automated support is needed to manage the multi-component, multi-license complexity.

	Modality	Object	License (optional)
Abstract Right	*May* or *Need Not*	*Any Under This License* *Any Source Under This License* *Any Component Under This License*	*This License* or *Object's License*
Concrete Right		Concrete Object	Concrete License
Concrete Obligation	*Must* or *Must Not*		
Abstract Obligation		*Right's Object* *All Sources Of Right's Object* *X Scope Sources* *X Scope Components*	Concrete License or *Right's License*

Figure 7. Modality, Object, and License

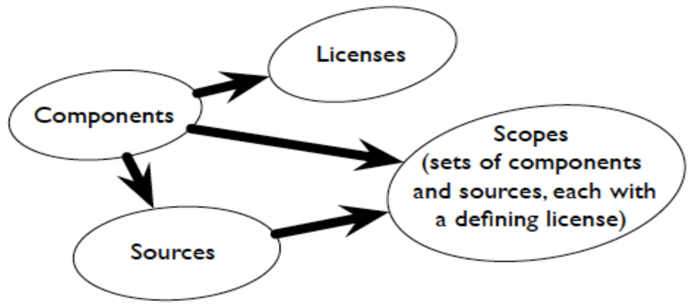


Figure 8. The License Architecture Meta-model

License Architectures

Our license model forms a basis for effective reasoning about licenses in the context of actual systems and calculating the resulting rights and obligations. In order to do so, we need a certain amount of information about the system's configuration at design-, build-, distribution-, and run-time. The needed information comprises the *license architecture*, an abstraction of the system architecture of its:

1. set of components of the system,

22. relation mapping each component to its license,
23. relation mapping each component to its set of sources, and
24. relation from each component to the set of components in the same license scope, for each license for which "scope" is defined (e.g., GPL) and from each source to the set of sources of components in the scope of its component.

With this information and definitions of the licenses involved, for example, as seen in Figure 9, we calculate rights and obligations for individual components or the entire system and guide heterogeneous license matching across components, as shown in Figure 10.

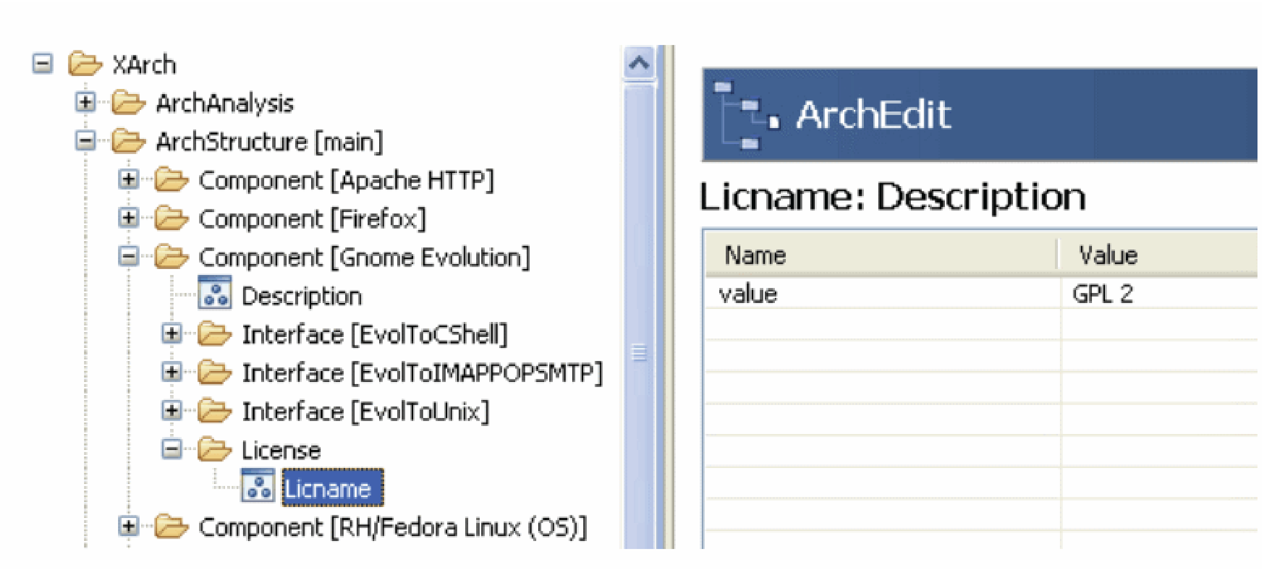


Figure 9. License Annotation of Gnome Evolution Component

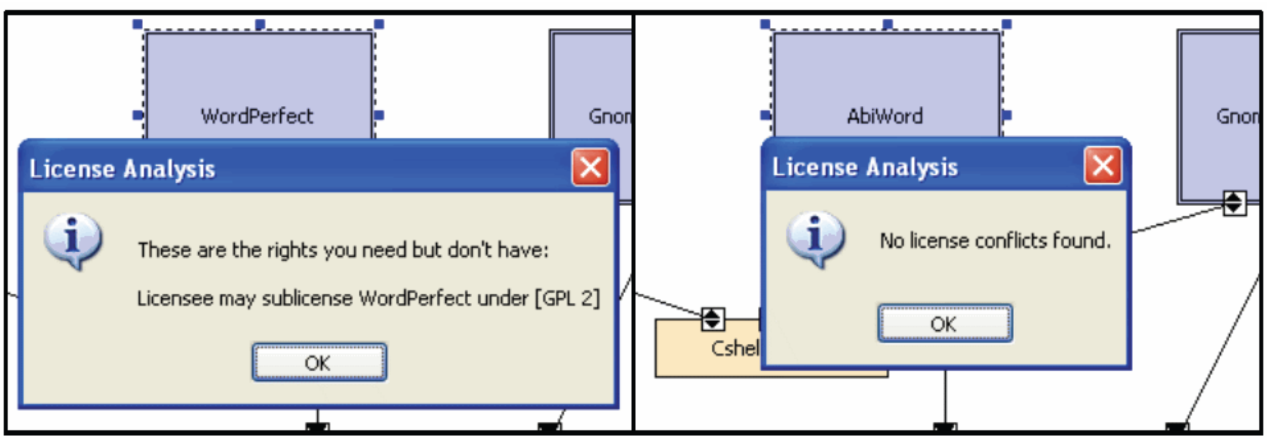


Figure 10. License Analysis Reports Before (left) and After (right) Replacing a Component System

License Analysis

Given a specification of a software system's architecture, we can associate software license attributes with the system's components, connectors, and sub-system architectures, resulting in a license architecture for the system, and calculate the copyright rights and obligations for the system's configuration. Due to the complexity of license architecture analysis and the need to re-analyze every time a component evolves, a component's license changes, a component is substituted, or the system architecture changes, OA integrators really need an automated license architecture analysis environment. We are developing a prototype of such an environment (Alspaugh et al., 2009a, May).

We use an architectural description language specified in xADL (Institute for Software Research, 2009) to describe OAs that can be designed and analyzed with a software architecture design environment (Medvidovic, Rosenblum & Taylor, 1999), such as

ArchStudio4 (Institute for Software Research, 2006). We have built the Software Architecture License Analysis module on top of ArchStudio's Traceability View (Asuncion & Taylor, 2009). This allows for the specification of licenses as a list of attributes (license tuples) using a form-based user interface in ArchStudio4 (Medvidovic, et al., 1999; Institute for Software Research, 2006). We analyze rights and obligations as described below (Alspaugh et al., 2009a, May) and as shown above in Figures 9 and 10.

Propagation of Reciprocal Obligations

We follow the widely accepted interpretation that build-time static linkages propagate the reciprocal obligations, but appropriate license firewalls do not. Analysis begins, therefore, by propagating these obligations along all connectors that are not license firewalls.

Obligation Conflicts

An obligation can conflict with another obligation or with the set of available rights by requiring a copyright right that has not been granted. For instance, a proprietary license may require that a licensee must not redistribute source code, but GPL states that a licensee must redistribute source code. Thus, the conflict appears in the modality of the two otherwise identical obligations, "must not" in the proprietary license and "must" in GPL.

Rights and Obligations Calculations

The rights available for the entire system (use, copy, modify, etc.) are calculated as the intersection of the sets of rights available for each component of the system. If a conflict is found involving the obligations and rights of linked components, it is possible for the system architect to consider an alternative linking scheme, e.g., using one or more connectors along the paths between the components that act as a license firewall. This means that the architecture and the automated environment together can determine what OA design best meets the problem at hand with available software components. Components with conflicting licenses do not need to be arbitrarily excluded but instead may expand the range of possible architectural alternatives if the architect seeks such flexibility and choice.

Discussion

At least two topics merit discussion following from our approach to semantically modeling and analyzing OA systems that are subject to heterogeneous software licenses. One is how our results might shed light on software systems whose architectures articulate a *software product line*, while the other is how our approach might be extended to also address the semantic modeling and analysis of *software system security requirements.*

First, organizing and developing software product lines (SPLs) relies on the development and use of explicit software architectures (Bosch, 2000; Clements & Northrop, 2001). However, the architecture of a SPL is not necessarily an OA—there is no requirement for it to be so. Thus, we are interested in discussing what happens when SPLs may conform to an OA, and to an OA that may be subject to heterogeneously licensed SPL components. Three considerations come to mind. If the SPL is subject to a single homogeneous software license, which may often be the case when a single vendor or government contractor has developed the SPL, then the license may act to reinforce a vendor lock-in situation with its customers. One of the motivating factors for OA is the desire to avoid such lock-in, whether or not the SPL components have open or standards-compliant APIs. Alternatively, if an OA system employs a reference architecture much like we have in the design-time architecture depicted in Figure 3, which is then instantiated into a specific software product configuration, as suggested in the build-time architecture (shown in Figure 4), then such a reference or design-time architecture as we have presented it here effectively defines a SPL consisting of possible different system instantiations composed from similar components instances (e.g., different but equivalent Web browsers, word processors, email, calendaring applications, and relational database management systems). Finally, if the SPL is based on an OA that integrates software components from multiple vendors or OSS components that are subject to heterogeneous licenses, then we have the situation analogous to what we have presented in this paper. So SPL concepts are compatible with OA systems that are composed from heterogeneously licensed components.

Second, as already noted, software licenses represent a collection of rights and obligations for what can or cannot be done with a licensed software component. Licenses thus denote non-functional requirements that apply to a software systems or system components as intellectual property (IP) during their development and deployment. But rights and obligations are not limited to concerns or constraints applicable only to software as IP. Instead, they can be written in ways that stipulate non-functional requirements of different kinds. Consider, for example, that desired or necessary software system security properties can also be expressed as rights and obligations addressing system confidentiality, integrity, accountability, system availability, and assurance (Breaux & Anton, 2005; 2008).

Traditionally, developing robust specifications for non-functional software system security properties in natural language often produces specifications that are ambiguous, misleading, inconsistent across system components, and lacking sufficient details (Yau & Chen, 2006). Using a semantic model to formally specify the rights and obligations required for a software system or component to be secure (Breaux & Anton, 2005; 2008; Yau & Chen, 2006) means that it may be possible to develop both a “security architecture” notation and model specification that associates given security rights and obligations across a software system or system of systems. Similarly, it suggests the possibility of developing computational tools or interactive architecture development environments that can be used to specify, model, and analyze a software system’s security architecture at different times in its development—design-time, build-time, and run-time.

The approach we have been developing for the past few years for modeling and analyzing software system license architectures for OA systems (Alspaugh et al., 2009, August/September; Alspaugh et al., 2009b, May; Scacchi & Alspaugh, 2008), may therefore be extendable to also being able to address OA systems with heterogeneous “software security license” rights and obligations. Furthermore, the idea of common or reusable software security licenses may be analogous to the reusable security requirements templates proposed by Firesmith (2004) at the Software Engineering Institute. Consequently, such an exploration and extension of the semantic software license modeling, meta-modeling, and computational analysis tools to also support software system security can be recognized as a promising next stage of our research studies.

Conclusion

This paper discusses the role of software ecosystems with heterogeneously licensed components in the development and evolution of OA systems. License rights and obligations play a key role in how and why an OA system evolves in its ecosystem. We note that license changes across versions of components are a characteristic of OA systems and software ecosystems with heterogeneously licensed components. A structure for modeling software licenses and the license architecture of a system and automated support for calculating its rights and obligations are needed in order to manage a system's evolution in the context of its ecosystem. We have outlined an approach for achieving these and sketched how they further the goal of reusing components in developing software-intensive systems. Much more work remains to be done, but we believe this approach turns a vexing problem into one for which workable solutions can be obtained.

Acknowledgments

This research is supported by grant #N00244-10-1-0038 from the Acquisition Research Program at the Naval Postgraduate School and by grant #0808783 from the National Science Foundation. No endorsement implied.

References

Alspaugh, T.A., & Anton, A.I. (2008, February). Scenario support for effective requirements. *Information and Software Technology*, *50*(3), 198–220.

Alspaugh, T.A., Asuncion, H.U., & Scacchi, W. (2009, May). Analyzing software licenses in open architecture software systems. In *Proceedings of the 2nd International Workshop on Emerging Trends in FLOSS Research and Development (FLOSS)*.

Alspaugh, T.A. , Asuncion, H.U., & Scacchi, W. (2009, August/September). Intellectual property rights requirements for heterogeneously-licensed systems. In *Proceedings of the 17th IEEE International Requirements Engineering Conference (RE'09)*.

Alspaugh, T.A., Asuncion, H.U., & Scacchi, W. (2009, May). Software licenses, open source components, and open architectures. In *Proceedings of the Sixth Annual Acquisition Research Symposium*. Monterey, CA: Naval Postgraduate School.

Asuncion, H., & Taylor, R.N. (2009, May). Capturing custom link semantics among heterogeneous artifacts and tools. In *Proceedings of the 5th International Workshop on Traceability in Emerging Forms of Software Engineering (TEFSE)*.

Bass, L., Clements, P., & Kazman, R. (2003). *Software architecture in practice*. Boston: Addison-Wesley Longman.

Bosch, J. (2000). *Design and use of software architectures: Adopting and evolving a product-line approach*. New York: Addison-Wesley Professional.

Breaux, T.D., & Anton, A.I. (2005). Analyzing goal semantics for rights, permissions, and obligations. In *Proceedings of the 13th IEEE International Conference on Requirements Engineering (RE'05)* (pp. 177–188).

Breaux, T.D., & Anton, A.I. (2008). Analyzing regulatory rules for privacy and security requirements. *IEEE Transactions on Software Engineering*, *34*(1), 5–20.

Brown, A.W., & Booch, G. (2002, April). Reusing open-source software and practices: The impact of open-source on commercial vendors. In *Proceedings of the International Conference on Software Reuse: Methods, Techniques, and Tools (ICSR-7)*.

Burk, D.L. (1998). Proprietary rights in hypertext linkages. *Journal of Information, Law and Technology*, *2*.

Clements, P., & Northrop, L. (2001). *Software product lines: Practices and patterns*. New York: Addison-Wesley Professional.

Feldt, K. (2007). *Programming Firefox: Building rich internet applications with XUL*. Sebastopol, CA: O'Reilly Media.

Firesmith, D. (2004). Specifying reusable security requirements. *Journal of Object Technology*, *3*(1), 61–75.

Fontana, R., Kuhn, B.M., Moglen, E., Norwood, M., Ravicher, D.B., Sandler, K., et al. (2008). *A legal issues primer for open source and free software projects*. New York: Software Freedom Law Center.

German, D.M., & Hassan, A.E. (2009, May). License integration patterns: Dealing with licenses mis-matches in component-based development. In *Proceedings of the 28th International Conference on Software Engineering (ICSE'09)*.

Hohfeld, W.N. (1913). Some fundamental legal conceptions as applied in judicial reasoning. *Yale Law Journal*, *23*(1), 16–59.

Institute for Software Research. (2006). *ArchStudio 4*. Retrieved from http://www.isr.uci.edu/projects/archstudio/

Institute for Software Research. (2009). *xADL 2.0*. Retrieved from http://www.isr.uci.edu/projects/xarchuci/

Kuhl, F., Weatherly, R., & Dahmann, J. (1999). *Creating computer simulation systems: An introduction to the high level architecture*. Englewood Cliffs, NJ: Prentice Hall.

Medvidovic, N., Rosenblum, D.S., & Taylor, R.N. (1999). A language and environment for architecture-based software development and evolution. In *Proceedings of the 21st International Conference on Software Engineering (ICSE'21)*, 44–53.

Meyers, B.C., & Oberndorf, P. (2001). *Managing Software Acquisition: Open Systems and COTS Products.* New York: Addison-Wesley Professional.

Nelson, L., & Churchill, E.F. (2006). Repurposing: Techniques for reuse and integration of interactive systems. In *Proceedings of the International Conference on Information Reuse and Integration (IRI-08)*, 490–495.

Open Source Initiative. (2008). *Open Source Initiative.* http://www.opensource.org/

Oreizy, P. (2000). *Open architecture software: A flexible approach to decentralized software evolution*. Unpublished doctoral dissertation, University of California, Irvine.

Rosen, L. (2005). *Open source licensing: Software freedom and intellectual property law*. Englewood Cliffs, NJ: Prentice Hall.

Scacchi, W., & Alspaugh, T.A. (2008, May). Emerging issues in the acquisition of open source software within the U.S. Department of Defense. In *Proceedings of the 5th Annual Acquisition Research Symposium*. Monterey, CA: Naval Postgraduate School.

Sen, R. (2007). A strategic analysis of competition between open source and proprietary software. *Journal of Management Information Systems*, *24*(1), 233–257.

Sen, R., Subramaniam, C., & Nelson, M.L. (2009). Determinants of the choice of open source software license. *Journal of Management Information Systems*, *25*(3), 207–240.

St. Laurent, A.M. (2004). *Understanding open source and free software licensing*. Sebastopol, CA: O'Reilly Media.

Stewart, K.J., Ammeter, A.P., & Maruping, L.M. (2006). Impacts of license choice and organizational sponsorship on user interest and development activity in open source software projects. *Information Systems Research*, *17*(2), 126–144.

Subramaniam, C., Sen, R., & Nelson, M.L. (2009). Determinants of open source software project success: A longitudinal study. *Decision Support Systems*, *46*(2), 576–585.

Tuunanen, T., Koskinen, J., & Karkkainen, T. (2009). Automated software license analysis. *Automated Software Engineering*, *16*(3-4), 455–490.

Unity Technologies. (2008). *End user license agreement*. Retrieved December 2008, from http://unity3d.com/unity/unity-end-user-license-2.x.html

Ven, K., & Mannaert, H. (2008). Challenges and strategies in the use of open source software by independent software vendors. *Information and Software Technology*, *50*(9-10), 991–1002.

Yau, S.S., & Chen, Z. (2006). A framework for specifying and managing security requirements in collaborative systems. In *Proceedings of the Third International Conference on Autonomic and Trusted Computing (ATC 2006)* (pp. 500–510).

Exploring Acquisition Strategies for Adopting a Software Product Line

Lawrence Jones—Lawrence G. Jones is a Senior Member of the Technical Staff in the Product Line Systems Program at the Software Engineering Institute (SEI) of Carnegie Mellon University. Prior to joining the SEI, he served a career in the US Air Force and is the former Chair of the Computer Science Department at the Air Force Academy. His PhD in computer science is from Vanderbilt University. He's a senior member of the IEEE and ACM, a Fellow of ABET and the CSAB, and a member of the ABET Board of Directors.

Lawrence G. Jones
Software Engineering Institute
Carnegie Mellon University
1155 Kelly Johnson Blvd., Suite 111
Colorado Springs, CO 80920
Office: 719 548 4744
Home: 719 488 0255
Cell: 719 661 8566
Fax: 719 548 4750
email: lgj@sei.cmu.edu

John Bergey—John Bergey joined the SEI in 1993 as a visiting scientist and became a member of the technical staff in 1995. Currently, Bergey is a member of the Research, Technology, and Systems Solutions Program (RTSS) and is active in the Architecture Centric Engineering and Product Line Practice initiatives. His role in these initiatives is to proactively assist DoD programs in applying SEI technologies (e.g., product line practices, the ATAM, and the QAW) to improve their acquisition practices and reduce software acquisition risk. Before coming to the SEI, Bergey was a software division manager with the US Naval Air Development Center.

John K. Bergey
Software Engineering Institute
Carnegie Mellon University
55 Woodard Court
Doylestown, PA 18901
Office: 215 348-0530
Home: 215 348-4991
Cell: 267 337-2944
Fax: 215 348-5255
email: jkb@sei.cmu.edu

Abstract

Many organizations are attracted to the well-documented benefits of a software product line approach. However, special challenges surround product line acquisition in the Department of Defense. We explain some basics of software product line practice, the challenges that make product line acquisition unique, and three basic acquisition strategies. We next describe the key contractual tasks a supplier must perform and map these to an enterprise view of product line acquisition. Using this context, we explain roles and responsibilities for the organizations involved, and describe important activities and deliverables. This provides a basis for building the necessary artifacts for a successful acquisition.

Introduction

Do you find yourself continually acquiring software-intensive systems that are similar to ones you have paid for in the past? Do you wish you could use your scarce resources to buy what is truly *new* without having to pay for re-development of essentially the same old solutions? If so, you should consider a software product line approach.

A *software product line* is a set of software-intensive systems sharing a common, managed set of features that satisfy the specific needs of a particular market segment or mission and that are developed from a common set of core assets in a prescribed way (Clements & Northrop, 2002). An increasing number of organizations are building their products as product lines in order to achieve large-scale productivity gains, improve time to field or market, maintain a market presence, compensate for an inability to hire, leverage existing resources, and achieve mass customization.

Commercial implementations of software product lines have resulted in some impressive results (Clements & Northrop, 2002; Schmid & Verlage, 2002). Although there has been some successful use of this technology within the Department of Defense (DoD), it carries special challenges for both the acquisition office and the supporting development contractors.

This paper addresses software product lines from the perspective of an acquisition organization. Product line acquisition involves adopting some new practices and rethinking some old practices. To introduce you to this new way of thinking we first provide a brief overview of software product line practice. We then describe the acquisition challenges implied by this technology, the basic acquisition strategies you can pursue, and the foundational contractual tasks that must be specified for successful product line acquisition. Against this background we then explore the structures, roles, and activities that will emerge during the lifetime of the product line from an enterprise perspective. We conclude by pointing to areas of future work to facilitate adoption of a product line acquisition approach.

Software Product Line Basics

An operating software product line organization embodies a core asset development activity and a product development activity, all orchestrated by a management activity. **Error! Reference source not found.** illustrates this triad of essential activities.

Figure 1. The Three Essential Product Line Activities

The arrows indicate not only that core assets are used to develop products, but also that revisions or even new core assets can evolve out of product development. The diagram does not specify where the process starts. In some contexts, existing products are mined for generic assets that are then migrated into a product line. At other times, the core assets may be developed first to produce an envisioned set of products. Core assets include plans, requirements, designs, documentation, and tests, as well as software.

While it is evident that product line practice calls for a new technical approach, new non-technical and business practices are equally crucial. There is a constant need for strong visionary management to invest the resources in the development of the core assets and to nurture the cultural change to view new products in the context of the product line, rather than as stand-alone systems.

In January 1997, the Carnegie Mellon® Software Engineering Institute (SEI) launched the Product Line Practice Initiative to help facilitate and accelerate the transition to sound software engineering practices using a product line approach. The goal of this initiative is to provide organizations with an integrated business and technical approach to systematic reuse, so they can produce and maintain similar systems of predictable quality more efficiently and at a lower cost.

A key strategy for achieving this goal has been the creation of the SEI *Framework for Software Product Line Practice*℠ ("the framework"). The framework describes the foundational product line concepts and identifies the essential activities and practices that an organization must master before it can successfully field a product line of software or software-intensive systems. The framework is a living document that is evolving as experience with product line practice grows. Version 4.0 is described in the book *Software Product Lines: Practices and Patterns* (Clements & Northrop, 2002), and the latest version is available on the SEI Web site (Northrop & Clements, 2009).

Software Product Line Acquisition Challenges

Bergey, Fisher, and Jones define acquisition as "The process of obtaining products and services through contracting. Contracting includes purchasing, buying, commissioning, licensing, leasing, and procuring of designated supplies and services via a formal written agreement" (Bergey, Fisher & Jones, 1999). Contracting works best when tasks are precisely defined. Moreover, contracting is best suited to efforts that are

- based on past experience, including use of familiar practices and processes
- based on well understood cost history data
- well bounded—that is, involving a fixed set of tasks and traditional deliverables in a well defined context (known requirements, quantity, schedule, and funding)
- unlikely to involve significant changes or redirection downstream

In the real world you won't have these ideal conditions, so typically there are challenges to any type of acquisition. What can make a product line acquisition especially challenging is when the acquisition must meet the needs of *multiple programs and target systems* that transcend *multiple platforms and developers*. DoD acquisition policies and

---

® Carnegie Mellon is registered in the US Patent and Trademark Office by Carnegie Mellon University.

℠ Framework for Software Product Line Practice is a service mark of Carnegie Mellon University.

infrastructure don't help since they are still largely predicated on acquiring one-of-a-kind, stove-piped systems. Other factors that make product line acquisitions more challenging are

- Planning a family of software products that rely on a common development effort is not a traditional DoD acquisition paradigm.
- There is no institutionalized means for funding the development and sustainment of a product line across multiple programs.
- Typically, neither program offices nor contractors are incentivized to adopt a product line approach.
- Adopting a product line approach may force the government to assume system integration responsibility.

Despite these challenges, many DoD organizations have successfully implemented software product lines. Several DoD and Army product line workshops have confirmed that programmatic issues—not technical issues—are the main impediments to widespread adoption of product line practices in the DoD (Bergey et al., 2003; Bergey, Cohen, Jones & Smith, 2004; Bergey, Cohen, Donohoe & Jones, 2005; Bergey & Cohen, 2006; and Bergey et al., 2009).

The essence of *product line acquisition* is obtaining a software product line through contracting. The first step, addressed in the next section, is to address the contracting challenges by selecting an appropriate acquisition strategy.

Basic Acquisition Strategies for Acquiring Software Products via a Product Line

Developing a suitable acquisition strategy is a key consideration in adopting a product line approach in the DoD. A program manager (PM) can choose from three basic strategies:

Acquisition Strategy 1: *A PM commissions a contractor to develop products using the contractor's proprietary software product line.* This strategy involves acquiring products directly from a contractor that has an existing product line. An example is the Textron Overwatch Intelligence Center (OIC) product line (Jensen, 2009).

Acquisition Strategy 2: *A PM commissions a government organization to develop a software product line.* This strategy involves acquiring a government-owned product line (production capability and products) using the in-house capabilities[8] of a designated government acquisition organization. An example is the Army's Advanced Multiplex Test System (AMTS) (Cohen & Capolongo, 2007).

Acquisition Strategy 3: *A PM commissions a contractor to develop a government-owned software product line.* This strategy involves acquiring a government-owned product line (production capability and products) from a contractor. An example is the Live, Virtual, Constructive Integrating Architecture (LVC-IA) product line of the Army's PEO STRI (Bergey et al., 2009).

The difficulty in executing these different strategies varies significantly since they require different levels of management sophistication and technical skills on the part of the

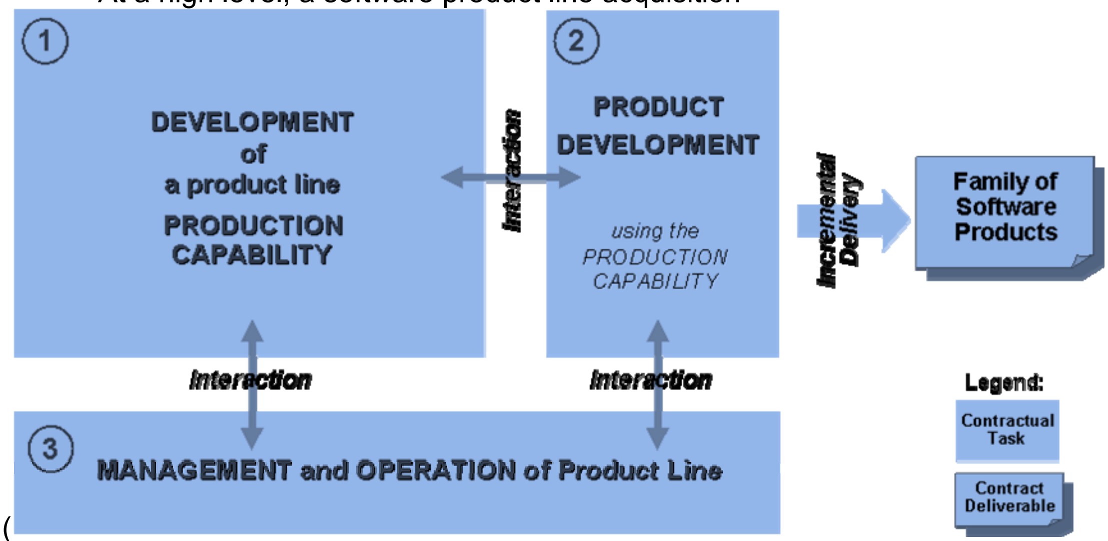
[8] This may include using the services of local Systems Engineering and Technical Assistance (SETA) contractors.

acquisition organization. Related considerations include the data rights to product line artifacts, and the risk of a supplier going out of business. Of the three approaches presented, the most challenging is Acquisition Strategy 3; the outcome is more unpredictable, thus making the risk to the government greater.

Some of the most successful product line efforts reported to date in government acquisitions were based on Strategy 1 (Brownsword & Clements, 1996; Jensen, 2007) and Strategy 2 (Cohen, Dunn & Soule, 2002; and Cohen & Capolongo, 2007). Strategy 3 offers potentially huge rewards but is the most challenging to execute. However, several success stories have been reported (Bergey et al., 2009; Bergey, Cohen, Donohoe & Jones, 2010).

Contractual Tasks for a Software Product Line Acquisition

At a high level, a software product line acquisition



(

) consists of three contractual tasks that a developer must perform. These tasks are

1. the development of a product line production capability

25. the development of a family of software products using that production capability

26. the management and operation of the product line

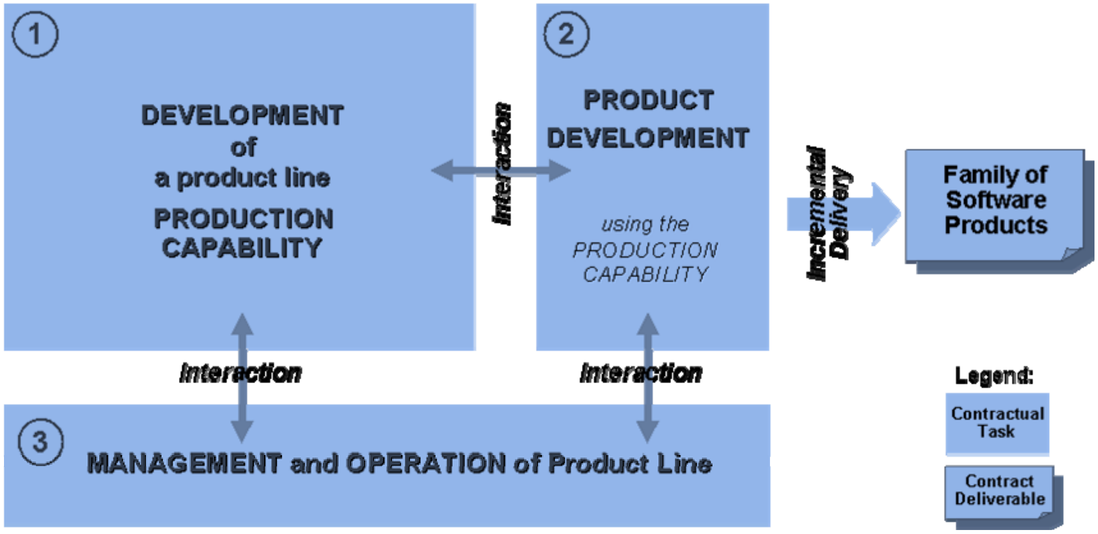


Figure 2. Three Major Contractual Tasks for a Software Product Line Acquisition

A *product line production capability* includes the product line core assets, a production plan to specify how to build products from the core assets (Chastek & McGregor, 2002), and the infrastructure to support the production operation. A *software development plan*, a traditional contractual deliverable, can be used to describe and govern the development of the product line production capability. Product developers then use the production capability to develop specific products within the product line. A *product line adoption plan* describes the approach for initiating the product line, and a *product line concept of operations* describes the approach for managing and operating the product line.

To operationalize these tasks we must assign specific responsibilities to specific organizational units. To help do this, it is useful to consider an enterprise view of the acquisition, described in the next section.

An Enterprise View for Software Product Line Acquisition

An enterprise view helps to frame the various aspects of a product line initiative. Such a view can help clarify important questions such as:

- How will the effort be organized?
- What will be the roles and responsibilities of the different organizational units?
- What deliverables will be produced? And what groups will be responsible for them?
- How will product line practices, such as *product line requirements engineering,* be implemented from an enterprise perspective?

Error! Reference source not found. shows an example of an enterprise view that corresponds to Acquisition Strategy 3 (described in Section 0). This example captures the essence of the major product line activities in an acquisition context and helps ensure that all stakeholders have a common understanding of the ramifications of the approach.

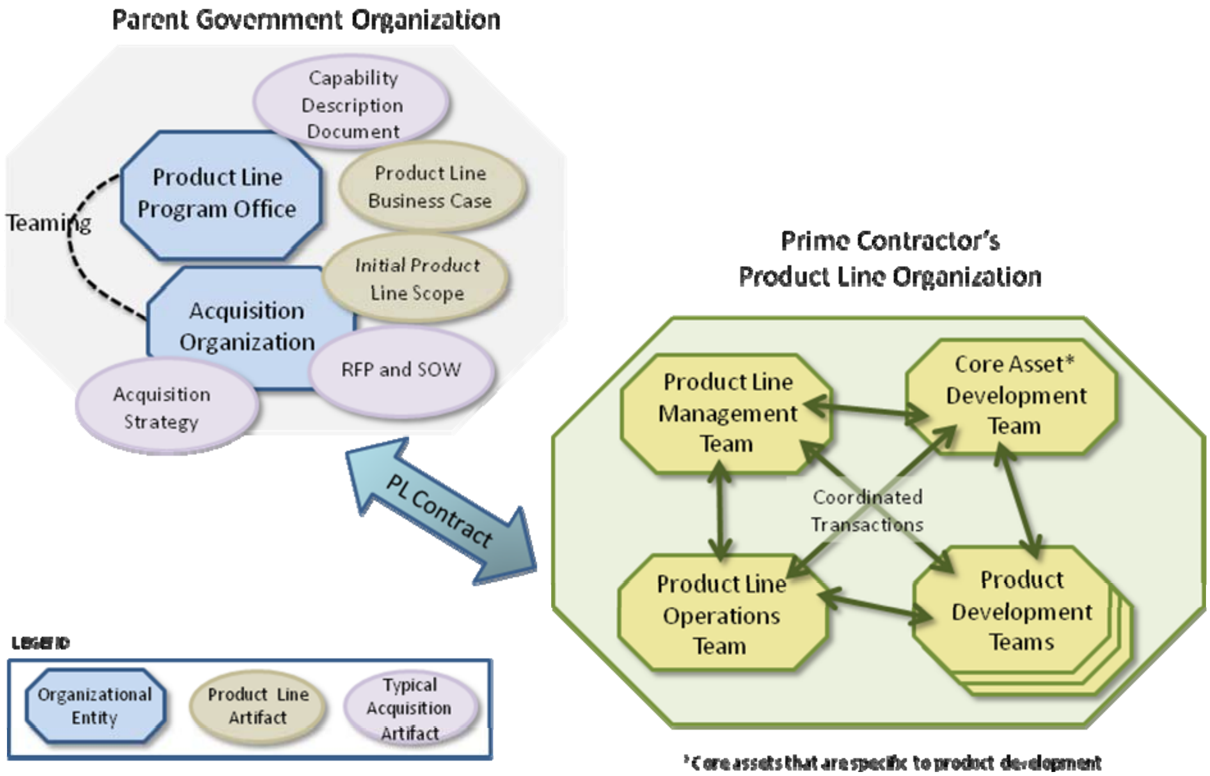


Figure 3. Sample Enterprise View of a Product Line Acquisition

The two primary organizational elements are the *parent government organization*, which is responsible for acquiring the product line, and the *prime contractor's organization*, which is responsible for implementing and sustaining the product line.

The subdivision of the prime contractor's product line organization into a *management team*, a *core asset team*, a *product development team*, and an *operations team* is just one example of how a developer organization might implement a product line approach. In this configuration, the *management team*, *core asset team*, and *operations team* are the organizational elements that are responsible for establishing the production capability that the *product development teams* will use.

The view in **Error! Reference source not found.** may be expanded to depict details of product development (Figure 4). This view shows how a *product development team* would interface with the other teams and use the *product line production capability* to develop products. Each product is an example of strategic reuse of the product line's core assets. This view identifies the contract deliverables that a product development team would produce. Since acquisition organizations have a penchant for thinking in terms of contractual deliverables, this view facilitates an understanding of how a product line functions and of the roles and responsibilities of the various teams.

Accordingly, this example shows that the product development team is initially responsible for producing a product specification. Following this, the team must develop any unique software components that are not part of the core assets and create a production plan for building the specific product that will satisfy the product specification. Another key deliverable is a product test plan that would draw on the existing testing artifacts that are also part of the core assets. Of course, this assumes the product team will be responsible for the initial testing of the product as well as generating any other related artifacts, such as test cases for unique software components and attached processes describing how to use them.

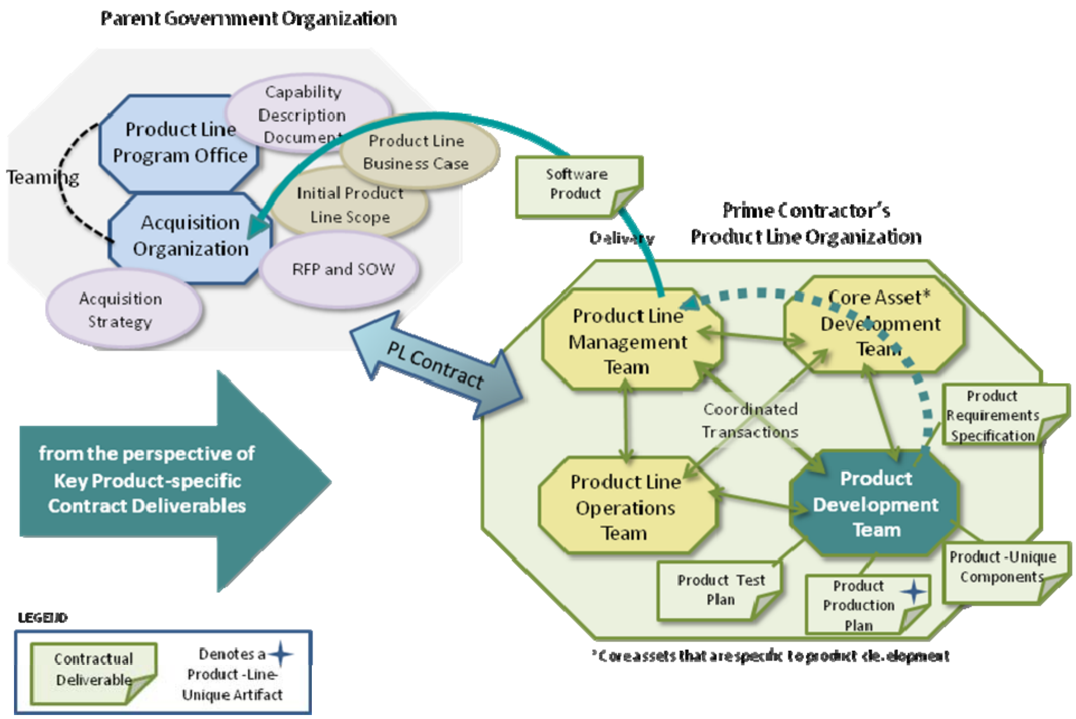


Figure 4. A View of a Product Development Team Using the Product Line Production Capability

Apart from the product itself, the major deliverable item is the *production plan*. It includes the process to be used for building a product from the core assets and lays out the project details to enable execution and management of the process (e.g., by including such details as the schedule, bill of materials, and metrics).

The importance of carefully specifying all deliverables cannot be overstated. The government needs to be proactive in specifying the required deliverables in the RFP/contract or the acquisition will be problematic.

A Customer View of Software Product Line Acquisition

Error! Reference source not found. depicts a product line acquisition from a customer perspective and shows the customer's interaction with the product line operations. While there are several potential customer views, this one depicts the simplest case where the program office is also the customer. The program office is the customer when the product being developed is for a system under the jurisdiction of the program office.

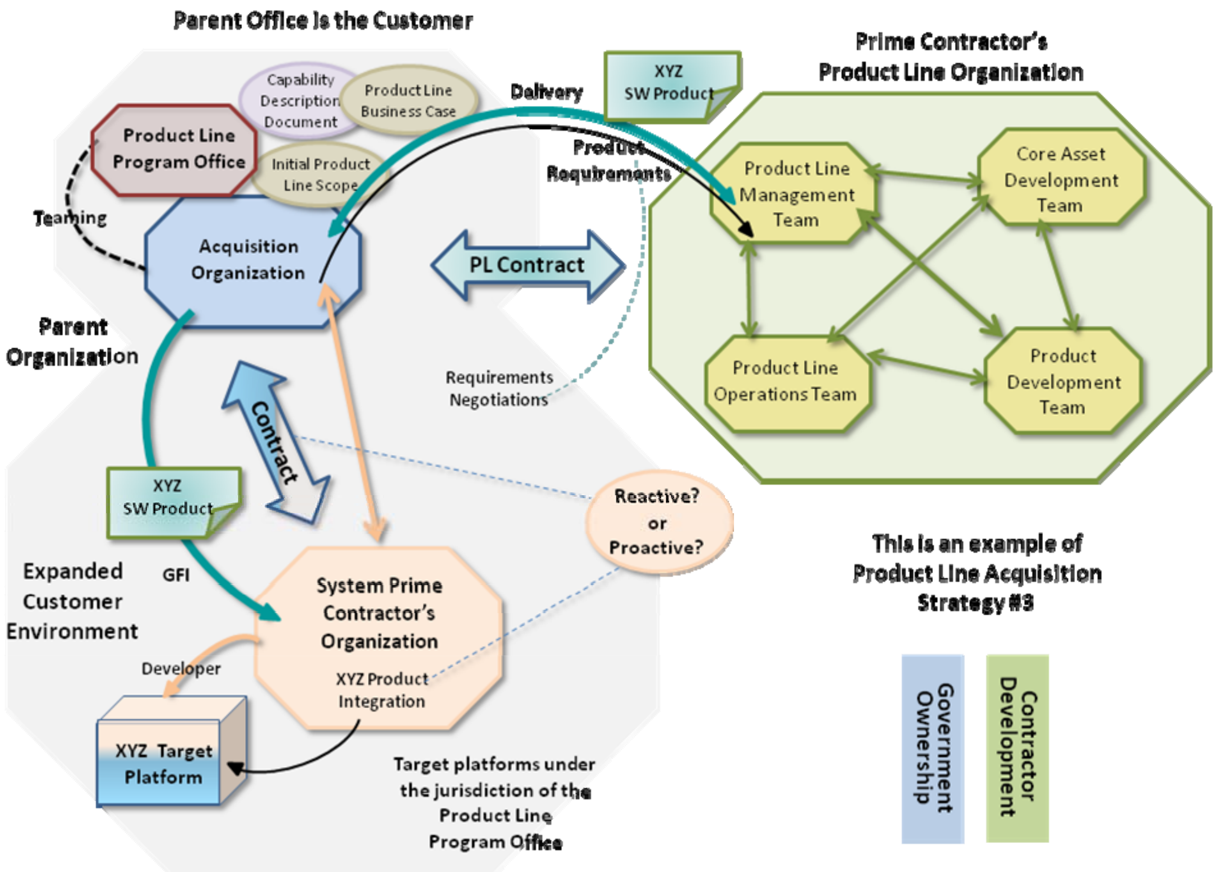


Figure 5. Simplest Case of Customer Interaction with Product Line Developer

While the program office is ultimately responsible for the product and its targeted system, a system prime contractor (under contract to the program office) is the agent that is actually responsible for developing and sustaining the target system. This situation corresponds to the relationship depicted in **Error! Reference source not found.** between the *parent organization* and the *expanded customer environment*. In this context, the arrows shown in **Error! Reference source not found.** depict a scenario that leads to delivering a new product (in the product line family) to the customer. The corresponding sequence of events in this scenario is described below:

1. The program office analyzes and specifies the new product requirements (in conjunction with its acquisition organization and the contractor responsible for the target system).

27. The program office tasks the product line contractor with developing a new product (in accordance with the negotiated product requirements).

28. The product line contractor develops the new product and delivers it to the program office.

29. The program office (in conjunction with its acquisition organization), in turn, supplies the product as a government furnished item (GFI) to the target system contractor.

30. The target system contractor appropriately integrates the product into the target system.

31. The program office or its acquisition organization appropriately brokers any problems that arise in deploying the product with the product line contractor and the target system contractor.

An interesting aspect is that if the product line developer were a government organization (e.g., an Army Software Engineering Center (SEC)) instead of a system prime contractor, it would give the program office more flexibility, since the *Federal Acquisition Regulation* (*FAR*) considerations wouldn't come into play. Contractors would still play be a significant role, however, because SEC's typically contract with many suppliers to acquire needed skills, expertise, and resources. Such a situation would correspond to Acquisition Strategy 2. Even though this arrangement simplifies things, the enterprise view is still useful for clarifying the concepts.

The ideas here can be extended to the more complicated situation, where the customer is not the program office that is responsible for the product line, but is rather a different program office that has jurisdiction over other target systems. Exploring that type of engagement can be important because it represents the vision of many product line advocates.

Future Considerations

Among future activities the SEI is pursuing to promote product line acquisitions and make them more effective and commonplace in the DoD are

- providing *sample acquisition strategies* (e.g., involving a competitive down select) that an acquisition organization can use via appropriate tailoring
- creating *a comprehensive work breakdown structure (WBS)* for use as a management tool to govern a product line initiative
- creating an *acquisition timeline* with deliverables and specifying a series of *contractual events for technically monitoring and evaluating a product line effort*
- creating sample *SOW contract language* for a acquiring a software product line
- creating a sample *contract data requirements list* (CDRL) and *data item descriptions* (DIDs) for key software product line deliverables

Conclusion

Developing a suitable acquisition strategy is a key element for any DoD program that is considering adopting a product line approach. Moreover, a proactive acquisition approach greatly enhances the likelihood that a DoD product line initiative will succeed. In a reactive approach, an acquisition organization may not have an effective contractual means for managing the product line and performing its technical oversight and contract monitoring responsibilities.

If a program office is going to commission a contractor or government organization to develop a product line, the acquisition organization needs to specify the SOW tasks carefully to ensure product line aspects are appropriately covered and key deliverables are included. Creating a product line-specific WBS and a product line concept of operations from the acquirer's perspective are a good starting point.

An enterprise view provides an apt basis for describing a product line initiative from an acquisition perspective. This enables stakeholders to have greater insight and understanding of what a product line acquisition actually entails; it is useful for

- determining the division of responsibilities between the program office, acquisition organization, and development organization
- understanding stakeholder interactions and interdependencies and assigning specific roles and responsibilities
- understanding the "contracting realities" of different candidate approaches that are typically glossed over and become problematic downstream unless they are addressed up front
- stimulating discussion, analyzing different "acquisition threads," and answering pertinent questions such as
  - How is the product line effort being organized and managed?
  - How do requirements flow from the customer to the core asset team?
  - How does an external developer use the core assets to develop a product?
  - What is the information flow for sustaining products that are in the field?

Experience has shown that if a program office is interested in adopting a product line approach, a good starting point is to conduct an acquisition-planning workshop with stakeholders early in the program-planning phase.

References

Bergey, J., Fisher, M., & Jones, L. (1999). *The DoD acquisition environment and software product lines* (CMU/SEI-99-TN-004). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University. Retrieved from http://www.sei.cmu.edu/library/abstracts/reports/99tn004.cfm

Bergey, J., et al. (2003). *Fifth DoD product line practice workshop report* (CMU/SEI-2003-TR-007). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University. Retrieved from http://www.sei.cmu.edu/publications/documents/03.reports/03tr007.html

Bergey, J., Cohen, S., Jones, L., & Smith, D. (2004). *Software product lines: Experiences from the sixth DoD software product line workshop* (CMU/SEI-2004-TN-011). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University. Retrieved from http://www.sei.cmu.edu/publications/documents/04.reports/04tn011.html

Bergey, J., Cohen, S., Donohoe, P., & Jones, L. (2005). *Software product lines: Experiences from the seventh DoD software product line workshop* (CMU/SEI-2005-TR-001). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University. Retrieved from http://www.sei.cmu.edu/publications/documents/05.reports/05tr001.html

Bergey, J., & Cohen, S. (2006). *Product line acquisition in a DoD organization–Guidance for decision makers* (CMU/SEI-2006-TN-020). Pittsburgh, PA: Software Engineering

Institute, Carnegie Mellon University. Retrieved from http://www.sei.cmu.edu/publications/documents/06.reports/06tn020.html

Bergey, J., Cohen, S., Donohoe, P., Fisher, M., Jones, L., & Little, R. (2009). *Software product lines: Report of the 2009 U.S. Army software product line workshop* (CMU/SEI-2009-TR-012). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University.

Bergey, J., Cohen, S., Donohoe, P., & Jones, L. (2010). *Software product lines: Report of the 2010 U.S. Army software product line workshop* (forthcoming). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University.

Brownsword, L., & Clements, P. (1996). *A case study in successful product line development* (CMU/SEI-96-TR-016). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University. Retrieved from http://www.sei.cmu.edu/pub/documents/96.reports/pdf/tr016.96.pdf

Chastek, G., & McGregor, J. (2006).*Guidelines for developing a software product line production plan* (CMU/SEI-2002-TR-006). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University. Retrieved from http://www.sei.cmu.edu/publications/documents/02.reports/02tr006.html
Clements, P., Northrop, L. (2002). *Software product lines: Practices and patterns*. Reading, MA: Addison-Wesley.
Cohen, S., Dunn, E., & Soule, A. (2002). *Successful product line development and sustainment: A DoD case study* (CMU/SEI-2002-TN-018). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University.Retrieved http://www.sei.cmu.edu/publications/documents/02.reports/02tn018.html

Cohen, S., & Capolongo, K. (2007, May). A product line architecture for Army aviation diagnostics and maintenance: Views and evolution. *SATURN 2007 Conference.* Pittsburgh, PA. Retrieved from http://www.sei.cmu.edu/library/assets/SATURNCohenPL.pdf

Jensen, P. (2007, September). Experiences with product line development of multi-discipline analysis software at Overwatch Textron Systems. In *SPLC 2007: Proceedings of the 11th International Software Product Line Conference* (pp. 35-43). Kyoto, Japan; Los Alamitos, CA: Institute of Electrical and Electronics Engineers, Inc.

Jensen, P. (2009, January). Experiences with software product line development. *CrossTalk*, *22*(1), 11–14. Retrieved from http://www.dtic.mil/cgi-bin/GetTRDoc?AD=ADA499051&Location=U2&doc=GetTRDoc.pdf

Northrop, L., & Clements, P., et al. (2009). *A framework for software product line practice version 5.0*. Retrieved from http://www.sei.cmu.edu/productlines/tools/framework/index.cfm

Schmid, K., & Verlage, M. (2002, July/August). The economic impact of product line adoption and evolution. *IEEE Software*, 50-57.

Panel #6 – Acquisition Reform: Beyond Marginal Adjustments

Wednesday, May 12, 2010	
1:45 p.m. – 3:15 p.m.	**Chair: John Birkler**, Senior Management Systems Analyst, RAND Corporation **Discussant**: **Charles Nemfakos**, Senior Fellow, RAND Corporation ***Goldwater-Nichols: Military-run versus Civilian-run Acquisition: Will the Twain Ever Meet in the DoN?*** **Irv Blickstein and Charles P. Nemfakos**, RAND Corporation ***Risk Factors versus Dollar Value: Changing How Weapon System Programs are Managed*** **Robert Murphy,** RAND Corporation

Goldwater-Nichols: Military-run versus Civilian-run Acquisition: Will the Twain Ever Meet in the DoN?

Irv Blickstein—Mr. Blickstein is a Senior Engineer at RAND, has 35 years of experience in the field of Defense analysis and management, with a specialty in planning, programming and budgeting as well as acquisition. For the past eight years, Mr. Blickstein has managed the research of activities of a series of projects in support of the office of the Chief of Naval Operations and the Undersecretary of Defense, Acquisition, Technology & Logistics. Mr. Blickstein's projects have covered a wide variety of topics, including directed energy, unmanned vehicle, reviews of foreign acquisition programs, sea basing and the cost analyses of both ships and aircraft. He has studied the cost of statutory and regulatory constraints in the DoD. Mr. Blickstein serves on the Chief of Naval Operations Executive Panel. He served 31 years in the Department of Defense before joining RAND, 18 years as a Senior Executive and was awarded four Presidential Rank awards.

Charles P. Nemfakos—Mr. Nemfakos became a Senior Fellow at RAND after leading Nemfakos Partners, LLC, in supporting public and private sector clients, here and abroad, in dealing with the demands of the emerging defense/security realities and the pressures of the global marketplace. Previously, Mr. Nemfakos was an executive with Lockheed Martin Corporation, directing efforts to rationalize product lines, providing program focus to enhance competitive strategies, and seeking new directions and opportunities for growth among the various Corporation companies by anticipating demands of transformational processes— both in industry and across the customer base. Mr. Nemfakos served in assignments as a budget analyst and as a planner in the Office of the Secretary of Defense and the Department of the Navy. As a Senior Executive in the latter, he served in a variety of financial positions—as Deputy Assistant Secretary for Installations and Logistics, as Deputy Under Secretary, and as Comptroller. During this service, Mr. Nemfakos was responsible for formulation, presentation, and execution of the Department's budget, directing the Department's base closure process, providing executive-level continuity for the Department in areas of institutional management and strategic planning, and supporting privatization initiatives, incentive structures, and right-sizing efforts. Finally, Mr. Nemfakos was the Department's Chief Financial Officer. During the last decade of his career he played a central role in the transformation of the Department after the Cold War. Drawing on this body of experience, Mr. Nemfakos provides research, strategically oriented analyses, support and advice to a broad variety of RAND clients.

Mr. Nemfakos has lectured extensively on public policy in resource allocation, on national security issues, on public administration policy, and on public/private entity relationships. He has served on Boards of Directors and/or Advisors of companies and non-profits, educational entities, as a Senior Fellow at the Center for Naval Analyses, and as an Adjunct at the National Defense University. He is currently the Chair of the Disaster and Humanitarian Relief Committee of SOLE—the International Society of Logistics.

Mr. Nemfakos, in addition to receiving many numerous awards and citations, has been recognized by three Presidents of the United States with four Presidential Rank Awards, by the Secretary of Defense as one of nine Career Civilian Exemplars in the 228-year history of the Armed Forces, by American University with the Roger W. Jones Award for Executive Leadership, and by the National Academy of Public Administration as an elected Fellow.

Abstract

In 1986, the military establishment underwent the most sweeping package of defense reforms to be enacted in almost 40 years, starting with the *Goldwater-Nichols Department of Defense Reorganization Act*. Related reforms followed shortly thereafter, including those contained in the *National Defense Authorization Act of 1987*, reflecting many of the recommendations of the Packard Commission. In the two decades following enactment of this legislation, the military establishment has taken numerous steps to implement them. However, some within the military services have grown increasingly

concerned about the effect of some of these reforms, perceiving a growing divide between a military-run requirements process and a civilian-run acquisition process that they regard inimical to the efficient and effective support of military forces.

This study describes analysis done, conclusions drawn and recommendations made to the Department of the Navy (DoN) regarding the closer integration of the interests of the Chief of Naval Operation (CNO), the Assistant Secretary of the Navy for Research, Development and Acquisition (ASN (RD&A)), and the Navy acquisition community writ large to increase material capabilities and readiness at reduced costs. The effort was pursued through an assessment of the implementation of the *Goldwater-Nichols Act of 1986* in the Department of the Navy and related acquisition reforms. It also includes a comparison of the DoN with that of the Air Force and Army.

Summary

The *House Armed Services Committee Panel on Defense Acquisition Reform: Findings & Recommendations*, dated March 23, 2010, made the following recommendation in its review of DoD acquisition: The Department and Congress should review and clarify the *Goldwater-Nichols Act*'s separation between acquisition and the military service chiefs to allow detailed coordination and interaction between the requirements and acquisition processes and to encourage enhanced military service chief participation in contract quality assurance.

The Panel is concerned that the divide established in the *Goldwater-Nichols Act* between acquisition and the military service chiefs has become so wide that it hinders both the acquisition and requirements process. While the fundamental construct in the *Goldwater-Nichols Act* correctly assigned lead responsibility for acquisition to the Department's civilian leaders, the act should be clarified to ensure that the requirements process that must coordinate with all categories of the defense acquisition system freely interacts with the acquisition process. The service chiefs should also be given greater authority and responsibility to oversee contract quality assurance especially for contracts that are highly operational in nature.

The report addresses how the Department of the Navy approached and later instituted the *Goldwater-Nichols Act* in it Acquisition functions. The hallmark of that was to create and increase the divide between those who developed the Departments' requirements and those who went on to procure them. Starting with the change in the officer core that separated officers of the line with acquisition officer and ending with the Department's "Gate" System, the Department has clearly developed parallel processes that are marked by division, discord and a lack of cooperation. The military side, building requirements, and the civilian side, buying the requirements rarely induced each other's point of view in their internal processes. As a result, requirements are sometimes overstated and unexecutable and the procurements process simply builds upon what is directed in the requirements process. The acquisition boards and committees that make decisions are managed by the acquisition executives while the military requirements personnel attend at lower levels than flag rank.

This paper discusses how the Department achieved that position over the years, addresses how the Army and Air Force instituted *Goldwater-Nichols* differently and makes some modest suggestions for change.

Introduction

The debate over the appropriate roles of the Chief of Naval Operations (CNO) and of the Secretary of the Navy (SECNAV) in the material management process stretches back to the Civil War era.[9] The essence of the debate is the role of uniformed leadership (i.e., the CNO) compared with that of the civilian leadership (i.e., the SECNAV) in determining what warfighting capabilities are required, what systems will be procured to provide these capabilities, how these systems will be supported when introduced into the fleet, and how these systems will be funded. In 1986, the *Goldwater-Nichols Department of Defense Reorganization Act* (US Congress, 1986) weighed in on these roles as a key element in its overall reform of defense organization and processes, giving responsibility for defense acquisitions to civilian secretaries while strengthening joint uniformed oversight over the requirements process.

In the two decades following enactment of this momentous legislation, the military services have taken numerous steps to implement its provisions and to respond to related acquisition reforms. However, some senior Navy officials have grown increasingly concerned about the unintended consequences of these reforms, perceiving a growing divide between a military-run requirements process and a civilian-run acquisition process.

Objectives and Approaches

RAND examined (1) the policy issues that drove the passage of the *Goldwater-Nichols Act* and related acquisition reforms and (2) the Department of the Navy's (DoN) implementation of these reforms, particularly with regard to their influence on military and civilian roles in the Navy's acquisition process. It describes the context in which the acquisition reform occurred and the effects of that reform on acquisition processes, focusing largely on the Department of the Navy. Drawing on a series of interviews with several officials who were present when the legislation was implemented, it argues that the effect was to focus the attention of the Chief of Naval Operations on requirements issues and to divorce him from the acquisition process in a way that has been detrimental to the effective and efficient acquisition of materiel for the Department of the Navy. It further argues that this separation went beyond what the legislation required and that there needs to be closer integration of the interests of the Chief of Naval Operation (CNO) with the Assistant Secretary of the Navy for Research, Development and Acquisition (ASN (RD&A)) and the Navy acquisition community to increase material capabilities and readiness at reduced costs.

The authors note that this paper deals with more than the *Goldwater-Nichols* legislation and considers several other influences such as the troubled history of the armed forces in coordinating joint operations and influential commissions such as the Packard Commission. These other influences all coalesced in the mid-1980s and created an environment—a perfect storm—that both made the passage of *Goldwater-Nichols* possible and colored its implementation. In essence, the *Goldwater-Nichols* legislation stands as a proxy for these other influences.

To understand the policy issues behind the *Goldwater-Nichols Act* and related acquisition reforms, RAND reviewed literature on the political and economic environment

[9] See Hooper (1978) and Hone (1987) for a richly detailed, historical examination of this debate.

leading up to these initiatives as well as analyses of Defense acquisition problems.[10] To understand how the DoN implemented acquisition reforms and the effect of this implementation, Navy implementation guidance was reviewed (such as *SECNAV Instruction 5400.15C*) as well as DoD guidance (*DoD 5000.2*), and both former and current DoN officials were interviewed, including officials outside of the DoN deeply involved in implementing *Goldwater-Nichols* and related reforms, including the following individuals: two Secretaries of the Navy, an Assistant Secretary of the Navy/Research, Development and Acquisition (ASN (RD&A)), Chief of Naval Operations, a Deputy Chief of Naval Operations/Logistics, a Navy General Counsel and Deputy General Counsel, a Vice Chairman, Joint Chiefs of Staff, two Undersecretaries for Defense/Acquisition, Technology and Logistics (USD (AT&L)), Systems Commanders (NAVAIR and NAVSEA), Program Executive Officers (Ships, Tactical Air, and Submarines), and Program Managers.

RAND also interviewed former US Army and US Air Force senior uniformed and civilian officials to compare their implementation of *Goldwater-Nichols* in those services and the effect of other reform influences with that of the DoN's. A synthesis of their views are captured and presented.

There is an inherent limitation in this approach: in terms of sheer numbers, as not very many people were interviewed, and those interviewed provided their recollections of events that happened more than 20 years ago. That said, those that were interviewed were key players during the implementation and are reporting first-hand experiences. Also, because they were interviewed them separately, the authors were able to crosscheck one account with another. Furthermore, much of our discussion with them revolved around the effect of that implementation, and those interviewed are uniquely qualified to analyze the effect of the legislation on the processes and the implications of the divide between the requirements and the acquisition processes.

The Context of *Goldwater-Nichols*

The passage of the *Goldwater-Nichols Act* culminated in 1986—the result of operational, organizational and fiscal pressures that had been building for a number of years and, indeed, continued after the act was passed. These pre- and post-enactment events are important because they provide the context in which legislation was passed and implemented in the Department of Defense and the military services. This chapter briefly describes these forces and their significance in the crafting, passage, and implementation of the legislation.

Timeline

Figure 1 portrays the timeline of events that occurred before, during, and after the passage of *Goldwater-Nichols*. The timeline underscores several points. First, the forces that eventually called *Goldwater-Nichols* into being began decades before the act was passed. Second, these forces manifested themselves in quite different venues: operational performance of US military forces, the performance of the system that governed the acquisition of military weapons and weapons systems, and the behavior and practices of

[10] Analyses included the Defense Management Review (1989), the Packard Commission report (1986), the Joint Defense Capability Study (2004), the Defense Acquisition Performance Assessment Report (2006), and assessments conducted by the Center for Strategic and International Studies and the Government Accountability Office.

those who operated in that system. Third, a remarkable number of important events occurred in a five-year period, 1985-1990, that built an almost unstoppable momentum to ensure that long-standing issues would finally be dealt with in a systematic way. In this case, the effect of the whole far exceeded that of the parts. The sections below briefly describe the events that contributed to the eventual Perfect Storm.[11]

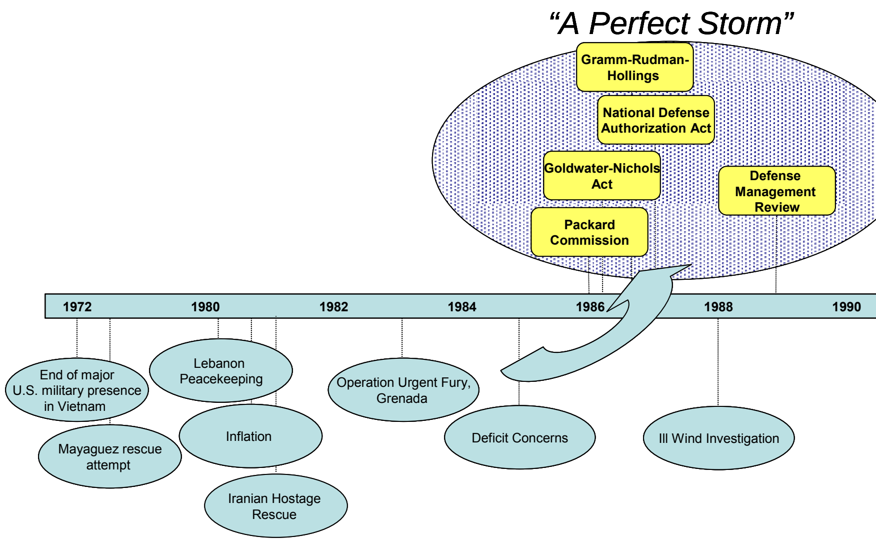


Figure 6. Events Contributing to the Context of Goldwater Nichols

The Context in Summary

The operational problems of the US military impelled Congress to change how the services selected personnel for assignment to joint duty and to change the entire military command structure. Poor acquisition outcomes and instances of fraud hardened congressional resolve to take steps in that arena as well. No single event necessarily led to the creation and passage of the *Goldwater-Nichols Act*. However, the combination of them, especially the ones that occurred in close succession in the latter half of the 1980s, contributed to the construction and passage of various legislation, the internal approaches regulation and to implementing that legislation and, subsequently, to the continuing resolve to ensure continual implementation of these various legislative provisions and regulations even in the face of emerging, unforeseen consequences. The next chapter describes how military acquisition was done before and after *Goldwater-Nichols* to provide a way to assess the nature of scope of the changes that legislation directed.

[11] The phrase "perfect storm" is used to describe an event in which a rare combination of circumstances exacerbates a situation drastically. It was also the title of a 1997 book and a 2000 movie adapted from the book.

The *Goldwater-Nichols Act* and the *National Defense Authorization Act of 1987*

This chapter briefly describes the main players in the enactment of *Goldwater-Nichols* and the key provisions of the act. It also discusses the *National Defense Authorization Act of 1987*.

Key Players

In 1985, Senators Barry Goldwater and Samuel Nunn brought many of the issues described in Chapter 2 to the attention of the Congress in a series of energetic floor speeches designed to garner political support for reform. An interesting and important perspective on staff roles was articulated in views expressed by Senator Goldwater. Nunn and Goldwater were joined in their efforts by Representative William F. Nichols in the House of Representatives. In Senate floor speeches, Senator Goldwater addressed what he perceived as the misguided financial focus of the military: "Our professional officer corps frequently behaves more like business managers than warriors." He also expounded on the issue of the civilian control in the military establishment: "[A] major problem created by the functional structure of OSD is that it encourages micromanagement of Service programs . . . it has the tendency to get over-involved in details that could be better managed by the Services." Two major points reflective of earlier reports on military operations were the following:

- First, there was the lack of true unity of command, and second, there was inadequate cooperation among US military services when called upon to perform joint operations (Anno & Einspahr, 1988).
- The preferred advice [from the Joint staff] is generally irrelevant, normally unread and almost always disregarded (US Congress, 1986).

Senator Nunn expanded on the issue of structural alignment: "The Office of the Secretary of Defense is focused exclusively on functional areas, such as manpower, research and development, and installations and logistics. This functional structure serves to inhibit integration of Service capabilities along mission lines, and thereby, hinders achieving DoD's principal organization goal of mission integration" ("Defense Organization," 1985).

Key Provisions of *Goldwater-Nichols*

The two Senators led the effort to draft the *Goldwater-Nichols Act*, which was signed into law in 1986; it made major changes in four broad areas: the chain of command and provision of military advice to the civilian leadership, the interaction of the military services, the personnel management of officers, and the acquisition of military equipment. The bill passed with wide bipartisan support, passing the House of Representatives, 383-27, and the Senate, 95-0. It was signed into law by President Reagan on October 1, 1986 (Goldwater-Nichols Act, 2010).

Each of the several key aspects of the *Goldwater-Nichols Act* addressed below had important ramifications for the DoD writ large, but their implementation in the DoN had consequences not fully understood at the time and as are more fully discussed in Chapter 5, in all probability not intended. The first two had an effect to disorient, the latter two served to disenfranchise.

The Chain of Command and Provision of Military Advice to the Civilian Leadership

In a key provision, military advice to civilian authority was streamlined and centralized in the person of the Chairman of the Joint Chiefs of Staff (CJCS), who became the principal military advisor to the President, the National Security Council and the Secretary of Defense. Previously, the chiefs of the individual services performed many of these roles. The CNO, for example, was the advisor to the President for naval matters. Further, the act established the position of Vice Chairman of the JCS, and increased the ability of the Chairman to direct overall strategy, while providing greater command authority to "unified" and "specified" field commanders.[12]

Interaction of the Military Services

The act also effected a sea change in service interactions by diminishing the role of the service chiefs and restricting the military services' operational control over forces, emphasizing their responsibility to support the Military Department secretaries in their Title 10 vote to "organize, train and equip" military forces for use by the combatant commanders. The services became "force providers" to the unified commanders (called the CINCs for Commanders in Chief). Their mission was to provide suitably trained and equipped forces to the CINC, which he requested through the Joint Staff. The CINC could be from any service, but he had authority to request assets from any service through the joint system (Nardulli, Perry, Pirnie, Gordon, & McGinn, 2002).[13]

These first two unraveled relationships, at least within the DoN, had developed and evolved for over 50 years. That is not to say that change is impermissible, but in this case, there was no clear sense of the nature of the new role to be played by the Service Chief; it was rather a product of what he wasn't to do.

The latter two went to the heart of the opportunity for operational personnel (officers of the line) to participate in acquisition matters and frankly, even if they desired to play a role. In Chapter 5, RAND will address the cultural manifestations of these changes.

Personnel Management of Officers

Another significant but more subtle change was the direction that an officer could not receive promotion to flag rank without having had a joint duty assignment.[14] Underlying this requirement was the perception on the part of lawmakers that the services were reluctant to send their best officers to joint duty assignments, preferring to keep them in their own ranks. Indeed, a joint duty assignment was perceived by many as a backwater and an indication that an individual's military career was not progressing well. Officers resisted going to such

---

[12] Unified commanders had geographical responsibilities, e.g., the Pacific area. Specified commanders had functional responsibilities, e.g., Strategic Air Command.

[13] CINC (now COCOM) requests go to the Joint Staff, which then coordinates the delivery of requested assets with the relevant service. Requests are not automatically approved, as the case of CINCEUCOM's request for Apache helicopters during the military operations in Kosovo during the effort to topple Serbia's Milosovic illustrate. All four services did not concur with the request, which was ultimately approved by the Secretary of Defense.

[14] Flag rank refers to generals in the Army, Air Force or Marine Corps or admirals in the Navy, so called because those achieving that rank are authorized a flag with the number of stars on it denoting their specific rank, e.g., a brigadier general's flag would have one star.

assignments and, if assigned to a joint billet, tried to leave them as soon as they could. Stipulating that promotion to flag rank could not occur without a joint duty assignment ensured that the services would be willing to assign their best officers to such billets.

Acquisition of Military Equipment

The *Goldwater-Nichols Act* also specifically addressed acquisition issues, giving sole responsibility for acquisition (as part of the assignment of several "functional" areas of responsibility) to the Secretary of each military department. For example, as it pertained to the DoN, Section 1045 stated:

> (C) (1) The Office of the Secretary of the Navy shall have sole responsibility within the Office of the Secretary of the Navy, the Office of the Chief of Naval Operations, and the Headquarters, Marine Corps, for the following functions: (A) Acquisition; (B) Auditing; (C) Comptroller (including financial management); (D) Information management; (E) Inspector General; (F) Legislative affairs; (G) Public affairs. (US Congress, 1986, 100 Stat. 1045)

As noted in this chapter, many of these functional responsibilities were already being performed by elements of the DoN Secretariat, unlike the situation in the other military departments. The word "sole" was to contribute to some interpretation of what was meant by the change. The act further stipulated that the Secretary designate a single organization within the Secretary's office—that is, a Service Acquisition Executive (SAE)— to manage the function of acquisition.

It is noteworthy that even after the legislative changes had been passed, Senator Nunn continued to debate the balance of service and civilian command and control. Relevant to the goal of this project in investigating the role of the Chief of Naval Operations is Senator Nunn's concern over barriers between the Military Department Secretary and Service Chief:

> Another area that was of concern is in the consolidation of the military and civilian staffs in the military departments. The conference agreed to consolidate several functions, such as acquisition, comptroller, inspector general, and legislative liaison, under the Secretaries of the military departments and directed that the service chiefs not set up competing bureaucracies within their staffs. In the conference, I was concerned that we not create an impenetrable wall between the staffs of the Service Secretary and the Service Chief. ("Defense Organization," 1985)

National Defense Authorization Act of 1987

The *National Defense Authorization Act of 1987* (US Congress, 1987) attempted to fill several policy concerns not addressed by the *Goldwater-Nichols Act*. For example, it addressed the problem of the excessive number of briefings program managers were required to give to get program approval, decreasing them to two: one to the Program Executive Officer and one to either of the DoD or Service Acquisition Executive (depending upon the acquisition approval threshold of the program). It also addressed the need for a streamlined reporting chain from PMs to PEOs to the Senior Acquisition Executive. These and other provisions both in this act and in legislation enacted in succeeding years—the latest being the *Weapon Systems Acquisition Reform Act (WSARA)*—give people proof that we are proceeding in a piece-meal fashion, patching together solutions episodically to address the crisis of the day. This approach has consequences as will be addressed in later chapters.

Acquisition Before and After *Goldwater-Nichols*

The primary focus of this paper falls on the Navy. However, changes that occurred in the Army and Air Force are evaluated as well because, in some instances, they responded to the legislation in ways that differed from the Navy (for purposes discussion of the Marine Corps, it falls under the DoN regulations applicable for the Navy), and those differences are illuminating. We first briefly viewed of the processes at the DoD level, and then follow with discussions about the Navy, Army and Air Force. In each of the latter cases, the discussion is guided by the change in service acquisition regulations, which is summarized in tabular form for each service.

OSD

Before the implementation of late 1980s acquisition reforms and the subsequent streamlining that resulted, each military department had an acquisition organization that included more stakeholders and more steps in the process. The DoD individual with responsibility for most functions that currently reside with the Undersecretary of Defense for Acquisition, Technology, and Logistics (USD (AT&L)) was the Undersecretary of Defense for Research and Engineering. Before 1986, the Secretary of Defense had overall responsibility for DoD Acquisitions. The Secretary of Defense and the Deputy Secretary of Defense presided over milestone decisions (*DoDD 5000.1*, 1986) similar to those that the current "Defense Acquisition Executive" or DAE is responsible for. Following the *Goldwater-Nichols* reforms, the most significant changes for the DoD-level acquisition were that many of the Secretary of Defense's acquisition decision authorities were delegated to the USD for Acquisitions (USD(A)) and the Director of Small and Disadvantaged Business Utilization now reported to the office of USD(A). Specifically, the USD(A) was designated as the Defense Acquisition Executive. This position is "the principal advisor to the Secretary of Defense on all matters pertaining to the Department of Defense Acquisition System" (*DoDD 5000.1*, 1987). Before the Deputy Secretary and various under secretaries (Research and Engineering, Policy), assistant secretaries (Acquisition and Logistics, Force Management and Personnel, Command, Control, Communications, and Intelligence, and Comptroller), and the Director, Operational Test and Evaluation were responsible for different aspects of the acquisition process. In response to the 1987 *Defense Authorization Act*, *DoDD 5000.1* (1987) also restricted the number of "management tiers" between the program manager (PM) and the DAE. These management tiers were designated as the Program Executive Officer (PEO) and the Service Acquisition Executive (SAE).

The Navy

Navy History and Culture

Each service has its own history and culture, and these profoundly influence how the services operate. In the case of the Navy, one of the signal differences appears in the titles of the chiefs of service. Both the Army and the Air Force are headed by an individual designated as the Chief of Staff, which implies an individual who oversees the workings of a staff and is himself a staff officer. The head of the Navy, however, is designated the Chief of Naval Operations, which implies an individual with operational command, and, indeed, this aspect of the CNO's office is deeply embedded in Navy history and practice. Of these three service chiefs, only the CNO held a position in which he was both heavily involved in service operational matters, from the time he was designated as the "Aide for Operations," and also ultimately served as the principal advisor to the President on such matters. The point is that the CNO historically focused on operational matters.

Up until 1966, the Navy was often informally referred to as "bi-linear," reflecting the fact that the CNO focused on the operational issues of the Navy with the Secretary of the Navy wholly responsible for the materiel component, including research and acquisition elements. The tension between the military and civilian leadership of the Department of the Navy over materiel matters was longstanding, and historically CNOs had pushed for a greater role in acquisition matters, to include lobbying the President.[15] Organizationally, the chiefs of the Navy's materiel bureaus reported to the Secretary of the Navy for all materiel matters. In 1966, the Secretary established the Navy Materiel Command (NMC), which was commanded by a senior admiral with extensive operational experience and who reported also to the CNO. This was a major change (one of the tectonic shifts alluded to above), because it placed the CNO directly in the line of materiel—including acquisition—issues. What was bi-linear had become uni-linear in that now both the CNO and the Secretary of the Navy had direct roles in the oversight of those organizations pursuing acquisition matters.[16]

Before the Storm

Acquisition before the passage of *Goldwater-Nichols* was governed by *SECNAV Instruction 4200.29A*, dated May 24, 1985, and entitled *Procurement Executives*. The wording in that instruction made the Secretary of the Navy the de facto "acquisition executive" referred to in subsequent legislation and regulation. It recognized his decision authority for acquisition matters pertaining to the Navy. The instruction designated the Assistant Secretary of the Navy, Shipbuilding and Logistics (ASN (S&L)) as the senior procurement executive and held him responsible for the performance of systems and for managing the career acquisition workforce. He was designated as the focal point for procurement and the logistical systems necessary to support the systems the Navy procured.

The instruction directed the CNO to support the ASN (S&L) in carrying out his duties. During this period, the three major warfare branches of the Navy—air, surface, and submarine—were each represented by a three-star admiral on the OPNAV staff who had direct contact with the systems commanders for material in their warfare area. Each had program officers who maintained liaison with the program managers reporting to the system commanders.

The CNO played directly in the procurement process in multiple ways. His most direct role was reviewing all programs going to the Secretary of the Navy for decision. The mechanism for this review was the CNO's Executive Board (CEB, pronounced "KEB"), on which the Vice Chief Naval Office also sat. But, as is discussed below, the system

---

[15] In a memorandum to the Secretary of the Navy dated March 2, 1934, President Franklin Roosevelt, himself a former Assistant Secretary of the Navy, said:

> In my judgment he [the President] would too greatly delegate this power [control of Naval Administration] if he delegated to the Chief of Naval Operations the duty of issuing direct orders to the bureaus and offices. . . . By this, I mean that the Chief of Naval Operations should coordinate to [sic] all repairs and alterations to vessels, etc., by retaining constant and frequent touch with the heads of bureaus and offices. But at the same time, the orders to Bureaus and offices should come from the Secretary of the Navy.

[16] The CNO always had influence in this area by virtue of his control over promotions and assignments, but with the organizational realignment he had directive authority.

commanders also reported to him through the CEB, giving the CNO another opportunity to engage in materiel management.

While the Systems Commanders reported directly to the four-star commander of the NMC, they also had reporting responsibilities to the CNO and two ASNs (S&L and RE&S) in their areas of responsibility, while coordinating matters through the NMC. The three warfare branch vice admirals on the CNO's staff identified above did the planning and programming for their individual warfare area systems and coordinated with the NMC and the Systems Commands. Programming reviews were carried out through a CNO chartered board. The program managers reported to the Systems Commanders through their functional flag officers. Figure 2 graphically depicts these complex relationships.

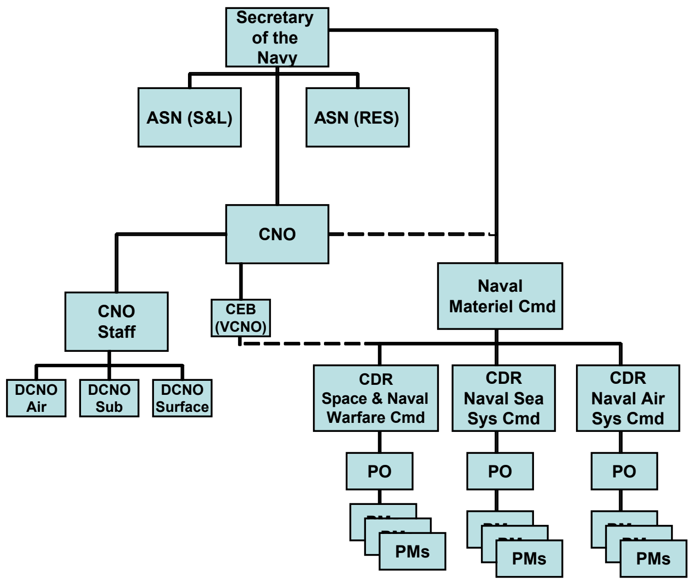


Figure 7. Navy Acquisition before *Goldwater-Nichols*

Although not codified in Navy instructions until later, in 1985, the Secretary of the Navy abolished the NMC, which was another of the tectonic shifts that occurred in Navy acquisition. The Chief of the Naval Material Command was a four-star officer of the line who brought senior level credibility to the materiel establishment and buffered the materiel community when needed. The disestablishment of the NMC eliminated this buffer and started the erosion of the operational credentials of the materiel community that occurred over time and the bona fides of decisions proposed by them. It has been argued that this very ability to argue for differing perspectives was also the proximate cause for the disestablishment of the NMC as was the fact that the NMC comprised another management layer, slowed the decision process, and ran counter to the views expressed by the Packard Commission on lines of authority.

Acquisition in the Aftermath of the Storm

The DoN implemented *Goldwater-Nichols* in two steps. First, it designated the Secretary of the Navy as Acquisition Executive. Second, it attempted to use as many of the existing processes as possible to accomplish the act's intent. Both steps drew fire from the Comptroller General (Conahan, 1989). The DoN's implementing instruction did incorporate language from the *Goldwater-Nichols Act* regarding establishment of a single organization within the SECNAV's office to assume authority over the acquisition system. In doing so, the instruction stated that the CNO and Commandant of the Marine Corps "will execute their responsibilities through the resource allocation process and their input to the acquisition decision-making process."

Implementing the *Goldwater-Nichols Act* imposed important changes on the Navy's acquisition process. In the view of a former Secretary of the Navy, the law simply allowed too much latitude in implementation. For example, a former General Counsel for the Navy and a former ASN RDA interpreted the provision that assigned authority for the acquisition process to the service secretaries as entirely excluding the service chiefs from the acquisition process. However, the first CNO to operate under the new provision said that he had been unclear about his role in the acquisition process. He added that he had been advised not to get involved in acquisition decision-making, but felt that he had to, and had ignored the advice because he was being held "accountable" by the Congress for acquisition failures, such as the A-12 aircraft program.[17]

Different interpretations are also reflected in the different forms that implementation took among the Navy, Army, and Air Force. Each of the military departments implemented the law differently and all came under fire from the Comptroller General for various reasons. The common theme of these attacks was that each service had PEOs (Program Executive Officers) reporting to the applicable military Systems Command structure. Following the GAO report, each severed the PEO structure from the Systems Commands.[18]

Following the passage of *Goldwater-Nichols*, the Navy issued a new instruction, *5430.96*, dated August 4, 1987 (and a companion instruction dated August 5, 1987). The instruction designated the Secretary of the Navy as acquisition executive for the Navy. Thus, he held not only program decision authority, but, as Acquisition Executive, he was also responsible for the acquisition process. In support of the Secretary in that role, the ASN (S&L) reported directly to him for acquisition matters. The ASN (S&L) was charged with responsibility for supplying, equipping, servicing, maintaining the Navy's equipment. He had responsibility for acquisition production and support for the Navy and the Marine Corps and to "provide such staff support as the CNO and [the Commandant] each consider necessary." The second instruction, *5430.95* and dated one day after *5430.96*, pertained to the ASN (R,E&S). He was responsible for all DoN acquisition except shipbuilding and conversion. He also had responsibility for matters related to research and development. In this role, the Chief of Naval Research reported to him. These instructions also codified the elimination of the Naval Materiel Command.

The most significant change occurred in the role of the CNO. The new instruction divested him of acquisition responsibilities. Rather, *5430.96* charged him with supplying, servicing, maintaining, outfitting and logistic functions. And *5430.95* directed him to formalize and prioritize requirements, conduct test and evaluation, prioritize research, development, test and evaluation, and directed him to provide advice and support to the Secretary. Thus, he became responsible for determining what equipment the Navy needed but not acquiring it. That function was now located wholly in the Secretariat.

---

[17] The CNO had to deal with the consequences of the unraveling of the A-12 program. In our interview with him, he expressed the view that Congress was demanding answers from him on a range of issues with regard to the A-12 replacement program, the F-18 E/F and that given what had occurred in the A-12, he had to be aware and involved in aspects of program decision-making, both to represent Navy interests and concerns before Congress and to be able to defend Navy resources.

[18] The Army and the Air Force later gained permission from the Office of the Secretary of Defense to place their PEO structure back under their Systems Command.

Under the provisions of *5430.96*, the Systems Commanders now reported to the DoN Acquisition Executive for all PEO matters under the direction of the ASN (S&L). Similarly, the PEOs also reported to the ASN (S&L).

Figure 3 depicts these changes. The bar and circle symbol indicates eliminations, in this case the Naval Materiel Command and the dotted line between the systems command and the CEB, which still existed but had lost any approval authority. Note also that the PEOs no longer report to the System Commanders. They now report directly to the AE, the Secretary himself at this juncture.

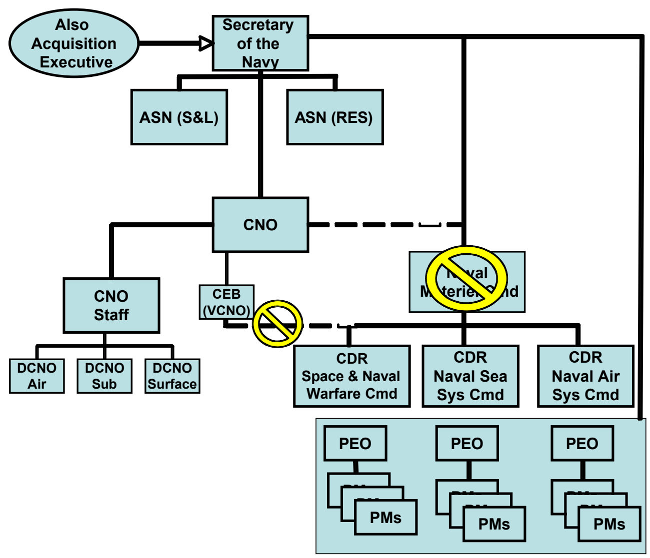


Figure 8. Navy Acquisition in 1987

Passage of *Goldwater-Nichols* did not lay to rest all acquisition issues nor were all applicable organizational and process changes implemented immediately. As indicated above, John Lehman, who was Secretary of the Navy from 1981 to 1987, designated himself as the Acquisition Executive. In the view of the Secretary of Defense, that did not accord with the intent of the legislation, and the Secretary and his Deputy pressed the DoN to designate an individual Assistant Secretary as the Acquisition Executive, eventually directing the DoN to make that change. An August 1991 instruction (*5400.15*), codified the Secretary of Defense's direction, providing that the Assistant Secretary of the Navy (Research, Development and Acquisition) (ASN RD&A) had the full-time role in development and procurement of systems, ensuring that operational requirements were transformed into executable processes. This change was another major shift in the Navy's acquisition processes

The 1991 instruction underscored the CNO's role in determining requirements and establishing a relative priority among them. It also indicated that he might be assigned responsibility for research and development matters and for operational test and evaluation, but it was clear that he could not assign himself a role in those areas. This instruction also codified the elimination of the "warfare branch admirals," and their relationships with the material establishment of the Department.

The instruction charged the systems commanders with the management of programs other than those assigned to a PEO and directed them to provide support services to the

PEOs. For their part, the PEOs were directed to report to the ASN (RD&A), and the instruction directed program managers to report to the PEOs. The reporting line from the PEOs now runs directly to the newly named ASN (RD&A) rather than to the Secretary of the Navy. The Secretary retains his approval powers, but not the direct management of the process, for those decisions he is empowered to make.[19] Figure 4 shows this continuing evolution of the Navy acquisition procedures. Key changes shown in the figure include both the changes shown in Figure 3 (elimination of Navy Materiel Command, the reduced role of the CEB, and the formal designation of the Secretary of the Navy as the AE) as well as some additional ones. A key one is that the AE is now the ASN (RD&A), and the PEOs report to him rather than to the Secretary. The position of the ASN (S&L) has been eliminated, as have the warfare branch admirals on the CNO's staff (again, shown by the bar and circle symbol). The chain of acquisition approval flows directly to the ASA (RD&A) rather than to the Secretary.[20]

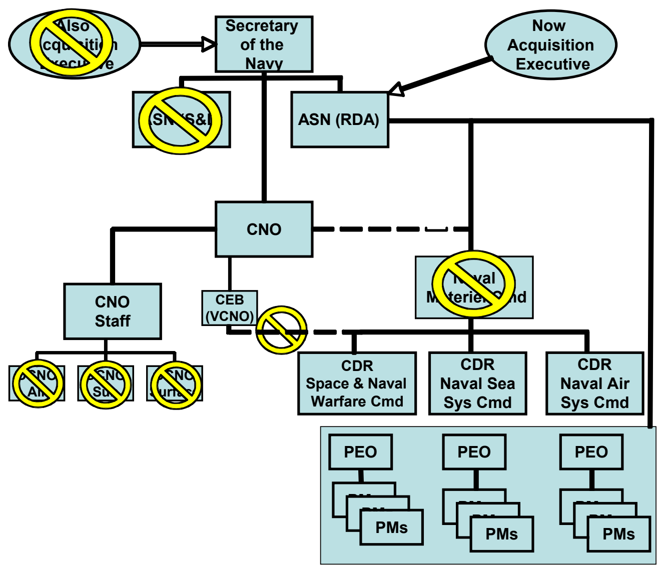


Figure 9. Subsequent Changes to Navy Acquisition Procedures

Three instructions were published subsequent to *5400.15* in 1991: *5400.15A* in 1995, *5400.15B* in 2005, and *5400.15C* in 2007. None changed the major responsibilities of the Secretary, the CNO or the Acquisition Executive, although they elaborated on some of the functions. For example, *5400.15B* designated the CNO as the principal advisor to the Secretary in the allocation of resources to meet programming and budget processes. In essence, the instruction conferred on the CNO the responsibility to advise the Secretary on what programmatic priorities to assign to the requirements, the development of which was his primary responsibility. He still stood outside the procurement process. *Instruction 5400.15C* charged the CNO to analyze alternatives in conjunction with the ASN (RD&A) before the development phase of a weapon system.

---

[19] Decisions about programs that cross certain thresholds in terms of dollars for research and development and procurement must be made at the Department of Defense level. These are referred to as ACAT 1 decisions.

[20] For a relatively few years, the Undersecretary of the Navy was designated as the AE, but the duties were eventually assigned to the ASA (RD&A), where they remain today.

Instructions after 1991 also elaborated on the responsibility of the Systems Commanders and the PEOs. For example, *5400.15A* stipulated that the Systems Commanders would exercise the authority of the Acquisition Executive to supervise acquisition programs directly and, notably, reporting to the CNO for execution of programs that *were not* development or acquisition projects. Thus, the wall between the CNO and the procurement process remained intact. PEOs were authorized to act for and exercise the authority of the Acquisition Executive with respect to their assigned programs and maintain oversight of the cost and schedule performance.

Summary of Key Changes in Navy Acquisition

The process of acquiring Navy equipment changed dramatically between 1966 and 1991. Some changes were more gradual than others. The creation of the "uni-linear" Navy took decades, and crystallized with the establishment of Navy Materiel Command in 1966. Its subsequent dissolution in 1985 marked an equally significant shift. However, most key changes occurred as part of the perfect storm of events that centered on the *Goldwater-Nichols* legislation. While the effects of that legislation rippled beyond the procurement process and within the process itself, the most critical ones were the roles defined for the Secretary of the Navy, the Assistant Secretaries, and the CNO. The effect on the CNO was, arguably, the greatest, since the result was his defined exclusion from the procurement process. The Secretary retained his approval power but was forced to delegate responsibility for the process to one of his Assistant Secretaries and subordinate elements of the Systems Commands. The ASN (RD&A) assumed responsibilities previously carried out by the Secretary, even though, as one of the Secretaries of the Navy interviewed opined, only the Secretary had the responsibility and gravitas in all elements of the decision process (requirements, resources and politics) to be able to perform the job well. The creation of the PEOs elevated their importance and visibility in the process while eliminating much technical senior oversight.

The Army

Before the Storm

The Secretary of the Army is responsible for all activities occurring within the department, including acquisition. Indeed, Army acquisition policy (*AR 70-1*) was either signed directly by the Secretary of the Army or "by order" of the Secretary. Before implementation of *Goldwater-Nichols* and the 1987 *Defense Authorization Act* provisions, the Secretary was supported by an assistant secretary who was almost always designated as the Army Acquisition Executive (AAE). In 1984 before the acts, the role of the AAE had been assigned to the Assistant Secretary of the Army for Research, Development, and Acquisition (ASA (RD&A)). At the time, this individual served as an advisor to the Secretary, chaired the Army Systems Acquisition Review Council (ASARC), and decided whether acquisition programs were ready to progress past key milestones. It appears, however, that the ASA (RD&A) did not directly supervise acquisition programs or personnel, as is currently the case. That duty resided with a uniformed officer on the Army staff, the Deputy Chief of Staff for Research, Development and Acquisition (DSCRDA). Program reviews, officer assignments and program management assignments all emanated from the DSCRDA. This three-star general had the staff to manage the acquisition process and worked with the ASA (RD&A), who had a very small staff.

The executing authority for acquisition programs resided with the Development and Readiness Command (DARCOM). The Army materiel commands, similar to the Navy systems commands, worked for the DARCOM (DARCOM's successor is the Army Materiel Command or AMC). DARCOM's Commanding General reported to the Chief of Staff of the Army (CSA) and the Secretary of the Army. Figure 5 depicts these relationships. Thus, even though the Secretary of the Army was the acquisition decision-maker and he had an assistant secretary who oversaw the acquisition system, by practice, the Chief of Staff of the Army, through DCSRDA and DARCOM, had the greatest influence over acquisition decisions. The Army's ASARC was the body that performed the highest level review function before a Secretary of Army decision (or recommendation in the case of a decision on a given program being made by the Secretary of Defense). The ASA (RD&A) and the Vice Chief of Staff of the Army co-chaired this council.

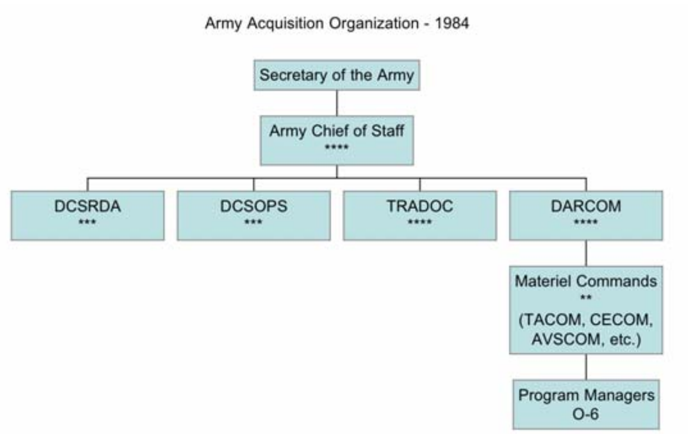


Figure 10. Army Acquisition before *Goldwater-Nichols*

After the Storm

Following the *Goldwater-Nichols* era reforms, the Army reissued its acquisition regulations four times (in 1988, 1993, 1997, and 2003). In 1988, the DCSRDA position was eliminated and the ASA (RD&A) was designated as the "Deputy Army Acquisition Executive and provided 'principal secretariat support' to the Acquisition Executive" (the Secretary of the Army). The regulations issued in 1993 implemented the first round of structural changes that are most representative of the changes that have endured to the present day. Because the *Goldwater-Nichols Act* required streamlined acquisition chains of command and limited "outside" influence over acquisition activities, the acquisition chains of command were shortened to three levels for service-managed acquisitions. As mentioned earlier, the Secretary of the Army exercised overall responsibility for activities within the department. This revision to Army Regulations saw the delegation of AAE responsibilities to the Assistant Secretary level. The AAE role was initially assigned to the ASA (RD&A), and later the name was changed to the ASA (AL&T) (Acquisition, Logistics, and Technology). Following the reforms, the AAE was more centrally positioned in the Army's acquisition process. One key aspect of that involved managing and supervising PEOs and PMs, a function that before *Goldwater-Nichols* had been performed by the DCSRDA.

Also, as was the case with the Navy, the service chief and the deputy chiefs of staff were no longer directly engaged in the acquisition process. They retained their responsibilities to produce requirements for the acquisition of new materiel and to develop the Program Objectives Memorandum (POM), which allocated funding to the priorities set by the President, Secretary of Defense, Secretary of the Army. But with regard to core acquisition management functions, their tasking was only to state requirements and support the PEOs and PMs. The one exception to this was the Vice Chief of Staff of the Army who, continued to co-chair the ASARC with the ASA (RD&A). In this role, the Vice Chief was able to represent the operational Army throughout the materiel acquisition process. Other commands and individuals such as the DCSOPS and TRADOC, now, the Army Materiel Command, which had replaced the DARCOM continued to report to the Army Chief of Staff. However, the principal acquisition functions that they once managed were reorganized into a different chain of command. Figure 6 depicts these changes.

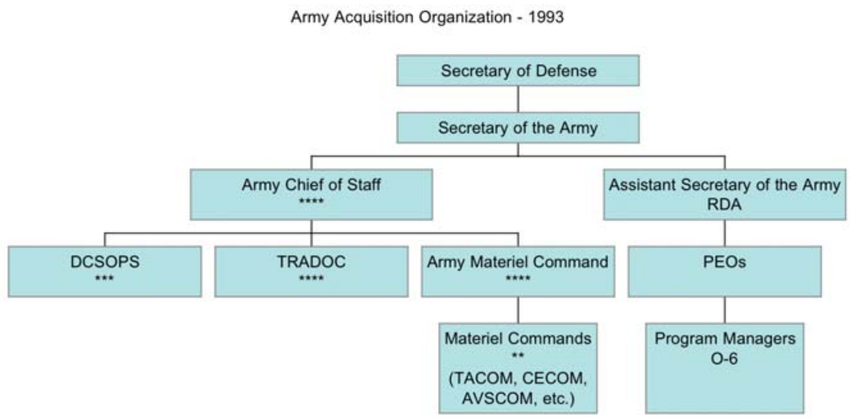


Figure 11. Army Acquisition in 1993

Summary of Key Changes in Army Acquisition

As with the Navy, the primary effects of the *Goldwater-Nichols* era reforms were to remove the Chief of Staff of the Army and his supporting organizations such as deputy chiefs and the Army Materiel Command and subordinate materiel commands from playing a direct role in the acquisition process. As will be seen in the next section, a similar pattern emerged in the Air Force.

The Air Force

Before the Storm

The evolution of responsibilities for the acquisition of major systems and components within the Air Force is similar to that of the Army's.

As was true with the other military departments, the Secretary of the Air Force is responsible for all activities occurring within the department, including acquisition. Throughout the period under examination, the Secretary was supported by an assistant secretary who was designated as the Air Force Acquisition Executive (AFAE). In 1986, this was the Assistant Secretary for Research, Development, and Logistics (SAF/AL). Later, the role of the AFAE was assigned to the Assistant Secretary of the Air Force for Acquisition (SAF/AQ). The Assistant Secretary also chaired the Air Force Systems Acquisition Review Council (AFSARC), which was the principal board that advised the Secretary. (According to the 1986 instruction, the Secretary did not delegate his role as the milestone decision authority.) Several members of the Air Force Chief of Staff's general staff were also assigned as members of the AFSARC. These members were the Vice Chief of Staff, Deputy Chief of Staff (DCS) Logistics & Engineering, DCS Research Development, and Acquisition

(RD&A). It is interesting to note that HQ USAF issued Program Management Directives (PMDs) that define the scope of the program being procured, and provide program direction and guidance. However, the implementing command appears to have had great leeway.

The executing authority for acquisition programs resided with the "implementing command," which was designated on a program-by-program basis by the AF Headquarters acquisition staff. One of the implementing commands named directly within the 1986 regulation was the Air Force Systems Command (AFSC). In its role as an implementing command, it was responsible for accomplishing program executive supervision in much the same way that PEOs do currently, albeit over a much larger set of programs. *AF REG 800-2*, dated September 16, 1985, stated that the "designated Line Authority for major decisions during the acquisition of weapon systems typically include the Secretary of Defense, Secretary of the Air Force, and the Commander, Air Force Systems Command." However, it also stated that, under "Responsibilities of the HQ USAF", that HQ issued Program Management Directives that established programs, provided program guidance and direction, designated implementing commands and issued the Justification of a Major System New Start (JMSNS) to begin the acquisition process.

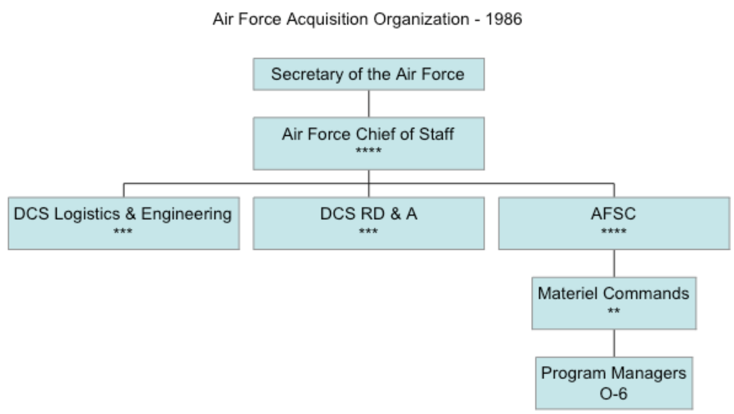


Figure 12. Air Force Acquisition before *Goldwater-Nichols*

After the Storm

Following the *Goldwater-Nichols* era reforms, the Air Force reissued its acquisition regulations five times (in 1990, 1993, 1994, 2005 and 2009). The 1994 instruction was the first to mention of the AFAE as directly managing acquisition programs and personnel. These post *Goldwater-Nichols* instructions make no mention of acquisition responsibilities within the AFHQ general staff until the 2009 reissues, when the Deputy Chief of Staff for Operations, Plans, and Requirements (HQ AF/A3/5) is tasked with "collaboratively work[ing] with the acquirer, tester, sustainer, and other key stakeholders in developing operational capabilities requirements documents."

With regard to the materiel commands, after 1994, the instructions task the Air Force Materiel Command (AFMC), (AFMC was the successor command, which absorbed in 1994 both the Air Force Systems Command and the Air Force Logistics Command, eliminating one four-star position) with formally and informally advising and assisting the AFAE, PEOs, and PMs. Figure 7 depicts these changes.

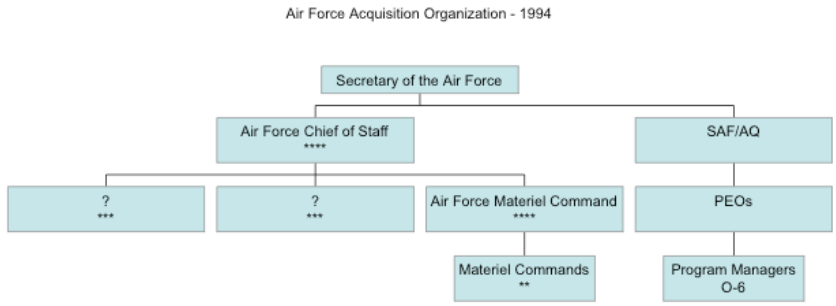


Figure 13. Air Force Acquisition in 1994DCS Plans & Operations, DCS Programs & Resources

A Comparison of the Before and After by Department

As stated earlier, the Navy's culture differed significantly from that of her sister services and was reflected in the organization of the DoN and its management structure. The CNO viewed the landscape in operational terms befitting his title of Chief of Naval Operations. That is, the original creation of the bi-linear Navy did not engage the CNO immediately in the administration of the various material "bureaus" that handled the acquisition and logistical functions supporting the Navy. The Bureaus of Ships, Ordnance, Aeronautics and Supply and Accounts were some of the bureaus that handled such functions and reported to the Secretary of the Navy. In 1966, with the creation of the Chief of Naval Material, subsequent CNOs played a greater role in the management of and production by the material establishment. Even so, for individual CNOs who had grown up in the operational world and particularly those without significant Washington, DC, experience, dealing with material (including acquisition) matters was somewhat foreign. Furthermore, because the DoN included two services, the Navy and the Marine Corps, the DoN Secretariat tended to act in greater scope from those of the Army and Air Force.

The term "Chief of Staff" meant chief of staff to the Secretary of the Army and Secretary of the Air Force. As no such office existed in the DoN, and its Secretariat was responsible for a broader set of functions, which in the other Military Departments were performed in the service headquarters staffs. A couple of examples in central departmental management functions (finance and contracts) are illustrative. First, the Department of the Navy's budget function reported to the Assistant Secretary of the Navy, Financial Management (ASN (FM)), not to either service chief. In its functions, that budget office excited ecumenical control the finances of the two DoN sister services and clearly worked for the Secretary. While in the planning and programming processes, the two services built their POM for the Secretary to approve, the subsequent budget fell under the management of the Secretary through the ASN (FM). In a second example, contract award approvals were managed by another DoN Assistant Secretary, depending on the item being procured. The contract award justification, called the Determination and Findings (D&F), had to be reviewed and approved by the appropriate office in the Navy Secretariat before contracts were awarded, which meant they played a major role in acquisition.[21] A last point is that the

[21] The D&Fs are written, signed, legally binding statements, submitted by an employee, to explain/justify the method and logic that he/she used to select material, services, or suppliers when committing federal, state, or district funds for purposes of procurement of materials or services.

DoN Secretariat was staffed to perform these regulatory and statutory functions and, as a result, was larger than those found in the other military departments. For example, the ASN (FM) staff, including the Comptroller, at times exceeded several hundred people.

A second major difference between the Navy and her sister services was the manner in which the staffs were structured for decision-making. The other services tended to look upon issues through functional ions. That is, when those services addressed issues, their reviews occurred at the functional level of manpower, logistics, modernization, etc. In the Navy, responsibility was held by three-star admirals who controlled surface, submarine and aviation portfolios. These three-star admirals also had a major voice in the requirements determination and acquisition processes before *Goldwater-Nichols* and had direct relationships with their three-star counterparts in the Navy's systems commands. Issues that arose between statement of requirements and the ability to develop acquisition programs could be resolved at this three-star level. Concurrent with the *Goldwater-Nichols Act* (but for reasons different from the passage of the act) and its movement of the acquisition function more fully into the DoN Secretariat, the Navy abolished those three-star billets and reduced the functions they performed to the two star level, impeding discussions with the systems commanders (who concurrently were removed from the acquisition chain because the new PEOs reported directly to the ASN (RD&A)). The Army and Air Force did not have those concurrent changes in their structures although the removal of the PEOs from the Systems Commanders also occurred in those Departments. It is interesting to note that both the Army and Air Force later requested and received waivers to allow the PEOs to report to what was in essence the equivalent to the Navy' Systems Commanders.

Another major distinction was that while the Army and Air Force had Systems Acquisition Review Councils (both before and after *Goldwater-Nichols*) the Navy did not. Programs wound their way through a set of systems command reviews, two-star review in the staff of the Chief of Naval Operations and then through the CNO-chaired CEB meeting and, finally, on to the Secretary of the Navy for decision.

The ASARC and AFSARC boards were co-chaired by the service vice chiefs, both before and after *Goldwater-Nichols*. Thus, the service chief had an important representative in councils dealing with acquisition decisions. Furthermore, based upon our interviews with Air Force and Army retired three-stars, the principal deputy position was generally filled by a senior uniformed executive (typically a three-star General) in each of the secretariat acquisition offices and played major roles in both the selection of acquisition personnel (including the management of the acquisition workforce) as well as the distinct function of briefing the Service Chief on all matters of acquisition interest prior to his attendance in any structured meeting with the Department's Secretary on such matters.

With the passage and eventual implementation of *Goldwater-Nichols*, the Navy acquisition programs no longer went through the following: systems commanders, two-star CNO staff board, and CNO Executive Board.

In its place, a Navy Program Decision Meeting (known at the NPDM) was created and chaired by the ASN (RD&A). While CNO staff flags were invited to the meetings, they were held at the behest and schedule of the ASN (RD&A), and our interviews indicated that they were poorly attend by Navy flag officers. This led to the perception of isolation of the service chief and his staff from those functions.

Thus, as shown above, the *Goldwater-Nichols Act* and the subsequent changes in the CNOs staff had greater ramifications for the Navy than it did for the other services.

How Navy Implementation Affected Acquisition

This chapter describes what we see as four major consequences that resulted from the manner in which the Department of the Navy implemented the *Goldwater-Nichols Act*. The first is the exclusive civilian control of the acquisition process with military disenfranchisement. The second is the loss of blended workforce. The third is the separation of the "line" (naval officers who have operational assignments that lead to their promotion and success in the Navy). The fourth is the continuing search for rebalance and the unintended consequences of the manner in which the current DoN leadership has chosen to attempt to re-integrate the operation naval officers (line) into the acquisition process.

Increasing Civilian Control over the Acquisition Process: Constructing an Impenetrable Wall

Senator Nunn stated during the conference leading up to enactment of *Goldwater-Nichols* that he had been "concerned that we not create an impenetrable wall between the staffs of the Service Secretary and the Service Chief" (Nunn, 1985, S12651). In our interviews with senior Navy and OSD officials directly involved in implementing the *Goldwater-Nichols Act*,[22] most had been concerned about this possibility or they became concerned after its passage. In fact, of the twenty-five former and current civilian and uniformed officials (including those in the Air Force and Army) interviewed, all but two had no doubt that a wall had, in fact, been created between operational officers and acquisition officials.

In terms of our senior-level interviewees, only one—a former USD AT&L—believed that a minimal amount of separation between military and civilian leadership resulted from implementation of *Goldwater-Nichols*. Moreover, he believed this separation had been constructive, contributing to creative tension and leading to a more efficient use of resources. In short, he believed the service chiefs could still influence acquisition decisions. However, the remaining interviewees were much less sanguine about the outcome. Similarly, an Air Force civilian executive with a rich background in acquisition matters did not believe that the military leadership was disadvantaged by the separation of the military requirements community from their acquisition brethren.

According to a former Principal Deputy ASN (RD&A), the acquisition community eliminated roles in the acquisition process traditionally filled from the Office of the Chief of Naval Operations (OPNAV) staff. A former CNO reported that he himself felt excluded from the acquisition process, as did all senior officers of all ranks and career fields who were interviewed. One former USD AT&L, who came to believe that the service chiefs and combatant commanders were now too far removed from the entire acquisition process, thought it essential that Service Chiefs become more involved in procurement planning, especially in helping to set realistic performance requirements and to make trade-off decisions during program development. Both Under Secretaries of Defense (AT&L) interviewed believed that establishing a four-star Vice Chief as co-chair of the Service Acquisition Board could overcome the growing divide between a military-based requirements process and a civilian-based acquisition process. In this scenario, the Vice

[22] In this instance, interviewees included a former CNO, a former Secretary of the Navy, a former Navy General Counsel, a former Asst. Secretary of the Navy (Research Development and Acquisitions (RDA), and two former Under Secretaries of Defense (AT&L).

Chief's role would be similar to that of the Vice Chairman of the Joint Chiefs in his role as co-chair of the Defense Acquisition Board.

The DoN leadership is not blind to this problem. It has attempted to breakdown the barriers between the CNO's staff and the Secretariat with regard to requirements and acquisition. The Navy Gate System is the latest effort to link the acquisition process and requirements process, initiated in a *SECNAV Notice 5000* in February 2008. The system established a six-gate process (the circled numbers in the Navy/USMC level) in which each gate represents a formal decision point at which the costs and benefits of a particular weapon system program are evaluated (see Figure 9).

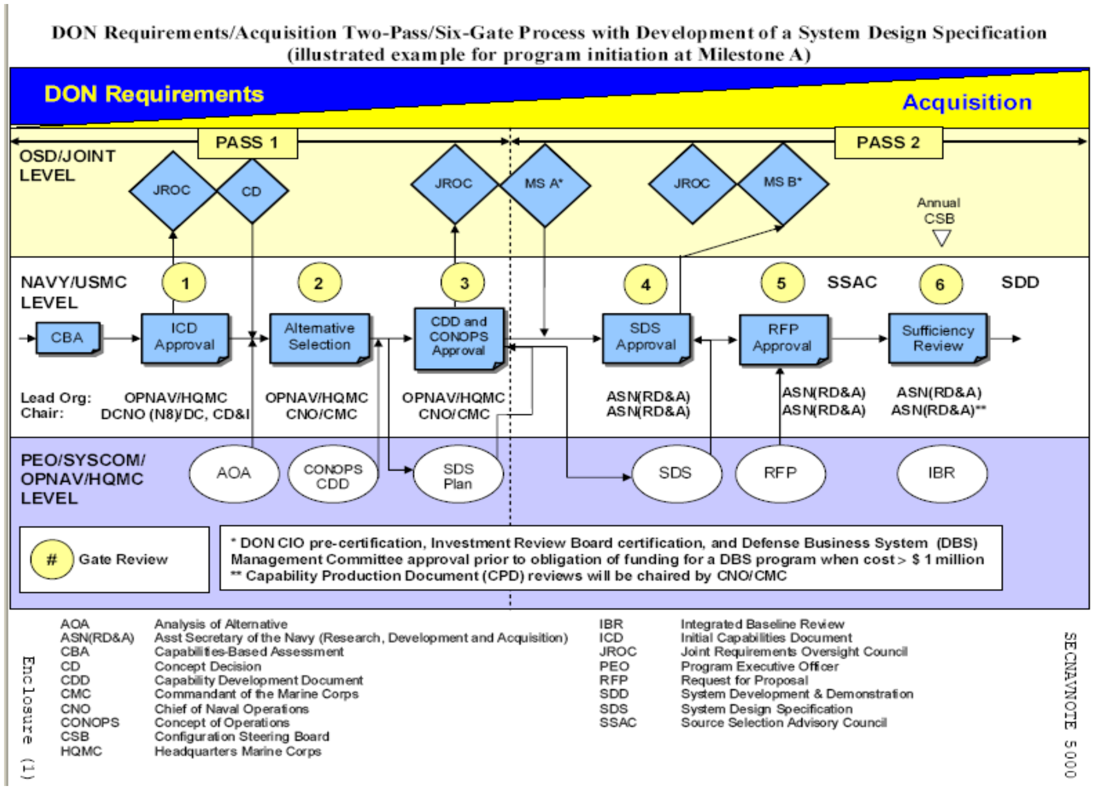


Figure 14. Navy Gate System
(Source: SECNAV Note 5000)

As seen on the chart, a vertical dotted line separates the first three decision gates from the last three. The first three gates are supposed to be managed on the requirements side[23] and the following three gates are managed on the acquisition side.[24] That the dotted line reinforces the notion of separation in Senator Nunn's "impenetrable wall."

Blended Workforce and the Engagement of Operational Officers in the Business of Acquisition

A principal motive of the *Goldwater-Nichols Act* was to improve our military's ability to fight in a more joint manner; consequently, not only weapon systems, but also officer experience and training must now include joint duty and considerations. All of the senior-level officials we interviewed reported that they had, at the time *Goldwater-Nichols* was implemented, believed there was a need for better communication among the military departments and more joint collaboration in operations. However, an unintended consequence of requiring officers to serve in a joint duty assignment to achieve flag or general officers rank was the migration of line officers away from the acquisition process because of the pressure of satisfying additional demands during a career whose length did not expand to accommodate the additional demands.

This migration became a particular concern in the Department of the Navy as it had maintained over time a blended workforce in its acquisition processes. Before *Goldwater-Nichols*, there existed a blend of naval and marine officers along with technically-oriented civilians worked across the material establishment. Program offices, systems commands, laboratories and field activities were generally managed by military officers who rotated into the material establishment from operational billets and brought with them a wealth of real-world fleet experience. Coupled with them were a group of highly skilled engineers and scientists who, working together developed and procured the nation's naval weapons systems.

Upon the implementation of *Goldwater-Nichols*, the creation of an acquisition workforce resulted in a formulaic career path for those whose intent was to work in acquisition. While it created certain incentives for the civilian element of the workforce, it also created significant differences between civilian and uniformed workforces. First, the civilians involved in acquisition and the uniformed personnel involved in acquisition had completely different chains of command and, consequently, a different performance evaluation and promotion structure. The new workforce structure also demanded new educational mechanisms, to prepare individuals for careers in the acquisition workforce. The Defense Acquisition University (DAU) was established with a heavy civilianized structure and outlook. The agility of acquisition was slowed by this new institutional training and the demand for military personnel to participate in the DAU courses heavily affected military career assignment and rotation.

Exacerbating the "civilianization" of the workforce was an unintended consequence of *Goldwater-Nichols* emphasis on joint warfighting to satisfy promotion requirements. Before *Goldwater-Nichols*, officers had more time to rotate through positions related to both the operational realm and the material management process, giving those officers a deeper

---

[23] Specifically, they are managed by the Deputy Chief of Naval Operations, for Integration of Capabilities and Resources (DCNO (N8))/DC and DC&I, CNO/Commandant, Marine Corps (CMC) in OPNAV/ Headquarters, US Marine Corps (HQMC).

[24] Specifically, they are managed by the Asst. Secretary of the Navy (RD&A).

understanding of the civilian side of the acquisition process. With the rigid requirement of joint duty service, however, officers no longer had time to rotate between operational duty assignments and material management assignments if they wanted to achieve flag or general officer rank in an operational role. Furthermore, those who now chose to devote their energies to acquisition saw their operational experience decline in comparison to the counterparts who served only in line assignments, which meant they lost some of their credibility when it came to weighing in on the value of a particular performance requirement, for example. As the number of officers serving in acquisition roles decreased, a sense emerged of the acquisition process "belonging" to the largely civilian material establishment, not the operations line community. Our interviews with senior Army and Air Force officers echoed these observations about their own services. Almost to a person, our interviewees remarked on the need to create an incentive for senior line officers to serve in acquisition roles. That is not to say that there is not a role for engineering restricted duty officers in the acquisition workforce. However, a blended workforce should contain officers with warfighting training and perspective to ensure a rich mix of talent is available to the acquisition leadership.

Unintended Consequences

In this chapter, the Navy Gate System is described. In that example, a large structured set of meetings and briefings needed to be established because the DoN's acquisition instructions explicitly left the uniformed Navy out of the processes. That said, it still stands. It should be noted that given Senator Nunn's explicit concern over such an outcome, this is also no unintended consequence.

In *SECNAV INST. 5400*, the applicable reference to the CNO and CMC is that the ASN (RD&A) "shall provide such staff support each consider necessary to perform his duties and responsibilities." There is no mention of any other responsibility for the service chiefs. When the act was passed, our interviewees indicated that the uniform Navy offered a three-star deputy to the ASN (RD&A), but that was refused and a senior executive was installed in that position. Throughout the ensuing twenty years, a mix of SESs and officers from one-to-three star rank have filled that position. But the DoN acquisition decision boards never had the uniformed Navy in any leadership position. In both the Army and Air Force, as evidenced from our interviews, the Vice Chiefs of each service at one time either chaired their acquisition decision boards, or, in more recent times, co-chaired those boards. The reason this is important is that the co-chairmanship gives the senior leadership an opportunity to demand and get information from the acquisition chain of command, starting with the program manager and going up to the PEO. That information flow is important to the decision process because it provides an understanding of what is happening in the program. This insight also allowed the uniformed Navy to see the consequences of its "requirements" process and the effect of changes made in various portions of the PPBE process. With this knowledge comes a shared responsibility for the end product, a most desired effect. Another consequence to knowledge is the loss of operational officers who understood the acquisition process. The result is that many of the flag officers who work in the office of the CNO have little to no experience with or understanding of the issues facing acquisition programs. Therefore, the requirements process sometimes imposes unreasonable demands and the PPBE process removes funding at critical times. In some of our interviews, some PEOs stated, "they are discussing my program in the Pentagon and I am not even invited."

Conclusions, Recommendations, and Issues that Warrant Further Study

Changes that affect the culture and processes of large bureaucratic organizations are always difficult to achieve. The notion of the need for and acceptance of the reality of a "burning platform" as enunciated by Gerstner at IBM is exactly for that reason. Military establishments because of their organization and the existential nature of their purpose are the most difficult to change, certainly to change quickly. In the case of the defense establishment in the '80s and '90s, it was change imposed by legislation that focused on "fixing" a myriad of both absurd and preserved problems without a clear understanding of the consequences that these fixes would being about. Legislated change in large enterprises has that effect, and it is noteworthy that the senior congressional protagonists were wary of unintended consequences but were driven to make changes because of the problems they had observed for well over a decade. In retrospect, it would appear that many of the interrelationships were not well understood. At each turn over the ensuing quarter century, many changes have been made in both statute and regulation to deal with "just one score" problem overlooked by previous attempts. It would be interesting to engage in time travel to see what the protagonists in the mid-'80s would have done if they were given a glimpse of the results. But that is not a particularly useful exercise because the problems were real and no one is contemplating reversing what has been done. Rather we must sift through the results of actions taken over time and see what may be practically done to address the concerns that were laid out by a previous CNO and form the core of our inquiry.

Just as the various statutes that were passed reflected the perception of members of Congress of the nature of the problems being experienced by the DoD, both operational, and in stewarding the public monies and trust, so perceptions of intent governed the promulgation of regulations to effect that legislation. The testimony we received strongly suggests that the intent was not clearly understood and there was a significant amount of interpretation, some of it self-serving, in promulgation of Instructions, Directives and Regulations. What appears to be a clear pattern is that many folks had reservations, not unlike Senator Nunn, but pressed forward anyway because the mandate for change at least was clear. What also became apparent is that the DoN, as a result of earlier resistance, was directed to proceed by higher authority an even more literal interpretation then necessary and did so. This "letter of the law" approach, taken even though there were reservations among the leadership, had the result of an implementation in the DoN different than those in the other military departments.

Clearly, Senator Nunn did not intend a rigid divide between civilian and military leaderships. Equally clearly, the Departments of the Army and Air Force have managed to avoid it to a certain degree under the same statutory and directive constraints that face the DoN. That leads us to conclude that the approach taken by the DoN is more malleable than believed. The authors would also conclude that the de facto exclusion of offices with an operational focus from the acquisition/material management process is not healthy. Finally, it is concluded that to achieve the results of the process improvements discussed in the recently issued *Quadrennial Defense Review (QDR) Report*, it is necessary that our best minds working together to solve problems, not sequentially engaging issues through choreographed organizational engagements.

Accordingly, a small number of specific recommendations are made and suggest several areas that would benefit from further study.

Recommendations

1. Make changes to applicable DoN Directives to undo the isolation conveyed by the Navy Gates Process and articulate a coherent and continuing role for the Service Chiefs across the range of the acquisition process, more like those of the other military departments.
2. Make changes to applicable DoN Directives to create and acquisition oversight body co-chaired by the ASN (RD&A) and the VCNO (and the ACMC for the discussion of Marine Corps systems of priority interest).
3. Create career opportunities for officers of the line in the material establishment.

Areas of Further Study

1. Best principles and approaches to expand and rebalance the acquisition workforce to enable informed collaboration in the requirements and resources processes.

    32. The granting of "joint-duty" credit for officers in large acquisition programs as suggested by the *QDR* for "recognizing joint experience whenever and wherever it occurs."

    33. Appropriate changes to DOPMA to create enhanced Senior Officer opportunities in acquisition.

References

Andrews, R., Cooper, J., Ellsworth, B., Sestak, J., Conway, K.M., Hunter, D., Coffman, M. (2010, March 23). *The House Armed Services Committee Panel on defense acquisition reform: Findings & recommendations.* Washington, DC: US GPO.

Anno, S.E., & Einspahr, W.E. (1988). *The Grenada Invasion command and control lessons learned: Iranian Rescue, Falklands Conflict, Grenada Invasion, Libya Raid* (Air War College Research Report). Maxwell Air Force Base, AL: Air University Press.

Assessment Panel of the Defense Acquisition Performance Assessment Project. (January 2006). *Defense acquisition performance assessment report*. Washington, DC: DoD. Retrieved April 1, 2010, from https://acc.dau.mil/CommunityBrowser.aspx?id=172904&lang=en-US

Conahan, F. (1989, June 1). *Acquisition reform: Military departments' response to the Reorganization Act* (Report No. GAO/NSIAD-89-70). Washington, DC: GAO. Retrieved March 31, 2010, from http://161.203.16.4/d25t7/138908.pdf

Defense organization: The need for change. (1985). *Armed Forces Journal International*, 3-22.

DoD. (2004). *Joint defense capability study of 2004*. Retrieved April 1, 2010, from https://acc.dau.mil/CommunityBrowser.aspx?id521805

Hone, T.C. (1987). *Power and change: The administrative history of the office of the chief of naval operations, 1946-1986* (Contributions to Naval History Ser., No. 2). Washington, DC: Naval Historical Center.

Hooper, E.B. (1978). *The Navy department: Evolution and fragmentation*. Washington, DC: The Naval Historical Foundation.

Nardulli, B., Perry, W.L., Pirnie, B.R., Gordon, J., IV, McGinn, J.G. (2002). *Disjointed war: Military operations in Kosovo* (Report MR-1406). Santa Monica, CA: RAND.

Retrieved March 10, 2010, from http://www.rand.org/pubs/monograph_reports/2007/MR1406.pdf

Nunn, S. (1985). *Statement of Senator Sam Nunn* (Conference Report). In Congressional Record 132, S 12651, Goldwater-Nichols Department of Defense Reorganization Act of 1986. Retrieved April 1, 2010, from Lexis-Nexis Congressional database.

US Congress. (1986). *Goldwater-Nichols Department of Defense Reorganization Act of 1986* (Public Law No. 99-433). Washington, DC: US GPO.

US Congress. (1987). *National Defense Authorization Act of 1987 (NDAA)*. Washington, DC: US GPO.

US Congress. (1990). Department of Defense implementation of the Packard Commission Report of 1986. Washington, DC: US GPO.

Risk Factors versus Dollar Value: Changing How Weapon System Programs are Managed

Robert Murphy—Robert Murphy retired as a member of the Senior Executive Service after a 33-year career in the US Navy's Nuclear Propulsion Program. He finished his career as the Program's Director of Resource Management, responsible for budget, acquisition and logistic support. While on active Government service, he participated in the acquisition of every nuclear submarine in the US fleet today, in addition to design contracts for SEAWOLF and VIRGINIA Class submarines. Since retiring from public service, he has been consulting for both commercial and government organizations, specializing in major system acquisitions. Such projects have included several cost studies of VIRGINIA Class submarines, in addition to advising on the most recent multi-year contract for acquisition of these submarines. RAND research projects have included several studies of the United Kingdom nuclear submarine industrial base. Murphy earned an MBA from George Washington University.

Abstract

The author discusses the current basis of the DoD's management and oversight of MDAPs (i.e., their dollar value) and proposes a new paradigm in which the level of management and oversight would be based on the level of risk an MDAP represents. The author also examines the extent to which the DoD is prepared to assess the following categories of risk: technical, system integration, design, production, and business. Finally, the author makes recommendations to improve the DoD's ability to assess these risks.

Preface

Today's defense environment is placing growing pressure on defense policymakers to be nimble and adaptive, particularly with respect to acquisition systems and processes. This occasional paper is one in a series drawing upon the expertise of core RAND Corporation staff to explore issues and offer suggestions on topics that are likely to be of critical importance to the new leadership: the use of competition, development of novel systems, prototyping, risk management, organizational and management issues, and the acquisition workforce. The papers are designed to inform new initiatives for markedly improving the cost, timeliness, and innovativeness of weapon systems that the Department of Defense (DoD) intends to acquire.

The Office of the Secretary of Defense (OSD) requires review of Major Defense Acquisition Programs and decisions by senior officials on the basis of a program's dollar value, irrespective of risk. In this paper, we propose a new paradigm in which the level of management and oversight would be based on the level of risk a program represents, including technical, system integration, design, production, and business innovation risk. We also examine the extent to which the DoD is prepared to assess these categories of risk and identify descriptive levels that could be used to assess and categorize design and business process risk.

This study was sponsored by the Office of the Under Secretary of Defense for Acquisition, Technology, and Logistics (OUSD(AT&L)) and conducted within the Acquisition and Technology Policy Center of the RAND National Defense Research Institute, a federally funded research and development center sponsored by the Office of the Secretary of

Defense, the Joint Staff, the Unified Combatant Commands, the Navy, the Marine Corps, the defense agencies, and the defense Intelligence Community.

For more information on RAND's Acquisition and Technology Policy Center, contact the Director, Philip Antón. He can be reached by e-mail at atpc-director@rand.org; by phone at 310-393-0411, extension 7798; or by mail at the RAND Corporation, 1776 Main Street, Santa Monica, CA 90407-2138. More information about RAND is available at www.rand.org.

Introduction

Currently, acquisition programs are grouped and then managed at the Office of the Secretary of Defense (OSD) by dollar value— depending on the dollar value, the OSD provides different levels of oversight and different management processes. This approach has been constantly refined over the years, without having produced any noticeable improvement in terms of reducing the cost growth, schedule slippage, and performance shortfalls that continue to plague the acquisition of weapon system programs. This paper argues for a different paradigm: The level of overall risk inherent in a program should be the main basis for determining the process and level of review a project should receive.[25]

Drawing upon examples from warship acquisition programs, this paper also argues that inadequate assessment and management of various discrete program risks results in adverse cost, schedule, and performance outcomes. We examine existing scales for assessing some of these discrete program risks and make recommendations to better assess and manage several programs within the Defense Acquisition Management System.

Managing by Risk Level versus Dollar Value

Currently, the OSD requires review of acquisition programs and decisions by senior officials on the basis of a program's dollar value, irrespective of risk, as shown in Table 1.

However, some very costly projects might have significantly less risk than projects of similar cost, and thus should require less oversight as well as the use of different criteria at milestone reviews.[26] Conversely, projects may cost little but have a lot of risk because they tend to push the state of the art in technology and may also involve novel business or design processes that may require more comprehensive oversight than just dollar value would otherwise indicate. An excellent example of this type of program—the Advanced SEAL Delivery System (ASDS)—was discussed in a May 2007 report by the US Government Accountability Office (GAO). The ASDS is a Special Operations Forces' battery-powered submersible, carried to a deployment area by a submarine. The operating parameters for the submersible required development of batteries that would push the state of the art in that technology. The initial design, construct, and deliver contract was awarded for $70 million in 1994 for delivery in 1997; because of the dollar value, Milestone Decision

---

[25] Cost is a factor that must be considered when determining the level of review. A multibillion dollar program requires high-level review because even a small amount of cost growth involves large dollar amounts.

[26] For example, the Navy is about to restart construction of two DDG 51-class destroyers at a cost in excess of several billion dollars. Over 60 destroyers of this class have already been delivered or are in the final stages of construction. Because of this track record, restarting construction of two new DDG 51s will no doubt expose the Navy to a far less risk of adverse cost, schedule, and performance outcomes than construction of three multibillion DDG 1000-class ships, which are now being designed and just entering construction.

Authority (MDA) resided with the Navy, which ultimately accepted delivery of ASDS in 2003 in "as is" condition at a cost in excess of $340 million. The GAO concluded that "Had the original business case for ASDS been properly assessed as an under-resourced, concurrent technology, design, and construction effort led by an inexperienced contractor, DoD might have adopted an alternative solution or strategy" (GAO, 2007, p. 13).

Table 1. Basis and Level of Program Oversight
(USD(AT&L), 2008)

Program Acquisition Category (ACAT)	Basis for ACAT Designation Milestone Decision Authority (MDA)	Milestone Decision Authority (MDA)
I	Estimated total expenditure for research, development, test, and evaluation (RDT&E) of more than $365 million or for procurement of more than $2.190 billion	ACAT ID: Under Secretary of Defense for Acquisitions, Technology, and Logistics ACAT IC: Head of DoD Component (e.g., Secretary of the Navy) or, if delegated, DoD component acquisition executive (e.g., Assistant Secretary of the Navy for Research, Development, and Acquisition)
II	Estimated total expenditure for RDT&E of more than $140 million or for procurement of more than $660 million	DoD component acquisition executive or designate (e.g., program executive officer)
III	Does not meet criteria for ACAT II or above; less than an MAIS program ACAT ID: Under Secretary of Defense for Acquisition, Technology, and Logistics	Designated by DoD component acquisition executive at the lowest level appropriate (e.g., program manager)

NOTE: Estimated expenditures are in FY 2000 constant dollars.

Focusing on Causes Rather than Consequences

Risk, or the exposure to the chance of failure, is a word heard frequently in the acquisition community. All acquisition programs face risk of some form or another. Arguably, any new major weapon system that could be developed, produced, and fielded with no risk involved is probably not worth acquiring.

Overtly or otherwise, much of a program manager's time is spent managing risk. After all, the Defense Acquisition Management System, shown in Figure 1, is, in essence, a risk-management process designed to ensure success in the timely delivery of weapon systems that meet warfighter requirements while staying within budget.

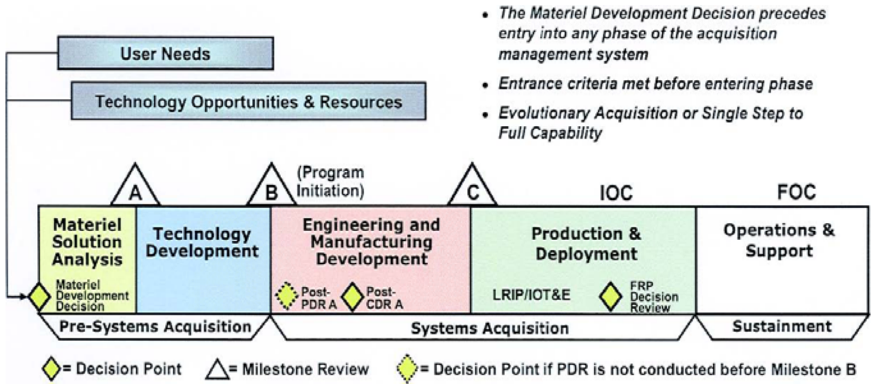


Figure 1. The Defense Acquisition Management System
(USD(AT&L), 2008)

The risks most frequently mentioned by defense acquisition officials are cost growth, schedule slippage, and performance shortfalls. This is not surprising as cost, schedule, and performance are the primary attributes by which programs are assessed for success or failure. Moreover, the Defense Acquisition University (n.d., p. 2) teaches that cost, or performance schedule, and performance are the risk factors that program managers must assess and manage "to ensure that DoD is acquiring optimum systems that meet all capability needs."

Assessment of cost, schedule, and performance is clearly a management task, and a good program manager assesses these risks using periodic data accumulated into management reports to identify problems, regain lost ground, and then stay on track. However, these are broad measures of risk. A better program manager proactively manages by using discrete program risks and submeasures that allow him or her to look ahead and act to avoid adverse cost, schedule, and/or performance trends and outcomes. In other words, managing by cost, schedule, and performance measures is akin to driving a car while looking solely in the rearview mirror—it is possible, but only if the road stays straight. A better driver looks mostly out the windshield, with only an occasional look in the mirror; this driver anticipates and easily handles curves in the road.

In this paper, we focus on five discrete programmatic risk categories:

- technical,
- system integration,
- design,
- production, and
- business.

Taken together, these risk categories portray overall acquisition program risk.[27] They interact in numerous ways to affect a project's cost, schedule, and/or performance outcomes: Obviously, technologies that do not work affect performance, but so can poor business decisions that increase cost and lead to features being deleted from the weapon system to remain within budget.

[27] For simplicity, risks involved in fielding, operating, and maintaining the weapon system are not addressed in this paper.

The Defense Acquisition Management System appears to adequately recognize that incorporation of new technologies into a weapon system presents risk, providing metrics to systematically assess this type of risk. Time is also provided in the acquisition process for system integration matters to be identified and resolved, although there is room for improvement. However, as will be discussed in subsequent examples, new approaches in design, production, and business areas of acquisition programs do not appear to receive the same skepticism and comprehensive oversight that new technologies and systems receive.

Well-Defined Process for Assessing Technical Risk Is in Place

"Technical risk" is exposure to the chance that development of critical technologies will not meet program objectives within cost and schedule constraints. In assessing technical risk, program managers must address the uncertainty in their estimates about how much time and effort it will take to make new technologies work. The importance of technical risk is well understood in the acquisition community. For example, DoD guidance states that "the management and mitigation of technology risk…is a crucial part of overall program management and is especially relevant to meeting cost and schedule goals" (USD(AT&L), 2008, para. 3.7.2.2).

Technical risk is also extensively addressed in the Defense Acquisition Management System. The system recognizes evolutionary acquisition as the preferred DoD strategy for rapid acquisition of mature technology for the user. One purpose of evolutionary acquisition (i.e., delivering capability in increments through spiral or incremental development) is to provide time to better manage technology risk and avoid adverse cost and schedule outcomes that often result from trying to achieve difficult requirements in one step.

The DoD has also established a well-defined process based on Technology Readiness Levels (TRLs) to categorize technical risk and help ensure that key decision-makers understand the risk of incorporating different technologies into weapon system acquisition programs (the TRLs are described in Table 2). Using this process, program offices conduct a technology readiness assessment under the auspices of the DoD Component Science and Technology (S&T) Executive; the Deputy Under Secretary of Defense (S&T) evaluates the technology readiness assessment and forwards findings to the Overarching Integrated Product Team leader and Defense Acquisition Board.

The TRLs are a good proxy measurement for technical risk: The lower the readiness level, the more development needed to incorporate the technology into a weapon system; and the more development needed, the greater the risk. Overall, technology risk has been handled fairly well in warship acquisition programs, which tend not to push the state of the art in technology as far as do weapons and sensors. A recent example is the USS Virginia, which incorporates various new technologies[28] and was still delivered within four months of the original schedule established a decade earlier (Casey, 2007).

---

[28] For example, a nonpenetrating photonics mast versus a periscope, a DC electric system, Lightweight Wide Aperture Sonar Arrays, etc.

Table 2. Technology Readiness Levels
(DUSD(S&T), 2005)

Technology Readiness Levels
1. Basic principles observed and reported
1. Technology concept and/or application formulated
2. Analytical and experimental critical function and/or characteristic proof of concept
3. Component and/or breadboard validation in laboratory environment
4. Component and/or breadboard validation in relevant environment
5. System/subsystem model or prototype demonstration in a relevant environment
6. System prototype demonstration in an operational environment
7. Actual system completed and qualified through test and demonstration
8. Actual system proven through successful mission operations

NOTE: See Mankins (1995).

System Integration Risk Is Assessed, But at Later Stages

The acquisition community also assesses system integration risk, but it lacks effective tools to measure and categorize this risk early in a program's life cycle. "System integration risk" is exposure to the chance that new and existing technologies being employed in a weapon system may not work together and/or interact with operators and maintainers to meet program objectives within cost and schedule constraints. System integration can be an issue within an individual acquisition program (e.g., when subsystems fail to interact). It can also be an issue between acquisition programs: Many programs develop capabilities that are a component of a larger warfighting capability; individually, the component programs might appear to be a low or moderate risk, but in combination with other programs, the overall risk might be much higher due to coordination and integration issues. A classic example occurred during the Grenada invasion when Army and Navy communications systems did not interact well during the joint operation.

System integration risk is extensively treated after Milestone B, during the engineering and manufacturing development (EMD) phase, at which time a program should demonstrate system integration, interoperability, safety, and utility (USD(AT&L), 2008, para. 3.7.1.1). While appropriate attention is given to system integration risk during this phase, this assessment occurs after the second of three milestones in the process, when programs have typically built up so much momentum that they are difficult to stop, regardless of performance or progress. Early consideration of system integration risk—at Milestone A—by senior decision-makers could result in developing and funding integration-risk mitigation plans that could considerably improve acquisition outcomes.

Combat systems in warships provide an example of the problems that arise when decisions are made without adequate consideration of system integration risk.[29] For example, early decisions on systems architecture and processing approaches made without adequate consideration of risk led to cost, schedule, and performance problems with submarine combat systems for the SSN 688I, SEAWOLF, and Australian Collins-class submarines. According to a report for the Parliament of Australia discussing the Collins-class submarine,

> Of the early decisions in the Collins program, the one which was to have the most public effect was that concerning the nature of the vessels' Combat Data System (CDS). It has been the subsequent failure of this system to meet its design requirements that has left the submarines with a severely impaired combat capability.
>
> By the end of 1982…[the Royal Australian Navy (RAN)] had decided that the electronic combat systems of the new boats would be fully integrated. Instead of the then standard central computer performing all data analysis, the new submarine CDS would use a data bus to distribute information to a number of smaller computer work stations. (Woolner, 2001)

The report then goes on to discuss the lack of appreciation for the risk of switching to the new integrated architecture for combat systems.

> The RAN was not alone in its "grand folly."… The Australian information technology (IT) industry assured the RAN of both the feasibility and inherent advantages of a fully integrated combat system and of its ability to contribute to such a program.
>
> Moreover, the RAN was not the only navy to think that the future of combat data processing lay with fully integrated systems. The USN [U.S. Navy] specified the same concept for its [BSY-2] Integrated Combat System for the U.S. Navy's Seawolf class nuclear attack submarines. This was an even more costly failure than the Collins CDS, absorbing…$1.5 billion [in U.S. dollars] before it was cancelled.[30]

Tools for assessing system-integration maturity earlier on have been proposed. For example, Sauser, Ramirez-Marquez, Magnaye, and Tan (2008) have proposed a System Readiness Level (SRL) index that would incorporate the current TRL scale as well as an Integration Readiness Level (IRL) scale. The IRL scale they describe would use nine levels, which appear compatible with the widely used TRLs and appear to be a good proxy measurement of system integration risk. The proposed IRLs are listed in Table 3.

The Risks of Design Process Management Are Not Well Understood

"Design risk" is exposure to the chance that the weapon system's design will not result in effective operation or be easy to produce. It is axiomatic that a good design is essential to a weapon system's performance, but the impact of design on a weapon system's production cost and schedule outcome is not as well appreciated. However, decisions made early in the design process quickly establish not only the performance but also the ease of manufacture and resultant cost of the weapon system. While the ability of the

[29] A combat system integrates information from sensors, synthesizes this information for combat commanders, and provides fire control solutions and guidance to weapons.

[30] The original citation mistakenly attributed this to the BSY-1 program.

design to operate effectively can be considered a subset of technical risk, a more holistic approach is for a program manager to assess the chance that the design process to be employed for the weapon system will generate an effective, easy-to-produce weapon.

Table 3. Integration Readiness Levels
(Sauser et al., 2008)

Integration Readiness Levels
1. An interface between technologies is identified with sufficient detail to allow characterization of the relationship.
1. There is some level of specificity to characterize the interaction (i.e., ability to influence) between technologies through their interface.
2. There is compatibility (i.e., a common language) between technologies to orderly and efficiently integrate and interact.
3. There is sufficient detail in the quality and assurance of the integration between technologies.
4. There is sufficient control between technologies necessary to establish, manage, and terminate the integration.
5. The integrating technologies can accept, translate, and structure information for their intended application.
6. The integration of technologies is verified and validated with sufficient detail to be actionable.
7. Actual integration is completed and mission qualified through test and demonstration in the system environment.
8. Integration is mission proven through successful mission operations.

The design process necessary for an effective and producible weapon system involves complex interactions between designers, suppliers, production experts, planners, and estimators. Design process complexity has also increased with the availability of more sophisticated design tools such as electronic product models and computational techniques (e.g., finite element analysis).

Outcomes from two current acquisition programs—the United Kingdom's ASTUTE-class submarine and the US Navy's LPD 17-class of amphibious transport dock ships—demonstrate why senior decision-makers in the OSD acquisition process need to better understand the risks new design processes and tools present. The ASTUTE was the first UK submarine to be designed through use of an electronic, three-dimensional computer product model. The prime contractor's inability to manage this new process resulted in extensive delays when design products needed to build the ship were late. General Dynamics ultimately had to be hired to augment and manage the final stages of the submarine's detail design process. Because of design and other problems, the ASTUTE program has overrun cost greatly and is years behind schedule.

With LPD 17, the US Navy competed the design and production of the first three ships of the class using as major evaluation and award criteria (1) the plans for accomplishing detail design and other functions, (2) Integrated Product Data Environment (IPDE) tools, and (3) life-cycle cost projections; these criteria were given more weight than price (Comptroller General of the United States, 1997). The then-Avondale Shipyard in New Orleans, Louisiana, partnered with a firm that was developing a new ship design IPDE tool and won the competition. Subsequently, the LPD 17 experienced considerable cost growth (about 70%) and schedule delays (CRS, 2008, p. 12). The GAO attributed much of this cost growth to the new design tool:

> In the LPD 17 program, the Navy's reliance on an immature design tool led to problems that affected all aspects of the lead ship's design. Without a stable design, work was often delayed from early in the building cycle to later, during integration of the hull. Shipbuilders stated that doing the work at this stage could cost up to five times the original cost. The lead ship in the LPD class was delivered to the warfighter incomplete and with numerous mechanical failures. (GAO, 2007)

Senior decision-makers should require a program manager proposing to use new design processes, tools, or organizations to design a weapon system to justify selection of the new process, tool, or organization and develop an appropriate risk mitigation plan. An example of a design process mitigation plan comes from the VIRGINIA-class submarine program. Prior to VIRGINIA-class construction using a new Integrated Product and Process Development (IPPD) approach, Electric Boat stated,

> a representative section of the ship about a year early with a portion of that section started about two years early. This early, controlled, closely monitored ship construction effort ensured thorough preparation for full-ship application and high confidence in the new process. (General Dynamics Electric Boat, 2002, p. 33)

Evaluation of Production Risks Lacks Rigor

An earlier and more rigorous evaluation of production risks could save the DoD much difficulty and taxpayers a lot of money. "Production risk" is exposure to the chance that the facility, labor, manufacturing processes, and procedures will fail to produce the weapon system within time and cost constraints. Producibility—or "production capability"—is a function of the design; production facilities; management skills, processes, and experience; and workforce skills and experience. The DoD requires assessment of contractors' production capability before production contract award in the production and deployment phase, but this may be too late because, at this point, production may be locked in by the organization that won the design contract. Moreover, in the authors' experience, and as exemplified in the LPD 17 source-selection criteria discussed earlier, the production category of risk does not receive the same emphasis in selecting a shipbuilder as other factors, such as design concepts, past performance, and estimated cost.

The Navy's DD 963-class of destroyers and LHA 1-class of amphibious assault ships are classic examples of programs in which the DoD considered design and production risk acceptable when awarding contracts, but which went on to experience about the worst of every production factor possible. These ships presented little technical and system integration risk, but ended up far behind schedule and over cost, due in part to identifiable production risks. Contracts were awarded to the lowest bidder, Litton Industries, which owned the Ingalls shipyard in Pascagoula, Mississippi. In the late 1960s, Litton Industries decided to invest in an expansion of design and production facilities for warships, building a new shipyard on the west bank of the Pascagoula River, across from its existing shipyard. The new shipyard was designed to be operated with a new production control system using modular techniques for building ships (Northrup Grumman, 2008).

After the award of the LHA- and DD 963-class contracts to Ingalls for nine LHAs and 30 DD 963s in the late 1960s, Ingalls' management decided to shift construction of some commercial container ships from the old, conventional yard to the new facility (Northrup Grumman, 2008). The expectation was that doing so would allow the new facility to start up and have any problems worked out while the LHA and DD 963 were being designed. However, production of the container ships using the new control system led to delays; consequently, the ships were occupying facilities and using manpower needed to start production of the LHAs and DD 963s. Production of the LHAs and DD 963s fell far behind and, in combination with other problems (design-related issues, inflation, etc.), the costs were overrun substantially and the ships were late (GlobalSecurity.org, 2008).

A greater emphasis on evaluating production risks could have saved an enormous amount of time and money, but the promised cost savings resulting from construction in a new, state-of-the-art ship fabrication and assembly facility proved too good to be true. The assessment that the facility would be derisked by building container ships first turned out to be wrong, and, meanwhile, two entire classes of ships had been priced and placed under contract.

A promising approach, initiated by the Missile Defense Agency, may provide program offices across the DoD with better insight about production risk. The agency extended the notion of TRLs to engineering and manufacturing by developing Engineering and Manufacturing Readiness Levels (EMRLs) to assess the maturity of a program's design, related materials, tooling, test equipment, manufacturing, quality, and reliability levels. There are five EMRLs, as shown in Table 4.

Table 4. Engineering and Manufacturing Readiness Levels
(DUSD(S&T), 2005)

Engineering and Manufacturing Readiness Levels
1. System, component, or item validation in laboratory environment or initial relevant engineering application or breadboard, brass-board development.
1. System or components in prototype demonstration beyond breadboard, brass-board development.
2. System, component, or item in advanced development. Ready for low-rate initial production.
3. Similar system, component, or item previously produced or in production. System, component, or item in low-rate initial production. Ready for full-rate production.
4. Identical system, component, or item previously produced or in production. System, component, or item in full-rate production.

The Risk of Early Business Decisions Is Not Fully Appreciated

Business decisions made early in a program's life can significantly affect cost, schedule, and performance outcomes. "Business risk" is exposure to the chance that the overall acquisition strategy for a program will not result in the desired cost, schedule, and/or performance outcomes. Decisions about the process to select who will build the weapon system, the standards to which it will be built, and the schedules for designing and building it all entail risk that is not always appreciated up front. To evaluate business risk, program managers should assess the following: (1) the extent to which the acquisition strategy can result in selection of the most effective, efficient design and most effective, efficient production entities; (2) whether cost estimates and schedules are valid; (3) whether proper government oversight organizations are in place; and (4) whether project personnel with proper training and experience are available.

A good example of early business decisions gone bad is the Navy's Littoral Combat Ship (LCS) Program. The lead ship, USS Freedom (LCS 1), was recently delivered after experiencing substantial cost overruns and delivery delays. In congressional testimony given to explain these outcomes, the US Navy (2007) identified the following tenets of the new business model used to acquire the LCS:

- Construction of LCS seaframes in midtier shipyards that "perform predominately commercial work, maintaining business processes and overhead structures that keep them competitive in the world market" (i.e., little warship experience).[31]
- "A rapid 24-month build cycle for each seaframe, as opposed to the five or more years that have become the norm in naval shipbuilding."
- "The LM lead ship detail design and construction effort was initiated simultaneously and the lead ship commenced construction only seven months after the start of final design (i.e., concurrent design/build)."
- "In order to address the challenges of technical authority under this environment (reduction in NAVSEA technical personnel), in February 2003, NAVSEA and PEO Ships made two joint decisions. The first was to work with the American Bureau of Shipping (ABS) to develop a set of standards (Naval Vessel Rules) that could be applied to non-nuclear naval combatant ships. The second was to utilize ABS to class[32] both LCS and DDG 1000 using the new rules."

No doubt there were good arguments for these individual program tenets. However, the cumulative effect of the risks involved in building a new design warship in small commercial shipyards with little warship experience during a rapid, concurrent design/build process and to a set of technical standards themselves under development appears to have been greatly underappreciated. In that same congressional testimony, the Navy identified cost drivers for LCS 1 as "concurrent design-and-build while incorporating Naval Vessel Rules (NVR), reduction gear delays created by a manufacturing error, and insufficient program oversight" (US Navy, 2007). The risks inherent in utilizing an entirely new business model to acquire warships were obviously neither adequately assessed nor managed.

One way to avoid such risk would be to require program managers proposing new and/or radical business models to fully justify why the new model is superior to past practice, recommend more frequent assessment points than now required by the Defense Acquisition Management System, and incorporate exit strategies in contracts for the government to use if the program fails to meet expectations.

---

[31] To better understand the differences between military and commercial shipbuilding, see Birkler et al. (2005).

[32] The American Bureau of Ships is known in the commercial shipping industry as a "classification society," which is an organization that sets standards for design and construction of vessels and integral machinery within.

Conclusions

The Defense Acquisition System Framework has sufficient tools and allows time for proper assessment and management of technical risk and, to some extent, of system integration risk. However, design, production, and business risks are not always adequately assessed and managed. As shown in this discussion, scales exist that represent good proxy measurements of technical, systems integration, engineering, and production risks; what is missing are descriptive levels that could be used to assess and categorize design and business process risk. We recommend that the DoD explore establishing such levels and, in Tables 5 and 6, offer starting points for doing so (based on the authors' experience), which may help program managers more carefully consider these risks.

In addition, we recommend the following actions to better assess and manage program risk overall:

- Assess, categorize, and individually review each technical, system, design, production, and business risk of a program at each milestone in the Defense Acquisition Management Framework.
- Require program managers to justify new or radical approaches to design, production, or business processes and develop and implement risk mitigation plans and/or contract off -ramps.

Table 5. Proposed Design Process Levels

Design Processes
1. New, unproven processes. New design tools under development. New design organization.
2. Large expansion of existing design organization. Many new designers and supervisors unfamiliar with design tools and processes.
3. Existing design organization using radically changed design tools, processes, and/or technologies.
4. Experienced design organization using new design tools with proven processes.
5. Experienced design organization using existing, proven design tools and processes.

Table 6. Proposed Business Process Levels

Business Processes
1. Using a new, unproven approach to source selection. Encouraging new sources of supply. Acquiring new technologies without well-established cost-estimating relationships. Requiring new government and/or contractor organizations to be formed.
1. Using new procurement process in established industry. Cost-estimating relationships only at high levels. Requires expansion of government and contractor organizations.
2. Evolutionary change from prior acquisition strategies. Good cost-estimating relationships. Existing government and contractor organizations can easily adapt to changes.
3. Using same approach to buying similar products. Well-established cost-estimating relationships exist. Experienced government and contractor organizations involved.
4. Acquiring more of what has been successfully bought before. Using the same contractor and government organizations.

Although such tools would enhance the ability of program offices to assess and manage risk, the DoD should also consider changes in oversight. As stated at the outset of this paper, the current acquisition system requires review and decisions by senior officials on the basis of a program's dollar value, irrespective of risk. A better use of their limited time may be to focus on programs with high risks, letting less-senior officials deal with lower-risk programs, regardless of dollar value. For example, the DoD could

- lower the MDA level for future milestones down
  - —two levels for programs with low risk in all risk categories33
  - —one level for programs with moderate risk in all risk categories.34
- continue to follow the patterns for decision authority as established in the Defense Acquisition Management System for any program with greater than moderate risk in any of the five categories of program risk.

In this way, senior decision-makers might be able to better concentrate their limited time on the real potential problem areas in a program before problems occur, and direct actions to be taken to avoid and/or mitigate potential problems.

References

[33] Determination of what constitutes "low risk" is obviously subjective. For our purposes, "low risk" would be technology and integration levels 8 and 9 and EMRL, design, and business levels 4 and 5.

[34] For our purposes, "moderate risk" would be TRL and IRL 5 and 6 and EMRL design and business levels 3.

Birkler, et al. (2005). *Differences between military and commercial shipbuilding: Implications for the United Kingdom's Ministry of Defence* (MG-236-MOD). Santa Monica, CA: RAND Corporation. Retrieved from http://www.rand.org/pubs/monographs/MG236/

Casey, J.P. (2007, March 8). Electric Boat Corporation's testimony before the Seapower and Expeditionary Forces Subcommittee of the House Armed Services Committee. Washington, DC.

Comptroller General of the United States. (1997, April 7). *Decision B-275830*.

Congressional Research Service (CRS). (2008, May 23). *Navy LPD-17 amphibious ship procurement: Background, issues, and options for Congress*.

DAU. (n.d.). *Systems acquisition management: An introduction* (Lesson 2). Retrieved September 15, 2009, from https://learn.dau.mil/CourseWare/1_13/media/Lesson_2_Summaryf.pdf

DUSD (S&T). (2005, May). *Technology readiness assessment (TRA) desk-book*. Washington, DC: DoD. Retrieved May 21, 2009, from http://www.dod.mil/ddre/doc/May2005_TRA_2005_DoD.pdf

Francis, P.L. (2007, July 24). *Defense acquisitions: Realistic business cases needed to execute Navy shipbuilding programs* (GAO07-943T). Testimony before the Subcommittee on Seapower and Expeditionary Forces, Committee on Armed Services, House of Representatives. Washington, DC: US GPO.

GAO. (2007, May). *Success of advanced SEAL delivery system (ASDS) hinges on establishing a sound contracting strategy and performance criteria* (GAO-07-745). Washington, DC: Author.

General Dynamics Electric Boat. (2002, February). *The Virginia class submarine program: A case study*. Groton, CT.

GlobalSecurity.org. (2008). *LHA-1 Tarawa class program*. Retrieved November 2008, from http://www.globalsecurity.org/military/systems/ship/lha-1-program.htm

Manager's guide to technology transition in an evolutionary acquisition environment (Vers. 2.0). (2005, June). Fort Belvoir, VA: DAU Press.

Mankins, J.C. (1995, April 6). *Technology readiness levels: A white paper*. NASA, Office of Space Access and Technology, Advanced Concepts Office. Retrieved May 21, 2009, from http://www.hq.nasa.gov/office/codeq/trl/trl.pdf

Northrup Grumman Corporation. (2008). *Ingalls operations: A chronological perspective*. Retrieved from November 8, 2008, from http://www.ss.northropgrumman.com/company/chronological.html

Sauser, B.J., Ramirez-Marquez, J.E., Magnaye, R.B., & Tan, W. (2008, May). System maturity indices for decision support in the defense acquisition process. In *Proceedings of the Fifth Annual Acquisition Research Symposium*. Monterey, CA:

USD(AT&L). (2008, December 8). *Operation of the defense acquisition system* (DoD Instruction 5000.2). Washington, DC: Author.

US Navy. (2007, February 8). Statement of the Honorable Dr. Delores M. Etter, Assistant Secretary of the Navy (Research, Development, and Acquisition); VADM Paul E. Sullivan, US Navy, Commander, Naval Sea Systems Command; RADM Charles S. Hamilton, II, US Navy, Program Executive Officer, Ships; and RADM Barry J. McCullough, US Navy Director of Surface Warfare, before the Subcommittee on Seapower and Expeditionary Forces of the House Armed Services Committee on Acquisition Oversight of the US Navy's Littoral Combat Ship Program. Washington, DC: US GPO.

Woolner, D. (2001). *Getting in early: Lessons of the Collins Submarine Program for Improved Oversight of Defence Procurement*. Parliament of Australia, Parliamentary Library. Retrieved September 15, 2009, from http://www.aph.gov.au/library/pubs/rp/2001-02/02RP03.htm

Wednesday, May 12, 2010	
1:45 p.m. – 3:15 p.m.	**Chair: Keith E. Seaman**, Acting Director, Defense Business Systems Acquisition Executive, Business Transformation Agency **Discussant:** Jeff Olinger, Technical Advisor to the Commander Air Force Operational Test and Evaluation Center ***Test and Evaluation at the Speed of Need*** **Steven J. Hutchison**, Defense Information Systems Agency ***Integrated Testing and Independent Evaluation Model*** **Darlene Moser-Kerner,** Developmental Test and Evaluation

Test and Evaluation at the Speed of Need

Steven J. Hutchison—Dr. Steven J. Hutchison assumed duties as Test and Evaluation Executive, Defense Information Systems Agency in August 2005. Dr. Hutchison supervises the Joint Interoperability Test Command, T&E Management Center, and the IT testbed in the Major Range and Test Facility Base.

Dr. Hutchison previously served with the Director, Operational Test and Evaluation and Army Test and Evaluation Command. Dr. Hutchison retired from the US Army in 2002.

Dr. Hutchison earned a PhD in Industrial Engineering at Purdue University, an MS in Operations Research at the Naval Postgraduate School, and is a 1982 graduate of the United States Military Academy.

Abstract

During this past year, Congress passed the *Weapons Systems Acquisition Reform Act*, which made several changes to DoD acquisition organizations and processes. More recently, Congress passed, and the President signed, the *National Defense Authorization Act for FY2010*, becoming Public Law 111-84, directing changes in DoD acquisition of *information technologies* (IT). The law requires the DoD to base the new acquisition process on recommendations in the March 2009 *Report of the Defense Science Board Task Force on Department of Defense Policies and Procedures for the Acquisition of Information Technology* (hereafter *DSB-IT*). The report recommends an agile model for acquiring information technologies (IT) similar to successful commercial practices (see http://www.acq.osd.mil/dsb/reports.htm). A second *DSB* report, also issued in March 2009, the *Report of the Defense Science Board Task Force on Achieving Interoperability in a Net Centric Environment* (*DSB-NC*), made recommendations to ensure that IT acquisition delivers information-assured, interoperable capabilities essential to modern warfighting. Together, the reports provide a foundation on which to build the new model for acquisition and testing of IT; this paper attempts to connect them and fill the remaining gaps necessary to truly transform to agile processes that foster rapid acquisition of enhanced IT capabilities for the warfighter.

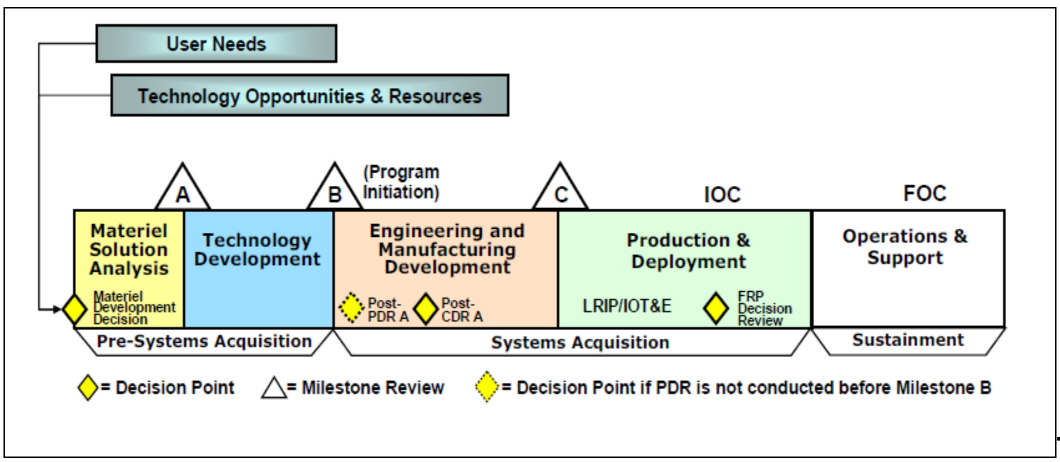
Test and Evaluation at the Speed of Need

Department of Defense acquisition is always under the watchful eye of the Congress. During this past year, Congress passed the *Weapons Systems Acquisition Reform Act*, which made several changes to DoD acquisition organizations and processes. More recently, Congress passed, and the President signed, the *National Defense Authorization Act for FY2010*, becoming Public Law 111-84, directing long overdue changes in DoD acquisition of *information technologies* (IT). According to section 804, “The Secretary of Defense shall develop and implement a new acquisition process for information technology systems.” The law requires the DoD to base the new acquisition process on recommendations in the March 2009 *Report of the Defense Science Board Task Force on Department of Defense Policies and Procedures for the Acquisition of Information Technology* (hereafter *DSB-IT*). The report recommends an agile model for acquiring information technologies (IT) similar to successful commercial practices (see http://www.acq.osd.mil/dsb/reports.htm). Interestingly, a second *DSB* report, also issued in March 2009, the *Report of the Defense Science Board Task Force on Achieving*

Interoperability in a Net Centric Environment (DSB-NC), made recommendations to ensure that IT acquisition delivers information-assured, interoperable capabilities essential to modern warfighting. Together, the two reports should be used as the foundation on which to build the new model for acquisition and testing of IT; this paper attempts to connect them and fill the remaining gaps necessary to truly transform to agile processes that foster rapid acquisition of enhanced IT capabilities for the warfighter.

Acquisition and Testing of Information Technologies in the DoD

The DoD acquires IT using the same acquisition model as tanks and ships and planes. Figure 1 is the familiar Defense Acquisition Management System taken from *DoD Instruction 5000.02*. This system essentially makes no distinction between major defense acquisition programs (MDAP) and major automated information systems (MAIS); program managers for IT capabilities manage the same set of milestones and decision points and are subject to the same governance processes and oversight. Make no mistake; this system has produced the best military equipment in the world, but in recognizing this fact, it is important to realize that the process works well when there is a looooooooong time between user need definition (far left of chart) and declaration of Initial Operational Capability (IOC) (subsequent to the final decision point on the chart). Therein lies the problem for IT: the fundamental reason this model does not work well for IT capabilities is that we typically want a very short time between user need definition and IOC.



T

Figure 1. The Defense Acquisition Management System

The *DSB-IT* describes the current DoD IT acquisition process as a “big bang approach,” meaning we try to get everything in the first increment. The report describes the approach as one that “begins with an analysis phase followed by an equally long development phase that culminates in a single test and evaluation event.” The *DSB-IT* cited an analysis conducted by the Assistant Secretary of Defense for Networks and Information Integration (ASD(NII)) of 32 major automated information systems, which showed that the average time to deliver an initial capability is 91 months! Figure 2, taken from the *DSB-IT* report, summarizes the length of time spent in each phase of the acquisition system, according to the ASD(NII) analysis. The *DSB-IT* concludes, “The conventional DoD acquisition process is too long and cumbersome to fit the needs of the many systems that require continuous changes and upgrades.”

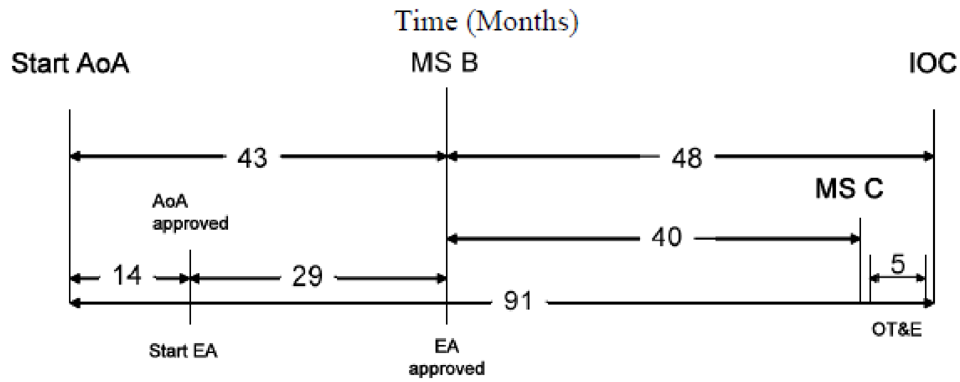


Figure 2. DoD IT Acquisition Timeline
(OUSD(AT&L), 2009a)

The *DSB-IT* reached the conclusion that current acquisition policies and processes (as defined in the DoD 5000 series directive and instruction) “do not address the fundamental challenges of acquiring information technology for its range of uses in DoD. Instead, a new acquisition approach is needed that is consistent with rapid IT development cycles and software-dominated acquisitions.” The *DSB-IT* proposed a new model for acquisition of IT, depicted in Figure 3. The proposed model is agile, based on successful commercial practices, and intended to deliver capability in “release” cycles of approximately 18 months or less. Releases are divided into “iterations” (nominally three iterations per release). Lastly, the model highlights integrated developmental test (DT) and operational test (OT).

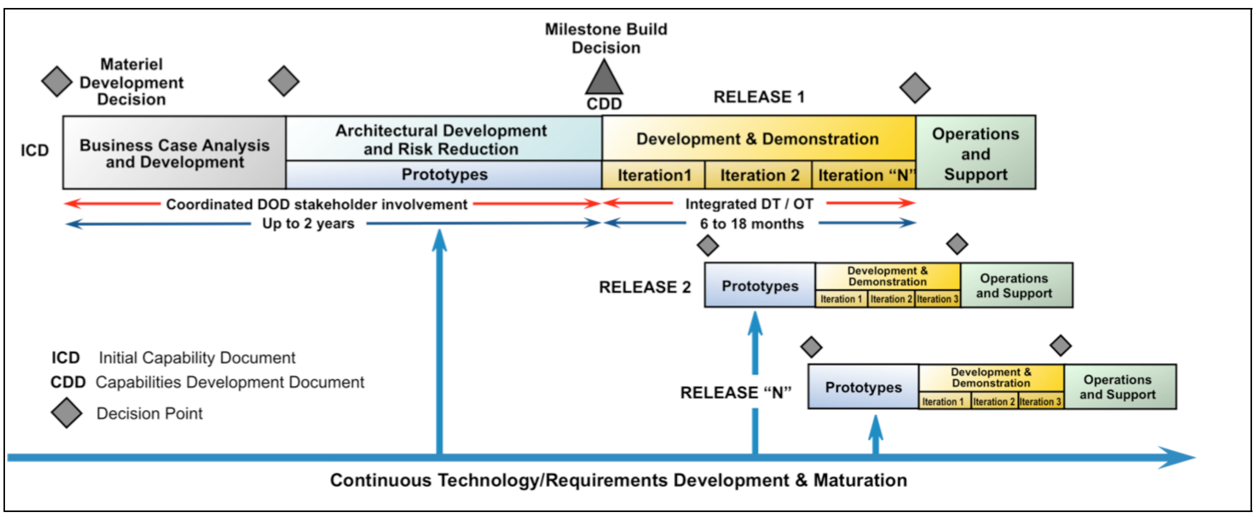


Figure 3. Proposed IT Acquisition Process
(OUSD(AT&L), 2009a)

Test and evaluation (T&E) is an essential part of the DoD acquisition system. Test and evaluation typically begins with early prototypes, and then becomes increasingly complex, as testing progresses from individual components to systems, then the “system of systems.” Likewise, test conditions generally evolve from benign, low stress, lab environments through early operational assessments with a limited user base, to full scale, formal OT&E on production representative systems with trained users. Figure 4 depicts the flow of test events, all of which are found on the right side of the “systems engineering V” diagram, as shown in the Integrated Defense Acquisition, Technology and Logistics Life Cycle Management System chart (DAU, 2009). Despite the increased emphasis on “integrated testing” today, test, evaluation, and certification (TE&C) activities still concentrate at the end of development. Moreover, the DoD version of the V, as depicted in the figure, does not connect the early test activities to the IOT&E or interoperability testing. In an acquisition model designed for IT, we have to transform the traditional “one way” V into an iterative process; likewise, testing should be early and often (parallel versus integrated) and always with a mission focus.

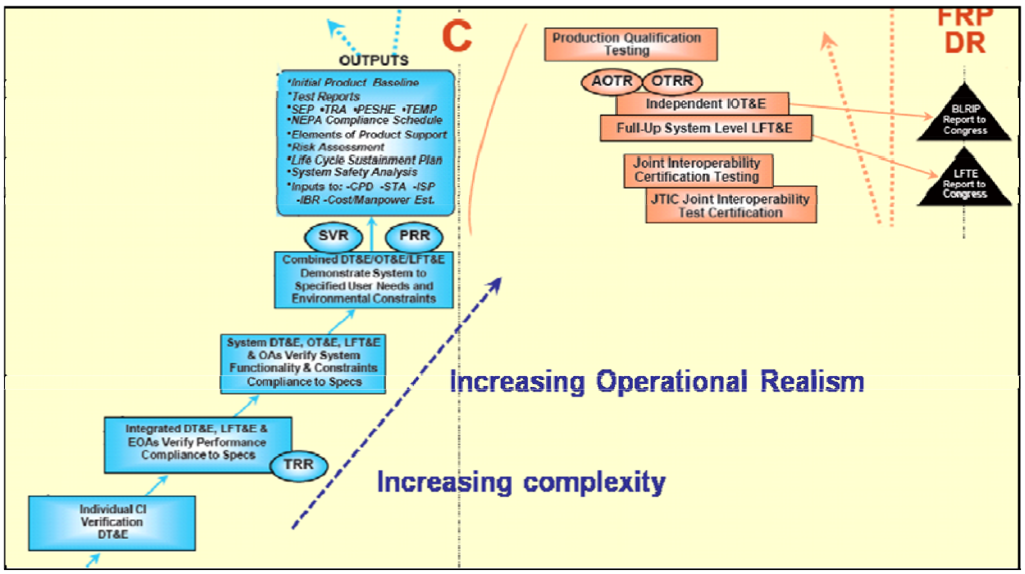


Figure 4. T&E in the Systems Engineering "V"

One of the concerns with the process depicted in Figure 4 is that programs engage different test organizations at different times, or change them mid-stream. This is particularly evident in the transition from the developmental tester to the independent operational test agent and may explain the disconnect noted above. For IT capabilities, the interoperability tester and the security (information assurance) tester conduct assessments and report results for separate decision-making (certification) purposes. This separation of test organizations and activities may have the effect of parsing information to different decision-makers as opposed to fusing results into a comprehensive evaluation. As we develop a new IT acquisition model, we should consider a TE&C model that synchronizes the efforts of all test organizations towards improving capability and providing comprehensive information to decision-makers.

Test and evaluation has its own “big bang” in the DoD acquisition system. The Initial Operational Test and Evaluation (IOT&E), shown in Figure 1, is the culminating event in a T&E strategy and is necessary to achieve a fielding decision. Title 10 USC, §139, mandates IOT&E for MDAPs for “the purpose of determining the effectiveness and suitability of the weapons, equipment, or munitions for use in combat by typical military users"; the *DoD 5000* applies this requirement to MAIS. The IOT&E is a complex endeavor; it takes a long time to plan, requires a test unit (sometimes hard to come by in a Department at war), time to train the test unit and the testers, a support system, extensive data collection and analysis, and time to prepare reports for decision-makers. The National Research Council observed, the "DoD is fast approaching a period in which a single all-encompassing large-scale operational test, as currently practiced, will cease to be feasible" (NRC, 2006). For warfighting platforms that have long developmental timelines, an IOT&E is likely to be a small proportion of the total program cost and short relative to the total program schedule. This is another factor to consider in development of an IT acquisition model. For IT capabilities following agile development, the current approach to IOT&E could have significant cost and schedule impact. The question is, therefore, how to reduce the impact without loss in rigor and objectivity.

Test, Evaluation, and Certification of DoD IT

Test, evaluation, and certification for IT has several facets. Figure 5 portrays a high-level view of the IOT&E "test execution window" for IT capabilities. Depicted in the figure are the various TE&C and supporting activities to satisfy the three decision-making processes necessary to field new IT capabilities:

- joint interoperability certification from the Joint Staff J6 (JS J6),
- information assurance certification and accreditation (IA C&A) from the Designated Accrediting Authority (DAA), and
- the acquisition decision from the Milestone Decision Authority (MDA).

There are likely to be several DT activities, such as integration and acceptance testing, which may occur prior to or within the window. Time must be allocated to train users and testers, and the programs have to implement support systems, such as the help desk, as intended to support the fielded system. The IA C&A typically precedes OT to obtain an authority to test, while interoperability testing may be a separate activity or in conjunction with the OT. All of these events set the stage for OT to confirm that the capability is ready for fielding.

The timeline in Figure 5 depicts a mix of both policy and practice. For example, policy requires a test concept brief 120 days prior to OT, and test plan approval 60 days prior, for programs on the T&E oversight list. In practice, OT duration varies by system; some tests can exceed what is shown by months. Likewise, final evaluation report preparation varies; the 60 days shown is probably conservative. Hence, the IOT&E test execution window can exceed 6 months. Figure 5 is not intended to imply that either interoperability or information assurance certification occurs within the time blocks shown, merely that these activities form an essential part of the IT T&E strategy and must be planned and resourced accordingly.

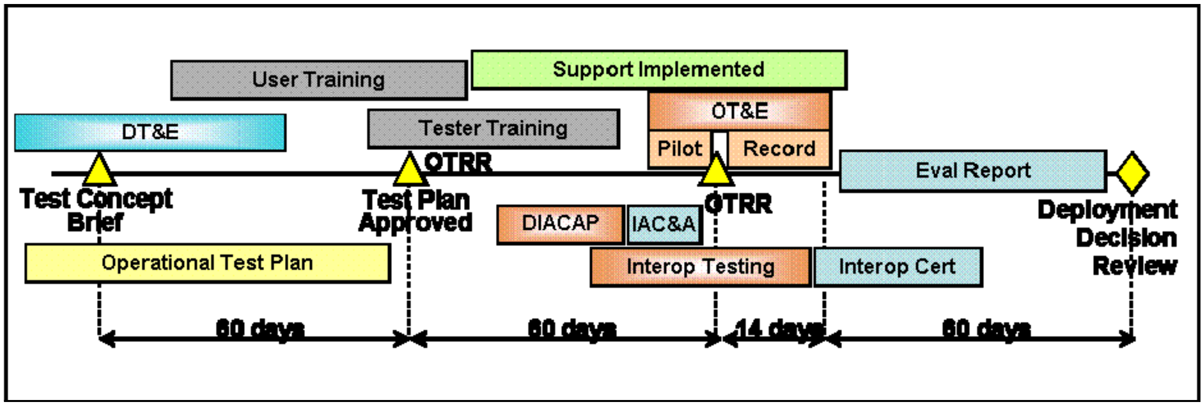


Figure 5. Test Execution Window

As stated above, effectiveness and suitability are not the only considerations for IT capabilities; information systems must also be interoperable and secure. Interoperability certification and the DoD Information Assurance Certification and Accreditation Process (DIACAP) are governed separately from the DoD acquisition system through various DoD and Chairman, Joint Chiefs of Staff directives and instructions. Separate governance processes can be disadvantageous in an acquisition system for IT; for example, it is possible today for the milestone decision authority to make a decision to buy the new capability for the department, while the DAA may deny operation on their network. In a new IT acquisition system, interoperability and information assurance processes should be integrated, not separate elements, and the testing activities associated with these certification processes should form an integral part of the IT T&E strategy.

Interoperability

One of the major complaints from the field today is lack of interoperability among the countless information systems at the strategic, operational, and tactical levels. In any new IT acquisition system, it seems clear that we are going to have to treat interoperability differently—elevate its place in the decision-making process and establish meaningful accountability. The *DoDI 5000.02* is weak in describing interoperability considerations and offers very little guidance on interoperability testing. Rather than being overseen by the milestone decision authority, interoperability is managed through a separate decision-making process governed by the *DoD 4630* directive and instruction and the *Chairman of the Joint Chiefs of Staff Instruction 6212*. As a result, joint interoperability testing is not well integrated into the overall T&E strategy of a system; for example, is the PM responsible for interoperability testing or is the operational test agent (OTA)? ... Who approves the interoperability test plan? ... Should the JS J6 sign the T&E Master Plan (TEMP)?

Interoperability is a key performance parameter (KPP), referred to today as the Net-Ready KPP (NR-KPP). The *Glossary of Defense Acquisition Terms* defines a KPP as a system characteristic “considered critical or essential to the development of an effective military capability.” The interoperability KPP has not been a stable element of the requirements system, however, and the final report of the Defense Acquisition Performance Assessment (DAPA) Project referred to the interoperability KPP as one “for which there is no method of testing.” From August 1999 to the present, the interoperability KPP has been defined and redefined four times.

The Interoperability KPP (I-KPP) was first introduced in the Requirements Generation System (RGS) in the August 1999 issuance of *CJCSI 3170.01A*. The methodology for assessing the I-KPP based on “information exchange requirements” (IERs) followed in the May 2000 *CJCSI 6212.01B*. The Joint Staff canceled the RGS in June 2003 and implemented the Joint Capability Integration and Development System (JCIDS) in *CJCSI 3170.01C*, then in November 2003, the Joint Staff replaced the I-KPP with the NR-KPP in *CJCSI 6212.01C*. The NR-KPP moved away from measurable and testable IERs to technical compliance attributes such as the “Net-Centric Operations and Warfare Reference Model (NCOW-RM),” “key interface profiles,” and “integrated architecture products”—none of which were particularly well suited to “hands-on” testing. In the March 2006 *CJCSI 6212.01D*, the NR-KPP statement changed to read in more operationally meaningful terms, but the threshold and objective requirements retained the same technical attributes. In December 2008, the NR-KPP changed again; the *CJCSI 6212.01E* replaced “key interface profiles” with the "Technical Standards/Interfaces" element, deleted the NCOW-RM, and introduced GIG Enterprise Service Profiles (GESPs)—again, not readily “hands-on” testable. Despite the continuous revisions, the NR-KPP remains arguably the least measurable and testable of all the required KPPs. An operationally meaningful, measurable, and testable interoperability KPP will be an essential element of a new IT acquisition system.

Information Assurance

Information assurance (IA) is another critical element in IT acquisition and requires security testing. Like interoperability, the *DoDI 5000.02* is weak in describing IA considerations and offers little guidance on security testing. Instead of being overseen by the milestone decision authority, information assurance is governed through the *DoD 8500* series and the *CJCSI 6510*. The *DoDI 8580.1*, *Information Assurance in the Defense Acquisition System*, does link the two governance processes though. Security T&E is another category of testing for which we do not have a standard approach in developing the overall T&E strategy; for example, who approves the security test plan? ... Should the DAA sign the TEMP?

The DoD implemented IA C&A in December 1997 with the release of the *DoDI 5200.40*, *DoD Information Technology Security Certification and Accreditation Process (DITSCAP)*. In November 2003, as threats to DoD information systems and networks were becoming increasingly apparent, the *CJCSI 6212.01C* included IA as an element of the newly defined NR KPP. In July 2006, the ASD(NII) canceled *DITSCAP*, issued interim guidance, and then, in November 2007, *DIACAP* became the process of record with the release of *DoDI 8510.01*. Completion of the DITSCAP or DIACAP process has essentially equated to satisfying the IA element of the NR KPP. Completing the DITSCAP or DIACAP process, however, has never been completely satisfying in the overall T&E strategy.

In November 1999, the Director, Operational Test and Evaluation (DOT&E) issued the *Policy for Operational Test and Evaluation of Information Assurance*. The policy required the independent OTAs to assess IA as part of the system evaluation, while leveraging to the extent possible other IA testing, such as DITSCAP security T&E, to reduce duplication. In some cases, the policy required “field penetration testing by a Red Team” as part of IOT&E. Inclusion of red teams in IOT&E adds a new level of complexity into the already challenging and resource intensive undertaking discussed earlier.

Unlike joint interoperability certification, which has a single process owner and single tester (although a recent change to the *CJCSI 6212* permits testing within the components for designated programs), IA has many owners and many testers. In our current IA C&A process, each information system has a DAA appointed by the component head or the mission area Principal Accrediting Authority (PAA). The DAA is responsible for the decision to accredit, and may authorize or deny operation or testing of their assigned information systems. The combined effect of multiple decision authorities and multiple test organizations is likely to contribute more to delay and inconsistency than efficiency and standardization. The *Defense Science Board Task Force on Achieving Interoperability in a Net Centric Environment (DSB-NC)* described the problem in these terms:

> Multiple certification processes and inconsistent retest processes exist, often resulting in the delivery of obsolete products or products that are no longer supported. Current test, evaluation, and certification (TE&C) processes take months and often years. In a wartime environment where information and technical capability is becoming more and more critical to the warfighter, a delay of months or years for redundant testing to deliver a new capability is unacceptable.

The *DSB-NC* observed that one cause of redundant testing is "Testing, evaluation, and certification that are performed by one Service or one agency are most often not accepted by other Services or agencies." The *DSB-NC*, therefore, recommended a new mandate: "test by one, accept by all." Recently, DoD PAAs signed a policy for reciprocity to accept each other's security assessments (DoD, 2009). This policy is a very positive step toward reducing redundancy and streamlining capability delivery to the enterprise.

As stated, the *DSB-IT* recommended a new, agile IT acquisition system. To its credit, the *DSB-IT* described the capability at each iteration as "tested and potentially deployable," and highlighted "integrated DT/OT" (refer back to Figure 3). Unfortunately, the *DSB-IT* retained an essentially status quo T&E approach, writing: "Following the nominal completion of three iterations, an Initial Operational Test and Evaluation (IOT&E) is accomplished prior to operationally fielding a release." This may not be the most efficient model; for example, capability developed and tested in early iterations is likely to be tested again in IOT&E. Moreover, if we conduct the IOT&E as we do it today (six months of TE&C activities), then the desired 18-month release cycle may in reality approach 24 months. More importantly, however, potentially deployable capability may be withheld from fielding until completion of the release and IOT&E. While this approach has the well-intentioned effect of reducing the churn of multiple fieldings on the operational force, it is not agile. Therefore, we might consider a model in which the decision to field, whether at iteration or release, is at the discretion of the gaining commander. Regardless of whether we test iteration or release, we are going to need a new T&E model that is responsive to agile IT programs.

Towards an Agile IT Acquisition and TE&C System

The preceding sections have made the case that acquisition of information technology in the DoD consists of multiple processes that do not necessarily share the goal of rapid delivery of enhanced capabilities to the warfighter. We lack an overarching process specifically designed for fielding IT capabilities to the enterprise. Likewise, we have challenges to overcome to create truly integrated TE&C processes that ensure capabilities are effective, suitable, interoperable, and secure.

From beginning to end—requirements definition, capability development, TE&C, governance, and operations—the department lacks agile processes designed for IT. An agile IT acquisition model must begin with agility in the requirements system; thus, one consideration (beyond the scope of this article) would be to develop a "JCIDS-light" requirements system for IT. An agile IT requirements system must shift from the current big bang, "everything in the first increment" approach to prioritizing capability needs for delivery in a series of little bangs. Additionally, we need operationally meaningful KPPs for interoperability and security.

An agile IT acquisition model requires agile oversight, so management and governance processes must be redesigned to foster rapid development and fielding cycles. DoD business IT systems have already moved to a "business capability lifecycle" (BCL) management process intended to be more flexible. The BCL "merges three major DoD processes [JCIDS, the DoD 5000 Acquisition System, and the Investment Review Board (IRB)/Defense Business System Management Committee (DBSMC) governance bodies] to provide a single governance and decision support framework to enable faster delivery of business capabilities" (http://www.bta.mil/products/bcl.html). The BCL leverages the Enterprise Risk Assessment Methodology (ERAM) "to reduce systemic risk and support informed decision making" (http://www.bta.mil/products/eram.html). Similar governance approaches could be adopted within the warfighting, intelligence, and enterprise information environment portfolios as well.

As requirements processes become more agile, programs will shift to design-build cycles based on prioritized requirements. Whereas the traditional systems engineering "V" model has the perception of being a one-way path, the agile development lifecycle is more iterative, less sequential. The TE&C community must be ready to engage agile programs through equally agile processes; the six-month test execution window that occurs at the end of an increment today has to be shortened and moved well left in the schedule, to focus on the development iterations. A key element of tester agility will be formation of a capability test team to merge the traditional DT, OT, interoperability, and security test activities into a comprehensive TE&C strategy.

Our objective in T&E should be mission-focused agility: rapidly composable mission-oriented test plans that permit objective assessments of technical and operational capabilities and limitations in each iteration. Likewise, we need agile DIACAP and interoperability certification, where "test by one, accept by all" is the norm. For capabilities developed in six-month iterations, the capability test team should be able to complete the entire test execution window—plan, execute, report—in six-weeks or less. Figure 6 depicts the TE&C paradigm shift. This can be accomplished only through a highly collaborative process that is responsive to changing requirements priorities and developer agility. Essential to this approach will be early and continuous involvement from the user community. In this model, the overarching theme is "build a little, test a little (learn a lot), field a little." Then, as capabilities are deployed, the fielding paradigm should be "start small, scale rapidly," while continuously monitoring to ensure the capability performs as desired.

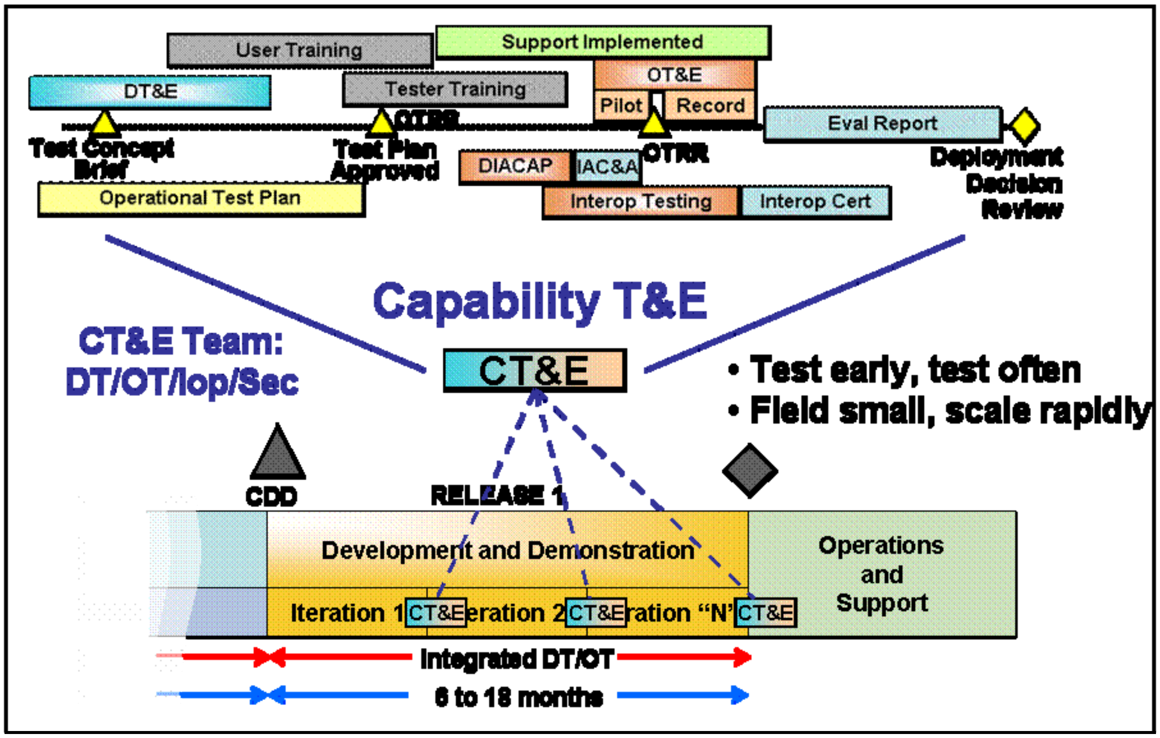


Figure 6. Agile T&E

Summary

Information technologies evolve rapidly, as is abundantly evident in the commercial sector. As the DoD acquires IT to enhance warfighting capabilities, we need to become more agile. Agility cannot just occur in capability development either; all aspects of the IT acquisition system must be redesigned for agility. To be responsive to operational requirements, and to ensure the capabilities work as intended, test, evaluation, and certification must move at the speed of need. The Defense Science Board reports provide a good starting point from which to build a new model for acquisition of IT; now, let's take the next bold step to implement agile processes that deliver enhanced IT capabilities for the warfighter.

References

CJCS. (1999). *Requirements generation system* (CJCSI 3170.01A). Washington, DC: Author.

CJCS. (2000). *Interoperability and supportability of national security systems and information technology systems* (CJCSI 6212.01B). Washington, DC: Author.

CJCS. (2003a). *Interoperability and supportability of national security systems and information technology systems* (CJCSI 6212.01C). Washington, DC: Author.

CJCS. (2003b). *Joint capability integration and development system* (CJCSI 3170.01C). Washington, DC: Author.

CJCS. (2006). *Interoperability and supportability of national security systems and information technology systems* (CJCSI 6212.01D). Washington, DC: Author.

CJCS. (2008). *Interoperability and supportability of national security systems and information technology systems* (CJCSI 6212.01E). Retrieved March 30, 2010, from http://www.dtic.mil/cjcs_directives/cdata/unlimit/6212_01.pdf

DAU. (2009, June 15). *Integrated defense acquisition, technology and logistics life cycle management system chart* (Vers. 5.3.4). Retrieved March 30, 2010, from https://acc.dau.mil/IFC/index.htm

DoD. (1997). *DoD information technology security certification and accreditation process (DITSCAP)* (DoD Instruction 5200.40). Retrieved March 30, 2010, from http://iase.disa.mil/diacap/i520040.pdf

DoD. (2004). *Interoperability and supportability of information technology (IT) and national security systems (NSS)* (DoD Directive 4630.05). Retrieved March 30, 2010, from http://www.dtic.mil/whs/directives/corres/pdf/463005p.pdf

DoD. (2007). *DoD information assurance certification and accreditation process (DIACAP)* (DoD Instruction 8510.01). Washington, DC: Author.

DoD. (2008). *Operation of the defense acquisition system* (DoD Instruction 5000.02). Retrieved March 30, 2010, from http://www.dtic.mil/whs/directives/corres/pdf/500002p.pdf

DoD. (2009). *DoD information system certification and accreditation reciprocity*. Memorandum. Retrieved March 30, 2010, from http://www.doncio.navy.mil/Download.aspx?AttachID=1055

National Research Council (NRC). (2006). *Testing of defense systems in an evolutionary acquisition environment*.

OUSD(AT&L). (2009a). *Department of Defense policies and procedures for the acquisition of information technology*. Retrieved March 30, 2010, from http://www.acq.osd.mil/dsb/reports.htm

OUSD(AT&L). (2009b). *Report of the Defense Science Board Task Force on achieving interoperability in a net centric environment*. Retrieved March 30, 2010, from http://www.acq.osd.mil/dsb/reports.htm

United States Code. (2009). Title 10, Chapter 4, § 139. Retrieved March 30, 2010, from http://uscode.house.gov/download/pls/10C4.txt

US Congress. (2009). *Weapons acquisition system reform through enhancing technical knowledge and oversight act of 2009*. Retrieved March 30, 2010, from http://thomas.loc.gov/cgi-bin/query/z?c111:S.454:

US Congress. (2010). *National defense authorization act of 2010*. Retrieved March 30, 2010, from http://thomas.loc.gov/cgi-bin/query/z?c111:S.454:

Integrated Testing and Independent Evaluation Model

Christopher DiPetto— Chris DiPetto is the Principal Deputy for Developmental Test and Evaluation (DT&E) in the Defense Research and Engineering Directorate of the Office of the Under Secretary of Defense for Acquisition, Technology and Logistics. In this capacity, he implements DT&E policy and program oversight to ensure service programs are realistic, relevant, and in compliance with Department of Defense and congressional directives. DiPetto is an aerospace engineer, a flight test engineer and a graduate of the US Navy Test Pilot School with more than 30 years of experience in the DoD. He is a 1994 Harvard Kennedy School of Government Senior Executive Fellow, a 1993 graduate of the Defense Systems Management College's Program Management Course, and a Level III acquisition and T&E professional.

Chris DiPetto, Principal Deputy
Office of the Under Secretary of Defense
Developmental Test and Evaluation
(Christopher.DiPetto@osd.mil, 703-695-4421)

Platform Areas

- Bridging System T&E from DT to OT
- Strengthening Early Developmental Testing Rapid Acquisition T&E

The foundation for future, successful, project-level, Defense T&E, especially in a Net-Centric environment, will be the early creation of an Integrated T&E Team (ITT). This presentation delineates a model for the successful development of a project-level ITT. Included are recommendations about how to maximize the use of the ITT throughout the System Development Life Cycle. The model describes methods to enhance the product's acceptability to the users and helps apply a better use of limited testing resources. When applied, the model will help with the design and implementation of an ITT.

Topic Addressed

The presentation specifically looks at how to improve the quality of the product while helping to shorten the time to market. The model focuses on the inter-relationships of the four members of the T&E community—Developmental Test & Evaluation (DT&E), Operational Test & Evaluation (OT&E), Interoperability (IOP), and Information Assurance (IA)—and outlines techniques for eliminating or reducing the effects of the current "Silo" testing methodology.

New DoD systems architectures and hardware have made it intuitively clear that testing must be seriously updated and replaced with an agile, streamlined tactic for Net-Centric systems that will help assure that the warfighters have the critical information they need to perform their mission successfully.

Relevance to Conference Theme

DoD T&E must advance into the integrated world in order to be successful in all the facets of a Net-Centric environment. New technological designs, including Family of Systems and Service-Oriented Architecture, require a more flexible T&E. The ITT concept comprises a series of progressive system verification and qualification processes that are intensity driven by an iterative succession of risk identification efforts.

Panel #8 – Bid Protests: Considerations for Prevention and Resolution

Wednesday, May 12, 2010	
1:45 p.m. – 3:15 p.m.	**Chair: Lenn Vincent, RADM, USN, (Ret.)**, Industry Chair, Defense Acquisition University ***Understanding and Mitigating Protests of Major Defense System Acquisition Contracts*** **Steven Maser** and G. Frederick Thompson, Willamette University ***Innovations in Defense Acquisition: Bid Protests*** **Peter Coughlan** and Bill Gates, Naval Postgraduate School ***Better Acquisition Management Through ADR and Other Practices for Preventing and Resolving Bid Protests*** **Max Kidalov**, Diana Angelis, Ben Sheinman and Paul Benishek, Naval Postgraduate School

Understanding and Mitigating Protests of Major Defense System Acquisition Contracts

Steven Maser—Steven Maser is a Professor of Public Management and Public Policy at Willamette University's Atkinson Graduate School of Management. His courses are about public policy, business and government relationships, negotiation, and conflict management. He has served on numerous boards and commissions, most recently chairing Portland's citizens' task force on bringing professional soccer to the City. Professor Maser, who holds degrees in Political Science from MIT and from the University of Rochester, has been a visiting scholar at Yale Law School, the Olin School of Business at Washington University in St. Louis, and the Hatfield School of Government at Portland State University. His academic research has appeared in journals such as the *Harvard Journal of Law and Public Policy*; the *Journal of Law, Economics, and Organization*; the *Journal of Law and Economics;* the *Journal of Economic Behavior and Organizations,* the *Journal of Politics, and the American Journal of Political Science*. He is on the editorial board of *Rationality and Society*. Maser has consulted on organizational design and conflict management for energy and construction engineering companies.

Fred Thompson—Fred Thompson is the founding editor of *International Public Management Journal*. He is the author of Responsibility Budgeting at the Air Force Materiel Command, *Public Administration Review* (Jan/Feb 2006, with Michael Barzelay), *Reinventing the Pentagon* (1994, with LR Jones), *Digital State at the Leading Edge* (2007, with Sandy Borins, Ken Kernaghan, David Brown, Nick Bontis, and Perri 6) and over a hundred other books and articles in journals such as the *Academy of Management Review*, *Journal of Policy Analysis and Management*, *Public Choice*, and the *American Political Science Review*.

Mr. Thompson has served as a contributing editor to *Policy Sciences* and *Journal of Comparative Policy Analysis*, and on more than a dozen other editorial boards, currently including, *Public Money and Management* and *Public Budgeting & Finance*. He is the recipient of PAR's Mosher Award, the NASPAA-ASPA Distinguished Research Award, and ABFM's Aaron B. Wildavsky Award for lifetime contributions to the field of public budgeting and finance. He recently served on the NRC-NIM's Committee on Accelerating the Research, Development, and Acquisition of Medical Countermeasures against Biological Warfare Agents and as a member of the United Nations Development Program's Blue Ribbon Commission on Macedonia. He has consulted on Treasury practice and policy in Ukraine, Georgia, and Armenia.

Mr. Thompson is the Grace and Elmer Goudy Professor of Public Management and Policy, Geo. M. Atkinson Graduate School of Management, Willamette University, Salem, Oregon. During his most recent sabbatical, he was a visiting professor in the Department of Management and senior fellow in the Centre for Risk and Regulation at the London School of Economics.

Abstract

The proposed research explores whether features of the source solicitation process—from the identification of need through contract award and potential protest—trigger bid protests. Building upon concepts drawn from transaction cost economics and dispute systems design, we explore patterns and elements in GAO-digested decisions posted during 2001-2009. We interviewed over 25 decision-makers spread across three contracting commands, three prime contractors, bid protest attorneys, the Office of the Secretary of Defense, Senate staff, industry trade associations, and the GAO.

The research treats the source solicitation process—contracting and protests—as a management system. We look at the alignment of organizational strategy, policy, personnel

(e.g., knowledge, skills and aptitudes), structure, and monitoring elements (e.g., performance measures). Misalignments can set the stage for errors that trigger protests.

Preliminary analyses are consistent with implications from our conceptual approach. We find no significant differences in protests and sustains as a function of weapon vs. nonweapon acquisitions. However, the bulk of the protests are small- to medium-sized, rejected offerors protesting small- to medium-sized awardees, particularly for contracts of long duration and high uncertainty in which evaluation is particularly difficult. Large companies with substantial knowledge and resources tend to be more successful in protesting, regardless of the size of the awardee. Analysis of the interview data suggests there exist management practices that expose source selections to higher risks of error that could generate protests. The paper concludes with recommendations, some of which are already under consideration at the DoD, for retaining the benefits of bid protests while mitigating their costs.

Innovations in Defense Acquisition: Bid Protests

Peter Coughlan—Peter J. Coughlan is an Associate Professor of Business and Economics at the Graduate School of Business and Public Policy, Naval Postgraduate School. He received his PhD in Economics from the California Institute of Technology in June 1999. Prior to joining the Naval Postgraduate School in 2004, Dr. Coughlan was an Assistant Professor of Business Administration in the Strategy Group at the Harvard Business School. Dr. Coughlan's research interests include experimental economics and game theory/mechanism design. He also has research interests in strategic and competitive dynamics, particularly applied to technological change.

Peter J. Coughlan
Graduate School of Business and Public Policy
Naval Postgraduate School
Monterey, CA 93943-5000
Fax: (831) 656-3407
E-mail: pjcoughl@nps.edu

William R. Gates—William R. Gates is an Associate Professor of Economics and Dean of the Graduate School of Business and Public Policy at the Naval Postgraduate School. He received his PhD in Economics from Yale University in 1984. Before joining the Naval Postgraduate School in 1988, Dr. Gates worked as an economist at the Jet Propulsion Laboratory in Pasadena, CA. Dr. Gates' research interests include game theory (matching games and auctions), incentives and asymmetric information (incentive contracting), and public goods (economics of defense alliances).

William R. Gates
Graduate School of Business and Public Policy
Naval Postgraduate School
Monterey, CA 93943-5000
Phone: (831) 656-2754
Fax: (831) 656-3407
E-mail: bgates@nps.edu

Abstract

The research will explore bid protests from the contractor's viewpoint. To accomplish this, we must first ask, "what are the contractor's goals and objectives in protesting?" The answer is likely "profit maximization" (i.e., a return to shareholders). We must also ask "what constrains the contractor's behavior?" Economists assume contractors are rational players (constrained by shareholder/owners) and are likely to engage in a benefit-cost analysis to determine whether or not to protest, where the benefit from protesting is the expected value of winning (probability of winning $x payoff), and the cost includes the cost of filing the protest and any additional costs (reputation effects, etc.). In deciding whether or not to protest, a company will weigh the discounted present value (or option value) of their expected benefits against the costs. This research stream will develop a split procurement model in which the contractor's share of the final contract depends on the contractor's relative procurement costs; the closer the relative costs, the more equal the procurement shares. Splitting the procurement lowers the contractor's probability of profits from winning the protest; at the same time, it may increase the DoD's procurement costs. This model examines the tradeoff between expected profits from a contract protest and the DoD's procurement costs. Research Issue Develop is a procurement model that generates testable hypotheses about the DoD's Return on Investment (ROI) of using a split procurement to reduce the expected benefits of a bid protest for the losing bidder(s). Research Result

Simulation results examine the tradeoff between the contractor's expected profits from a bid protest and the DoD's procurement costs.

This model is part of a technical report expected in December 2009 as the deliverable from our FY2009 funded research project. The presentation will summarize completed research.

Keywords: Bid Protest, Contract Protest, Split Procurement

Better Acquisition Management Through ADR and Other Practices for Preventing and Resolving Bid Protests

Max Kidalov—Max Kidalov received a BS *cum laude* in business economics & finance and a JD from the University of South Carolina. He will also receive a Master's of Government Contracts Law from the George Washington University in May 2010. Kidalov is a graduate of the Congressional Research Service Legislative Institute.

He is a member of the bars of the Supreme Court of South Carolina, the US Court of Federal Claims, and the US Court of Appeals for the Federal Circuit, as well as a member of the Small Business Advisory Council, US Department of Energy. His professional legal and policy experience in government, private, and non-profit sectors focused primarily on procurement, federal claims, and innovation issues., including positions such as the following: Vice President & General Counsel at a small Federal government contractor with over 400 employees and offices in Washington, DC, California, and across the nation; Vice Chairman of a trade association representing small hi-tech government contractors; counsel on the US Senate Committee on Small Business and Entrepreneurship with responsibilities for over $120 billion in procurement and innovation programs, oversight of waste, fraud, and abuse, nominations, and matters of Senate procedure; attorney at a private practice in Washington, DC, representing clients in Federal contracting matters; attorney serving as a two-term clerk to Chief (now Senior) Judge Loren A. Smith of the US Court of Federal Claims; Past Vice Chairman, Bid Protest Committee, American Bar Association Section of Public Contract Law. Kidalov was awarded the Tibbetts Special Award for outstanding achievements in the advancement of the Small Business Innovation Research Program (typically awarded once in a lifetime) and the Award of Appreciation from the Federal Laboratory Consortium for Technology Transfer. He considers his greatest accomplishments to be finishing five Marine Corps marathons and one Blue Ridge Relay and summiting Pikes Peak twice.

Max V. Kidalov, JD, Esq.
Assistant Professor, Procurement Law & Policy
Naval Postgraduate School
mkidalov@nps.navy.mil (202) 406-0205

Diana I. Angelis—Diana I. Angelis, Associate Professor, studied accounting at the University of Florida and received a BS in business administration in 1977 and a BS in electrical engineering in 1985. She received her PhD in industrial and systems engineering from the University of Florida in 1996. Her research interests include cost accounting, activity-based costing, valuation of R&D and acquisition innovation. She was commissioned an officer in the United States Air Force in 1984 and served as a program engineer until 1989. Dr. Angelis is a Certified Public Accountant and a Lt Col in the US Air Force Reserve. She joined the DRMI faculty in 1996.

Contact: diangeli@nps.edu, or (831) 656-2051

Executive Summary

Recently, defense leadership and Congress have expressed concerns over costs and delays from bid protests of procurement contracts. This paper addresses these concerns by examining how well the buying agencies, particularly military agencies, manage the bid protest process to save time and taxpayers' dollars. Specifically, the paper examines the extent to which the buying agencies utilize all strategies authorized for prevention, resolution, and mitigation of bid protests. This paper treats bid protests as a

manageable part of the acquisition process—an approach fundamentally different from the typical implicit treatment of bid protests as a legal process controlled by legal nuances and not generally susceptible to management.

Successful resolutions of protests depends on a number of factors, including government and private-sector protest management and litigation strategies, Alternate Dispute Resolution (ADR) policies of federal agencies, legal and regulatory requirements, and remedies available to contractors. The paper examines these factors in five ways. First, the paper converts legal rules and procedural milestones into an acquisition manager's guide to managing bid protests at various stages of the procurement process and in various legal forums. This guide identifies key decision points in the protest process and enables managers to make informed trade-offs between time, cost, and positional advantages offered by the various prevention, resolution, and defense strategies. To that end, the paper identifies and analyzes best ADR practices and other remedies and preventions for resolving bid protests. Areas examined include processes and remedies utilized by selected federal agencies and obstacles to improved cooperation between industry and government that may preclude win-win resolutions to bid protests. Guidance on the most efficient strategies in terms of delay and cost reduction is offered.

Second, the paper contains a survey of acquisition and legal professionals regarding their perceptions, opinions, and recommendations on bid protest practices and the use of ADR procedures. Survey objectives were to identify ADR and other process improvement recommendations that are crucial to effective contracting and support the government's efforts to improve adjudicative forums for resolution of contract disputes and bid protests. Research suggests that agencies can mitigate protest expenses and interruptions by managing the protest process in a systematic, business-like way. At the present time, agencies rarely use most procedural tools that are required or authorized under federal laws and regulations to reduce time delays and costs from bid protests.

Third, the paper examines some common objections to the use of ADR practices and other protest time- and cost-mitigation strategies.

Fourth, the paper applies the bid protest management paradigm to selected actual procurements and highlights where the agencies could have achieved lower costs or faster procurements with different protest management strategies.

Fifth, the paper makes recommendations for buying agencies and procurement policymakers. Among other things, the paper recommends energetic agency approaches to preventing disputes (e.g., quality debriefings), and dealing with disputes (e.g., formal cost-benefit analysis of agency defense strategies, strong defense of agency actions, and full use of ADR methods). The paper also recommends measures to ensure ADR as the default method for settling bid protests.

Panel #9 – Software Control and Development for DoD Weapons Systems

Wednesday, May 12, 2010	
1:45 p.m. – 3:15 p.m.	**Chair: Rear Admiral James J. Shannon, US Navy**, Commander, Naval Surface Warfare Center
	Ontology-based Software Repository System
	Jean Johnson, Naval Postgraduate School
	Enabling Software Acquisition Improvement: Government and Industry Software Development Team Acquisition Model
	Joe Heil, Naval Surface Warfare Center, Dahlgren
	On Open and Collaborative Software Development in the DoD
	Scott Hissam, Charles Weinstock and Len Bass, Carnegie Mellon University

Ontology-based Software Repository System

Jean Johnson—Jean Johnson is a Lecturer in the Systems Engineering Department at the Naval Postgraduate School, Monterey, California. After serving on active duty in the US Navy, she supported the NAVSEA Warfare Systems Engineering Directorate (NAVSEA06) before coming to Naval Postgraduate School. Her current research focus areas are software repositories to enable reuse and the use of modeling and simulation in DoD acquisition. She is currently a PhD candidate in Software Engineering.

Jean M. Johnson
Systems Engineering Department
Naval Postgraduate School
Monterey, CA
e-mail: jmjohnso@nps.edu

Abstract

The reuse of software and related artifacts is a key tenant of DoD acquisition improvement initiatives, including the Naval Open Architecture program. While there are many inhibitors of reuse, software repositories are considered enablers in that they provide a central store of artifacts as well as capabilities for search, retrieval, and reconfiguration of existing components into newly developed systems. However, current software repositories lack robust search and discovery capabilities and are thus limited enablers.

This research expands on previous efforts to reform the organizational approach to software repositories by using ontologies as the framework of repository information. By combining metadata with domain, architectural, and other information, more sophisticated search techniques are enabled. In this paper, we describe the approach and demonstrate, through a use case, a new type of search that takes advantage of the context provided by the ontologies and emphasizes human interaction. New navigation techniques will be employed that guide human users, offering suggestions based on projected needs. The improved search capability will encourage developers to consider reuse and improve the software reuse enabling power of software repositories.

Introduction

Architectural and other information about software systems may be captured in a domain-specific ontological framework for a software repository to enable new types of searches. The relations in the ontology are used to determine associations between repository artifacts to facilitate intuitive navigation. A fisheye graph view enables visualization of artifacts within a contextual framework that provides suggestions based on users' actions. The emphasis is to provide a rich human interface that maximizes the combined knowledge of both the community of human users and the computer-based repository system. Capturing ontologies in standard formats results in an extensible framework, which can easily be shared between multiple repositories using XML-based technologies, thereby improving interoperability.

The initial target of the research is the US Navy's Software, Hardware Asset Reuse Enterprise (SHARE) repository. In 2007, researchers at the Naval Postgraduate School (NPS) were tasked to develop a component specification and ontology for the SHARE software repository, which was then recently established. The component specification and ontology provide a rich structural and semantic framework for SHARE that enables multiple kinds of search and discovery techniques. Follow on work was assigned to develop designs for a software repository tool that will fully utilize the repository framework to improve the usefulness of SHARE. This paper captures the results of the effort with the intention of illustrating how the approach can be extended to additional applications and domains.

Current State of the Art

Improvements to the current state of the art for software reuse repositories are required (Shiva & Shala, 2007). Current software reuse repositories such as SourceForge (http://sourceforge.net) and Comprehensive Perl Archive Network (CPAN) (http://www.cpan.org) typically enable search and discovery of software artifacts through keyword searches over metadata or browsing through metadata via broad categories of artifacts (i.e., faceted classification) (Guo & Luqi, 2000). This approach is only effective in relatively small repositories or in situations where the users are well familiar with the contents of the repository. This is because successful discovery depends on the searchers' ability to express the desired results in the same vocabulary used by the artifact submitter or repository manager. In other words, if you know exactly what you are looking for, and how to ask for it, you will find it.

Search tools are not designed to aid users that do not already know the desired outcome of the search. A repository interface should guide searchers through the discovery process based on the users' context, suggesting search threads and recommending items for retrieval. However, current repositories do not support this type of repository navigation.

Current repositories in general do not relate artifacts to any context other than characterizations identified within the metadata. These typical characterizations enable a list-type search result, similar to the popular children's card game "Go Fish." Users can request, "Give me all of your artifacts of type xyz." Unfortunately, if you don't know what you want, you cannot ask for it. If you ask for artifacts of type xyz and there are no artifacts labeled as such in the repository, the search ends (go fish). Without contextual linkages between artifacts in the repository, guided searches are not possible. But, with a guided search, the system can recommend other potential solutions based on a given context.

Proposed Improvements

This research aims to produce a new kind of software repository that addresses the current shortfalls. There are four design characteristics that constitute the novelty of the approach.

First we take a broad view of reuse. Although reusable software artifacts are often defined to include any product related to the development of software, typical software repositories enable only the reuse of code or executable files and maybe some architecture and design products. We consider all types of artifacts from the software engineering life cycle—including requirements, test scripts, etc.—and plan for them in the design.

Second, we propose the use of ontologies to provide the contextual framework for the repository. We will show how these ontologies can be used to guide users to discover artifacts that they may find useful.

Third, we will exploit the use of domain-specific information in our repository design. By narrowing the reuse efforts to a particular domain, we increase our likelihood of developing a repository that will be relevant to the user group. In the same way that successful strategic reuse has most often been associated with a domain-centric or product line development approach, a reuse repository is likely to be most successfully employed if it embodies, and is limited to, the domain knowledge base.

Finally, we propose the use of multiple views that will allow the user to view the repository contents in a comfortable visual arrangement for their particular use, depending on the experience of the user. The multiple views approach is analogous to Kruchten's multiple views of software architecture as depicted in his famous “4+1” paper (Kruchten, 1995).

Paper Organization

The remainder of the paper is organized as follows. In section two, we describe the proposed repository system. In section three, we provide a use case demonstration of the repository functionality. Sections four and five provide discussions of relevant related work, a summary, and suggested related future work.

A New Repository Tool

We propose the development of a new software repository tool that will encourage improved (increased and more effective) reuse of software artifacts by presenting information about the contents of the repository to the user in ways that allow the user to project their individual context onto the repository. Here we consider the user's context to include their instantaneous progress in the development process at the time of search, the system modeling paradigms with which they are comfortable (e.g., UML or DODAF), and their understanding of the particular domain for which they are developing systems.

In this section we will present the key features of the new repository tool, including an explanation of what is meant by the guided search and descriptions of the proposed ontology-based repository framework and envisioned visualization techniques.

Guided Search

The tool will support the human user by enabling smart navigation of the repository contents based on information collected. It is important to note that we can never completely automate the search process if we intend to incorporate unique user situations into the search algorithms. Therefore, the tool we suggest will guide the user through a search but cannot complete the search on its own. This is an important claim since the resulting necessary interaction between human and machine is an essential feature of the tool.

We envision a graphical “point and click” user interface that enables navigation of repository contents reflecting user interests. This requires an interface that allows users to project their context onto the search mechanisms. In other words, the users bring particular information needs and goals based on the problem they are trying to solve. For example, users may seek particular functionality best obtained through a functional organization of the information in the repository; they may seek particular artifacts best obtained through a document resource organization of the information; or, they may seek information on certain testing methodologies that have been applied so that a work activity organization of the information would best apply.

In our approach, relationships among assets and artifacts are recorded in the repository ontology framework, allowing users to navigate to and select artifacts based on their various relationships to an item of interest. Simple questions are posed of the user to initiate and direct the search at key moments. The items that are shown most prevalently to the user are determined by a prioritization scheme that takes into account previous user actions. Additionally, the user has the ability to "turn off" the relationships that are not helpful for the current search objectives. These capabilities are demonstrated in more detail for the SHARE repository in section three.

Repository Framework

In this section, we describe the framework for the repository that enables the functionality described. In (Johnson & Blais, 2008), we proposed two major aspects for this framework: a component specification and ontology. The component specification is a description or model of the items in the repository and consists of both typical metadata and a behavioral model of the component. The ontology describes concepts and relationships to create various perspectives or contexts for examining the contents of the repository. These aspects of the framework are discussed further below.

Component Specification—Metadata

The metadata for each artifact should incorporate all necessary data for discovery and implementation. The metadata will aid repository users in determining if the item is suited for their use and will provide information about how to use the asset when it is retrieved. We refer to the "standard" or "typical" metadata since there are many existing examples of metadata similar in concept to that developed for the SHARE repository. The intent of the metadata is to describe artifacts and assets contained in the repository in sufficient detail to aid the repository user in determining if the item is worth retrieving for a particular use.

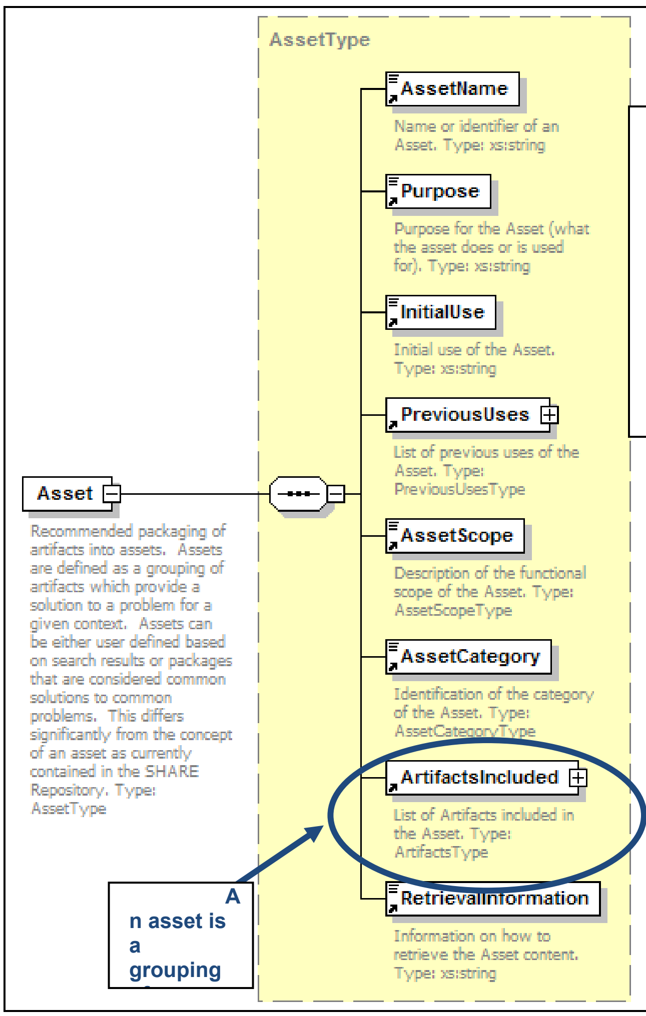
To be clear, we must provide our definition of two terms. The Navy Open Architecture (OA) program has adopted similar definitions for asset and artifact as those used in the Object Management Group (OMG) Reusable Asset Specification (RAS). In the RAS, artifacts are defined as "any work products from the software development lifecycle," and assets are a grouping of artifacts that "provide a solution to a problem for a given context" (Object Management Group, 2005). Accordingly, the RAS describes an approach for packaging artifacts into an asset. This is consistent with the current SHARE approach and remains consistent in the proposed metadata XML schema described here. Artifacts are described individually and the asset description consists of the listing of artifacts included for that asset along with some descriptive information (see Figure 1).

The artifacts schema is designed to be flexible in its implementation. All elements, types, and attributes in the schema are defined globally to facilitate reuse. The root element, *Artifacts*, is simply a container for any number of artifacts contained in a single instance of the schema. A specific artifact can be incorporated into the file in one of three ways—by providing the full artifact description or by reference, either to a physical location or by URL.

The guts of the artifact metadata are captured in the *ArtifactDescription* sub-element of the full artifact description. The information necessary to describe the artifacts differs depending on whether the artifact is software code or some other type. Therefore, the schema allows a choice between two types of artifact descriptions as shown in Figure 2. The *NonCodeDescription* element applies to any artifact not considered software code. The group of elements contained therein (shown in Figure 3) is also required for artifacts that fall under the *CodeDescription* element category, but additional elements are required for code artifacts. For brevity, we have presented a small subset of the schema developed.

Detailed descriptions of each element of the SHARE metadata schema are available in (Johnson & Blais, 2008).

While much of the metadata described is lacking in novelty, a subset of the elements identified as part of the *NonCodeDescription* element begins to reveal the unique approach we have developed. First, the *ArtifactType*, *ApplicableSystems*, and *ObjectiveArchitectureTags* all serve the specific purpose of relating individual repository artifacts to the ontological framework described later in the section. Second, the *SoftwareBehaviorDescription* element is a specific focus of the design. Since this piece of the component specification is not commonly incorporated into repositories in a standardized manner, we feel it is a specific focus area to identify the appropriate representation mechanisms for software behavior in the repository context.



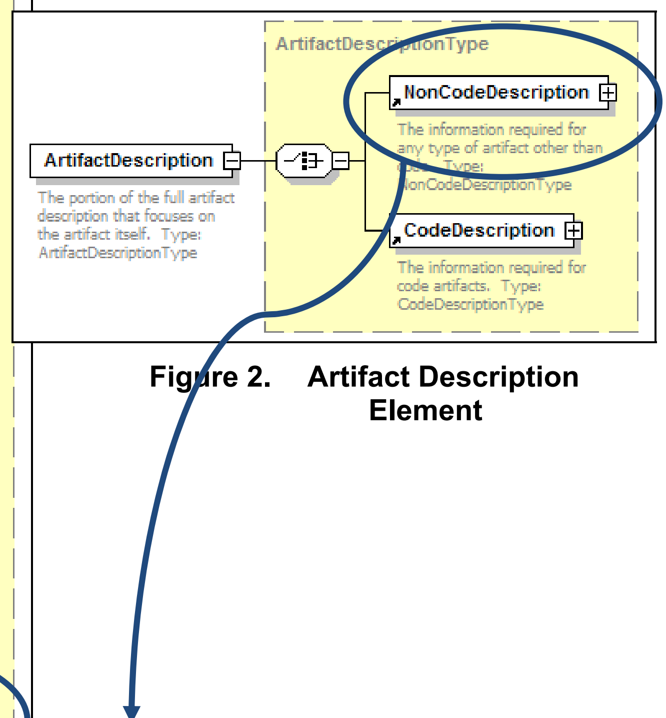


Figure 2. Artifact Description Element

Figure 1. Asset Element[35]

[35] Diagrams of the XML structures have been generated by Altova XML-Spy. The product is available at http://www.altova.com/products/xmlspy/xml_editor.html. Altova offers a free 30-day license for trial use of the product. The Altova presentation of elements incorporates a plus (+) symbol on the right edge of a box to indicate that the element contains sub-elements.

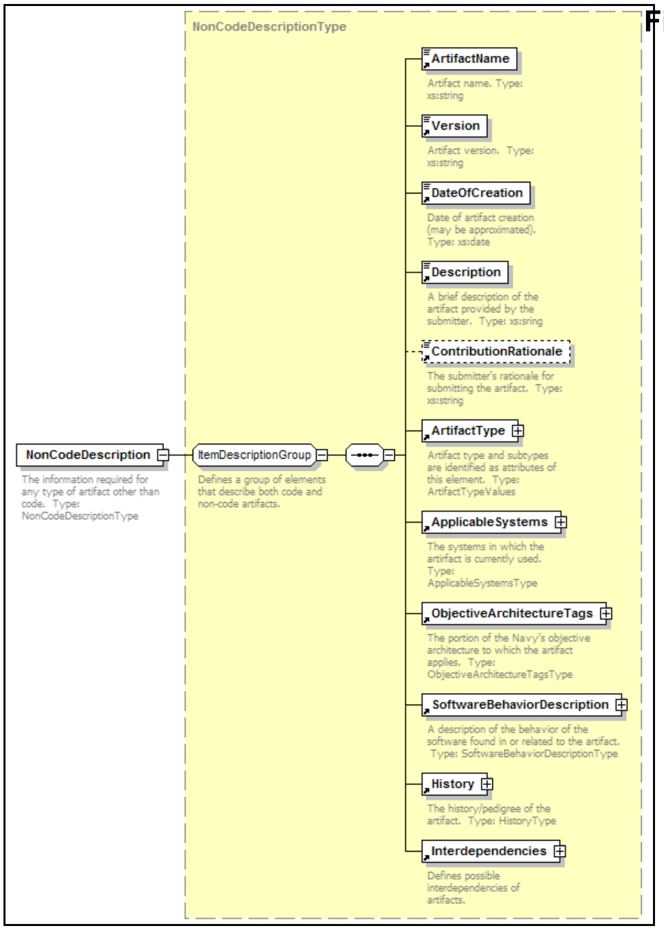


Figure 3. NonCodeDescription Element

Component Specification—Software Behavior Description

One of the loftier goals of a software repository is to support automatic composition of systems from reusable components. This is a difficult problem, which many have tried to solve.[36] It is especially difficult if the components were not originally designed for reuse. As a necessary first step towards more sophisticated uses of a repository, behavioral descriptions must be machine readable in order to support automated search and discovery. Furthermore, the behavior descriptions must be formalized and consistently applied to each item in the repository if the intent is to automatically compose them into a larger functioning system.

The array of contributors to SHARE and non-homogeneity of the repository contents requires caution in dictating standards that impact the development processes of the asset developers. It would not, for example, make sense to insist on a specific component technology across all Navy software programs in order to standardize the interface protocols. Yet, this is the type of precision required to truly enable software composition from reusable components. Recognizing that we fall short of this goal for this phase of the effort,

[36] The proceedings from the International Symposium on Software Composition, an annual event, provide examples of research into the breadth of research topics currently being pursued in the area of software composition. The web site for the 2008 conference is located at http://www.2008.software-composition.org/.

we have sought a balance between method robustness and ease of implementation in our software behavior specification.

In our approach, component behavior is potentially captured in two ways (as shown in Figure 4). First, the functionality of the software related to the artifact is identified by a list of functions selected from the Navy's Common System Function List (CSFL). The Navy's CSFL is a listing of functions that are performed by Navy systems. It provides a standardized taxonomy for the functionality found in this application domain. We have converted the CSFL into an ontology expressed in the Web Ontology Language (OWL), and acceptable entries for the *CommonSystemFunction* metadata element are validated against this ontology. If we require asset submitters to state the functionality of the components in these terms, we can then build the tools to guide users in selecting desired behavior in the same terms.

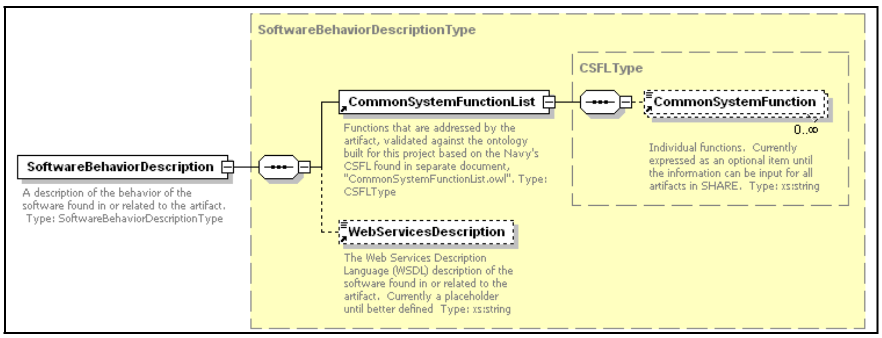


Figure 4. SoftwareBehaviorDescription Element

Second, the interface information may be captured as a Web Service Description Language (WSDL) document. In this research, we explored characterization of software interfaces based on current and emerging Web Services (e.g., WSDL) and Semantic Web Services (e.g., WS-BPEL, OWL-S) approaches. The work is preliminary, and it will be necessary to adopt a more precise description of code artifacts to introduce these techniques. As a start, we included the option of inserting a WSDL description of software services in the *SoftwareBehaviorDescription* element.

As the DoD moves toward Service Oriented Architectures (SOA), services may become a more frequent part of the SHARE repository. In that case, the WSDL describing those services (often automatically generated by the software development or execution environment of modern software systems) can be directly utilized in the repository to provide a detailed view of the service interfaces and operations. For software that is not developed and deployed as services, it is still feasible for public methods within the software to be parsed automatically to create WSDL-like descriptions. These may be incomplete descriptions with respect to full compliance to WSDL structures, but could still provide a well-defined way to describe the software for search and discovery.

Ontology Framework

The ontology framework provides contextual semantics that describe relationships among items in the repository to aid in associating artifacts with users' needs. It includes descriptions of the relationships of the components to form a contextual model of the repository items.

The taxonomies/ontologies we developed for SHARE are based on several types of relationships between the items in the repository, as well as with relevant domain architectural descriptions and other information. They capture an artifact's place in the software engineering lifecycle, its architectural fit in its original system, its architectural fit in any system in which it was subsequently used, and identification of the component's fit in the Surface Navy Objective Architecture. Each of these ontologies is described in this section.

Software Lifecycle-Artifact Ontology

The software lifecycle-artifact ontology relates software artifacts to activities in the software engineering lifecycle. Aside from the "has subclass" relationship that exists in the software artifacts and lifecycle activities taxonomies, there are four additional properties that link these class structures:

- mayProduceArtifact—For each lifecycle activity, identifies which artifacts are most commonly produced as a result of that activity. The inverse property is oftenDevelopedDuring. The property maps items in the LifecyclePhases class (domain of the property) to the SoftwareArtifact class (range of the property).
- oftenDevelopedDuring—For each artifact, identifies the activity or activities that most commonly produce it. The inverse property is mayProduceArtifact. The property maps items in the SoftwareArtifact class (domain) to the LifecyclePhases class (range).
- mayRequireUseOf—For each lifecycle activity, identifies the most commonly needed artifacts. The inverse property is oftenUsedDuring. The property maps items in the LifecyclePhases class (domain) to the SoftwareArtifact class (range).
- oftenUsedDuring—For each artifact, identifies the activity or activities in which it is most commonly needed. The inverse property is mayRequireUseOf. The property maps items in the SoftwareArtifact class (domain) to the LifecyclePhases class (range).

To demonstrate, a diagram showing the relations captured for the RequirementsSpecification and RequirementsDatabase classes of the software artifact taxonomy are shown in Figure 5.

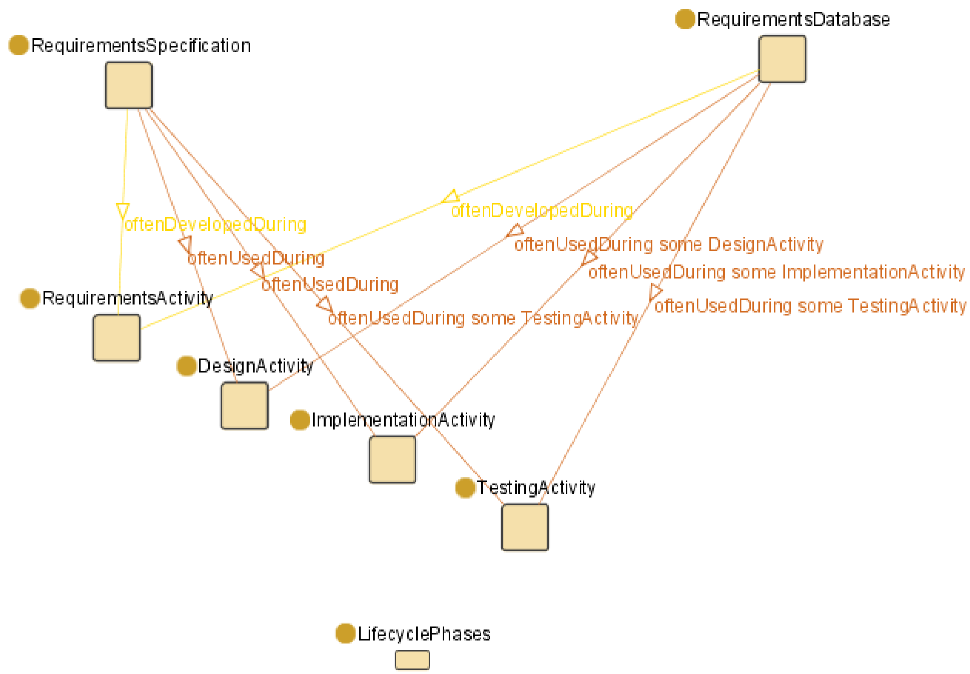


Figure 5. Properties Assigned to RequirementsSpecification and RequirementsDatabase Classes (developed using Jambalaya tab in Protégé)[37]

Objective Architecture Taxonomy

This taxonomy represents the decomposition of the common architecture for Navy combat systems, and was built directly from the already existing Surface Combat System Top-Level Objective Architecture. The taxonomy enables the repository system to correlate artifacts that have similar relationships based on commonality within the architecture to suggest them as possible items for retrieval.

System-SubSystem Ontologies

Here we provide one example (Figure 6) of how systems/subsystems and their interfaces can be captured as an ontology to complement the repository framework. Our recommendation is that ontologies be developed to capture each of the systems contained in the repository. As mentioned previously, the system/subsystem taxonomies would be used to verify the entries for the *System* and *Subsystem* elements in the metadata in order to assign artifacts to classes and subclasses (as individuals) within the ontology. Once these are assigned, the repository application could derive interface and other relationships from the ontology.

Each piece of the repository framework enhances the search capabilities in different ways. The basic metadata in the XML schemas provide search criteria for finding components of interest in the repository as well as specific information about the artifacts to determine if they are appropriate for retrieval. OWL taxonomies and ontologies enable identification of functionality and associated resources that may be beneficial to users. In

---

[37] We used Stanford's Protégé-OWL tool to develop all taxonomies and ontologies. This is an open source, free ontology editor, available online at http://protege.stanford.edu/.

short, the metadata is evaluated to enable retrieval decisions, the software behavior representations enable searches based on functionality, and the ontologies point the user to helpful artifacts that they may not have initially considered.

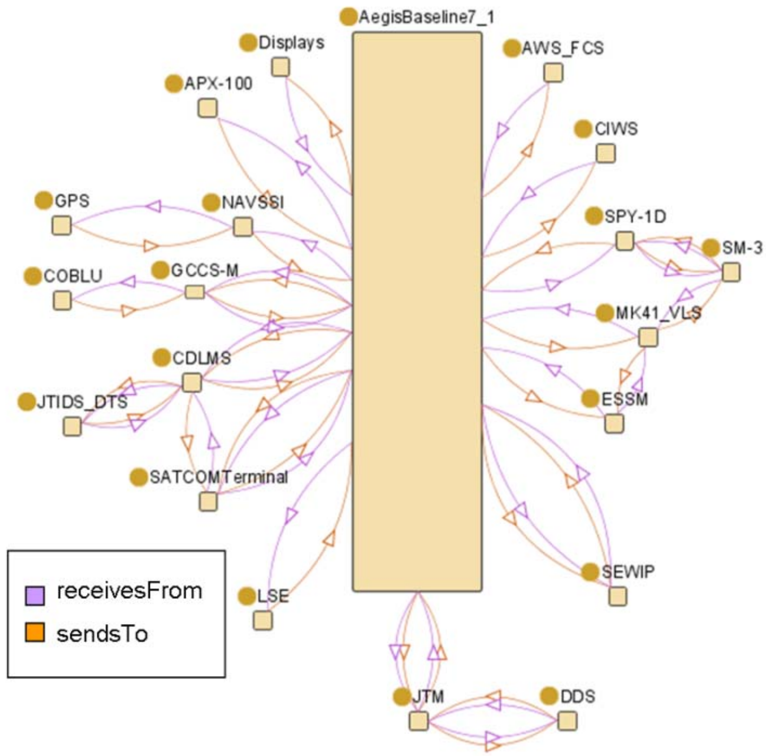


Figure 6. System Ontology Example (Aegis) (Jambalaya Graphic in Protégé)

Visualization

The tool must provide visualization techniques that will exploit the contextual information captured in the ontology and enable guided search capabilities. We suggest that multiple visualization tools be made available so that users can view the repository context and contents in a familiar setting.

One example of the type of tool that will be supported by the framework is a fish-eye graph. Fish-eye graphs display objects of interest to users, along with the relationships the objects have with other items. As the relationships interesting to users are explored, the graph highlights the item and brings it to the front of the display. Users can then weed out uninteresting items by removing from view the relationships that are not important. The results are a single or small grouping of items that users have found interesting with supporting information available by the click of a mouse.

Since domain information is captured in the repository framework, other architectural views may be used to present the repository contents in a display that is familiar to the user. For example, various views from the Department of Defense Architectural Framework (DoDAF) may be used as the backdrop to artifact nodes in order to provide a frame of reference that is comfortable for users accustomed to the associated graphics. Artifact groupings may also be represented as UML diagrams to allow for easy interpretation by software developers.

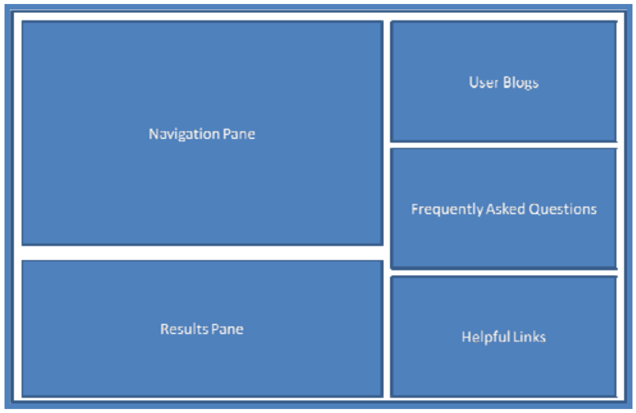


Figure 7. Screen Template

Use Case Demonstration

Here we describe, omitting some details, a use case scenario for the new tool to bring to light how each of the major features of the repository framework comes into play. This represents one possible scenario of interaction between a user and the repository system. The system reactions are described in terms of the individual windows on the screen that will update based on human-driven events. These windows include the Navigation Pane, Results Pane, User Blogs, Frequently Asked Questions, and Helpful Links, as shown in Figure 7.

In this scenario, the users need to build a replacement for a particular subsystem of the Aegis combat system, generically termed “Submodule B.” They consult the SHARE repository to find artifacts that will help in the development of the requirements for the new subsystem. Potentially, there are requirements for an existing system that can be reused. There may also be additional artifacts to be discovered that may be helpful in the requirements development process. When the tool is first initiated, an initial (home) screen is provided to the user (Figure 8).

Figure 8. SHARE Home Screen

The initial navigation pane includes a short welcome and some guidance for how to get started. There is a list of initial questions that helps the system orient the view specifically for the user. These questions are intended to provide a starting point for the guided search by "anchoring" the initial search results appropriately within the ontologies based on relevant information provided by answering simple questions. If the user does not care to answer the questions, the guided search can begin immediately by pressing the SEARCH button. Tips are provided as popup windows that can be opened by left-clicking the hyperlink if the user is not sure how to get going, or if he is unsure about how to answer specific questions. The most often retrieved artifacts are presented as the default "results" in the result pane. Blogs that pertain to the repository system as a whole are presented (those with the most activity listed first) in the user blogs pane. The most often asked questions are provided, with a link to additional questions, in the FAQ pane. The questions displayed pertain to the entire repository. Finally, links to general information about the repository, related repositories identified by stakeholders, and other locations relevant to the content of the repository (Navy Open Architecture) are displayed in the helpful links pane.

The available answers in the drop down menus for each of the initial questions are dependent on the ontologies represented in the repository framework. As the questions are answered, each of the panes is updated to reflect each choice. A priority scheme is applied after each user selection, and items ranked highest according to the scheme may appear in the display if applicable. After a user has answered all questions in this scenario, the individual panes may appear as in Figure 9. The chosen answers appear in the drop down windows of the navigation pane. All other panes have been updated to reflect the choices made by the user to this point. The results, user blogs, FAQs, and helpful links panes all show reprioritized items that are associated with the requirements activity, the surface domain, the Aegis system, and Submodule B, where applicable. Additionally, items that are not specifically tagged to each of these selections may be listed based on the graphical distance captured through the use of the ontology relations.

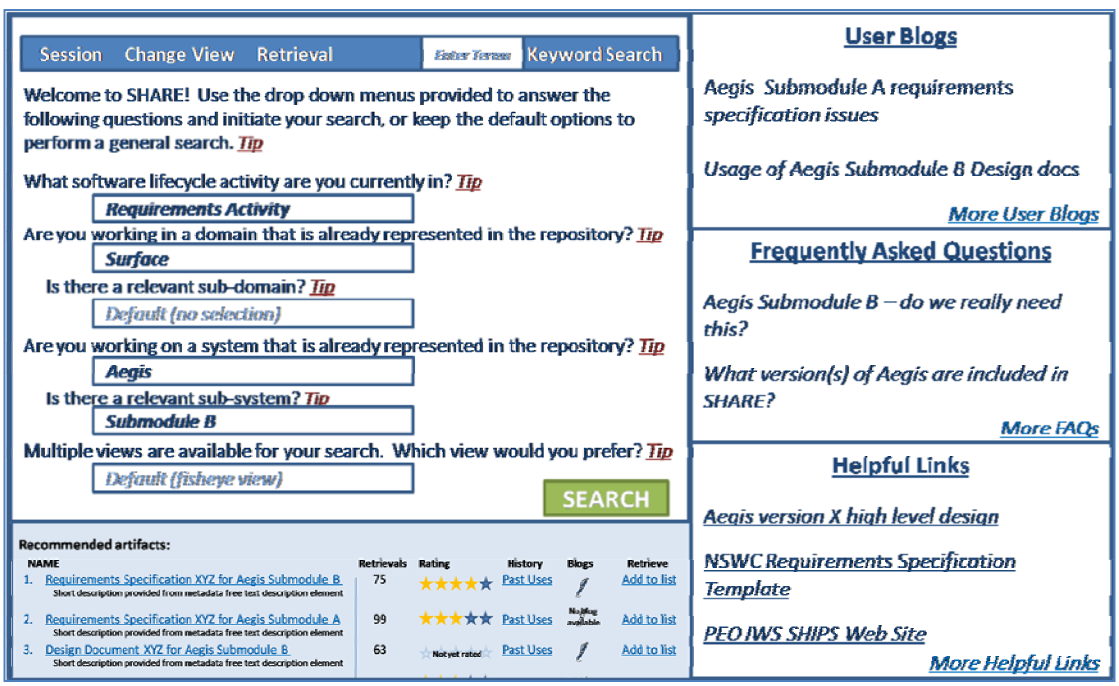


Figure 9. Home Screen After all Questions Answered

Once the user has selected the appropriate answers to each question, the SEARCH button is pressed. Since the default view was chosen, the navigation pane switches to the fisheye view of the repository contents (Figure 10). The fisheye view presents the artifacts of the repository as a graph that centers on the most relevant items. The positioning and size of the artifacts in the fisheye graph are determined by the prioritization scheme applied after certain user actions. The connectors between artifacts are the relations captured in the ontologies. Each of the types of relations is listed in an interactive menu that allows the user to turn the relations off and on depending on interest.

Additional features of the fisheye navigation include:

- Pop-up windows for artifacts and relations—Activated by mouse-scrolling actions, the artifact pop-up window contains a subset of the metadata for the artifact (see
-). The relation pop-up window describes how the two connected artifacts are related.
- Artifact detail page—Left-clicking on an artifact node opens an artifact detail page, which provides more of the artifact metadata.
- Action window—Right-clicking on an artifact node opens a drop-down action window that allows the user to open more information about the artifact or add it to the retrieval listing.

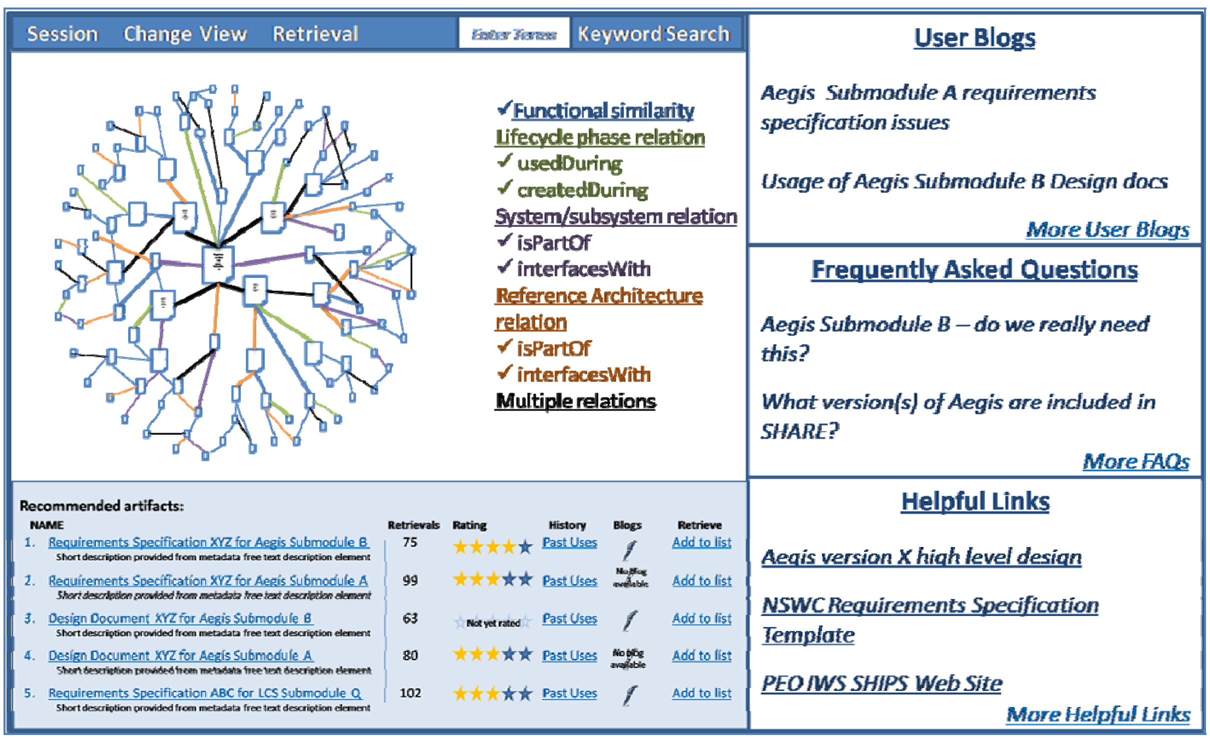


Figure 10. Initial Search Return—Fisheye View

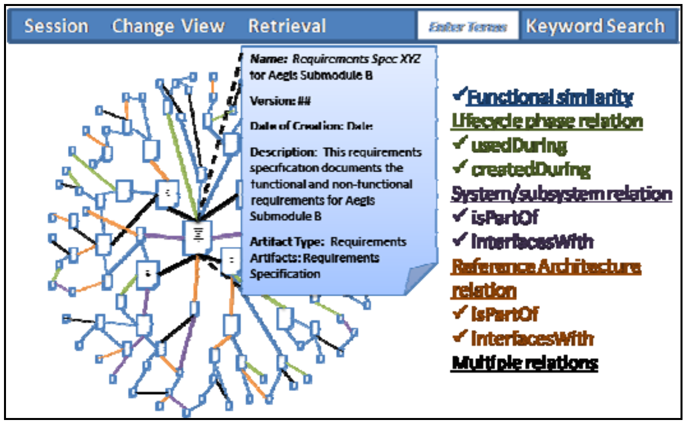


Figure 11. Artifact Pop-Up Window

Users can also add an item to their retrieval listing by selecting the option from the results pane. Each time the user selects an item for retrieval, the prioritization scheme is reapplied and each of the panes is updated to reflect the highest priority items.

After choosing all interesting items, the user selects Retrieval->Retrieve Items from the navigation pane drop down menu. A separate window is provided that contains the user's choices to this point, as shown in Figure 12. Select metadata is presented in addition to the names of artifacts to help the user review the list. The user can modify the list by deleting anything deemed irrelevant at this point. When the user is satisfied that the list of desired items is complete, the user presses the RETRIEVE button.

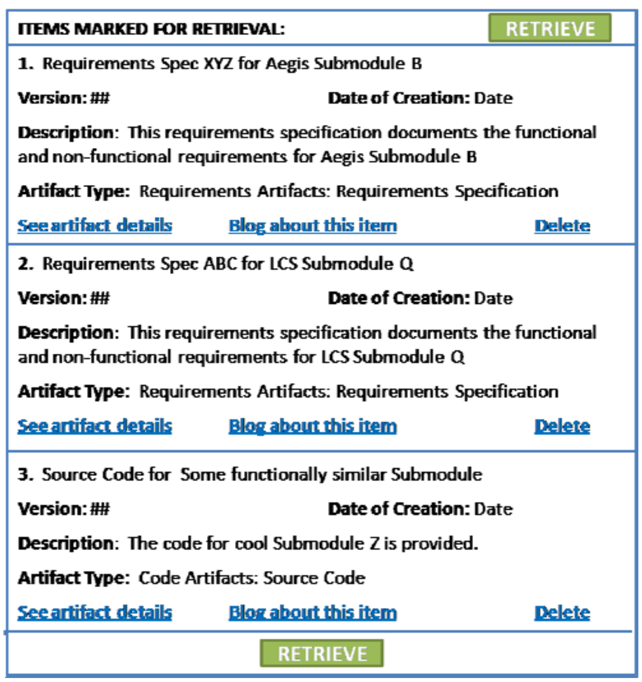


Figure 12. Retrieval List

Information is provided to assist the user in requesting and retrieving the items. Any items available for immediate download are made available using appropriate hyperlinks. Items that require request/approval processes are also enabled through a step-by-step automated process. This extra step is required for repositories that have security and intellectual property limitations, such as the SHARE repository. If the Ontology Based Software Reuse Repository System is developed for an open source repository, it should be possible to simply provide links to the artifacts themselves. In this case, it would be desirable to replace the “Retrieve” column in the results pane with a “Download” column, and to add a menu option for downloading the artifact when the user right-clicks on an item in the navigation pane.

Related Work

Sugumaran and Storey propose the use of domain knowledge in repositories to aid in the natural language processing of queries for component retrieval (Sugumaran & Storey, 2003). In their prototype system, the ontology captures synonyms and relations between objects in the domain. The system enables the user to enter queries using natural language, and the ontology enables more coverage in the returned items by including items that contain the relations captured in the ontology.

This work is closely related to the system proposed herein, since they both address some limitations of traditional keyword and faceted classification-based searches. However, there are several key differences. First, the Sugumaran et al. ontology is limited to a single view of the typical objects and terms within a specific application domain; whereas our approach includes multiple views as described. Second, the visualization enabled by the ontology is vastly different. In the Sugumaran et al. approach, syntactic analysis is conducted on a query entered through a free text interface, resulting in lists of processes, actions, and matching or related components that the user can then choose to view in more detail. Our approach enables the user to navigate the repository contents in a more interactive way. Finally, the use of the ontology to provide a lexicon for matching terms is extended in our approach since artifacts in the repository are captured as individual items in the ontology classes. This approach provides wider use of the ontology in representation of repository contents and user interaction.

Yao and Etzkom also focus on the use of ontologies for enhancing search retrieval based on natural language queries. They extend the idea by suggesting the use of Semantic Web technologies such as RDFS/DAML+OIL to apply the methodology to the World Wide Web as a large software repository (Yao & Etzkorn, 2004).

Summary and Future Work

In this paper, we have presented an ontology-based approach for the development of a software reuse repository. Our claim is that the knowledge captured by the ontologies enables new ways of discovering desired software artifacts based on computer aided navigation rather than the more traditional query/response discovery. We described the repository framework that provides the contextual depth to support such navigation and demonstrated the approach using a use case.

Throughout the project we have identified several areas for future work. First, we recognize a need to investigate the automated population of artifact metadata. A significant challenge will be the generation of XML metadata from existing reusable resources and help for users in describing future submissions to the repository. Current approaches for automatic generation of metadata from content libraries should be explored for potential application to the ontology-based repository and more specifically for the SHARE metadata context. We suspect that a combination of techniques will be useful.

Second, providing a practical software behavior representation remains a challenging area for continued exploration. The current work provides an identification of principal functionality of an artifact through the CSFL and possible description of operations and input/output messages from a service perspective using WSDL. The work is admittedly preliminary. Research into related areas of Semantic Web Services, Business Process Execution Language, and others continues to hold promise for this aspect of the repository framework.

Third, additional user views may be desirable other than those we have described here. Some investigation into the feasibility of translating the ontological information between various model types is warranted.

Finally, in addition to the Navy's CSFL, similar lists have been developed for operational activities (COAL) and for information elements (CIEL). It would be interesting to express these taxonomies in OWL, as was done with CSFL, and then to create interrelationships across the classes, for example, to determine what information elements are generally employed in performing certain system functions, or what information elements are generally produced by performing certain system functions. Further exploration with subject matter experts (SMEs) is needed to determine potential benefit from such approaches.

Acknowledgments

This research would not have been possible without the financial and moral support of Mr. Nick Guertin, Director of the Navy Future Combat Systems Open Architecture Program. I would also like to sincerely thank Dr. Mikhail Auguston, who has been my mentor and guide throughout this research.

References

Guo, J., & Luqi. (2000). A survey of software reuse repositories. In *Proceedings of the Seventh IEEE International Conference and Workshop on the Engineering of Computer Based Systems*, 2000 (ECBS 2000) (pp. 92-100).

Hassan, A. E. (2008). The road ahead for mining software repositories. In *Proceedings of the 2008 IEEE Frontiers of Software Maintenance* (pp. 48-57). Beijing, China.

Johnson, J., & Blais, C. (2008). *Software hardware asset reuse enterprise (SHARE) framework: related work and development plan*. Monterey, CA: Naval Postgraduate School.

Johnson, J., & Blais, C. (2008). *Software hardware asset reuse enterprise (SHARE) repository framework final report: Component specification and ontology*. Monterey, CA: Naval Postgraduate School.

Kruchten, P. (1995). The 4+1 view model of architecture. *IEEE Software*, *12*(6), 42-50.

Object Management Group. (2005). *Reusable asset specification* (Vers. 2.2).

Shiva, S., & Shala, L. (2007). Software reuse: Research and practice. In *Proceedings of the Fourth International Conference on Information Technology, 2007 (ITNG '07)* (pp. 603-609).

Sugumaran, V., & Storey, V. C. (2003). A semantic-based approach to component retrieval. *The DATA BASE for Advances in Information Systems*, *34*(3), 8-24.

Yao, H., & Etzkorn, L. (2004). Towards a semantic-based approach for software reusable component classification and retrieval. In *Proceedings of the ACM 42nd Southeast Conference (ACMSE '04)* (pp. 110-115). Hunstville, AL.

Enabling Software Acquisition Improvement: Government and Industry Software Development Team Acquisition Model

Joe Heil—Joe W. Heil has worked as a Software Engineer for the Naval Surface Warfare Center Dahlgren Division (NSWCDD) for over 23 years. The majority of Mr. Heil's career was spent as the Software Integrated Product Team Lead for the Tomahawk Weapon Control System (TTWCS). As the TTWCS SW IPT lead, he was responsible for defining the software development processes and coordinating the software development efforts for the TTWCS integrated Industry and Government software development team. Mr. Heil is currently the Senior Software Engineer for the NSWCDD Strategic and Weapons Control Systems Department (K) and is focused on improving software engineering, development, and acquisition.

Abstract

The growth, complexity, and reliance on software (SW) as part of the Department of Defense and Navy (DoD/Navy) warfare systems is continuing to increase. This increase in SW complexity and reliance has been accompanied by an increase in well documented SW intensive system acquisition cost, schedule and technical performance failures. The DoD/Navy is not consistently performing as a smart buyer of software intensive systems. The government and private industry have not been successful in applying the latest software methodologies and technologies nor consistently providing high quality and reliable systems that are delivered on schedule and within budget. The typical acquisition approach utilized over the past several decades of relying primarily on private industry for architecting, designing and implementing SW intensive systems has resulted in the loss of government in-house applied SW expertise necessary to achieve truly open architected systems and systems-of-systems.

The key enablers for improving SW intensive system acquisition are the reconstitution and utilization of government in-house software subject matter experts (SMES) that can lead and work with industry SW engineers as part of an integrated SW Development Team. Figure 1 summarizes the current state and desired future state trends.

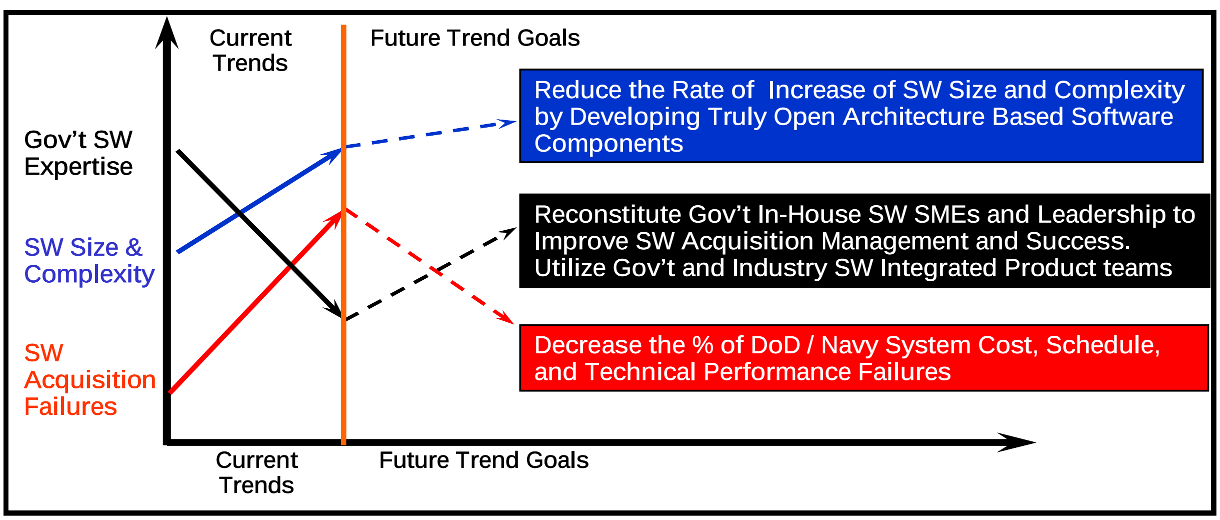


Figure 1. SW Acquisition Trends

Current State: SW Technical Challenges

There are numerous technical challenges associated with the growth and reliance on software within the DoD/Navy's mission critical warfare systems such as:

- Designing and implementing truly Open Architected systems that fully meet the goals of standardized interfaces, scalability, reliability, portability, modularity and reusability; and thereby lead to higher system quality while also reducing cost and schedule.
- Assessing, successfully utilizing, and rapidly integrating the most advanced software technologies and methodologies such as Model Driven Architectures, Service Oriented Architectures (SOA), multi-core parallel processing, automated code generation, cloud computing, next generation programming languages, and agile development processes.
- Integrating the mix of legacy SW components, new Commercial-Off-The-Shelf (COTS) SW components and DoD/Navy developed highly specialized and unique SW components to provide integrated net-centric systems composed of hundreds-of-millions (possibly billions) of lines of code that can execute as systems-of-systems and fully meet mission level objectives and Key Performance Parameters (KPPS).
- Achieving Information Assurance (IA) and protection against SW-based Cyber-Attacks while trying to maximize COTS utilization and Net-Centric communications.
- Maintaining government corporate knowledge of the system architecture, design and technology utilization as the responsibility for system and software development transitions among different private industry organizations during the program lifecycle.

In order to address the SW engineering and development technical challenges listed above, as well as many not listed here, it is imperative that the government maintain the applied software technical experts that can serve as both leaders and team-mates with peer industry software engineers.

Current State: Acquisition Approach

Figures 2 and 3 provide high level models with a rough indication of the relative involvement of government versus industry technical experts and the typical acquisition approach utilized for SW Intensive system acquisition and development. Government engineers are primarily used during the initial system concept, system level requirement phase, and system validation phase of the acquisition process. In the initial stage of system acquisition, the government system engineers define the capability need and the associated highest level system requirements and key performance parameters (KPPs). During the initial phase, government system engineers may work with multiple Industry organizations to perform Technical Assessment of Alternatives (AoA) where industry provide prototypes or advanced technology demonstrations (often proprietary) advertised to fully meet the system capability needs and can be developed in a timely and cost effective manner.

The government then relies almost entirely on Industry technical experts for the detailed system and software architecting, designing, coding and software level integration and test. The System's SW requirements, design, code and test artifacts are developed almost entirely by Industry. Government insight into the detailed software architecture and design is primarily via the utilization of milestone reviews (System Requirement Reviews (SRRs), Preliminary Design Reviews (PDRs), etc.). The government then takes the lead for System Integration, Testing and Certification. And as described in the next section, this acquisition approach fails more often than not with regards to cost, schedule and performance.

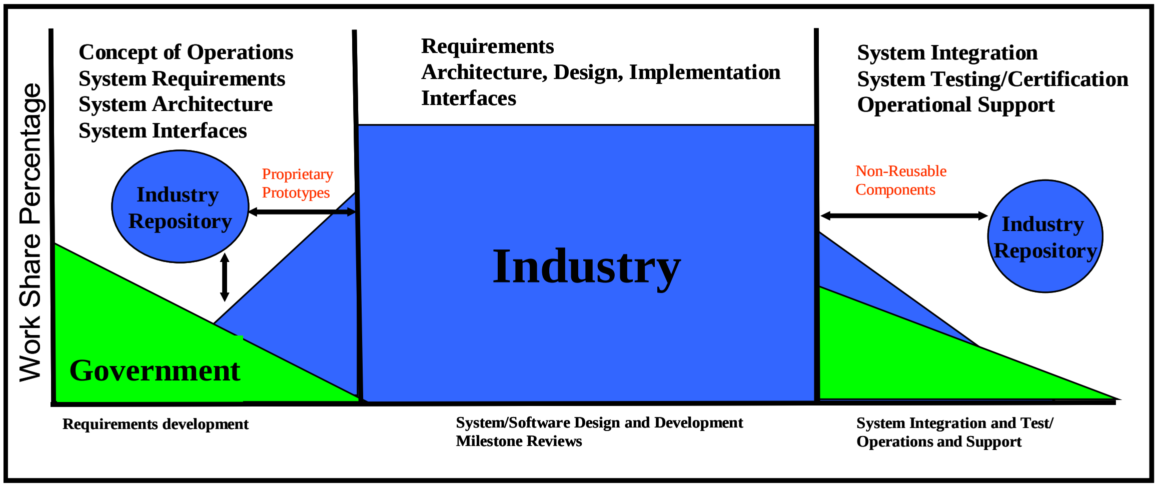


Figure 2. Current Acquisition Approach

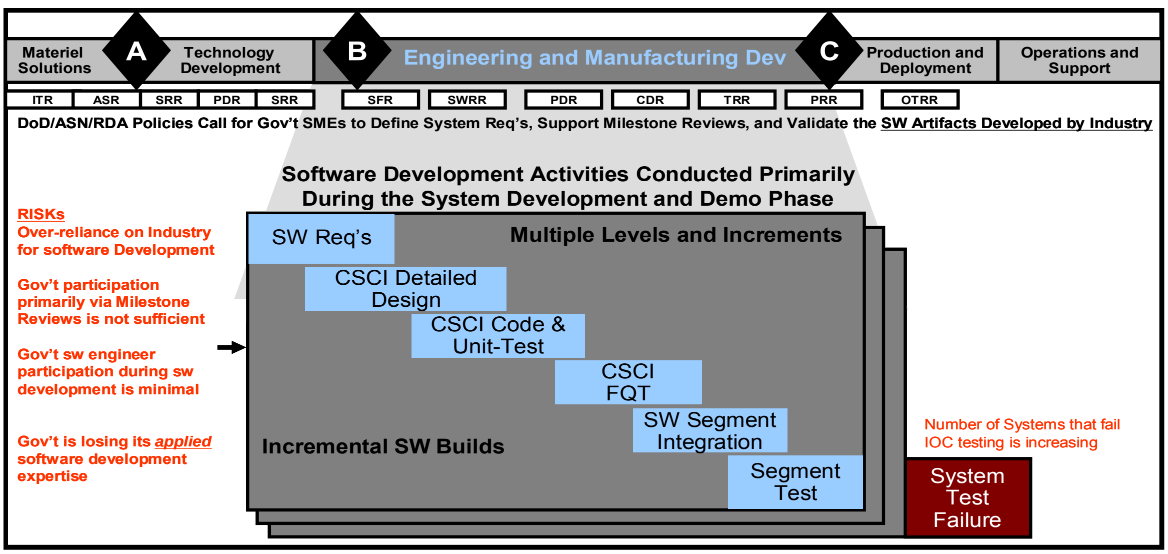


Figure 3. Current Acquisition Approach

Current State: Results

The increase in DoD/Navy SW intensive warfare system cost, schedule, and technical performance failures over the past 20 years are well documented in numerous reports and studies from organizations such as the Defense Science Board (DSB), the Government Accountability Office (GAO), Crosstalk, and Assistant Secretary of the Navy Research Development and Acquisition (ASN/RDA). Figure 4 summarizes some of the key studies and their cost, schedule, and quality metrics.

The November 2002 Report of the DSB Task Force on Defense Software reported:

- Only 16% of programs are completed on schedule and within budget
- Up to 31% of programs are cancelled and the remaining 53% have cost growth greater than 89%
- The average final product includes only 61% of the original intended features.

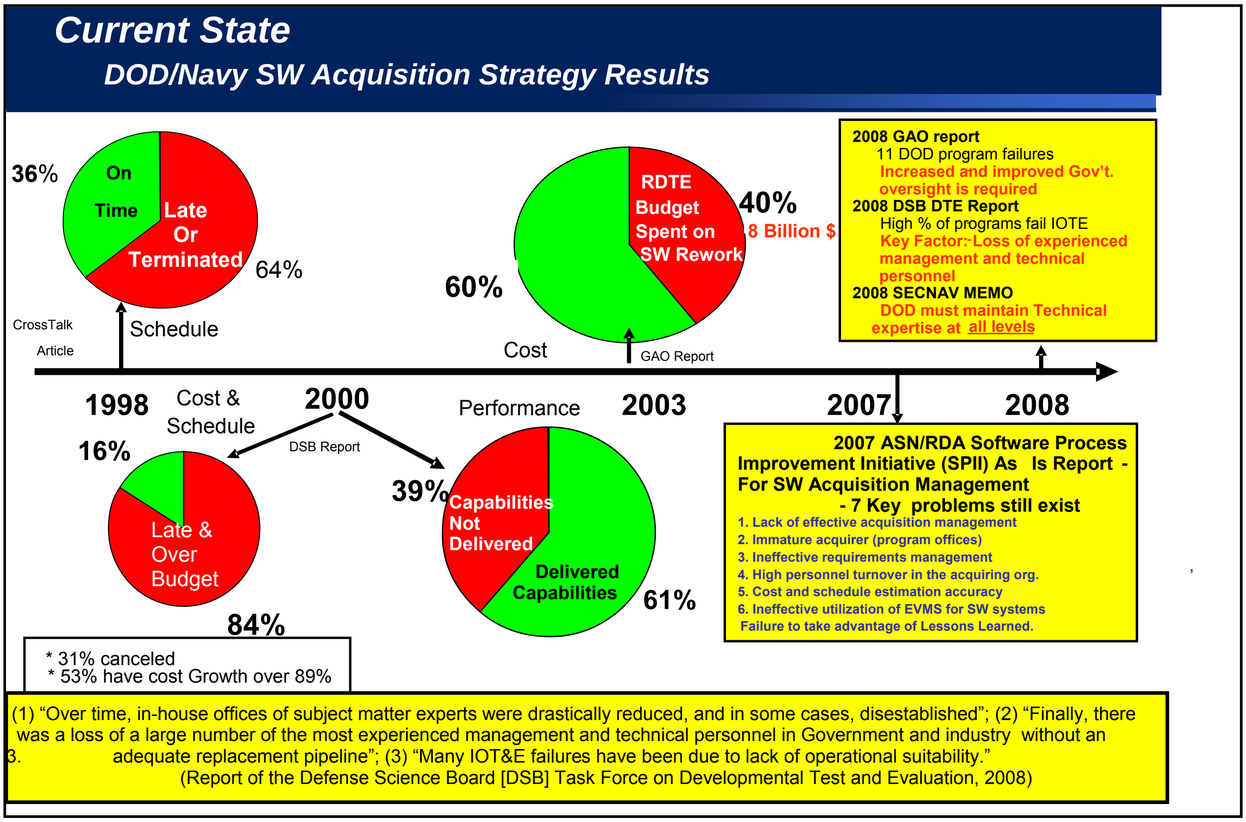


Figure 4. DoD Software System Acquisition Report Findings

In 2004, the GAO reported that the DoD spent 40% of its software development budget reworking software because of quality related issues (GAO, 2004). In 2008 the DSB reported that the majority of DoD weapons systems are failing Initial Operational Testing. In 2008, the ASN/RDA SPII SAM focus team published a report that documented the following critical problems that apply to the vast majority of DoD/Navy SW program acquisition offices:

- Lack of effective management.
- Immature acquirer (program offices).
- Ineffective requirements management.
- High personnel turnover.
- Unrealistic cost and schedule estimates.
- Ineffective utilization of Earned Value Management Systems (EVMS) for SW.
- Failure to utilize of lessons learned.

In 2009, Senator Carl Levin reported that since 2006 nearly half of the DoD's largest acquisition programs have exceeded Nun-McCurdy, and that 95 major defense programs have had their acquisition costs grow by an average of 26% and have experienced an average schedule delay of almost 2 years.

The DoD has lost the ability and expertise required to consistently successfully team with industry to acquire SW intensive weapon systems on time and within budget.

Current State: The Devil Is in the Details

Although software has evolved into one of the most complex and critical elements of mission critical systems, the typical DoD/Navy acquisition strategy tends to treat the software components as black boxes with the internal software architecture and design development (and understanding) left almost entirely in the hands of private industry software engineers. As shown in Figure 5, a typical SW system may include:

- Hundreds to thousands of system level requirements,
- Thousands to tens-of thousands software level requirements,
- Tens to hundreds of external system interfaces,
- Hundreds to tens of thousands computer software components (CSCs),
- Thousands-to tens of thousands internal software interfaces and interactions,
- Millions to hundreds of millions of logic threads,
- Millions to hundreds of millions of source lines of code (SLOC), and
- Billions of software characters.

And note that all it takes is for single erroneous character within the millions of lines of SW to cause a total system failure.

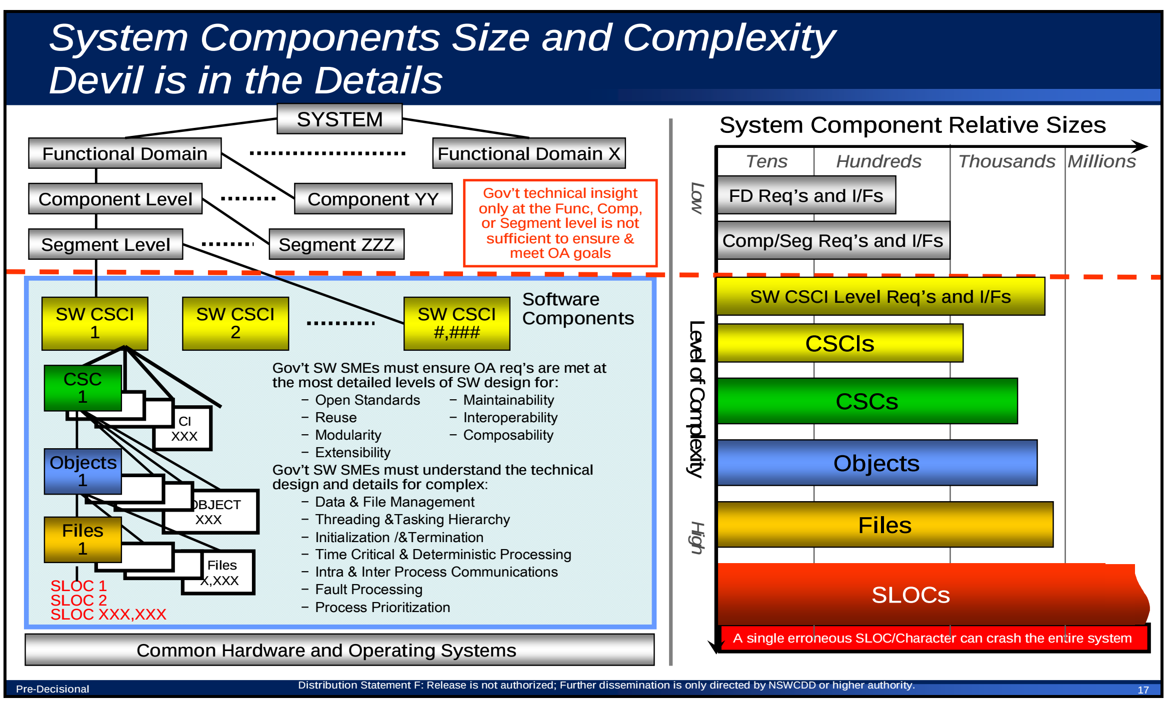


Figure 5. System and Software Complexity

One of the most significant challenges facing the DoD/Navy's complex software intensive system acquisition is the rapid rate of change associated with software technologies, methodologies, processes, processors, and tools. In order for program office leadership to successfully maintain existing software systems, and acquire new software systems, it is imperative that they have access to in-house technical experts that have applied expertise with both the older software environments and the latest cutting edge software technologies and environments.

In addition to having experience working with the latest software technologies, Government in-house software engineers must also be able to apply these technologies to the unique, complex, and challenging context of Navy system functional requirements (e.g., time critical processing, real-time processing, numerous external and internal hardware and software interfaces, complex algorithms, safety critical, and nuclear critical). The current typical acquisition approach limits the government's technical understanding to a few pages of high level system and software architecture diagrams, and understanding and "controlling" the interfaces between the software components only at the highest level of system abstraction, the Government is not able to maintain corporate technical expertise required to successfully acquire software intensive systems.

The allocation of all the software architecture, design, code, and test responsibility to private industry is causing the government to lose the applied SW development experience and expertise to consistently *successfully* perform all of the following critical software system intensive acquisition activities:

- Maintain awareness and expertise in modern SW technologies and methodologies necessary to understand when/if/how these new technologies should be utilized;
- Assess industry's technical approaches, and also provide government developed technical approaches;
- Evaluate industry's technical cost and schedule estimates;
- Ensure Open Architecture (OA) design and verify that the OA design is actually implemented in the SW code;
- Fully understand the technical, cost, and schedule impacts of requirement changes
- Define and manage SW EVMS;
- Define and utilize SW metrics-based control processes; and
- Identify and manage software risks.

Architecting and designing only at the higher levels of system abstraction (i.e., segment, component, and functional domain) is not sufficient for the government to maintain applied SW expertise. The amount of required expertise, experience, and effort required to successfully architect and design the software components increases at each lower level of system decomposition. An *applied* in-depth understanding of software technologies and methodologies is necessary to architect, design, and implement the software components at the CSCI level and below. The government must understand the sub-component SW elements to successfully address the following technical challenges:

- Asynchronous real-time event processing,
- Time Critical (milli/micro/nano second) accuracy and timing,

- Safety Critical requirements,
- Anti-Tampering and Information Assurance protection,
- Data Security/Classification protection and segregation,
- 24/7 system reliability and system accessibility, and
- Protection against Cyber-Attacks.

The typical acquisition approach of Milestone reviews provide too little insight and occur too late in the acquisition schedule as the damage has already been done. Many "design" reviews now focus more on compliance to SW processes versus actually providing an in-depth review of the SW architecture/design/code. Even if private industry provides a detailed and thorough presentation of their software architecture and design at the milestone reviews, the government typically, except for a few rare cases, lacks the applied in-house software experience and expertise to ensure the software components meet all OA objectives including modularity, scalability, reliability, maintainability, and quality; and ensure the implementation artifacts (code) and design artifacts remain consistent with each other. If the government identifies any significant technical software architecture or design issues during the milestone review, the contractor typically responds with such severe cost and schedule impacts that often the only option left is to trade-off planned new capabilities for significant architecture and design corrections.

Some software intensive programs utilize government in-house software engineers to participate with industry during software development. This participation is typically via peer-review during design and code activities. This approach assumes that Government software engineers will be willing to review other engineer's work rather than being responsible for designing and coding software components themselves. The government cannot attract the best talent, nor sustain highly motivated and high quality software SMEs by limiting their tasking to looking-over-the-shoulders of industry software engineers. Government SW engineers must have hands-on development responsibility in order to maintain expertise.

Future State: SW Acquisition Goals

The primary goal is to improve the DoD/Navy's ability to consistently deliver high quality SW intensive weapon systems that fully meet the warfighters' needs, while also delivering these systems in both a timely and cost effective manner.

A second major goal, as shown in Figure 6, is to achieve truly Open Architected systems and move from stove-pipe, proprietary, redundant and non-common systems towards product line multi-platform non-proprietary common reusable systems and software components. Achieving truly OA systems will improve system quality, promote competition and innovation, and reduce cost and schedule.

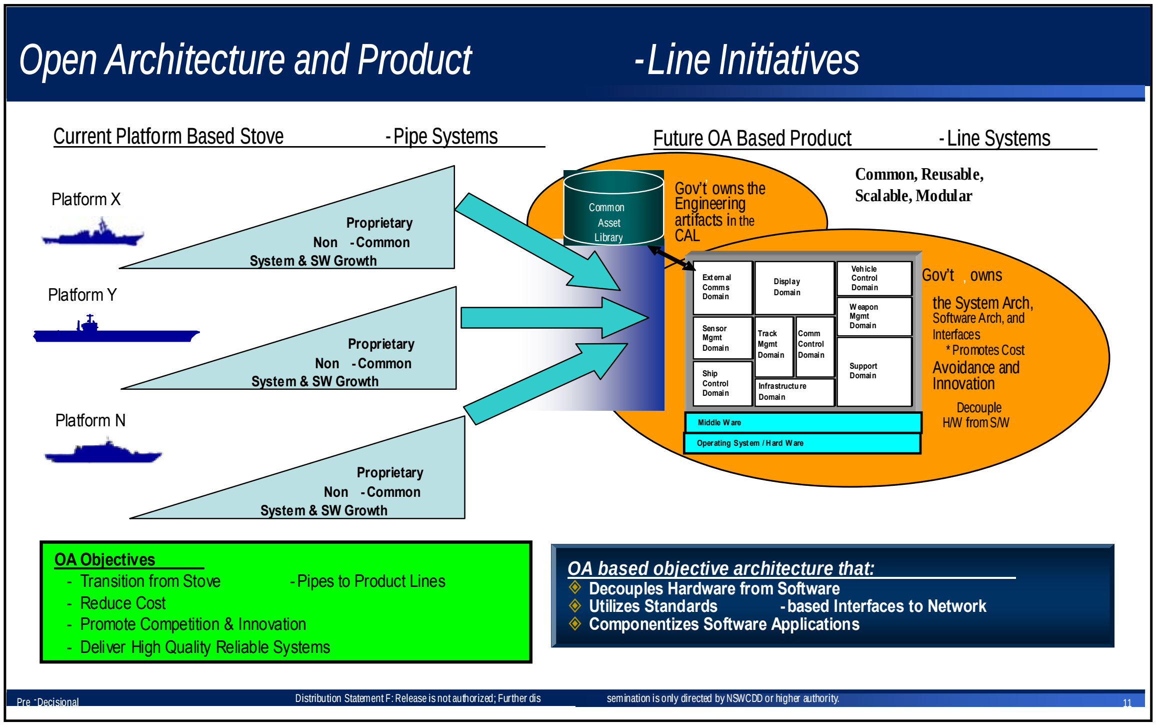


Figure 6. Open Architecture Goal

Future State: Team-Based SW Acquisition Approach

In order to achieve these major goals, the DoD/Navy must reconstitute and maintain a sufficient level of SW expertise with the applied experience required to team with Industry and address the numerous SW development technical challenges. Figures 7 and 8 comprise a high-level model of an alternative SW acquisition approach that enables the government to maintain applied SW engineering expertise.

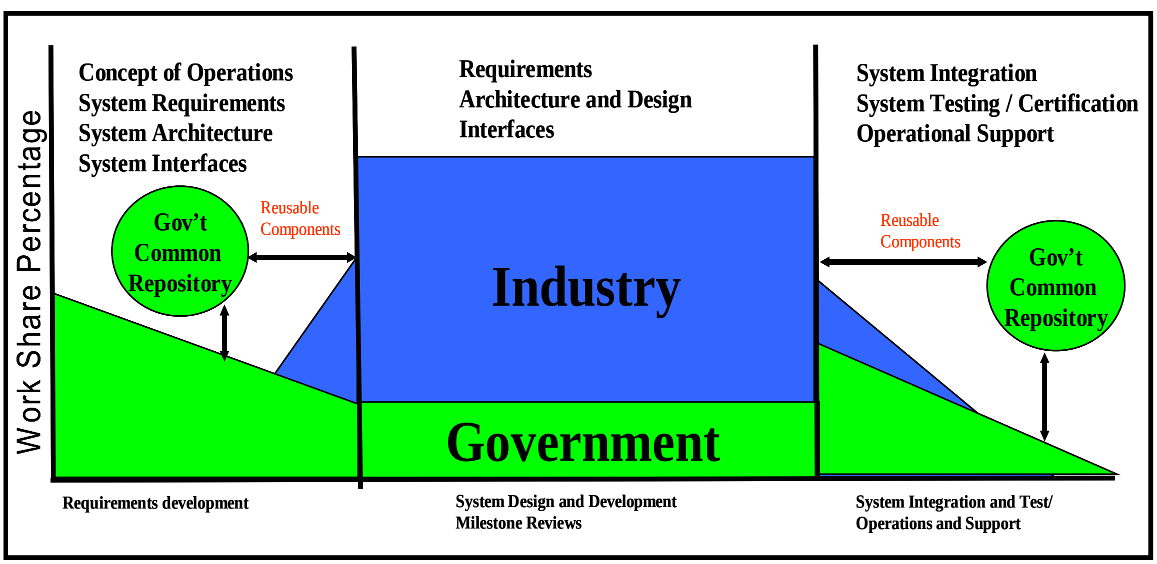


Figure 7. Alternative SW Acquisition Approach

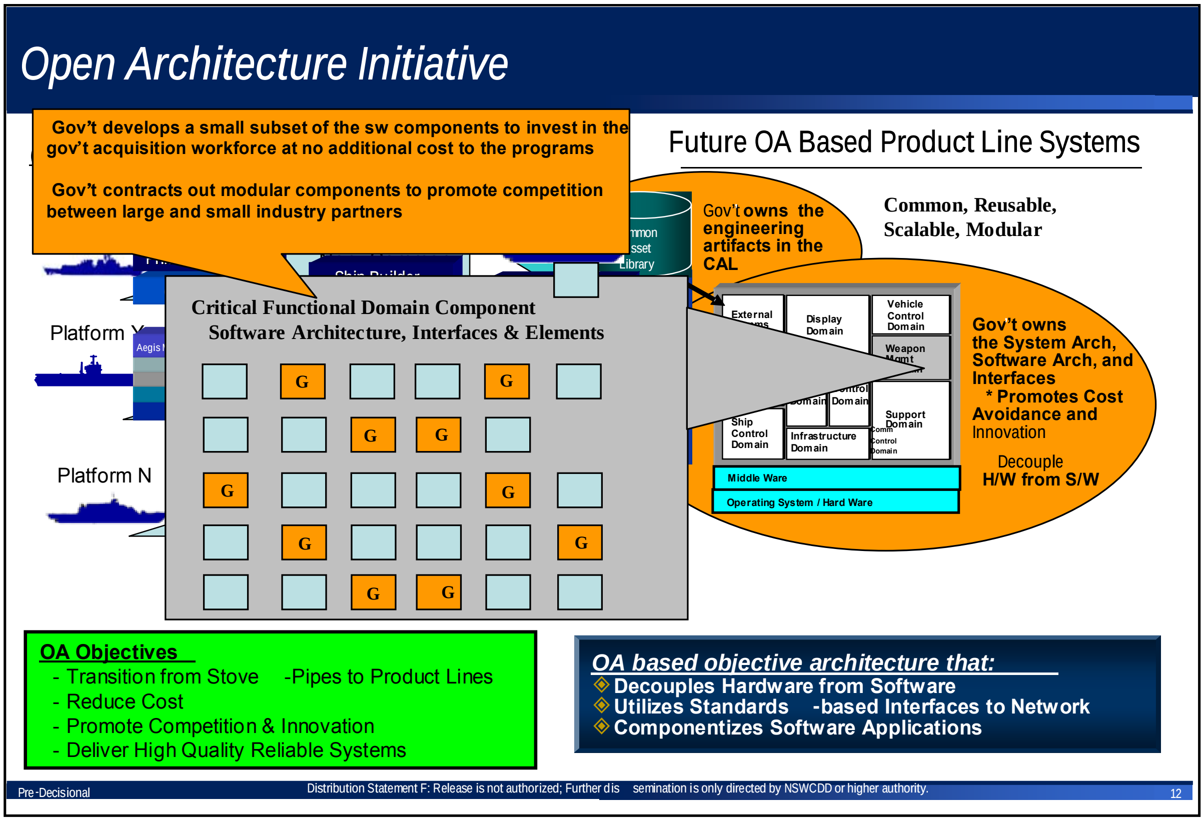


Figure 8. Government SW Development

The key differences between this SW acquisition approach and the typical approach is that the government SW engineers are responsible for developing and delivering a subset of the mission critical tactical system and software components. Government in-house SW engineers are responsible for developing and delivering the associated technical artifacts for their SW components, including requirements specifications, architecture and design documents, code, and test procedures. Note that Industry will still develop the vast majority of the SW components and artifacts.

The government SW engineers are also responsible for providing the critical management products as well including development process documents, metrics, schedules, cost/schedule progress (EVMS), interdependencies, and risks. This is required to develop and maintain in-house SW SMEs with the applied experience required to be able to successfully architect, design, and manage (accurately estimate and track cost, schedule, and risk) the software development effort at all levels of software intensive system decomposition (Functional Domain, Component, Segment, CSCI, and down to the CSCI sub-component Object and Class level).

The government SW engineers are given the opportunity to provide SW prototypes and advanced technology and methodology approaches during the pre-milestone A and B acquisition phases.

The SW artifacts (requirement specs, design documents, code, etc.) are developed by Integrated Government and Industry SW development teams that utilize cross organizational design/code peer reviews to ensure high quality products and conformance to best-practices.

The government SW Development engineers have the same expectations and requirements relative to cost, schedule and technical performance as their industry peers. System testing is performed by an independent government team with a separate management chain of command from the government SW team.

By assigning actual SW development responsibility to in-house engineers, the government can reconstitute and maintain the SW expertise pipe-line as shown in figure 9, and thereby develop the senior level SW expertise required to perform as peer level team-mates with Industry. This approach will provide in-house software SMEs that maintain applied experience and corporate knowledge (as the system evolves and as some of the component development is conducted by different industry organizations over time) with:

- Complex system and software functional requirements such as: Safety critical, Mission critical, Complex external and internal interfaces, Real-time processing Security sensitive data processing, and Complex algorithms;
- Latest software technologies and methodologies; and
- Applied open architecture (modular, scalable, reusable, maintainable, and reliable) software design and implementation at the sub-component level.

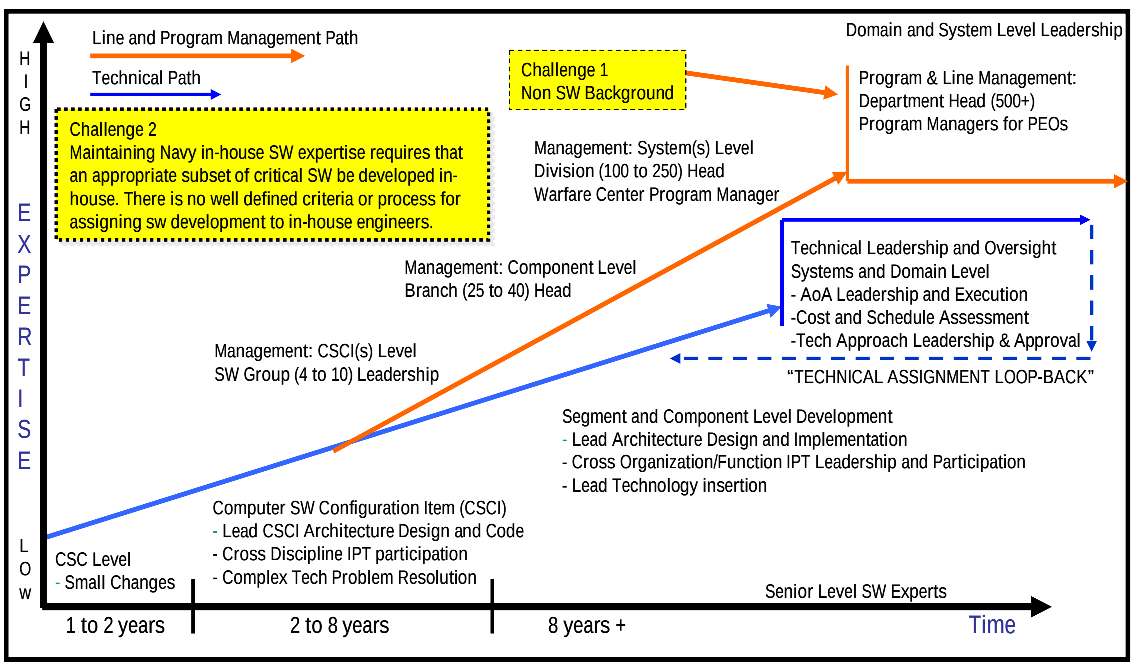


Figure 9. SW Expertise Pipeline Future State: Success Examples

The alternative government and industry SW development team acquisition approach described in the previous section has been successfully utilized for over 50 years by the Naval Surface Warfare Center Dahlgren Division (NSWCDD) for various strategic and tactical weapon and fire control missile and gun systems. For example, NSWCDD government software engineers have been, and still are, responsible for the architecting, designing, coding and testing of many of the most critical and complex (e.g., safety critical, real-time, highly complex algorithms, external and internal interface functionality) software components for programs such as the Tactical Tomahawk Weapon Control System (TTWCS).

The government SW engineers have successfully worked with private industry SW engineers as an integrated SW development team. The cost, schedule, and technical performance of these SW IPTs have been consistently exceptional over multiple decades as compared to the vast majority of complex weapon system programs that have relied primarily on industry for the SW development and have failed (per the references and metrics provided previously). The TTWCS SW IPT has been consistently successful in meeting the SW technical challenges and future state goals as previously described in this paper. Some specific examples are provided in the following paragraphs.

Over the past several decades, the TTWCS SW IPT has consistently successfully delivered software upgraded to incorporate and integrate the latest SW technologies; evolving from Mil-spec processors (ROLM 1666) to modern processors (HP744, X86); from mil-spec operating system (RMX/RDOS) to open system OS (LINUX); from first generation programming languages (Assembly, Fortran) to modern languages (Ada, Java, C, C++).

The SW IPT has successfully incorporated new SW development methodologies; transitioning from functional design to object-oriented design, from waterfall development to spiral/increment development; from human-only generated coding to graphic-user-interface and auto-code generation tools; from point-to-point interfaces to FDDI/ETHERNET H/W and SOA-based SW interfaces.

The TTWCS IP has achieved and demonstrated Open Architecture design and implementation. As shown in Figure 10, the TTWCS SW engineers utilized object-oriented design to achieve scalability and reusability with regards to the goal of easily interfacing with multiple platforms and their unique launching systems. The TTWCS System has been easily upgraded to support not just US Surface Ship Vertical Launching Systems, but also US Submarine and United Kingdom Royal Navy Submarine platforms. When the TTWC system was recently upgraded to interface with the SSGN platform, within less than a year the government SW engineers were able to define the SW req's, document the design modifications, implement and test the associated new Launcher Interface code changes. In addition, SW Components were reused from the TTWCS SW within the SSGN Launching System software which resulted in a faster than usual successful integration of the two systems.

The TTWCS system has successfully met interdependency deliveries with the Tomahawk missile segment upgrades and passed the vast majority of its Initial Operational Test Events.

The resulting quality of the TTWCS SW has been consistently high with the integrated SW meeting all KPPs and with SW quality consistently averaging little over 1 Defect/KSLOC. And more importantly, the TTWCS software developed by the government and industry team has performed exceptionally well in tactical operations.

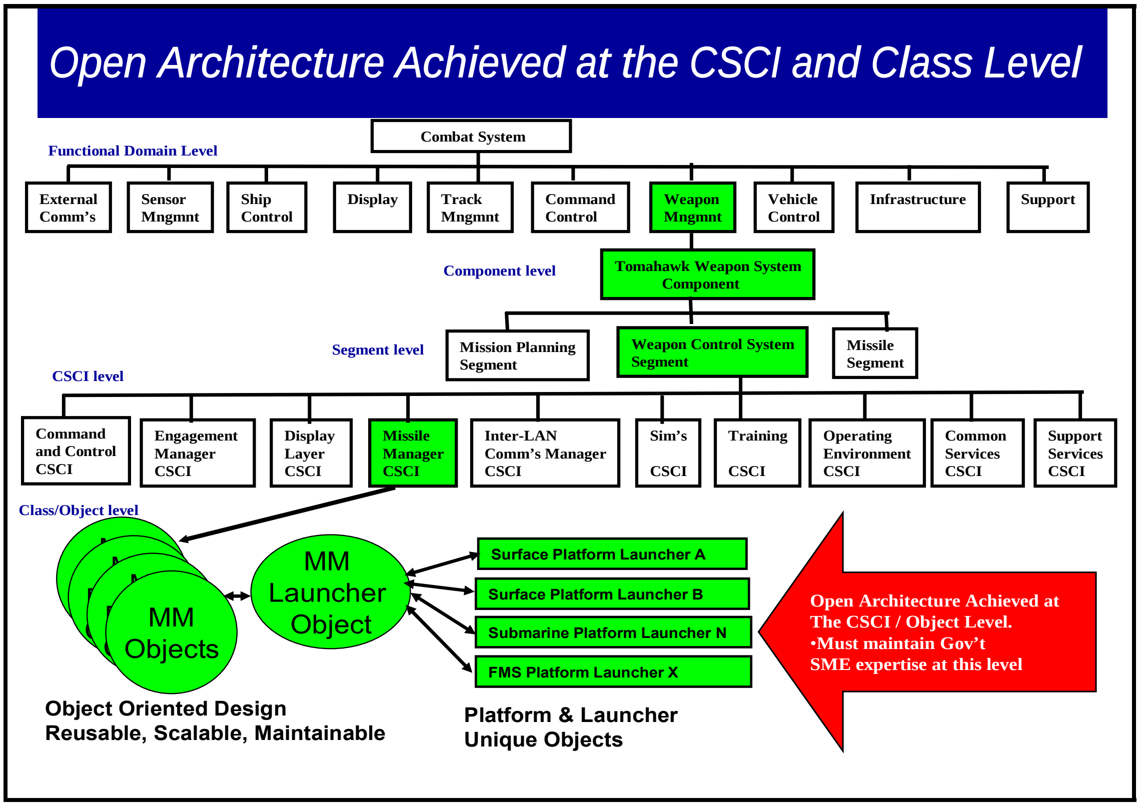


Figure 10. Open Architecture Design

In addition to working for the large (multi-million source lines of code), multi-year TTWCS development effort; the industry and government SW team approach has also been demonstrated to also work well for rapid development efforts. Government engineers have teamed with industry to utilize agile SW development methodology to successfully deliver the integrated sensor and weapon capabilities for marine/army vehicles such as Gunslinger, Full Spectrum Effects Platform (FSEP), and Wolfpack. This integrated agile development team has also been utilized for the Naval Expeditionary Overwatch (NEO) Intelligence, Surveillance, and Reconnaissance (ISR) systems. These rapid development efforts were lead by government engineers that quickly assessed and integrated multi-vendor hardware and software technologies to provide the deployed warfighters with much needed capabilities that met emergent mission critical needs.

Future State: Team-Based SW Development Benefits

As demonstrated by the consistent success of the TTWCS, SLBM, and Rapid Development weapon programs highlighted in the previous section, the government and industry SW development team model is not just a theory. There are many benefits to utilizing this SW acquisition approach. The senior level government SW engineers are capable of working with industry to address the significant SW challenges that include:

Designing and implementing truly Open Architected systems that fully meet the goals of standardized interfaces, scalability, reliability, portability, modularity and reusability; and thereby lead to higher system quality while also reducing cost and schedule.

- Successfully assessing and rapidly integrating the most advanced software technologies and methodologies into the SW development processes, environments and systems.
- Successfully integrating the complex mix of legacy SW components, new Commercial-Off-The-Shelf (COTS) SW and hardware components and DoD/Navy developed highly specialized SW components to provide integrated net-centric systems that can execute as systems-of-systems and fully meet mission level objectives and KPPs.
- Delivering systems with demonstrated Information Assurance (IA) and protection against SW-based Cyber-Attacks, while maximizing the utilization of COTS and Net-Centric architectures.

In addition to addressing the technical challenges above, the reconstitution of in-house SW expertise will also enable mitigation of the following key problems identified in the ASN/RDA SPII SAM AS-IS and T0-BE State Reports.

- The Program Offices will have access to in-house SW experts with the technical and acquisition process experience to aid the program offices in managing the industry development teams.
- The in-house experts will have the applied knowledge to assess industry technical approaches and also their SW development processes. This includes having in-house experience and metrics from SW cost and schedule estimates and thereby be able to provide support for independent cost and schedule assessments.
- The in-house SW experts will have applied experience with developing and implementing system requirements at all levels, and this will enable them to support requirement management and volatility risk reduction.
- The government SW engineers will have in-depth knowledge of various weapon system architectures and maintain the corporate knowledge required to mitigate the risk of program office leadership and personnel turnover.
- The in-house SW engineers will have applied experience with EVMS and can aid the program offices in setting up realistic and meaningful SW-based EVMS processes and tools.
- By maintaining SW engineers with applied experience in both previous and current complex SW development efforts, the program offices will have a source of objective lessons learned and metrics that can be applied to future SW development process improvement.

Another challenge of relying on private industry for 100% of the software development is that it leaves the program office with no leverage over the contractor; and with very few schedule, cost or performance risk mitigation strategies when the private contractor is failing to meet the program needs. By the time the program office realizes the contractor has significant problems, the program is in “too deep” with that company to have any other choice than to continue funding the poor performing contractor and hope for the best.

Firing the contractor and transferring the work to another private industry contractor is rarely a viable option. The impact to cost and schedule is such that the only risk mitigation options include:

- Significantly increasing funding
- Significantly delaying the schedule
- Significantly reducing or eliminating planned capabilities
- Canceling the program

By establishing and maintaining integrated SW development teams, the program office leadership will have the option to augment the contractor SW team with on-site government SW engineers, or transfer the responsibility for SW component development from the contractor to the Government. This can be accomplished easily as the Government software engineers are part of the software development team from the beginning. There will be no need to perform a costly re-competition to assign the work to another private industry team that would be unfamiliar with the program requirements and plan. Under the proposed new software acquisition strategy, the Government would have contracts in place that specify all developed system artifacts become the property of the US Government. This mitigation technique only accelerates the delivery. There is of course still some added schedule risk as the in-house team must work with the contractor to transfer all necessary artifacts to assume full development responsibility. If the program office and development items established an Integrated Development Environment (IDE) however, this transfer of artifact responsibility is relatively easy.

Program offices will also have the option of directing the Government in-house software experts to provide onsite support to aid the contractor in recovering schedule progress or resolving technical problems. Given the DoD approach for rotating the military leaders to gain a wide range of experiences, it is common for a software intensive system to have acquisition program leadership personnel that have no significant training or, more importantly, any applied experience in software engineering. A closely related challenge is that acquisition program office leadership transition may occur at any point during the system development effort.

A single Program manager may not manage the system acquisition and development program from the beginning (version content definition) completely through to the end (through IOC). The development organizations are faced with the challenge of still meeting the previously defined development milestones and delivery dates, while simultaneously changing organizational structures, reporting chain of commands, tasking priority changes, funding reallocation, and development process changes directed by the new leadership. Maintaining an experienced Government SW development organization mitigates the impact of frequent senior leadership changes. The experienced SW development team can provide the following benefits to the acquisition office's new leadership:

- Maintain critical system functional, architectural, and design corporate knowledge
- Aid the new leadership in quickly coming up to speed on the history of the program, the system's architecture and functionality, the various development organization's roles and responsibilities, current development process, and status of the current development efforts (schedule, progress, and risk)
- Provide impact assessment for proposed organizational and/or process changes
- Perform the Technical Authority responsibilities for those leaders without extensive training or applied experience in software intensive systems

Future State: Establishing the Pipe-Line

The DoD/Navy must re-assume leadership of software architecture and design. Government software architecture and technical authority must be demonstrated not just at the highest system composition level (i.e., Objective Architecture Functional Domain level), but must extend down into lower critical sub-component levels as well as illustrated in Figure 10. In-house software SMEs should serve as the software technical authority and the software architects, and lead critical software sub-element development IPTs.

The DoD/Navy must develop and document a software acquisition improvement vision with a quantifiable goal. Critical weapon and warfare system program offices should work with the in-house software development organizations to develop transition plans to achieve the vision goals. This software expertise pipeline must be continually fed and maintained. In order to attract and keep the best and brightest software engineers, the Government must offer:

- Challenging software development and leadership responsibilities
- Opportunities of architecting, designing and implementing solutions to the most complex types of system functional capabilities and problems
- Opportunities to utilize the latest software technologies, methodologies, processes, tools

Government engineers should not be limited to developing tactical software only (where tactical software is defined as software utilized with delivered warfighting systems with strategic or tactical mission critical requirements). They must stay abreast and have applied expertise with all the latest software technologies. In addition to performing tactical SW development, another way to achieve this goal is to assign non-tactical (e.g., system or architecture modeling software, simulation software, testing software, media generation software, data distribution software) to in-house engineers. It is often possible to use the latest software development technologies and methodologies for non-tactical software as the acquisition cycle may be much shorter and the certification process less stringent than for tactical systems

Development of non-tactical and non-critical software components can serve as a test bed and as a cost, schedule, and technical performance risk mitigation strategy for determining if new software technology is of sufficient maturity and capability to be incorporated into the current or next version of critical tactical system(s). The two key questions that must be addressed when determining what software should be assigned to a Government software development organization are:

1. Will this assignment help maintain the software expertise pipeline?

5. Will this assignment maintain corporate expertise and mitigate the cost, schedule, and/or technical performance risks of existing or future systems?

As directed in the 2008 Mr. Donald Winter SECDEF memo: "This combination of personnel reductions and reduced RDT&E has seriously eroded the Department's domain knowledge and produced an over-reliance on contractors to perform core in-house technical functions. This environment has lead to outsourcing the "hands-on" work that is needed in-house, to acquire the Nations best science and engineering talent and to equip them to meet the challenges of the future Navy." And "In order to acquire DoN platforms and weapons systems in a responsible manner, it is imperative the DoN maintain technical domain expertise at all levels of the acquisition infrastructure."

The current undefined, undocumented, non-standardized, and non-disciplined “ad hoc” assignment of SW development to in-house SW development organizations is insufficient to achieve and maintain the much needed SW expertise pipe-line. The DoD/Navy should develop a well defined and documented software development assessment and assignment process and criteria. This process and criteria will be utilized by software intensive system acquisition program offices to assign software development responsibility to integrated government and software development teams.

References

Assistant Secretary of the Navy Research Development and Acquisition (ASN/RDA). (2007, 2008). Chief Engineer, Software Process Initiative Software Acquisition Management Focus Team. *As-is* and *to-be state reports*.

Assistant Secretary of the Navy Research Development and Acquisition (ASN/RDA). (2009, July). *Department of the Navy (DoN) software measurement policy for software intensive systems*. ASN/RDA Memorandum.

Assistant Secretary of the Navy Research Development and Acquisition (ASN/RDA). (2008, December). *Strategy to balance acquisition in-house and contractor support capabilities.* ASN/RDA Memorandum.

Assistant Secretary of the Navy Research Development and Acquisition (ASN/RDA). (2008, November). *Meeting of the Navy laboratory/center competency group.* ASN/RDA Memorandum.

GAO. (2004, March). GAO-04-393. Washington, DC: Author.

GAO. (2008, February). *Report to Congressional committees best practices.*

Levin, C. (2009, March). *US Senate Committee of Armed Services*. Press Release.

OUSD(AT&L). (2000, November). *Report of the Defense Science Board (DSB) task force on defense software*.

OUSD(AT&L). (2008, May). *Report of the Defense Science Board (DSB) task force on developmental test and evaluation*.

Secretary of Defense (SECDEF). (2008, October). *Department of the Navy Acquisition*. SECDEF Memorandum.

On Open and Collaborative Software Development in the DoD

Scott Hissam—Scott Hissam is a Senior Member of the Technical Staff for the Carnegie Mellon Software Engineering Institute, where he conducts research on component-based software engineering, open source software, and multicore. Mr. Hissam is a founding member and secretary of the International Federation for Information Processing (IFIP) Working Group 2.13 on Open Source Software and co-organizer of its annual conference. His publications include two books (*Building Systems from Commercial Components* and *Perspectives on Free and Open Source Software*), papers published in international journals, and numerous technical reports. He has a BS in Computer Science from West Virginia University.

Scott A. Hissam
Carnegie Mellon Software Engineering Institute
4500 5th Avenue
Pittsburgh, PA, 15213 USA
+1.412.268.6526
shissam@sei.cmu.edu

Charles B. Weinstock—Charles B. Weinstock is in the Research, Technology, and System Solutions Program at the Software Engineering Institute. His main interest is in dependable computing. For the last several years, he has been developing assurance case technology. He is also active in the open source software community. Previously, Weinstock worked at Tartan Laboratories and SRI International. Weinstock has a PhD in Computer Science, an MS in Industrial Administration (MBA), and a BS in Mathematics, all from Carnegie Mellon. He is a Senior Member of the IEEE and a member of IFIP Working Group 10.4 on Dependable Computing and Fault Tolerance.

Charles B. Weinstock
Carnegie Mellon Software Engineering Institute
4500 5th Avenue
Pittsburgh, PA, 15213 USA
+1.412.268.7719
weinstock@sei.cmu.edu

Len Bass—Len Bass is a Senior Member of the Technical Staff at the Software Engineering Institute. He has authored two award-winning books in software architecture and several other books and papers in various computer science and software engineering areas. He has been a keynote speaker or a distinguished lecturer on six continents. He is currently working on techniques for the methodical design of software architectures, supporting usability through software architecture, and to understand the relationship between software architecture and global software development practices. He has worked in the development of numerous software systems, ranging in a multitude of domains.

Len Bass
Carnegie Mellon Software Engineering Institute
4500 5th Avenue
Pittsburgh, PA, 15213 USA
+1.412.268.6763
ljb@sei.cmu.edu

Abstract

The US Department of Defense (specifically, but not limited to, the DoD CIO's Clarifying Guidance Regarding Open Source Software, DISA's launch of Forge.mil and OSD's Open Technology Development Roadmap Plan) has called for increased use of open source software and the adoption of best practices from the free/open source software (F/OSS) community to foster greater reuse and innovation between programs in the DoD. In our paper, we examine some key aspects of open and collaborative software development inspired by the success of the F/OSS movement as it might manifest itself within the US DoD. This examination is made from two perspectives: the reuse potential among DoD programs sharing software and the incentives, strategies and policies that will be required to foster a culture of collaboration needed to achieve the benefits indicative of F/OSS. Our conclusion is that to achieve predictable and expected reuse, not only are technical infrastructures needed, but also a shift to the business practices in the software development and delivery pattern seen in the traditional acquisition lifecycle is needed. Thus, there is potential to overcome the challenges discussed within this paper to engender a culture of openness and community collaboration to support the DoD mission.

Keywords: Open source software, software engineering, reuse, collaborative development

Introduction

Free and open source software (F/OSS) has been available, in one form or another, for several decades. Successful F/OSS projects benefit from the efforts of a large, usually diverse set of developers. For such projects, the software developed is often as good as or better than the best commercially available software. An even larger community is able to make use of and reap the benefits of this software. The DoD (US Department of Defense) would like to capitalize on this success and adopt an F/OSS model to exploit both reuse among DoD programs and collaboration to improve quality, spark innovation, and reduce time and cost.

The Open Technology Development (OTD) Roadmap Plan prepared for Ms. Sue Payton, Deputy Under Secretary for Defense, Advance Systems and Concepts, identified the following advantages sought from adopting OSS development methodologies (Herz, Lucas & Scott, 2006):

- Encourages software re-use [*sic*],
- Can increase code quality and security,
- Potentially subject to scrutiny by many eyes,
- Decreases vendor lock-in,
- Reduces cost of acquisition,
- Increases customizability, and
- Meritocratic community.

Most recently, Dan Risacher, Office of the Assistant Secretary of Defense (ASD), Networks and Information Integration (NII), was quoted by *Government Computing News* (Jackson, 2008) regarding the benefits of F/OSS as it might apply to defense agencies:

> By using open-source software, the services can update their software as soon as a vulnerability is found or an update is needed, rather than wait for the vendor to supply a patch. Open source also promises faster prototyping of systems, and lower barriers to exit. And if a government-written application is released into open source, outside developers could work to fix the problem, lowering maintenance costs of software.

This office is in the process of updating the Stenbit memorandum clarifying the use of F/OSS in DoD programs (Stenbit, 2003).

What is important about these two data points is that they illustrate the level of expectation that is driving the push for the adoption of the F/OSS model of open and collaborative software development in the DoD software community.

This paper explores the idea of adapting the F/OSS model to the DoD software community. While there are a number of other significant concerns mentioned, this paper concentrates on addressing two that are of interest. The first is reasoning how an open and collaborate approach would need to operate in the DoD community, assuming that community was motivated to behave in the same manner as seen in the public F/OSS community. The second focuses on this assumption and reasons as to how to incentivize the DoD community to make use of, and contribute to, such a resource.

The remainder of this paper is laid out as follows: Section 2 looks at the progressive movement towards F/OSS and some of the software reuse repositories (and their challenges) that proceeded today's F/OSS movement. Section 3 takes an abstract view of a project's operation in SourceForge.net as a means for understanding how such resources support the F/OSS community and what they do not do to illustrate a gap that is needed to be filled to support reuse across the DoD community. Section 3 then instantiates this abstract view for use in the DoD to consider the ways in which a DoD-specific resource would compare to that seen in the F/OSS community. Section 4 addresses the prior assumption about behavior expected by the DoD community to consider the incentives necessary to create a healthy and collaborative DoD OSS community. Sections 5 and 6 provide final thoughts on points not yet addressed (perhaps motivating further discussion) and summarize the positions stated in this paper.

The following closely related and relevant topics are beyond the scope of this immediate paper: data rights/licensing issues (commercial, F/OSS, or otherwise); security classifications; various software lifecycle stages beyond IOC (initial operational capability), i.e., pre-RFP (request for proposal) tensions; maintenance of fielded system; field upgrade (new capability); and new systems reusing or proposing to reuse from prior systems.

History of Collaboration and Reuse

There are a number of papers, articles, and publications on the history of F/OSS, some tracing their beginnings to SHARE and the SHARE library in 1955, "to help scientific users grapple with the problems of IBM's first major commercial mainframe" (Gardner, 2005). Others trace to the earlier PACT (Project for the Advancement of Coding Techniques) initiative in 1953, a collaboration between the military and aviation industries (Melahn, 1956; Feller & Fitzgerald, 2001). Feller and Fitzgerald's book provides a nice treatise on the history of F/OSS from these beginnings through the Berkeley Software Distribution, T_EX, the creation of the Free Software Foundation (FSF) and GNU (GNU is Not Unix) and, eventually, to the creation of the Open Source Initiative (OSI). With the advent of the ARPANET during these emerging beginnings of the modern F/OSS movement, general software repositories began to appear; the most popular included SIMTEL20, originally hosted at MIT (Granoff, 2002), as well as tools to aid in searching these repositories, such as Archie and gopher (Howe, 2009).

With the ever-growing increase in the availability of F/OSS, the benefits of software reuse was also gaining traction within the DoD. In the late 80s (particularly with the DoD's adoption of the Ada programming language) and early 90s, various software reuse efforts within the DoD emerged, including STARS, STARS SCAI, ASSET, CARDS, PRISM, DSRS, ELSA, DSSA ADAGE, and RICC (Department of the United States Air Force [USAF], 1996). Although differences did exist among these repositories with respect to artifact management philosophies, some adopted a generally common theme centered on repositories of reusable software artifacts (code, documentation, etc.) having domain- and/or application-specific classifications, taxonomies, and software architectures all supported by techniques and methods embracing reuse in software development—essentially advocating the concepts that are among the underpinnings of software product lines (SPL) (Clements & Northrop, 2001).

Many of these repositories listed above are no longer in existence, even though their concepts are (in the authors' opinion) sound. Although a case study to completely understand why these efforts ceased would be nice—not the purpose of this paper—we will briefly touch on some of the technical challenges that faced some of the efforts. These include:

- **Quality Arbitration:** The administrative function of deciding what is and what is not included in the repository. This ranges from accepting everything (perhaps resulting in a junk yard or flea market) to a decisive selection (an inventory of few precious selections). Deciding which is the most appropriate is challenging. For the latter, repository customers have higher confidence in artifacts extracted at a higher cost of upfront qualification and an administrative bottleneck in populating the repository. This philosophical difference resulted in two camps: managed and unmanaged repositories.
- **Search and Browse:** At the time of these repositories, free text search and retrieval was a serious resource and computational problem. Free text was not practical; search was a matter of defining a well-crafted database schema, typically relational. There were two approaches. In one, a general purpose schema was defined; in another, domain analysis was used to identify domain specific concepts and terminology. Frakes demonstrated, however, that there was no substantial gain in user search performance obtained by the extra cost and effort of domain analysis (Frakes & Nejmeh, 1987). With time and advances, such free text search capabilities are now more common place (e.g., Google) and no longer presents a major hurdle.
- **Beyond Search and Browse:** Some argued that critiquing domain analysis with respect to retrieval of single reuse items missed the point. Capturing the (sometimes complex) relationships among domain concepts, spanning requirements, algorithms, architecture, code, test, and other artifacts was what was important. The CARDS repository (Wallnau, 1992), for example, used the KL-ONE (Brachman & Schmolze, 1985) semantic network formalism to capture these relations, and use them to support reuse of large-scale domain structures. Today's work in Web Ontologies also uses a descendant of KL-ONE, and for much the same purpose.

Altogether, this history lesson is worth remembering. In comparison, we believe that the infrastructures supporting the F/OSS community are superior for collaborative development for the projects they service—something that past reuse repositories never imagined. For the larger F/OSS community, these infrastructures are similar to past unmanaged reuse repositories capable of great (seemingly effortless) free text search suitable for opportunistic reuse. We examine this position in more detail below.

Infrastructures for Reuse and Collaboration

There are a number of resources available to the F/OSS community for F/OSS projects including SourceForge.net, RubyForge, JavaForge, Tigris.org, and freshmeat.net, only to name a few. An abstract view of SourceForge.net is created here for the purpose of understanding what such resources commonly do to support the F/OSS community and also what they don't do as a means to illustrate gaps in what is needed to support reuse across the DoD community as well as what would be needed in the DoD to support open and collaborative software development.

SourceForge.net®

SourceForge.net, owned and operated by SourceForge, Inc. (SourceForge, 2009a), is by all accounts one of the most successful source code repositories in the last decade, now boasting over 180,000 projects and nearly 2 million registered users (SourceForge, 2009b). However, simply referring to SourceForge.net as a (software reuse) repository is a great misnomer. Yes, SourceForge.net contains software source code (some of which is reused everyday), but SourceForge.net provides a wealth of other IT-related (hosting and backup) services to the F/OSS community as well as collaborative software engineering and project management tools.

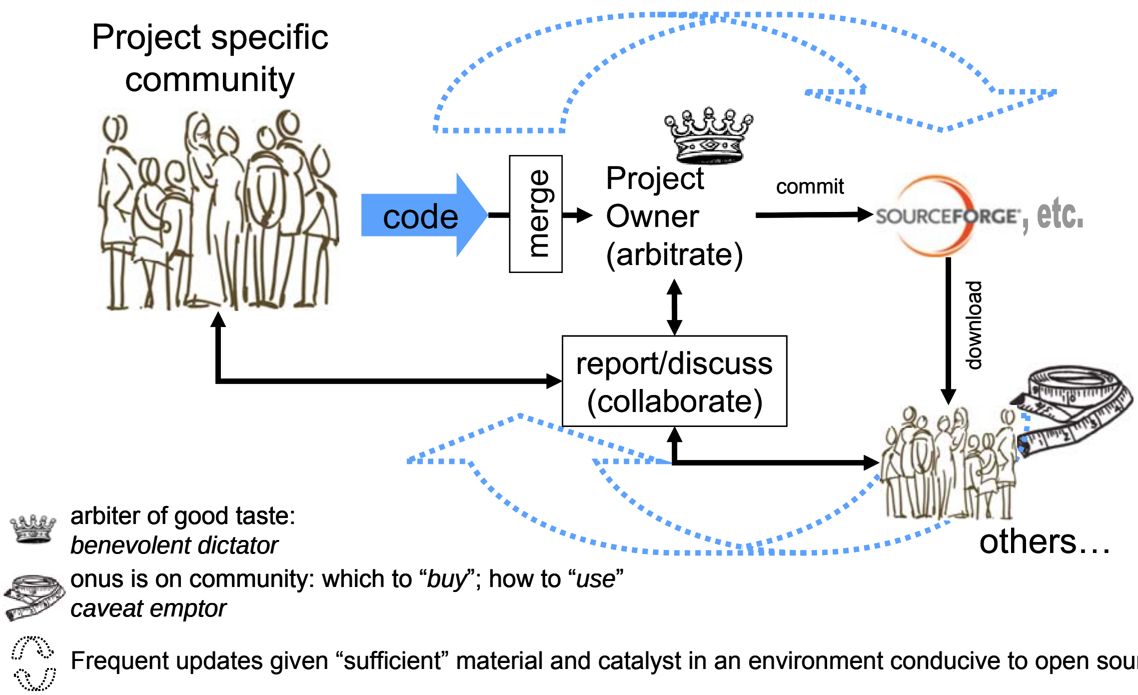


Figure 1. Abstract View of a SourceForge.net Project's Operation

SourceForge.net can best be thought of as a collection of self-contained projects. Each project is administered and owned by a project owner(s) who arbitrates (and delegates) ultimate control over what is committed into the project's code (or artifact) base, what software features are added or removed (over time), and the priorities upon which work progresses. The project's ownership determines the degree of control that is asserted over the project. The project owner is depicted as a crown in **Error! Reference source not found.** as a means to connote the "power" those arbitrators have over the project.

As work progresses, those arbitrators are continuously making collaborative decisions about what is to be done next. For simplicity, the focus for this discussion is on changes offered from the project specific community (on the left of **Error! Reference source not found.**) to the projects artifacts. By balancing their priorities and plans, the arbitrator make decisions on how to merge the interests of this community and the larger F/OSS communities to make changes (and commit those changes) to the artifact base. This churning effect (represented by the cyclic, thick arrows in **Error! Reference source not found.**) is an important and vital aspect of F/OSS collaborative software development. Succinctly, it is this churning and frequent updates (i.e., "release early, release often") to the artifacts that spark innovation through incremental improvements to early and emerging design and source code artifacts given that such updates are open and observable by all in the F/OSS community (Goldman & Gabriel, 2005). This is a continuous, open, and insightful process that is not driven by some external calendar, fiscal boundaries or legal/acquisition milestones.

Lastly, others are free to download software artifacts from the project's repository codebase. This group (in the lower right of **Error! Reference source not found.**) is separated from the project specific community to the left as a means to indicate others[38] that have tangentially "stumbled" upon the project (by whatever means—by search, by reputation, etc.). This group serves a useful purpose in this paper to illustrate another crucial point—that is Eric Raymond's caution in *The Cathedral and the Bazaar*, *caveat emptor*—"let the buyer beware" (Raymond, 2001). This is represented by the large measuring tape in **Error! Reference source not found.**.

Like the earlier users of SIMTEL20, Archie, and gopher, the onus is on this group to determine the degree of fit between artifacts retrieved from the project's codebase and their own needs. One aspect of this determination is partially driven by the need to ascertain if a search actually returned a relevant hit. That is, did the search terms find that which was sought? This is something that was recognized early and many of the DoD software reuse repositories tried to address this with various approaches to classifications and data definitions, for instance ASSET's approach was a faceted classification schema (USAF, 1996; Kempe, 1998) in which CARDS's approach was a domain-specific repository (software for a specific application domain, e.g., command centers). SourceForge.net's classification scheme for projects themselves is limited to broad project categories (for example, Games/Entertainment, Scientific/Engineering, and Security) and subcategories as well as filters allowing other search criteria such as language, operating system, and even licensing. SourceForge.net also provides mechanisms to search across projects (limited to free text searches of project's names and descriptions), to conduct searches within a project (for example within its documentation, forums, bugs, mailing lists, and configured download

[38] Such individuals may become part of the F/OSS community for a project through a variety of actions, including reporting bug, finding bugs, helping others, porting, contributing ideas, code, etc.

packages), and find published files (but not within CVS or SVN—two popular version control systems).

Another important aspect is determining the quality of the artifacts found. If quality is assumed by reputation (e.g., Apache, MySQL, and a host of other reputable F/OSS offerings), this may be no more difficult than in the past with the reputable software of that era (e.g., wuftpd, X, and many of the popular GNU offerings). However, putting reputation aside, quality of the software artifact is at the sole discretion of the project owner—and this has to be discovered in effort expended by the "buyer" through learning, inspection, trial, and testing.

Perhaps the most important aspect is determining if the artifact can actually be reused in the context of the "buyer's" need. The software found may be relevant, and it may be of high quality (by reputation), but may be architected and designed with assumptions that are inconsistent with the context in which it is intended to be reused. One example the authors experienced was to discover that a highly relevant and reputable MP3 encoder/decoder library could not be reused due to the fact that the decoder was implemented in a manner that was not thread safe, even though the encoder portion was. This resulted in an architectural mismatch that prevented reuse in this case. The CARDS and STARS SCAI (USAF, 1996) were some of the earliest DoD software reuse repositories that recognized the need to minimize this mismatch by adopting architecture-centric approaches as a means for qualifying software for reuse within a specific domain.

To summarize key points taken from this abstract view:

- These F/OSS resources (such as SourceForge.net) are for IT-related services housing F/OSS projects and their artifacts with facilities supporting open and collaborative development.
- Project artifacts themselves are managed by a project owner(s) having sole arbitration over the entire project.
- Artifacts are frequently updated and churned over by the F/OSS community, resulting in better quality and innovation.
- It is up to others expending real effort to find, inspect, and assess project artifacts for reuse within their context.

DoDSF

The idea of creating a "SourceForge.net" within the US Government or US Department of Defense, i.e., a "SourceForge.mil" was not invented by us. We credit Schaefer (2005) for the name. Furthermore, the OTD Roadmap called for "an internal DoD collaborative code repository" (Herz et al., 2006). So rather than conflate our analysis with any intent others may have with this idea (either in the past, present or future), we instantiate our thinking by using the term "DoDSF" (a DoD SourceForge).

Like SourceForge.net, DoDSF could also support the IT-related (hosting and backup) services to the DoD community as well as the collaborative software engineering and project management tools, but cast in the setting of a DoD program acquisition.[39] Using **Error!**

[39] This is not intended to be narrow, as we recognize that post deployment maintenance and long-term support would also have to benefit from open, collaborative and continuous software development. The description here is suitable for our discussion.

Reference source not found. as a basis for DoDSF, **Error! Reference source not found.** illustrates a number of similarities and differences that can immediately be teased out.

Working left to right in **Error! Reference source not found.**, the project specific community is the first difference. In this case, the project specific community is not identical to the wider F/OSS community served by F/OSS collaborative resources on the Internet. In the case of DoDSF, it is likely and expected that DoDSF will be gated in some manner, thus losing the 'F/O' as in F/OSS. The reality is that there will be classified software that the DoD hopes and expects to be reused and to evolve in a collaborative sense. Therefore, the openness assumed and intended for DoDSF will be as open as it can be for those in the gated community. This is not unprecedented; over the last decade, many private corporations—also wanting to reap the benefits of open and collaborative software development—have adapted F/OSS ideals. Such initiatives have been labeled using the terms corporate source (Dinkelacker, & Garg, 2001), progressive open source (Melian, 2007), and inner corporate source (Wesselius, 2006).

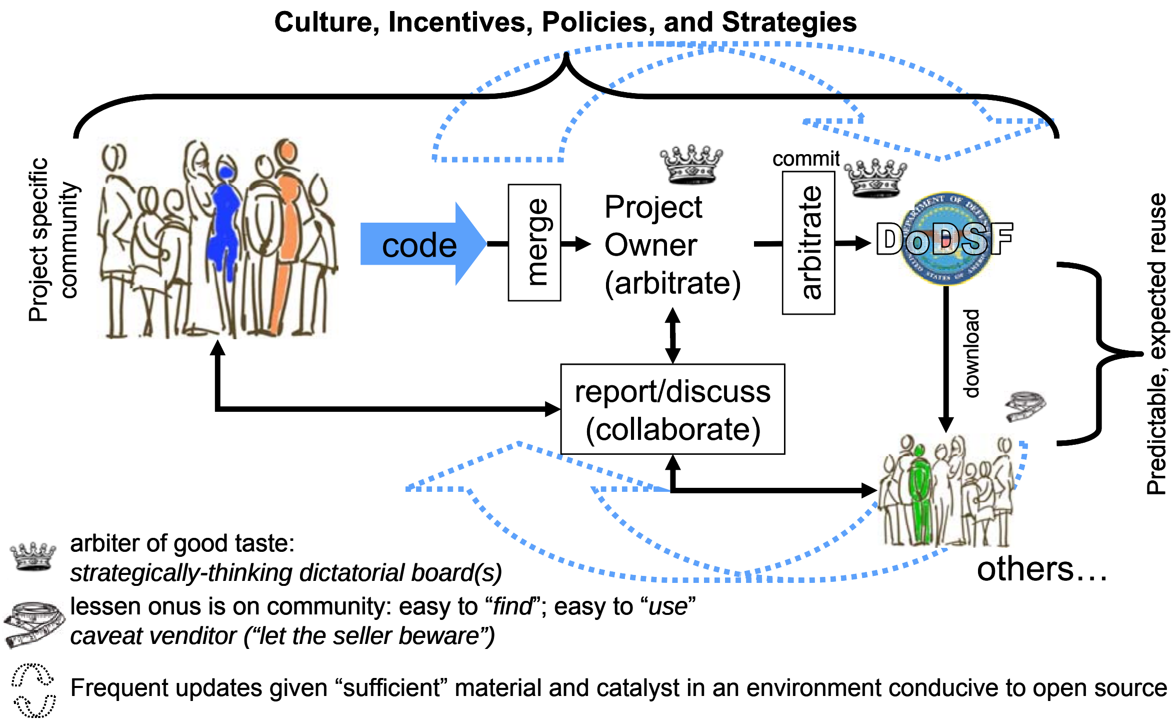


Figure 2. Abstract View of a DoDSF Project's Operation

The other difference in this community is its mix (as denoted by the shading of some of the characters in **Error! Reference source not found.**). Some from the community will likely be employees of private companies under contract to the DoD and under the oversight of a government program office—it is not assumed that these are the same private companies, contracts or government offices; it is only assumed that they share common needs and concerns. This too, is not unprecedented. In the F/OSS community, an increasing number of private companies allocate resources to F/OSS projects and some companies even sponsor F/OSS projects, for example, MySQL, IBM for Eclipse, and Sun Microsystems for OpenOffice.org.

Moving further to the right in **Error! Reference source not found.**, the next significant difference is the introduction of an additional commit and arbitration step and a second crown. This abstraction is added to our DoDSF as a means to rectify weaknesses in the SourceForge.net abstraction discussed earlier regarding *caveat emptor* and the burden that is placed on the larger community having to assess a project artifact's degree of fit. As in F/OSS projects, it is expected that projects will continue to have "project owner(s)" that arbitrate (and delegate) ultimate control over what is committed into the project's code (or artifact) base, what software features are added or removed (over time), and the priorities upon which work progresses.

What is different with the introduction of the additional step is that these project owners are not the sole arbitrator as to what (specifically) from the project's codebase is actually committed to DoDSF. This additional arbitration step is needed to ensure that which is being submitted to DoDSF is consistent with the domain- or application-specific nature reflected onto DoDSF—in other words, the project's artifact is consistent with the architecture and variation mechanisms expected and needed for effective reuse of artifacts contained within DoDSF (Bachmann & Clements, 2005). How and who conducts that additional arbitration certainly would need to be addressed. Some software reuse repositories discussed earlier, specifically STARS SCAI and CARDS, used domain engineering approaches (i.e., domain managers) reflective of software product lines (i.e., product line manager) to oversee such consistency (USAF, 1996; Clements & Northrop, 2001). This, in effect, would empower the administrators or arbitrators (the second crown) of DoDSF with a role in quality arbitration not seen in SourceForge.net and reminiscent of earlier software reuse repositories, thereby affording the opportunity for a software product line approach.[40]

Given this additional step, the intent would be to reduce the real effort expended by others who find and assess artifacts downloaded from DoDSF for fitness for use and to increase the likelihood that those artifacts can be reused within their context (denoted by the smaller size of the measuring tape in **Error! Reference source not found.**). This represents a fundamental shift from the model in the F/OSS community of *caveat emptor* with the onus on the "buyer" to *caveat venditor*, or "let the seller beware," as the onus would shift to the product line managers to ensure that the artifacts committed to DoDSF are fit for (re-)use.

Continuing on the journey around **Error! Reference source not found.**, the next visual clue introduced is that in the lower right, depicting the group separate from the project specific group. This group is the same as that served in the F/OSS abstraction discussed earlier—a group that has come to DoDSF to find and reuse artifacts suitable for their context. However, this group has the foreknowledge that artifacts within DoDSF have been developed following product line practices. That would mean that DoDSF could have domain- and/or application-specific classifications, taxonomies, and software architectures that are meaningful to the DoD community and commonality across similar projects.

Like **Error! Reference source not found.**, **Error! Reference source not found.** also includes cyclic, thick arrows to represent, in this case, a need for frequent updates to artifacts contained within DoDSF. Like the F/OSS community, the DoD community should also be continuous in its endeavor to improve the quality of its software through open and collaborative development. And, like its F/OSS counterpart, updates of artifacts to DoDSF should not be bound exclusively by fixed or planned milestones, as traditionally thought in contracted software acquisition. Rather, here, updates are driven by the DoD community.

[40] Additional opportunities for collaboration are possible with the "project owners," including the supplier, users, and others in the DoD community with the arbitrators working this additional step.

Without this cyclic churning, for example, a project artifact is only submitted to DoDSF at or near the “completion” of a project; there then is no opportunity for DoD community feedback and participation in the open and collaborative process that is expected to improve quality or spark innovation. Inclusion of this cyclic churning is a significant break from the software development delivery pattern seen in the traditional DoD software acquisition lifecycle. To summarize key points taken from this DoDSF view:

- Like SourceForge.net, DoDSF would be a resource for IT-related services housing artifacts from DoD projects supporting open and collaborative development.
- Although the “project owner” has purview over the DoD project itself, the artifacts that are committed to DoDSF are arbitrated in a manner that is consistent with a product line approach.
- The DoD community here is a gated community similar to the F/OSS collaborative model adapted by private companies.
- The mantra of “release early, release often,” indicative of F/OSS, is necessary to stimulate collaboration and spark innovation, as it does in the F/OSS community.

Throughout this discussion of DoDSF, it was assumed that the DoD community was motivated to behave in a manner that was consistent with the behavior often exhibited by the F/OSS community. We now turn our attention to this assumption.

Incentivizing a Culture of Collaboration, Innovation and Reuse

There is one final visual in **Error! Reference source not found.** to be discussed, that is the overarching “umbrella” of culture, incentives, policies, and strategies that must exist to engender the DoD community to behave in a manner that is indicative of openness and collaboration. The intent of this “umbrella” is to achieve the goals of reuse, quality and innovation coveted of the F/OSS community. Returning again to the OTD Roadmap, which recognized that their Roadmap “entails a parallel shift in acquisition methodologies and corporate attitude to facilitate discovery and re-use of software code across DoD.” The Roadmap goes on to explain that today’s acquisition model treats “DoD-developed software code as a physical good, DoD is limiting and restricting the ability of the market to compete for the provision of new and innovative solutions and capabilities.” So any reformulation of today’s acquisition model will fundamentally have to change the laws, policies and even thinking of the software code, not so much as a product, but more as means to mission capabilities and perhaps services. This is understandably a daunting task (white paper or not).

F/OSS Collaboration, Innovation and Reuse

Raymond’s comprehensive insight into the motivation of the F/OSS community is foundational (Raymond, 2001). For some, necessity is the only impetus—a simple need for something. And, fortunately, many in the F/OSS community have the ability to fulfill that need through coding. And when their ability is outstripped by the realities of the problem, they create an F/OSS project and hope that others having the skills join (the birth of a project community). Such people that lend their helping hands often do with the greatest of intentions perhaps motivated by the same need or simply just feel the need to do some technically interesting work (i.e., “scratch an itch” in **Error! Reference source not found.**).

Sometimes that “need” can already be satisfied by product offerings from the commercial marketplace (i.e., the Cathedral) but the desire is to make a better alternative to that offering, one that is free and open to all. Many F/OSS projects started this way.

(self) motivation | community | product(s)

simple need	innovators	useful
“scratch an itch”	developers	intended purpose
altruism	users	other purposes
competition	free riders, others…	

Figure 3. Culture of Collaboration in the F/OSS Community

As touched upon briefly above in Section 3, there is precedence for business models based on F/OSS projects. Many new projects have come and are coming into existence through software contributions *en masse* (e.g., Netscape’s Mozilla, Sun’s Java, IBM’s Eclipse, MySQL) as business opportunities appear from ancillary services through the contribution of these codebases and through their use. However, this in and of itself is not an answer, but it certainly presents evidence to the behavior that is desirable in the DoD community. The Ultra-Large-Scale Systems (ULS) study called for research in Social and Economic Foundations for Non-Competitive Social Collaboration as inspired, in part, by the F/OSS movement; “as pure self-interest is supplanted by altruistic motivations and the desire to be perceived as productive and intelligent” while at the same time recognizing the need for incentive structures encouraging the community to cooperate (Feiler et al., 2006).

It is also important to recognize those that are motivated to voluntarily offer their time and contribute to F/OSS projects. Some of the motivations just discussed apply to these individuals as well (i.e., altruism, itching, etc.), but further extend to the meritocratic—that is to (socially and in governance)—rise in the community to which they serve. Further, some see F/OSS projects as venues to show off their prowess, to develop skills that make them more employable, or to network with others (a social phenomenon). And practically, others need (not just want) to see that their modifications, enhancements, and features find there way back into the mainstream product. Otherwise, if the F/OSS community does not accept such changes, the only recourse is to reincorporate those changes into all future versions (i.e., rework) (Hissam & Weinstock, 2001).

Reasoning about DoDSF (Section 3) based on resources like SourceForge.net show DoDSF must differ if there is to be effective reuse for the DoD. For one, a DoD project is not likely to be incorporated in its entirety within some other DoD project. The projects are simply too big. However, there are certainly subsystems or modules of those overall projects that lend themselves to the DoDSF model. An example might be a subsystem that develops a common operational picture from a series of incoming tracks. To be able to reuse such a subsystem will require commonality at many levels, including mission needs, requirements, software architecture, design, data- and function-interdependencies, and other software artifacts.

Practically all of the Linux distributions (Debian, Fedora, Ubuntu, etc.) reuse the Linux kernel (www.kernel.org), which itself (Linux) has been ported to a wide variety of hardware architectures. In those distributions, other F/OSS applications are included (a list which is simply too long to even begin to enumerate). At the same time, like the Linux distributions, there are other POSIX-based distributions that are Linux-free, for example, Apple's Mac OS X, which is based on the Berkeley Software Distribution (BSD) of The Open Group's UNIX. And those same applications available to the Linux distributions are mostly available to Mac OS X. For the F/OSS community; the reasons for this are obvious: the underlying operating system, its architecture, interfaces (both for applications and device drivers), and interdependencies are openly specified, architected and, when necessary, debated. This leads to a shared understanding and context.

Baldwin and Clark (2006) argued that the architecture of F/OSS projects is a critical factor of the open and collaborative software development process in that it is the modularity of those architectures and the option values stemming from such modular architectures that contribute to collaboration and innovation. They noted that codebases that are more modular have more option value, thus attracting volunteers. That is the more modular and option rich, the more active and larger the innovator community is likely to be. Furthermore, it is these innovators that are incentivized to form voluntary, collective groups for the purpose of sharing and improving ideas. This, in and of itself, increases the likelihood of future variations and experimentation. Finally, the ULS report identified modularity as key to managing the complexity of software and to producing software systems amenable to change and to *concurrent* development—something that is clearly indicative of F/OSS collaborative development.

Looking again to some of the F/OSS "poster children," specifically Linux, Apache, and now Firefox (direct descendent of Netscape), those projects did not start out with wonderfully modular architectures. They only became modular after the complexity of features, project management, distributed development became too overwhelming and had to adapt. Chastek, McGregor, and Northrop (2007) identified Eclipse's plug-in (modular) architecture as one of the project's most valuable core assets, providing for multiple forms of variation including extension points of various types and (in the authors' opinion) learning from the lessons from past F/OSS projects.

To summarize key points taken from this F/OSS view:

- Some of the incentives that motivate individuals, groups, and companies to participate and collaborate in the F/OSS community can be explained, but more study is warranted.
- Some private companies have moved from treating software source code as a physical good and have found market opportunity in services from the use of the software.
- Modularity of an architecture not only promotes reuse, but is a key factor in spurring innovation in collaborative communities.
- Like F/OSS projects, software emanating from DoD projects will have to have architectures and interfaces that promote modularity and option value.

DoDSF Collaboration, Innovation and Reuse

Taking the key points from the previous sections, the "big money" question is how do these map into the gated DoD community that was established back in Section 3 (recall civilian and military personnel, along with employees of private companies under contract to the DoD and under the oversight of a government program office having common needs and concerns)? Furthermore, what needs to be done to change acquisition policy and strategy and to establish the incentives that will enable a culture and behavior similar to that seen in the F/OSS community?

As daunting as these questions may be, we humbly offer a few suggestions.

Recognize Product Line Practices are Not Free

Creating modularized subsystems and components that are consistent with the architecture and variability expected and needed for effective reuse will cost development dollars with payoff that may not be realized until the reuse of the component can be amortized. **Strategically, this should be expected and not avoided.** Furthermore, and before new components are created (or existing components are refactored), resources will have to be expended to identify product-line-wide architectures that are suitable for DoDSF and against which project artifacts are assessed before commitment to DoDSF. Such activities will likely require planning and development that are beyond any one project, yet are necessary for the projects themselves. Such planning includes mission objectives, product strategies, requirements analysis, architecture and design modifications, extra documentation, and packaging. Incentivizing the program managers that oversee these projects would require some combination of providing extra funding and making performance evaluation dependant on contributions to DoDSF.

Incentivize the "Churn"

If effort is to be expended to create a product-line-wide architecture for the DoDSF, and individuals across the DoD-wide enterprise are empowered as product line managers, the DoDSF has to be more than a "field of dreams" followed by the often cursing mantra "If you build it, they will come." Recognize that reuse is not free and that reuse does not come easily or by happenstance (Tracz, 1995). If the desired behavior of the DoD community is to use the DoDSF for finding project artifacts, then those artifacts have to be meaningful, relevant, and, by reputation, sound. Recall, the desire is to unburden the "buyer" from assessing the component's degree of fit—as expected in software product lines. By reducing this burden as a significant barrier to reuse, incentives may be necessary to bootstrap or kick start reciprocating contributions, feedback, improvements, and otherwise collaborative behaviors—but observations from the F/OSS community would lead to the belief that such incentives would not be necessary. But this is not entirely clear in the gated DoD community. Talented, willing and able civilian and military personnel may be more likely to behave in this manner. Employees of private companies—while on contract—might also behave in this manner. Again, there is precedent in the F/OSS community for private companies to commit resources to F/OSS projects. Following this model, perhaps there are incentives for contracting companies that are successful in getting subsystems and components into DoDSF—that being negotiated service contracts, thereby allowing for continued involvement servicing the DoD community.

There are good reasons (perhaps un-incentivized) that a new DoD project would prefer to see bidders propose using proven artifacts from DoDSF. Such includes less risk to the project—a subsystem taken from DoDSF is already a known quantity, and lower development costs allowing valuable program dollars to be used elsewhere in the program. A possible disincentive (or opportunity, perspective is everything) is that it may be viewed by Congress that the project should be built for less money because it uses a subsystem(s) in

DoDSF; the program office may be given less money to get the job done, which may be viewed as a negative outcome by some.

A supplier bidding on a project really has only two incentives to use an artifact contained in DoDSF. If the program office has indicated that the use of such artifacts will be a determining factor in a successful proposal, then there is a strong incentive to do so. In the absence of such a requirement, the supplier may be incentivized to reuse an artifact to enable it to be the lowest bidder.

Incentivize Software as a Non-Rivalrous Good

Treating source code as if it were a physical good is a mentality that inhibits collaboration. Rivalry should be encouraged between competing subsystems or components for the same role in a produce-line-wide architecture (i.e., let the stronger or better prevail). But the source code itself should serve as the source of inspiration, innovation and improvements for that "better" subsystem—rather than the opaque enigma requiring resources to be expended to re-engineer from scratch (or worse, reverse-engineer because the source code is long forgotten and lost).

Last Thoughts

Governance

Reminiscent of reuse repositories discussed in Section 2, great care has to be given in governance of DoDSF. The DoD must have a vested interest in seeing that the artifacts in DoDSF can be reused in subsequent projects. It has invested in them and would like to see a payback in terms of reduced development time, risk, and cost in the future. Thus, there is an upfront quality requirement for items to be placed into DoDSF. For SourceForge.net, the evaluation is ultimately done by the F/OSS community (using or not using) the project. For DoDSF there is presumably a contractual requirement regarding the subsystem. Someone has to evaluate the subsystem and its suitability for reuse, which needs to be a part of the original development contract. Otherwise there is every incentive for the supplier to place something into DoDSF that is ultimately unusable by anyone other than the original supplier.

Who does this evaluation? In the body of this paper, we placed the onus on the "seller" (*caveat venditor*), which, in this case, was tagged as the product line manager or the "second crown." In reality, that role will come down to real people in the DoD community. Determining just who exactly those individuals are is beyond the scope of this white paper, but it is certainly something that will have to be decided.

Security

In this white paper, we acknowledge that classification of project artifacts in DoDSF is a reality. This presents a challenge for DoDSF. If an artifact is from a top-secret project, then it may be difficult to declassify it for contribution to a DoDSF that does not respect security issues. But allowing DoDSF to embrace a multi-level security model raises concerns. Here's one example. Is a top secret project able to use an artifact classified at a lower level? If so, how does it trust it? If it makes modifications (even a bug fix) what happens to the security classification of the artifact when the modification is given back to DoDSF? Does this result in a security-level fork? There are many such questions that could be raised, but a further discussion of this is beyond the scope of this paper.

Summary

The number of references that were used in the preparation of this white paper was far more than any of the authors expected. This simply illustrates, in our opinion, the tip of a very large iceberg on the topic of reuse and F/OSS openness and collaboration coming from various disciplines.

Perhaps the most relevant reference that we came across for this paper was the Open Technology Development Roadmap Plan (Herz et al., 2006). Those interested in following up on some of the discussion covered in this paper should consider getting the latest progress on the actions called for within that Roadmap Plan. That plan called for very specific actions with respect to changing the traditional acquisition lifecycle. Most interesting was the recommendation: "Evaluate the potential use of the Defense Acquisition Challenge (DAC) program to demonstrate Open Technology alternatives to projects or programs that have implementation issues; e.g., make application of open source based products or development methodologies a specific interest item for DAC."

On the topic of product lines, it is worth noting that there are case studies that show how product line approaches can be effective and successful in industry and government ventures (USAF, 1996; Clements & Northrop, 2001; Jensen, 2007; Mebane & Ohta, 2007). Furthermore, there are efforts and thinking happening now to merge F/OSS models with software product lines (Chastek et al., 2007) and (van Gurp, Prehofer & Bosch, 2010) along with three international workshops on Open Source Software and Product Lines (specifically OSSPL 2006, OSSPL 2007, and OSSPL 2008).

F/OSS works today because of the culture, environment, and motivation touched upon in this white paper. It is important to note that this F/OSS culture was not planned at all, but is founded by a loose set of principles and rules (some of which are formalized through F/OSS licenses) that guide the behavior to achieve freely available, lightly controlled software developed in a collaborative manner. This behavior is informed by centuries of human populations and communities creating new knowledge and building off each other's work.

The question the readers should ask themselves (and we would not have done our job if you didn't ask yourself) is what would such principles and rules look like in a gated DoD community, a community itself informed by approximately 200 years of contracting, procurement and competition? Additionally, what is needed to foster the behavior the DoD wants to engender? What can the DoD control and what control must the DoD relinquish?

Acknowledgements

We would like to thank Gary Chastek, Terry Dailey, Bob Gobeille, Guy Martin, Catherina Melian, Linda Northrop, Robert Vietmeyer, and Kurt Wallnau for their thoughtful review and suggestions and with a special thanks to Nickolas Guertin whose curiosity, energy, and interest in the topic inspired this paper.

References

Bachmann, F., & Clements, P. (2005). *Variability in software product lines* (CMU/SEI-2005-TR-012). Pittsburgh, PA: Carnegie Mellon Software Engineering Institute.

Baldwin, C.Y., & Clark, K.B. (2006). The architecture of participation: Does code architecture mitigate free riding in the open source development model? *Management Science, INFORMS*, *52*(7), 1116-1127.

Brachman, R.J., & Schmolze, R.J. (1985). An overview of the KL-ONE knowledge representation system. *Cognitive Science*, *9*(2), 171-216.

Chastek, G.J., McGregor, J.D., & Northrop, L.M. (2007). *Observations from viewing eclipse as a product line*. Second International Workshop on Open Source Software and Product Lines 2007 (OSSPL07). Limerick, Ireland.

Clements, P., & Northrop, L. (2001). *Software product lines: Practices and patterns*. Boston, MA: Addison Wesley Professional.

Department of the United States Air Force (USAF). (1996). *Guidelines for successful acquisition and management of software intensive systems* (Vers. 2.0). Software Technology Support Center.

Dinkelacker, J., & Garg, P.K. (2001). *Corporate source: Applying open source concepts to a corporate environment*. First International Workshop on Open Source Software Engineering. The 23rd International Conference on Software Engineering (ICSE). Toronto, Canada.

Feiler, P., Gabriel , R., Goodenough, J., Linger, R., Longstaff, T., Kazman, R., Klein, M., Northrop, L., Schmidt, D., Sullivan., & Wallnau, K. (2006). *Ultra-large-scale systems: The software challenge of the future*. Retrieved from http://www.sei.cmu.edu/library/assets/ULS_Book20062.pdf

Feller, J., & Fitzgerald, B. (2001). *Understanding open source software development*. Boston, MA: Addison Wesley Professional.

Frakes, W.B., & Nejmeh, B.A. (1987). Software reuse through information retrieval. *SIGIR Forum*, *21*(1-2), 30–36.

Gardner, D. (2005, August 17). SHARE, IBM user group, to celebrate 50th anniversary. *InformationWeek*. Retrieved from http://www.informationweek.com/news/showArticle.jhtml?articleID=169400167

Goldman, R., & Gabriel, R.P. (2005). *Innovation happens elsewhere*. San Francisco, CA: Morgan Kaufmann.

Granoff, M. (2002). *The story of SIMTEL20*. FARNET Stories Project, Coalition for Networked Information, Interop, Inc., and the National Science Foundation. Retrieved March 2009, from http://www.cni.org/docs/farnet/story149.NM.html

Herz, J.C., Lucas, M., & Scott, J. (2006, April). *Open technology development roadmap plan*. Prepared for Ms. Sue Payton, Deputy Under Secretary of Defense, Advanced Systems & Concepts. (Vers. 3.1). Retrieved from http://www.acq.osd.mil/jctd/articles/OTDRoadmapFinal.pdf

Hissam, S., & Weinstock, C. (2001). *Open source software: The other commercial software*. First International Workshop on Open Source Software Engineering; 23rd International Conference on Software Engineering (ICSE). Toronto, Canada. Retrieved from http://opensource.ucc.ie/icse2001/hissamweinstock.pdf

Howe, W. (2009). *A brief history of the Internet*. Walt Howe's Internet Learning Center. Retrieved March 2009, from http://www.walthowe.com/navnet/history.html

Jackson, J. (2008, October 8). Pentagon: Open source good to go. *Government Computing News*. Retrieved from http://gcn.com/articles/2008/10/08/pentagon-open-source-good-to-go.aspx

Jensen, P. (2007). *Experiences with product line development of multi-discipline analysis software at Overwatch Textron Systems*. 11th International Software Product Line Conference (SPLC 2007). Kyoto, Japan.

Kempe Software Capital Enterprises. (1998). *ASSET: The asset source for software engineering technology*. Ada Home. Retrieved March 2009, from http://www.adahome.com/Network/Repositories.html

Mebane H., & Ohta, J.T. (2007). *Dynamic complexity and the Owen Firmware Product Line Program*. 11th International Software Product Line Conference (SPLC 2007). Kyoto, Japan.
Melahn, W. (1956). A description of a cooperative venture in the production of an automatic coding system. *Journal of the ACM (JACM)*, *3*(4), 266-271.
Melian, C. (2007). *Progressive open source: The construction of a development project at Hewlett-Packard*. Unpublished doctoral dissertation, Stockholm School of Economics, Sweden.
Raymond, E.S. (2001). *The cathedral & the bazaar: Musings on linux and open source by an accidental revolutionary*. Sebastopol, CA: O'Reilly Media.
Schaefer, T.M. (2005, January). [Letter to the Editor]. *Crosstalk, The Journal of Defense Software Engineering*, 18(1). Hill AFB, Utah. Retrieved from http://www.stsc.hill.af.mil/CrossTalk/2005/01/0501LetterToEditor.html
SourceForge, Inc. (2009a). *Terms of use*. Retrieved March 2009, from http://alexandria.wiki.sourceforge.net/Terms+of+Use
SourceForge, Inc. (2009b). *What is SourceForge.net?* Retrieved March 2009, from http://alexandria.wiki.sourceforge.net/What +is+SourceForge.net%3F
Stenbit, J.P. (2003, May 28). *Open source software (OSS) and the Department of Defense (DoD)*. Memorandum from the United States Department of Defense. Retrieved from http://cio-nii.defense.gov/docs/OpenSourceInDoD.pdf
Tracz, W. (1995). *Confessions of a used program salesman: Institutionalizing software reuse*. Boston, MA: Addison Wesley Longman.
van Gurp, J., Prehofer, C., & Bosch, J. (2010). Comparing practices for reuse in integration-oriented software product lines and large open source software projects. *Software: Practice and Experience*, *40*(4), 285-312.
Wallnau, K.C. (1992). An introduction to CARDS. *Crosstalk, The Journal of Defense Software Engineering*, (32). Hill AFB, Utah.
Wesselius, J. (2006). The bazaar inside the cathedral: Business models for internal markets. *IEEE Software*, *25*(3), 60-66.

THIS PAGE INTENTIONALLY LEFT BLANK

Panel #10 – Aligning Requirements with the Defense Acquisition Process

Wednesday, May 12, 2010	
3:30 p.m. – 5:00 p.m.	**Chair: Colonel Raymond D. Jones, US Army**, Program Manager, Airborne, Maritime and Fixed Station, Joint Tactical Radio System ***Illustrating the Concept of Operations (CONOPs) Continuum and Its Relationship to the Acquisition Lifecycle*** **Robert Edson** and Jaime Frittman, Analytic Services Inc. ***The Illusion of Certainty*** **Grady Campbell**, Carnegie Mellon University ***Towards Real-time Program Awareness via Lexical Link Analysis*** **Ying Zhao**, Shelley Gallup and Douglas MacKinnon, Naval Postgraduate School

Illustrating the Concept of Operations (CONOPs) Continuum and Its Relationship to the Acquisition Lifecycle

Robert Edson—Robert Edson is Vice President for Enterprise Development at Analytic Services Inc., and Director of the Applied Systems Thinking Institute (ASysT). He is responsible for strategic and transformational programs within the corporation. In his role as Director of ASysT, he leads an institute whose mission is to advance the application of systems-thinking principles in the fields of national security and homeland security. Robert is an Adjunct Professor at Stevens Institute of Technology and has a BS in Biology from George Mason University and a MS in Physical Oceanography and Meteorology from the Naval Postgraduate School.

Robert Edson
Analytic Services Inc, (ANSER.org) The Applied Systems Thinking Institute (ASysTi.org) 2900 South Quincy St. Suite 800 Arlington VA, 22206
Phone. 703.416.2000
Robert Edson@anser.org

Jaime Frittman—Jaime is a Senior Analyst with Analytic Services Inc. Jaime began her career in the United States Air Force working in the field of Readiness and Emergency Management. Jaime applied her knowledge of Chemical, Biological, Radiological, Nuclear (CBRN) Warfare Defense and Response to the development and acquisition of CBRN Defense Weapon Systems. Jaime has a Master's of Engineering in Systems Engineering from Stevens Institute of Technology.

Jaime Frittman
Analytic Services Inc, (ANSER.org) The Applied Systems Thinking Institute (ASysTi.org) 2900 South Quincy St. Suite 800 Arlington VA, 22206
Phone. 703.416.2000
Jaime.Frittman@anser.org

Abstract

Though consistently noted as critical to successful system design and implementation, the Concept of Operations (CONOPs) artifact appears to be underutilized. This report demystifies the CONOPs artifact. It delves into the barriers that prevent optimal use of CONOPs and presents a framework for incorporating an "integrated" CONOPs into the Defense Acquisition Lifecycle.

Introduction

The ability of development programs to avoid challenges associated with schedule, budget, and technical performance has been consistently poor (Turner, Verma & Weitekamp, 2009, p. 7). A recent FAA sponsored study noted that in order to avoid these pitfalls, "one of the most significant artifacts is the creation of a CONOPs" (Turner, et. al., 2009, p. 27). The report further noted the need to have "alignment between the evolving CONOPs, the [Enterprise Architecture],[41] and the governance system" (Turner et al., 2009,

[41] An enterprise architecture (EA) describes the "fundamental organization of a complex program…as a minimum, the EA relates the requirements, resourcing (funding), acquisition system and the program office within an agency and the overall business framework of key stakeholders" (Turner et al., 2009, pp. 17-18).

p. 32). The *Manual for the Operation of The Joint Capabilities Integration and Development System* (JCIDS, 2009) provides an illustration of the alignment of the enterprise architecture and the governance system by connecting JCIDS activities with milestone decisions. While important, this illustration is missing the alignment of a critical system success component, the CONOPs document. In order to encourage successful system development and acquisition we must understand the context of the CONOPs as it relates to the larger total acquisition lifecycle.

Research Goals and Objectives

This research informs the acquisition development lifecycle process by articulating the importance of the CONOPs–Acquisition relationship and by illustrating how various CONOPs documents are introduced at critical points in the JCIDS development timeline to create a more robust and integrated concept of operations.

Goals of the research include:

- Define the various CONOP types
- Explain the relationship of system-level CONOPs to acquisition activities
- Assess the current alignment of CONOPs and CONOPs-related documents with DoD acquisition governance and enterprise architecture processes
- Explore the maturity phases of CONOPs documents
- Document the relationship of each instantiation of the CONOPs to acquisition-related activities
- Assess the use of CONOPs and the disconnect, if any, between the perceived importance of CONOPs and the actual utilization of CONOPs.

Methodology

This research was conducted by combining traditional research methods with systems thinking tools and practices. Traditional analysis included literature review, data analysis, and comparative analysis. The Conceptagon[42] framework for systems thinking was applied to the research data. This framework encourages holistic system analysis by providing a series of seven “triplets” related to specific system characteristics. Use of the Conceptagon provided insight into interior and exterior boundaries, information flows, hierarchies, and other relevant system characteristics. Though the individual sets of triplets are not explicitly discussed in this paper, each of the seven triplets served as a cornerstone for consideration of system characteristics throughout this research.

Literature Review. A literature review of documents related to the role of CONOPs was conducted. This review included documents published in industry, in professional journals, acquisition journals, and in Department of Defense (DoD) regulations, instructions, and publications. The literature review also included the Defense Acquisition University Website which provided access to publications, communities of interest, and *ask a professor* question and answer forums.

[42] The Conceptagon is a systems thinking framework introduced by Boardman and Sauser (2008). For additional information on the Conceptagon as a systems thinking tool, reference *Systems Thinking: A Primer* (Edson, 2008) available at www.asysti.org.

In addition to existing literature, a questionnaire related to the use and usefulness of CONOPs was developed and distributed (see Appendix A). The pool of survey respondents was too small to enable the extraction of valid conclusions. To overcome the lack of respondents, results of the survey were compared to a similar survey[43] on the same subject.

Data Analysis. Information collected during the literature review was assessed for:

- Terms used
- Purpose
- Relationship to acquisition activities
- Relationship to integrated CONOPs

This assessment was instrumental in establishing a baseline for the CONOPS artifact and its use within the development and larger acquisition process.

Terms Used and Purpose of the Document. For the first assessment, a broad search of terms used synonymously with CONOPs was conducted. The initial assessment covered an array of CONOPs documents, looking at CONOPs that describe the actions of a military force or organization as well as CONOPs that detail characteristics of a system from an operator's point of view. The intent of this assessment was to determine consistency of the meaning and purpose of the term CONOPs and to identify terms used in place of "CONOPs." Once a set of recurring terminology was identified, the intended purpose of each document was recorded. This allowed us to assess similarities and variances associated with each of the terms. This assessment also gave us insight into role of CONOPs, if any, in acquisition activities as well as any barriers to the use of CONOPs.

Relationship to Acquisition Activities. Variances among purpose and meaning were detected in the initial assessment. To account for the variance, each CONOPs-related document was plotted on a JCIDS-Acquisition relationship diagram. This enabled us to visualize different points of input and influence of each of the identified CONOPs-related documents. Using this assessment we further identified three distinct phases of CONOPs development that directly correspond to acquisition related activities.

Relationship to Integrated CONOPs. Appearance of CONOPs-related documents in the JCIDS-Acquisition timeline revealed that CONOPs, either in an integrated form or in several smaller instantiations, occurred across the entire acquisition lifecycle. These documents (some termed "CONOPs," others operating under a different name) were then assessed for their similarity to an integrated ConOps document spanning the full acquisition lifecycle.[44]

Systems Thinking. Analyzing system characteristics by use of the Conceptagon provided a comprehensive view of the acquisition lifecycle. Each set of triplets was considered as we looked at each aspect of the project. To illustrate, as we looked at the

[43] The Roberts survey, conducted in 2008, inquired about the use, usefulness and upkeep of CONOPs. Roberts' survey had a larger pool of respondents numbering 108 responses from 18 different companies. This pool significantly outnumbered the 6 responses gained from our own survey. Unlike the Roberts survey which was sent to engineers, and was composed of system engineers, lead system engineers, test engineers, design engineers, and project managers (Roberts, 2008), our pool of respondents included members of the user community, which offered an additional perspective to data gained from the Roberts survey.

[44] For the purpose of this paper, the IEEE format for ConOps (IEEE, 1998) is representative of an integrated CONOPs. The IEEE nomenclature for a concept of operations is *ConOps* as opposed to *CONOPs*.

landscape of the system (i.e. governance, enterprise architecture, and CONOPs) we considered the triplet of wholes, parts, relationships. The larger acquisition system which included all three primary elements of the landscape was the whole, individual processes and inputs to the processes were considered parts, and the purpose of each input, and its effect on the whole constituted the relationships.

The Value of a CONOPs to System Development. The value of a CONOPs to system development is multi-faceted wherein the CONOPs plays a role across the entire life-cycle: from need identification, to system inception and development, to system disposition and disposal. Our research of literature, standards, and instructions indicates a number of ways in which the CONOPs adds value to acquisition and system development processes. Some of the key ways in which a CONOPs adds value are provided in Table 1.

Table 1. Value of the CONOPs

CONOPs Value
Helps scope the problem & solution
Bridges where we are and want to be
Illustrates how a system will function
Facilitates communications among stakeholders
Provides a logic trail of capability
Provides baseline for measuring system efficacy
Provides basis for requirements

Under-Utilization of the CONOPs

Despite its value, the CONOPs, at least in its full form, is not consistently used in system development. In fact, a recent survey showed that 1/3 of all programs queried did not have a CONOPs (Roberts, 2008, p. 39). Similarly, in a series of interviews and surveys conducted for this research, the majority of respondents indicated that a CONOPs was "critical" to the system's success but was under-utilized. Comparable studies on CONOPs have pointed out that even when a CONOPs is written it is often after the system is developed and done so in an effort to satisfy a Milestone Decision requirement; this "box-checking" activity strips the CONOPs of its intended role in the creative process (Nelson, 2007, pp. 5-6). Our survey results appear to support this, with our respondents indicating that a concept for how the system will be employed is usually written, but it is written after the system is developed. This means the CONOPs is based on the requirements as opposed to the requirements being based on the CONOPs. Similarly, in the Roberts survey, 18% of respondents said that CONOPs on programs they worked "were not completed until after the requirements were complete" (Roberts, 2008, p. 28). With the CONOPs document seen as critical to defining and employing a proposed system, why is it that the CONOPs is often missing or developed as an afterthought?

Barriers to Effective CONOPs Use

Our research indicates that there are four barriers to the use of CONOPs throughout the acquisition lifecycle. These barriers include:

1. Definition and Purpose. There is variance in the term used to describe a CONOPs document, as well as an inconsistent application of the term. Often, this results in misunderstanding of what a CONOPs is, how it is used, and what type of information it should contain.
    1. Targeted Audience. Closely tied to the variance in definition and use, the intended audience of the CONOPs document is unclear.
    2. Timing and Placement in the Acquisition Development Lifecycle. There is confusion regarding to what phase of development a CONOPs applies.
    3. Comprehensive View and Consistent Involvement by Stakeholders. Many forms of the CONOPs document are just a small subset of what system development really needs– these subsets do not incorporate a complete view of the system. Additionally, many of these CONOPs are by various authors using different stakeholder sets.

Definition and Purpose. Our study detected considerable variance in the application of the term "*CONOPs*." As a result of the variance, misunderstanding and purpose are major barriers to the use of CONOPs. This variance makes it unclear what a CONOPs is, how it should be used, by whom it should be used, and when it should be used.

Military Concepts and System-Level CONOPs. Within the Department of Defense (DoD) there are higher-order and lower-order CONOPs. Higher-order CONOPs, include Capstone, Institutional, Operating, Functional, and Integrating concepts, which, in descending order, become more narrow in scope and more detailed by applying to a smaller mission set (for clarity we will term higher-order CONOPs "military concepts" for the remainder of this document). These concepts "describe the conduct of military action at the operational level of war" (Schmitt, 2006, p.1). Military concepts are derived directly from military strategy and provide a premise for the future capabilities the military will need. According to Joint Publication 1-02, these CONOPs are a "verbal or graphic statement that clearly and concisely expresses what the joint force commander intends to accomplish and how it will be done using available resources" (JP1-02, p. 114).

The materiel capabilities needed to achieve the goals of the military concepts are described in lower-order, system-level CONOPs. The system-level CONOPs band includes capability- specific CONOPs. According to the Institute for Electrical and Electronics Engineers (IEEE), these CONOPs are a "user-oriented document that describes system characteristics for a proposed system from the user's viewpoint" (IEEE, 1998, p. i).

Additional CONOPs Variance. Within both military concepts and system-level CONOPs there are several types of documents–all of which are called either "CONOPs" or some variation of the term "CONOPs." Joint Service and Component Service publications[45] have already defined and documented the different types of military concepts (i.e., institutional, operating, etc). However, the different types of system-level CONOPs are less

---

[45] Types of military concepts are defined in publications such as the Chairman of the Joint Chiefs of Staff Instruction 3010.02 "Joint Operations Concept Development System"; the Air Force Policy Directive 10-28 "Air Force Concept Development"; CONOPs TO DOCTRINE: Shaping the Force From Idea Through Implementation (Fleet Forces, 2004).

well defined and vary from publication to publication. Adding to the confusion is the fact that each CONOPs document may include similar or dissimilar information.[46]

The Perceived Purpose of the CONOPs Document. As discussed above, the purpose of a CONOPs can range from describing aspects of a military operation (military concept) to describing characteristics of a specific system (system-level CONOPs). But even within system-level CONOPs, the purpose can range from describing all system attributes, system stakeholders, and system tasks, to focusing solely on the employment of the system. Examples of different CONOPs document names and associated purposes are provided in Table 2.

Table 2. Perceived Purposes of CONOPs

Term	Purpose	Reference
User CONOPs	[Defines] basic system requirements. [It] describes what the user wants the system to achieve and the context in which the system will be utilized	Companion & Mortimer, n.d.
System CONOPs	[Defines] how the system will actually be used and provides insight into the total system solution for both short-term and long-term requirements. Similar to ANSI/AIAA[47] OCD	Companion & Mortimer, n.d.
ConOps	Provides the user community a vehicle for describing their operational needs that must be satisfied by the system under development	Jost, 2007, p. 14
CONOPs	Provides "a capability description (what an operational unit does) or a prescription of what should be done."	Nelson, 2007, p. 2
ConOps	"[Transforms] the allocated what to the how and so completes a chain all the way to an instantiation…of the system that enables capabilities."	Nelson, 2007, p. 2
Concept of Employment	Description of "how a weapon system will be [used] in a combat environment"	Ask a Professor (AAP, 2009
CONOPs	[Provides] the vision and intent for how the system should work within an operational environment in an easy to read format.	Daniels & Bahill, 2004, p. 306, sec 4.1
Use-Case Scenarios	Similar to a CONOPs (see preceding CONOPs definition) but less ambiguous and therefore can be used directly for extracting requirements in an unambiguous way.	Daniels & Bahill, 2004, p. 307, sec 4.1

[46] Daniels and Bahill (2004) point out that "CONOPs documents are rarely consistent in content, detail, and format" (para. 4, p. 307).

[47] ANSI/AIAA is an abbreviation for the American National Standards Institute/American Institute of Aeronautics and Astronautics. ANSI/AIAA standard G-043-1992, *Guide for the Preparation of Operational Concept Documents*, discusses information that relates to system operational concepts and describes "which types of information are most relevant, their purpose, and who should participate in the effort" (ANSI/AIAA, 1993, abstract).

Targeted Audience. Just as the name for and content of a CONOPs document varies, so may the intended audience. Currently the audience is dependent on who is writing the CONOPs, what type of CONOPs is being written, and its intended placement within the acquisition timeline. The CONOPs can be written to speak to all communities or can focus on individual communities, such as engineers, customers, test agencies, or decision authorities. A CONOPs that only speaks to a specific community may be problematic in its potential to lack sufficient detail required by the unaddressed audience.

Timing and Placement in the Acquisition Lifecycle. The placement of the CONOPs refers to the insertion point of the CONOPs into the larger acquisition activity. The "input" of the CONOPs into the acquisition process is dependent upon the identified purpose of the CONOPs document. For instance, a military concept will occur earlier in the acquisition lifecycle than a system-level CONOPs.

With the relative importance of the CONOPs widely understood (see Table 1), it is difficult to envision proceeding through the acquisition lifecycle without some form of the CONOPs document. To that end, we believe that although not necessarily called a CONOPs, and not in a neat and integrated package, critical elements of the CONOPs are occurring in an ad-hoc manner across the acquisition timeline. Nelson echoes this belief, stating that, "[the] main reason we overlook the central role and scaling of the ConOps is that we give different names to the same thing at different scales" (Nelson, 2007, p. 9).

CONOPs Placement According to Official Literature. In DoD acquisition documents, such as the DoD 5000 series and JCIDS documents, CONOPs are identified as an input to a larger acquisition process. In these documents the term "CONOPs" on its own usually refers to a military concept. As such, it is placed as a precursor to system concept selection. Figure 1 provides an organizational construct for the position of the military concept to the JCIDS timeline. This construct depicts the hierarchy of military concepts as a linear progression from broad to most narrow focus (left to right). These military concepts then feed into the JCIDS-Acquisition process as a basis for the Capabilities Based Assessment (CBA).

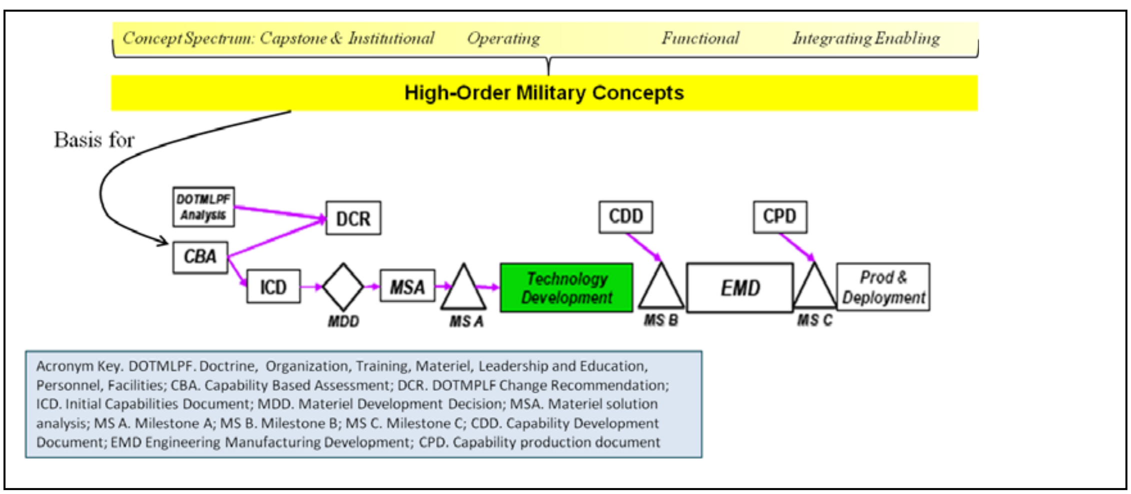


Figure 1. Relationship of the Military Concept to JCIDS-Acquisition
(Modified from JCIDS, 2009)

The Manual for the Operation of JCIDS references at least two more “types” of CONOPs. The first of these is found in Enclosure B of the JCIDS Manual. Here, the reference is to a “sustainment CONOPs” (JCIDS, 2009, p. B-B-5). The manual states that as a sustainment key performance parameter (KPP) metric, the sustainment CONOPs should be traceable to the system’s Initial Capability Document (ICD) and Capability Development Document (CDD). This implies that the sustainment CONOPs is an input to the acquisition lifecycle process after the ICD and CDD have been written[48].

A second reference to CONOPs is made later in Enclosure B. This time, the reference is to a “CONOPs… for the system”. Contextually, this CONOPs for the system, or *System CONOPs*, provides documentation of “a comprehensive analysis of the system and its planned use, including the planned operating environment, operating tempo, reliability alternatives, maintenance approaches, and supply chain solutions” (JCIDS, 2009, p. B-B-6). Based on this description the JCIDS System CONOPs is similar to the latter half of the IEEE ConOps. This System CONOPs is an input to the JCIDS process post Milestone B, upon program initiation.

JCIDS also acknowledges the analysis of alternatives (AoA) that is part of the larger acquisition process. The AoA is a precursor to the Milestone A decision. The analysis of alternatives, though likely more detailed than what is included in the CONOPs, corresponds to the IEEE ConOps Alternatives section, which discusses system alternatives considered but not selected. Figure 2 provides a rough illustration of the relationship of these documents (to include the concept of employment (COE), which is recognized by DAU as an input to capability development documents) to JCIDS-Acquisition decisions.

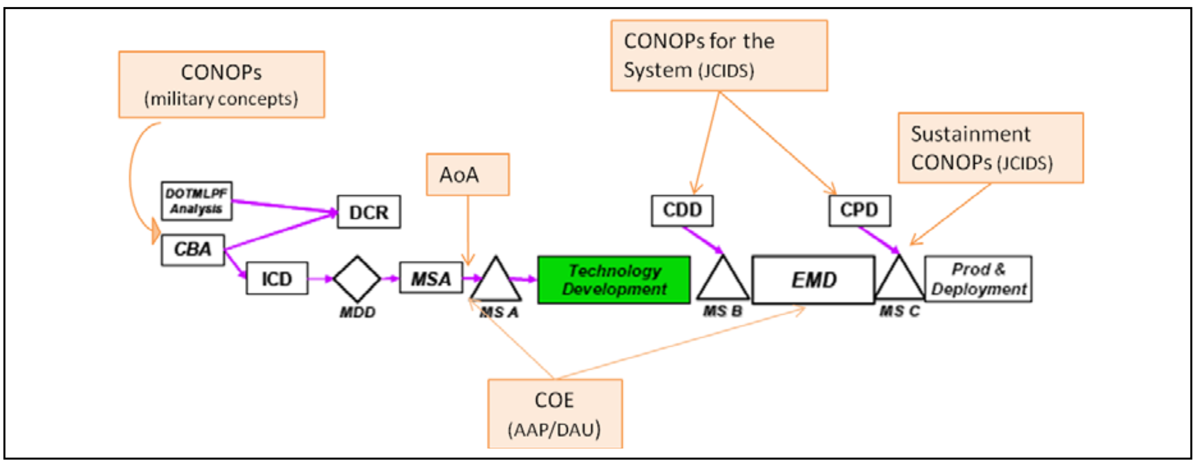


Figure 2. Relationship of Official CONOPs-Related Documents to JCIDS-Acquisition Activity

---

[48] Though not explicitly defined in the manual, contextually the sustainment CONOPs is a concept of operations specific to maintenance approaches and supply chain solutions. This definition makes the implied position of the sustainment CONOPs somewhat confusing, because maintenance and sustainment concepts communicate desired sustainment instructions that inform system design and planning. The sustainment CONOPs will likely be more detailed post milestone B and C, but per the IEEE format, should be written, if even in an immature state, with the original system-level CONOPs.

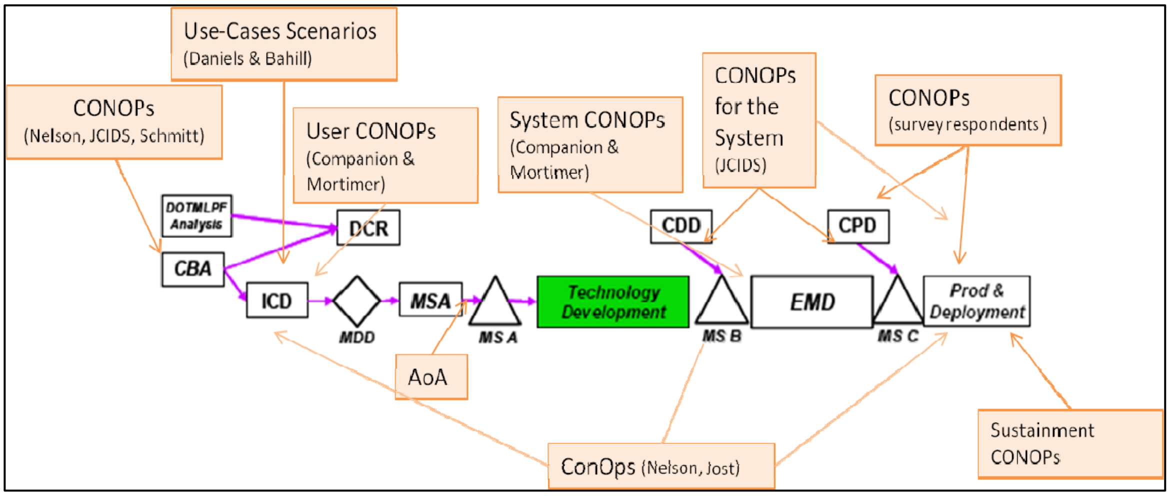


Figure 3. Relationship of Unofficial CONOP-Related Documents to JCIDS-Acquisition Activities

In addition to documents described in the JCIDS manual, our research revealed that there are many other documents in use that serve as inputs to the acquisition process. Such inputs include ConOps (described by Nelson), use-cases (as described by Daniels & Bahill, 2004), user CONOPs, and system concepts (see Table 2). The relationship of these inputs to the JCIDS-Acquisition timeline is illustrated in Figure 3.

This mapping of CONOPs documents to the acquisition lifecycle suggests that CONOPs documents are developed throughout the acquisition lifecycle.

Comprehensive View and Consistent Involvement by Stakeholders. As illustrated above, many separate CONOPs documents are written. A risk to proper use of these CONOPs is that these various CONOPs are independently authored by various individuals or groups that may have different perspectives on the system and on system objectives. Without a single, integrated, system-level CONOPs to draw from, requirements may be unintentionally derived from multiple sources that may or may not include a complete understanding of system uses. This can create “[different] and potentially conflicting views of system use [that] will result in a system that only partially meets the user’s expectations” (IEEE, 2008, para. 3.3.3, p. 23).

Additionally, breaking the CONOPs into smaller non-integrated documents runs the risk of reducing stakeholder coordination. This practice also reduces the comprehensiveness of the stakeholder input, which in-turn degrades the completeness of perspectives and resulting system requirements.

Key to Effective Use of CONOPS in the Acquisition Life Cycle: The Integrated CONOPs

Despite the occurrence of the various types of CONOPs documents, Nelson (2007) argues that although many documents go by the name CONOPs, there is only one “true”

ConOps[49] and it is the ConOps described in the IEEE[50] standard 1362-1998. Nelson goes on to state that the power of the IEEE ConOps comes from its comprehensive assessment of both the "what" (system identification) and the "how" (system employment). In our assessment, the IEEE format also includes the "why" in its section titled *Justification*. Traceability to the why, or *justification*, is an important factor in maintaining system validity and verification.[51] Unique to the IEEE format is its emphasis on describing not only the desired capability and end-state, but also the current capability and situation. This end-to-end emphasis provides a logic trail from original need to capabilities pursued.

Integration of Individual Inputs

Because the many types of CONOPs-related documents appear to span the entire lifecycle of the system, we wondered how these individual CONOPs would relate to the IEEE ConOps. To find out, we delineated the relationship of the individual CONOPs documents to sections of the IEEE ConOps (Figure 4). To conduct this assessment, the content of each of the CONOPs documents used as an input to the acquisition process was analyzed. This content was then compared to the content in each section of the IEEE ConOps to identify similarities. Arrows are provided from CONOPs documents to applicable IEEE ConOps sections to show a relationship between the content.

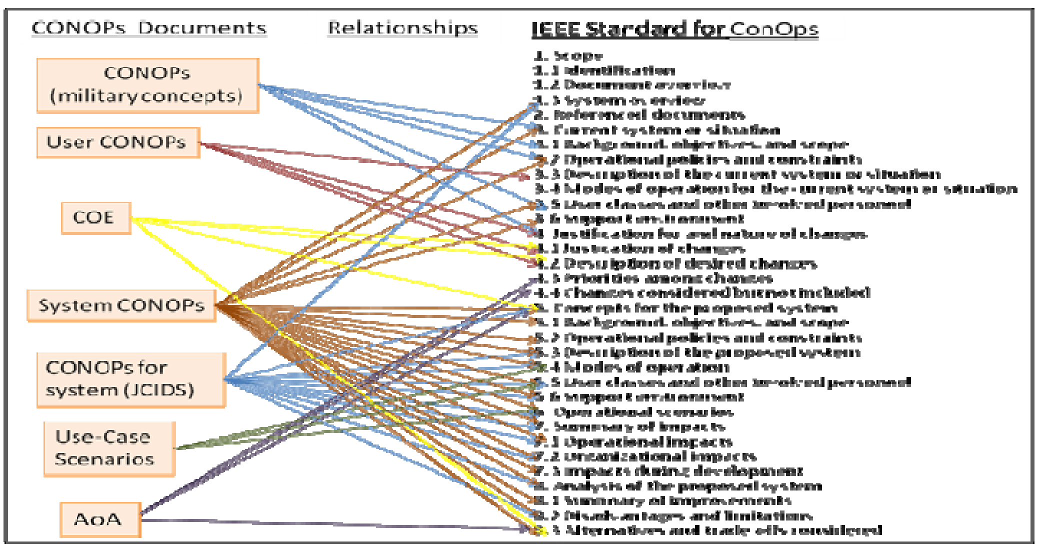


Figure 4. Relationship of CONOPs Documents to IEEE

An Example of a CONOPs Document–IEEE ConOps Relationship. The guiding military concept provides insight into the operational environment, the scope of the mission set, and the need for the system. Although this concept does not provide an exhaustive list of system user classes, it will provide insight into an initial group of potential

---

[49] Nelson uses the IEEE abbreviation for ConOps to distinguish it from other forms of concept of operations documents, which are often abbreviated as "CONOPs" (Nelson, 2007).

[50] IEEE, pronounced (I-triple –E) IEEE is recognized as a leading institution in systems and system standards. Their format for ConOps documents (IEEE, 1362-1998) is comprehensive and is used by many agencies/ organizations.

[51] In their paper on famous failures, Bahill and Henderson note that "[system] validation requires consideration of the environment that the system will operate in" (2005, p. 3).

system stakeholders. Therefore a relationship is shown between the military concept and the IEEE sections: Current Situation, Background and Scope, Policies and Constraints related to the current situation, User Classes, and Justification for Change.

Identification of these relationships confirmed that, while not necessarily an integrated ConOps such as IEEE, elements of the IEEE ConOps are being utilized in the acquisition process via the many, currently occurring CONOPs documents. All IEEE ConOps sections are addressed in the currently occurring CONOPs-related documents with the exception of a detailed explanation of the modes of operation for the current system (this would include modes for legacy systems currently in place) and administrative sections such as referenced documents and document overview. This means there may be an opportunity to integrate elements of each of these documents into an integrated CONOPs, such as the IEEE ConOps.[52]

The Value of an Integrated CONOPs over the Current Way of Business

Although already occurring independently across the lifecycle, integrating individual CONOPs documents into a single CONOPs document has potential to increase both traceability and continuity.

Traceability. According to IEEE, traceability is "a key tool for ensuring that the system developed fully meets the needs and requirements defined by the user" (IEEE, 2008, para. 4.2, p. 38). An integrated system-level CONOPs resolves, or at least mitigates, the problem of conflicting system views and partially met requirements by using the same resources to create a more complete view of the problem, the proposed solution, the user community, and the intended uses. This pooling of information allows stakeholders an opportunity to recognize requirement needs and contradictions that are otherwise overlooked.[53]

Continuity. Both IEEE and ANSI identify a purpose of the CONOPs as a means by which to communicate system characteristics in such a way that is understandable by all system stakeholders.[54] Continuity is an enabler of the required communication and stakeholder involvement. Similar to what is seen with traceability, continuity suffers when the integrated system-level CONOPs is broken out into multiple documents.

---

[52] Integrating each of these inputs into a comprehensive CONOPs document does not preclude the use of, or independent improvement of, particular sections. Products, such as an AoA, can continue to stand-alone; in fact AoAs can inform later iterations of the CONOPs document. The point of the integrated CONOPs is not to enforce a rule set–but rather to serve as a means for conducting holistic problem and solution space exploration and for providing a document owned by all stakeholders that clearly and logically expresses the system's characteristics. In this way, the CONOPs serves as the baseline document to which all subsequent documents are loyal.

[53] According to the INCOSE SE Handbook, version 2a (2004) one use of a CONOPs is "[t]o validate requirements at all levels and to discover implicit requirements overlooked in the source documents" (para. 8.2 f, p. 104).

[54] "The CONOPs document is used to communicate overall quantitative and qualitative system characteristics to the user, buyer, developer, and other organizational elements" (IEEE, 1362-1998, p. 1). The CONOPs document "facilitate[s] understanding of the overall system goals with users…, buyers, implementers, architects, testers, and managers" (ANSI/AIAA, 1993, p. 1.)

The Relationship of the Integrated CONOPs to the Acquisition Lifecycle

Current literature provides an illustration of the relationship of military concepts to JCIDS-Acquisition activities (Figure 5). This is in line with our assessment of the relationship of military concepts to acquisition activities (see Figure 1). What appears to be missing though is an illustration of the relationship of the system-level CONOPs to the acquisition lifecycle. If we integrate the many CONOPs documents into a single integrated CONOPs document, what will its relationship to the acquisition lifecycle look like?

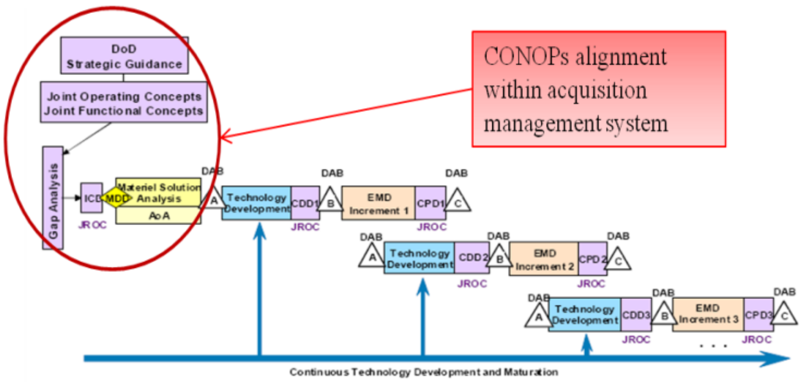


Figure 5. Requirements and Acquisition Process Flow
(Modified from USD(AT&L), 2008)

The streamlined CONOPs-Acquisition lifecycle relationship, or CONOPS continuum, is most easily depicted by building upon a baseline graphic (Figure 6) and gradually adding in additional relationships. Initially this illustration depicts two major inputs to the JCIDS-Acquisition process. The first of these is the higher order military concept which is a basis for a CBA and identifies the context in which the proposed system will operate. System-level CONOPs emerge when the CBA process identifies capability gaps for which a materiel solution is the preferred solution. This results in a relationship between higher-order military concepts and acquisition and between higher-order military concepts and system-level CONOPs (which we will term simply “CONOPs”). As illustrated in Figure 6, military concepts drive CBAs and Doctrine, Organization, Training, Materiel, Leadership, Personnel, Facilities (DOTMLPF) changes and are the context for CONOPs, which must always support the activities outlined in the military concepts.

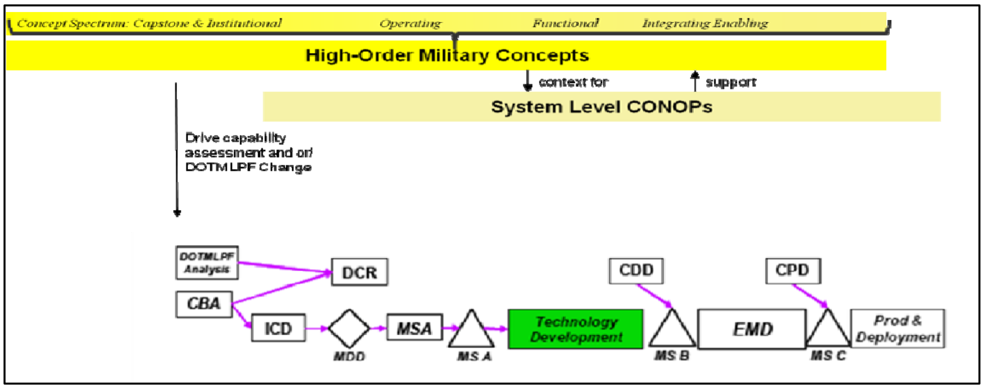


Figure 6. Concept-CONOPs-JCIDS-Acquisition Relationships (CONOPs Continuum)

Higher-order military concepts are the basis from which CBAs and systems in development are derived. As such it is imperative that the vision, mission, and goals of military concepts are valid. Invalid or inaccurately assessed military concepts can result in faulty CONOPs and ineffective systems. Therefore, the exercise and evaluation process that validates the military concept is an equally essential part of successful CONOPs development and, ultimately, of successful system development. Figure 7 depicts the relationship of experimentation to military concepts, with military concepts driving experimentation, which informs or validates the military concept. Military concepts that are invalidated should be changed and retested. Once validated, the military concepts then drive CBAs or DOTMLPF changes.

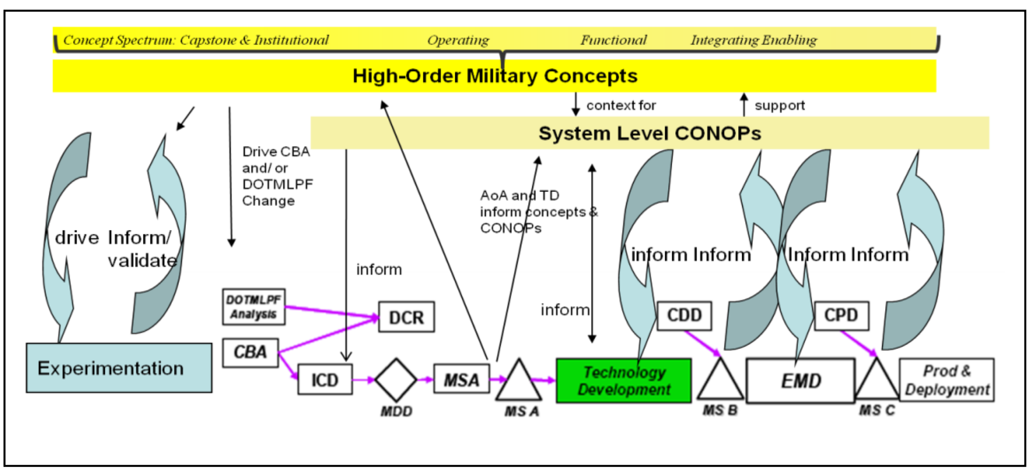


Figure 7. Experimentation as Part of the CONOPs Continuum

Because CONOPs are the basis for system requirements, they also interact with the JCIDS-Acquisition processes. Initially, the CONOPs informs the ICD. As the system progresses through the acquisition lifecycle, events in system develop should inform the CONOPs. This process of revisiting and updating the CONOPS will help ensure it remains relevant. This ongoing cycle of informing and updating is illustrated in Figure 8.[55] The importance of the figure resides in its simplistic illustration of the interconnectedness of many activities. The picture now shows a series of ongoing interconnected processes each reliant on and influencing the other, versus strict swim lanes of disparate processes.

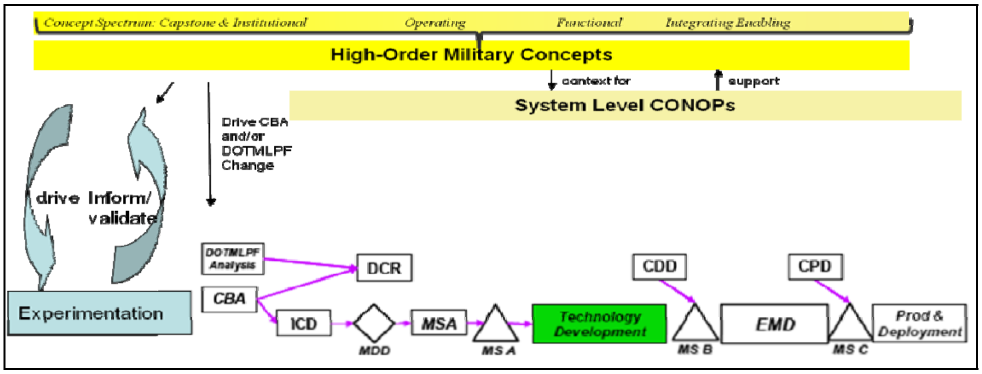


Figure 8. Interconnectedness of CONOPs and Acquisition Activities

CONOPs Maturation and Phases

Ideally, the CONOPs should be updated throughout the acquisition lifecycle, such that as the system matures, the CONOPs increases in specificity. We have identified specific phases of CONOPs maturity each of which coincides with events in the acquisition lifecycle.

CONOPs (Initial Phase). Initially, the CONOPs describes the proposed system as a 'black box' and in its most ideal form (i.e., all user desired capabilities). This initial phase of the CONOPs will be used to guide the development of the ICD and serves as a basis for system requirements.

[55] The figure also shows the independent role of the AoA. The AoA informs both the higher and system-level concepts. The AoA has potential to influence DOTMLPF solutions that impact the way a Service fights, trains, and equips. The AoA informs the system-level CONOPs section on alternatives considered.

CONOPs (Discovery Phase). The suggested update cycle is triggered by specific events in the acquisition lifecycle. Initially, the CONOPs informs the Technology Development[56] (TD) phase by communicating desired capabilities, for which technology must be developed. Likewise, the TD process informs the CONOPs, by revealing actual technological possibilities. As TD progresses and technological possibilities become more evident, the CONOPs document should be updated to reflect *actual* capability. This activity will ensure that the user gets the system expected and that the system, though not as initially envisioned, still meets the operational need(s) described in the military concepts.

The updated CONOPs and the ICD are the foundation for the CDD requirements generation process. Following the CDD and Milestone B, the system enters the Engineering, Manufacturing, and Development (EMD) phase.[57] Results from the EMD activities bring additional clarity to the system's operational limitations and advances. Therefore, EMD results should further inform the CONOPs such that it can again be updated to reflect actual system capability. Again, the updated CONOPs will be used as the basis for the requirements captured in the Capability Production Document (CPD). As with previous updates, the updating process will continue to maintain traceability between the user's expectations and the operational mission the system supports.

CONOPs (Employment Phase). Shortly after Milestone C system prototypes enter into low rate initial production (LRIP). LRIP models will generate user feedback, which will further inform the CONOPs. LRIP feedback provides the information needed to fully understand how the final system can and will be used. This feedback should be incorporated into a final version of the CONOPs. Throughout the updating cycle, the CONOPs must be compared to the higher order military concept(s) it supports to determine that it still supports the goals, missions and activities of the military concepts. The continuous review of the military concepts and CONOPs relationship is particularly important in long-lead time acquisition programs; over time introduction of new systems, new threats, and new political environments can change the battle field to the point that the system no longer serves a valid mission set.[58]

The CONOPs maturity phases align with major phases of the acquisition cycle (i.e. MDD, TD, EMD, and deployment and employment). This results in three distinct phases of the CONOPs document. We have termed these phases: Initial, Discovery, and Employment, the names of which correspond to the level of system understanding discussed above. A graphical representation of these phases as they relate to the JCIDS-Acquisition timeline is provided in Figure 9.

---

[56] During the technology development phase, "technologies are developed, matured, and tested" (Schwartz, 2009, p. 9)

[57] During the EMD phase "various subsystems are integrated into one system and a development model or prototype is produced" (Schwartz, 2009, p. 10).

[58] Although we have identified specific drivers of a CONOPs review, the idea of updating CONOPs is not new. Most guiding documents agree that the CONOPs is a "living document" (ANSI/AIAA, 1993; IEEE Standards, 1998; Daniels & Bahill, 2004). Even still, in a survey of systems engineers and lead systems integrators, respondents indicated that "[over] 50% of [CONOPs] were not updated throughout the entire program lifecycle and of those 49% were only updated through preliminary design review. (Roberts, 2008, p. 28).

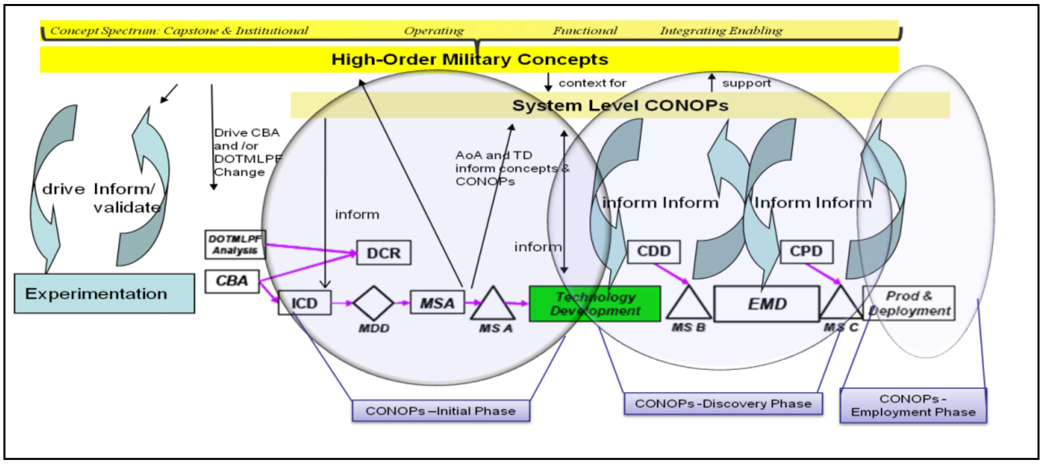


Figure 9. CONOPs Phases as Related to JCIDS-Acquisition

The integrated, maturing CONOPs provides a mechanism for tracing system elements, such as requirements, perspectives, decisions, and solutions.

Summary

Undeniably, the system-level CONOPs has the ability to influence the success of activities across the entire acquisition lifecycle. To fully realize the benefit of these CONOPs we must first resolve outstanding issues related to barriers of the CONOPs use. First, we must work as a community to promulgate a single, agreed-upon definition for the term CONOPs. Secondly, we must work to combine existing CONOPs documents into an integrated and comprehensive document that speaks to many audiences. The integrated CONOPs will reduce the risks of inconsistent and unmet requirements by ensuring effective collaboration by stakeholders throughout the development life cycle. Once the CONOPs is created, we must remember its alignment with military concepts and acquisition activities and the influence each of these has on the other. Finally, we must remember to revisit the CONOPs and allow it to mature over time. Although a potentially demanding and lengthy process, use of CONOPs will amplify the rapid and cost effective delivery of *usable* systems that meet warfighter needs.

References

American Institute of Aeronautics and Astronautics (ANSI/AIAA). (1993). *Guide for the preparation of operational concept documents. American national standard*. Washington, DC: American Institute of Aeronautics and Astronautics.

Ask a Professor (APP). (2009, May 20). *Question & answer detail program management*. Retrieved November 2009, from https://akss.dau.mil/askaprof-akss/qdetail2.aspx?cgiSubjectAreaID=9&cgiQuestionID=28681

Bahill, T.H., & Henderson, S.J. (2005). Requirements development, verification, and validation exhibited in famous failures. *Systems Engineering*, 8 (1), 1-12. Retrieved October 2009, from www.sie.arizona.edu/sysengr/publishedPapers/famousFailures.pdf

Boardman, J., & Sauser, B. (2008). *Systems thinking: Coping with 21st century problems*. Boca Raton, FL: CRC Press Taylor & Francis Group.

Chairman of the Joint Chiefs of Staff Instruction. (2006, January 27). *Joint operations concepts development process (JOpsC-DP),* (CJCSI 3010.02B). Washington, DC: Government Printing Office.

Companion, M., & Mortimer, C. (n.d.). *Designing for change: A modeling and simulation system approach*. Arlington, Texas: Raytheon Systems. Available at www.link.com/pdfs/itsec2.pdf

Daniels, J. & Bahill, T. (2004, July 7). The hybrid process that combines traditional requirements and use cases. *Systems Engineering*, 7 (4), 303-319. Retrieved October 2009, from www.sie.arizona.edu/sysengr/publishedPapers/hybridProcess.pdf

Edson, R. (2008). *Systems thinking. Applied. A primer*. Arlington, VA: Analytic Services, Inc.

Fleet Forces Command. (2004). *CONOPs to doctrine: Shaping the force from idea through implementation* (Pre-Decisional-Draft Working Papers). SEA TRIAL Presentation to the Corps of Naval Attaches. Retrieved June 2009, from www.ndia.org/Divisions/Divisions/.../Content/.../CONOPS_101.pdf

IEEE Press/Standards Information Network. (2008, November 4). *Guide for implementing IEEE Std 1512™ using a systems engineering process*. New York: Institute of Electric and Electronics Engineers. Retrieved from http://standards.ieee.org/standard press/

IEEE Standards Board. (1998). *IEEE standard 1362-1998, IEEE guide for information technology—System definition* (Concept of Operations (ConOps) Document). Standard. New York: Institute of Electric and Electronics Engineers.

International Council on Systems Engineering (INCOSE). (2004). *Systems engineering handbook: A "what to" guide for all SE practitioners* (INCOSE-TP-2003-016-02) (Ver. 2a).

Joint Capabilities and Development System (JCIDS). (2009). *Manual for the operation of the joint capabilities integration and development system*. Washington, DC: Author.

Joint Publication 1-02. (2001, April 12, as ammended through October 2008). *Department of Defense dictionary of military and associated Terms*. Washington, DC: Author.

Jost, A. (2007, October). ConOps: The cryptex to operational system mission success. *Cross Talk: The Journal of Defense Software Engineering*, *20*(10), 13-16. Retrieved from http://www.stsc.hill.af.mil/crosstalk/2007/10/0710Jost.html

Nelson, G. (2007). The ConOps in a self-similar scale hierarchy for systems engineering (Paper No. 69). In *Conference on Systems Engineering Research.* Hoboken, NJ.

Roberts, N. (2008). *An analysis of concept of operation development* (Master's Thesis). Hoboken, NJ: Stevens Institute of Technology.

Schmitt, J. (2002, December) *A Practical Guide for Developing and Writing Military Concepts*. (Defense Adaptive Red Team Working Paper). McClean VA: Hicks and Associates, Inc. Retrieved from http://www.au.af.mil/au/awc/awcgate/dod/dart_guide.pdf

Schwartz, M. (2009, July 10). *Defense acquisitions: How DoD acquires weapon systems and recent efforts to reform the process*. Retrieved September 2009, from www.fas.org/sgp/crs/natsec/RL34026.pdf.

Turner, R., Verma, D., & Weitekamp, M. (2009). *The next generation air transportation system (NextGen)*. Hoboken, NJ: School of Systems and Enterprises, Stevens Institute of Technology.

USD(AT&L). (2008, December 8). *Operation of the defense acquisition system* (DoD Directive 5000.02). Washington, DC: Author.

US Air Force. (2003, September). *Air Force policy directive 10-28*. Air Force Concept Development.

Appendix A. Survey

Questionnaire

The following questionnaire is issued to gain insight into current acquisition processes and to understand perceived shortcomings (if any) and suitable fixes to the identified shortcomings. The ultimate goal of this research is to contribute to efforts to provide the warfighter with needed capabilities in a timely and cost-efficient manner.

Answers to this questionnaire are non attributable, meaning answers provided will not be credited to any particular respondent. Furthermore, prior to being included in any published document, any/all answers will be generalized such that specific programs, offices, and/or systems under development will not be identifiable. Your honest answers are greatly appreciated. Respondents who would like a copy of our research end product can request so by emailing me at jaime.frittman@anser.org.

1a. In how many programs of record do you currently participate, or have you previously participated (an estimate is “ok”)?

1b. What is or has been your role in these programs?

---

2. How do you define the term *concept of operations* (CONOPs)?

---

3. In your personal opinion, what is the role/purpose of a CONOPs?

4a. Have you ever written a CONOPs?

4b. If so, what type of content did you include in the CONOPs?

4c. Did you use a certain standard or prescribed template? If so, which one?

4d. Was the CONOPs updated throughout the development cycle?

4e. Do you believe the CONOPs was properly utilized, or underutilized?

5. How important do you think a CONOPs is to the successful development and employment of the system to which it applies?

6. Off the top of your head, can you think of any shortcomings related to CONOPs as they relate to current systems under development? Please describe these shortcomings.

The Illusion of Certainty

Grady Campbell—At CMU's Software Engineering Institute (SEI), Grady Campbell identifies, develops, and transitions improvements in the process and practices of software acquisition and engineering. In the early 1990s, Mr. Campbell was responsible at the Software Productivity Consortium for conceiving and developing the first comprehensive software product line methodology. Subsequently, he was an independent consultant in software product lines and a Visiting Scientist to SEI's Product Line Practices Initiative. Other highlights in his 40 years of experience include a Naval Research Laboratory Software Cost Reduction project and a project to build an application generation environment based on adaptable software.

Grady Campbell
CMU Software Engineering Institute
4301 Wilson Blvd., Suite 200
Arlington, VA 22203
703-908-8223
ghc@sei.cmu.edu

Abstract

Acquisition policy and, even more so, acquisition practice today presumes that certainty is key to success, and that uncertainty or delays in achieving certainty regarding user needs or solution approach will necessarily impede progress. This means that when uncertainty arises during an acquisition effort, the natural response is to make decisions that resolve this uncertainty. Uncertainties arise for various reasons, such as poorly understood, conflicting, or changing needs. If, under pressure to maintain progress, an acquisition effort makes decisions to resolve these uncertainties without sufficient information, expertise, or deliberation, they are really only creating an illusion of certainty; in a practical sense, the uncertainty still exists. This artificial certainty then leads to flaws such as insufficient detail concerning specific needs or premature limiting of solution options. A new approach to acquisition is needed that recognizes that hiding uncertainty is detrimental to success. Systematically exposing uncertainties will be beneficial toward making acquisitions more flexible, cost-effective, and responsive to changing needs.

The Requirements Challenge in Acquisition

The purpose of the acquisition system is to acquire products that provide users in an enterprise (e.g., warfighters) with capabilities needed to perform their mission effectively and efficiently. To this end, user needs must be understood in terms of opportunities for improving the enterprise's operational systems. Based on this understanding, a product must then be conceived/identified, acquired/developed (engineered and manufactured), deployed, and sustained/evolved. As long experience has shown, many challenges arise in trying to determine actual user needs and achieve a satisfactory product that properly addresses those needs. The essence of those challenges is achieving a balance among needed capabilities, enabling technology, cost, and timeliness in providing a product.

The beginning of the acquisition life cycle is the identification of user needs and technology opportunities that suggest the potential for improved capabilities. These needs are progressively refined, elaborated, and reviewed to create specifications of requirements in increasing levels of detail, feeding into efforts that interpret those requirements to create a conformant product. Again, proper understanding of actual needs in sufficient detail to permit acquisition of a product that meets those needs is a significant challenge, but, further, needs to be met by a product are not static but continue to change. Future sustainability and improvement requires an acquirer and developer of a product to understand not only existing needs and technologies but how those needs and technologies are likely to evolve in the future.

What experience and numerous studies over the years by the Department of Defense, Office of Management and Budget, Government Accountability Office, National Academies, and industry groups, have suggested is that properly determining needs and expressing these as requirements for product acquisition is difficult and prone to inflexibility and error. For example, quoting from the Defense Science Board Report (2009): "Today, 'requirements' are used to define capability needs, implying that nothing less than a specified set of criteria is sufficient. Instead, a more prudent answer is to buy the best capability affordable, in the quantity desired, and fielded in as timely a manner as possible." However, even this opinion fails to fully address how the acquisition approach could be improved so that products would better address both current and future needs. The following discussion proposes the notion that premature decision making, leading to only an illusion of certainty, is a factor in the requirements problem and that a greater awareness and explicit accommodation of uncertainty throughout the acquisition process would be beneficial.

Understanding the Concept of Requirements

"Requirements" is a term that everyone understands on an intuitive level but it can specifically mean many different things. In acquisition policy (USD (AT&L), 2008) requirements is used variously to mean:

- Capabilities needed by a community of users (e.g., mission, user, or capability *requirements*)
- Rules that must be followed in performing acquisition activities (e.g., statutory and regulatory *requirements*)
- Criteria against which the acceptability of a product development effort will be evaluated (e.g., program *requirements*)
- A specification of the expected behavior of a product being acquired (e.g., product, operational, or system *requirements*) (i.e., the guidance that product developers are given to know what to build or to describe what has been built)

In a general sense, all of these uses are consistent but the practical implications of each for acquisition differ substantially. In particular, user requirements, program requirements, and system requirements are all different expressions of the single notion that a product is needed that will allow users to perform their mission more effectively. For an acquisition to be successful, all of these expressions must be consistent.

In fact, however, this presents a dilemma: acquisition rules require that an acquisition program can proceed only after its requirements have been approved. Program requirements are derived based on user requirements and are the basis for defining product requirements. With respect to an envisioned or existing product, its requirements can be thought of as a model of the observable behavior that the product must exhibit to provide the capabilities needed by users. If user requirements are inaccurate, this will undermine program and product requirements unless there is a means for modifying them during the course of product development.

What the experience of many people suggests, supported by recurring government and industry studies, is that requirements at any level of detail are often flawed: incomplete, inaccurate, misunderstood, and prone to unforeseen change. The best means we have for mitigating these flaws is systematic iteration through all aspects of product development, allowing for the progressive refinement of a shared understanding of needs, constraints, potential solutions, and tradeoffs. If acquisition policy or practice dictates that requirements at some point early in the acquisition process must be viewed as complete and immutable, it is likely that resulting products will both embody difficult-to-correct flaws and fail to keep up with changing needs.

The acquisition system today seems to encourage, and practitioners conform to, the idea that a proper acquisition depends on achieving certainty, in requirements and in the cascade of subsequent decisions that flow through the acquisition process. What they actually achieve is the appearance of certainty but such certainty can in fact be an illusion built upon premature, inadequately reasoned decisions, inadequate understanding of needs, and failure to account for changing needs, technology, and operational context. Explicit recognition and accommodation of uncertainties is a way around this dilemma that will help programs avoid commonly experienced cost overruns, schedule delays, and product defects, while supporting concerns for proper accountability.

Causes of Uncertainty in Requirements

In thinking about the nature of requirements, we can easily identify several potential sources of uncertainty. To properly understand and specify requirements, these need to be exposed, analyzed, and documented with rationale:

- Incomplete knowledge. User needs are usually specified by people who are knowledgeable in the mission of the enterprise and how it currently works. However, they often have only limited knowledge of how those needs may be realized in solution products, how different aspects may interact in a solution, or how new solutions could change the way the enterprise works; as a result, they may express needs in ways that unintentionally seem to limit the potential solution. Furthermore, there are usually aspects of user needs about which even experts disagree. Because no one can have complete knowledge of all aspects of any endeavor, it is likely that any description of user needs will mask areas of uncertainty or disagreement. Customers may recognize that this uncertainty exists but, lacking a proper awareness of the need and means to communicate the variety of alternatives that they see, they may instead make a reasoned but ultimately arbitrary choice.
- Imprecise understanding of needs. While customers may be competent to define user needs, developers are likely to lack the same depth of understanding. Experts in a field may share assumptions, concepts, and terminology that enable them to describe needs in simpler terms that acquisition agents or developers with less of that expertise may misinterpret. It may not be apparent that developers have a different understanding until a product exists and its behavior can be observed in use. Needs are often better understood after potential solutions have been developed and comparatively evaluated, preferably with exposure to knowledgeable users, leading then to being able to define better requirements.
- Differing needs among users. Users doing similar jobs may in fact legitimately not have exactly the same needs. If requirements characterize all users doing

similar jobs as having the same needs, the developer may create a product that properly meets the needs of only some users. A product that imposes a particular viewpoint on all such users will make some of those users less effective.

- Changing needs. Needs change over time because of changes in mission, operational context, and technology. Defining needs only from the perspective of a single point in time ensures that these are inaccurate with respect to other times. Framing needs as fixed without consideration of potential change over time imposes uncertainty on what is potentially predictable change that may be better accommodated by developers if known.

Why Apparent Certainty in Requirements May Be an Illusion

By the time a product is deployed, its actual ("as-built") requirements have been effectively determined. Still, although in a practical sense there can be no actual uncertainties about the behavior of a deployed product, there may not be a complete and accurate specification of what those requirements are. In that sense, there may be no one who can be certain about all aspects of the product's behavior.

Although acceptance of a product depends on compliance with acceptance criteria usually expressed in the form of requirements, the same problem can exist for those: there may be unacknowledged uncertainties that have been improperly resolved. In a process in which encountered uncertainties are simply decided away, without proper identification and systematic analysis of factors and tradeoffs, and not documented with rationale, it is likely that some uncertainty still exists relative to actual current or future needs. This same argument applies as the reasoning behind particular requirements is traced back through acquisition decision making.

Uncertainty can exist at any level of requirements. The fundamental uncertainty is how does someone determine and communicate what they want or need. That phrase itself reveals a basic issue: how do we distinguish aspects that we must have from what we might like to have from what we would accept. The essence of engineering is identifying and weighing tradeoffs among alternatives but the nature of defining requirements is not only to make a definitive statement about what a customer needs but in doing that to also eliminate unsuitable solutions. When this is done without full knowledge of actual possibilities, potential solutions can be prematurely constrained.

In looking at how requirements are determined, there are many factors that can lead to the various kinds of uncertainty:

- No individual or collection of people will have complete knowledge of all aspects of an existing system and its operational context; requirements are inevitably only a partial description that requires particular expertise for correct understanding.
- It is not possible to communicate all that an individual or collection of people know about a system or how it might be improved; a complete description of what is known would require years to produce and years to consume.
- Among a collection of people, even with similar spans and depths of knowledge in an area, there will be disagreements, some that can be resolved and some that are fundamental; the conventional answer is to insist on achieving agreement, even though the substance of the disagreement may itself provide more accurate insight into a correct solution than either individual or consensus viewpoint.

- Even if people are able to correctly characterize needs at a point in time, needs as well as enabling technologies change over time; requirements that only describe what is needed currently will be incorrect at other times in the future.
- Both natural language and graphical notations are frequently understood differently by different people, particularly when lacking similar expertise; two examples of gaps in communications are between users and developers and between systems engineers and software engineers.

There are other factors in the nature of requirements that can also lead to uncertainties about the needed product:

- Users typically view their needs in terms of being able to accomplish their job and new or improved capabilities that would enhance this; this view is often constrained by inaccurate assumptions about what is possible and what can and cannot be changed.
- When a product built to address users' needs is deployed, it often changes the users' perceptions not only of what their needs are but also of what is possible, leading to their needs "changing" yet again.
- Requirements being only approximate descriptions of needs and constraints on potential solutions may in fact omit information that the customer knows and assumes but that the product developer does not.
- Requirements are frequently not limited to what is absolutely needed but also reflects the customer's perception of what is desirable without distinguishing between these; this in fact often precludes options that would allow the developer to make better tradeoffs in creating a product that is a best fit to purpose within given cost and schedule constraints.

Using Uncertainty in Writing Better Requirements

Parnas and Clements (1986) argue for the ideal of a rational process for the design and building of (software) products. As part of this, they characterize requirements as a definition of the expected observable behavior of a needed product, sufficient to answer developer questions about what is to be built. A way to understand this is to recognize that requirements constitute a *model* of a needed product. From this perspective, requirements should define the capabilities that a product needs to enable for its users.

However, it is not enough to describe a product at a fixed point in time and with uncertainties hidden. In fact, Parnas and Clements argued that a rational process is not usual practice because of incomplete and inaccurate information in documentation caused by underlying issues in how documents are written (e.g., poorly organized, stream of consciousness exposition, poorly and inconsistently written by multiple authors, dispersed repetition of related or conflicting information, confusing and inconsistent terminology, narrowly conceived). In fact, these problems still exist and are often symptoms of false certainty. Authors must produce requirements that appear certain even if uncertainty has not been properly resolved. No means is given to indicate areas of uncertainty, doubt, or likely change that are left to be resolved.

A common effect of resolving uncertainty with arbitrary decisions is to prematurely limit solution options. Not having a means of specifying requirements so as to permit alternative solutions, the customer may instead resort to describing a particular solution that has worked for similar problems in the past. By hiding the uncertainty and precluding further analysis, the developer may be forced to adopt the described solution rather than having the option of developing and evaluating other potentially more appropriate solutions.

A factor in being able to evolve a product as user needs change is the ability to recover the rationale for why requirements are what they are. Some products today include obsolete capabilities and are difficult to change simply because no one is sure why the product does all that it does, why it is built the way it is, and whether any aspects of what it does are no longer needed by users. A natural by-product of focusing on and explicitly analyzing uncertainties is that the rationale for the resolution actually chosen will be documented. By capturing this rationale, we have at least a partial characterization for the product not only as it finally exists but also as it might have been differently done, giving a basis from which to revise the product when needs change.

An Existing Approach for Limited Accommodation of Uncertainty

Some experience already exists with building products with a focus on uncertainty. When an organization has a need to build multiple products, in support of customers having similar but not identical needs, a product line approach provides a process for building a set of similar products from a common base of reusable software, documentation, and test assets (Campbell, Faulk & Weiss, 1990; Clements & Northrop, 2002). This approach depends on identifying precisely the ways in which the needed products will be alike (commonalities) and the ways in which they will differ (variabilities). The techniques for identifying how products will differ and in resolving those differences to create a specific product is similar to the model proposed for identifying and resolving requirements uncertainties in general.

With a product line approach, not all types of uncertainty are addressed but only those related to customers' changing or diverse needs. Uncertainties related to potential changes in customer needs or to needs that differ among customers are systematically identified and formulated as decisions that will be resolved late in the production of each specific product in consultation with the individual customer for that product. The ability to resolve these uncertainties in different ways is systematically engineered as production options. This provides the means to deliver a customized product to each customer and to deliver a revised product to each customer as their needs change. Identifying decisions that encapsulate the implications of diverse and changing customer needs on product requirements for a set of customized products is integral to the concept of a product line. This approach also provides the means to rapidly build alternative solutions to particular customer needs as a means to helping the customer find the best fit to their needs.

A Strategy for Comprehensively Accommodating Uncertainty

A recent National Academies study (*Achieving Effective*, 2009) has recommended that system requirements ("big R") for IT systems be defined strictly and fixed at the mission capabilities level and that more detailed requirements ("little r") continue to be developed and refined throughout the acquisition process. Consistent with the challenges of uncertainty, this study advised that requirements evolve progressively through ongoing interactions with end users and assessments of available technologies. The study alluded to a recent Joint Capabilities Integration Development System policy that prescribed a similar approach.

Uncertainty in requirements is not a “problem” to be eliminated. Recognizing and properly exposing uncertainty is an aid to communicating more effectively about needs and potential solutions. Some uncertainties, once recognized as such, can be resolved through an analysis of alternatives and tradeoffs during the development process; others are inherent aspects of the problem being addressed and require different resolutions over time or for different customers. From experience with product lines, there are good techniques for expressing uncertainty which can guide developers in providing needed flexibility with mechanisms for tailoring and product customization to better accommodate the needs of customers over time.

A strategy comprising three elements will give the means to better expose and resolve or accommodate uncertainties in requirements:

- View product development as a process whose ancillary purpose is the elaboration, refinement, and correction of requirements as initially defined.
- Document all differences and their implications when domain experts have differing views on any aspect of requirements.
- For any aspect of requirements for which there are alternatives, that are the subject of tradeoffs, or that may change in the future, document the rationale for its current realization in comparison with identified alternatives.

These elements together are meant both to improve the requirements as a description of the product being acquired and also to provide the basis for revising those requirements as understanding of needs improve or when needs or technology change in the future. When uncertainties exist and are resolved, capturing the rationale provides valuable insight to future developers. When needs change, this rationale provides a basis for understanding the implications of having to differently resolve those previous uncertainties in order to revise the product. To build a product, all requirements’ uncertainties have to be resolved in some way to finally build a product, but the goal is to not resolve an uncertainty prematurely or for all time.

A key focus suggested in a proposed roadmap for improving software producibility was bridging the conceptual gap between customers and product developers, based on recognizing that there are alternative ways of expressing any problem and many potential solutions that can result (Campbell, 2008). With this perspective, no specific expression of requirements is the “right” one; rather different expressions may be suitable for different purposes. However, underlying all valid expressions are a set of assumptions about certainty, which aspects of needs and associated potential solutions are intrinsic and fixed and which are tradable and changeable. In product lines, these assumptions, of commonality and variability, provide a framework in which true certainties provide a framework within which uncertainties are identified as choices that customers and developers must make through an ongoing process of evaluation and refinement over the life of a needed product.

As is true for product lines, a shift to uncertainty-based acquisition does not require major changes in acquisition policy but rather a change in the level at which programs are required to establish binding requirements (Campbell, 2002). These should be at the level of observable mission-enabling capabilities to be achieved through an iterative process of learning and refinement rather than with a premature over-constrained specification of a specific top-down solution to narrowly conceived needs. This will require changes in the practice of acquisition, replacing the illusion of certainty with a process that, by exposing uncertainties, builds a stronger foundation for the efficient, predictable delivery of correct and effective capabilities needed by customers.

References

Achieving effective acquisition of information technology in the Department of Defense, S4-S5. (2009). Washington, DC: The National Academies Press.

Campbell, G.H. (2002, June). *A software product line vision for defense acquisition* (CMU/SEI-2002-TN-002). Pittsburg, PA: CMU, Software Engineering Institute.

Campbell, G.H. (2008, December). Advancing producibility for software-intensive systems. *Software Tech News*, *11*(4), 13-17.

Campbell, G.H., Faulk, S.R., & Weiss, D.M. (1990, June). Introduction to synthesis. *Software Productivity Consortium*. Retrieved from http://www.domain-specific.com/PDFfiles/IntroSyn.pdf

Clements, P.C., & Northrop, L.M. (2002). *Software product lines: Practices and patterns*. Upper Saddle River, NJ: Addison-Wesley Professional.

Defense Science Board Task Force (DSB). (2009, April). *Creating a DoD strategic acquisition platform, (XDDSB)*. Defense Science Board Report. Washington, DC: Author.

Parnas, D.L., & Clements, P.C. (1986, February). A rational design process: How and why to fake it. *IEEE Trans. on Software Engineering*, *12*(2), 251-257.

USD (AT&L). (2008, December 8). *Operation of the defense acquisition system* (DoD Instruction 5000.02). Washington, DC: Author.

Acknowledgements

The views expressed here are solely those of the author and do not represent positions of the Software Engineering Institute, Carnegie Mellon University, or the US Department of Defense. Comments by John Robert, Dr. Ken Nidiffer, Patricia Oberndorf, and Terry Dailey were very helpful to me in improving this paper.

Towards Real-time Program Awareness via Lexical Link Analysis

Ying Zhao—Dr. Ying Zhao is a Research Associate Professor at the Naval Postgraduate School (NPS). Dr. Zhao joined NPS in May 2009. Her research is focused on knowledge management approaches, e.g., data text/mining such as lexical link analysis, search and visualization for system self-awareness, decision-making and collaboration. She received her PhD in mathematics from MIT and co-founded Quantum Intelligence, Inc. She has been Principal Investigator (PI) for six DoD Small Business Innovation Research (SBIR) awarded contracts. She is a co-author of two patents in knowledge pattern search from networked agents, fusion and visualization for multiple anomaly detection systems.

Dr. Ying Zhao
Research Associate Professor
Naval Postgraduate School Information Sciences Department
1411 Cunningham Road, Root Hall 229
Monterey, CA 93943-5000
E-mail: yzhao@nps.edu; Phone: 831-656-3789

Shelley Gallup—Dr. Shelley Gallup is a Research Associate Professor at the Naval Postgraduate School's Department of Information Sciences, and Director of Distributed Information and Systems Experimentation (DISE). Dr. Gallup has a multi-disciplinary science, engineering and analysis background, including microbiology, biochemistry, space systems, international relations, strategy and policy and systems analysis. He returned to academia after retiring from naval service in 1994 and received his PhD in engineering management from Old Dominion University in 1998. Dr. Gallup joined NPS in 1999, bringing his background in systems analysis, naval operations, military systems and experimental methods first to the Fleet Battle Experiment series (1999-2002) and then to the FORCEnet experimentation in the Trident Warrior series of experiments (2003-present).

Dr. Shelley P. Gallup
Research Associate Professor
Naval Postgraduate School; Director
Distributed Information and Systems Experimentation (DISE) Group
Information Sciences Department
1411 Cunningham Road, Root Hall 103D
Monterey, CA 93943-5000
E-mail: spgallup@nps.edu; Phone: 831-656-1040

Doug MacKinnon—Dr. Doug MacKinnon is a Research Associate Professor at the Naval Postgraduate School (NPS). Dr. MacKinnon led an NPS research team to assess new MDA, spiral-1 technologies being fielded by PEO C4I developing original decision matrix structures and metrics structures to leverage the new technology. He has also led the assessment of TPED (Tasking, Planning, Exploitation, and Dissemination) process during field experiments Empire Challenge 2008 and 2009 (EC08/09). He holds a PhD from Stanford University, conducting theoretic and field research in Knowledge Management (KM). He has served as the program manager for two major government projects of over $50 million each, implementing new technologies while reducing manpower requirements. He has served over 20 years as a Naval Surface Warfare Officer, amassing over 8 years at sea, serving in four US Navy warships with five major, underway deployments.

Dr. Douglas J. MacKinnon
Research Associate Professor
Naval Postgraduate School
Information Sciences Department
Graduate School of Operational and Information Sciences
1411 Cunningham Road, Glasgow West, Room 3008
Monterey, CA 93943-5000
E-mail: djmackin@nps.edu; Phone: 831-656-1005

Abstract

DoD acquisition is an extremely complex system, comprised of myriad stakeholders, processes, people, activities, and organizations in an effort to provide the most useful capabilities to warfighters at the best possible value to the government. This effort is being accomplished by acquisition analysts who despite years of experience are encumbered by mountains of available data. To assist the analyst, we consider that the cognitive interface between decision-makers and a complex system may be expressed in a range of terms or "features," i.e., specific vocabulary to describe attributes. This offers the opportunity to more easily compare two competing technologies, which, in turn, may be compared to the Navy warfighter requirements. This effort can allow decision-makers to become aware of what programs, systems, and specific features are available for acquisition and how well they match warfighter's needs and requirements with greater effect and immediacy—possibly in real-time. We present a data-driven automation method, namely, Lexical Link Analysis (LLA), to facilitate and automate acquisition system s*elf-awareness*.

Introduction

DoD acquisition is an extremely complex system, comprised of myriad stakeholders, processes, people, activities, and organizations in an effort to provide the most useful capabilities to warfighters at the best possible value to the government. According to the *Chairman of the Joint Chiefs of Staff Instruction for Joint Capabilities Integration and Development System (JCIDS) (J-8 CJCSI 3170.01G)* (JCIDS, 2009), there are three key processes in the DoD that must work in concert to deliver the capabilities required by the warfighter: the requirements process; the acquisition process; and the Planning, Programming, Budget, and Execution (PPBE) process. In particular, the requirements process is implemented in a process called Joint Capabilities Integration and Development System (JCIDS), as shown in Figure 1. JCIDS plays a key role in identifying the capabilities required by the warfighters to support the National Defense Strategy, the National Military Strategy, and the National Strategy for Homeland Defense. The Defense Acquisition System (DAS) looks on enterprise asset acquisition based on JCIDS requirements, and PPBE is focused on the management of financial resources in accomplishing enterprise asset creation, sustainment and reuse. The leadership and decision-makers constantly contend with two major questions:

1. Are we responding to strategic guidance and joint capability needs?
    1. Are we getting the best value for taxpayers?

As shown in Figure 1, JCIDS alone produces a large amount of detailed documents (e.g., Initial Capabilities Document (ICD), Formal Capability Development Document (CDD), for material solutions or doctrine, organization, training, materiel, leadership and education, personnel, or facilities (DOTMLPF), Change Recommendations (DCR) for non-material solutions, and Capability Production Document (CPD)). Each involves diversified stakeholders such as sponsors, program managers, developers, the Joint Requirements Oversight Council (JROC) and the Milestone Decision Authority (MDA).

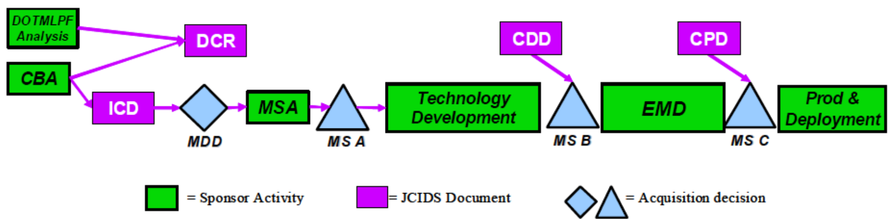


Figure 1. JCIDS Process and Acquisition Decisions
(JCIDS, 2009)

Warfighters' requirements are documented in Universal Joint Task List (UJTLs) or Joint Capability Areas (JCAs), which are collections of required capabilities functionally grouped to support mission analysis, capability analysis, strategy development, investment decision-making, capability portfolio management, and capabilities-based force development and operational planning.

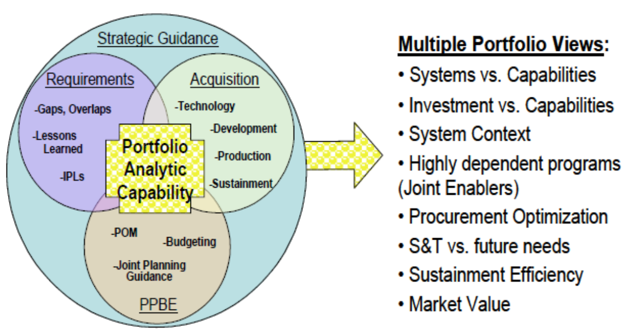


Figure 2. Portfolio Analytic Capability
(Appleton, 2009)

In summary, the major challenges in the current process can be summarized as follows:

1. To make optimal investment decisions, acquisition managers must analyze a full spectrum of data, including data that encompasses capability requirements, planning, development, integration, testing, architecture, standards, cost and schedules. This can be a daunting, if not impossible, task.

2. The pace of technology change also requires agile decision-making and challenges program management to maintain constant awareness of what is available for acquisition.

3. When considering an overall demand and supply in the trade space management of the Department of Defense, as shown in Figure 2, decision-makers require advanced portfolio analytic capability that can intercept all three business processes of requirements, acquisition and PPBE under the DoD warfighting strategic guidance in the

contexts of many factors, such as systems versus capabilities, investment versus capabilities, highly dependent programs, etc., in order to maximize Return of Management (ROM) and Yield on Cost (YOC) (Appleton, 2009).

4. The information produced in the process is too voluminous and unformatted to lend itself to analysis on a large scale. Decision-makers require large-scale automation and discovery tools that can speed up the analysis quickly in response to the pace of technology change, therefore adapting DoD program development and associated funding mechanisms in an agile manner. The decision-makers also require a much more fine-grained level of analysis for program-to-program and program-to-program elements analysis using the unstructured documents directly. This is a big leap that is not provided by the current analysis capabilities.

One method to reduce unknown performance measures is through participation in annual large-scale field experimentation exercises as part of the Research, Development, Test & Evaluation (RDT&E). These experiments can provide close interaction among users, developers, the test community, and decision-makers. At Distributed Information Systems Experimentation (DISE) laboratory at NPS, we collect and analyze data, help the Navy learn and manage information and knowledge resulting from large-scale annual experimentation (e.g., Trident Warrior and Empire Challenge). We believe this experiential data, together with Lexical Link Analysis methods, will produce deepened awareness of current program effectiveness for acquisition decision-makers.

Methods

Program Self-awareness

Here we consider that the cognitive interface between decision-makers and a complex system may be expressed in a range of terms or "features," i.e., specific vocabulary or lexicon, to describe attributes and the surrounding environment of a system. This process is similar or can be modeled using human cognitive processes, where the simplest form of such a model is relationships between noun/verb. In math, the model becomes variable/function; in engineering it becomes operand/operator; in information technology, it becomes data/process or description/procedure. We have borrowed from notions of "awareness," and implement the term self-awareness of a complex system as the collective and integrated understanding of system features. A related term, "situational awareness" is used in military operations and carries with it a sense of immediacy and cognitive understanding of the warfighting situation. Here, system self-awareness, or program awareness (Gallup, MacKinnon, Zhao, Robey & Odel, 2009), allows decision-makers to be aware of what systems, programs, and products are available for acquisition, how they match warfighters' needs and requirements, recognize relationships among them, improve efficiency of available collaboration, reduce duplication of effort, and re-use components to support cost effective management—with greater immediacy, possibly in real-time.

Through our research, we present a data-driven automation method, namely, a Lexical Link Analysis (LLA) for program self-awareness. This methodology is demonstrated by extracting realistic sample data related to systems and programs included in experimentation programs, Urgent Needs Statements (UNS), and CENTCOM/NAVCENT warfighting gap/priority lists, a large-scale data set from OSD with regards to Major Defense

Acquisition Programs (DMAP) and Acquisition Category II (ACATII) weapon systems and their RDT&E documentations.

Lexical Link Analysis (LLA)

Data mining includes analytic tools that may be applied to both structured and unstructured data to confirm previously determined patterns, or to discover new patterns that are yet unknown. Text mining is the application of data mining to unstructured or less structured text files. Text mining represents an emerging field with a wide range of software implementing innovative visualization and navigation techniques. These techniques graphically represent networks of documentation that are related conceptually. Visualization of relationships enables concept discovery, automated classification, and understandable categorization of unstructured documents.

Lexical Analysis (LA, 2010) is a form of text mining in which word meanings are developed from the context from which they are derived. Lexical Analysis (LA) can also be used in a learning mode, where such words and context associations are initially unknown and are constantly being "learned," updated, and improved as more data become available. Link analysis, a subset of network analysis that explores associations between objects, reveals the crucial relationships between objects when collected data may not be complete. Lexical Link Analysis (LLA) is an extended lexical analysis and link analysis enabled in a learning mode.

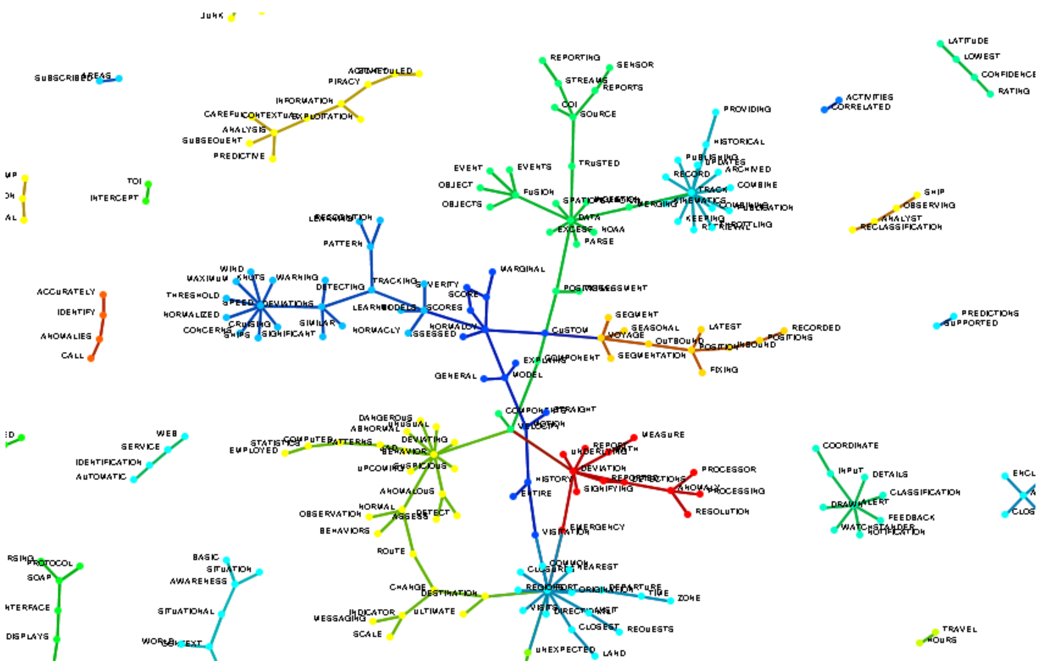


Figure 3. A Word Hub Showing the Detail on the Linkage in Figure 3

This approach clusters words and then correlates words with their textual contexts (co-occurrence), and produces a data-driven and dynamic word network. This approach is related to a number of extant tools for text mining, including Latent Semantic Analysis (LSA)

(Dumais et al., 1998), advanced search engine (Foltz, 2002), key word analysis and tagging technology (Gerber, 2005), and intelligence analysis ontology for cognitive assistants (Tecuci et al., 2007). What results from this process is a learning model—like an ethnographic *code book* (Schensul, Schensul & LeCompte, 1999)—containing descriptions of both patterns and anomalies, generated using encountered terms. As an example shown in Figures 3 and 4, we applied our approach to Maritime Domain Awareness (MDA) technologies that were evaluated in Trident Warrior 08. Figure 3 shows a visualization of LLA with connected keywords or concepts extracted from the documents of MDA technologies. Words are linked as word pairs that appear next to each other in the original documents. Different colors indicate different clusters of centralization among word groups. They are produced using a link analysis method, a social network grouping method (Girvan & Newman, 2001): words are connected as shown in one color as if they are in a social community. A "hub" is a word centered with a list of other words ("fan-out" words) centered around other words. For instance, in Figure 4, the word "behavior" is centered with "suspicious, bad, dangerous, abnormal, usual, and anomalous," etc., showing the ways to describe "behavior" in the MDA area.

Figures 5 and 6 show a visualization of lexical links between Systems 1 and 2. Each node is a feature, or word hub; each color refers to the collection of lexicon (features) to describe a system, the overlapping area nodes refer to *lexical links* between systems. The nodes toward the two ends of the links represent the unique features related to each system.

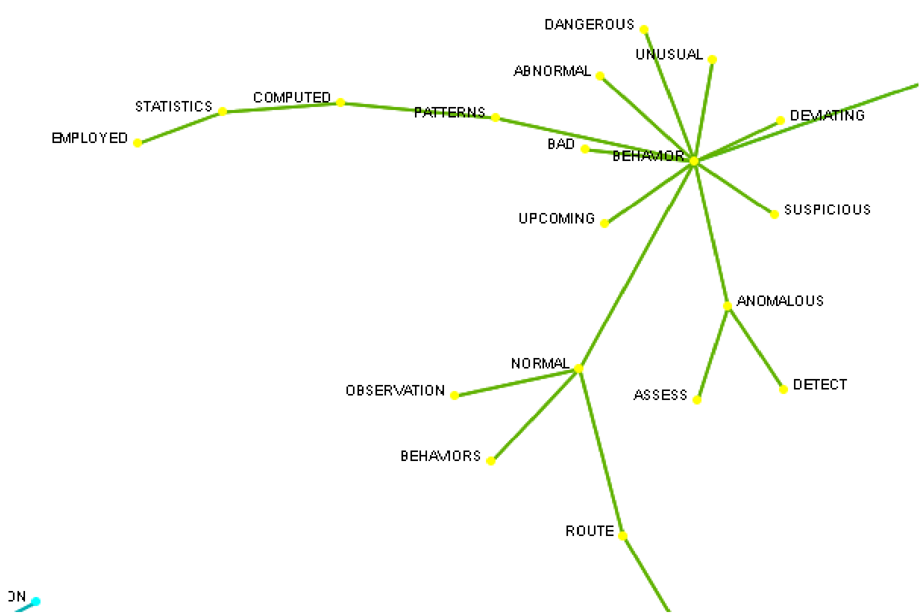


Figure 4. A Word Hub Showing the Detail on the Linkage in Figure 3

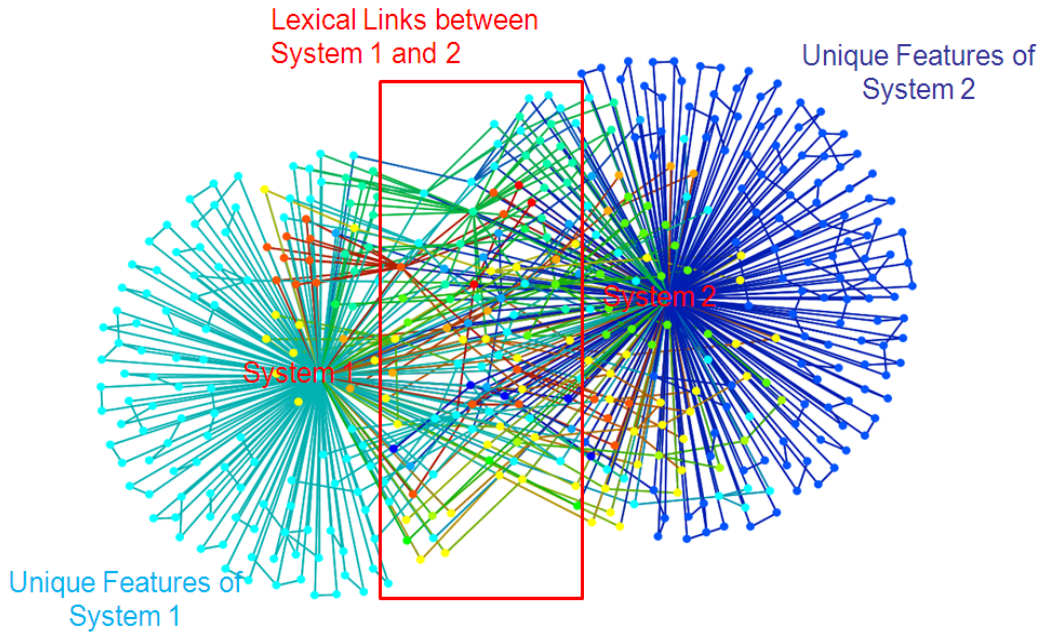


Figure 5. Visualization of Lexical Links

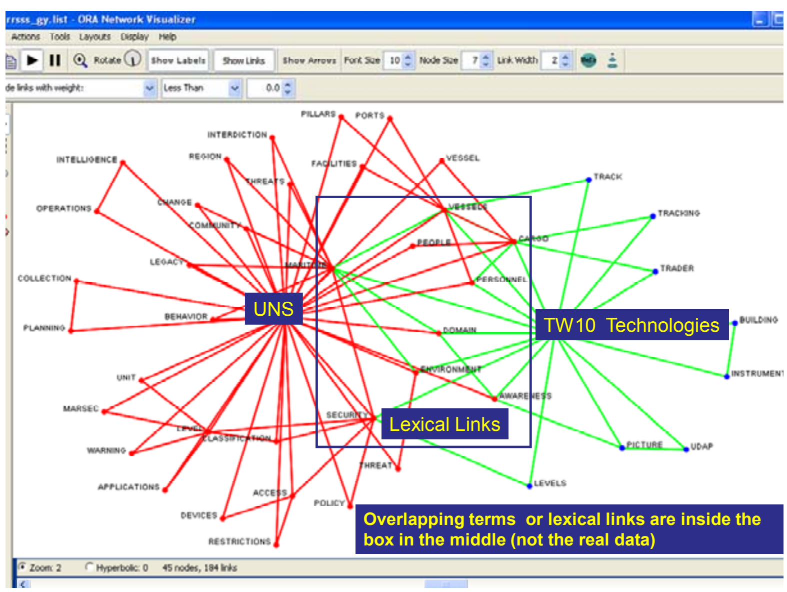


Figure 6. Overlapping Terms or Lexical Links, Shown in the Middle of Two Word Networks as the Result of the LLA Analysis

In summary, LLA provides a methodology and tools to address the following specific areas that can impact acquisition decision-making:

- LLA provides a metric to link warfighters' needs with the capabilities by directly comparing the documents that resulted from the business process—for example, linking "programs," specifically MDAPs, to operational capabilities. The number of lexical links, extracted to reflect the meaning of the documents between two systems or programs, can be a measure of consensus or synergy between the two. This compelling perspective is central to the notion of portfolio management, for example, to answer the questions: What are the programs (e.g., MDAPs) related to a given capability? What are the gaps of warfighter requirements not addressed by current programs? Currently, human analysts are responsible to answer these questions manually. Automation is needed to facilitate human analysis and to process large volumes of data quickly.

- LLA visualization is also important for acquisition decision-making. Producing a picture illustrating where the needs are met and where the overlapping efforts and gaps are will allow decision-makers to become aware of the overall situation, thus allowing them to see trends in a larger, broader scale and in a longer timeframe. For example, combining the analyses of the Army, Navy, and Air Force from RDT&E and procurement documents might show the linkages within and among programs, as they mature from development to production. Modified programs can be illustrated to show the trend toward (or deviation away from) warfighters' needs during the program's life span. One may also visually see the resource sharing (or wasting) practices and note opportunities for growth when all the data can be summarized in a discernable picture.
- LLA discovers latent, implicit, or second-order relationships by examining the detailed budget justification documents. In general, programs retain their identities from development to production, yet may change their names or be re-designated, resulting from a milestone decision or other action. The "New Attack Sub" or "NSSN" during development, for instance, was referred to as the "Virginia Class Sub" in production. The "Joint Strike Fighter" and "F-35" are also synonymous. The official "decoder" for these transformations is the DAMIR system. We note that the mapping of MDAPs to their predecessors, successors, constituents, or dependent partners is non-trivial and is, in fact, one of the fundamental challenges for acquisition analysts.
- LLA could affect the fundamentals of acquisition processes through automation and discovery. In the defense acquisition community, decision-makers are interested in determining the costs of these programs relative to their predicted baselines (e.g., Milestone B or C). They must also determine why costs change over time. Historically, acquisition researchers only considered endogenous factors (e.g., poor program management skills) as drivers of cost changes. The notion of interdependence as a potential driver of cost may be determined by LLA. It may also help determine whether this interdependence among programs may be manifested in the sharing of resources among programs, as described by the budget artifacts. Budget artifact data are voluminous, and unstructured, which make empirical analysis extremely difficult—if not humanly impractical. Previous research has been done in this area using manually identified program interdependencies (M. Brown, personal communication, 2010) and has made great progress in establishing that interdependence exists and how they might be correlated with the program costs. LLA could automate this process of identifying interdependencies and, thus, reveal aspects of interdependence that would otherwise remain obscure.

LLA Processes

The LLA Analysis

We began at the Naval Postgraduate School (NPS) by using Collaborative Learning Agents (CLA) (QI, 2009) and expanded to other tools, including AutoMap (AutoMap, 2009) for improved visualizations. Results from these efforts arose from leveraging intelligent agent technology via an educational license with Quantum Intelligence, Inc. CLA is a computer-based learning agent or agent collaboration, capable of ingesting and processing data sources. Each CLA is capable of revealing patterns that occur frequently and anomalies that occur rarely. Anomalies that might be interesting are thus revealed so that human analysts are alerted and can further investigate them. The CLA is able to separate the patterns from anomalies using the "patterns and anomalies separation" algorithm in each CLA to select feature-like word pairs for the LLA method.

The following are the steps for the LLA analysis:

1. Read two documents into the CLA (e.g., Urgent Needs Statement (UNS)) and a targeted technology document set (e.g., Trident Warrior 2010 (TW10).

    5. Select feature-like word pairs based on clusters using the CLA anomaly search method (Zhao & Zhou, 2008).

    6. Apply social network algorithm to group the word pairs into word categories.

    7. Apply AutoMap to visualize the associations of the requirement document set (UNS) and targeted technologies (TW10) document sets, as shown in Figures 5 and 6.

    8. Generate lexical link matrices used for further analyses, as shown in Figures 8, 9, and 10.

When mining text data or performing lexical analysis, we also apply entity extraction, known as Named Entity Recognition (NER), (NER, 2010; Nadeau, Turney & Matwin, 2006), which recognizes named entities such as persons, organizations, locations, expressions of times, quantities, monetary values and percentages in context. The extracted entities could also be examined separately. Excluding these modifiers from the terms resulting from Lexical Link Analysis (LLA) can provide an improved comparison by focusing on term semantics.

In some applications, differentiating nouns from verbs and adjectives, or having the ability to parse the syntax into nouns, verbs, subjects, and objects, could be helpful to acquisition managers to develop understanding. We also use a Part-of-Speech (POS) *tagger* as pre- or post-processing filters for this purpose. A POS tagger is a piece of software that reads text in some language and assigns parts of speech to each word, such as a noun, verb, adjective, etc. We have chosen the Stanford Natural Language Processing (NLP) tool (Toutanova, Klein, Manning & Singer, 2003; Stanford NLP, 2009) to perform this task. The POS taggers are usually language dependent. Our method is statistically based and can, therefore, employ NER and POS as pre- or post-processing filters.

Data Sets

We report a case study using LLA comparing US Navy Urgent Need Statements (UNS) with Trident Warrior 10 Technologies. The goal is to compare the two respective data sets, the first one is an Excel file (UNS.xls) representing Urgent Need Statements collected from C4I users. Each urgent need is listed as a statement. The UNS.xls is classified; therefore, details of this document set are not reported in this paper. The second data set is called "Focus Area Assignment TW 10.xls," also in an Excel format. It includes information from each selected technology in Trident Warrior 10.

Trident Warrior (TW) is an annual Navy FORCEnet operational experiment. At the Distributed Information Systems Experimentation (DISE) laboratory at NPS, we collect and analyze data from this and other experimentation venues to help the Navy learn and manage information and knowledge resulting from large field experiments such as Trident Warrior to provide a basis for DoD acquisition of systems and technologies. The technology information includes each technology's objective(s) for the experimentation, including Concept of Operations (e.g., how a warfighter will utilize it), and what each technology provider intends to learn from the experimentation (e.g., decrease timeline, standardized process, and/or reduced workload, etc.). TW data also includes decisions that may affect experimentation findings.

Result Presentation and Visualization Tools

Figure 7 illustrates a result summary revealing terms or word pairs combined into word categories, displayed in a radial graph. The categories with radius = 2 represent overlapping word categories that are found in both requirements (UNS) and technologies (TW10). The categories with radius = 1 indicate where gaps exist, i.e., terms that show in the UNS but not in the TW10 technologies or vice versa. We determine that there is between a 60% and 70% match overlap of technology correlations between UNS and TW 10 technologies. For example, 42 of 67 (62%) of the UNS word categories matched (were served by) with TW10 technologies.

In addition, word network views of lexical links are produced using a network tool, AutoMap. We also developed several outputs to view the detailed LLA analysis results as shown in Figures 8, 9, and 10. Figure 8 shows an Excel document output, including a few columns of information as follows:

- Terms: Matching terms or word categories discovered automatically via the LLA method.
- UNS: Values can be 0, 1, 2, specifically:
  - 0: terms not found in UNS,
  - 1: terms only found in UNS, and
  - 2: terms found in both UNS and TW10.
- UNS IDS: UNS documents in which the terms can be found.
- TW10: Values can be 0, 1, 2.
  - 0: terms not found in TW10,
  - 1: terms only found in UNS, and
  - 2: terms found in both UNS and TW10.
- TW10 IDS: TW10 documents in which the terms can be found.
- Tech Features: Terms only belong to TW10.

- As one scrolls down, if there is “0” in the TW10 column, then it indicates a gap area for TW10. Similarly, in scrolling further, if there is a “0” in the UNS column, then this indicates a gap in UNS.

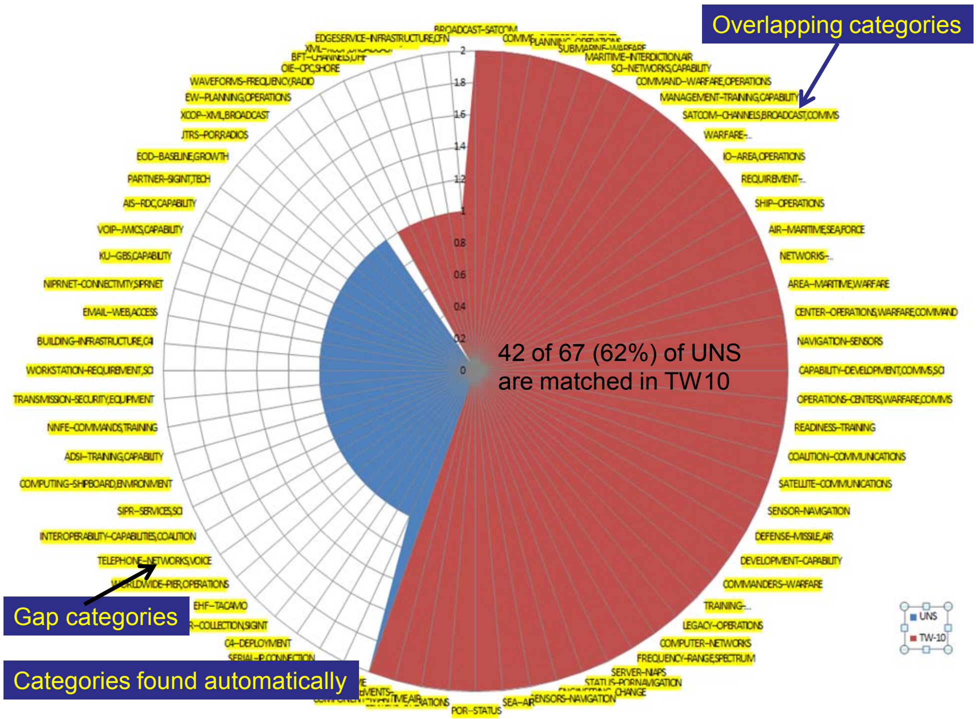


Figure 7. Resulting View and Visualization Illustrating “Overlapping” and “Gap” Word Categories

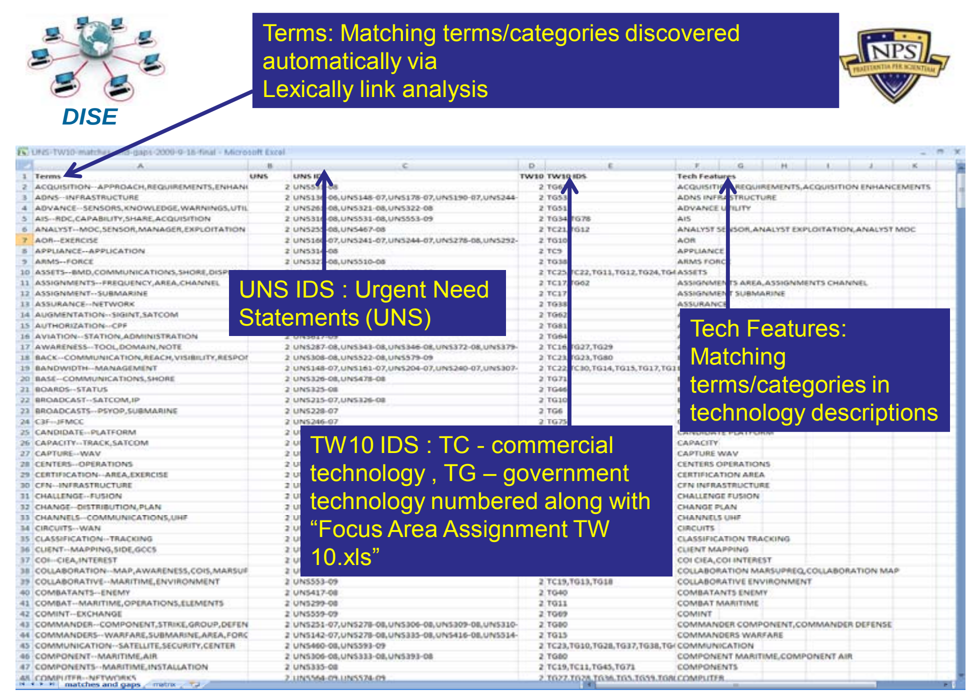


Figure 8. The Spreadsheet View of the LLA Analysis with “Matched” Terms and “Gap” Terms

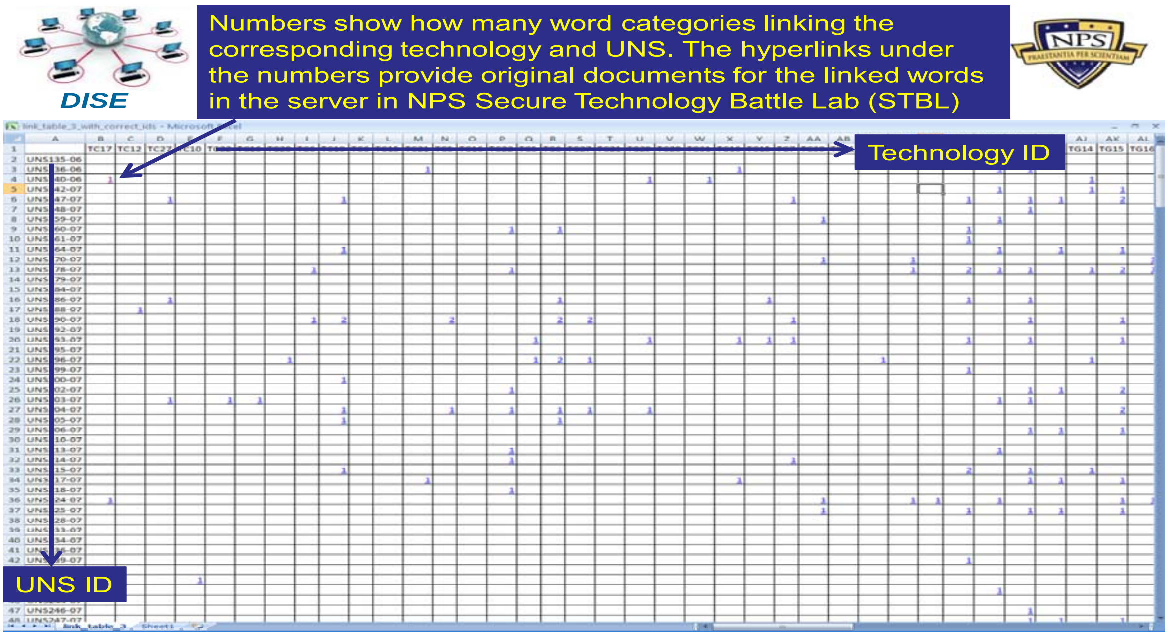


Figure 9. The Matrix View of the LLA Analysis

Figure 9 shows a matrix view of UNS to TW 10 technologies. Where numbers are seen indicates a numerical reference to the number of the "concepts" (terms or word categories) included between UNS and technologies that are being satisfied. Usually, there are multiple concepts within a UNS statement and a tech description. Each number is also a hyperlink back to the original document in a server where it is stored, e.g., the server in the NPS Secure Technology Battle Lab (STBL) for classified documents.

These results can be increasingly focused as the Intelligent Agent (IA) becomes “tuned,” or *learns* what it is that the researcher is attempting to understand. This effort can then become increasingly automated.

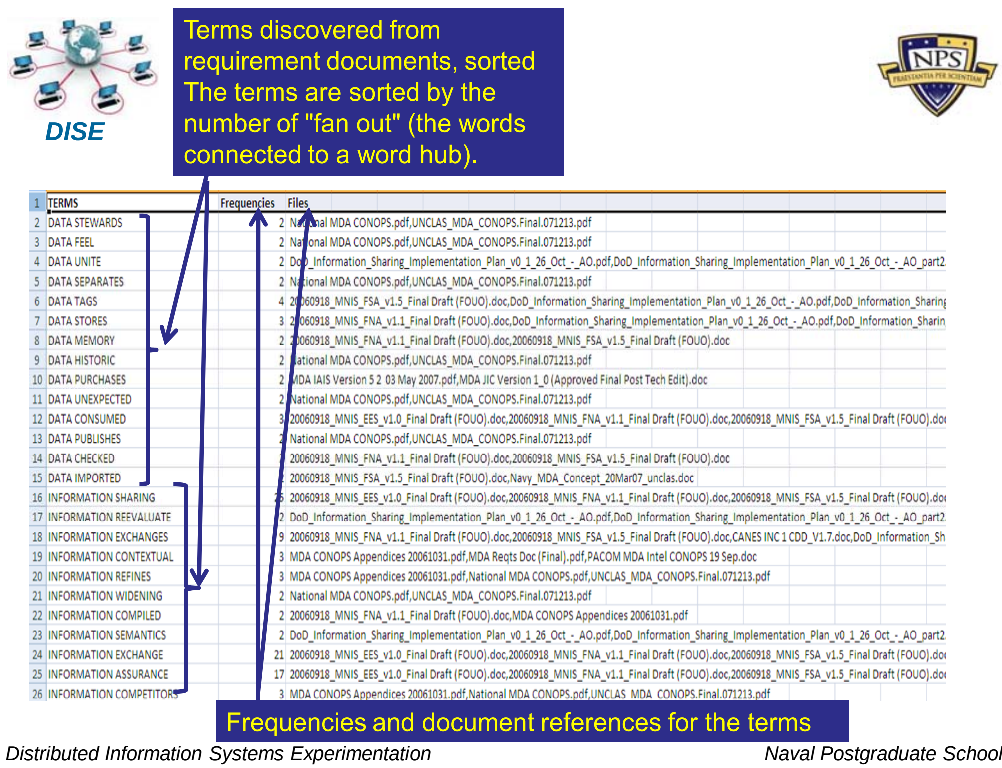


Figure 10. Frequency Count and Document References

Figure 10 shows a summary spreadsheet listing the terms and number of files in which the terms appear. This output can be used to discover concepts (terms) that are cross-validated by at least two documents in a document set. The terms are sorted by the number of "fan out" (the words connected to a word hub), showing the critical concepts being addressed across multiple documents. The top few sorted word groups, e.g., “data” and “information” in this case, are the key requirements that result in substantial consensus across different levels of requirement generation mechanisms—for example, Joint Integrating Concept (JIC), Joint Capability Areas (JCA), the Universal Joint Task List (UJTL), and user communities such as US Northern Command, US Pacific Command, and sponsors that are interested in Interagency Investment Strategies (IISs).

Validity

Several methods are being investigated to validate LLA methods. Currently, we have shown these proof-of-concept results to Subject-matter Experts (SME) from various organizations (e.g., Joint Force Development and Integration, the J-7 Staff) for evaluation and comment. One MDA expert has commented on the summary spreadsheet by saying, “it is very useful, particularly the frequency count and the documented reference.” Other SMEs comment that “LLA has great potential to help us link the UNS with the technology and further fill in the gaps that are out there.” “This would be highly useful and has great potential to help us in the larger N9/Sea Trial construct and spoke further of the possibility of using LLA at the Joint Warfighter Challenges level.” We will consider quantitative content validation methods between SMEs and LLA, such as correlation and inter-rater reliability scores (Cohen's Kappa; Kerlinger & Lee, 1992), as well as large-scale correlation calculation used in sections below.

Towards a Large-Scale Example of Program Self-Awareness

We have worked with OUSD(AT&L)/ARA/EI on the broader data sets and a large-scale application of program self-awareness via LLA.

Data Sets

Figure 11. DoD Budget Documentation

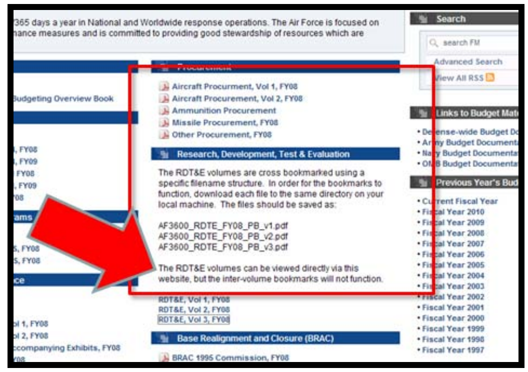
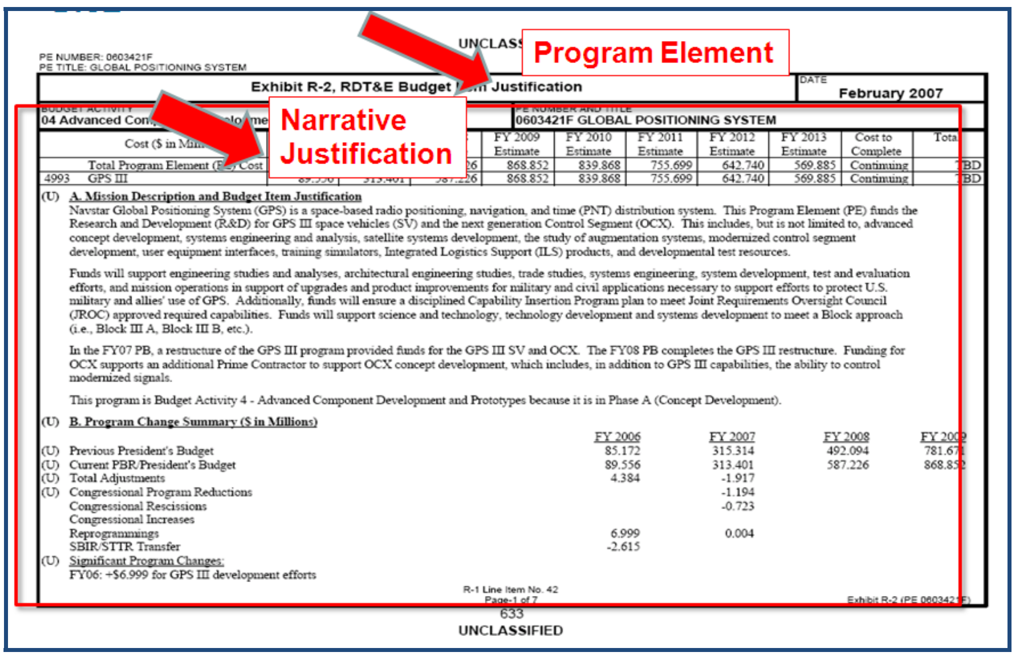


Figure 12. Research, Development Test & Evaluation (RDT&E)



UNCLASSIFIED

PE NUMBER: 0603421F
PE TITLE: GLOBAL POSITIONING SYSTEM

Exhibit R-2, RDT&E Budget Item Justification — DATE: February 2007

BUDGET ACTIVITY: 04 Advanced Component Development — PE NUMBER AND TITLE: 0603421F GLOBAL POSITIONING SYSTEM

Cost ($ in Millions)				FY 2009 Estimate	FY 2010 Estimate	FY 2011 Estimate	FY 2012 Estimate	FY 2013 Estimate	Cost to Complete	Total
Total Program Element (PE) Cost			587.226	868.852	839.868	755.699	642.740	569.885	Continuing	TBD
4993 GPS III	89.556	313.401	587.226	868.852	839.868	755.699	642.740	569.885	Continuing	TBD

(U) A. Mission Description and Budget Item Justification

Navstar Global Positioning System (GPS) is a space-based radio positioning, navigation, and time (PNT) distribution system. This Program Element (PE) funds the Research and Development (R&D) for GPS III space vehicles (SV) and the next generation Control Segment (OCX). This includes, but is not limited to, advanced concept development, systems engineering and analysis, satellite systems development, the study of augmentation systems, modernized control segment development, user equipment interfaces, training simulators, Integrated Logistics Support (ILS) products, and developmental test resources.

Funds will support engineering studies and analyses, architectural engineering studies, trade studies, systems engineering, system development, test and evaluation efforts, and mission operations in support of upgrades and product improvements for military and civil applications necessary to support efforts to protect U.S. military and allies' use of GPS. Additionally, funds will ensure a disciplined Capability Insertion Program plan to meet Joint Requirements Oversight Council (JROC) approved required capabilities. Funds will support science and technology, technology development and systems development to meet a Block approach (i.e., Block III A, Block III B, etc.).

In the FY07 PB, a restructure of the GPS III program provided funds for the GPS III SV and OCX. The FY08 PB completes the GPS III restructure. Funding for OCX supports an additional Prime Contractor to support OCX concept development, which includes, in addition to GPS III capabilities, the ability to control modernized signals.

This program is Budget Activity 4 - Advanced Component Development and Prototypes because it is in Phase A (Concept Development).

(U) B. Program Change Summary ($ in Millions)

	FY 2006	FY 2007	FY 2008	FY 2009
(U) Previous President's Budget	85.172	315.314	492.094	781.671
(U) Current PBR/President's Budget	89.556	313.401	587.226	868.852
(U) Total Adjustments	4.384	-1.917		
(U) Congressional Program Reductions		-1.194		
Congressional Rescissions		-0.723		
Congressional Increases				
Reprogrammings	6.999	0.004		
SBIR/STTR Transfer	-2.615			

(U) Significant Program Changes:
FY06: +$6.999 for GPS III development efforts

R-1 Line Item No. 42
Page-1 of 7
Exhibit R-2 (PE 0603421F)
633
UNCLASSIFIED

Figure 13. Program Element RDT&E Budget Justification

1. We have obtained program element (PE) data, which are used for DoD budget justification each year, as shown in Figure 11. One PE component is Research, Development, Test & Evaluation, which is the budget estimation, allocation and justification used for programs in the earlier stages of development. The procurement of PE components is the counterpart used for mature products. RDT&E books are obtained from the Air Force, Army

(http://asafm.army.mil/Document.aspx?OfficeCode=1200) and Navy (http://www.finance.hq.navy.mil/fmb/11pres/BOOKS.htm) websites.

9. The *Weapon Book* (Weapon, 2008), which summarizes weapons and their basic functions and missions, combined total cost from RDT& and procurement.

10. MMT databases contain cost and schedule information for each program. They consist of MDAPs and weapon systems. MMT databases also contain various program interdependencies identified by human analysts that can be used for validation. MMT databases also contain JCAs and UJTLs mapped to programs that are handmade by human experts.

According to program managers Data (1) and (2) are so voluminous, unformatted and unstructured that traditional analysis methods are difficult to apply on this scale; therefore, they are the major focuses of the analysis for LLA. There are about ~500 PEs and ~80 weapon systems extracted from data sets (1) and (2), with a total size about ~200M. Data (3) is unstructured and various previous research has been conducted on this data and, therefore, can be used to validate the LLA method against human analyses.

LLA Analysis

The focus of this paper is to show that the LLA method is capable of improving system self-awareness. LLA is able to produce this by providing an improved methodology and toolset for automation and discovery of patterns and anomalies within structured and unstructured data. This discovery can be used to produce graphics illustrating gaps and overlaps existing between systems and the needs of the DoD by basing comparisons on the *features* of each system. This methodology can have the effect of improved savings for the DoD, while developing high-value products that meet warfighters' needs.

pe_ujtls.matrix.html - Microsoft Excel

	A	B	C	D	E	F
1		0101113F	0101122F	0101221N	0101226N	0101313F
2	OP 1.1.1	28,THEATER,THREAT,STRATEGIC	9,STRATEGIC	64,PREPARE,CONSISTENT,TRANSITION,THREAT,STRATEGIC	1	78,THEATER,REQUESTS,PREPARE,(TANT,COUNTER,COMMANDER,TH
3	OP 1.1.2.1	8,COMBAT,MISSIONS	8,COMBAT,MISSIONS,SUSTAINMENT	47,COMBAT,SPECTRUM,ENTRY,SUSTAINMENT		14,MISSIONS,SPECTRUM
4	OP 1.1.2.2			16,ASSETS		2,ASSETS
5	OP 1.1.2	5,COMBAT,MODE	7,COMBAT	33,COMBAT,DEPLOYMENT,DEPLOYED	2,DESIGNATED	23,COMMANDER,ADMINISTRATIV OYMENT,DEPLOYMENT
6	OP 1.1.3.1	6,THEATER,PROCESSING		17,CONTRACTORS,PROCESS		19,THEATER,PROCESSING,ADMINI VE
7	OP 1.1.3	25,PODS,PROCESSING,INTEGRATION,TACTICAL	10	45,INTEGRATION,SHIPS,ACHIEVE,READINESS,DEPLOYED	4,READINESS	35,PROCESSING,INTEGRATION,CC DER,TACTICAL,PRE
8	OP 1.1	21,THEATER,INTEGRATE,AIR,OPERATIONAL,MODE	8,AIR,OPERATIONAL	11,LAND,RELATIVE,AIR,OPERATIONAL		23,THEATER,INTEGRATE,COORDIN PERATIONAL
9	OP 1.2.1	5,TACTICAL	6,EXTENSION	29,READINESS,EXTENSION	2,READINESS	21,COORDINATE,COMMANDER,DI ACTICAL
10	OP 1.2.2	24,AIR,MISSILE	26,RETAIN,MISSILE,AIR	44,MISSILE,AIR,EFFECTIVENESS	7,EFFECTIVENESS	77,ORGANIZATIONS,MISSILE,COO TE,COMBINED,COMMANDER,ENG
11	OP 1.2.3.1	2,REPORTING	4,CONTRACTOR	4	6,CONTRACTOR	
12	OP 1.2.3	13,STRATEGIC,OPERATIONAL	11,STRATEGIC,OPERATIONAL	62,ACHIEVE,STRATEGIC,OPERATIONAL	1,DESIGNATED	34,COMMANDER,STRATEGIC,OPE AL
13	OP 1.2.4.1	4,ASSIGNED		21,DEMONSTRATE,EMPLOYING,CONDUCTING,ASSIGNED		22,EMPLOYING,CONDUCTING
14	OP 1.2.4.2	12,ACTION	6	30,LAND,ACTION,DEMONSTRATION		43,ACTION,EMPLOY,ADVERSARY
15	OP 1.2.4.3	24,STRIKE,STRATEGIC	14,STRIKE,STRATEGIC	68,STRIKE,MARITIME,STRATEGIC		50,STRIKE,DECISIVE,STRATEGIC
16	OP 1.2.4.4	5,AIR	4,AIR,TERRITORY	21,EXPAND,AIR		17,HOSTILE
17	OP 1.2.4.5	4,PENETRATION	9,TERRITORY	43,CONDUCTING,SECURE	1	16,HOSTILE,CONDUCTING
18	OP 1.2.4.6	11,OFFENSIVE,THREAT	12	20,SECURE,THREAT,OBJECTIVES	3,DEFENSIVE	21,THREAT,OBJECTIVES
19	OP 1.2.4.7	21,STANDOFF,CAPTURE,PRECISION,DAMAGE	7,TARGETS	44,PRECISION,MATERIAL,TARGETS	3,DESIGNATED	25,GUIDED,EMPLOYMENT
20	OP 1.2.4.8	21,EQUIPPED,SUPPORTED,INTEGRATION,WARFARE,OFFENSIVE	7,EXTERNAL	29,SUPPORTED,COVERT,INTEGRATION	9,WARFARE	16,INTEGRATION,WARFARE,EXTE

pe_ujtls.matrix

Figure 14. An Example of LLA Matrices of Program Elements (PE) against UJTLs

First, we want to show how LLA provides a new metric to measure how warfighters' needs are matched with resources and products that are being considered. Figure 14 shows an LLA matrix result using program elements as columns and UJTLs as rows. The number in each cell is a match score generated from the LLA method. Next to the score are word hubs that indicate which term is matched. Sorting this matrix according to the matched scores vertically and horizontally answers the following questions:

- Which programs (e.g., MDAPS) are related to a given capability? Which PEs are related to a given capability?
- How is the acquisition process responding to expressed capability needs? How much of the weapon systems acquisition budget is being allocated to any given operational need (e.g., UJTL).

Note that this LLA matrix can be generated for any pair of document collections that are desired for comparison, e.g., PEs versus UJTLs, weapon systems versus UJTLs and weapon systems versus weapon systems. When applied to weapon systems (MDAPs) versus UJTLs, we can answer the following question by sorting the LLA matching scores:

- Which capability(ies) does any given MDAP support? How much does the MDAP contribute to this capability?

The LLA matrices may also help to reconcile the gaps between the final products and what warfighters need after the long process of design and development. Furthermore, they may also provide new prospective for portfolio analysis. A conventional treatment of portfolio analysis is that it is typically expressed as a simple correlation between an MDAP and a capability. This simple correlation ignores the fact that no individual program (system, platform, etc.) can contribute to any capability unless other programs/systems/capabilities are in place. The analogy is that a fighter jet is useless unless it has all the supporting capabilities/infrastructure (airfield, ammo, fuel, personnel, etc.), and complementary systems (e.g., GPS, C2, satellite imagery, mission planning, etc.) to enable it to operate effectively. Considering a single MDAP in terms of how much it contributes to a given capability without considering its linkages to other systems/programs/capabilities might be counterproductive, and would likely drive bad decisions. The better approach is to consider a program in the context of its interdependencies with respect to their collective contribution to a specific capability. The interdependencies should be identified from operational needs, engineering constructions and programmatic budget justifications. Therefore, the combinations of the LLA matrices—for example, PEs versus UJTLs, weapon systems versus UJTLs and weapon systems versus weapon systems may also help to redefine portfolios and improve portfolio management.

Validity

In order to realize the potential of the LLA method, an important first step is to establish the validity of the method in the context of realistic large-scale data sets. For that, we used the matrix generated from PEs versus PEs, compared with what human analysts have identified previously. As shown in Figure 15, in each program element artifact, another program element might be referenced, indicted as precedent or directionally linked program elements. A *backward* link is usually a stronger indicator of importance of a PE than a *forward* link. This is similar to the information retrieval or page ranking in a search engine (e.g., Google). Here, we use the number total *forward* and *backward* links together, identified by human analysts, as the attributes to validate the LLA method. For example, Figure 15, PE 0604602F references PE 0605011F, in which we define it as a *forward* link, for PE 0604602F; while PE 0605011F is referenced by PE 0604602F, which we define as a *backward* link, for PE 0605011F. As shown in Figure 16, the top yellow row contains the total number of unique word hubs for a PE, matched with all PEs other than itself; and the bottom yellow row contains the total number of forward and backward links for the same PE. The Pearson correlation of the two rows is 0.39, with a p-value < 0.0000001 (bi-directional t-test with a sample size N=461). This indicates that the positive correlation between the LLA-identified links and human-analyst-identified links is statistically significant and, therefore, is a validation for the LLA method.

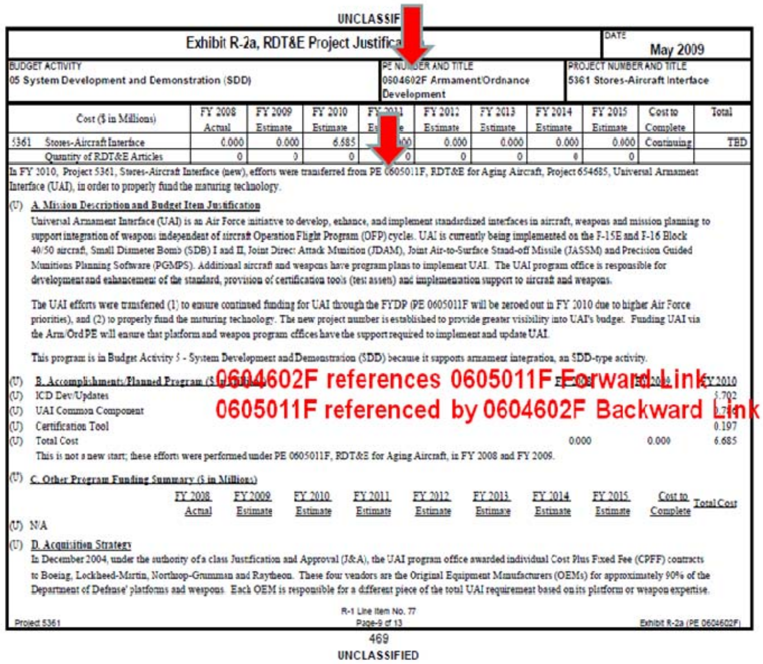

UNCLASSIF

Exhibit R-2a, RDT&E Project Justifica

DATE May 2009

BUDGET ACTIVITY 05 System Development and Demonstration (SDD)

PE NUMBER AND TITLE 0604602F Armament/Ordnance Development

PROJECT NUMBER AND TITLE 5361 Stores-Aircraft Interface

Cost ($ in Millions)	FY 2008 Actual	FY 2009 Estimate	FY 2010 Estimate	FY 2011 Estimate	FY 2012 Estimate	FY 2013 Estimate	FY 2014 Estimate	FY 2015 Estimate	Cost to Complete	Total
5361 Stores-Aircraft Interface	0.000	0.000	6.685	[illegible]	0.000	0.000	0.000	0.000	Continuing	TBD
Quantity of RDT&E Articles	0	0	0	0	0	0	0	0		

In FY 2010, Project 5361, Stores-Aircraft Interface (new), efforts were transferred from PE 0605011F, RDT&E for Aging Aircraft, Project 654685, Universal Armament Interface (UAI), in order to properly fund the maturing technology.

(U) A. Mission Description and Budget Item Justification

Universal Armament Interface (UAI) is an Air Force initiative to develop, enhance, and implement standardized interfaces in aircraft, weapons and mission planning to support integration of weapons independent of aircraft Operation Flight Program (OFP) cycles. UAI is currently being implemented on the F-15E and F-16 Block 40/50 aircraft, Small Diameter Bomb (SDB) I and II, Joint Direct Attack Munition (JDAM), Joint Air-to-Surface Stand-off Missile (JASSM) and Precision Guided Munitions Planning Software (PGMPS). Additional aircraft and weapons have program plans to implement UAI. The UAI program office is responsible for development and enhancement of the standard, provision of certification tools (test assets) and implementation support to aircraft and weapons.

The UAI efforts were transferred (1) to ensure continued funding for UAI through the FYDP (PE 0605011F will be zeroed out in FY 2010 due to higher Air Force priorities), and (2) to properly fund the maturing technology. The new project number is established to provide greater visibility into UAI's budget. Funding UAI via the Arm/Ord PE will ensure that platform and weapon program offices have the support required to implement and update UAI.

This program is in Budget Activity 5 - System Development and Demonstration (SDD) because it supports armament integration, an SDD-type activity.

(U) B. Accomplishments/Planned Program ($ in Millions)	FY 2008	FY 2009	FY 2010
(U) ICD Dev/Updates			5.702
(U) UAI Common Component			[illegible]
(U) Certification Tool			0.197
(U) Total Cost	0.000	0.000	6.685

This is not a new start; these efforts were performed under PE 0605011F, RDT&E for Aging Aircraft, in FY 2008 and FY 2009.

(U) C. Other Program Funding Summary ($ in Millions)

	FY 2008 Actual	FY 2009 Estimate	FY 2010 Estimate	FY 2011 Estimate	FY 2012 Estimate	FY 2013 Estimate	FY 2014 Estimate	FY 2015 Estimate	Cost to Complete	Total Cost
(U) N/A										

(U) D. Acquisition Strategy

In December 2004, under the authority of a class Justification and Approval (J&A), the UAI program office awarded individual Cost Plus Fixed Fee (CPFF) contracts to Boeing, Lockheed-Martin, Northrop-Grumman and Raytheon. These four vendors are the Original Equipment Manufacturers (OEMs) for approximately 90% of the Department of Defense' platforms and weapons. Each OEM is responsible for a different piece of the total UAI requirement based on its platform or weapon expertise.

Project 5361 R-1 Line Item No. 77 Page-9 of 13 Exhibit R-2a (PE 0604602F)

469

UNCLASSIFIED

Figure 15. Program Element Cross-References Identified by Human Analysts

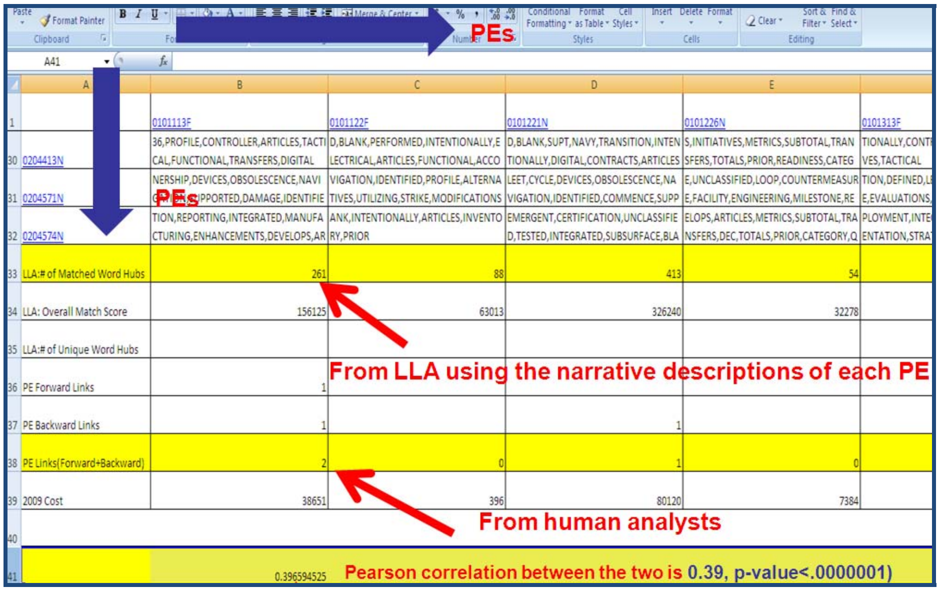


Figure 16. The Correlation Between LLA Word Hubs and PE Links Identified by SME's is Statistically Significant

Acquisition Decision-making

To support effective decision-making, we need to form a full understanding of a program in context; we need to understand the linkages and interdependencies across the operational, constructive, and programmatic domains.

An LLA matrix using programs such as weapon systems as rows as well as columns is shown in Figure 17. The lexical links output from this view show the relationships among weapon systems, therefore representing a constructive view of programs in context. The hypothesis is that more lexical links among programs may be correlated with the overall higher program total costs. The correlation between the overall LLA match score and the program total cost found in the weapon data—which includes RDT&E and procurement costs together—is 0.21, with a p-value < 0.032. This indicates there is a statistically significant relationship between the number of lexical links as an interdependency measures among programs and total cost of programs.

Similarly, a programmatic view of an LLA matrix can be generated by using weapon systems as columns and program elements as rows. The correlation between the overall LLA match scores and total program costs is 0.13 with a p-value < 0.12. This indicates that this correlation is not statistically significant based on the analyzed data.

An operational view of the LLA matrix was generated by using weapon systems as columns and UJTLs as rows. The correlation between the overall LLA match scores and total program costs is 0.086, with a p-value < 0.12, indicating that this correlation is not statistically significant.

From an acquisition management and resource analysis perspective, we conclude that

- Major programs are interdependent on one another. Interdependence can be shown by their lexical links in budget documentations in constructive, programmatic and operational views. The degree that programs are interdependent can be measured by the number of lexical links.
- Highly interconnected programs in a constructive view are statistically significantly and more expensive than less-interconnected programs (correlation 0.21, p-value < 0.032). The word hubs selected from LLA suggest the "threads" that link a portfolio of programs through shared resources. As an example, in Figure 18 ADVANCED MEDIUM RANGE AIR-TO-AIR MISSILE (AMRAAM) and AIR INTERCEPT MISSILE – 9X (AIM-9X) are connected through "COUNTERMEASURES," which may share resources from PE 030140N.

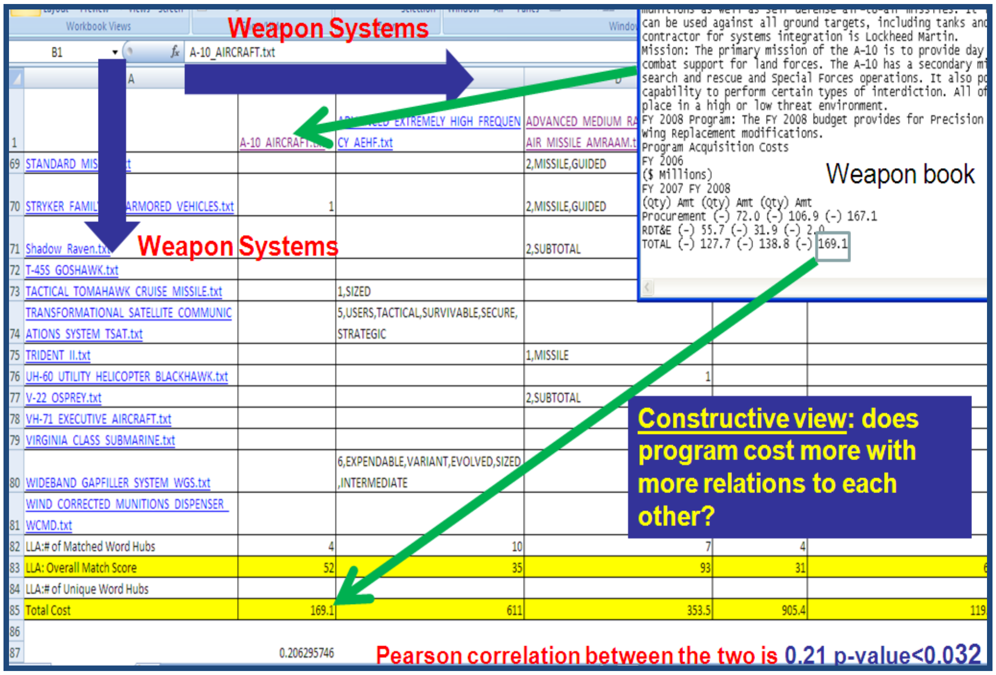


Figure 17. A Constructive View: An LLA Matrix Weapon Systems versus Weapon Systems

(Note: The correlation between the LLA overall match scores and total program costs is statistically significant.)

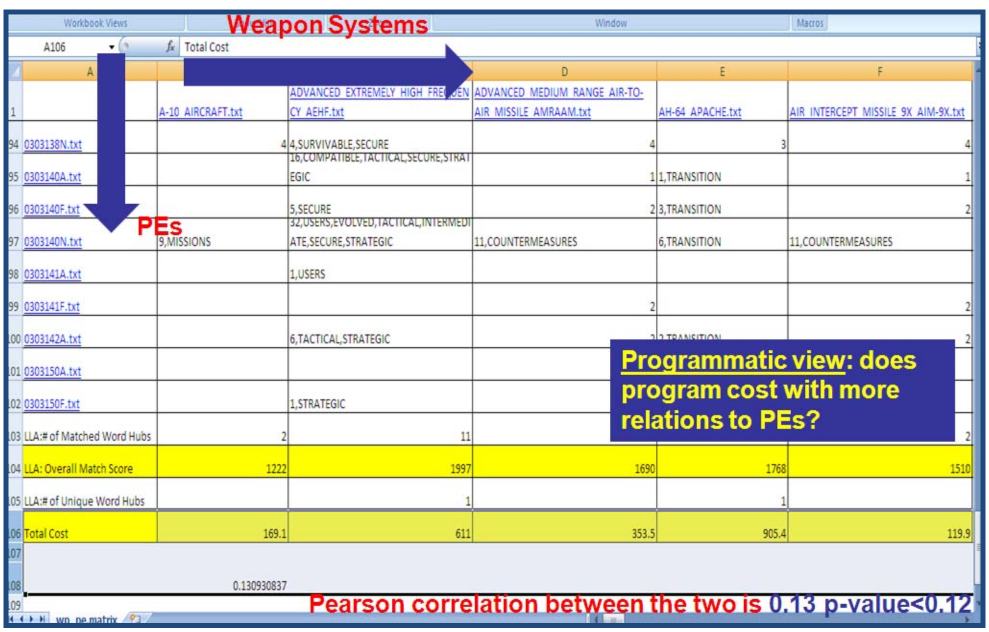


Figure 18. A Programmatic View: Weapon Systems versus Program Elements

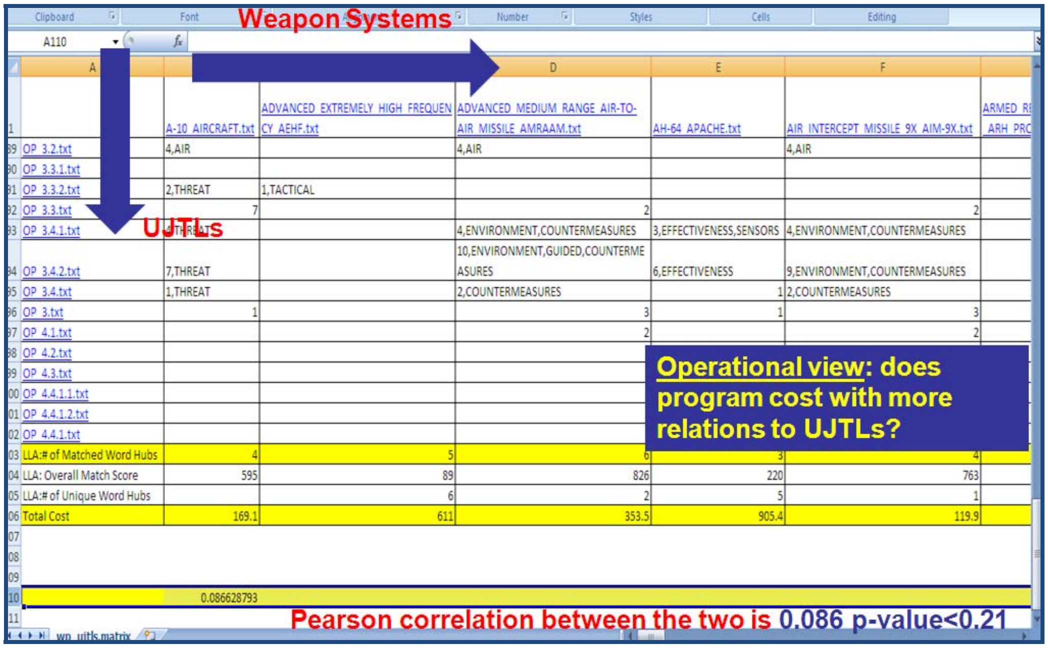


Figure 19. An Operational View: Weapon Systems versus UJTLS

Our near-term plan is to apply the method jointly with the unstructured data with the MMT databases to illustrate if the LLA method can be used to address the following questions:

1. The narrative sections reference program–to-program interdependencies (e.g., Wideband Gapfiller System flies on an EELV launch vehicle). How could this be compared with program interdependence information from the DAES, or the ISP from our data set?

11. Are these programs more or less likely to incur cost growth relative to their milestone B baselines? Are they more or less likely to breach their cost/schedule/performance baselines?

12. How do we determine the correlation using metrics that fundamentally affect acquisition decision-making? For example, total program cost and cost growth relative to the Milestone B baseline cost. (To do that, we would need to capture the total program cost (development, procurement, and the two combined) estimated at milestone B, and compare that with these values at milestone C. These data are in the MMT data set.)

13. Can LLA of budget documentation provide an aggregate dollar figure that describes the value/magnitude of resources being shared among these entities? Is this a reasonable proxy for the degree or significance of interdependence?

14. Is there additional latent risk to programs that share resources? Is there potential for unanticipated "ripple effect" that could magnify budget perturbations? Can these effects be modeled or predicted? Would this suggest that new approaches to budget analysis are needed?

Large-scale and Real-time Consideration

A large number of CLA agents work together in a parallel fashion. This allows the LLA method to scale up to distributed, large-scale and real-time data sources. At the time of this printing, we have prototyped a multi-agent network of ~10 to 100 agents in the NPS High Performance Computing Center (HPC) in the Hamming Linux Cluster (HLC), which provides the requisite supercomputing for the visualization of the results. Servers are also being built in the NPS Secure Technology Battle Lab (STBL) to process classified data.

Conclusion

We show in this paper how to use the Lexical Link Analysis (LLA) to match system features with those defined in the original requirements, discover relationships among systems, and identify gaps with respect to warfighters' needs. We initially validate the LLA method and show results by correlating program interdependencies resulted from the LLA method with those from subject-matter experts. The Pearson correlation for a sample of 461 program elements (PEs) is 0.39 with a p-value < 0.0000001. This indicates the positive correlation between the LLA identified links as compared to human-analyst-identified links and that they are reasonably correlated with statistical significance. We also found that Major Defense Acquisition Programs (MDAP's) are interdependent from one another and that such interdependence can be shown by their lexical links in documentations in constructive, programmatic, and operational views. The number of lexical links can be used as a metric to measure interdependencies among new technologies. Highly interconnected programs in a constructive view are statistically significantly and more expensive than the less-interconnected programs (correlation 0.21, p-value < 0.032). Ultimately, in this vein, we seek to use the LLA method to automate and improve program self-awareness and make it feasible for acquisition decision-makers to analyze and dynamically monitor large-scale acquisition documents. The resulting system analyses will facilitate real-time program awareness and can reduce the workload of decision-makers who would otherwise perform the relations-building task manually, thus making a profound impact on the agility and perhaps the long-term success of acquisition strategies.

Acknowledgements

We thank Mr. Dave Summer from NNWC for helping us understanding warfighter requirements in the Maritime Domain Awareness (MDA) area. We thank Mr. Robert Flowe from OSD, who pointed to large-scale data sets, provided critical acquisition research, and relevant questions along with insightful discussions.

References

Appleton, D.S. (2009). *Trade space management in the Department of Defense*.

AutoMap. (2009). Retrieved from http://www.casos.cs.cmu.edu/projects/automap/

Dumais, S.T., et al. (1988). Using latent semantic analysis to improve information retrieval. In *Proceedings of CHI'88: Conference on Human Factors in Computing* (pp. 281-285).

Foltz, P.W. (2002). Quantitative cognitive models of text and discourse processing. In *The Handbook of Discourse Processes*. Mahwah, NJ: Lawrence Erlbaum Publishing.

Gallup, S.P., MacKinnon, D.J., Zhao, Y., Robey, J., & Odell, C. (2009, October 6-8). Facilitating decision making, re-use and collaboration: A knowledge management approach for system self-awareness. In *Proceedings of the International Joint Conference on Knowledge Discovery, Knowledge Engineering, and Knowledge Management (IC3K)*. Madeira, Portugal.

Gerber, C. (2005). Smart searching, new technology is helping defense intelligence analysts sort through huge volumes of data. *Military Information Technology*, *9*(9).

Girvan, M., & Newman, M.E.J. (2002). *Community structure in social and biological networks*.

JCIDS. (2009). *Chairman of the joint chiefs of staff instruction for joint capabilities integration and development system* (JCIDS) (J-8 CJCSI 3170.01G). Retrieved from http://www.intelink.sgov.gov/wiki/JCIDS

Kerlinger, F., & Lee, H. (1992). *Foundations of behavioral research* (3rd ed.). Wadsworth.

LA. (2010). Retrieved from http://en.wikipedia.org/wiki/Lexical_analysis

Nadeau, D., Turney, P.D., & Matwin, S. (2006). Unsupervised named-entity recognition: Generating gazetteers and resolving ambiguity. In *Proceedings of the Canadian Conference on Artificial Intelligence*.

NER. (2010). Retrieved from http://en.wikipedia.org/wiki/Named_entity_recognition

QI. (2009). Retrieved from http://www.quantumii.com

Schensul, J.J., Schensul, S.L., & LeCompte, M.D. (1999). *Essential ethnographic methods: Observations, interviews and questionnaires*. Rowman Altamira.

Tecuci G., Boicu, M., Marcu, D., Boicu, C., Barbulescu, M., Ayers, C., & Cammons, D. (2007). Cognitive Assistants for Analysts. In J. Auger & W. Wimbish (Eds.), Joint Publication of the National Intelligence University, Office of the Director of National Intelligence, and US Army War College Center for Strategic Leadership.

Toutanova, K., Klein, D., Manning, C., & Singer, Y. (2003). Feature-Rich Part-of-Speech Tagging with a Cyclic Dependency Network. In *Proceedings of HLT-NAACL* (pp. 252-259).

Weapon. (2008). Retrieved from http://comptroller.defense.gov/defbudget/fy2008/fy2008_weabook.pdf

Zhao, Y., & Zhou, C. (2008a). *Fusion and visualization for multiple anomaly detection systems* (US Patent 12/042338).

Zhao, Y., & Zhou, C. (2008b). *Knowledge pattern search from networked agents* (US Patent 20080086436).

Stanford NLP. (2009). Retrieved from http://nlp.stanford.edu/software/tagger.shtml

THIS PAGE INTENTIONALLY LEFT BLANK

Panel #11 – Satisfying Requirements While Achieving Life-Cycle Cost Goals

Wednesday, May 12, 2010	
3:30 p.m. – 5:00 p.m.	**Chair: Rear Admiral Kathleen M. Dussault, US Navy**, Director, Supply, Ordnance and Logistics Operations Division
	Discussant: J. David Patterson, Executive Director, National Defense Business Institute, The University of Tennessee
	Achieving Life Cycle Capability: Ensuring Capability for Today and Tomorrow
	Lou Kratz and Bradd Buckingham, Lockheed Martin Corporation
	Acquisition of Mine Resistant, Ambush Protected (MRAP) Vehicles: A Case Study
	William Lucyshyn, Jacques Gansler and William Varettoni, University of Maryland

Achieving Life Cycle Capability: Ensuring Capability for Today and Tomorrow

Lou Kratz—Mr. Louis A. Kratz is the Vice President and Managing Director for Logistics and Sustainment, Lockheed Martin Corporation. Mr. Kratz is responsible for coordinating Lockheed Martin's logistics and weapon system sustainment efforts. Mr. Kratz leads Lockheed Martin's Automatic Identification Technology implemtation, including RFID and UID. After successfully completing eight pilot projects, Mr. Kratz is now guiding Lockheed Martin's enterprise implementation of RFID. He also guides Lockheed Martin's logistics strategic planning, performance-based logistics efforts, logistics technology development, logistics human capital development, and cross-corporate logistics business initiatives. Previously, Mr. Kratz served as the Assistant Deputy Under Secretary of Defense (Logistics Plans and Programs), within the Office of the Deputy Under Secretary of Defense (Logistics and Materiel Readiness). As such, he was responsible for guiding the DoD's logistics transformation to meet the operational requirements of the 21st Century. Mr. Kratz oversaw the DoD's long-range logistics planning to meet the requirements of the Quadrennial Defense Review (QDR) and Joint Vision 2020.

Mr. Lou Kratz
Vice President and Managing Director
Logistics & Sustainment
Corporate Engineering & Technology
Lockheed Martin Corporation
10530 Rosehaven Street, Fairfax, VA 22030
Louis.kratz@lmco.com
Phone: 703-251-7266

Bradd A. Buckingham—Mr. Bradd A. Buckingham is a Senior Business Development Analyst for Lockheed Martin Corporate Engineering and Technology, Logistics and Sustainment. He holds a Bachelor of Liberal Studies degree in Conflict, Politics, and National Policy from the University of Mary Washington. Mr. Buckingham currently provides research and analytic support in theater opening and sustainment, and in support of Army, USMC, and Defense Logistics Agency Depot management. His prior experience includes supporting the Army Manufacturing and Technology Program at Fort Belvoir, VA, as well as providing support to the Joint Defense Manufacturing and Technology Program's (JDMTP) Warfighter Brochure, and as a subject-matter expert for Army ManTech Small Business Innovative Research (SBIR) program. Mr. Buckingham was also the acting Army representative for the JDMTP SBIR Working Group.

Mr. Bradd A. Buckingham
Senior Business Development Analyst
Logistics & Sustainment
Corporate Engineering & Technology
Lockheed Martin Corporation
10530 Rosehaven Street, Fairfax, VA 22030
bradd.a.buckingham@lmco.com
Phone: 571-282-7716

Abstract

In the article Achieving Outcomes-Based Life Cycle Management (*Defense Acquisition Review Journal*, Vol. 17, January 2009), the authors traced the history of DoD acquisition reform efforts and highlighted the dramatic geo-political changes that impact the acquisition process. The authors provided three recommendations to enhance US life cycle agility and affordability to posture the DoD life cycle processes to meet the demands of the 21st Century:

- Effects-based requirements,
- Commercially driven research and development, and
- Outcome-based partnership life cycle product support.

Since that effort, the DoD and Congress have moved forward with several policy-level efforts, directed towards enhancing accountability and agility over the life cycle, including:

- *Weapon System Acquisition Reform Act* implementation,
- Insourcing,
- Product Support Assessment Team,
- *National Defense Authorization Act*, Section 805, and
- HASC Panel on Defense Acquisition Reform

This paper reviews those recent policy efforts and assesses the potential impact of those efforts on the inherent, structural incentives that are embedded in DoD life cycle processes. The paper provides several recommendations for policy implementation to further enable life cycle agility and affordability.

Introduction & Background

In the article Achieving Outcomes-Based Life Cycle Management, the authors summarized 60 years of acquisition reform efforts and concluded that incremental reform efforts are insufficient to enable the agility and efficiency required by the current national security environment. The geo-political environment of the 21st century is dramatically different than the post-World War II environment (that enabled the current acquisition process), as summarized in Table 1. Those differences required a fundamental re-assessment of DoD life cycle principles.

Table 1. Prior Acquisition Reform Efforts

	Acquisition and Logistics Characteristics		Acquisition and Logistics Outcomes		
Reform Effort	**Strengths**	**Weaknesses**	**Capability**	**Agility**	**Efficiency**
Packard Commission	Attention to acquisition streamlining	Expensive, lengthy acquisitions continue	**YES**	**NO**	**NO**
Specs/Stds Reform	Best commercial practices	Modernization "death spiral"	**YES**	**NO**	**NO**
JCIDS	Capabilities based on joint warfighter needs	Disconnect between born joint and employed joint	**YES**	**NO**	**NO**
The *Acquisition Reform Act of 2009*	- Independent cost estimates - Strengthened oversight - Improved DoD workforce	-No inherent performance incentive -"Inspect in" Program Stability	**YES**	**NO**	**NO**
Product Support Assessment Team (PSAT)	-DLA -JSCA -Government & Industry Partnership	- Extended BCA Process -Extended Peer Review -Shortened Contract length	**YES**	**NO**	**NO**
National Defense Authorization Act, Section 805	-Outcome Focused -Enhanced Accountability -Improved Workforce	Limited Metrics	**YES**	**NO**	**NO**
2010 *Quadrennial Defense Review*	-Identifies need for improving and sustaining workforce -Promotes military-commercial dual use technology use	-Further reviews = Increased oversight -review process includes parties with "no skin in the game"	**YES**	**NO**	**NO**
Future Strategies					
Effects-based Requirements	Innovation and industry competition		**YES**	**YES**	**YES**
Commercially Driven R&D	Leverage commercial R&D		**YES**	**YES**	**YES**
Industry Provided Outcome-based LCPS	Successful partnerships with DoD providers		**YES**	**YES**	**YES**

Those core differences were noted by Secretary Gates:

> What we need is a portfolio of military capabilities with maximum versatility across the widest possible spectrum of conflict. As a result, we must change the way we think and the way we plan, and fundamentally reform the way we do business and buy weapons. It simply will not do to base our strategy solely on continuing to design and buy, as we have for the last 60 years, only the most technologically advanced weapons to keep up with or stay ahead of another superpower adversary, especially one that imploded nearly a generation ago. (Gates, 2009)

Based upon those differences, the authors concluded that the DoD required life cycle management processes that built upon inherent incentives and competition and enabled for greater agility and efficiency (affordability). The authors proposed three fundamental reforms:

- Effects-based requirements,

- Commercially driven research and development, and
- Outcome-based partnership life cycle product support.

Since that publication, the DoD and Congress have initiated several reform efforts. The question to be assessed is, “Do current reform efforts enhance agility and affordability?” Key criteria to address that question include:

- Recognition and migration to a warfighter driven, effects-based requirements process;
- Enablement of a more commercial-like R&D model, where industry has a vested interest in moving through product development quickly;
- Outcomes-based sustainment models that provides required readiness at reduced costs;
- Competitive industrial base that naturally fosters innovation and agility; and
- Life cycle workforce that includes the appropriate core competencies in sufficient strength.
-

The authors reviewed ongoing reform efforts against those key criteria. Ongoing efforts included:

- *Weapon System Acquisition Reform Act of 2009*,
- Insourcing,
- Product Support Assessment Team,
- FY2010 *National Defense Authorization Act*, Section 805, and
- HASC Panel on Defense Acquisition Reform (Interim Findings and Recommendation).

These major reform efforts are summarized below.

Acquisition Reform Initiatives

The *Weapon Systems Acquisition Reform Act of 2009 (WSARA)*

On May 22, 2009, President Obama signed the *Weapon System Acquisition Reform Act*, marking an important step in the procurement reform process. The objective of the 2009 *Weapons System Acquisition Reform Act* is to eliminate some of the waste and inefficiency in defense projects. The *Reform Act* targeted improving the DoD’s ability to efficiently and effectively provide the warfighter with necessary weapons and equipment through the following provisions (Levin, 2009):

- Assessing the extent to which the Department has in place the systems engineering capabilities needed to ensure that key acquisition decisions are supported by a rigorous systems analysis and systems engineering process.
- Establish organizations and develop skilled employees needed to fill any gaps in such capabilities.

- Require the DoD to reestablish the position of Director of Developmental Test and Evaluation.
- Require the military departments to assess their developmental testing organizations and personnel, and address any shortcomings in such organizations and personnel, making it the responsibility of the Director of Defense Research and Engineering (DDR&E) to periodically review and assess the technological maturity of critical technologies used in MDAPs. The DDR&E's determinations would serve as a basis for determining whether a program is ready to enter the acquisition process.
- Establish a Director of Independent Cost Assessment to ensure that cost estimates for major defense acquisition programs are fair, reliable, and unbiased.
- Require the Joint Requirements Oversight Council (JROC) to seek and consider input from the commanders of the combatant commands in identifying joint military requirements.
- Require consultation between the budget, requirements and acquisition stovepipes—including consultation in the joint requirements process—to ensure the consideration of trade-offs between cost, schedule, and performance early in the process of developing major weapon systems.
- Require the completion of a PDR and a formal post-PDR assessment before a major defense acquisition program receives Milestone B approval to ensure a sufficient knowledge base as well as to ensure technological maturity and avoid "a long cycle of instability, budget and requirements changes, costly delays and repeated re-base lining."
- Require the Department of Defense to implement competitive prototyping, dual-sourcing, funding of a second source for next generation technology, utilization of open architectures to ensure competition for upgrades, periodic competitions for subsystem upgrades, licensing of additional suppliers, government oversight of make-or-buy decisions—to maximize competition throughout the life of a program, periodic program reviews, and requirement of added competition at the subcontract level.
- Enhance the use of *Nunn-McCurdy* as a management tool by requiring MDAPs that experience critical cost growth: (a) be terminated unless the Secretary certifies (with reasons and supporting documentation) that continuing the program is essential to the national security and the program can be modified to proceed in a cost-effective manner; and (b) receive a new Milestone Approval (and associated certification) prior to the award of any new contract or contract modification extending the scope of the program.
- Prohibit systems engineering contractors from participating in the development or construction of the major weapon systems on which they are advising the Department of Defense.
- Require tightened oversight of organizational conflicts of interests by contractors in the acquisition of major weapon systems.
- Establish an annual awards program—modeled on the Department's successful environmental awards program—to recognize individuals and teams who make significant contributions to the improved cost, schedule, and performance of defense acquisition programs.

Congress intended to build on and strengthen its previous reform efforts by tightening regulations designed to foster competition and by requiring termination of programs that run over-budget and attempt to change how major defense acquisition programs are acquired. Congress and the administration heralded the legislation as a much-needed fix to the Pentagon's acquisition process.

The DoD is moving forward with *WSARA* implementation. As those efforts unfold, some suggest that *WSARA* may exacerbate some of the problems the act was intended to rectify by duplicating existing regulations with additional layers of bureaucracy and an oversight that could slow even further a system that already lacks agility and responsiveness (Erwin, 2010).

Furthermore, the act appears to increase the probability that weapons programs that breach *Nunn-McCurdy* legislation will be terminated when they exceed their projected costs by 25%. Missing from the act is any acknowledgment of the DoD's role in making changes to programs, adding requirements, and/or demanding additional conditions on the development of the weapons system that caused costs to rise (Goure, 2009).

The above concerns may be valid, but with only a year of implementation, little empirical evidence exists to ascertain the effectiveness of *WSARA* in enhancing agility and affordability. The act does provide guidance concerning COCOM engagement with the requirements process, broader competition (and enablers) for system development and upgrades, and enhanced acquisition workforce. The potential benefits of the structural aspects of the act may be illuminated by comparing the DoD's recent efforts on the Mine Resistant Ambush Protected (MRAP) vehicles and the ongoing Joint Light Tactical Vehicle (JLTV) program.

In February 2005, the United States Marine Corps (USMC) identified an urgent operational need in Iraq and Afghanistan for armored tactical vehicles to increase crew protection and mobility of Marines operating in areas containing improvised explosive devices (IEDs), rocket-propelled grenades, and small arms fire (Sullivan, 2008). The ensuing MRAP acquisition program established minimal operational requirements and relied heavily on commercially available products (Sullivan, 2008). The development of MRAP significantly reduced the IED threat to United States ground forces operating in Iraq, swiftly and effectively. Within two years of program start, more than 16,000 vehicles were produced at rates occasionally exceeding 1,000 vehicles per month (Sullivan, 2008).

In comparison, the Joint Light Tactical Vehicle program was developed in response to similar threats and was intended to be the successor to the 11 different versions of the High Mobility, Multi-Wheeled Vehicle (HUMMWV) (Feickert, 2009). In late 2006, the DoD launched a major procurement initiative. Seven industry teams conducted initial design efforts: AM General and General Dynamics Land Systems, BAE Systems, Cadillac Gage, Force Protection, Lockheed Martin, Oshkosh and Protected Vehicles. The program acquisition strategy employed competitive prototyping, which resulted in three teams brought forward into a prototype phase (as contracted to the MRAP that were immediately procured).

As MRAP was fielded and the JLTV prototypes emerged, military leaders refined their requirements for JLTV, requesting a tactical mobile vehicle with traditional combat capabilities. The extended prototype phase afforded the Services the opportunity to exert requirements creep. As a result, payload requirements have increased for most of the Army variants, including the utility vehicle, up 200 pounds to 5,500; the command vehicle, up 880 pounds to 5, 100; and the ground maneuver vehicle, up 400 pounds to 6, 700 (Osborn, 2007). Other added requirements include:

- Make 30 kilowatts of electricity,
- Tow a trailer with ammunition and supplies,
- Carry more ammo,
- Increase fuel efficiency to 90 ton-miles per gallon at maximum gross vehicle weight,
- Be equipped with the A-kit armor and add on option to add a B-kit that includes a gunner's protective shield, and
- Be able to run on two flat tires and keep going after a small-arms attack.

Unlike the MRAP program, the JLTV program did not integrate the available components and COTS subsystems early in the process. The Services continue to modify subsystems to meet additional requirements or develop new technologies and lengthen the system's acquisition schedule. This contrast in approaches to requirements determination and acquisition strategy results in the development timelines shown in Figure 1.

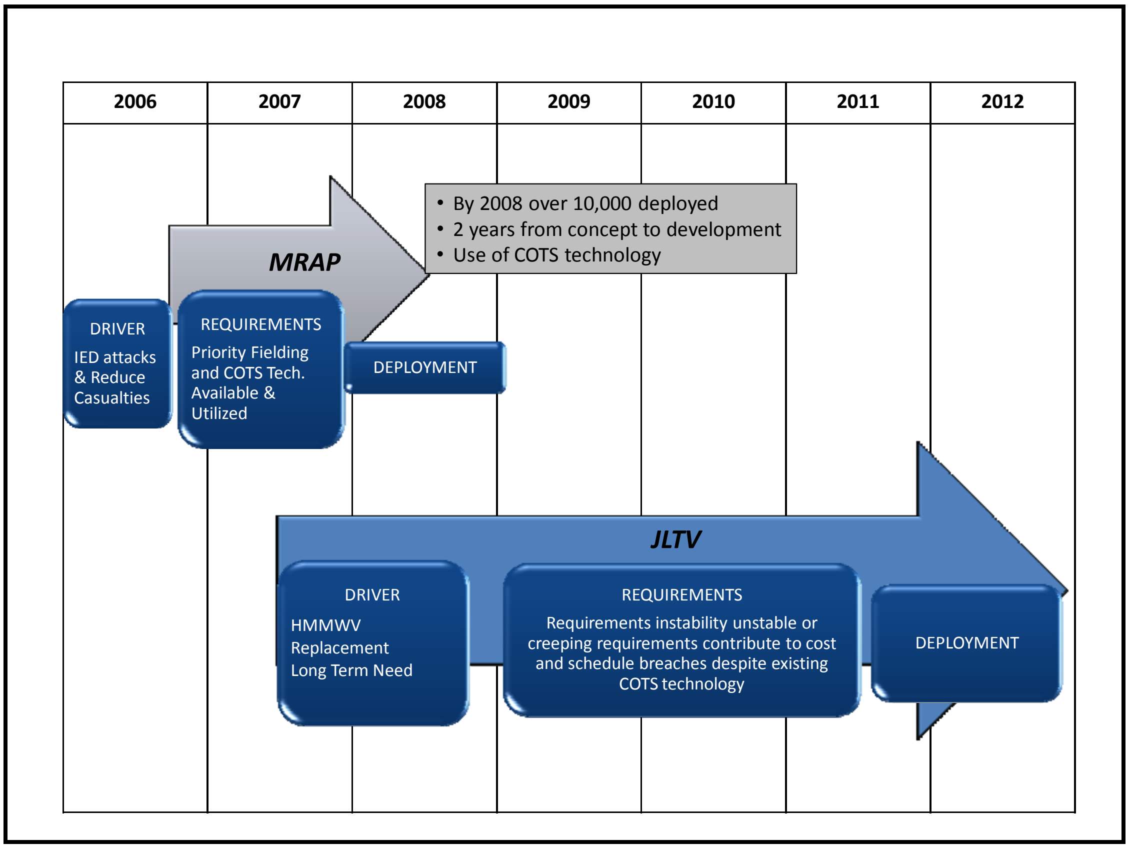


Figure 1. MRAP & JLTV Program Timelines

The extended development timeline for JLTV results in additional requirements for the Army to reset/recapitalize HMMWVs returning from Iraq as a gap filler. The recent HASC Panel on Acquisition Reform chided the DoD on its extended development timeline; however, the DoD and Congress need to evaluate and rationalize their desires for rapid acquisition and competitive prototyping.

The USMC Unmanned Aerial Re-supply (UAR) effort may provide an illustrative example for future consideration. The UAR program was initiated in spring 2008 in response to an urgent operational requirement to provide vertical supply distribution in Iraq with requirements focused on lift capability and endurance. The program will become a force multiplier and lessen casualties by reducing USMC ground convoy logistics requirements. The USMC awarded a competitive fly-off of existing capabilities in 2009, which will be followed by industry proposals. The USMC intends to select a UAR vehicle by late 2010 and obtain industry-provided service capability by early 2011. From requirements to capability, a total time of approximately 28-30 months is achievable. Naval Air Systems Command is assessing acquisition strategy alternatives that include a traditional development and production options. Such an approach would delay fielding existing capabilities.

Insourcing

By the *Weapon System Acquisition Reform Act of 2009*, the United States Congress reversed two decades of acquisition workforce reduction. The act includes explicit requirements to "strengthen the DoD program management, systems engineering, cost analysis, and contract administration workforce." The act also requires the DoD to "insource" program management and acquisition support functions that had been previously contracted out.

To fulfill the requirements of the act, the DoD resourced 20,000 additional acquisition positions in its FY10 budget. The Office of the Secretary of Defense also issued guidance on the insourcing process, including specific acquisition functions and broader contract services. The guidance anticipates the insourcing process will proceed through 2012 with concentration on acquisition management positions.

In a recent paper by the Federal Acquisition Innovation and Reform Institute (FAIR), the authors advise a deliberate and systematic approach to insourcing, based on facts and analysis, to include business case analysis and full consideration of inherently governmental positions, as well as core competencies. The paper further recommends careful assessment of federal pay scales to ensure competitive recruiting (Sharma, 2009).

The concerns noted by FAIR appear justified based upon the recent injunction by the Federal District Court of San Antonio to stop an Air Force insourcing of audio/visual support, which had been provided by Rohmann Services, Inc., a small business. The Rohmann Services, Inc., suit contended the Air Force used inaccurate cost estimates to justify the insourcing, and the cost analysis failed to include government overhead, benefits, and overtime (Hendricks). The Air Force now has opted to recomplete the contract of August 2010.

The Defense Science Board (DSB) study on integrating commercial systems into the DoD provides additional insights into the future requirements of the DoD workforce. As the DoD and Congress move to accelerate development timelines, one reasonable approach is greater reliance on commercially available systems and subsystems (DSB, 2009). The DSB study highlighted the wide dispersion of how commercial solutions are acquired across the DoD. In some cases, DoD design authorities rigidly enforced long-standing (military) design specifications, which drove major changes to COTS equipment. If the DoD is to capitalize on a competitive commercial market, insourcing efforts must allow for commercially savvy acquisition personnel to join the federal workforce.

Finally, the focus on acquisition "insourcing" has been extended by the Air Force to include weapon system sustainment tasks. Recent statements by the Secretary of Air Force indicate a clear desire to "insource" both product support integration and supply chain management functions. This stated desire is based on a perception that the Service is losing its product support capability. These statements are inconsistent with current DoD policy, FY10 *NDAA* Section 805 provisions, and best practices, as demonstrated by ongoing performance-based partnership programs.

DoD Product Support Assessment Team (PSAT)

In September 2008, a DoD Product Support Assessment Team (PSAT) was formed to analyze DoD product support enterprise activities, performance, and cost, and to outline actions for a way ahead for life cycle product support management (Deputy Under Secretary of Defense, 2008). The team completed an assessment of overall and program-specific progress in capturing, managing, and reducing weapon system support costs while maintaining necessary readiness levels and mitigating sustainment risk (PSAT, 2009).

The PSAT found that DoD product support is characterized by a dependence on transactional-based systems and processes, inadequate human capital, organizational challenges, and a lack of shared goals (PSAT, 2009). Additionally, the PSAT study found that performance-based (outcome-based) product support strategies with government-industry partnering, have delivered superior materiel readiness across multiple weapon system applications. The PSAT provided eight principle recommendations (PSAT, 2009):

- Adopt a product support business model that drives cost effective performance and capability for the warfighter across the weapons system life cycle and enables the most advantageous use of an integrated defense industrial base;
- Align and expand the collaboration between government and industry that produces best-value partnering practices, both within and beyond the depots;
- Connect platform product support strategies to enterprise supply chain approaches that produce best value across the DoD components;
- Improve weapons system governance so sustainment factors are better considered early and consistently across a weapons system life cycle;
- Develop an overarching DoD sustainment metric and management strategy for life cycle product support that strengthens formal data collection and analysis capabilities while providing insight and learning to support life cycle planning and operational management;
- Make life cycle affordability a core business process for all communities and stakeholders involved in system acquisition and sustainment;
- Clarify and codify policies and procedures pertaining to the use of analytical tools in the life cycle product support decision-making process; and
- Integrate product support competencies across the logistics and acquisition workforce domains to institutionalize successful traits of an outcome-based culture.

The DoD is moving forward with implementing the PSAT recommendations. As that implementation proceeds, the requirements for greater scrutiny and competition for service contracts are also being implemented. These simultaneous implementations create conflicting pressures that are evidenced by:

- Extended timelines to conduct, review, and approve business case analyses;
- Extended peer reviews at the Service and OSD level; and

- Reduce contract lengths to enable continuous re-competitions.

As these pressures unfold, the DoD is refining its performance-based partnerships for programs such as the C-17 and F-22. The refinements of the C-17 and F-22 platform-level PBL sustainment strategies were both preceded by business case analyses (BCAs). Both BCAs documented difficulties in characterizing future performance for sustainment options, accurately capturing government costs, and estimating potential transition costs. As a result, the USAF requested an independent assessment of both BCAs by OSD.

Both programs have been designated as lead programs for implementing the PSAT recommendations. The work share between government and industry is currently being evaluated for both programs, and transition plans are being developed. The key issue to be addressed through the transition is to retain an outcomes-based strategy for both programs as work (and responsibility) is re-aligned to the Air Force.

Life Cycle Management and Product Support: The *National Defense Authorization Act*, Section 805

Section 805 of the 2010 *Authorization Act* provided statutory guidance on life cycle management, including the requirement for a product support manager for all major systems, maximize competition at the system, subsystem, and component level, and outcome-focused product support strategies. The product support manager shall be responsible for (House of Representatives, 2009):

- Development and implementation of a comprehensive product support strategy for the weapon system;
- Providing appropriate cost analysis to validate the product support strategy, including cost-benefit analysis as outlined in *Office of Management and Budget Circular A-94*;
- Assuring achievement of desired product support outcomes through development and implementation of appropriate product support arrangements;
- Adjusting performance requirements and resource allocations across product support integrators and product support providers as necessary to optimize implementation of the product support strategy;
- The periodic review of product support arrangements between the product support integrators and product support providers to ensure the arrangements are consistent with the overall product support strategy; and
- Revalidating any business-case analysis performed in support of the product support strategy.

Section 805's enactment is intended to enhance competition while leveraging industry and government capabilities to avoid high product-support costs while improving performance. More importantly, it begins to attack an important issue of acquisition reform: accountability. The DoD is currently preparing its implementation plan and report.

Based on recent market data, the Section 805 focus on competition for product support is well founded. Figure 2 presents the competitive nature of the sustainment market. As shown, several elements of sustainment are intensively competitive; however, spare parts continue to be a relatively non-competitive market. These data suggest that the

DoD should focus on developing alternate sources for critical parts, rather than shortening the contract of existing PBLs to foster more recompetes.

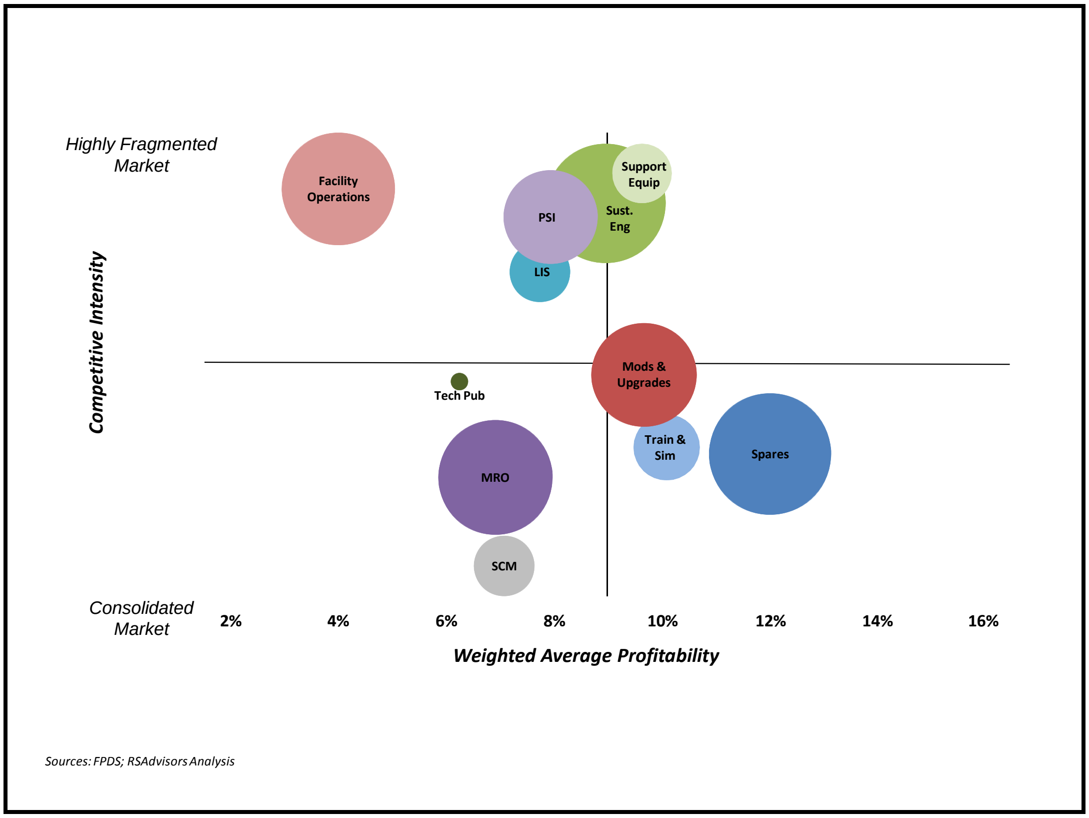


Figure 2. Competitive Product Support

The Defense Logistics Agency (DLA) is aggressively moving forward to achieve end-to-end supply chain management and foster greater competition. The Department of Defense (DoD) directs the largest and most complex supply chain in the world. The DoD spends at least $150 billion a year on goods and services and their delivery to end users (Daily, 2005). DLA manages an inventory of tens of thousands of items, valued at approximately $80 billion. The DoD supply chain also includes hundreds of original equipment manufacturers, many of which not only produce new items but also help support systems and platforms in the field (DLA, 2006).

USA's BRAC 2005 process recommended that the US Defense Logistics Agency privatize a series of product commodities, and eliminate the government's wholesale stock in key areas. These Commodity Management Privatization (CMP) activities take place with goals that include improved delivery and management, lower, more transparent cost of ownership, and a Strategic Supplier Alliance—an Umbrella partnering agreement defining mutually beneficial objectives to improve logistics operations and warfighter support (DLA, 2006).

The Defense Supply Center Columbus (DSCC, DLA), the supply chain manager for tires, competitively awarded a contract worth $368 million for aviation tires with Michelin for a base period of five years and an additional five-year option period, worth more than $300 million. Under this contract, Michelin has the responsibility for procurement, storage, and distribution of these tires, as well as the disposal of scrap tires for CONUS locations and pick-up of re-treadable tires for CONUS and OCONUS locations.

The privatization effort of aircraft tires continues to save the customer and the DoD money on costs associated with procurement, storage, maintenance, and disposal by placing these requirements on Michelin. This privatization effort provides the warfighter direct benefits as they now receive their supplies from Michelin, who provides direct delivery of these commodities from their stock. As of Calendar 2009, Michelin and the DLA have delivered 9,235 orders for 26,636 tires, with the average delivery time of 1.97 days and a 98.9% on-time delivery rate. Program-to-date, the on-time delivery rate has been 98.5% and a 100% fill rate—with no backorders incurred and with a project annual savings of $46 million (NSSC, 2009).

HASC Panel on Acquisition Reform

The House Armed Services Committee (HASC) Panel on Defense Acquisition Reform was appointed by Chairman Ike Skelton and then-Ranking Member John McHugh in March 2009 to carry out a comprehensive review of the defense acquisition system. The HASC review was motivated by the lack of responsive within the DoD acquisition system to today's mission needs, not rigorous enough in protecting taxpayers, and not disciplined enough in the acquisition of weapons systems for tomorrow's wars (HASC, 2010). The Panel took a year to perform its review, holding 12 hearings and numerous briefings covering a broad range of issues in defense acquisition.

The Panel found that while the environment of defense acquisition has significantly changed, the defense acquisition system has not, with the current acquisition system structured largely for the acquisition of weapon systems at a time when the acquisition of services, and of information technology, represents a much larger portion of the DoD budget. The Panel also reported that there is little commonality across the defense acquisition system with the acquisition of weapon systems, commercial goods, commodities, services, and information technology. The Panel recommended the following:

- A Rapid Acquisition Fielding Agency be created to meet urgent operational needs, and the "DoD and Congress should not accept development timelines routinely measured in double digits."
- Recognize accelerated life cycle for IT acquisition (including embedded software). Defense related IT systems are typically taking 2-3 years to deliver; a time-frame that ensures the technology is two to three generations out of date by the time it is delivered.
- Achieve auditable financial systems. The Panel recommended The Under Sectary of Defense (Comptroller) and the Comptrollers of the military departments should rely more on individual obligation and expenditure plans for measuring program financial performance.
- Expand outreach to commercial/small business. The Panel recommended improving competition and access to more innovative technology by utilizing more of the industrial base, especially small and mid-tier businesses.
- Enhance requirements process and analytics with "greater emphasis on the up-front market analysis to best leverage limited funds by buying good solutions from the commercial market when they are available, and husbanding resources for development for instances when there is no other provider."

These recommendations directly enhance effects-based requirement, commercial-like R&D (for IT systems), and a healthy, competitive industrial base. The effect on these recommendations is dependent upon DoD implementation.

Initial Assessment

As summarized, Congress and the DoD initiated significant acquisition reform efforts simultaneously. As noted, several of the reform provisions are not strategically aligned. Furthermore, in some cases, DoD implementation has extended development and procurement timelines, demonstrating a lack of agility. Finally, DoD and congressional desires to expand the industrial base (to include more innovative, mid-sized companies) must be enabled by life cycle processes that foster greater private-sector involvement. Current reform efforts to expand oversight, extend development and test, and insource may actually inhibit greater commercial involvement.

Based upon these considerations, an initial assessment of the effect of current reform efforts on agility and affordability is shown in Table 2. As presented, across the numerous reform efforts, positive steps are being taken to address warfighter-focused requirements, commercial-like R&D for IT systems, and outcome-based sustainment. Unfortunately, these positive indicators are offset by other aspects of reform such as increased oversight, additional milestones, and expanded testing.

Table 2. Initial Assessment of Reform Efforts

Key Characteristics	WSARA	Insourcing	PSAT	Section 805	HASC Panel
Effects-based requirements	✓		✓		✓
Commercially- driven R&D	-				✓
Outcome-based partnership product support	-	-	✓	✓	-
Competitive industrial base	✓	-	✓	✓	✓
Enhanced DoD workforce	✓	-	✓	✓	✓

LS10-0102-02 1

Recommendations

Based on this initial assessment of DoD acquisition reform policy efforts and their potential impact of those efforts on the inherent, structural incentives that are embedded in DoD life cycle processes, the following policy actions are recommended:

1. Accelerate requirement process reform: Incentivize industry to control requirements creep, select mature technologies for product integration, and develop solutions in an incremental and timely fashion with the timely and collaborative development of requirements and potential solutions at the commencement of the specific program. Increase requirements acceleration by increasing the reliance on commercially available systems and subsystems.

15. Strategically balanced insourcing and desire for competitive industrial base: Implement and enforce a deliberate and systematic approach to insourcing based on facts and analysis, including business case analysis and the full consideration of inherently governmental positions, as well as core competencies. Simultaneously, the DoD needs to continue to cultivate partnerships with industry. If the DoD is to capitalize on a competitive commercial market, insourcing efforts must allow commercially knowledgeable acquisition personnel to join the federal workforce.

16. Develop competitive sustainment framework consistent with *NDAA*, Section 805: Maximize the value of Department of Defense funding by providing the best possible product support outcomes at the lowest operations and support cost. This is achieved by providing guidance on life cycle management, to include the requirement for a product

support manager to consider competitive alternatives at the system, subsystem, and component level every five years.

17. Transition fielded systems to outcome-based sustainment: Implement an outcomes-based sustainment model and strengthen total life cycle systems management, Depot Maintenance Partnering, and Condition-Based Maintenance, enabling end-to-end weapon system support,providing required readiness at reduced costs.

As the United States advances into the 21st century, the DoD will continue to be faced with the challenge of maintaining a persistent expeditionary military presence while engaged in a long-term conflict. Victory in part, will be measured by the DoD's ability to effectively sustain and maintain equipment, while concurrently preserving its ability to display flexibility in meeting the evolving and changing operational conditions of irregular warfare and stateless actors. Furthermore, both the global economic environment and the requirements associated with growing competition for scarce resources generate conditions in which the DoD will have to do more with less.

References

Daily, D.I. (2005, November). *US Defense Logistics Agency lays out FY 2006 transformation roadmap*. Retrieved January 2009, from http://www.defenseindustrydaily.com/us-defense-logistics-agency-lays-out-fy-2006-transformation-roadmap-01422/

Defense Science Board (DSB). (2009). *Buying commercial; gaining the cost.schedule/benefits for defense systems.* Washington DC: DoD.

Erwin, S. (2010, January). *Acquisition reform act: The backlash has begun*. Retrieved March 2010, from http://www.nationaldefensemagazine.org/ARCHIVE/2010/JANUARY/Pages/AcquisitionReformActTheBacklashHasBegun.aspx

Feickert, A. (2009). *The Joint Light Tactical Vehicle; issues for Congress*. Washington, DC: Congressional Research Service.

Gates, R. (2009). *Meeting of the Economic Club of Chicago.* Remarks by Secretary Gates at a meeting of the Economic Club of Chicago. Chicago: US DoD News Transcript.

Goure, D. (2009, December 3). *Congress deforms the acquisition process*. Retrieved February 2010, from http://www.lexingtoninstitute.org/congress-deforms-the-acquisition-process?a=1&c=1171

Goure, D. (2009, December 15). *Air Force should stick with performance-based support of F-22.* Retrieved January 2009, from http://www.lexingtoninstitute.org/air-force-should-stick-with-performance-based-support-of-f-22?a=1&c=1171

House Armed Services Committee (HASC). (2010, March 10). *House Armed Services Committee Panel on Defense Acquisition Reform: Interim findings and recommendations*. Retrieved March 2010, from http://armedservices.house.gov/pdfs/DARPanelReport/PanelReportInterim030410.pdf

House of Representatives. (2009, June 2, 18). *2010 defense appropiations bill* (HR 2647). 111th Congress. Washington, DC: US GPO.

Kratz, L., & Bradd, B. (2010). Achieving outcomes-based life cycle managment. *Defense Acquisition Review Journal*, *10*(1), 45-64.

Levin, C. (2009, February 24). *Summary of the Weapon Systems Acquisition Reform Act of 2009.* Retrieved March 6, 2010, from http://levin.senate.gov/newsroom/release.cfm?id=308525

NSSC. (2009, June 16). *Nominantion for the 2009 Secretary of Defense Performance Base Logistics Awards Program*. Retrieved February 2010, from Unversity of Tennessee, Knoxville, http://thecenter.utk.edu/images/Users/30/pblawards/Component2009/NavyTirePerformance.pdf

Osborn, K. (2007, July). *U.S. delays Humvee replacement; MRAP pushes JLTV arrival two years to 2012*. Retrieved March 2010, from http://www.generaltacticalvehicles.com/U.S.%20delays%20Humvee%20replacement.html

PSAT. (2009, Novemebr). *DoD product support assessment; shaping next generation life cycle sustainment.* Retrieved January 2009, from https://acc.dau.mil/CommunityBrowser.aspx?id=328610

Reniassance Strategic Advisors. (2009). *Competitor analysis phase II—task E for Lockheed Martin logistics and sustainment.* Arlington, VA: Reniassance Strategic Advisors and Lockheed Martin.

Sullivan, M. (2008, July). *Rapid acquisition of mine resistant ambush protected vehicles.* Retrieved January 2009, from http://www.gao.gov/new.items/d10155t.pdf

Acquisition of Mine Resistant Ambush Protected (MRAP) Armored Vehicles: A Case Study

Jacques S. Gansler—The Honorable Jacques S. Gansler, former Under Secretary of Defense for Acquisition, Technology, and Logistics, is a Professor and holds the Roger C. Lipitz Chair in Public Policy and Private Enterprise in the School of Public Policy, at the University of Maryland, and is the Director of both the Center for Public Policy and Private Enterprise and the Sloan Biotechnology Industry Center. As the third-ranking civilian at the Pentagon from 1997 to 2001, Professor Gansler was responsible for all research and development, acquisition reform, logistics, advance technology, environmental security, defense industry, and numerous other security programs.

Before joining the Clinton Administration, Dr. Gansler held a variety of positions in government and the private sector, including Deputy Assistant Secretary of Defense (Material Acquisition), Assistant Director of Defense Research and Engineering (electronics), Executive Vice President at TASC, Vice President of ITT, and engineering and management positions with Singer and Raytheon Corporations.

Throughout his career, Dr. Gansler has written, published, and taught on subjects related to his work. Gansler recently served as the Chair of the Secretary of the Army's Commission on Contracting and Program Management for Army Expeditionary Forces. He is a member of the Defense Science Board, and also a member of the National Academy of Engineering and a Fellow of the National Academy of Public Administration. Additionally, he is the Glenn L. Martin Institute Fellow of Engineering at the A. James Clarke School of Engineering, an Affiliate Faculty member at the Robert H. Smith School of Business and a Senior Fellow at the James MacGregor Burns Academy of Leadership (all at the University of Maryland). For 2003–2004, he served as Interim Dean of the School of Public Policy. For 2004–2006, Dr. Gansler served as the Vice President for Research at the University of Maryland.

jgansler@umd.edu
Center for Public Policy & Private Enterprise,
School of Public Policy University of Maryland
2101 Van Munching Hall
Room 2211B
College Park, MD 20742

William Lucyshyn—William Lucyshyn is the Director of Research and Senior Research Scholar at the Center for Public Policy and Private Enterprise, in the School of Public Policy, at the University of Maryland. In this position, he directs research on critical policy issues related to the increasingly complex problems associated with improving public sector management and operations, and how government works with private enterprise.

Current projects include: modernizing government supply chain management, identifying government sourcing and acquisition best practices, and department of defense business modernization and transformation. Previously, Mr. Lucyshyn served as a program manager and the principal technical advisor to the Director of the Defense Advanced Research Projects Agency (DARPA) on the identification, selection, research, development, and prototype production of advanced technology projects.

Prior to joining DARPA, Mr. Lucyshyn completed a 25-year career in the US Air Force. Mr. Lucyshyn received his Bachelor's degree in Engineering Science from the City University of New York, and earned his Master's degree in Nuclear Engineering from the Air Force Institute of Technology. He has authored numerous reports, book chapters, and journal articles.

lucyshyn@umd.edu
Center for Public Policy & Private Enterprise,

School of Public Policy University of Maryland
2101 Van Munching Hall
Room 2211B
College Park, MD 20742
Phone: 301-405-8257

William Varettoni—William Varettoni is a PhD student at the Maryland School of Public Policy focusing on international development and security. Prior to joining the Center for Public Policy and Private Enterprise, Varettoni was a Foreign Affairs Analyst with the US Department of State and covered NATO, OSCE, and Ukraine portfolios. In Ukraine, Mr. Varettoni served as a Peace Corps Volunteer, researched as a Fulbright Scholar, and worked as a field reporter for the US Embassy. In Tanzania and Senegal, he performed field evaluations on US Agency for International Development-funded development projects. William holds an MS in Sustainable Development and Conservation Biology from the University of Maryland, and a BBA in Accounting, Computer Applications, and Environmental Science from the University of Notre Dame.

wvarettoni@gmail.com
Center for Public Policy & Private Enterprise,
School of Public Policy University of Maryland
2101 Van Munching Hall
Room 2211B
College Park, MD 20742
Phone: 301-405-8754

Abstract

Improvised explosive devices (IEDs) have been responsible for the majority of the casualties suffered by US forces in military activities during Operation Iraqi Freedom in Iraq and Enduring Freedom in Afghanistan. A family of heavily armored vehicles, that had been in use in very limited numbers for specialized missions, quickly gained a reputation for providing superior protection for their crews, leading to a suggestion that similar vehicles might be a better alternative for transporting troops in combat than Up-Armored High Mobility Multipurpose Wheeled Vehicles (HMMWVs). The decision was made to develop and procure a fleet of combat vehicles capable of sustained operations in a chaotic, mine-infested, non-linear battlespace. These vehicles were identified by the acronym MRAP—Mine-Resistant Ambush-Protected. Due to urgent fielding requirements, the MRAP program pursued a very aggressive schedule, while at the same time grappling with a significant number of unknowns, such as the total quantity required and the long-term sustainment strategy. The MRAP program's successes highlight the limitations of both the traditional acquisition system and the ad hoc organizations that cater to rapid acquisitions. This case clearly identifies the need for a separate rapid acquisition agency within the Department of Defense.

Executive Summary

As the largest and fastest industrial mobilization since World War II, the Mine-Resistant Ambush-Protected (MRAP) vehicle program is a testament to the scale and efficiency possible when government and industry collaborate with a sense of urgency, patriotism, and pragmatism. Public pressure over rising casualty numbers, intense political scrutiny, and support from the highest levels of government all combined into a set of unique circumstances. Given great uncertainty in the nature of future security issues, however, urgent and unforeseen needs will frequently press the procurement system. The MRAP program, precisely because of its size and scope, brings into sharp relief the merits and deficiencies of the current system for rapid acquisitions.

Improvised explosive devices (IEDs) were the number one killer of troops in Iraq and Afghanistan. In response to increasing IED attacks, the Defense Department began adding armor kits to High Mobility Multipurpose Wheeled Vehicles (HMMWVs) and procuring up-armored HMMWVs. Even with added armor, the HMMWV's flat bottom made it vulnerable to buried IEDs. Beginning in early 2005, field commanders made formal requests for MRAP vehicles. These vehicles—essentially armored trucks—have V-shaped hulls and high ground clearance to deflect and diffuse bomb blasts. A small number of MRAPs were in theater as part of explosive ordinance disposal teams. They had a reputation for survivability—about 400% safer than a HMMWV. Despite earlier requests for MRAPs to be procured for use in combat missions, it would take until November of 2006—almost two years later—for MRAP requirements to be validated and a request for proposals to be released.

The MRAP program had one primary objective—to field the maximum number of survivable vehicles in the shortest period of time. Cost and all other concerns were explicitly secondary. Using funds from supplemental war requests outside the normal budget process, Congress flooded the program with money, often providing amounts in excess of initial requests. Secretary Gates declared MRAPs the military's highest priority acquisition and put them at the head of the queue for scarce steel armor and tires.

The MRAP program sought commercial off-the-shelf technology, with minimal requirements. It provided production contracts to all manufacturers that could meet minimum automotive and survivability standards. In fact, production orders were signed even before initial testing was completed, in order to prime industry. This was risk acceptance by the DoD on two fronts—it was agreeing to buy vehicles it might not ever field, and it was committing to flood the theater with several different MRAP variants (each with its own parts, support, and logistics needs). The DoD provided incentives to vendors and subsidized capacity expansion. As testing progressed, designs were modified for subsequent models. Manufacturers' representatives were embedded at the Aberdeen testing site to speed communication back to the production line. User feedback from the field was also fed into ongoing production modifications. The entire acquisition process was compressed for MRAPs. Testing, production, fielding, and feedback were all done concurrently and continuously.

The MRAP program's successes highlight the limitations of both the traditional acquisition system and the ad hoc organizations that cater to rapid acquisitions. The traditional acquisition system is ill-suited to rapid acquisitions, and its bureaucratic processes can at times resemble the convoluted means used in a "Rube Goldberg" machine. It is linear, stove-piped, and risk-adverse. Because of extraordinary levels of support at the highest levels, the MRAP program was able to extend deadlines, or bypass a number of bureaucratic processes. The program office is still in the process of completing some of the pre-production paperwork processes, even though production has finished. This suggests that perhaps some of the bureaucratic requirements are not particularly relevant to rapid acquisitions. The experience of MRAP procurement demonstrates that there is ample room to streamline the process without sacrificing results and accountability.

Only because of media scrutiny, Congressional pressure, and the personal involvement of Secretary Gates could the MRAP program proceed so quickly once it received approval from the requirements process. A high-level MRAP Task Force convened regularly to get all decision makers into the same room to solve problems. The nature of future combat is likely to require more—not less—fielding of urgent and unanticipated equipment for the troops. Congressional pressure and involvement of the Secretary of Defense cannot be expected to quickly materialize to push through urgent requests. So, the current system is unlikely to ever reproduce the impressive results of the MRAP program without serious reform.

Rapid acquisitions will always be in a disadvantaged position if they are forced to work within the same system (and compete for the same funds) as traditional, deliberate acquisitions. We recommend a stand-alone rapid acquisition organization within the DoD, with requirements different from the existing deliberate acquisition process. It should have its own bankable funding stream.

Absent creation of a separate organization, it is clear that rapid acquisition projects would benefit from a senior-level champion to shepherd them through the acquisition system, ensuring they do not get sidelined in one of the myriad stovepipes. Better still would be a task force periodically assembled with relevant stakeholders empowered with decision authority to solve problems and clear delays in real time.

THIS PAGE INTENTIONALLY LEFT BLANK

Panel #12 – Cost Containment in an Evolutionary Acquisition Environment

Wednesday, May 12, 2010	
3:30 p.m. – 5:00 p.m.	**Chair: Lorna B. Estep**, Executive Director, Air Force Global Logistics Support Center, Air Force Materiel Command ***The Correct Use of Subject Matter Experts in Cost Risk Analysis*** **Richard Coleman** and Peter Braxton, TASC, Inc., and Eric R. Druker, BAH ***Addressing Cost Increases in Evolutionary Acquisition*** **Doug Bodner**, Farhana Rahman and Bill Rouse, Georgia Institute of Technology ***It's Time to Take the Chill Out of Cost Containment and Re-energize a Key Acquisition Practice*** **Robert Tremaine** and Donna Seligman, Defense Acquisition University

The Correct Use of Subject Matter Experts in Cost Risk Analysis

Richard L. Coleman—Richard L. Coleman is a Naval Academy graduate, received an MS with Distinction in Operations Research from the US Naval Postgraduate School and retired from active duty as a Captain, USN, in 1993. His service included tours as Commanding Officer of *USS Dewey (DDG 45)*, and as Director, Naval Center for Cost Analysis. At TASC and Northrop Grumman, he worked extensively in cost, CAIV, and risk for the Missile Defence Agency (MDA), Navy ARO, the intelligence community, NAVAIR, and DD(X). He has supported numerous ship programs, including the DDG 1000 class, DDG 51 class, Deepwater, LHD 8 and LHA 6, LPD 17 class, *Virginia* class SSNs, CVN 77, and CVN 78. He was the Director of Independent Cost Estimation for Northrop Grumman Information Systems and conducted Independent Cost Evaluations on Northrop Grumman programs. He is currently at TASC, Inc., as a PM for cost and risk. He has more than 75 professional papers to his credit, including five ISPA/SCEA Best Paper Awards and two ADoDCAS Outstanding Contributed Papers. He was a senior reviewer for Cost Programmed Review Of Fundamentals (CostPROF) and Cost Estimating Body of Knowledge (CEBoK) and lead author of the Risk Module. He has been Regional and National VP of the Society of Cost Estimating & Analysis (SCEA) and is currently a Board Member. He received the SCEA *Lifetime Achievement Award* in 2008.

Richard L. Coleman (TASC)
6211 Elati Ct, Alexandria, VA, 22310
richard.lewis.coleman@gmail.com
703.615.4482

Peter J. Braxton—Peter J. Braxton holds an AB in Mathematics from Princeton University and an MS in Applied Science (Operations Research) from the College of William and Mary. He currently serves as a senior cost and risk analyst for TASC, Inc. He has worked in cost estimating and risk analysis, Cost As an Independent Variable (CAIV), and Target Costing on behalf of the Navy Acquisition Reform Office (ARO), the DD(X) development program, and other ship and intelligence community programs. He has reviewed several high-profile proposals as part of the Independent Cost Estimation (ICE) function for Northrop Grumman. He has over 12 professional papers to his credit, including SCEA/ISPA International Conference award-winners in CAIV (1999) and Management (2005). He served as managing editor for the development of both the acclaimed CostPROF body of knowledge and training course materials and its successor CEBoK. He serves as SCEA's Director of CEBoK and an elected Director, and was named SCEA's 2007 *Estimator of the Year for contributions in Education*. He serves as a TASC Technical Fellow, and received a TASC *President's Award* and a Northrop Grumman *Corporate Contracts, Pricing, & Supply Chain Award* in 2008.

Peter J. Braxton (TASC)
9937 Wood Grouse Ct., Burke, VA, 22015
peter.braxton@tasc.com
703. 449.3623

Eric R. Druker— Eric R. Druker CCE/A graduated from the College of William and Mary with a BS in Applied Mathematics in 2005, concentrating in both Operations Research and Probability & Statistics with a minor in Economics. He is employed by Booz Allen Hamilton as an Associate. He currently is President of the St. Louis SCEA Chapter. In 2009, he was named SCEA's *National Estimator/Analyst of the Year for Technical Achievement*. He currently supports NASA, performing joint cost & schedule risk analysis across a variety of projects. In his previous position as a Technical/Research Lead at Northrop Grumman, he developed the risk analysis process/methodology used to assess financial risk to the company during Independent Cost Evaluations (ICEs). As a part of this, he participated in or led over 30 ICEs on Northrop Grumman proposals. He has been an invited presenter at SCEA, the Naval Postgraduate School's Acquisition

Research Symposium, SCEA, DoDCAS and the NASA PM Challenge. Prior to coming to Booz Allen, he served as lead author of the Regression and Cost/Schedule Risk modules for the Cost Estimating Body of Knowledge (CEBoK) update.

Eric R. Druker (BAH)
4584 Emerald View Ct, Eureka, MO, 63025
druker_eric@bah.com,
314.368.5850

Abstract

Subject Matter Experts (SMEs) are commonly used in cost risk analysis (and in other fields as well) for values that either are not available in historical data or for which no appropriate analogy can be found. Problems commonly arise in two areas in particular: (1) when multiple experts give opinions on a single effect or entity and the inputs are not identical in distribution (which is almost inevitable), and (2) when a single expert provides distributional information that is intractable or suspiciously unlikely in its form (which is common).

This paper will put forward correct solutions in case (1), in which the authors' experience shows that practitioners (and even experts) use incorrect solutions. It is important to note that the commonly exercised incorrect solution underestimates the dispersion, and thus the 80th percentile, in some cases by a large margin. The authors believe that their solution is rare and, further, are unaware of any use of the solution, and will recommend tenets to guide the practitioner. In preparation for the solutions laid out above, the authors will first describe the method of expert-based risk analysis, with the erroneous method for combining SME testimony, and then show the correction. An analytical treatment will quantify the impacts of the erroneous approach. The paper will also explain why the new method of conflating expert assessments is to be preferred to the common Delphi technique, which may fall prey to both anchoring and domination by a vocal minority.

The paper will also briefly address case (2) by presenting common examples of problematic formulations and proposed resolutions. These include intractable specification of a triangular distribution, specification of a discrete categorical distribution when triangular was intended, and specification of a triangular with low and high values that are not the true extremes as well as errors committed by the risk analyst.

In any situation, correct treatment of risk is important. In the current era, with 80th percentiles required for all weapon systems cost estimates by the *Weapon Systems Acquisition Reform Act of 2009*, and budgeting to the 80th percentile as the default practice, the correct determination of the distribution is more important than ever before.

Overview

Expert-based risk methodologies are a common approach to cost risk. Expert-based risk methodologies are defined for the purposes of this paper as follows. Notwithstanding that the cost estimate may be based on actuals, expert-based risk methods rely on elicitation of the parameters of the risk distribution from expert opinion. These parameters are for the distribution of various types of risk such as (typically, but not exclusively) triangles for cost risk, Bernoullis for technical risk and occasional normals.

Single or multiple experts may offer estimates (expert testimony) of a particular risk via some form of parameterization.

This paper will discuss two topics in correction of expert testimony: 1) The “best” approach to converting extrema and quartiles from expert opinion into risk distributions, and 2) The “best” approaches to conflating multiple views of the parameterization of a single risk.

For completeness, the paper will also discuss some difficult characterizations that they have encountered and the approach that they have evolved for “correcting” them. Problems with inconsistent percentiles and problems with unusual characterizations will both be discussed.

This topic was addressed in general in a prior paper by Coleman, Braxton, Druker, Cullis, and Kanick (2009) under the rubric “Omission of Elements of Variability.” A paper by St. Louis, Blackburn, and Coleman (1998) espoused a form of combination of expert testimony that this paper now recommends against.

The “Best” Approach to Converting Extrema and Quartiles from Expert Opinion into Risk Distributions

Correcting Extrema and Quartiles for Truncation

The Problem. Our estimated distributions tend to be “too tight,” as shown by Brown (1973) and Alpert and Raifa (1982). Without feedback, we provide extreme values near the 20^{th} percentile and 80^{th} percentile when we are asked Min and Max. This can be improved, with feedback to the 10^{th} and 90^{th} percentile. This can be improved by asking for more-extreme values. For example, “astonishingly-low-probability outcomes” equate to the 0.1^{th} percentile and 99.9^{th} percentile. Without feedback, we give 25^{th} and 75^{th} quartiles that actually contain only 33% of the outcomes versus the expected 50%. This can be improved with feedback to 43% versus the expected 50%. Independent investigations of this over-tightness are remarkably consistent in the degree to which it occurs, as shown by Brown (1973) and Alpert and Raifa (1982). Our ability to probabilistically characterize the past or future or to estimate our certainty on general-knowledge facts are all about comparable, as noted by Lichtenstein, Fischhoff, and Phillips (1982).

Correcting Extrema and Quartiles. For extrema, assume that experts will return 20th and 80th percentiles when asked for the full range. In other words, when given highs and lows, assume you are getting something more like standard deviations masquerading as extrema; it’s not quite that bad, but it’s close. It’s about 0.316 of the real base (see Appendix A). This could be presumed to improve to 10th and 90^{th}, but only if the experts can be assumed to have gotten specific feedback about their accuracy at this task in the past. Note that this is not the same as saying they are very well qualified; it refers specifically to feedback training. We believe that practitioners have mistaken expertise for being trained and that this is why many practitioners believe experts provide 10th and 90th percentiles. For quartiles, although we don’t typically ask for quartiles, we recommend assuming that a claimed 25-75 inter-quartile range is actually a 33-67 percentile range. This can be improved to a 28-72 range with specific feedback. The two distortions above are not strictly coherent, meaning that they yield different corrections. The full range case is a greater understatement than the inter-quartile case. The wider the confidence interval you ask for, the more the witness will understate it. When given expert testimony, therefore, it is appropriate to correct the testimony by adjusting the standard deviation or the end points using the two general results above, depending on the form given.

Errors of Extrema—Pictorially. The 20th percentile occurs at a point that is 0.316 of the base, so the understatement of experts is on the order of 1/3. Pictorially, then, we are experiencing a reduction in distribution on the order of the blue (claimed) to the white (actual) portrayed in Figure 1. For a tutorial on computing percentiles, see Appendix A.

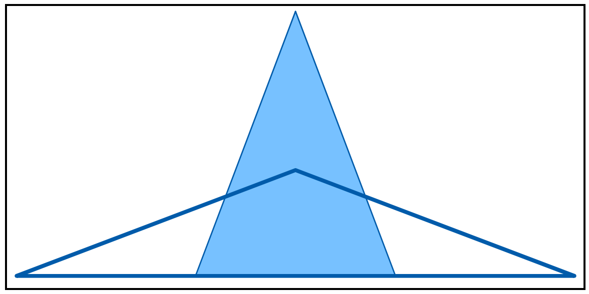


Figure 1. Visualization of Expert Truncation of Dispersion

The "Best" Approaches to Conflating Multiple Views of a Distribution

Conflation

By definition, conflation refers to the combining of different (independent) views of a thing to arrive at a single (better and more complete) view of it. We seek to conflate expert testimony principally because we will arrive at a better estimate for the mean, but, what about the dispersion? Conflation is the most difficult problem for expert-based risk methodologies; this is not immediately obvious, but it is so. Dispersion is, in turn, the hard part of conflation. Ad hoc conflations are often used for k experts each giving estimates for the same risk or WBS element. For example:

1. Use the individual expert testimonies in each run of the Monte Carlo:
   a. Make k random draws from the k different distributions and average them (as done by St. Louis, Blackburn, and Coleman (1998)).
   b. Make k random draws from the k different distributions with correlation and average them.

18. Derive the parameters of a single distribution from the parameters of the expert testimony and then Monte Carlo:
   a. Make a new distribution with i) the mean of the k expert means and ii) the mean of the standard deviations, for normals, as demonstrated by Brown (1973), or the means of the respective end points for triangles [Average the Parameters].
   b. Make a new distribution with the mean of the k experts and the lowest low and the highest high as end points.

19. Sampling has been endorsed by Brown (1973). Sampling is done as follows: for each run of the Monte Carlo, pick the answer from a randomly selected expert who provided testimony.

We will only examine ad hoc methods 1a, 2a and sampling. The others can be inferred from those. Also, note that in backup, we prove that 1b and 2a are equivalent for symmetric triangles, and we speculate that for asymmetric triangles there is no significant difference, and so there is nothing to separate these beyond ease of implementation.

The First Question

The first question in conflation is to determine what we believe to be the underlying model. No single conflation method will work for the two possible scenarios that can confront the estimator, namely single or multiple realities.

"Single Reality." There is a one (typically uni-modal) distribution, which we do not know, but which experts are presumed to know to some degree of accuracy. Examples: What is your estimate for the GNP of Brazil for 2009? How big is a brown bear? What is the range of technical risk for the cost of the engine?

"Multiple Realities." There are k (typically uni-modal) distributions; we generally know neither k nor the individual distributions, but experts are presumed to know at least one each to some degree of accuracy. Examples: How far away is your favorite planet? (There could be up to 9 answers, depending on the inclusion of Pluto and Earth!) How big is a panda? (There is a lesser panda and a greater panda, but we don't happen to know that and fail to specify) What is the cost risk for the engine on the F-35? (There is a main and an alternate engine, each has a range.)

This problem may seem silly, but it is not, and our choice of conflation methods depends on the case we believe to apply. We will recommend approaches for both; but first, decide which case applies. The amount of spread in your expert testimony will give you an idea whether single or multiple realities is more likely. We recommend against feedback or "drilling down" until after testimony is gathered because witnesses are notoriously vulnerable to witness leading, anchoring and all other sorts of mischief; you'll never know if you lead the witness.

Desiderata for Single and Multiple Realities. Cases dictate different characteristics for the conflation technique. Single reality requires the best estimate for the mean, the best estimate for the dispersion and the best estimate for the distribution. Multiple realities dictate the best portrayal of the multiple choices we are confronted with. We will discuss each in turn.

We will describe the apparent preferred solution for each method after asserting them. For single reality, average the parameters and correct for the understatement of extrema (using method 1b or 2a from above). For multiple realities, sample from the experts after correcting each for understatement of the extrema. If we cannot discern whether we are in single or multiple reality, then we recommend sampling because this is more conservative, meaning it will have wider dispersion. We reject the use of averaging answers on each iteration despite having used the method in a Conference Best Paper by St. Louis et al. (1998). To see why, we will show its characteristics and indicate why it is probably unsuitable.

Recommendation—Single Reality. The mean of the single reality not troublesome, almost any reasonable approach will yield the same mean. (We use the word "reasonable" with trepidation.) The standard deviation presents the problem, since individuals are known to under-report, and some methods are vulnerable to distortions. We recommend averaging parameters of the expert testimony, as shown below, because it is clear what is happening. Correct each expert's testimony for truncation of the standard deviation, or correct the average; there is no obvious difference in the order of the operations. Techniques for correcting the standard deviation were shown in the first part of the paper.

Conflation: Averaging on Each Iteration. Averaging on each iteration can have an unexpected result: Three very different sets of testimony by two experts will produce exactly the same picture. This is not obvious at first, but it is so. The standard deviation of k identical but scattered triangles, with SD = s, when iteration-averaged will produce a standard deviation $s/\sqrt{k}$. The SD of the conflation can be thus be arbitrarily small, if k is sufficiently large. This does not comport with our desire that the SD be well modeled. Correction for k can be achieved by a spreading with $\sqrt{k}$, but this is likely to be done wrong or omitted altogether, and at best, would require row-by-row corrections. Correction for expert truncation can be achieved by treating the end points as if they were 20/80 points; this can be done before or after conflation.

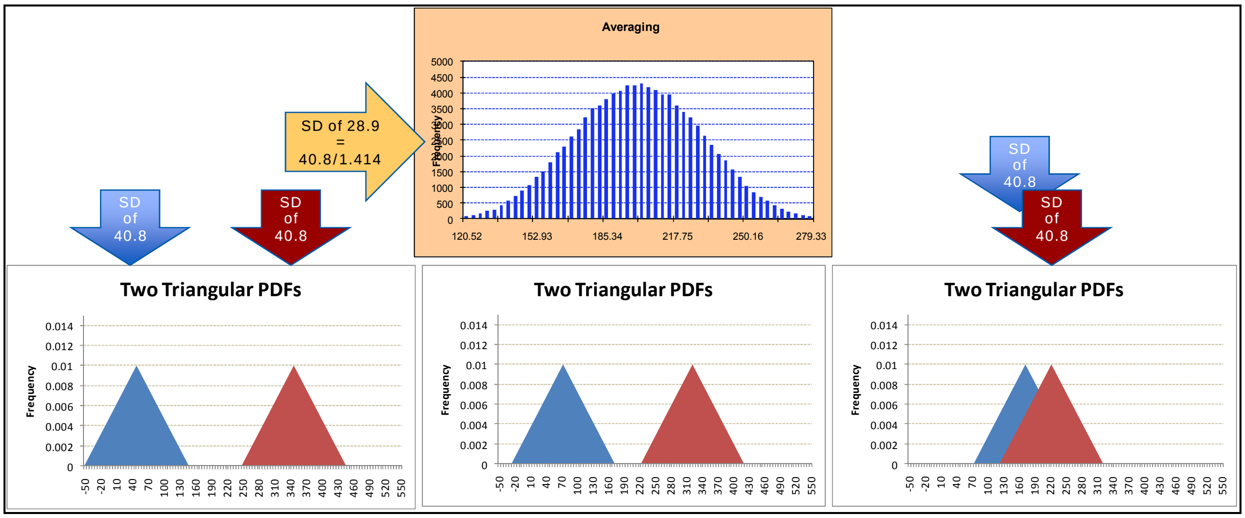


Figure 2. Conflation by Averaging on Each Iteration

We conclude that averaging on each distribution has some good and bad characteristics but, on the whole, is not desirable. It produces a good confidence interval for the mean of the experts, but this is not what we want. We already know the mean of the experts; the point estimate is the simple average of the means of each. What we really want is the full range of the possible outcomes, but averaging on each iteration does not do this; instead, it shrinks the answer. By analogy, this is the same problem as the confidence interval for a CER … it bounds the line, but not the data … what we really want is the prediction interval. It is only a candidate (and flawed at that) for clear cases of single reality.

Conflation: Averaging Parameters. Averaging parameters provides simple results: Three very different sets of testimony by two experts will produce exactly the same picture. The standard deviation of k identical but scattered triangles, with standard deviation of s, when iteration-averaged will produce a standard deviation s. The standard deviation of the conflation will not vary with k. Correction can be achieved by a spreading with $\sqrt{k}$, but this is likely to be done wrong or omitted altogether and, at best, would require row-by-row corrections.

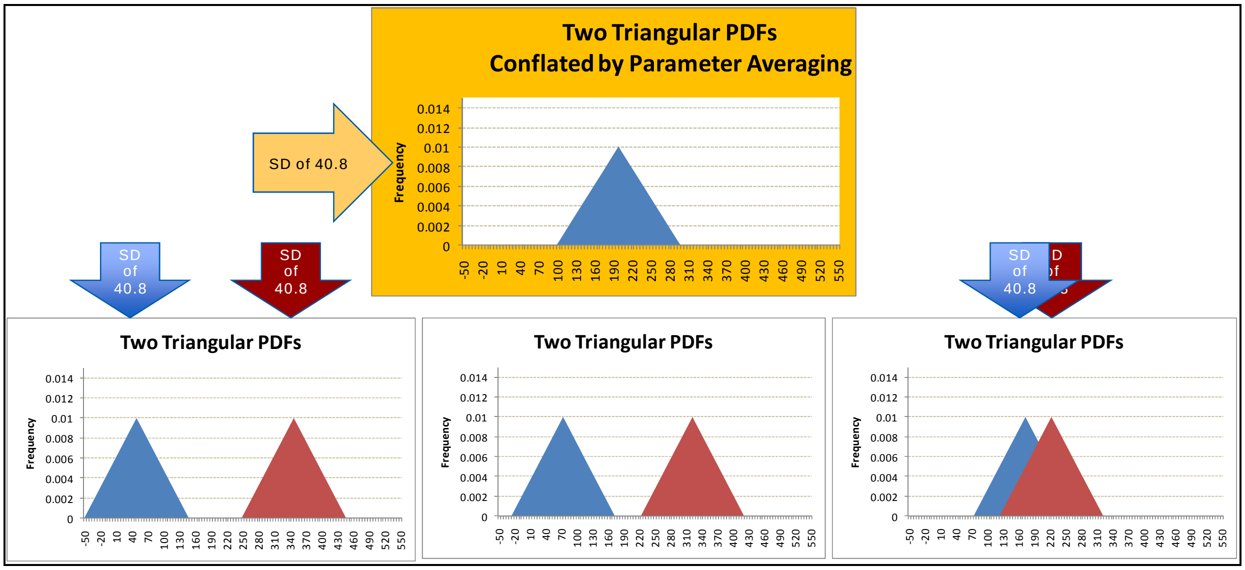


Figure 3. Conflation by Averaging Parameters

We conclude that averaging parameters has some good and bad characteristics but, on the whole, is simple and wieldy. It produces good estimates of the mean and the standard deviation. It is insensitive to scatter of expert testimony, so is only useable in clear cases of single reality. Correct the parameters as shown earlier because each expert is likely to truncate. The order of the operations does not matter.

Conflation: Sampling "Average." The probability distributions of the k experts, using one of two schemes, depending on the speed implications and the ease of implementation in your model. Put all the distributions in the mix, and scale each by 1/k, creating a (probably) multi-mode custom distribution, as recommended by Brown (1973). We will see this pictorially on the next slide. Alternatively, characterize each of the k distributions and choose a first random number to select which expert distribution to use for each run of the Monte Carlo and a second random number to draw from that expert's distribution, as used by Flynn et al. (2010). The two methods are mathematically identical. The resulting distribution will have two characteristics: 1) a better estimate of the mean and, generally, better predictive performance than other conflation schemes; 2) a wider (actually, "not narrower") standard deviation for the conflated result than those of the original individual distributions. We don't know the degree to which conflation will correct dispersion, although the more experts the wider the dispersion; we plan to attempt a study of this. We will give a demonstration of this effect with representative data.

To conflate two triangular distributions, "average" them as illustrated in Figure 4.

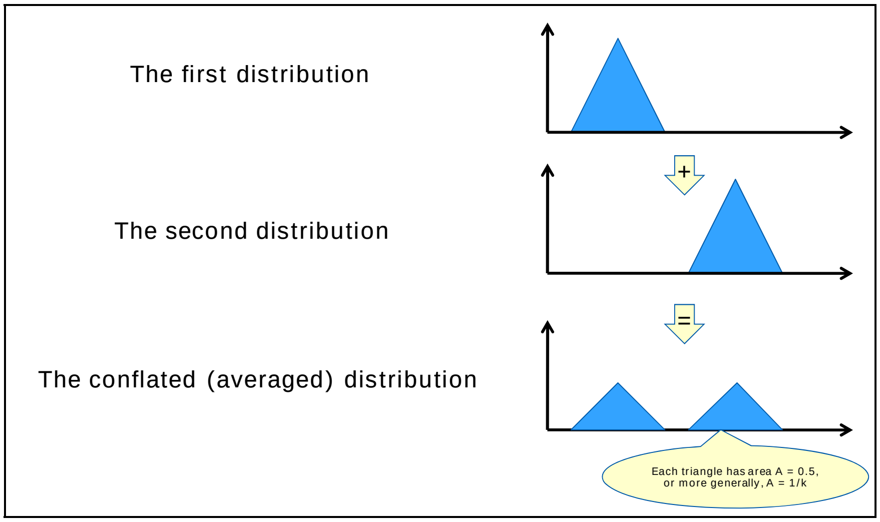


Figure 4. Conflation by Sampling

The charts in Figure 5 portray the conflation of two triangles as the respective experts who estimated them come into alignment. Each original individual triangle is symmetric, has a base length of 200, and a standard deviation of 40.8. Conflation is done by averaging the two probability density functions (PDFs), (also described as sampling). The two triangles move closer in such a way that the conflated mean remains constant at 200 to allow us to discuss the CV in a meaningful way. When the two triangles merge, we get a triangle that has the height and width of each individual triangle before conflation. The standard deviation of the conflated distribution will be shown in Figure 6.

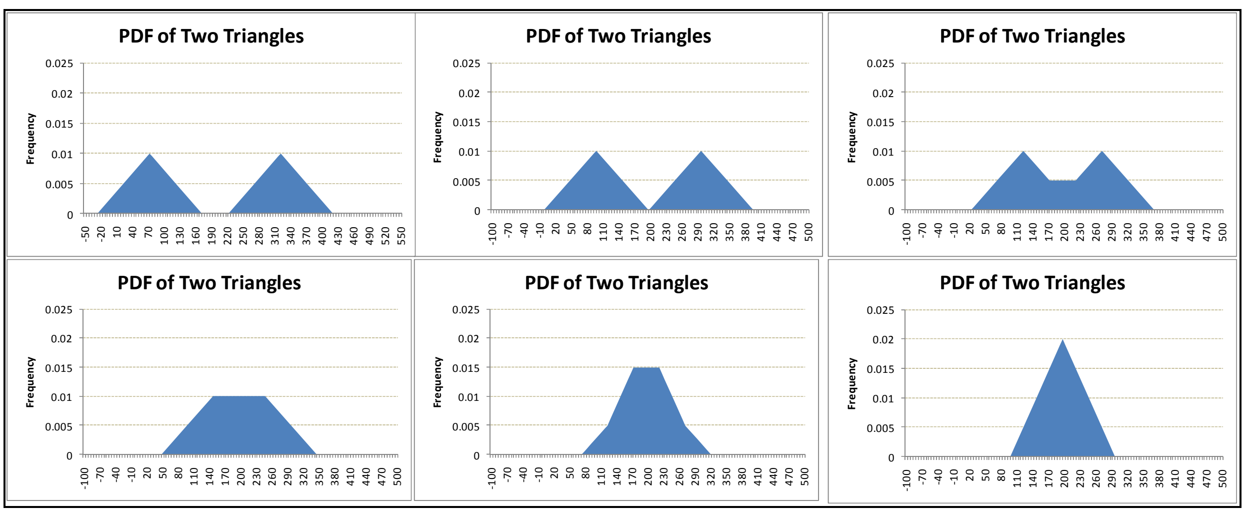


Figure 5. Conflation of Two Triangles by Sampling Maintaining a Constant Mean

As two triangular PDFs move closer, the conflated standard deviation and CV drop until the triangles merge and achieve the same standard deviation as that of each triangle. Since we chose to maintain the mean of the conflation at 200, the CV drops. The unsettling conclusion is that the CV of conflated expert opinion can be uncontrollably large, depending on how far apart their triangles. Note that the variance of two identical triangles separated by distance 2d can be shown to be $\sqrt{(\sigma^2+d^2)}$, which we prove in Appendix A.

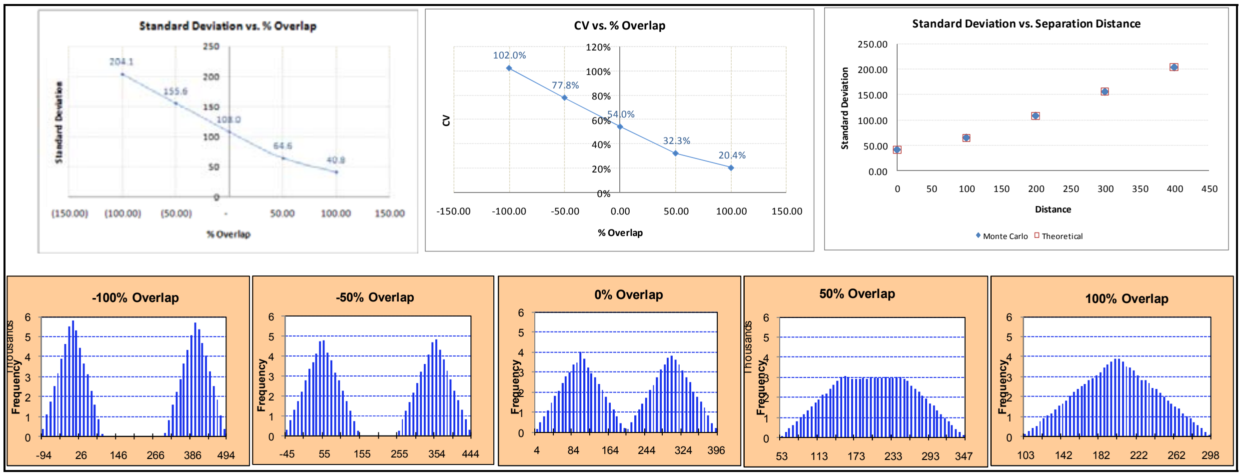


Figure 6. The Standard Deviation and Coefficient Deviation of Two Sampled Triangles as a Function of Their Separation

The Dispersion of Sampled Distributions

Let:

σ = SD of the underlying risk

S_e = SD for the individual experts (we think it is about ½σ)

S_m = SD for the meta-distribution of the experts opinions

S_c = SD of the conflation

Then, by examination,

if $S_e = 0$, then $S_c = S_m$

if $S_m = 0$, then $S_c = S_e$

And, further

$S_c \geq \max(S_e, S_m)$

This also implies that if S_e is corrected to σ, then S_c exceeds σ. We have shown, in backup, that once the experts have produced k triangles, then:

$S_c = \sqrt{(S_e^2 + S_t^2)}$

where S_t is the calculated sum of the squares of the differences of the k triangles from their means. We have yet to prove that

$S_c = \sqrt{(S_e^2 + S_m^2)}$

but we believe it to be true.

Thoughts on the Distribution of Expert Opinion. We will now speculate on the distribution of the experts themselves, which we have come to call the meta-distribution. Our assumptions are that: 1) Experts will not be versed in the distribution of costs, but will be estimating the distribution based on the outcomes they have experienced and perhaps some hearsay; and 2) Experts are most likely to be technical people, not cost estimators, so will have experience in a handful of projects and hearsay of somewhat larger number.

The implications of the above are that: 1) Experts will perceive a mean (and perhaps the mode?) according to Chebyshev's inequality or a confidence interval bounded by $\sigma/(\sqrt{n})$, at best, where n is the number they have observed; and 2) Experts will perceive a standard deviation as a variance σ times a chi-square (n) divided by n, at best.

The above thoughts do not yet consider the implications of truncation of the value of σ, but this needs to be incorporated.

Combining Corrections for Extrema and Conflation. We have shown that individual distributions can be corrected for a consistent pattern of understatement. We have shown that sampling of multiple experts will improve the mean and widen the spread. But, we don't have a good sense of how much the spread will be improved. The implication is that we should not endeavor to both expand and sample expert distributions. If we correct the individual distributions, then we will have the dispersion "about right." If we then sample them, then we will have a dispersion that exceeds "about right." So, for "single reality," do one or the other, but not both. Expansion of a single distribution focuses on the dispersion. Sampling of diverse experts focuses on getting the mean right. Since we generally recommend correcting lower order moments first, as recommended by Coleman, Summerville, and Gupta (2002), sampling is the priority. Sampling of each distribution has excellent characteristics; it replicates what the experts told us exactly. It has a problem in use for a single reality situation because the standard deviation is not easily correctible for scatter nor is it useable without correction. We can easily correct each expert's testimony for truncation, but we cannot undo the growth caused by expert scatter, which is theoretically unbounded … the adjustment would be a function of k, the number of experts, and has yet to be ascertained. We conclude that, despite its popularity in the literature, the sampling technique is too tricky in a single reality case and should not be used.

Recommendation—Multiple Realities. The mean of the multiple realities case is not troublesome; almost any *reasonable* approach will yield the same mean. (Again, that dangerous word "reasonable"!) The standard deviation does not present as much of a problem in a multiple reality case because we believe each expert, like the six blind men, sees a piece of the truth. We recommend using sampling. Be sure to correct each expert's testimony before sampling; you cannot easily correct it afterwards—order matters.

Conclusion for the Conflation of Single and Multiple Realities

As asserted, we have illustrated that the averaging of parameters for k triangles, is equivalent to averaging of draws from those k triangles with a single draw of a random number used to simulate expert's draw, and then averaging the draws. We have demonstrated why those two equivalent methods give the simplest and clearest result for single reality and seem the best representation of what the k experts seem to have meant. We have shown why sampling of k experts gives the best representation of what the k experts seem to have meant in the case of multiple realities. The issue of deciding between single and multiple realities remains the most difficult issue. Sometimes it will be as simple as learning that each expert has in mind "a different engine," and sometimes it will be a concession to the wide dispersion and the recognition that there "must be a reason." We will now move to a different topic, that of correcting mischaracterization of distributions, without which this paper would seem incomplete.

Correcting the (Mis)characterization of Distributions. The problem is that "experts" who may know a lot about the technical issues, and maybe even the cost of them, will not necessarily be well versed in probability. Consequently, the characterizations they will produce will not be easily used and will sometimes be incoherent (meaning, internally contradictory). That said, expert testimony in risk analysis should be accorded the same respect that cost data is in cost analysis. We recommend three tenets in correcting apparently erroneous expert testimony. We will list them, and we will apply them in several actual examples of errors the authors have encountered, chosen because they are the most common.

Tenet 1. "Do no harm," meaning preserve as much of what the expert said as is possible in achieving coherence.

Tenet 2. Preserve lower order moments above higher order moments.

Tenet 3. If particular aspects are more important than others, preserve those aspects (e.g., if the variability or upper percentiles are the focus, accord that greater priority).

When making corrections, it is preferable to make the corrections with direct feedback to the expert, but this feedback should be done under the same precepts as the corrections, meaning follow the tenets in your persuasions and probing.

Example One—Implausible Percentiles. The expert told us that "The 20/50/80 are \$0.0M/\$0.9M/\$3.6M." The difficulty is that no triangle can fit this, and the distribution is very skewed, so simplifying steps were taken. We assumed that the stated "50% percentile" is really the mode. We took the 20 and 80 as "about true," and assume they are $\pm\sigma$. We used the rule that the half-base lengths of a symmetric triangle are $\sqrt{6}*\sigma$. We noted that these triangles are not symmetrical, but we still used it as a factor that probably does a decent job. The results are in the table in Figure 7.

Inputs		Outputs	
20%-ile	0	L	-1.305
50%-ile	0.9	M	0.900
80%-ile	3.6	H	7.514

Figure 7. Table of Inputs and Corrections

Note that the correction *may* be distorting the central tendency, but this distribution is clearly intended to be skewed, and the mean is therefore above the median. We cannot actually compute the mean with the information given. We also knew that in this analysis, the ROS at the 80th percentile was a particular focus, so we felt that preservation of that point should take priority (Tenet 3).

Example 2—Unlikely Distributions. The expert gave us three discrete points: 20% probability of -\$2M, 40% probability of \$0, 20% probability of +\$4M. Suspecting that this was a just clumsy way to characterize a triangle, we asked if a triangle with the below characteristics was along the lines of what the expert meant: 20% percentile =-\$2M, Mode = 0M, 80th percentile = +\$4M. The expert agreed readily that the precise distribution wasn't what he meant, and the triangle captured the sense of it.

Example 3—Errors of Characterization Induced by the Risk Analyst. Below are three typical errors of characterization introduced by the risk analyst after the expert has given his testimony. They are actual examples, chosen because they are the most common.

Categorical Risk Distributions. Risk models cannot always easily (or, rather, obviously) implement a categorical random variable beyond a Bernoulli. Categorical risk distributions are like Bernoulli's but allow 2 or more values (the Bernoulli is a member of the categorical family.) Many models can handle categoricals, but most analysts don't realize that. For a 3-value categorical, with choices of 0, 1 and 2, many analysts implement it as two independent Bernoulli's with values of 0 or 1 and 0 or 2. This is an error as the results are not the same, the two Bernoulli's can turn out as 1 and 2 at the same time, but the original formulation prohibits that. To fix this problem, either implement it as a categorical or create two Bernoulli's with the right characteristics.

Triangular Risk Distributions. Sometimes the end points are set at the standard deviation of the formulation; sometimes triangles are used instead of normals, even when the normal was proposed—out of aversion to negative outcomes—even though in practice, negative outcomes are harmless in Monte Carlo; negative outcomes ought to be fairly rare anyway.

Normals. Sometimes triangles are substituted incorrectly (see above.) If the mean and standard deviation are captured correctly, then there is little harm; but this is often not done right. Sometimes the negative portion of the normal is truncated, despite that this causes a shift of the formulated mean and a reduction in the standard deviation.

Conclusion for Correcting Mischaracterizations. We have presented tenets by which apparent errors of characterization may be corrected and have listed the most common risk-analyst-induced errors. We finish by reiterating that the testimony of the experts we consult should be handled much as we should handle data. We must be careful in not ignoring the symptoms of the testimony and avoid such elementary errors as causing anchoring* and "leading the witness." We should, nonetheless, carefully repair any clear errors caused by the unfamiliarity with probability that can result in unlikely distributions.

Final Thoughts

The conflation of expert testimony has received some attention in the literature, but the conclusions seem to have permeated the cost risk discipline. We hope that we have provided a reasonably thorough paper by which risk analysts might be guided. We also hope that we have provided a few good tenets for correcting mischaracterization, along with some illustrative (actual) examples.

We hope to be able to take on the issue of what we call the meta-distribution, the likely distribution of individual expert testimony. Without a good model for the meta-distribution, the full demonstration of the best answers will remain incomplete because the meta-distribution is the unseen ground truth against which these answers can be measured. Until we can be satisfied we have the meta-distribution, we are confined to showing the behavior of various methods and deciding if that behavior seems correct.

References

Alpert & Raiffa. (1982). A progress report on the training of probability assessors. In D. Kahneman, P. Slovic & A. Tversky (Eds.), *Judgment under uncertainty: Heuristics and biases*. Cambridge University Press.

Brown, T.A. (1973). *An experiment in probabilistic forecasting* (R-944-ARPA).

Coleman, R.L., Braxton P.J., Druker, E.R., Cullis, B.L., & Kanick, C.M. (2009). *Are we at the 50^{th} percentile now and can we estimate to the 80^{th}?* SCEA Conference, St. Louis, MO; ADoDCAS Conference, Williamsburg, VA.

Coleman, R.L., Summerville, J.R., Gupta, S.S. (2002). *The manual for intelligence community CAIG independent cost risk estimates*. ADoDCAS, Williamsburg, VA;

SCEA Conference, Scottsdale, AZ; ASC/Industry Cost & Schedule Workshop, Seattle, WA.

Flynn, A.J., Nethery, B.J., Thomas, K., Gerstner, A.E., Dickey, B., Kanick, C.M., & Braxton, P.J. (2010). *Determining the cost of the certification and accreditation process using expert opinion and Monte Carlo simulation*. SCEA Conference, San Diego, CA.

Lichtenstein, Fischhoff, & Phillips. (1982). Calibration of probabilities: The state of the art to 1980. In D. Kahneman, P. Slovic & A. Tversky (Eds.), *Judgment under uncertainty: Heuristics and biases*. Cambridge University Press.

St. Louis, N., Blackburn, F.K., & Coleman R.L. (1998). *Risk analysis of a major government information production system, expert-opinion-based software cost risk analysis methodology*. SCEA/ISPA Conference, Toronto, Canada; ADoDCAS Conference, Williamsburg, VA; Journal of Parametrics.

Appendix A. Derivations and Proofs

The Geometry of Symmetric Triangles. For a symmetric Triangle(L, M, H), where M-L = H-M, find points l and h such that l and h are the p^{th} and $1-p^{th}$ percentiles (see Figure 8).

If l-L = 1/4*(H-L), H-h = 1/4*(H-L), then p = 2*$(1/4)^2$ = 1/8 = 12.5%

If l-L = 1/9*(H-L), H-h = 1/9*(H-M), then p = 2*$(1/9)^2$ = 1/18 = 5.6%

The p^{th} percentile corresponds to the √(p/2) base fraction, so the 20^{th} percentile, expressed as 1/5, occurs at point √(1/10) = 0.316228 base fraction.

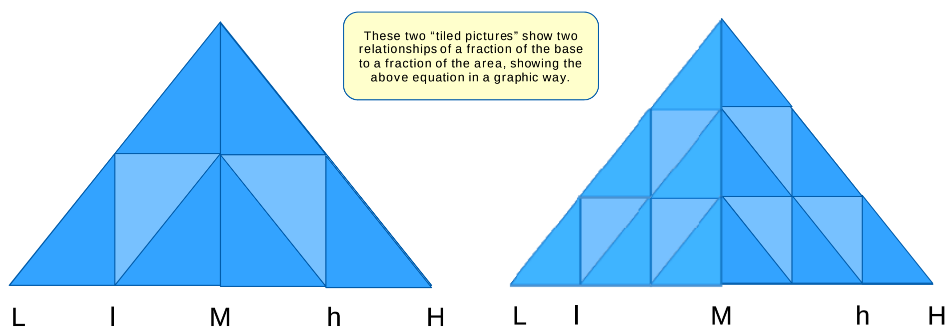


Figure 8. Visual Aid to Demonstrate the Relationship of Percentiles and Base Fraction

Triangles with Related Areas. We wish to know how to draw triangular distributions that are related to one another for our illustrations:

Triangles of Constant Area. For area to remain constant, in this case A = 1, as the base increases by a factor, the height must be multiplied by the reciprocal of that factor:

$$A = \frac{1}{2}bh = \frac{1}{2}(bk)\left(\frac{h}{k}\right)$$

Similar Triangles of Reduced Area. The dimensions of a similar triangle must be reduced by the square root of that factor:

$$A_2 = \frac{1}{k}A_1 = \frac{1}{2k}b_1h_1 = \frac{1}{2}\left(\frac{b_1}{\sqrt{k}}\right)\left(\frac{h_1}{\sqrt{k}}\right)$$

Reduction of Height to Reduce Area with Constant Base. For area to be reduced by a factor, the height must be reduced by that factor, if the base is to remain constant:

$$A_2 = \frac{1}{k}A_1 = \frac{1}{2k}b_1 h_1 = \frac{1}{2}(b_1)\left(\frac{h_1}{k}\right)$$

Triangular Distribution—PDF and Mean. For a Triangle(L,ML,H), denote L = a, H = b, ML = c denoted T(a,c,b). Since the area of the triangle must be 1 (100%), the height is twice the reciprocal of the base. We can then derive the PDF by using similar triangles.

$$p(x) = \begin{cases} \frac{2}{b-a}\frac{x-a}{c-a} & a \le x \le c \\ \frac{2}{b-a}\frac{b-x}{b-c} & c \le x \le b \end{cases}$$

$$\mu = E[X] = \int_a^b xp(x)dx = \int_a^c \frac{2x}{b-a}\frac{x-a}{c-a}dx + \int_c^b \frac{2x}{b-a}\frac{b-x}{b-c}dx$$

$$= \frac{1}{b-a}\left[\left.\frac{\frac{2}{3}x^3 - x^2 a}{c-a}\right|_a^c + \left.\frac{x^2 b - \frac{2}{3}x^3}{b-c}\right|_c^b\right] = \frac{1}{b-a}\left[\frac{2}{3}c^2 + \frac{2}{3}ac + \frac{2}{3}a^2 - ac - a^2 + b^2 + bc - \frac{2}{3}b^2 - \frac{2}{3}bc - \frac{2}{3}c^2\right]$$

$$= \frac{1}{b-a}\left[\frac{bc-ac}{3} + \frac{b^2-a^2}{3}\right] = \frac{a+b+c}{3}$$

Triangular Distribution—Variance

$$\sigma^2 = E\left[(X-\mu)^2\right] = E(X^2) - \mu^2$$

$$E(X^2) = \int_a^b x^2 p(x)dx = \int_a^c \frac{2x^2}{b-a}\frac{x-a}{c-a}dx + \int_c^b \frac{2x^2}{b-a}\frac{b-x}{b-c}dx = \frac{1}{b-a}\left[\left.\frac{\frac{1}{2}x^4 - \frac{2}{3}x^3 a}{c-a}\right|_a^c + \left.\frac{\frac{2}{3}x^3 b - \frac{1}{2}x^4}{b-c}\right|_c^b\right]$$

$$= \frac{1}{b-a}\left[\frac{1}{2}(c^3 + ac^2 + a^2c + a^3) - \frac{2}{3}(c^2 a + a^2 c + a^3) + \frac{2}{3}(b^3 + b^2 c + bc^2) - \frac{1}{2}(b^3 + b^2 c + bc^2 + c^3)\right]$$

$$= \frac{2}{3}(c^2 + bc + ac + b^2 + ab + a^2) - \frac{1}{2}(c^2 + bc + ac + b^2 + ab + a^2) = \frac{a^2 + b^2 + c^2 + ab + ac + bc}{6}$$

$$\mu^2 = \left(\frac{a+b+c}{3}\right)^2 = \frac{a^2 + b^2 + c^2 + 2ab + 2ac + 2bc}{9}$$

$$E(X^2) - \mu^2 = \frac{3a^2 + 3b^2 + 3c^2 + 3ab + 3ac + 3bc}{18} - \frac{2a^2 + 2b^2 + 2c^2 + 4ab + 4ac + 4bc}{18}$$

$$= \frac{a^2 + b^2 + c^2 - ab - ac - bc}{18} = \frac{b^2 - 2ab + a^2 + c^2 + ab - ac - bc}{18} = \frac{(b-a)^2 - (b-c)(c-a)}{18}$$

Note that the variance is thus the square of the base minus the product of the half-bases.

Substituting a Triangular for a Normal: The √6 Factor. For a symmetric Triangle(L, ML, H), let ML = m, L = m-w, H = m + w, where w is the half-base. Then the mean is m, and the variance is w2/6 (see previous proofs) and the variance is thus w/√6. It follows that the half-base is greater than the standard deviation by a factor of √6. So, to approximate a normal, the factor of √6 is multiplied by the standard deviation of the original normal to be emulated to produce the half-base of the triangle we wish to use in emulation. By this means, end points are found that will produce a triangular distribution to emulate the underlying Normal(μ, σ) in mean and standard deviation. This symmetrical triangular distribution, Triangle(μ-√6σ, μ, μ+√6σ) differs from the underlying normal in all other moments, and at all percentiles other than the median and two "cross-over" points, but the difference is minor, as shown in Figure A-2.

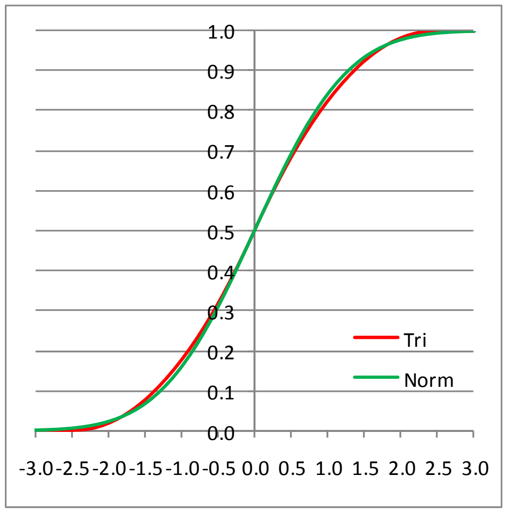


Figure 9. Comparison of Triangle(μ-√6σ, μ, μ+√6σ) and Normal(μ, σ)

Variance of Hybrid Distributions—A Pythagorean Relationship. The Mean Suppose k distributions with PDF $p_i(x_i)$, mean μ_i, and standard deviation σ_i are sampled. Then the PDF of the hybrid distribution is the "average" of the PDFs:

$$p(x) = \frac{1}{k}\sum_{i=1}^{k} p_i(x_i)$$

The mean of the hybrid distribution is the average of the means

$$\mu = E(X) = \frac{1}{k}\sum_{i=1}^{k}\int x_i p_i(x_i)dx_i = \frac{\sum_{i=1}^{k}\mu_i}{k}$$

The variance of the hybrid distribution is the average of the variances plus the variance of the means taken as a discrete probability distribution! See the next proof for the derivation of the variance.

Variance of Hybrid Distributions—A Pythagorean Relationship—The Variance.

$$E(X^2) = \frac{1}{k}\sum_{i=1}^{k}\int x_i^2 p_i(x_i)dx_i = \frac{\sum_{i=1}^{k}(\sigma_i^2+\mu_i^2)}{k}$$

$$\sigma^2 = E(X^2) - \mu^2 = \frac{\sum_{i=1}^{k}(\sigma_i^2+\mu_i^2)}{k} - \left(\frac{1}{k}\sum_{i=1}^{k}\mu_i\right)^2$$

$$= \frac{\sum_{i=1}^{k} \sigma_i^2}{k} + \left[\frac{\sum_{i=1}^{k} \mu_i^2}{k} - \left(\frac{1}{k} \sum_{i=1}^{k} \mu_i \right)^2 \right]$$

In the special case of two congruent distributions with centers at m-d and m+d, the variance is:

$$= \sigma^2 + \left[\frac{(m-d)^2 + (m+d)^2}{2} - m^2 \right] = \sigma^2 + d^2$$

Equivalence of Averaging Distributions and Averaging Parameters for Symmetric Triangles. In the case of symmetric triangles, averaging the individual triangles (with perfect rank correlation) can be shown to be equivalent to averaging the parameters. We will prove it in the case of two triangles, but the proof can easily be extended to more.

As previously shown, the p^{th} percentile (p<0.5) for a symmetric triangle is at the √(2p) half-base fraction, so the p^{th} percentiles of the two triangles and their average are:

$$a_1 + \sqrt{2p}(c_1 - a_1) \quad a_2 + \sqrt{2p}(c_2 - a_2) \Rightarrow \frac{a_1 + a_2}{2} + \sqrt{2p}\,\frac{(c_1 - a_1) + (c_2 - a_2)}{2}$$

But this is clearly just the p^{th} percentile of the average distribution. A similar proof works for p>0.5. Since all percentiles are equal, the resulting distributions are identical. Monte Carlo simulation could be used to explore the difference between the two methods for asymmetric triangles, but it is expected to be small.

Addressing Cost Increases in Evolutionary Acquisition

Doug Bodner— Doug Bodner is a senior research engineer in the Tennenbaum Institute at the Georgia Institute of Technology. His research focuses on computational analysis and decision support for design, operation and transformation of organizational systems. His work has spanned a number of industries, including automotive, electronics, energy, health care, military acquisition, paper and pulp, semiconductors and telecommunications. He is a senior member of the Institute of Electrical and Electronics Engineers (IEEE) and the Institute of Industrial Engineers (IIE), and a member of the Institute for Operations Research and Management Science (INFORMS). He is a registered professional engineer in the State of Georgia.

Douglas A. Bodner, PhD, PE
Senior Research Engineer
Tennenbaum Institute
760 Spring Street, NW
Atlanta, GA 30332-0210
E-mail: doug.bodner@gatech.edu

Farhana Rahman—Farhana Rahman is a PhD student in the School of Industrial & Systems Engineering at the Georgia Institute of Technology. She holds an MS in engineering from Louisiana State University (with a concentration in supply chain systems) and from the Bangladesh University of Engineering and Technology (with a concentration in production scheduling). She also holds a Bachelor of Industrial and Production Engineering from the Bangladesh University of Engineering and Technology.

Farhana Rahman
Research Fellow
Tennenbaum Institute
760 Spring Street, NW
Atlanta, GA 30332-0210
E-mail: r.farhana@gatech.edu

Bill Rouse—Bill Rouse is the Executive Director of the Tennenbaum Institute at the Georgia Institute of Technology. He is also a professor in the College of Computing and School of Industrial and Systems Engineering. Rouse has written hundreds of articles and book chapters and has authored many books, including most recently *People and Organizations: Explorations of Human-Centered Design* (Wiley, 2007), *Essential Challenges of Strategic Management* (Wiley, 2001), and the award-winning *Don't Jump to Solutions* (Jossey-Bass, 1998). He is editor of *Enterprise Transformation: Understanding and Enabling Fundamental Change* (Wiley, 2006), co-editor of *Organizational Simulation: From Modeling & Simulation to Games & Entertainment* (Wiley, 2005), co-editor of the best-selling *Handbook of Systems Engineering and Management* (Wiley, 1999), and editor of the eight-volume series *Human/Technology Interaction in Complex Systems* (Elsevier). Among many advisory roles, he has served as Chair of the Committee on Human Factors of the National Research Council, a member of the US Air Force Scientific Advisory Board, and a member of the DoD Senior Advisory Group on Modeling and Simulation. Rouse is a member of the National Academy of Engineering, as well as a fellow of four professional societies—Institute of Electrical and

Electronics Engineers (IEEE), the International Council on Systems Engineering (INCOSE), the Institute for Operations Research and Management Science, and the Human Factors and Ergonomics Society.

William B. Rouse, PhD
Executive Director and Professor
Tennenbaum Institute
760 Spring Street, NW
Atlanta, GA 30332-0210
Phone: 404-894-2331
E-mail: bill.rouse@ti.gatech.edu

Abstract

Acquisition programs are under pressure to deliver increasingly complex capability to the field without the cost growth associated with recent programs. Evolutionary acquisition was adopted to help reduce system cost (through the use of mature technologies) and to improve system performance (through faster deployment of incremental capability). While the ultimate verdict is not yet in on this decision, our previous simulation-based results have demonstrated that evolutionary acquisition can deliver improved capability more quickly than traditional acquisition, but that cost may actually increase over that of traditional acquisition. This is due to the overhead resulting from more frequent system deployment and update cycles. Are there other factors that can help reduce the cost of evolutionary acquisition? This paper investigates the role of system modularity and production level in the cost of evolutionary acquisition. Modularity typically imposes upfront costs in design and development, but may result in downstream savings in production and sustainment (including deployment of evolutionary new capability). A simulation experiment is conducted to determine under which conditions cost increases are minimized.

Introduction

In today's fiscally constrained environment, cost is a major issue that must be addressed in military acquisition. In particular, cost growth is of concern, where cost growth is the amount by which actual and projected costs increase over time from earlier cost estimates. A recent report from the Government Accountability Office (GAO) noted that for the fiscal year 2008 portfolio of weapons systems, there has been cost growth of $296 billion (GAO, 2009). Cost growth can result in fewer systems being produced than envisioned or desired (e.g., the F-22), or in program cancellation (e.g., the Navy Area Missile Defense). In the current and projected fiscal environment, there is considerable pressure to reduce costs and to rein in cost growth.

One driver of cost is the uncertainty and risk associated with development of new technologies for systems. In an effort to reduce this risk, and hence attempt to reduce costs and cost growth, evolutionary acquisition was adopted, whereby systems are developed using more mature technologies and deployed in increments of capability. At the same time, sustainment cost is increasingly factored into the overall acquisition cost equation, especially as system lifetimes increase. For instance, the contract for the F-35 Joint Strike Fighter calls for a combined production and sustainment supply network to promote cost efficiencies over the system lifecycle.

Not enough evidence has been collected to establish the effectiveness of evolutionary acquisition and preliminary efforts to address sustainment costs. Hence, our work uses computer simulation to assess the potential effectiveness of different strategies to address cost. Previous work indicates that evolutionary acquisition alone may not be sufficient to control costs (Pennock & Rouse, 2008). This paper investigates the effectiveness of evolutionary acquisition when system modularity and production level are considered. Modularity is hypothesized to help reduce sustainment costs associated with maintenance, repair and technology upgrades.

The remainder of the paper is organized as follows. Section 2 discusses acquisition costs, cost growth and the potential role of system modularity in helping address both. The simulation model used to study these issues is described in Section 3. Section 4 presents an experiment conducted to test the effect of modularity and production level on cost. Section 5 provides analysis of the overall experimental results. Section 6 focuses on those results relevant to evolutionary acquisition. Finally, Section 7 concludes and presents future research directions.

Acquisition Cost and Cost Growth

High cost and cost overruns have long been an issue with military systems. Cost growth occurs for a variety of reasons, including uncertainty and lack of knowledge about technology, design, and manufacturing (GAO, 2009). Candreva (2009) points to the role of institutional factors in organizational failures such as cost growth. One effort to address more cost effective acquisition was the introduction of evolutionary acquisition. Under evolutionary acquisition, the focus is on using technologies for new systems that are relatively mature, as opposed to traditional acquisition, which emphasizes use of new and immature technologies. The theory is that use of mature technologies tends to reduce cycle times for new system development, due to less risk in the technology development phase of acquisition (Johnson & Johnson, 2002). This should translate into reduced cost growth.

Our previous research has studied cost by focusing on the process aspects of acquisition. For instance, we have demonstrated that evolutionary acquisition processes can yield quicker deployment of capability than traditional acquisition processes, but at potentially higher cost due to overhead from the increased frequency of development cycles (Pennock & Rouse, 2008). This work did not address sustainment, however. Sustainment is estimated to constitute approximately 60% of lifecycle cost (Andrews, 2003). As a result, the acquisition community is putting more focus on sustainment, its associated costs and its potential for cost growth.

One avenue that may help address high cost in sustainment is the concept of system modularity. Modularity is an important concept in design of systems and products (Baldwin & Clark, 2000; Ulrich & Tung, 1991). Modular design seeks to reduce the dependencies between various system components. This has the potential to help improve the maintainability of a system over time and to reduce the cost of sustainment by facilitating repair and upgrade activities. Little research has quantitatively studied reduction in sustainment due to modularity, though. A number of hypotheses have been formulated in the research literature that are of interest in terms of the impact of modularity on cost.

1. Increasing modularity decreases the cost of implementing technology upgrades for deployed systems (Fleming & Sorenson, 2001; Garud & Kumaraswamy, 1995; Gershenson, Prasad & Zhang, 2003; Huang & Kusiak, 1998; Ulrich, 1995; Ulrich & Tung, 1991).

20. Increasing modularity decreases the mean time to repair a system that has failed and, hence, potentially the cost (Cheung & Hausman, 1995; Gershenson et al., 2003; Tsai, Wang & Lo, 2003).
21. Increasing modularity increases the upfront engineering design hours required for a system and hence potentially the cost (Ulrich, 1995).

These hypotheses suggest that there is a trade-off between cost savings in sustainment due to system modularity and the cost of designing a modular system. One goal of this paper is to explore this issue. Previous work has demonstrated that increased system modularity tends to facilitate reduced sustainment costs, in terms of repair and technology upgrade activities (Bodner, Smith & Rouse, 2009). The relationship is strongest for high levels of modularity, with diminishing returns as modularity levels are reduced. Thus, one question is what levels of modularity are required to balance the upstream costs of modularity design with the downstream savings from modularity.

Model Description

The simulation model used for this research can be characterized as three interacting sets of processes. First, there are the systems being developed for deployment. These are housed within programs that conduct the various activities required for acquisition. Second, there is the acquisition enterprise, which consists of a set of processes through which programs develop systems. We address both procurement and sustainment. Finally, there are exogenous effects that impact systems and acquisition.

We use discrete-event simulation, which has been used extensively for analysis of process-oriented domains such as acquisition (Law & Kelton, 2000). Our model is implemented using ARENA 12.0, a commercially available and widely used discrete-event simulation package (Kelton, Sadowski & Sturrock, 2004).

System Model

The acquisition enterprise develops a number of different systems for military use. In our model, a system is characterized primarily by its technologies in development and by its architecture in sustainment. In development, each system has several technologies that must be matured and integrated into the system so that the system can be successfully deployed. Each technology has an application area, a maturity level and a capability level. The application area describes the function of the technology (e.g., radar or stealth). The maturity level dictates its stage of progress from new and potentially promising to proven and mature. It is measured using the technological readiness level (TRL) scale (Kim, 2005) recently adopted by the DoD (DoD, 2006). The capability level characterizes the functional capability of the technology relative to others in the same application area.

In sustainment, the system architecture relates how different system components are arranged within the whole system. This architecture provides the basis for systems to have a specified modularity. In modeling system modularity, we assume that a system consists of *n* components, one of which is the system infrastructure. The infrastructure serves as the platform that integrates other components, and it is assumed to be a large-scale and static in nature over the life of the system (e.g., an airframe). Modularity is then conceptualized as a matrix denoting the relationships between components. A relationship exists when two components are connected with each other, or more specifically when changes to one component necessitate changes to another. In complex systems, components may be organized into modules, which then combine to form a system. Relationships can then exist between components within a module, between modules, or between components spanning two different modules. A relationship between two components affects whether a repair to a particular component requires a repair to another, and whether a technology upgrade to a particular component requires an upgrade to another.

For a system k, composed of n_k components, we define its repair modularity using an $n_k \times n_k$ matrix $\mathbf{R}_k$. Each entry r_{ijk} in this matrix represents the degree to which component i is related to component j for purposes of repair. This is expressed as the Bernoulli probability that a repair to component i results in a repair to component j. Similarly, we define $\mathbf{U}_k$ as the modularity matrix associated with technology upgrades, with u_{ijk} representing the Bernoulli probability that an upgrade to component i requires a compatibility upgrade to component j. Note that neither $\mathbf{R}_k$ nor $\mathbf{U}_k$ is assumed to be symmetrical. Figure 1 illustrates the concept of a repair modularity matrix $\mathbf{R}_k$ with $n_k = 6$. In the matrix, diagonal elements are defined to equal 1.0. In addition, component 1 is defined to be the infrastructure, and we assume that r_{1jk} =1.0 for all $j \neq 1$, and that $r_{i1k} = 0.0$ for all $i \neq 1$. In other words, all components are assumed to be affected by a change in infrastructure, while infrastructure is assumed not to be affected by a change in any component.

1.0	1.0	1.0	1.0	1.0	1.0
0.0	1.0	0.3	0.0	0.0	0.0
0.0	1.0	1.0	0.0	0.3	0.5
0.0	0.5	1.0	1.0	0.0	0.0
0.0	0.0	0.0	0.0	1.0	0.0
0.0	0.0	0.0	0.0	0.0	1.0

Figure 1. System Modularity

We use the concept of a modularity index to parameterize the extent to which a system is modular (Guo & Gershenson, 2004; Hölttä-Otto & de Weck, 2007). Since modularity is matrix-based and, hence, multidimensional in nature, an index provides a more concise characterization of modularity. Our particular index for a repair modularity matrix is defined below:

$$m_{rk} = \frac{1}{(k-1)(k-2)} \sum_{i=2}^{n_k} \sum_{j=2, j \neq i}^{n_k} r_{ijk}$$

This index is the average of the probabilities that two different non-infrastructure components are related for repair purposes. A system whose index m_{rk} is small, is considered more modular that one whose index is large. The modularity index m_{uk} for technology upgrades is defined similarly.

A system typically has a projected production level, targeted to provide the number of units needed to meet the need for which the system is being developed. This projected production level can change over time, as threats change, as new technologies/systems become available, or as costs escalate. Let P_k = the actual production level for system k.

Acquisition Enterprise Model—Procurement

The acquisition enterprise consists of the five phases of a defense acquisition program, as defined by the DoD *Defense Acquisition Guidebook* (DoD, 2006). These phases include concept development, technology development, system development, production & deployment, and operations & support. Here, the focus is on the procurement phases of acquisition, i.e., the first four phases. Operations and support is analogous to sustainment and is discussed in the next section. These phases are illustrated in Figure 2.

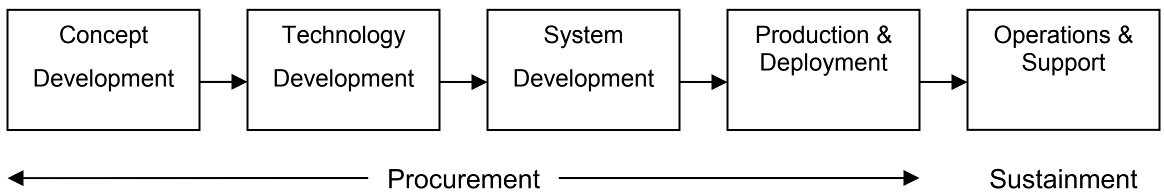


Figure 2. Phases of Acquisition

A program starts in the first phase and proceeds through the remaining phases. Sometimes, this process involves concurrency, or overlap between parts of two different phases (e.g., overlap between final testing in system development and low rate initial production). In the procurement phases (the first four phases of acquisition), each phase is characterized by duration and a cost for the program. Once a phase is concluded, the program moves onto the next phase. The model assumes no concurrency. This basic model, which does not incorporate modularity or production level, is described in greater detail by Pennock and Rouse (2008), who also discuss the parameters used for cost and duration figures for the various procurement phases. Elaborations in system development and in production are discussed below.

- Concept development. Concept development is assumed to last for a duration of between 2 and 7.5 years, with a mode of 4.9. Cost is assumed to be a linear function of duration, with rate $20 million/year.
- Technology development. The cost and duration of the technology development phase depends on whether the program is using evolutionary or traditional acquisition, as the program must mature the technologies to be used for the system if it uses traditional acquisition. Technologies for systems being acquired are developed exogenously to the acquisition enterprise, from the military science and technology (S&T) enterprise. These technologies are then brought into a program when needed in the technology development phase. The maturity level of the technologies brought into a program is used to characterize traditional acquisition versus evolutionary acquisition. The development process is risky, in that individual technologies may fail, thus causing a longer technology development phase (and higher cost) for the program. A program that uses immature technologies typically has a longer, costlier and riskier technology development phase than one that uses relatively more mature technologies. The duration and cost of this phase is the time and cost needed to develop the needed technologies for the system.
- System development. Here, we assume that the development cost and time are dependent on the level of modularity. In particular, they are increasing functions of modularity. We assume a linear relationship between cost and time. For

simplicity, repair and upgrade modularity are assumed to be the same in the model, with a modularity index denoted simply as $m_{rk} = m_{uk} = m_k$. Recall that as m_k increases, the system is characterized by a lesser degree of modularity. We then let the system development time T_{dk} be defined

$$T_{dk} = Ae^{-am_k}$$

Here, A is a scale factor related to the time in years, and a is a scale factor that represents the increasing effort it takes to make a system marginally more modular. In the model, system development time is scaled to lie between 1.5 and 8 years, and its cost is a linear function of duration at a rate of $1,000 million.

- Production & deployment. The cost and duration of the production phase are, obviously, dependent on the production level of the system in question. We use the Cobb-Douglas production function, a standard micro-economic model that relates input units to units produced to allow for increasing, constant or decreasing returns to scale (Kreps, 1990). Letting X_k = input resources used for system k, P_k = production of k, and b = scale factor, the functional form is

$$P_k = X_k^b$$

Inputs here are capital, labor and materials. Letting $b > 0$ yields increasing returns to scale, which is typical for production of complex systems. Assuming a constant cost for inputs B_k, and letting Z_k = the cost of the production phase, we obtain

$$\begin{aligned} Z_k &= B_k X_k \\ &= B_k P_k^{1/b} \end{aligned}$$

To determine the mean time needed for the production phase for a given production level, Z_k is scaled to a cost between $6,000 million and $18,800 million, and the production time is determined using a linear relationship between cost and duration with a rate of $4,000 million/year.

Acquisition Enterprise Model—Sustainment

The sustainment model includes repairs and technology upgrades for a particular system k. The production level P_k is assumed to be the fleet size for the duration of sustainment (i.e., no systems are lost or retired). Failures and technology upgrades are assumed to occur randomly according to a Poisson process. Each failure or technology upgrade affects only one component directly. Due to modularity, though, a repair to component i may require a repair to another component j (with probability r_{ijk}). Similarly, an upgrade to i may require an upgrade to another component j (with probability u_{ijk}). The rates for the failure and technology upgrade processes are defined below.

- f_i is the failure rate associated with component i.
- g_i is the repair rate associated with component i.
- t_i is the arrival rate of new technology upgrades for component i.
- v_i is the upgrade rate for component i.
- p_i is the cost of repairing component i.

- q_i is the cost associated with a technology upgrade to component *i*.
- c_{ij} is the compatibility cost associated with making component *j* technologically compatible with component *i* if *i* is upgraded, and if the interaction between *i* and *j* necessitates that *j* be made compatible to the new technology for *i*.

Since infrastructure is not affected by repairs or technology upgrades, we assume that f_1, g_1, t_1, v_1, p_1, q_1, c_{i1} and c_{1j} are not defined. In general, it is assumed that $f_i > t_i$, $g_i > v_i$, and $p_i < q_i$.

Once a system is deployed into sustainment, it experiences failures and technology upgrades for its various components according to the above rates. Technology upgrades are a significant concern for systems with long sustainment lifecycles (Singh & Sandborn, 2006). A failure invokes a repair process, and a technology upgrade opportunity invokes an upgrade process. Upon occurrence of a failure in component *i*, all other components *j* (≠ 1) with $r_{ijk} > 0$ are evaluated probabilistically, using random number generation, to determine if a repair is necessary for *j*. Any additional components are then repaired. The cost is the summation of the repair to the original component *i* and the cost of other components being repaired. Upon occurrence of a technology upgrade for component *i*, all other components *j* (≠ 1) with $u_{ijk} > 0$ are evaluated probabilistically, using random number generation, to determine if a compatibility upgrade is necessary for *j*. Any additional components are then made compatible. The cost is the summation of the summation to the original component *i* and the cost of other components being made compatible. If a failure or technology upgrade for *i* arrives while the system is in downtime, that failure or technology upgrade queues until the downtime is resolved. Multiple entities in this queue are processed first-come-first-served. This process continues for the sustainment life of the system over the active fleet, which is assume to range between 15 and 40 years, with a mode of 20 years at a median rate of $965 million per year. The actual rate is influenced by the production level.

This model provides cost for the direct activities involving maintenance and repair (including upgrades) within the sustainment phases. Clearly, sustainment encompasses many other costs. Unger (2009) presents analysis of the sustainment cost categories from the DoD's Cost Analysis Improvement Group (CAIG) for Air Force programs. Approximately 52.5% of sustainment cost can be categorized as related to maintenance and repair. We assume that approximately half of these costs (i.e., 25% of sustainment costs) are tied to direct repair and upgrade activities represented in our sustainment model.

It should be noted that there is no effect on the sustainment phase from the acquisition policy used in the procurement phases (i.e., traditional versus evolutionary).

Exogenous Model

Exogenous to the acquisition enterprise are two important outside influences. The first is a model of technical progress, which represents basic research performed in the commercial world or via non-defense government funding. Results from this model are immature technologies that are input into the second exogenous influence, the DoD science and technology (S&T) enterprise. These new technologies enter the S&T enterprise at TRL 1. Technologies generated by the model of technological progress increase in capability as time progresses using a capability growth function that combines a learning effect (from other DoD applications) and an exogenous progress effect (from commercial and outside technical progress).

The technology development process model matures new technologies for DoD systems, typically using a staged process of development whereby ideas are reduced to working technologies that can be integrated into a system. There is considerable technical risk in the development process, as ideas often do not work in practice, do not scale up to production, or do not integrate into systems. The staged process mitigates risk by not fully funding a technology's development, allowing it to be culled if it fails or if it is outpaced by competing technologies. It should be noted that the S&T enterprise model consists of a single, unified organization, rather than the myriad agencies that comprise the actual DoD S&T enterprise.

Experiment

This section describes the experiment to be conducted.

Parameters

We use the following parameter values in the simulation model for the experiment:

- $a = 1.3$
- $b = 1.25$
- $1/t_i$ ~ Triangular (5, 8, 15) years for all i and over all systems
- $1/f_i$ ~ Triangular (1.5, 2.5, 4) years for all i and over all systems
- p_i ~ uniform (0.025, 0.075) $ million for all i and over all systems
- q_i ~ uniform (0.2, 0.2375) $ million for all i and over all systems
- c_{ij} = $0.1 million for all i and j and over all systems
- k = 6 over all systems

It should be noted that t_i and f_i are parameters for Poisson processes, and that their inverses are the inter-arrival times for those respective processes, distributed as exponential variables. Thus, their inter-arrival times are shown here. Repair times and upgrade times are assumed to be instantaneous.

Experimental Design

The goal is to study the effect of system modularity and production level on acquisition cost, including sustainment, with a particular focus on cost of evolutionary acquisition. Thus, we adopt a factorial experimental design with three independent variables, as outlined in Table 1.

Table 1. Experimental Design

Independent Variable	Level 1	Level 2
Acquisition Policy	Traditional	Evolutionary
Modularity Index	0.50 (Low)	0.25 (High)
Production Level	250 (Low)	500 (High)

This results in a 2^3 factorial experiment. We are interested in studying the following six dependent variables:

- C_1 = program procurement cost
- C_2 = program sustainment cost
- C_3 = total program cost = $C_1 + C_2$
- C_4 = annualized procurement cost over all systems
- C_5 = annualized sustainment cost over all systems
- C_6 = annualized cost over all systems = $C_4 + C_5$

Each simulation replication is run for a period of 150 years, with a warm-up period of 50 years preceding. This allows analysis of the steady-state enterprise behavior, since the enterprise model needs a warm-up period to reach steady-state. After the warm-up period, the statistics collection begins. Ten replications of each combination of factors are conducted to allow for statistical significance.

Experimental Results

Table 2 provides summary experimental results. Columns two through four contain the level of each independent variable as shown in Table 1. The value shown for each dependent variable in columns five through ten is the average over the ten replications. The units of the dependent variables are in millions $.

Table 3. Summary Experimental Results

Run	Policy	Mod.	Prod.	C_1	C_2	C_3	C_4	C_5	C_6
1	1	1	1	12,657	22,248	34,906	5,305	5,075	10,380
2	1	1	2	18,640	26,409	45,049	6,378	5,653	12,031
3	1	2	1	13,832	20,623	34,455	5,378	4,441	9,819
4	1	2	2	20,078	23,096	43,174	6,189	4,296	10,485
5	2	1	1	11,518	22,248	33,766	5,960	6,316	12,276
6	2	1	2	17,548	26,409	43,957	7,071	6,556	13,627
7	2	2	1	12,629	20,623	33,252	5,884	5,210	11,094
8	2	2	2	18,910	23,200	42,109	6,981	5,290	12,271

Analysis of Overall Results

We use a balanced analysis of variance (ANOVA) method to determine which independent variables (or factors) have significant effects (Box, Hunter & Hunter, 1978). The ANOVA also computes whether there are significant interaction effects among more than one factor. In performing the analysis of variance for this experiment, Minitab® version 15 software is used. Tables 3 through 8 report the analysis of variance for each of the dependent variables, C_1 through C_6, respectively. Main effects from each independent variable are noted, as are interaction effects among independent variables.

Table 4. Analysis of Variance for Program Procurement Cost (C_1)

Source	DF	SS	MS	*F*	*p*
Policy	1	26486575	26486575	384.31	0
Mod	1	32315116	32315116	468.88	0
Prod	1	752794480	7.53E+08	10922.85	0
Policy*Mod	1	24451	24451	0.35	0.553
Policy*Prod	1	8505	8505	0.12	0.726
Mod*Prod	1	330508	330508	4.8	0.032
Policy*Mod*Prod	1	207	207	0	0.956
Error	72	4962185	68919		
Total	79	816922028			

From the analysis, we infer that each of the main effects are significant (with $p < 0.10$), as is the interaction effect between modularity and production level. Similar to Pennock and Rouse (2008), programs using evolutionary acquisition tend to have a lower program cost than those using traditional acquisition, due to higher technology development costs. As expected, low levels of modularity have lower procurement costs, due to less systems engineering work in the development phase. Also as expected, higher production levels lead to higher procurement costs. High modularity and high production level interact to increase procurement cost more than each individual factor.

Table 5. Analysis of Variance for Program Sustainment Cost (C_2)

Source	DF	SS	MS	*F*	*p*
Policy	1	13394	13394	0.01	0.933
Mod	1	119354720	119354720	63.02	0
Prod	1	223440510	223440510	117.97	0
Policy*Mod	1	13394	13394	0.01	0.933
Policy*Prod	1	13394	13394	0.01	0.933
Mod*Prod	1	13382709	13382709	7.07	0.01
Policy*Mod*Prod	1	13394	13394	0.01	0.933
Error	72	136366569	1893980		
Total	79	492598085			

For sustainment cost, there is no effect from acquisition policy. This is to be expected, since the simulation model assumes no difference in the sustainment profile between the two policies. However, the effects from modularity, production level and their interaction are significant. Low levels of modularity tend to cause higher sustainment costs, and high production levels, of course, result in higher sustainment costs. Introducing high modularity into a system tends to mitigate the sustainment cost increase associated with high production levels.

Table 6. **Analysis of Variance for Total Program Cost (C_3)**

Source	DF	SS	MS	*F*	*p*
Policy	1	25308713	25308713	12.6	0.001
Mod	1	27460953	27460953	13.67	0
Prod	1	1796490516	1796490516	894.42	0
Policy*Mod	1	1651	1651	0	0.977
Policy*Prod	1	43247	43247	0.02	0.884
Mod*Prod	1	9506987	9506987	4.73	0.033
Policy*Mod*Prod	1	10271	10271	0.01	0.943
Error	72	144615531	2008549		
Total	79	2003437868			

Looking at total program cost, each of the main effects is significant, as is the interaction between modularity and production level. The effect of the main factors is explained by the combined effect of these factors on the constituents of C_3 (i.e., C_1 and C_2). However, the effect of modularity combines opposite effects noted in C_1 and C_2. The effect of reduced cost from increased modularity noted in sustainment cost wins out, as sustainment costs overpower development costs. Similarly, the interaction effects between modularity and production level in each of the constituents are in opposite directions. Once again, the effect from sustainment costs wins out, and we can infer that high modularity helps mitigate the effect of cost increases due to high production levels.

Table 7. **Analysis of Variance for Annualized Procurement Cost (C_4)**

Source	DF	SS	MS	*F*	*p*
Policy	1	8745812	8745812	853.98	0
Mod	1	99301	99301	9.7	0.003
Prod	1	20921804	20921804	2042.9	0
Policy*Mod	1	3027	3027	0.3	0.588
Policy*Prod	1	131703	131703	12.86	0.001
Mod*Prod	1	94557	94557	9.23	0.003
Policy*Mod*Prod	1	78033	78033	7.62	0.007
Error	72	737368	10241		
Total	79	30811604			

In analyzing annualized procurement cost, we find that each of the main effects is significant, as are most of the interaction effects. Confirming Pennock and Rouse (2008), the annualized procurement cost for evolutionary acquisition is significantly higher than that of traditional acquisition. This is due to the higher number of refresh procurement cycles. Similar to C_1, high production levels are associated with higher annualized costs. However, in this case, low levels of modularity are associated with higher annualized costs, in contrast to the effect from C_1. This is likely due to the increased development time required for high levels of modularity, which reduces the number of systems deployed over the lifecycle and, hence, the annualized cost. Likewise, there is a corresponding significant interaction effect between modularity and production levels whereby low levels of modularity in conjunction with high production levels are associated with increased costs. Interestingly, there is a significant interaction effect here between acquisition policy and production level. The increased number of programs under evolutionary acquisition interacts with high production levels to increase annualized procurement costs more than each individual factor. Finally, there is a three-way interaction effect among all independent variables. This is manifested as a reduction in the difference in cost between different production levels and acquisition policies as modularity level is increased.

Table 8. Analysis of Variance for Annualized Sustainment Cost (C_5)

Source	DF	SS	MS	*F*	*p*
Policy	1	19079848	19079848	134.12	0
Mod	1	23789768	23789768	167.23	0
Prod	1	707084	707084	4.97	0.029
Policy*Mod	1	181790	181790	1.28	0.262
Policy*Prod	1	16048	16048	0.11	0.738
Mod*Prod	1	971557	971557	6.83	0.011
Policy*Mod*Prod	1	397276	397276	2.79	0.099
Error	72	10242758	142261		
Total	79	55386129			

For annualized sustainment cost, the three main effects are again significant. For modularity and production level, these effects are consistent with the reasons noted for program sustainment cost C_2. For acquisition policy, the effect is consistent with the analysis for annualized procurement cost C_4, i.e., the increased number of refresh cycles means that there is an increased number of programs needing sustainment. In this model, we assume that the program sustainment profiles are the same under evolutionary and traditional acquisition. This does not account for the possibility that, under an evolutionary system, it may be the case that system lifecycles are reduced, allowing a reduction in sustainment costs. The interaction effect between modularity and production level is consistent with the effect noted for individual program sustainment cost C_2, whereby high modularity mitigates cost increases associated with high levels of production.

Table 9. Analysis of Variance for Annualized Total Cost (C_6)

Source	DF	SS	MS	*F*	*p*
Policy	1	53661197	53661197	292.72	0
Mod	1	26963050	26963050	147.08	0
Prod	1	29321345	29321345	159.95	0
Policy*Mod	1	231729	231729	1.26	0.265
Policy*Prod	1	55804	55804	0.3	0.583
Mod*Prod	1	1672308	1672308	9.12	0.003
Policy*Mod*Prod	1	827448	827448	4.51	0.037
Error	72	13199104	183321		
Total	79	125931985			

Finally, in terms of annualized total cost, each of the three main effects is significant, as is the interaction effect between modularity and production level. The effect for acquisition policy is explained by the effect noted for the constituents of C_6 (i.e., C_4 and C_5), i.e., the increased number of programs due to evolutionary acquisition. The effect for modularity is explained by the larger effect of modularity in reducing sustainment costs than increasing procurement costs, while the effect of production level is explained simply by the increased cost of producing and sustaining more units.

Analysis of Evolutionary Acquisition

We now focus on evolutionary acquisition by analyzing the experimental results that just pertain to it. The observations below summarize our findings relative to those of the overall experiment. These results come from reducing the observations to a 2^2 factorial experiment involving only modularity and production level as independent variables.

- For C_1, the program procurement cost, both modularity and production level have similar significant effects. The interaction effect is somewhat weaker, though, registering only a *p*-value of 0.136. Thus, we infer that this interaction effect predominates for traditional acquisition due to the relatively larger program procurement cost.
- Similarly, for C_2, the annualized sustainment cost, both modularity and production level have the same type of significant effect. The interaction effect is also somewhat weaker than in the overall experiment, with a *p*-value of 0.077.
- The same observations hold for C_3. Here, for the effect of modularity and the interaction effect between modularity and production level, the sustainment cost predominates, causing increased modularity to have a reducing effect on the total program cost. The *p*-value for the interaction effect is relatively weak at 0.138.
- For the annualized costs (C_4, C_5 and C_6), modularity retains its significant effect in the same direction as in the overall experiment. However, production level is significant only for C_4 and C_6. In addition, the interaction effect between modularity and production is not significant across the three dependent variables. This is likely captured in the three-way interaction effect among the three independent variables noted in the overall experiment for C_4, C_5 and C_6. It may be that additional replications are needed to get statistically significant results for these effects. This bears further investigation.

Discussion and Future Research

The above results imply a number of conclusions relevant for military acquisition.

- Increased system modularity yields increases in the system development cost, but the decrease in sustainment cost over the system lifecycle may more than compensate for these increased costs. This points to the need to view acquisition as an investment process. While the short-term budgeting nature of the federal government works against this perspective, a longer term view does show the benefit of investing current costs to achieve long-term savings.
- Modularity can help mitigate the cost increases associated with higher production levels through an interaction effect between these two factors. Similar to the previous point, this effect involves the way in which sustainment costs overpower those of development, due to long lifecycles.
- Evolutionary acquisition seems less susceptible, especially from an annualized cost point of view, to this interaction effect between modularity and production levels. While this bears further investigation, it should be noted that those programs that do maintain characteristics of traditional acquisition may wish to investigate this phenomenon.

- Evolutionary acquisition may decrease individual program costs, but the more frequent refresh cycles may drive cost growth in overall procurement and sustainment. Thus, discretion is needed in managing these refresh cycles, especially when high production levels are involved.

This work has addressed the process aspects of the acquisition enterprise. Clearly, processes are an important and critical part of the acquisition enterprise. However, acquisition occurs in the context of organizational behavior that is impacted by incentives and information availability. The DoD has spent significant resources on incentive programs to facilitate positive acquisition outcomes. Some research suggests that these resources have not been used effectively (GAO, 2005). However, economic research suggests that it is possible to design incentive programs under different information availability scenarios (Hildebrandt, 2009). Thus, an avenue of future research is to integrate organizational behavior modeling of acquisition, combined with process modeling.

Acknowledgments

This material is based upon work supported by the Naval Postgraduate School under Award No. N00244-09-1-0015. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of the Naval Postgraduate School.

References

Andrews, R.A. (2003). An overview of acquisition logistics. White Paper. Ft. Belvoir, VA: DAU.

Baldwin, C.Y., & Clark, K.B. (2000). *Design rules: The power of modularity* (Vol. 1). Cambridge, MA: The MIT Press.

Bodner, D.A., Smith, R.J., & Rouse, W.B. (2009). Simulation-based decision support for acquisition policy and process design: The effect of system and enterprise characteristics on acquisition outcomes. In *Proceedings of the Sixth Annual Acquisition Research Symposium* (Vol. II, pp. 93-110). Monterey, CA: Naval Postgraduate School.

Box, G.E.P., Hunter, W.G., & Hunter, J.S. (1978). *Statistics for experimenters: An introduction to design, data analysis, and model building*. New York: John Wiley & Sons.

Candreva, P. (2009). Examining the institutional factors affecting cost growth in defense acquisition: Additional insights may yield more effective policy interventions. In *Proceedings of the Sixth Annual Acquisition Research Symposium* (Vol. II, pp. 258-259). Monterey, CA: Naval Postgraduate School.

Cheung, K.L., & Hausman, W.H. (1995). Multiple failures in a multi-item spares inventory model. *IIE Transactions*, *27*, 171-180.

DoD. (2006). *Defense acquisition guidebook*. Retrieved March 26, 2009, from http://asks.dau.mil/dag/

Fleming, L., & Sorenson, O. (2001). Technology as a complex adaptive system: Evidence from patent data. *Research Policy*, *30*(7), 1019-1039.

GAO. (2005). *DOD has paid billions in award and incentive fees regardless of acquisition outcomes* (GAO-06-66). Washington, DC: Author.

GAO. (2009). *Assessments of selected weapon programs* (GAO-09-326SP). Washington, DC: Author.

Garud, R., & Kumaraswamy, A. (1995). Technological and organizational designs for realizing economies of substitution. *Strategic Management Journal*, *16*, 93-109.

Gershenson, J.K., Prasad, G.J., & Zhang, Y. (2003). Product modularity: Definitions and benefits. *Journal of Engineering Design*, *14*(3), 295-313.

Guo, F., & Gershenson, J.K. (2004). A comparison of modular product design methods based on improvement and iteration. In *Proceedings of the ASME Design Engineering Technical Conference* (Vol. 3, pp. 261-269). New York: ASME.

Hildebrandt, G. (2009). Effect of information and decision-making on DoD performance incentives and award fees. In *Proceedings of the Sixth Annual Acquisition Research Symposium* (Vol. II, pp. 10-21). Monterey, CA: Naval Postgraduate School.

Hölttä-Otto, K., & de Weck, O. (2007). Degree of modularity in engineering systems and products with technical and business constraints. *Concurrent Engineering: Research and Applications*, *15*(2), 113-126.

Huang, C.-C., & Kusiak, A. (1998). Modularity in design of products and systems. *IEEE Transactions on Systems, Man and Cybernetics, Part A*, *28*(1), 66-77.

Johnson, W.M., & Johnson, C.O. (2002). The promise and perils of spiral acquisition: A practical approach to evolutionary acquisition. *Acquisition Review Quarterly*, *9*, 174-189.

Kelton, W.D., Sadowski, R.P., & Sturrock, D.T. (2004). *Simulation with ARENA* (3rd ed.). New York: McGraw-Hill.

Kim, M. (2005). Examining NASA's use of technology readiness levels. Technical report. Washington, DC: NASA.

Kreps, D. (1990). *A course in microeconomic theory*. Princeton, NJ: Princeton University Press

Law, A.M., & Kelton, W.D. (2000). *Simulation modeling and analysis* (3rd ed.). New York: McGraw-Hill.

Pennock, M.J., & Rouse, W.B. (2008). The costs and risks of maturing technologies, traditional vs. evolutionary approaches. In *Proceedings of the Fifth Annual Acquisition Research Symposium* (pp. 106-126). Monterey, CA: Naval Postgraduate School.

Singh, P., & Sandborn, P. (2006). Obsolescence driven design refresh planning for sustainment-dominated systems. *The Engineering Economist*, *51*(2), 115-139.

Tsai, Y.-T., Wang, K.-S., & Lo, S.-P. (2003). A study of modularity operation of systems based on maintenance consideration. *Journal of Engineering Design*, *14*(1), 41-56.

Ulrich, K. (1995). The role of product architecture in the manufacturing firm. *Research Policy*, *24*(3), 419-440.

Ulrich, K., & Tung, K. (1991). Fundamentals of product modularity. In *Issues in Design/Manufacture Integration* (Vol. 39, pp. 73-79). New York: ASME.

Unger, E.J. (2009). An examination of the relationship between usage and operating-and-support costs of U.S. Air Force aircraft. Santa Monica, CA: RAND.

It’s Time to Take the Chill Out of Cost Containment and Re-energize Key Acquisition Practice

Robert Tremaine—Mr. Rob Tremaine is the associate dean for outreach and mission assistance at the Defense Acquisition University West Region. He is a retired Air Force colonel and has over 26 years of experience in air, missile, and space weapons systems acquisitions. He holds a BS from the US Air Force Academy and an MS from the Air Force Institute of Technology. Mr. Tremaine is level III certified in both program management and systems planning, research, development, and engineering. (E-mail: robert.tremaine@dau.mil)

Donna Seligman—Ms. Donna Seligman is a management information systems manager at the Defense Acquisition University West Region. She has extensive experience with developing complex business knowledge applications, performing system analyses, and conducting research. She has also supported legacy code and reengineering efforts of other major business system applications. Ms. Seligman holds a BS in Information Decision Systems from San Diego State University. (E-mail: donna.seligman@dau.mil)

Acknowledgements

The authors are grateful to Defense Acquisition University West Region Dean Andy Zaleski for his tremendous insights on the topic of cost containment. The authors also thank the many current and former PMs who took the time to respond to the opinion survey—a vital source of information upon which the authors based the findings of this research. Their perspectives proved invaluable.

Abstract

Unless program managers (PM) tackle cost containment head-on, future weapon system acquisition successes may be jeopardized, resulting in fewer products and services to equip the nation’s warfighters. The United States can ill afford any decrease in its preparedness when the nation is currently waging war on two fronts. This research examines cost containment in the context of Total Life Cycle Cost Management. A more thorough understanding and aggressive application of cost-containment strategies could conceivably shift acquisition outcomes to a more cost-effective posture. Responding to a survey conducted as part of this research, 887 Department of Defense (DoD) acquisition professionals provided input on cost containment, including tool types and associated processes. Of those 887 respondents, 223 were current or former DoD PMs with over 11 years of experience—the primary basis of this research analysis.

Keywords: Life Cycle Cost Management (LCCM), Cost Containment, Cost as an Independent Variable (CAIV), Performance Based Logistics (PBL), Cost Analysis Requirements Description (CARD), Earned Value Management (EVM), Technology Readiness Level (TRL)

Introduction

Is there a superior acquisition development decision aid that can assure more program successes and help contain costs? Interestingly enough, some of the most basic tools currently at our disposal in the Department of Defense (DoD) are already ideally suited to help achieve acquisition excellence. They can also have a significant impact on fiscal outcomes. For some time, program managers (PMs) have had access to these in the form of a customized Tool Kit that outlines and characterizes a wide array of helpful decision aids and measures (DAU, 2009b), including:

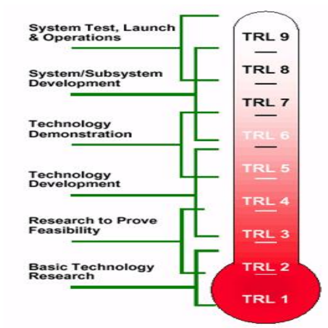


Figure 1. Technology Readiness Level

Technology Readiness Level (TRL).

Tempers technology insertion by measuring technology maturity; ensures technology properly finds its way into development efforts, while accounting for any associated risks; and considers performance and life-cycle factors before a technology solution is finalized.

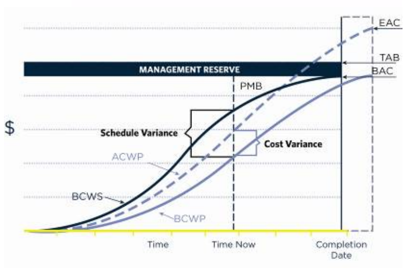


Figure 2. Earned Value Management

Earned Value Management (EVM).

Predicts cost and schedule perturbations, provides early warning, and serves as a forecasting tool that ties itself to traceable physical work packages (under an overall Work Breakdown Structure (WBS)).

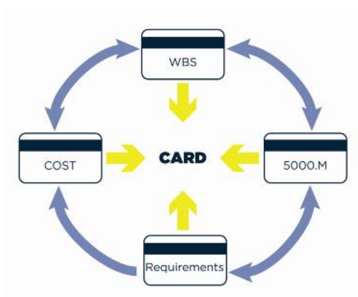


Figure 3. Cost Analysis Requirements

Cost Analysis Requirements Description (CARD). Provides comprehensive and detailed descriptions of acquisition programs; supports **Program Office Estimates (POE), Component Cost Analyses (CCA)**, and **independent Life Cycle Cost Estimates (LCCE).**

Technical Processes	Technical Management Processes
Top-Down Processes (include requirements development, logical analysis, and design solution)	Technical Planning
	Technical Assessment
Bottom-Up Realization Processes (include implementation, integration, verification, validation, and transition)	Decision Analysis
	Technical Control Processes (include requirements management, risk management, configuration management, and technical data management)

Figure 4. Technical and Management Processes

Technical and Management Processes. Ensure products properly evolve from concept to deployment; set the stage for the selection of a wide range of alternative design approaches through an integrated superset of design, assessment, and control processes.

Figure 5. Performance-based Logistics

Performance-based Logistics (PBL). "Provides a means for the resource-constrained program management office to develop, implement, and manage the sustainment of a system over its life cycle" (Fowler, 2009).

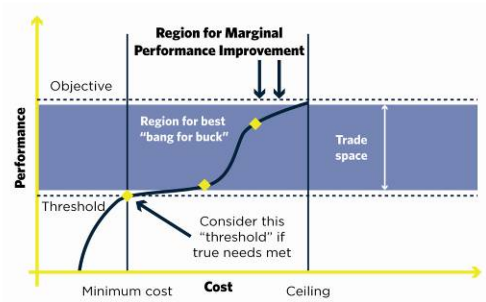


Figure 6. Cost as an Independent Variable

Cost as an Independent Variable (CAIV). Weighs affordable performance capabilities and scheduling based on cost goals that can be realized by a set of decisions that balances programmatic risks (Rush, 1997). Also serves as a trade-off tool to achieve Reduced Total Ownership Costs (Pallas & Novak, 2000).

Taken together, these tools can give PMs the power to overcome many of the looming programmatic hurdles that continue to surface as often as the weather changes. Many other helpful decision aids are available and designed specifically to combat the challenges PMs face every day. Considering this wide and diverse array of decision aids, what is missing? What have we actually failed to characterize that ostensibly fuels cost growth? Why do examples keep surfacing like the MV-22 Osprey, in which costs per flight hour—currently at $11,000—are expected to more than double the target estimate (Clark, 2009)? If so many variable costs can fluctuate, can they be properly tracked and addressed in time to contain costs?

One methodology in particular was expected to give truthful predictions of total costs. But, its value has presumably diminished in the face of the very dynamic and complex processes normally associated with acquisition programs in the DoD. It goes by the name Life Cycle Cost Management (LCCM). Up to now, it has been used to understand both the wide array of system costs that start with a program's initial baseline and run all the way through disposal.

Discussion

Conceptually, LCCM is not new. As early as 1936, T. P. Wright had already created cost-estimating equations to predict the cost of airplanes over long production runs (Hamaker, 1994). Oddly enough, many are still in use today. In varying degrees, support for LCCM continued to grow ever since. In 1975, an Air Force working group recommended five required actions to effectively institute LCCM capabilities in program offices. They recommended:

- Program offices be provided with a source of personnel familiar with analytical techniques;

- Engineers and analysts be given general guidance on how to develop, adapt, and use life-cycle cost models for specific applications;
- Program office and supporting personnel have access to a short course in the subject of development and application of LCC models and methods;
- Periodic life-cycle cost methods workshops be held; and
- Program office personnel be provided with a central focus of expertise in which lessons learned in each new life-cycle application are integrated with existing LCC models and methods (McKenzie, 1978).

LCCM is certainly not an underdeveloped concept, either. Over the years, a number of LCC models have surfaced to help programs fashion their overall funding profiles. Each model takes into account the broad range of a system's true costs, including its economic life, inflation rates, discount rates, total number of cost elements that comprise the system, magnitude of cost elements, salvage value, etc. But to this day, when asked about their experience with LCC models, their applicability, usefulness, ease of use, and limitations are viewed as questionable by many, including the DoD's most experienced program managers (see Table 1).

Table 1. Program Managers Quantify LCC Models

LCC Model Types		Model Owners	Applicability & Usefulness Across Life Cycle	Ease of Use	Data Dependencies & Model Limitations	Current users
ACARA	Availability, Cost, And Resource Allocation	NASA	?	?	?	?
CASA	Cost Analyses Strategy Assessment	LOGSA	?	?	?	?
EDCAS	Equipment Designer's Cost Analysis System	TFD Group	?	?	?	?
MAAP	Monterey Activity-base Analytical Platform	TFD Group	?	?	?	?
FLEX	Navy Material Command LCC Model	NAVAIR	?	?	?	?
LCCA	Life Cycle Cost Analyzer	Northrop Grumman	?	?	?	?
LCCH	Life Cycle Cost Model	Air Force(TASC)	?	?	?	?
Price	Family of Models for Costing/Evaluation	Lockheed Martin	?	?	?	?
ZCORE	Cost Oriented Resource Estimating Model	USAF	?	?	?	?
ACEIT	Automated Cost Estimating Integrated Tools	(USAF, USA)	?	?	?	?

Sentiments like those expressed by the National Aeronautics and Space Administration (NASA, 2008) are common among many acquisition professionals with comparable years of experience on the subject of developing/relying upon the accuracy of LCC estimates that models like these provide:

> It involves using incomplete, inaccurate, and changing data for an outmoded & ineffective space system to derive the precise cost of purchasing an unknown quantity of an undefined new space system to satisfy an overly exaggerated and unvalidated requirement at some time in the future, under uncertain conditions, with a minimum of funds. (p. 17)

Whatever model or methodology is selected, carefully (and frequently) applying it can have a lasting effect on cost containment. Of primary importance is the selection of the most suitable LCC model(s). Each characterizes a number of important variables a little differently. Nonetheless, each LCC model also has the capacity to magnify cost drivers, early and often. Regrettably, Booz Allen-Hamilton reported that the “real issue is one of obtaining the data in a timely manner and of reducing the redundant data collection effort needed every time a cost-effectiveness question arises in the decision-making arena” (Leggitt, 1981, p. 13). However, unless PMs alter their views on their usefulness and frequency of use, these models/methodologies will likely have less influence on key decisions.

Fundamentally, LCCM is actually an extraordinary concept, which is generally described through two manifestations. The first, LCC, accounts for research and development costs, investment costs, operating and support costs, and disposal costs over the system’s entire life cycle. The second, Total Ownership Cost (TOC), consists of LCC elements as well as other infrastructure or business process costs not necessarily attributable to the program (USD(AT&L), 2008). Understanding all the costs and all the implications associated with LCCM may seem intimidating. So many unknowns and so many combinations and permutations come into play that can easily vary, making it difficult to quantify any system’s total costs, especially when it matters most—during the birth of a program.

In 2006, to raise more awareness, the DoD elevated the ranking of ownership costs to a Key System Attribute (KSA) in anticipation of drawing more attention early on (Kobren, 2009). Have we given LCCM enough attention to have an impact though? Probably not. And if not, how can we garner even more attention and emphasis on this KSA? Perhaps we should just call it what it is—Aggregate Management. After all, it aggregates everything that could possibly affect the cost of materializing anything that actually gets built and eventually fielded in the DoD.

Investment budgets are shrinking, and without additional attention, initial concepts designed to meet some requirement might take a lot longer to materialize or cost a whole lot more to produce and sustain—both problematic scenarios that we as a nation can ill afford. LCCM needs to be somehow re-energized. Increasing its use would trigger the robust part of the LCCM challenge—encouraging deeper thinking, acting more critically, and pursuing more creative methods to contain overall costs. “Years earlier, Lt Gen James T. Stewart, USAF (Ret.), indicated one of the threats to cost containment and described it as ‘yo-yo funding’ (Dapore & Bryant, 1984) that persists even today in the DoD’s Planning, Programming, Budgeting, and Execution (PPBE) process.”

Exchange with Subject Matter Experts

The authors conducted two focus sessions with a handful of acquisition experts who teach the art and science of LCCM and cost estimating. Their experiences, combined with frequent contact with acquisition colleagues inside and outside the classroom, highlighted specific cost-containment issues that PMs face every day.

Their first meeting was with the Logistics Subject Matter Experts (SMEs). Each SME confirmed that LCCM issues persist. They noted LCCM considerations continue to be minimized up front, where they could have the most significant impact. They also stressed any discussion on LCCM tends to be short-lived, especially further down the acquisition continuum and after initial modeling (R. Burroughs, personal communication, September 17, 2009).

To amplify the importance of LCCM, the SMEs recommended instituting an LCC breach construct (similar to the intent behind Nunn-McCurdy breaches). For example, if a program exceeded its LCC baseline by a fixed cost percentage similar to the construct established by Nunn-McCurdy, PMs would have to report any infringement to Congress. They also indicated that it would be beneficial to establish a formulary similar to TRLs where a program could not proceed to the next phase until it demonstrated some minimum level of achievement (M. Sherman, personal communication, September 17, 2009).

Currently, the DoD expects LCC reassessments after an initial one is developed, but do these subsequent updates give enough attention to cost containment? Not explicitly.

The logistics SMEs emphasized both the lack in LCCM discipline and the absence of cross communication in programs that generally need it the most throughout a program's life cycle. They accentuated that funding allocations and key decisions typically seem to be focused on development and not sustainment. And, without a tool to respond to the dynamic nature of LCC that accounts for all costs, including Operations and Support (O&S), there will be little forewarning that a sustainment breach might be close at hand (M. Sherman, personal communication, September 22, 2009).

O&S costs constitute the majority of a program's total costs—a widely recognized tenet in DoD program management. As recently as March 2007, the Cost Analysis Improvement Group (CAIG) reaffirmed that "projected O&S costs average 60-65 percent of projected life-cycle costs after reviewing 34 Major Defense Acquisition Programs, or MDAPs" (CAIG, 2007). Just as strikingly, at the end of a program's research and development effort and just prior to production or operations, 95% of the cumulative LCC has already been committed (DoE, 1997). So, is the lack of attention actually warranted in subsequent life-cycle phases given the questionable ability to influence O&S costs? The authors suspected so, but were anxious to hear and consider divergent views from the Budget, Cost, and Financial Management experts.

The authors next met with four Budget, Cost Estimating, and Financial Management (BCEFM) SMEs. This group echoed the same sentiment voiced by the Logistics SMEs: Sustainment tends to get minimized early in the development phase. However, they added that the "ilities" are generally not well defined. They stated LCCM typically suffers from a lack of sufficient cost detail to adequately address sustainment costs that predominate once systems find their way into operations (J. Rego, personal communication, September 22, 2009).

The BCEFM SMEs quickly reached a consensus on one of the major obstacles to cost containment. They stated that funding instability makes cost containment an insurmountable prospect. Already faced with many other daily programmatic challenges, they asserted that funding instability, typically manifested by perpetual budget cuts, creates a gyrating funding baseline on top of other strategic concerns including:

- Industry partners who are not necessarily motivated by cost containment,
- Frequent changes in requirements,
- Internal staffing shortfalls that are sometimes tough to fill,
- Lack of certain key functional experience in program offices, and
- Cultural realities that emphasize program survival over program affordability.

The BCEFM SMEs also affirmed that if PMs found a cost metric that had a strong influence in controlling costs well after the "truthful predictions," then it would be widely used and could perhaps help contain costs (J. Rego, personal communication, September 22, 2009). EVM satisfies the forecasting piece of the equation, but without specific and practical motivational methods that help contain costs, its usefulness is questionable. So, do those specific methods exist today? The answer is yes. Contract incentive strategies are one of many tools available, and have been used extensively in the DoD to help curb some of the escalating technical risks and associated costs. However, they have tended to provide more short-term gains than the ones needed for longer-term, and more enduring, outcomes in the past few years, especially when technology maturity is so fluid (GAO, 2005).

LCC in Practice Today

Today, in the context of containing costs in acquisition programs in the DoD, PMs are compelled to address LCCM across their program's life cycle. As mentioned earlier, though, well before a PM's arrival, much of the projected life-cycle costs for future systems or products is rooted in decisions made during the early phases of advanced planning and conceptual design (Blanchard, 1992). Consequently, initial LCC assessments have always been a key component of a program's "go/no go" decision process since they address a program's affordability and are ultimately dependent on the military department's (or agency's) ability to secure the necessary funding. Each military department and agency gives LCCM a lot of attention at the beginning of a system's life cycle. However, in addition to LCCM concerns, military departments and agencies must balance today's operational needs with future requirements, while simultaneously ensuring that they do not neglect more capable systems still in various stages of development. These considerations are critical—all designed to either boost current system performance or meet new warfighter/user requirements.

LCC projections are not expected to be dormant once PMs take charge. Title 10 of the *United States Code* § 2434 requires the Secretary of Defense to consider an independent Life Cycle Cost Estimate (LCCE) before approving Engineering and Manufacturing Development (EMD), or Production and Deployment (P&D) of an MDAP. In practice, LCC gets looked at closely via an assortment of predictive analyses (probabilistic and deterministic) that sometimes can be difficult to absorb. So much so, that it is generally left to the experts to decipher. Very few PMs ever find themselves digging into LCC parameters. Besides, they have the experts in their respective program offices who analyze and weigh the output. Even so, many variables make it sometimes difficult for even the experts to fully quantify. The experts, who generally populate the models with key assumptions, do their best to leverage the behavior of analogous systems. Still, quantifying all the assumptions is a daunting task when so many parameters are so variable or have not been captured or qualified. Ultimately, the responsibility resides with the PM to embrace LCC estimates, but do they and their staffs revalidate these estimates on a more routine basis? Do they dive deeper into the basis of the original LCC estimate and make any necessary adjustment(s) to contain costs?

PMs recognize that LCC generally starts out with an "inferred" cost-containment element before their programs leave the initial approval process gate. What happens later is a combination of art and science mixed with some uneasiness. PMs are expected to quantify the anticipated costs of their development system across the Future Years Defense Program (FYDP). For ACAT IC and ID programs, LCC is carefully revisited by Congress in the context of Program Acquisition Unit Cost (PAUC) when costs escalate by least 15% or more of the current baseline, or 30% or more over the original baseline (DAU, 2009b, p. 31).

However, in addition to LCCM concerns, military departments and agencies must balance today's operational needs with future requirements, and not neglect more capable systems still in various stages of development—designed to either boost current system performance or meet new warfighter/user requirements.

After Milestone B (formal initiation of an acquisition program), PMs tend to narrow their focus on managing their programs day-to-day. This day-to-day strategy is about program survival. PMs dwell on cost, schedule, and performance parameters in the face of too little funding, too little schedule flexibility, and too many technology hurdles. If LCC models are seen as an initial forecasting apparatus only to give a reasonable grounding of all known costs—but not necessarily designed to contain costs—how could cost, schedule, and performance become more tightly integrated into the overall LCCM equation? And, what about CAIV? Where does it fit in? As originally envisioned, CAIV was designed to give PMs the flexibility to balance all the factors that could help contain costs—but has it? What do PMs have to say about CAIV? How are LCC and CAIV related? Are they related? What do PMs think about these questions? Their perspectives follow.

Survey Findings

The objective data generated by this opinion survey confirmed what some earlier studies found in LCCM. In addition, the data offered quite a few other interesting perspectives as well, especially in the way PMs view LCCM and CAIV regarding cost-containment principles. The survey also reinforced how PMs unevenly apply LCCM principles and cost-containment strategies across their programs.

Even though the opinions expressed in this survey were based on fundamental beliefs, opinions invariably drive decisions since they are inextricably linked to "experiences"—an imperative in the DoD's acquisition enterprise, and one of the key factors designed to help meet the certification requirements of the acquisition corps. In other words, opinions matter in the acquisition profession when such opinions are steeped in years of acquisition experience. Burrowing into the invaluable experiences that have shaped the DoD's current PM workforce can also be a very meaningful bellwether. In this particular survey, PMs provided specific narrative comments that acknowledged certain cost-containment hurdles. The survey also found a couple of misconceptions regarding the use and usefulness of some of these cost-containment tools in the Tool Kit. The discussion that follows addresses noteworthy findings.

LCC Model Familiarity and Experience

When PMs were asked to rate the LCC models that they had previously used, many were simply unfamiliar with the models. Provided below are representative comments from the opinion survey results (See Table 2). ACAT I Program Managers with over 11 Years of Experience, Review LCCM Models

Sorry, just not that familiar with the models. Somebody else uses them and provides data to me.

~

As a PM, I have not been involved with the detailed execution of the specific model used to derive cost estimates. In many instances, costs and cost estimates were derived from legacy numbers of the previous program.

~

To be honest, not my field of expertise, and I am only familiar with the tools to the extent my team uses them.

~

I have no first-hand knowledge of any of these systems/models.

Table 2. ACAT I Program Managers with over 11 Years of Experience, Review LCCM Models

LCCM Models	No Experience with Model	Thoughts based on Experience with Model		
	Not Familiar or Not Used	Not Useful	Useful	One of the Best
ACARA	87%	2%	10%	1%
CASA	78%	2%	18%	2%
EDCAS	90%	2%	7%	1%
MAAP	89%	2%	7%	2%
FLEX	91%	3%	4%	2%
LCCA	72%	3%	22%	4%
LCCH	74%	2%	21%	3%
PRICE	73%	2%	23%	3%
ZCORE	92%	2%	3%	0%
ACEIT	70%	2%	24%	4%

Usefulness of LCC Models

PMs believed that the P&D and O&S phases are better predictors of costs, while the Technology Development (TD) and EMD phases are generally the most influential in driving decisions. Contrary to what the DoD would prefer, they did not believe the pre-acquisition phases (Materiel Solution Analysis and TD) are suitable for cost containment given their inability to qualify, let alone quantify, some of the major "unknowns." More importantly, by the time their programs entered EMD, a large number of PMs declared that LCC models have significantly underestimated costs. PMs also stated these models need more precision in the early stages of program initiation since they drive so many future decisions (Table 2). Organizations like the CAIG recommended that PMs should seek more research that focused on "scrubbing development and procurement, more detailed analysis of sustainment profiles, and identification of causal factors" (CAIG, 2007).

Representative Narrative Comments. A sampling of comments on the way PMs view LCCM and its cost-containment principles follows.

Most models have many assumptions, and those assumptions are not monitored over time; and risks are not addressed to keep the assumptions valid, so the models are not valuable when decision makers really need the information.

LCC for O&S appears to be generally unrealistic.

As programs proceed along their life cycle, LCC doesn't seem to be appropriately updated.

LCCM never captures changes allowed/forced on programs, and fails to "predict" well. Models are used early on, but eventually lose influence as "inertia" takes over and programs enter "make the best of it mode."

Overly optimistic estimates.

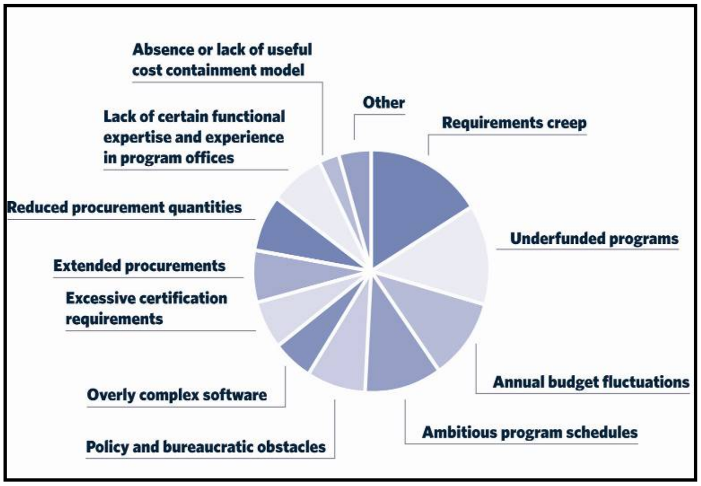
No one seems to put the thought and time into a thorough estimate of determining LCC.

No one seems to update LCC and use it as a yardstick.

Major Obstacles to Cost Containment

Of the many typical challenges that PMs face, five obstacles accounted for a noticeable majority of the reasons that made cost containment difficult to overcome, according to PMs with at least 11 years of experience. As seen in the Figure 7, those five standing in the way included requirements creep, underfunded programs, annual budget fluctuations, ambitious program schedules, and too many policy and bureaucratic obstacles.

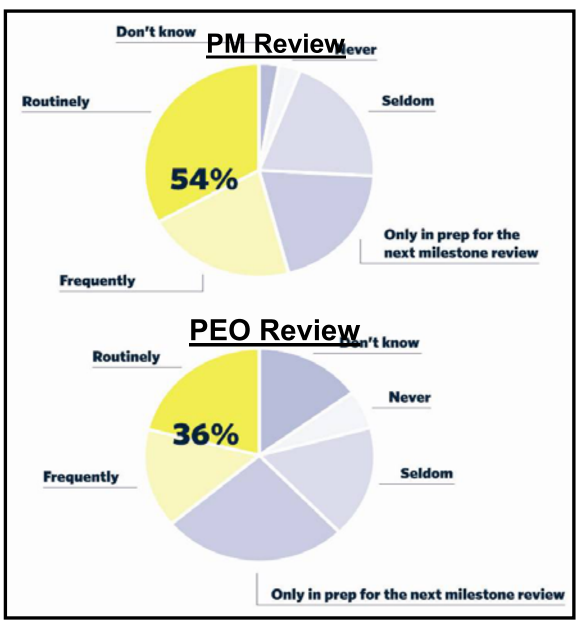
Figure 1. Program Managers Rate the Challenges They Face



Revisit Rates for LCC Estimates

Despite whether revisiting LCC estimates was viewed as a burden or resource constraint, about half of the PMs routinely or frequently reviewed their program's LCCs unless in preparation for an upcoming milestone review (Figure 8). While a great forcing function, performing LCC updates only in preparation for the next milestone is probably too late to significantly influence cost containment. However, PEOs and/or senior managers showed even less interest in LCC estimates, other than for preparation for the next milestone (Figure 8). Without more frequent and intensive reviews by either PMs or PEOs, the ability to make cost adjustments becomes more difficult to defend.

Figure 2. Program Manager and PEO LCC Review Frequency



Representative Narrative Comments. A sampling of comments on revisiting LCC highlights this seemingly low level of interest in LCC estimates other than for milestone reviews.

The costs that are of the most concern to me are those in the immediate execution year. I have considered out-year costs but not as much as I should have.

~

My focus is on providing most capability within budget, not on future life-cycle costs.

~

General knowledge on cost containment among all program office personnel is very low.

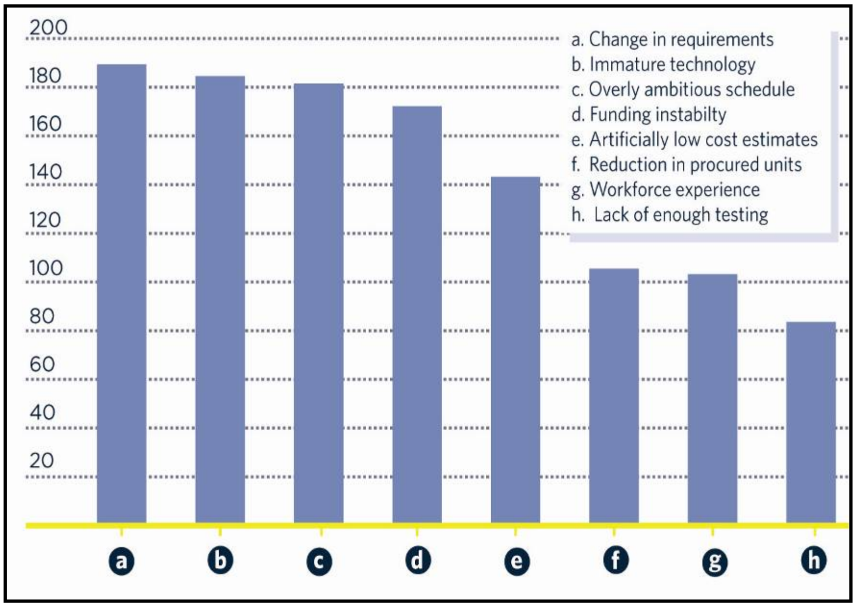
~

Many of the cost growths are based on not really understanding the requirements and instead based on assumptions on both sides.

Significant Cost Drivers

Identifying and knowing the significance of key cost drivers are paramount. Otherwise, the ability to contain costs could easily weaken. As seen in Figure 9, when asked how they would rate the significance of many of the classic cost drivers, PMs expressed that changing requirements, maturing (immature) technology, ambitious scheduling, funding instability, and artificially low cost estimates, were the most significant. With the addition of artificially low cost estimates and too many policy and bureaucratic obstacles, these were the same obstacles that made cost containment difficult to overcome when an even wider selection of survey choices was posed to PMs in an earlier question (see Figure 7).

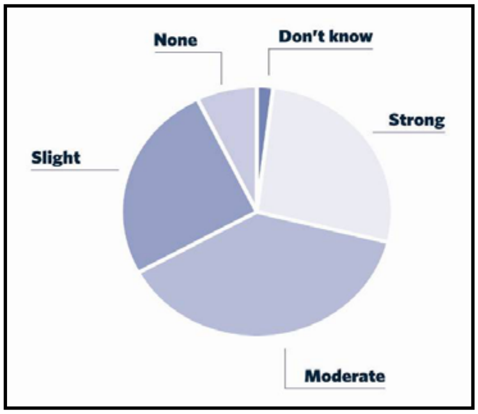
Figure 4. Program Managers Rate Their Cost Drivers



Connection Between CAIV and LCC

CAIV is another key tool available to help contain costs as previously discussed. It gives PMs a flexible instrument to help quantify the undeniable relationship(s) between certain performance requirements and realistic cost constraints. However, only 65% of the PMs acknowledged either a “strong” or “moderate” connection to LCC (see Figure 10). Subsequently, PMs might see CAIV as a quick fix only, and not fully appreciate the extent of the long-term gain; not believe there is a long-term gain; or perhaps not fully believe in the concept as a whole.

Figure 3. The CAIV and LCC Connection



Representative Narrative Comments. A sampling of comments on the relationship between CAIV and LCC shows a program management community less comfortable with CAIV as a cost control tool.

Strong in theory but weak in practice.

~

I think the relationship between LCC and CAIV has been diminished.

~

I've never seen CAIV used to contain costs on a program.

~

I don't believe CAIV has anything to do with CAIV. It's an artificial constraint that prevents the PM from meeting the requirements.

~

I didn't see CAIV used in any organized way because hardly anyone on the PM team has enough practical experience.

~

Unfortunately, the CAIV tool of last resort became common to overcome cost overruns due to funding stability and poor execution.

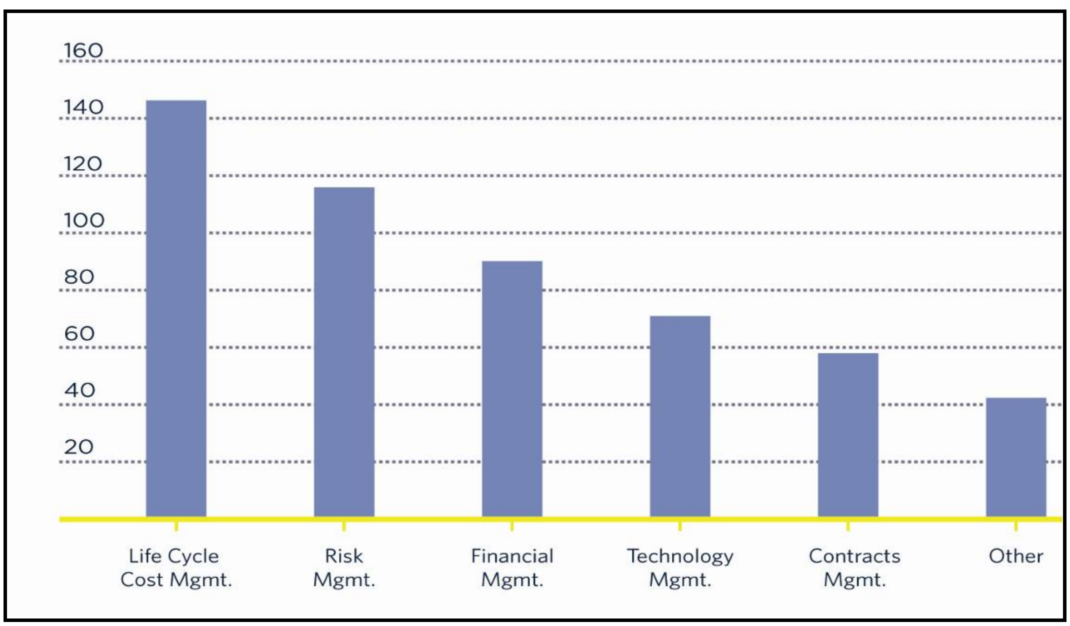
~

CAIV trades are rarely supported by the requirements community. The requirements community is 99 percent focused on capability and mildly interested in long-term O&S cost-reduction efforts.

Training Challenges

PMs stated a need for additional training, primarily LCCM and Risk Management training, to help them better contain costs. Perhaps this increased training could help strengthen cost-containment strategies.

Figure 5. Program Managers Rank Training Need



Recommendations

To reconcile some of the shortcomings of LCC and, just as importantly, better prepare PMs to contain costs and achieve more successful acquisition outcomes, the authors of this research recommend the following:

- Take the chill out of cost containment and re-energize LCCM. Make it everyone's business. Even though PMs cannot serve as LCC experts, they and their teammates should know the basis of their own LCC estimates throughout their program's life cycle, and not wait until the next milestone to make any necessary adjustment(s).
- Elevate LCC to a Key Performance Parameter—it will compel more PMs and senior personnel to rigorously exercise LCCM principles. Establishing LCC as a KSA is not enough.
- Continuously challenge strategies that are tightly coupled with their underlying assumptions.
- Base cost decisions on programmatic realities and more current data since these influence LCC outcomes.
- Establish an LCC Continuous Learning Model (CLM) that amplifies the objectives and characteristics of an LCC model and identifies the family of LCC models that best apply where, how, and when.
- Add an LCC best practice link to each functional Community of Practice (CoP) where PMs can learn from others.
- Establish LCCM trip wires throughout a program's life cycle, and do not penalize PMs for reporting unfavorable but essentially accurate program information to seniors or higher headquarters.
- Reward and incentivize PMs for containing and/or lowering costs.
- Develop cost-containment strategies that are carefully evaluated and painless to execute.
- Embrace innovation and dismiss mundane strategies that guarantee less-than-optimal outcomes.
- Promote more CAIV. Conceptually, CAIV was placed into the acquisition arsenal to give PMs a little more latitude with performance versus cost trade-offs. As ADM Mike Mullen, USN, Chairman of the Joint Chiefs of Staff, recently said at the Program Executive Officer/Systems Command Commanders' Conference at Fort Belvoir, Virginia, on November 4, 2009, "The acquisition community and the warfighter will have to jointly accept the 80 percent solution…we have to be realistic with what we can afford" (Mullen, 2009).
- Let PMs lead. PMs have the knowledge, skill, and ability to carefully guide their programs in the face of a complex and difficult environment.

Conclusions

This research reinforced the many contrasting perspectives that PMs possess with respect to cost containment and their ability to influence and/or control it. As originally conceived, understanding the usefulness and criticality of LCCM can have a major impact on weapons systems developments by keeping a lid on rising costs—a growing necessity. The acquisition environment will invariably change. Budgets will shrink; fewer new systems will be built and fielded; more pressure will be exerted on extending and sustaining current systems; and more pressure can be expected on containing costs—*much* more pressure. The remaining weapons systems under development will come under political fire. As external scrutiny swells, programmatic decisions will be challenged since there will be so much more information immediately available about emerging systems. So, how can PMs once and for all silence the skeptics and achieve positive acquisition outcomes? For starters, they can shock the critics by challenging the programmatic "cost status quo" at every juncture and not just the major milestones. They can no longer "kid themselves" about what something is going to cost, as Under Secretary of Defense for Acquisition, Technology and Logistics Ashton Carter recently stated (Carter, 2009). They can increase programmatic "cost accuracy" by better understanding and re-energizing one key cost-containment practice that has seen less action or become ineffective in recent years—LCCM. Inarguably, yo-yo funding will continue. Poor outcomes need not. The DoD cannot afford more of the same. Changes to *DoD 5000.02* that now call for Preliminary Design Reviews (PDR) prior to Milestone B and earlier measured prototyping to lower out-year costs will go a long way. Warfighters need every penny applied to capability, not cost overruns. Ultimately, PMs and their staffs must be more introspective and tightly integrate the art and the science of containing costs in the face of global economic changes. It's time to take the chill out of containing costs. The DoD depends on it; our nation depends on it; and the warfighters need to count on it.

References

Blanchard, B.S. (1992). *Logistics engineering and management*. Englewood Cliffs, NJ: Prentice Hall.

Carter, A.B. (2009, November). *Where is defense acquisition and where do you see it going?* Presentation at the PEO/SYSCOM Commanders' Conference, Fort Belvoir, VA.

Clark, C. (2009). Shut down osprey production. *DoD BUZZ*. Retrieved October 28, 2009, from http://www.dodbuzz.com/2009/06/23/shut-down-osprey-production/

Cost Analysis Improvement Group (CAIG). (2007, March 29). *Life cycle costs for MDAPs*. Presentation to the Defense Acquisition University. Washington DC: Author.

Criscimagna, N. (n.d.). Cost as an independent variable (CAIV). *START: Selected topics in assurance related technologies*, *5*(2). Retrieved November 12, 2009, from http://www.theriac.org/pdfs/startsheets/caiv.pdf

Dapore, T.L., & Bryant, L.W. (1984). *A man for the time, an oral history interview with Lieutenant General James T. Stewart*. Conducted at Aeronautical Systems Division, Bona Bills Platt, Fairborn, OH, October 18, 1983: Wright-Patterson AFB, OH: History Office, Aeronautical Systems Division, Air Force Systems Command, p. 312.

DAU. (2009a). *DAU earned value management gold card*. Retrieved November 12, 2009, from https://acc.dau.mil/CommunityBrowser.aspx?id=19577

DAU. (2009b). *Program managers tool kit* (15th ed., Ver. 1.0.). Fort Belvoir, VA: DAU Press.

DoE. (1997). *Life cycle cost estimate* (DoE G 430.1-1). Washington, DC: Author.

Fowler, R. (2009, January–February). Misunderstood superheroes: Batman and performance based logistics. *Defense AT&L Magazine*, *38*(1), 8–13.

GAO. (2005, December). *Defense acquisitions: DoD has paid billions in award and incentive fees regardless of acquisition outcomes* (GAO 06-66). Washington DC: US GPO.

Hamaker, J. (1994). But what will it cost? The history of NASA cost estimating. In F.T. Hoban, W.M. Lawbaugh & E.J. Hoffman (Eds.), *Readings in program control* (NASA SP-6103). Washington, DC: NASA, Scientific and Technical Information Office. Retrieved November 12, 2009, from http://cost.jsc.nasa.gov/hamaker.html

Kobren, B. (2009). Shaping the life cycle logistics workforce to achieve desired sustainment outcomes. *Defense Acquisition Review Journal*, *52*(3), 254-267.

Labay, K. (2009, May 1). *ISP: WHEN it will happen*. Cleveland, OH: NASA Glenn Research Center. Retrieved November 12, 2009, from http://www.grc.nasa.gov/WWW/InSpace/when.html

Leggitt, A.C. (1981, October). *Cost data base development, a twelve-year perspective*. Paper presented at the Sixteenth Annual Department of Defense Cost Analysis Symposium, Arlington, VA.

McKenzie, M. (1978, May-June). Evaluation of the developing DSN life cycle cost standard practice. *DSN Progress Report*. Retrieved November 7, 2009, from http://tmo.jpl.nasa.gov/progress_report2/42-46/46V.PDF

Mullen, M. (2009, November). *How does the 20th century warfighter differ from the 21st century warfighter?* Presentation at the PEO/SYSCOM Commanders' Conference, Fort Belvoir, VA.

NASA Advisory Council. (2008, February). *Report of audit and finance committee, Kennedy Space Center*. Retrieved November 2, 2009, from http://www.nasa.gov/pdf/314880main_AFC_KSC_NAC_Feb-5-2009.pdf

Pallas, S., & Novak, M. (2000, November-December). Reduction of total ownership costs, recent history and future prospects. *Program Manager*, *29*(159), 60-66.

PBL 12-Step Process Model. (n.d.). Acquisition Community Connection. Retrieved November 12, 2009, from https://acc.dau.mil/CommunityBrowser.aspx?id=32529&lang=en-US

Rush, B.C. (1997, Spring). Cost as an independent variable: Concepts and risks. *Acquisition Review Quarterly*, 161-172.

Tremaine, R. (2009). A new acquisition brew: Systems engineering and lean six sigma make a great mix. *Defense Acquisition Review Journal*, *16*, 100.

USD(AT&L). (2008, December 8). *Operation of the defense acquisition system* (DoD Instruction 5000.02). Washington. DC: Author.

Appendix: List of Abbreviations and Acronyms

AoA	Analysis of Alternatives
ACAT	Acquisition Category
ACWP	Actual Cost of Work Performed
ADM	Admiral
BAC	Budget at Completion
BCEFM	Business, Cost Estimating, and Financial Management
BCWP	Budget Cost for Work Performed
BCWS	Budget Cost for Work Scheduled
CAIG	Cost Analysis Improvement Group
CAIV	Cost as an Independent Variable
CARD	Cost Analysis Requirements Description
CDR	Critical Design Review
EAC	Estimate at Completion
DAU	Defense Acquisition University
CDD	Capability Development Document
CPD	Capability Production Document
DoD	Department of Defense
DoE	Department of Energy
EMD	Engineering and Manufacturing Development
EVM	Earned Value Management
FOC	Full Operational Capability
FRP	Full Rate Production
FYDP	Future Years Defense Program
GAO	Government Accountability Office
ICD	Initial Capabilities Document
IOC	Initial Operational Capability
KPP	Key Performance Parameter
KSA	Key System Attribute
LCC	Life Cycle Cost
LCCE	Life Cycle Cost Estimate
LCCM	Life Cycle Cost Management
Lt Gen	Lieutenant General
NASA	National Aeronautics & Space Administration

O&S	Operations and Support
PAUC	Program Acquisition Unit Cost
PBL	Performance Based Logistics
PDR	Preliminary Design Review
PM	Program Manager
POE	Program Office Estimate
MDAP	Major Defense Acquisition Program
P&D	Production and Deployment
PEO	Program Executive Office
PPBE	Planning, Programming, Budgeting, and Execution
Ret.	Retired
SME	Subject-matter Expert
SYSCOM	Systems Command
TAB	Total Allocated Budget
TD	Technology Development
TOC	Total Ownership Cost
TRL	Technology Readiness Level
USAF	United States Air Force
USN	United States Navy
WBS	Work Breakdown Structure

Panel #13 – Software Testing in an Open Architecture Environment

Wednesday, May 12, 2010	
3:30 p.m. – 5:00 p.m.	**Chair: Captain Brian Gannon, US Navy**, Program Manager, Naval Open Architecture, PEO IWS ***An Information-theoretic Approach to Software Test-retest Problems*** **Karl Pfeiffer**, Valery Kanevsky and Thomas Housel, Naval Postgraduate School ***The Rapid Integration and Test Environment: A Process for Achieving Software Test Acceptance*** **Commander Patrick V. Mack, USN**, and Chuck Datte, Space and Naval Warfare Systems Center Pacific ***Improved Software Testing for Open Architecture*** **Valdis Berzins** and Paul Dailey, Naval Postgraduate School

An Information-theoretic Approach to Software Test-retest Problems

Karl Pfeiffer—Presenter and author: Karl D. Pfeiffer is an Assistant Professor of Information Sciences at the Naval Postgraduate School and an active-duty Air Force officer. His current research interests include decision-making under uncertainty, particularly with regard to command and control (C2) systems; stochastic modeling of environmental impacts to weapons and communication systems; and probability modeling and numerical simulation in support of search, identification, and pattern recognition applications (e.g., complex system testing, allocation of effort for reconnaissance, etc.).

Valery A. Kanevsky—Valery A. Kanevsky is a Research Professor of Information Sciences at the Naval Postgraduate School. His research interests include probabilistic pattern recognition; inference from randomly distributed inaccurate measurements, with application to mobile communication; patterns and image recognition in biometrics; computational biology algorithms for microarray data analysis; Kolmogorov complexity, with application to value allocation for processes without saleable output; and Monte Carlo methods for branching processes and simulation of random variables with arbitrary distribution functions. Valery's most current work is focused on statistical inference about the state of a system based on distributed binary testing. Another area of interest is in the so-called needle-in-a-haystack problem: searching for multiple dependencies in activities within public communication networks as predictors of external events of significance (e.g., terrorist activities, stock market anomalies).

Thomas J. Housel—Thomas J. Housel is a Professor of Information Sciences at the Naval Postgraduate School. Prof. Housel specializes in valuing intellectual capital, knowledge management, telecommunications, information technology, value-based business process re-engineering, and knowledge-value measurement in profit and non-profit organizations. His current research focuses on the use of knowledge-value added (KVA) and real options models in identifying, valuing, maintaining, and exercising options in military decision-making. His work on measuring the value of intellectual capital has been featured in a Fortune cover story (October 3, 1994) and Investor's Business Daily, numerous books, professional periodicals, and academic journals (most recently in the *Journal of Intellectual Capital*, 2005).

Abstract

Open architecture systems developed from a standard library of reusable components should be fielded faster than systems crafted as monolithic software projects. This reliance on reusable components, however, will add to the complexity of life-cycle maintenance. The cost to repair or upgrade any one module in this library will require that we perform regression testing across all systems where this module is employed. This test-retest cycle is required to ensure that no previously satisfied requirements have been left uncovered; that is, we seek to guarantee with a suite of tests that we have not "broken" any existing functionality. Models of software debugging abound, and much of this previous research has focused on models of software fault (or bug) distributions throughout the body of the system. In this work, we examine not only this fault distribution but also the effectiveness of the test suites employed to find these faults. We evaluate the suite of regression and diagnostic tests for their potential information return versus cost. This analysis framework is flexible enough to cover many testing scenarios, and is grounded in a mathematical model suitable for rigorous analysis and Monte Carlo simulation. The goal of this work is to construct a decision-support tool for the Navy Program Executive Office Integrated Warfare Systems (PEO IWS) offering quantitative information about cost versus diagnostic certainty.

Keywords: diagnostic testing, regression testing, automated testing, Monte Carlo simulation, sequential Bayesian inference, knapsack problem

The Rapid Integration and Test Environment: A Process for Achieving Software Test Acceptance

Patrick V. Mack—Commander Patrick V. Mack, US Navy is a graduate of the Naval Postgraduate School with degrees in Computer Science and Operations Research. He is the Principal Assistant Program Manager (PAPM) for the Navy's Maritime C2 Program Office (PMW 150) responsible for the development of the Global Command and Control System (GCCS) Maritime and Navy version of GCCS-Joint programs. An Engineering Duty Officer, he has served five tours at the Space and Naval Warfare Systems Command (SPAWAR): Technical Director for the DoD's Joint Simulation System–Maritime component; Flag Aide for Commander, SPAWARSYSCOM; Deputy for the APM for Naval C2 Systems, Research and Development; and PEO Integrated Warfare Systems (PEO IWS) as the Program Director for the Cooperative Engagement Capability before his current assignment. Other assignments include OIC of SPAWAR Systems Facility Pacific, Yokosuka, Japan, and a one-year tour in Baghdad, Iraq, at the Multi-National Security Transition Command, deputy Chief of Staff for reconstruction, where he was awarded the Bronze Star.

Introduction

The *Rapid Integration and Test Environment (RITE)* initiative, implemented by the Program Executive Office, Command, Control, Communications, Computers and Intelligence, Command and Control Program Office (PMW-150), was born of necessity. Existing processes for requirements definition and management, as well as those for software development, did not consistently deliver high-quality Navy command and control (C2) systems on time and within budget. Navy C2 software programs experienced an increase in software defects that were not discovered until the completion of development activities and, because of the pressure to deploy software on schedule, product releases were distributed with defects. These defects were then repaired post delivery at significant cost. This situation was untenable and required new procedures and processes to solve the programmatic and technical challenges while operating with reduced budgets.

This paper introduces a new life cycle model for Navy C2 software that places increased emphasis on early and frequent software testing, as well as on necessary software engineering practices at the source code level. *RITE* is a more structured approach to software development, taking full advantage of technology advances and open source models to automate processes and shorten development cycles—thus increasing the maintainability of the software baselines. The initiative also clarifies software delivery requirements, adding additional engineering rigor to deliverables and reducing opportunity for misunderstanding between customers and developers. Its goal is to reduce overall cost, streamline delivery of quality C2 software, and, ultimately, resource focus toward the early stages of the life cycle, where the return on investment is maximized. *RITE* provides comprehensive oversight of software development from initial product design to customer acceptance.

RITE has four foundation pillars:

- **Software Development Contracts.** The need to provide detailed system requirement specifications and acquire favorable product licensing agreements.
- **Process improvement.** The adoption of industry software engineering best practices; testing early and often to detect, track and correct software defects while the impact on project cost and schedule is minimal.

- **Infrastructure development.** The establishment of a centralized repository with web interfaces to streamline and automate product testing, information sharing, and end-product distribution.
- **Organizational change.** The alignment of technical skills and staffing levels to support new life cycle processes.

RITE was initially developed within the context of the Maritime Global Command and Control System (GCCS) Family of Systems (FoS) (MGF) project at SPAWAR Systems Center Pacific (SSC Pac). However, it is applicable to a wider range of software development programs. This paper compares and contrasts the *RITE* Life Cycle with current Navy C2 development processes, highlighting program benefits achieved through the new initiative. Also, future implementation activities are presented, along with proposed program metrics and areas for further consideration.

Current Navy C2 Development Model

Total appreciation of the benefits associated with the *RITE* Life Cycle requires an understanding of existing development activities and how they have been adapted under the *RITE* initiative. The Navy C2 release life cycle is a subset of the overarching Department of Defense (DoD) Acquisition System described in *DoD Instruction 5000.2* (USD(AT&L), 2008). It takes place within the Engineering and Manufacturing Development (EMD) Phase and follows the Evolutionary Acquisition (EA) model used for rapid acquisition of mature technology by implementing a spiral development approach.

This section describes the current release life cycle and presents limitations inherent in existing processes that prevent effective EA performance.

Current Release Life Cycle

The current life cycle consists of the five stages shown in Figure 1. These stages run serially and are scheduled annually. The percentages associated with each life cycle stage are work-years of the level of effort, and to some extent project timelines, expended during a complete project life cycle. Projects spend a majority of the total ownership cost (TOC) after software development is completed. Because the model produces software components with "audidable" defects, there is a self-perpetuating cycle of allotting little time, or funds, for upfront requirements, design, and development, causing the majority of the budget expenditure in the later release stages on defect detection and fixes. Usually, multiple large scale development tests (DTs) are required, resulting in schedule creep and installation delays. Historically, few programs make it through operational test (OT) with a deficiency-free report.

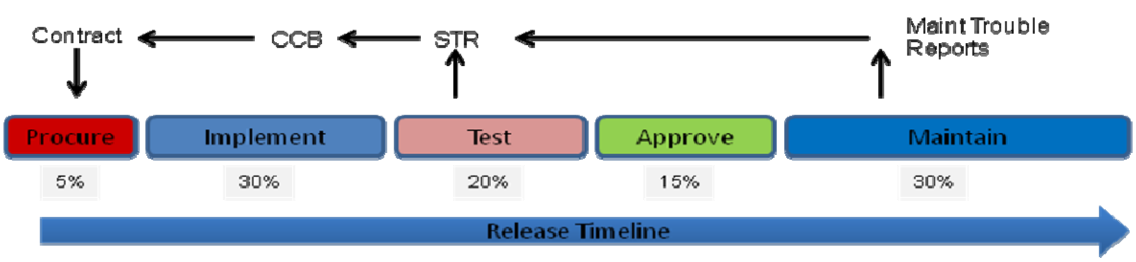


Figure 1. Current Release Life Cycle

Procurement Stage. The product release life cycle begins with the contracting officer's preparation of the contract request for proposal (RFP). The RFP is based on the list of specifications (product features) provided by the Acquisition Program Manager (APM). New specifications are based upon key performance parameters (KPP) that are derived from operational requirements and are influenced by corrective actions required as a result of the software trouble report (STRs) process. New features may be often designed to fix problems discovered either during the previous product testing or as a result of fielded system trouble reports. The final specifications are the result of a trade-off between the prioritized list of specifications and the allotted RDT&E budget, operational schedules, and established product release date. It is important to note that many of the detailed product and software documentation requirements are not clearly established in contract language under this model. The Procurement stage output is the award of an executed contract.

Implementation Stage. After contract award, the developer conducts a modified product design review and develops the software to contractual specifications. There tends to be limited interaction during this stage between the contractor development team and the Government's project team because under the terms of the contract, the contractor has proprietary ownership of the software product and sole responsibility for product delivery. Outputs from this stage are the executable software segment and any contractually required documentation.

Test Stage. The project team accepts delivery and assumes responsibility for the integration and several levels of testing. Software defects discovered during this stage are reported to the Configuration Control Board (CCB) via the STR process, where corrective action (fix) and prioritization decisions are made. By contract, the developer is required to fix critical and high-priority defects (referred to as Priority 1 and 2, respectively) prior to undergoing final testing. Lower priority category defects may be repaired, if time and budget permits. Once the software product successfully passes a final testing, a recommendation is made to support a fielding decision. Exit criteria include a demonstration that the software has matured to an acceptable level of Fleet readiness and the software meets systems integration and interface standards.

Approval Stage. The Approval stage involves many approval steps, including security certification and accreditation (C&A), successful operational test (OT), and the formal release approval. These activities are primarily conducted by outside certification agencies, such as the Commander Operational Test and Evaluation Force (COMOPTEVFOR). The output from this stage is the final fielding decision and the granting of final release approval by the Milestone Decision Authority (MDA).

Maintenance Stage. The final stage includes installation, training and continued maintenance of the C2 system.

Life Cycle Limitations

There are many program limitations inherent in the current life cycle model. In previous efforts to streamline and shorten the development cycle, the Government allowed the software developer to assume too much responsibility for the project's success. There were insufficient checks and balances built into the model to ensure that the Government received a quality product on schedule and within budget. Major limitations are highlighted below and were the drivers for the *RITE* initiative.

***Limited Requirements Definition and Detailed Design*.** The previous life cycle model allowed the selected software developer to assume responsibility for detailed system design. Detailed system requirements should be developed by the Government and specified to the developer as part of the contracting process. Additionally, requirements need to be based upon end user (warfighter) inputs and prioritized to meet the most pressing operational needs. Contracts lacked the detailed design specificity needed to fully define the end product. Developers need to have specifications to build to, and test and evaluation (T&E) personnel need those specifications to test against. A frequently cited study, conducted by the Standish Group in 2000, reported that American companies spent $84 billion for cancelled software projects. Another $192 billion was spent on software projects that significantly exceeded their time and budget estimates. The Standish Group, and other studies, list three top reasons why software projects fail:

- Requirements and specifications are incomplete
- Requirements and specifications change too often
- There is a lack of user input (to requirements)

The cost of cancelled and failed projects is likely to have increased since the initial study but indications are that the reasons for failure have not changed. Under the current release life cycle model, making Navy software programs suffer from all three of these shortfalls.

Insufficient Noncommercial Computer Software Rights. Previous Navy C2 contracts failed to acquire the appropriate rights to the software products, thereby allowing the developer to control the product and, essentially, all future enhancements, citing proprietary ownership. As defined within the *Defense Federal Acquisition Regulation Supplement (DFARS)*, the Government can receive “Unlimited Rights” for noncommercial computer software, including source code, whenever the product is developed solely with Government funding. When Government and industry team in the research and development (R&D) effort, using both Government funds and industry funds, the Government is able to retain “Government Purpose Rights” which still affords the authority to receive, assess, and modify the source code of noncommercial computer software products. The lack of Government control of the software source code prevents the Government from ensuring the quality of the software products. Without the source code, product reviews are conducted at too high of a level to determine the condition of the deliverable. It is too easy for defects to go undetected and even for defects to find their way back into new builds after having been previously repaired. Without the rights to the source code, the true quality of the released product is often not known until receipt of user trouble reports.

Limited Schedule Control. Under the existing life cycle model, the SPM only has direct responsibility for the Test stage activities. All other stages are under external ownership and, from a project viewpoint, are essentially fixed in duration and effort. Even during the Test stage, the SPM has limited control because the level and schedule of the integration and testing conducted is primarily dependent upon the quality of the executable software and associated work products delivered by the developer. The Government needs to have greater involvement in the Implementation stage, working in partnership with the developer, to ensure the quality of the delivered product. Doing so allows the Government to monitor and influence the development schedule and to have better control of the subsequent Test stage.

Insufficient Government Technical Oversight. As stated above, current contracts do not provide rights to the noncommercial computer software source code at the level necessary for effective Government technical oversight. However, just obtaining “Unlimited Rights” will not, in itself, solve the technical oversight shortfall. The historical lack of direct Government responsibility for software development has taken its toll on the quantity and quality of resident software development skill sets. Professionally trained software developers were faced with the career decision to either attain new skill sets or transfer to private industry and write software code. Therefore, for the Navy to provide effective technical oversight and serve as "trusted agents" requires a retooling of its work force. New software engineers will need to be recruited or current staff will need to be trained in new software development techniques and tools.

Reduced Competitive Environment. Lastly, many software development contracts have essentially become sole source contracts. Because incumbent developers have proprietary ownership of the software source code, new contractors attempting to compete for follow-on development contracts basically have to "rewrite" the code because the incumbent is not generally required to relinquish control of their work products. Further, much of the current contract language may not require detailed documentation of the software, making it difficult for anyone other than the original developer to understand or modify the delivered code. These barriers-to-entry greatly reduce the competitive landscape and afford the incumbent a significant competitive advantage over its competition. Also, with little or no true competition, the Government experiences a reduction in pricing power and control over the final contract.

The *RITE* Initiative

The implementation of the *RITE* Life Cycle by PMW 150 represents a dramatic shift in the way the Navy C2 Program Office develops noncommercial computer software. *RITE* provides a software oriented set of engineering standards, processes and guidance, tools, and contract language, all available through a software development, test and distribution infrastructure. It impacts all stages of the life cycle and facilitates Government control of the various stages, reducing project costs and improving schedule performance.

The *RITE* Life Cycle Model

As previously stated, a major goal of *RITE* is to reduce overall cost and streamline delivery of quality Navy C2 software. It promotes an open standards-based culture of modularity and reuse to keep pace with evolving technology. The national security implications of open technology development are clear: increased technological agility for warfighters, more robust and competitive options for program managers, and higher levels of accountability in the defense industrial base. Technologically advanced Navy C2 systems are vital to the warfighter's ability to plan and execute missions. The *RITE* initiative entails a parallel shift in acquisition methodologies and business processes to accelerate the delivery of advanced C2 systems to the operational forces.

This section describes *RITE's* open architecture approach and how information will be accessed, used, reused, applied, distributed, and managed under the new initiative. *RITE* involves changes in organizational structure, processes, strategies, policies, and business practices, including the shifts in traditional Government and contractor software development roles. It provides the necessary guidance to organize, manage, and employ a distributed, interoperable, and scalable net-centric, collaborative development and distribution environment.

RITE Pillars

The *RITE* initiative is designed around four functional pillars.

RITE Contract. A baseline requirement for *RITE's* implementation is the adoption of specific contract language that changes the existing relationship between the prime software developer and the Government project team. New contracts address the following contract stipulations.

- **Requirement Definition.** The Government assumes responsibility for developing the system requirements and baseline design specifications used by the software developer and the Government project team for contract performance. These requirements are based upon operational requirements and are at a level of specificity that provides developers and testers product acceptance criteria. Requirements definition involves the interaction of all stakeholders early in the Procurement stage. This Government engineering task is a vital part of preparing the contract language to insure the Government gets the desired product from the developer.
- **Licensing Agreement.** The Government obtains either Government Purpose Rights or Unlimited Rights, as defined in *DFARS* and applicable agency supplements, for all noncommercial computer software items developed with Government funding. This includes the delivery of software source code and related software version design documentation.
- **Process Adherence.** The Government mandates that use of the *RITE* Life Cycle be processed through the Statement of Work (SOW). New SOW language includes:
    - Contract Data Requirements Lists (CDRLs) and Data Item Descriptions (DIDs) that define an expanded set of delivered software work products, including source code and software version documentation;
    - Streamlined test processes, requiring the use of automated and focused testing procedures;
    - Contractor Performance Acceptance Reporting System (CPARS) metrics that satisfy *RITE* entrance and exit acceptance criteria;
    - Specified Quality Management (QM) procedures;
    - Specified Configuration Management (CM) to the source code level;
    - Implementation of disaster recovery techniques; and
    - Software auto-installation capability.

RITE Process. The *RITE* Life Cycle is shown in Figure 2. A major change is the coupling of the Implementation and Test stages and the direct involvement of the SSA project team in software development. There are a number of test events (engineering drops) of the software as it is being developed. Similarly, developers integrate *RITE* processes, techniques and tools into their development process. Both stages now take place seamlessly as part of the *RITE* process, aligning early defect detection, tracking and resolution with development activities. The *RITE* Life Cycle includes the implementation of front-end engineering, source code quality management, a distributed development environment, and automated development and test tools. A key assumption of *RITE* is that software development projects will always contain bugs and defects regardless of the skill and diligence of the development team. *RITE* has been designed to mitigate the overall program impact of software problems by the use of early and frequent software assessments.

Also shown in Figure 2 are the adjusted levels of effort (LOE) associated with the each life cycle stage. *RITE* places an increased emphasis on the early stages in an effort to detect and correct errors in the product design and code while the cost to correct is relatively inexpensive. Therefore, the Procurement stage has been expanded to allow for additional upfront Government effort needed for the development of requirement specifications and detailed designs. Additionally, the Implementation and Test stages have merged, signifying closer continuity between the two stages. Frequent testing of incremental software builds, referred to as "engineering drops," during the Implementation stage has been accommodated by an increase in development schedule time. The increase in LOE during the early stages is offset by a reduction in the time needed to complete the formal Test, Approval and Maintenance stages, respectively. Under *RITE*, the software release exits the Implementation stage with fewer defects, thereby reducing the uncertainty, and the project duration, associated with the latter stages. *RITE* improves the overall life cycle process to the extent that TOC is expected to be reduced while the frequency and quality of the product releases increase.

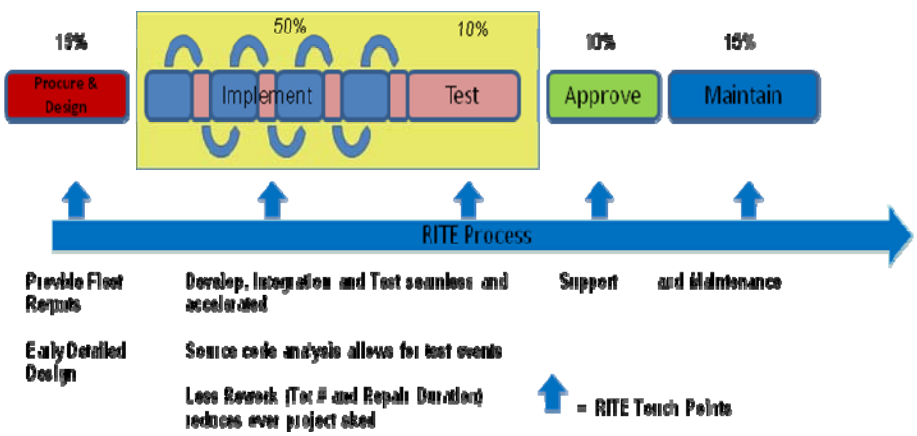


Figure 2. *RITE* Life Cycle Model

Automated and Focused Testing. Testing is critical to the overall development process success and is the means to validate and drive software quality improvement. *RITE*'s testing philosophy is based upon the need for early and frequent testing of software source code. Software development and defect detection activities begin almost simultaneously during the Implementation stage. Therefore, *RITE* mandates additional incremental testing throughout the Implementation stage, not waiting until the end-of-program to test the product. Quality needs to be built-in (and validated with incremental integration and testing), not tested for at the end.

The RITE concept of testing is based on testing to a level of acceptable risk. Since it is not feasible to test 100% of the software code, available resources, such as funding and personnel, time or urgency, and expertise of the test team, influence the extent of acceptable risk. To minimize the level of risk, the RITE uses automated inspection and test tools wherever possible, thereby increasing test coverage and allowing faster discovery of defects remaining after requirements and design inspection and assessment. The automated tools range from simple scripts to complex commercial tools and exercise the software and identify outstanding defects. Testers use the tools to measure software complexity and assign quality ratings to segments entering the integration process. They also use automated test tools to perform time-consuming repetitive procedures, such as executing a test case multiple times under a variety of conditions, over an extended period of time, or both. Automated tools also are useful to simulate large numbers of users for performance or load testing, or exercise software that does not have a graphical user interface, such as device drivers or software libraries.

Where manual testing is necessary, the *RITE* test team follows a rigorous test methodology that focuses on predetermined test cases derived from real world situations. Testers prepare and execute detailed test procedures that provide clear and concise test steps and expected outcomes.

Testers document test results derived from automated and manual testing in standardized test reports that incorporate quality metrics indicating the level of software maturity (lack of defects). Sponsors use these reports to reduce risk and determine when the software is ready for release.

- Detection and Acceptance Process. *RITE* implements a systematic integration and test process to maximize efficiency in defect detection, thereby accelerating the release of high-quality software products to the Fleet. This systematic approach to testing allows more coverage per unit of test time, leverages automated testing to help identify bugs early, and uses Navy use-cases for test scenarios. Figure 3 illustrates the *RITE* Gated Acceptance and Detection process performed on each delivered incremental software build. This process is a part of the overall *RITE* Process, as shown in Figure 2, and is conducted repeatedly throughout the various life cycle stages to identify product defects and to validate product development milestones. Inputs to the process are the engineering drops that include the executable segments and associated software work products such as Software Version Description (SVD), test plans, test procedures, test reports, and load and installation instructions. Process outputs include a qualified system, system test reports, installation instructions, and training plans. The process gates are described below. Note that these gates are serial. Regardless of gate, whenever a drop fails to pass an inspection or test, team members notify configuration management (CM), who, in turn, notify the development contractor. If resolution of the root cause for the failure requires a software modification, the process starts over.

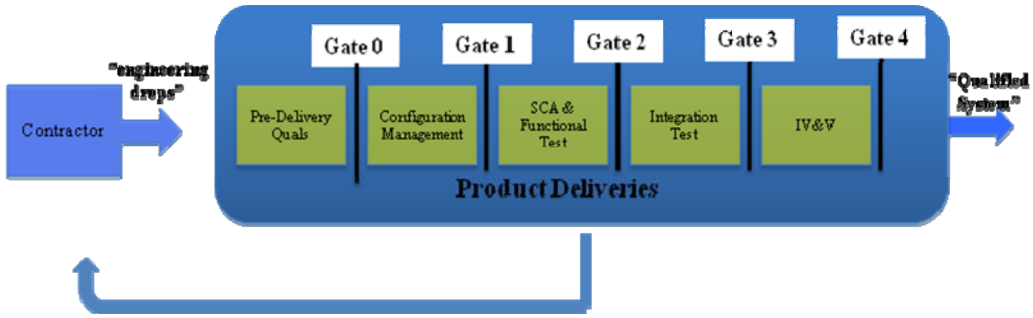


Figure 3. *RITE* Gated Detection and Acceptance Process

- *Gate 0–Pre-Delivery Qualifications*. Prior to the delivery of a new software component from the contractor, candidates are required to meet specified criteria that include contractor conducted pre-delivery testing and the development of test reports.

- *Gate 1–Configuration Management*. The contractor delivers an engineering drop to the *RITE* CM team. The team reviews the delivery to ensure all contractually required work products are present. Upon validation, the CM team delivers the software to the *RITE* Acceptance Test team.

- *Gate 2–Source Code Analysis (SCA) and Functional Testing*. The *RITE* Acceptance Test team schedules an Acceptance Readiness Review during which they check the delivery against the acceptance checklist to ensure the delivery media is readable and that the media and documentation are correct and complete. This process includes checking that all required licenses are present and current, and that test plans, procedures, and installation instructions are included. Following the review, the Acceptance Test team performs an installation test to ensure the segment installs correctly. Automated tools are used to perform an analysis at the source code level. Exit criteria for this phase are a readable and correct segment, correct documentation, a successful installation test, and a quick look test report. The Acceptance Test team notifies CM to deliver the software to the Integration team.

- *Gate 3–Integration with Baseline System*. After CM delivers the software, the Integration team reviews the segment installation procedures and attempts to integrate the segment with other C2 system segments into a complete system or "build." If they can successfully build the complete system, they perform high-level checks to ensure the build starts up correctly and major functionality is present. Exit criterion for this phase is a successful Independent Verification and Validation (IV&V) Test Readiness Review indicating that the system is ready for more in-depth testing.

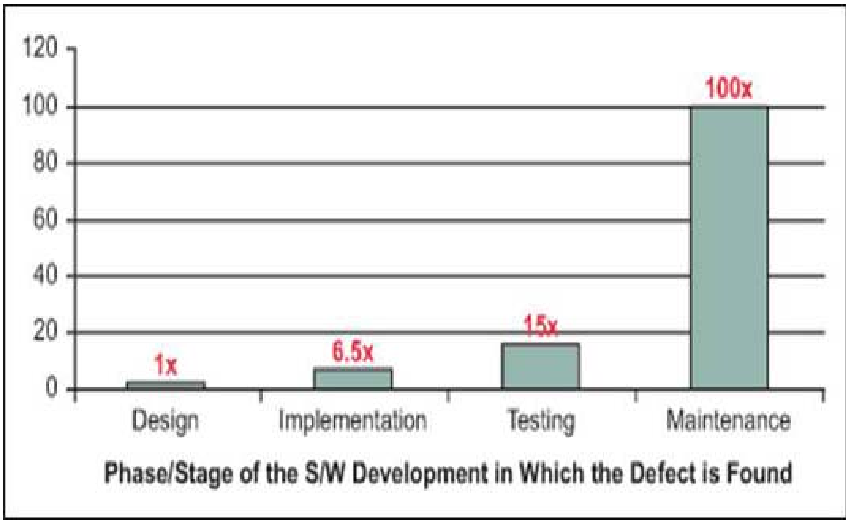
- *Gate 4–Independent Verification and Validation (IV&V) Test*. The IV&V team develops test plans and procedures covering all types of functional testing. They perform functional testing to verify that the build meets specified requirements and validates that it achieves the desired functionality, and perform interface testing to ensure that the build meets external interface requirements. If both these tests are successful, the IV&V team performs system-level and stress testing in an environment that closely simulates the operational environment. Exit criteria for this phase are a successful IV&V Test Review and delivery of the IV&V Test Report to PMW-150 or other sponsors.

Once PMW-150 accepts the IV&V Test Report, the *RITE* team supports a Developmental Test (DT) by performing laboratory DT testing at the operational site. The *RITE* team provides on-site training to shipboard operators and resolves issues that occur during DT testing. Exit criterion for this phase is a qualified system that is ready to enter the Approval stage.

- **Early Defect Detection.** Since the foundation of the testing process is based upon the early identification of software defects, research supporting this principle is provided in this section. Software defects have different names in different agencies but, for the purposes of this paper, a software defect is any development error, issue, bug, defect or incident.

In an Internet article, Mukesh Soni (2010) states that the "Prevention is better than the cure" adage applies to software defects as well as medicine. Potential software defects detected during the early stages of software development, such as during requirements specification, are easier and cheaper to resolve than during later stages presented in the IBM Systems Science Institute study chart shown in Figure 4. Defects introduced during the requirements and design phase are not only more probable but also more severe and more difficult to remove during later stages of development, test and maintenance. This is because of the increasing number of interfaces and dependencies that exist in the code as well as the time it takes for developers to refresh their knowledge of the specific code being repaired the further removed they are from the original development. Pre-test reviews and inspections are the most efficient way to detect errors in requirements and design.

A result of early detection is the reduction in the number of defects released with delivered systems. This will reduce the need for expensive software maintenance programs and free up future budget dollars for increased RDT&E expenditures. It is a *RITE* goal to demonstrate the ROI associated with the new life cycle processes by validating the improved system performance of fielded systems. A premise is that over time, budget allocations can be adjusted to allocate more money to the early stages (Procure and Implement) where the ROI is the greatest.



Source: IBM Systems Sciences Institute

Figure 4. Relative Cost Required to Fix Errors During Software Development

RITE Infrastructure. A Distributed Development Environment (DDE) is a virtual collaborative environment that spans multiple organizations and/or multiple physical locations. In a DDE, project members share ideas, information and resources, and actively collaborate to achieve a common goal. They may not see each other face to face, but are all working collaboratively toward the project outcome. This may be accomplished through e-mail, the Internet and other forms of long-distance communications. The primary advantage of DDE is availability of resources and access to software development tools from different locations. The objective is to lower development costs, increase productivity, decrease time-to-release, and improve product quality.

Software development in the Navy is transitioning to geographically distributed development environments. Distributed development is one of the highest forms of collaboration in the development environment, but many challenges face project managers responsible for the success of distributed teams. Four characteristics common to many of today's collaborative failures include:

- Cultural incompatibility,
- Leadership struggles,
- Lack of trust, and
- Inbred notions of competition.

RITE DDE strives to overcome collaboration challenges. Success requires understanding relationships and taking practical and affordable actions to achieving successful virtual operations. These include building an organization that supports working in a distributed development, with the right incentive systems that reward collaboration. It requires urbane management and oversight, a highly efficient infrastructure, a well-developed organization, and daily interaction with open communication.

The hub of the *RITE* DDE infrastructure is the Development and Distribution (D2) Center. The D2 Center allows access to, and sharing of, applicable Navy C2 program software, test tools, program governance and guidance documentation and other project technical documentation generated as part of the *RITE* Life Cycle process. Developers, testers, and other stakeholders have access to the Center through a private cloud using a web-based interface and a set of intuitive tools for locating and extracting desired components and associated work products. The D2 Center provides strong configuration control of the various project artifacts and assures that contractor and Government teams are working from a common set of project components.

Project members, using the RITE D2 Center, have the ability to specify and validate requirements using interactive simulations and a collaborative process that involves all stakeholders. Project members take ownership for achieving the overall project goal. No one can succeed without everyone being successful. Accurately identifying software requirements and effectively managing those requirements throughout the life cycle are keys to reducing rework activity and creating applications that accurately reflect end users' needs.

The infrastructure architecture is shown in Figure 5 and takes advantage of an open architecture to support the following project functions:

- Government management of key project artifacts,
- Management of source code,
- Definition and management of the development and integration environment,
- Configuration Management (CM) for validation and control of software deliveries,
- Support tool development,
- Architecture, and
- Guidance and governance documentation.

RITE D2 products are available on the site for sharing by all stakeholders and improve project communication and coordination while providing a common set of standards and tools for use throughout the project. Examples of how the D2 Center might facilitate software development and distribution include:

- Development. Using the D2 Center, a developer can log into the site and, by reviewing the available service catalog, discover that a new multi-tactical data links capability (MTC) component exists. Coding changes to the software component can then be made and automated acceptance tests can be re-run, all done remotely.
- Distribution. Similarly, Fleet users can access the site to download new Navy C2 components and automatically upgrade and test their systems without requiring the assistance of outside installation teams.

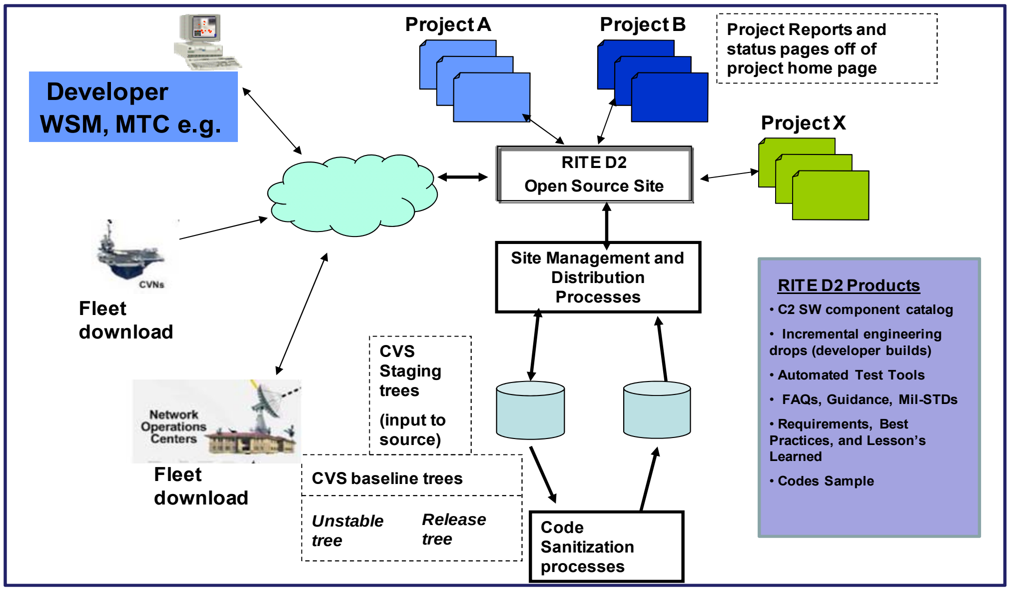


Figure 5. *RITE* Infrastructure Architecture

RITE Organization. Lastly, one of the key components of the *RITE* initiative is the organizational change needed to efficiently and effectively perform within the new life cycle model. Current project structure evolved to support existing processes and personnel skill sets were optimized for the needed job task capabilities. As the model changes, increasing the need for more software engineers and reducing the number of fleet installation teams, the organizational core competencies need to change. The projected changes include:

- Project Manager Performance Measures. New performance metrics are needed for the Government management team. In a distributed work environment where success is dependent upon frequent communication and collaboration, success factors need to reduce the current competitive environment. Additionally, success needs to be measured by program efficiencies and effectiveness that result in budget optimization, not the overall program budget size. Therefore, additional metrics associated with product quality improvement and the ability to meet Fleet operational requirements need to be captured and used for overall performance assessment.
- Personnel Qualifications. As highlighted previously, the Government lacks the number of qualified personnel, either educated or trained in the software engineering disciplines, to perform the job task functions required by RITE. These technical qualifications include knowledge of current operating systems, databases, and functional applications. Of importance are skills associated with open architecture development and web design. The Government needs to begin the transition, selecting a cadre of technically qualified software engineers to lead the workforce shift from current methods and processes while initiating focused recruitment and training programs.
- Organizational Structure. In addition to the personnel qualifications, the project organizational structure needs to evolve to meet the changing life cycle model. Under RITE, the staffing levels associated with software development and testing will need to grow to meet the increased level of effort and product throughput associated with those stages. Conversely, although not immediate, there will need to be a reduction in staffing associated with installation and integration

activities performed during the Maintenance stage. As the ability to remotely control the distribution of new software releases through the D2 Center becomes widely accepted, the need for installation teams is reduced. Therefore, the organization needs to be modified to reflect these changes in staff levels to free up budget dollars for use elsewhere.

Case Study—Multi-Tactical Data Links Capability

Casualty Description

The casualty to the multi-tactical data links capability (MTC) on the USS John C. Stennis (CVN 74) was first reported in May 2008 via the casualty report (CASREP) system. The stated problem was a "channel crash" that prevented the use of XXXX and degraded the Stennis ability to perform in mission areas

Contractor Approach

As a result of the CASREP, the development contractor was tasked with the responsibility of troubleshooting and repairing the problem. Their initial response was to form a technical "fly-away" team and travel to the forward deployed ship location to conduct an onboard investigation. In August 2008, the team boarded the Stennis, while on deployment, and began troubleshooting the reported problems, working alongside shipboard technicians. While onboard, the team did attempt to repair the MTC by reloading the current software release version (v4.5.9.14) but this did not resolve the problem. Upon return to San Diego, the contractor team continued troubleshooting at their facility as part of the future product release version (v4.5.9.15)) but achieved little or no success. By November 2009, unable to isolate and repair the MTC problems, an STR was written to more thoroughly document the problems and provide a current status update. Subsequently, in January 2009, seven months after the problems were initially reported, the contractor again sent a team of system experts to the ship to further investigate the issues. Their actions included the installation of the new MTC software release. Initial indications were that the new release corrected the channel crash but further investigation determined that the problem persisted. In April 2009, the program office called for a review with the contractor to determine a course of action. The decision was made to use the *RITE* process to aid in a timely repair. This effort was implemented in phases to monitor the effectiveness of the approach. Throughout this initial phase, the SSC Pac SSA team had limited government oversight involvement in the software maintenance activities by only tracking activities via the STR process and providing laboratory support, as needed. The initial Phase milestones are outlined in Figure 6.

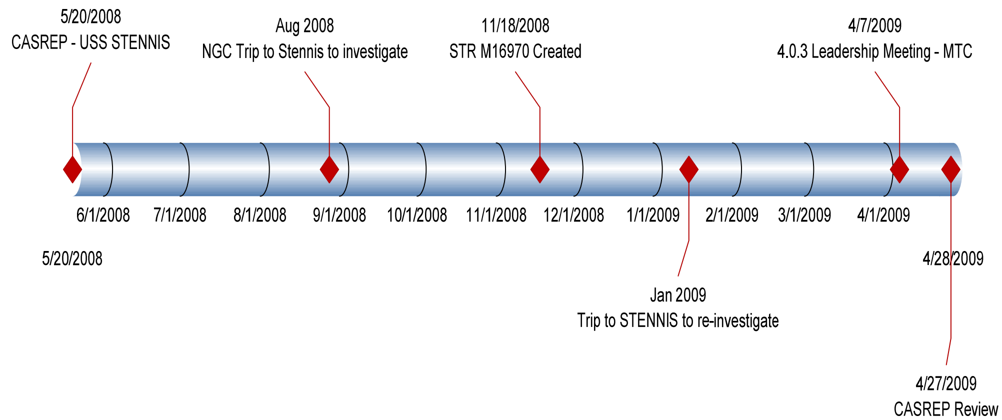


Figure 6. Phase 1 Milestones

In Phase 2, the next level the *RITE* process implementation was instituted as a software repair had not be identified. Phase 2 instituted the Engineering drop process consisting of engineering reviews and MTC assessments conducted by the SSA team for cause and effect. The engineering drop process by itself did not yield a repair. A modification to the existing process instituted a lower level STR investigation by conducting source code analysis using a set of automated source code analysis tools and peer reviews. The analysis identified key potential failure areas within the code.

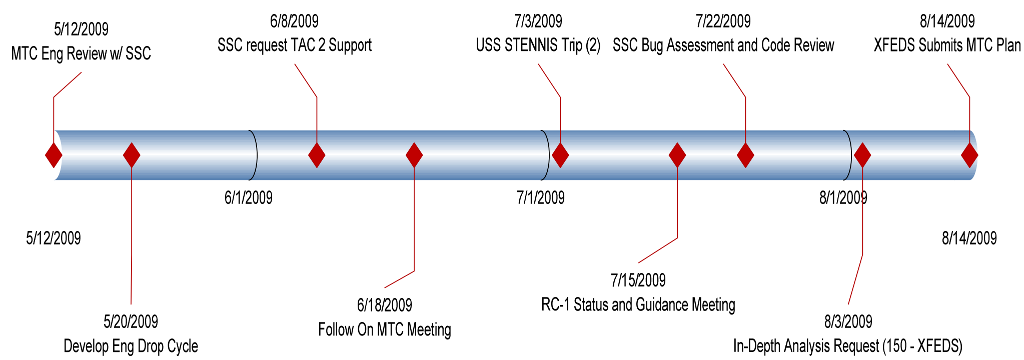


Figure 7. Phase 2 Milestones

The *RITE* Solution

In August 2009, Phase 3 was initiated. The activities undertaken in this phase were the combination of several independent testing process changes previously implemented in segments of the MGF program. Key actions taken to successfully correct the MTC repair included:

- Incorporating the use of automated code assessments, memory usage analysis, and debuggers.
- Establishing a tiger team (2 team: contractor and SSC-Pac), working from a common work list. Responsibilities were divided between the teams with the SSA team primarily responsible for engineering oversight and source code analysis. The SSA became the source code authority and a repository was established at the government site.

- Implementing one-week build cycles with relevant sections of current code (RC) being distributed from the repository each week. The weekly schedule was fixed with specific functions being performed (and monitored) daily.
- Testing of each incremental build was conducted by the SSA and incorporated into the baseline release version.
- Working from the existing STRs, STRs were reviewed and updated to accurately explain the root causes of general symptoms. No new MTC STRs were to be generated unless new symptoms were observed and were not already covered.

MTC Lessons Learned

By implementing the activities discussed above, the SSA was able to identify, track and repair the causes of the long-standing Stennis MTC problems. Using the *RITE* processes, problems that had lingered for over fifteen months were satisfactorily repaired in less than three. Using an integrated Government/industry team and employing code level assessments, strong configuration control, and diligent development oversight, software code issues that had gone undetected throughout many versions of the software release were repaired. Key lessons learned have been incorporated into the evolving *RITE* initiative and are highlighted below.

- **Delivery code assessments**
    - Established standardized process for acceptance product delivery are accepted
- **Identified STR causes in code**
    - Debugging and defect isolation
    - Recommended fixes passed back to developer as necessary
- **STR fix management and oversight**
    - Was correct thing fixed?
    - Reduce churn with fix attempts
    - Used SECP (Software Engineering Change Proposals)
    - Final fix incorporated code from both the government and contractor team SSC, XFEDS, NGMS

RITE Benefits

Although *RITE* is a relatively new initiative, it is achieving positive results in the Navy's C2 development activities and providing significant benefits to the program office. Benefits include the following:

Budget Multiplier

By allocating time and budget dollars to the earlier stages of software development, the Government is getting "more bang for its buck." As shown in Figure 4, for every dollar that is spent in the implementation phase, you get more than a 10 times multiplication effect when compared to dollars spent during the maintenance stage. Therefore, *RITE's* early and frequent defect detection activities are ultimately saving overall Navy C2 program dollars for the Government.

Increased Product Features or Reduced Cost

By reducing the TOC of Navy C2 programs, the Navy is theoretically faced with the decision to either reduce its overall program budgets or increase the number of component features assigned to future programs. It is recognized that because the budget categories (RDT&E verses OMN) are distinctly different, the ability to move money between the categories is not as simplistic as the author is suggesting. However, it is believed that over time, reductions in software rework will allow the OMN (maintenance) budgets to be reduced in order to increase RDT&E expenditures. Much like the RDT&E, budgets have shrunk to cover the costs associated with the rework needed for previous systems.

Improved Schedule Performance

The ability to accurately predict Navy C2 software delivery schedules, a weakness under the existing release cycle process, has been improved with *RITE*. Because the *RITE* model is based upon early and frequent interaction between the developer and the Government project team, there are fewer project surprises. The Program Manager has the opportunity to adjust the project execution, including the allocation of additional development staff, the adjustment to key milestone delivery dates, or even the reduction of product features if project issues are discovered early. Also, because *RITE's* automated and focused testing identifies software development problems earlier in the cycle, there is less corrective action required during the later stages, allowing the software development team to more accurately assess the impact of the defects and the time it will take to correct. “Focused testing” allows for testing and retesting of specific problem areas in a surgical precision model and does not require the (re) testing of the total deliverable whenever defects are repaired. This makes it less manpower intensive and therefore less expensive to conduct.

RITE is not only about highlighting issues; it also provides continual status updates to the Program Manager, demonstrating positive tangible results of project successes. The ability to repeatedly integrate and compile the binary code into executable software validates that the system is performing as expected. *RITE*, through its use of automated testing tools, also provides solid development metrics that can be used to track the project progress and process improvement.

Lastly, because the *RITE* process ensures that the software being developed is monitored and corrected throughout the development cycle, when the system enters the Approval stage, including the security accreditation and certification requirements and the operational testing (OT) program, the success rate is expected to be higher, thereby reducing the time and cost of performing this stage.

Improved Product Quality

A major benefit of the *RITE* process is that the released system (end product) is delivered to the warfighter with fewer defects, thereby reducing the need for continual software rework using the trouble reporting process. This is due to the improved testing processes, as well as the detailed acceptance criteria derived from the design specifications. The Government is able to hold the developers accountable for the quality of the software being delivered and not just for their ability to submit a “deliverable” at a designated delivery date. The importance of delivering a quality product cannot be overstated. Besides the benefit of reducing the costs associated with the product rework, the inconvenience to the user caused by numerous system errors and the loss of credibility and confidence in the delivered product has long-term ramifications to future development programs. Studies have shown that most customers place a higher importance on quality than on timely delivery. As critical operational components of the US Navy, it is paramount that Navy C2 systems perform satisfactorily when released for operational use.

Shortened Release Life Cycle

Navy C2 is able to deliver new functionality to the end user sooner due to the reduction in timely and costly defect repair. One of the major game changers for *RITE* has been the ability to manage and control the software source code by acquiring "unlimited licensing rights" for the Government. This licensing agreement has established a partnership between the software developer and the Government's project team, giving the project team authority to require more frequent submission (drops) of the development product. By implementing a more frequent validation and inspection program, which requires the integration and testing of the software at more frequent intervals, the team has identified potential software defects earlier in the development cycle. Additionally, being able to view the development program down to the source level has allowed the project team to more accurately project program schedule and cost and, ultimately, has led to more realistic schedule development. In many programs, deliverable due dates were arbitrarily set early in the contracting stage and rarely adjusted to accommodate development progress.

Contract Competition

Lastly, and perhaps most importantly, having unlimited rights to the noncommercial computer software source code and the ability to share this code with 3rd party software vendors, greatly improves the Government's ability to implement a true competitive contracting environment, and will ultimately help improve product quality while reducing the overall program cost. The Government has begun to lower the barriers to entry into this market and has reduced the program risk, thereby improving its negotiation position.

2011 And Beyond

Future Implementation Activities

RITE is a dynamic program with much left to accomplish. PMW-150 has taken an aggressive approach to changing its software development life cycle management and is currently reviewing plans for FY 2011 and beyond. Areas for consideration are shown in Table 1 and final decisions will be made as part of the annual budget process.

Table 1. Future Program Considerations

Pillar	Next Steps
Contract	▪ ***Detailed Arch and Design specs as part of contract. Include stakeholders in process*** ▪ ***Refine acceptance criteria*** ▪ ***Establish defined stages for deliverables/testing*** ▪ ***Define how to manage large software library and to resolve legacy issues through future maint contracts*** ▪ ***Refine CDRLs/DIDs***
Process	▪ ***Determine correct/applicable perf Metrics for Project*** ▪ ***Determine metrics for "acceptable risk" to assist with test exit criteria*** ▪ ***Establish Performance Scorecards*** ▪ ***Measure respective Stage Level of Effort (LOE)*** ▪ ***Validate cost savings assoc with "rework" reduction*** ▪ ***Improve tool set (dependency tool)*** ▪ ***Focus testing on "what changed" not total build*** ▪ ***Increase number (and fidelity) of operationally based "test cases"***
Infrastructure	▪ ***Implement RITE Conops into MGF POR*** ▪ ***Establish Applications Store as part of D2*** ▪ ***Coordinate/Share with Industry (3rd Party developers)*** ▪ ***Establish partnerships with other testing facilities*** ▪ ***Expand RITE to Team SPAWAR and DISA—provide programmatic support***
Organization	▪ ***Personnel technical qualifications—"Trusted Agent" role requires diff tech skills*** ▪ ***Institutionalize RITE development model for all s/w dev*** ▪ ***Conduct staffing assessment—compare to competency requirements***

Conclusion

It is widely accepted in the software development industry that early detection and repair of software defects is most cost effective. Early detection also contributes to enhanced software quality and better schedule performance. However, to achieve this requires key changes in the way the Navy C2 program office conducts its software development programs. Fundamentally, the relationship between the development contractor and the Government project team needs to change. The Government needs to exercise its oversight responsibility throughout all stages of the life cycle. To achieve the needed changes, PMW-150 has implemented the *RITE* initiative based upon four pillars consisting of contract standards establishing the Government's responsibilities in the development processes; life cycle process changes to implement the automated and focused integration and testing needed to reduce defect rework requirements; infrastructure enhancements required to expand the communication, cooperation, and collaboration amongst all stakeholders in the Navy C2 program; and, lastly, organizational changes to ensure that the Government has the requisite skills to monitor and support the development contractors in the performance of their contractual obligations.

Although additional capture and analysis of *RITE* metrics is needed to fully validate the total program benefits, early indications are that changes implemented with the *RITE* initiative provides the Navy C2 Program Office a potentially significant return on its investment and should be considered for broader Navy software program adoption.

References

USD (AT&L). (2008, December 22). *Operation of the defense acquisition system* (DoD Instruction 5000.2). Washington, DC: Author.

Soni, M. (2010). *Defect prevention: Reducing costs and enhancing quality*. Retrieved from http://software.isixsigma.com/library/content/c060719b.asp

Improved Software Testing for Open Architecture

Valdis Berzins—Valdis Berzins is a Professor of Computer Science at the Naval Postgraduate School. His research interests include software engineering, software architecture, computer-aided design, and software evolution. His work includes software testing, reuse, automatic software generation, architecture, requirements, prototyping, re-engineering, specification languages, and engineering databases. Berzins received BS, MS, EE, and PhD degrees from MIT and has been on the faculty at the University of Texas and the University of Minnesota. He has developed several specification languages, software tools for computer-aided software design, and fundamental theory of software merging.

Valdis Berzins
Computer Science Department
Naval Postgraduate School
Monterey, CA 93943
Phone: 831-656-2610
E-mail: berzins@nps.edu

Paul Dailey—Paul Dailey is a systems engineer with the Naval Postgraduate School and has worked for the Department of the Navy for seven years, including as a test and evaluation engineer for the Naval Surface Warfare Center, Port Hueneme Division, Detachment Louisville from 2002 to 2009. He holds a MS in Systems Engineering from the Naval Postgraduate School and a BS in Electrical Engineering from the University of Louisville. He is currently pursuing a PhD in Software Engineering from the Naval Postgraduate School, focusing his research on the automated testing of software.

Paul Dailey
Software Engineering PhD Student
Naval Postgraduate School
Monterey, CA 93943
E-mail: prdailey@nps.edu

Abstract

Applying traditional manual US Navy testing practices to OA systems will limit many benefits of OA, such as system scalability, rapid configuration changes, and effective component reuse. Pairing profile-driven automated software testing with test reduction techniques should enable these benefits and keep resource requirements at feasible levels. Test cases generated by operational profiles have been shown to be more effective than those developed by other methods, such as random or selective testing, and more resource-efficient than exhaustive approaches. This research effort increases the fidelity of the operational profile, creating an environment model referred to as a High-Fidelity Profile Model (HFPM) that can statistically describe individual software inputs. Samples from the HFPM's probability distributions can generate operationally realistic or overly-stressful test cases for software modules under test. This process can be automated and paired with output checking functions, enabling automated effective software testing, and potentially improving reliability. Such models would be ideal for US Navy Open Architecture (OA) software because of the defined interface standards. HFPMs can enable effective testing in software reuse applications and are ideal for testing multiple releases of maturing software. This research defines the HFPM, presents a methodology to develop, validate, and apply it.

Keywords: Software Testing, Software Reliability, Operational Profile, Software Reuse, Open Architecture

Introduction

Current software testing methods will limit some of the key benefits that Open Architecture (OA) can provide for the US Navy. More specifically, the ability to rapidly change a system's configuration in order to meet new requirements is possible when using an OA but if current Test and Evaluation (T&E) practices and policies are applied, the updated system will likely not be fielded in a timely manner. With the ability to rapidly update software comes a need to rapidly field that software (Berzins & Dailey, 2009).

In order to rapidly field US Navy combat and weapons system software, two new approaches are required. First, the current software testing process needs to be changed from a manually conducted process to an automated process that provides better test coverage for a given cost and period of time. Second, the total amount of testing required should be safely reduced to a minima acceptable level. Instead of conducting complete end-to-end testing after every configuration change, testing should only be conducted where necessary. The ability to test more rapidly while providing better coverage combined with the ability to determine when retesting is not necessary should enable the ability to rapidly field OA combat and weapon system software (Berzins & Dailey, 2009).

Model Driven Automated Software Testing

The recommended automated software testing process, outlined in detail by Dailey, Berzins, and Luqi (2009; 2010), focuses on developing a High-Fidelity Profile Model (HFPM) for each software component under test (SUT) and then using it to automatically generate test cases, execute test cases, check SUT outputs and analyze the results. Analyzing the results automatically can be challenging for services with new or modified requirements, but can be accomplished easily and economically for components whose behavior is not supposed to change from the previous release. This can be done by running both the new and the previous version of the software component on each input generated by the HFPM and then comparing the results. That process is easy to automate.

The HFPM contains High-Fidelity Profiles (HFPs), which are validated probability distribution functions (PDFs) that characterize the component's environment. Operationally-realistic or stress-inducing test cases are automatically created by sampling from those HFPs and processing the samples through test case generation algorithms. Once generated, the test cases are queued up for automated SUT execution by the software tools implementing the HFPM. Following execution, output analysis algorithms integrated into the HFPM, are used to automatically check the test case outputs and calculate the resulting reliability of the SUT with respect to the HFPs used in testing. The overall process (Figure 1) and the HFPM functional concept (Figure 2) are outlined below. For a more detailed description, see Dailey and Luqi (2010).

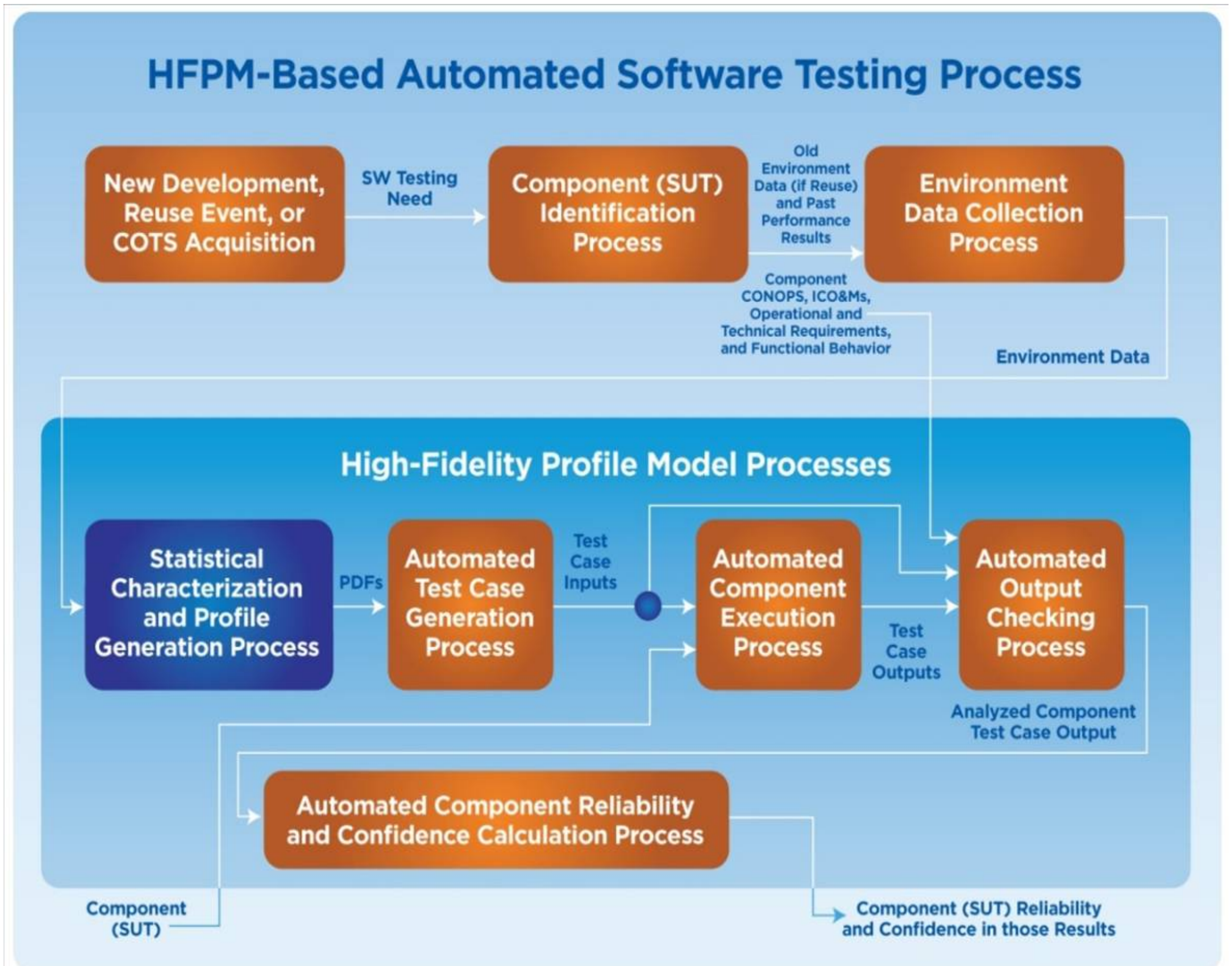


Figure 1. HFPM-Based Automated Testing Process
(Dailey & Luqi, 2010)

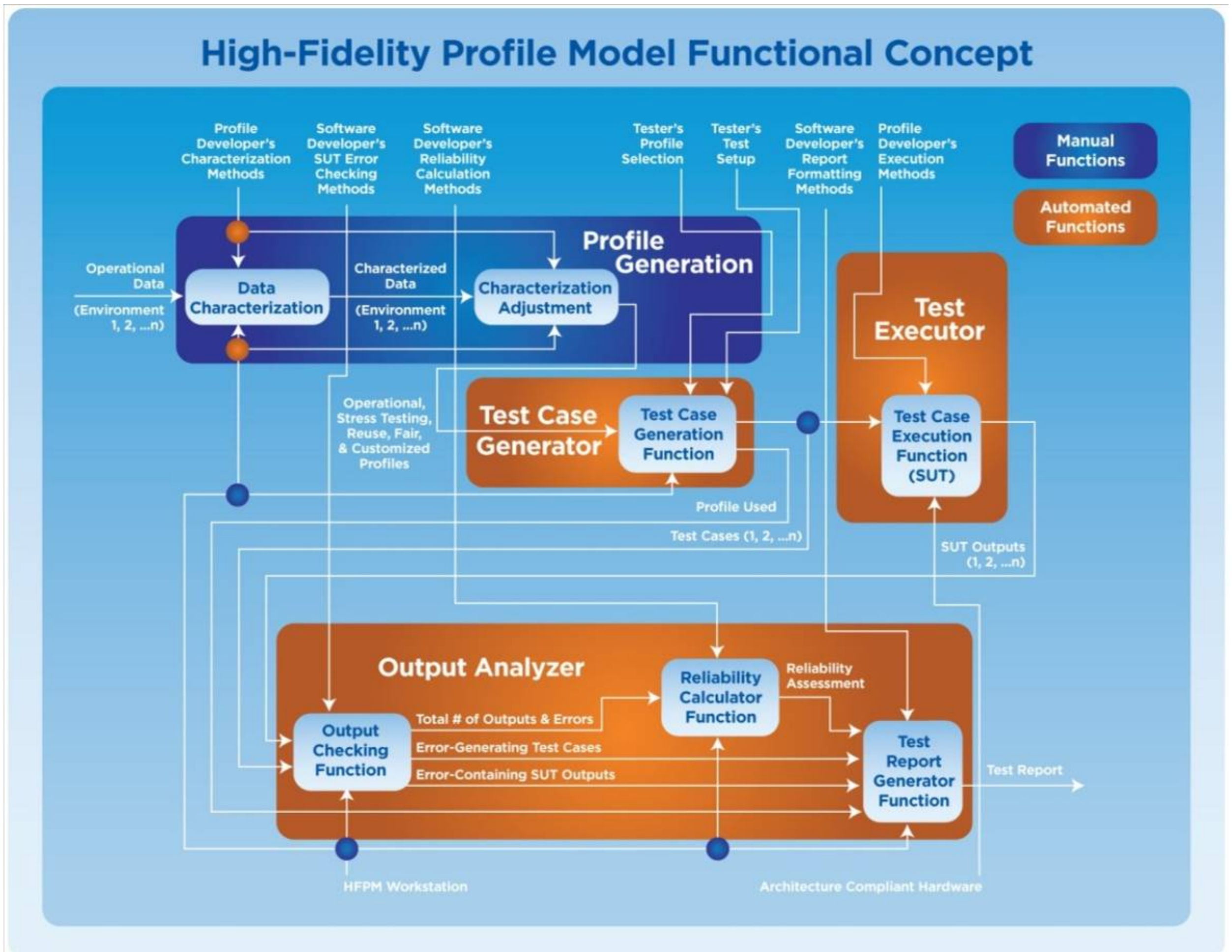


Figure 2. HFPM Functional Concept
(Dailey & Luqi, 2010)

Application to US Navy Acquisition

In order to make the process described above work for US Navy acquisition, it should be employed in a way that enables the HFPM model to be used by all relevant commands that play a role in software development or T&E. This type of focus provides a common practice across the acquisition testing community with the ability for customization for specific roles. The HFPM should be developed in parallel with new components and should be created for a component when acquired off the commercial shelf or in reuse applications where one does not yet exist. The research, development and acquisition agency should use the HFPM to check each component as it is developed and/or integrated into its specific operating environment until such time that the component is ready for Independent Validation and Verification (IV&V). At that time, the component along with the HFPM, are passed to an IV&V test team, which has the ability to modify the HFPs as desired, for Developmental Testing (DT). The IV&V test team can be another group of independent testers in the same command as the software developers or they can be part of the In-Service Engineering Agency (ISEA) responsible for maintaining the software once fielded. This level of DT is generally the most stressful type of testing, focused on identifying bugs by wider ranges of test inputs than expected in the nominal operating environment.

Once the DT IV&V testing is complete, the results along with the profile(s) used in the testing are passed back to the software development team. If bugs exist that require correction, the software development team can make the proper changes, update the configuration, test internally and send out for another round of DT IV&V. If the software has reached a desired level of maturity for field use, the software component is sent out for Operational Test (OT) certification. OT should be conducted by a command outside of the software development and ISEA, such as the Commander Operational Test and Evaluation Force (COMOPTEVFOR), ensuring independent certification and utilizing more operationally realistic HFPs for test case generation. Often however, such OT agencies do not have the technical expertise to evaluate all types of software. In such cases, members of the software development team can become OT trusted agents and provide support for OT evaluation under control and supervision of the primary OT command. If OT is unsuccessful, the test results and profile(s) used are sent back to the software development team for analysis and correction. Upon successful completion of OT, results are passed back to the software development team and the software is certified for deployment. This concept is illustrated in Figure 3.

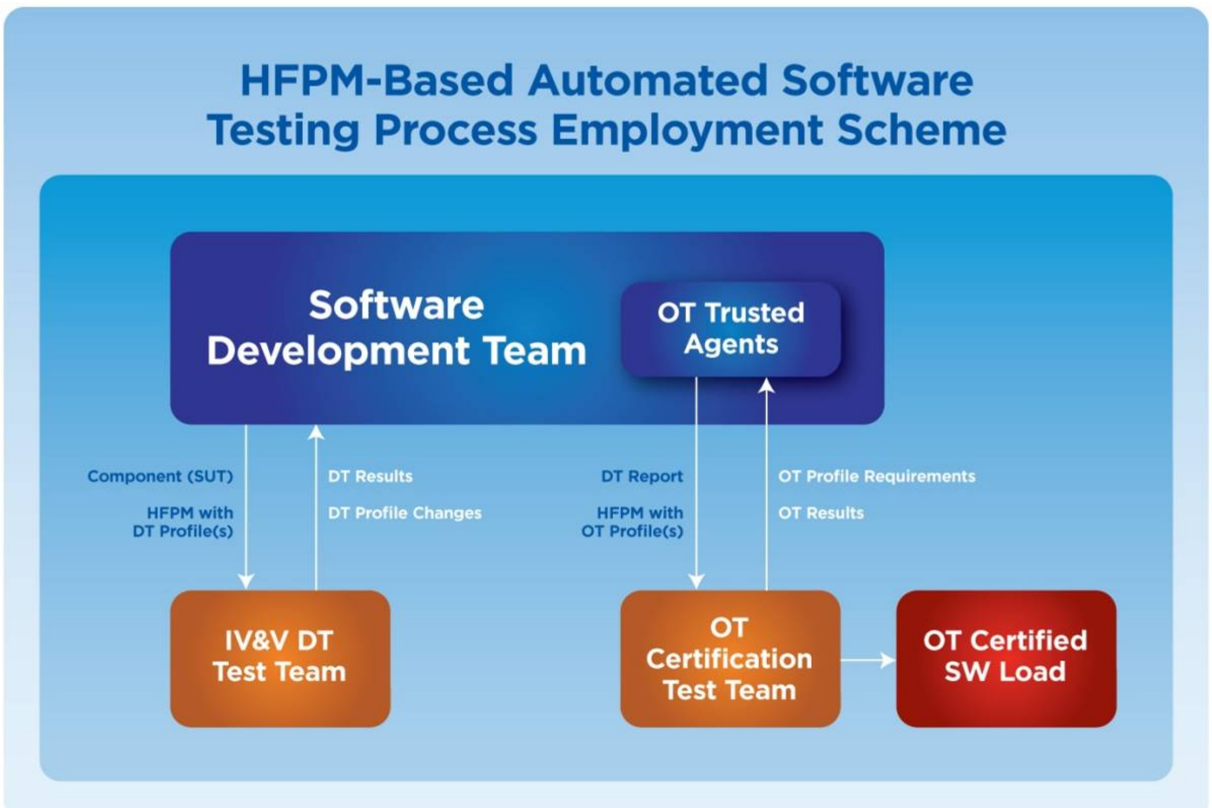


Figure 3. HFPM-Based Automated Software Testing Process Employment Scheme
(Dailey & Luqi, 2010)

Deriving HFPs from Historical Data

The most important element of the HFPM-driven automated testing process is deriving the HFP(s) for use in automated test case generation as the reliability calculated during testing is only accurate relative to the HFP(s) used. If we develop HFPs characterizing several different deployment environments, the methods described in this paper can be used to determine the reliabilities to be expected in each deployment environment. These can vary considerably.

Collecting Historical Data

The first step in deriving HFPs from historical data is collecting the historical data. To effectively do this, the component to be tested must be understood, including its operational and technical requirements, functional behavior, and expected inputs and outputs. Once all the component inputs and outputs are identified and defined operationally and functionally, the next task is to collect data that can directly or indirectly be used to form characterizations of the expected component inputs in the operating environment.

Depending on the specific application and information available, any type of historical or environment data can potentially be useful in this process. The most ideal case is to obtain actual input data that will be processed by the component in the new environment and directly characterize that data. If this is not obtainable, other indirect but relevant data can be collected and characterized along with information that relates the collected data to the SUT inputs. For applications where the operating environment is not known, a method proposed by Voas (2000) can be helpful if access to the end users during development is possible. In this process, an instrumentation tool is used to collect data from fielded software that can then be used to generate accurate operational profiles. If access to the end users is not possible, it is up to the software development and acquisition team to determine how to best collect useful environment data in each specific application for analysis and HFP generation. Specific methods could include trial data collection efforts during training exercises, Advanced Concept Technology Demonstrations (ACTDs), modeling and simulation, or technical intelligence collection and analysis. Once collected and characterized, indirect data may require further processing by input test case generation algorithms if necessary, in order to transform samples from those characterizations into usable test case inputs.

Characterizing Historical Data

Once a particular set of raw environment data is collected and related to the specific component input(s), the data can be analyzed using one of many established data characterization methods and available commercial tools for HFP PDF generation. One such example is the Matlab® Dfittool application within the Statistics Toolbox® (“Dfittool,” 2009). Regardless of the tool used, parametric methods such as Maximum Likelihood Parameter Estimation, and Maximum *A Posteriori* Probability Estimation, or non-parametric methods such as the Histogram, Kernel Density Estimation (KDE) (Wikipedia et al., 2009), or Parzen Neural Network (PNN) (Trentin, 2006) methods can be applied to generate HFP PDFs using available raw environment data. Parametric methods should be used when an understanding of the data is available prior to characterization. If there is no such prior understanding of the data, nonparametric methods can be used more effectively. The desired tool(s) used to perform the necessary analysis should have the flexibility to modify the method used for calculation in order to compare methods and determine the best type of

PDF fit. The output of this analysis process should be one or more PDFs that can be used to either directly or indirectly generate test case inputs based on samples from those PDFs. Direct examples include applications where a sample from the PDF can be used as a component input. Indirect examples include PFD samples that require further processing in order to generate component inputs. These PDFs are referred to as HFPs in this study.

Simple Example of Deriving HFPs from Historical Data

Dailey (2010) illustrated the concept creating HFPs from collected environment data. In the example, performance data on various small boat platforms from a US Navy study was acquired and modeled using Matlab®. The US Navy data provided the following data on six different types of small boat platforms:

Table 1. Small Boat Collected Data
(Dailey, 2010)

Platform	Max Velocity (kts)	Boat Length (m)	Acceleration (kts/sec)	Deceleration (kts/sec)	Turning Rates (deg/sec)	Speed Loss in Turns (deg/sec)
Boghammer	40	13	1.5	4	10	10
FB 38	50	11.85	2.5	4.3	15	12
7m RHIB	27	7.25	2.5	4.2	28	7
Boston Whaler	36	6.78	2.5	4	30	8
Zodiac	23	4.7	6.25	4.5	32	5
Wave Runner	44	3.66	6.25	4.4	47	15

The data in Table 1 was entered into Matlab® and then characterized using the Statistics Toolbox® *dfittool* resulting in a HFP PDF and inverse cumulative distribution function (Inverse CDF) for each of the parameters. Due to the limited number of points per parameter and the lack of specific knowledge on the specific type of distribution applicable to each parameter, the nonparametric KDE calculation was used to characterize the data. The KDE function is:

$$\hat{f}(x) = \frac{1}{n}\sum_{i=1}^{n} K\left(\frac{x - x_i}{\;}\right)$$

where *K* is some kernel and h is a smoothing parameter called the bandwidth (“Kernel Density,” n.d.). In this case, *K* was taken to be a standard Gaussian function.

Several iterations of characterizations were generated taking into account the actual data as well as establishing logical finite ranges for each parameter. The result is a collection of distributions that effectively describes a notional small boat platform from a technical perspective. Two HFPs generated from this data can be seen below.

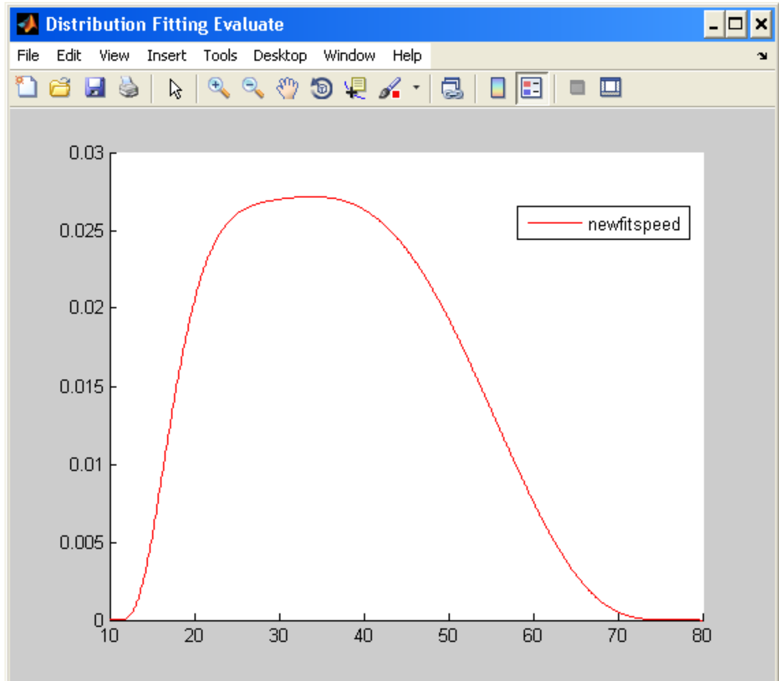


Figure 4. Notional Small Boat Maximum Velocity PDF (Knots)
(Dailey, 2010)

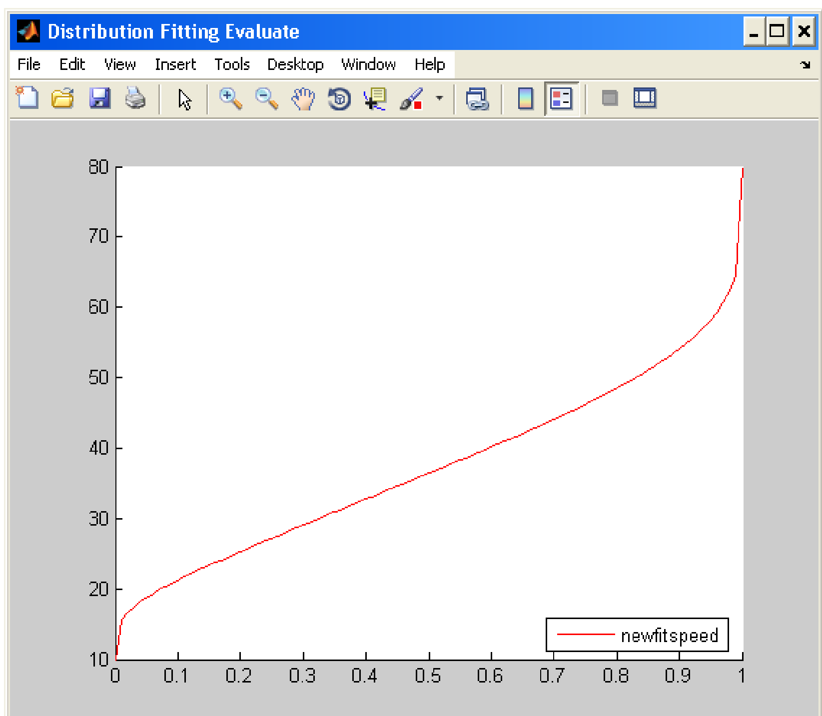


Figure 5. Notional Small Boat Maximum Velocity Inverse CDF (Knots)
(Dailey, 2010)

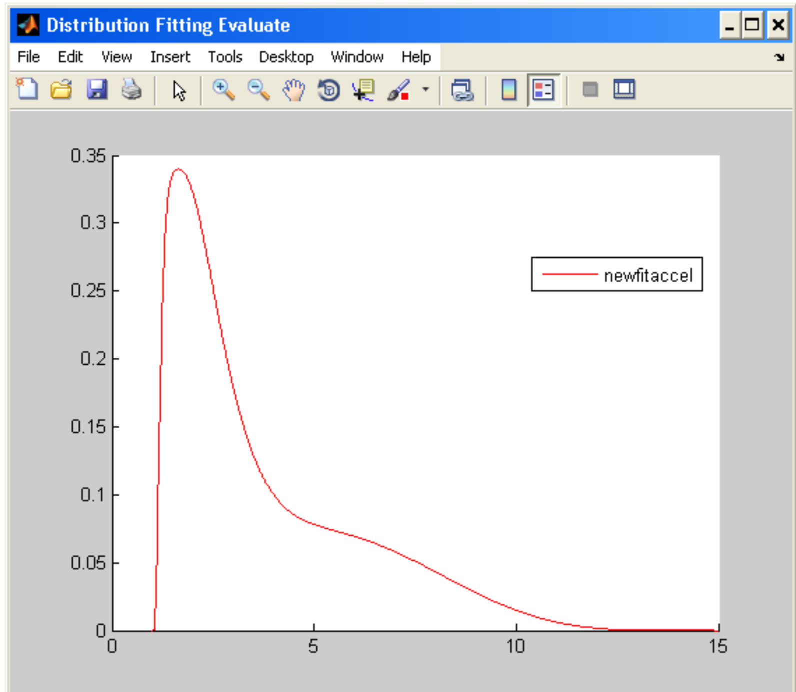


Figure 6. Notional Small Boat Acceleration PDF (Knots/Second)
(Dailey, 2010)

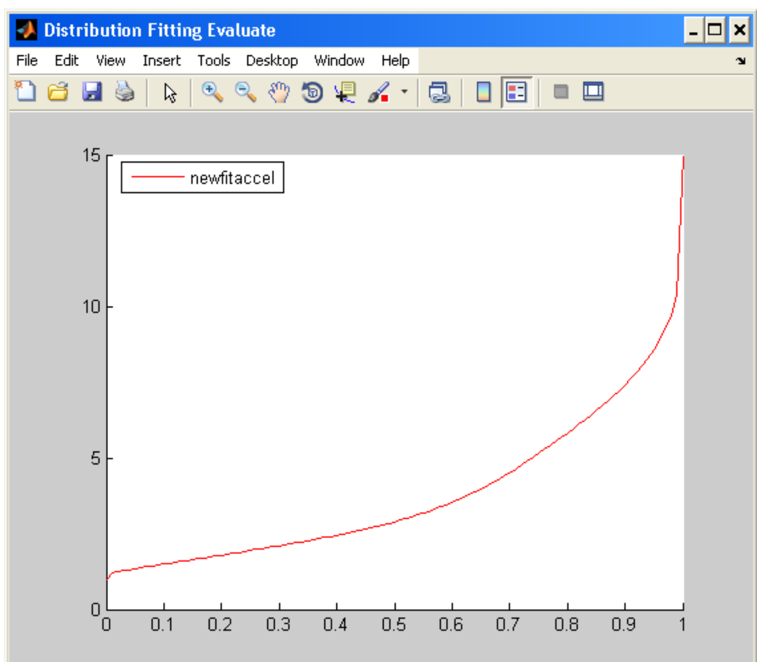


Figure 7. Notional Small Boat Acceleration Inverse CDF (Knots/Second)
(Dailey, 2010)

The HFP functions generated above were exported to the Matlab® workspace for use in automated test case generation as part of a HFPM concept demonstration prototype. For more information, see Dailey (2010).

Validating High-Fidelity Profiles

Since testing results from this process are only valid with respect to the HFP(s) used to generate the test cases, it is important to take measures to check their validity. In any application, a qualitative analysis of the HFP(s) should be conducted by subject matter experts to ensure that the derived profile(s) provide adequate coverage for testing. In addition to the qualitative assessment, it would be very useful to define a quantitative process to perform this function. When trying to determine the best characterization method, it is possible to compare the different methods taking into account the methods themselves as well as their results. Various methods currently under investigation to assess the best characterization method include the use of Bayesian Information Criterion (BIC) and goodness of fit tests.

BIC can be used to compare multiple alternative parametric models with different numbers of parameters of a particular environment. When estimating parameters using maximum likelihood estimation, it is possible to modify or increase the likelihood using additional parameters, but this also can result in overfitting. In this method, the model with the lowest BIC score has the best fit. This technique does not apply to non-parametric characterizations such as KDE, but is useful for deciding between different parametric techniques. In addition to BIC, other similar approaches, such as the Akaike Information Criterion (AIC), also exist. BIC applies a stronger penalty than AIC for having additional parameters. The formula for the BIC is as follows:

$$(-2)\ln\, p(x|k) \approx BIC = (-2)\ln(L) + (k)\ln(n)$$

where *x* is the observed data; *n* is the number of data points in *x*; *k* is the number of free parameters to be estimated; and *L* is the maximized value of the likelihood function for the estimated model ("Bayesian Information," n.d.).

Another approach for comparing different characterization methods is to perform a goodness of fit test for each characterization to the actual empirical data. One specific type of calculating the goodness of fit of a PDF to an empirical distribution is the Cramér-von-Mises criterion. It is defined as:

$$\omega^2 = \int_{-\infty}^{\infty} [F_n(x) - F^*(x)]^2 dF^*(x)$$

where $F^*(x)$ is the characterized distribution and $F_n(x)$ is the empirical environment data distribution ("Cramér-von-Mises," n.d.).

The methods described above are useful for comparing different HFPs to determine the best method. Ongoing research is being conducted to determine the level of confidence in the HFP with respect to the sample size of empirical environment data. This would be beneficial as it can be used to determine how much environmental data collection is adequate.

Deriving Stress-Testing HFPs from Historical Models

By definition, stress testing exercises a software system beyond the range of normal operating conditions. There are two basic approaches to this–black box and clear box. Black box approaches can be combined with the profile model transformations described in this section to carry out automated stress testing that can support statistical reliability estimates relative to the stress testing profile(s) (PDF(s)). The black box approaches described in sections 6.1-6.3 can be combined with the method for reducing retesting of reusable components described in Berzins and Dailey (2009) to eliminate redundant repetition of test cases from the previously tested ranges.

Clear box approaches are heuristic methods that seek to uncover particular types of errors. Although clear box criteria can be applied using stress-profiles, other methods should also be considered, as discussed in more detail in sections 6.4 and 6.5.

Standard Deviation Based Methods

The simplest kind of stress testing profile is based on the mean and standard deviation of the HFPM that characterizes the expected operating conditions (Berzins & Dailey, 2009). This approach is applicable to numerical data types and uses a distribution that exercises two intervals symmetrically placed about the mean, from one to N standard deviations set off from the mean in both directions. The parameter N determines how far beyond the expected operating range will be exercised by the stress test. We recommend a series of stress tests with increasing values of N such as (10, 100, 1000, …) up to the entire range supported by the underlying data type.

The approach can readily be generalized to vector data types by choosing a uniform distribution that takes the form of a ring (in 2 dimensions) or a shell (in 3 or more dimensions). The distribution is centered on the mean of the HFPM, and the radius from the center ranges from 1 to *N* standard deviations. If the HFPM is not isotropic (not the same in all directions), an ellipsoid with different radii along each axis can be used, derived from the covariance matrix of an HFPM over a 2 or more dimensional input space.

Scale Expanding Transformations

Another approach that works for numerical or vector valued inputs is to use a scale expanding transformation. If the HFPM is a distribution *P(x-m)* where m is the mean of the HFPM, then the stress testing profile derived via the approach in *P((x-m)/s),* where s is a numerical scale factor. The stress testing profile then has the same mean as the HFPM, and a standard deviation that is s times larger. The shape and orientation of the stress profile are similar to the original, but spread out more by a factor of *s*, which is similar to the parameter *N* in the previous section. We recommend a sequence of tests with *s* = [10, 100, 1000, …] for applying this method.

Probability Scaling Transformations

The approaches described in sections 6.1 and 6.2 apply only to numerical or vector-valued input data. In contrast, probability scaling transformations apply to any kind of input data, including discrete enumerations such as classification categories and other non-numerical data types. For a HFPM with a distribution *P(x)* the stress testing profile derived using this approach is proportional to $P(x)^{1/N}$, where *N* is a numerical parameter with *N* >1, and where the proportionally constant must be chosen to normalize the distribution to make all probabilities add up to 1. This family of transformations increases the probabilities of rare events and decreases the probabilities of the frequent ones, as illustrated by the example shown in Table 2.

Table 2. Original and Derived Probabilities

	Original	N = 2	N = 3	N = 4	N = 5	N = 10	N = 15	N = 20
P1	0.88888889	0.670925	0.526601	0.432891	0.369481	0.233181	0.134859	0.128692
P2	0.1	0.225035	0.254214	0.250707	0.238684	0.187417	0.128468	0.12409
P3	0.01	0.071162	0.117996	0.140983	0.150599	0.14887	0.12206	0.119418
P4	0.001	0.022504	0.054769	0.079281	0.095022	0.118252	0.115971	0.114922
P5	0.0001	0.007116	0.025421	0.044583	0.059955	0.093931	0.110186	0.110595
P6	0.00001	0.00225	0.0118	0.025071	0.037829	0.074612	0.10469	0.106432
P7	0.000001	0.000712	0.005477	0.014098	0.023868	0.059266	0.099468	0.102425
P8	0.0000001	0.000225	0.002542	0.007928	0.01506	0.047077	0.094506	0.098568
P9	0.00000001	7.12E-05	0.00118	0.004458	0.009502	0.037395	0.089792	0.094857

Table 2 shows an original PDF and a series of transformed and renormalized derived stress testing PDFs. Note that the probabilities in each column add up to 1 and that the original distribution spans a wide range of frequencies of occurrence. These distributions are shown as bar graphs in Figure 8.

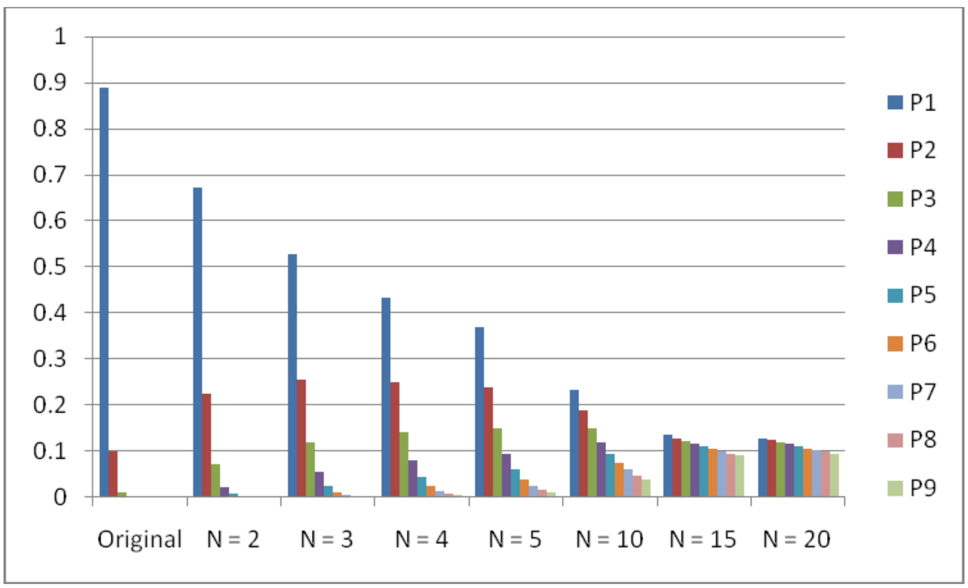


Figure 8. Original and Derived Probabilities

The transformations increase the proportions of the rare cases in the stress testing samples, while preserving the rank ordering of the probabilities. The degree of enhancement of the rare events increases with the parameter *N*.

Dominance Relations and Stress Testing

Stress testing does not have to be done solely using HFPM's. Another useful approach is based on the concept of dominance. One test case *dominates* another one if the first one will expose at least as many software faults as the second, and may expose more. Even if there is not a single test case that dominates all of the others, often there will be some that are more likely to expose errors than others. This approach is particularly useful when the tester is focusing on a specific class of errors. Many of the commonly used testing heuristics are based on this idea and can contribute to efficient testing. A representative sample of these is listed in Table 3, organized by the error type addressed by each.

Table 3. Focused Stress Testing Strategies

Error Type	Heuristics for choosing stress test cases
Numeric Overflow	Largest and smallest representable numbers
Buffer Overflow	Very long input string
Free Storage Overflow	Create many new objects
Wrong Conditional Logic	Data values close to the both sides of an interval boundary
Unprotected Pointers	Null pointer
Unprotected Division	Zero

Coverage Criteria and Stress Testing

Traditional coverage criteria, such as statement coverage and branch coverage, are useful for checking low probability paths through the software. This can be an important defense against unwanted features deliberately placed in the code by malicious insiders. For example, an "Easter Egg" is a hidden feature function in the software that is triggered only when a particular input is provided. Such code is typically deliberately hidden and can easily be made statistically invisible to black box testing approaches. For example, if the function is triggered only when a particular input sting is provided the probability of detection by black box testing is 1 in 88^n, where n is the number of characters in the input and we are assuming all the characters on a standard keyboard can be used. For a field of length 30 the number of test cases needed to detect such a path this is about 2.16×10^{58}, which is not technically or economically feasible.

However, a branch coverage criterion coupled with a constraint logic solver for finding test cases to exercise infrequent branches has be found to be effective at detecting such faults (Molnar, 2008).

Conclusions

Effective and cost-efficient testing for US Navy OA software can be achieved by a mixture of automation methods to determine which tests can be safely eliminated by reusing previous test results, and methods for choosing test cases that are most likely to expose errors without duplicating coverage of other test cases.

This paper explains how automated testing can be systematically performed based on historical data, in a way that exposes the most frequently manifesting errors earliest in the process. We also identify some of the weaknesses of purely statistical approaches to testing and identify methods for overcoming these weaknesses.

References

Bayesian information criterion. (n.d.). Retrieved March 2010, from Wikipedia, http://en.wikipedia.org/wiki/Bayesian_information_criterion

Berzins, V. & Dailey, P. (2009). How to check if it is safe not to retest a component. In *Proceedings of the Sixth Annual Acquisition Research Symposium*. Monterey, CA: Naval Postgraduate School.

Cramér-von-Mises criterion (n.d.). Retrieved March 2010, from Wikipedia, http://en.wikipedia.org/wiki/Cram%C3%A9r%E2%80%93von-Mises_criterion

Dailey, P., & Luqi. (2010, March). *Profile-based automated testing process for open architecture track-processing software* (NPS-CS-10-005). Monterey, CA: Naval Postgraduate School.

Dailey, P. (2010, June (est.)). *High-fidelity profile-based automated testing of open architecture track-processing software*. Unpublished doctoral dissertation. Monterey, CA: Naval Postgraduate School.

Dfittool. (n.d.). Retrieved March 2010, from MathWorks™, http://www.mathworks.com/access/helpdesk/help/toolbox/stats/dfittool.html

Kernel density estimation. (n.d.). Retrieved October 2009, from Wikipedia, http://en.wikipedia.org/wiki/Kernel_density_estimation

Molnar, D., Godefroid, P., Levin, M. (2008). Active property checking. *EMSOFT 2008.*

Musa, J.D. (1993). Operational profiles in software reliability engineering. *IEEE Software*, *10*(2), 14–32.

Trentin, E. (2006). Simple and effective connectionist nonparametric estimation of probability density functions. *ANNPR 2006, LNAI 4087*, 1–10.

Voas, J. (2000, March/April). Will the real operational profile please stand up? *IEEE Software*, 87–89.

2003 - 2010 Sponsored Research Topics

Acquisition Management

- Acquiring Combat Capability via Public-Private Partnerships (PPPs)
- BCA: Contractor vs. Organic Growth
- Defense Industry Consolidation
- EU-US Defense Industrial Relationships
- Knowledge Value Added (KVA) + Real Options (RO) Applied to Shipyard Planning Processes
- Managing the Services Supply Chain
- MOSA Contracting Implications
- Portfolio Optimization via KVA + RO
- Private Military Sector
- Software Requirements for OA
- Spiral Development
- Strategy for Defense Acquisition Research
- The Software, Hardware Asset Reuse Enterprise (SHARE) repository

Contract Management

- Commodity Sourcing Strategies
- Contracting Government Procurement Functions
- Contractors in 21st-century Combat Zone
- Joint Contingency Contracting
- Model for Optimizing Contingency Contracting, Planning and Execution
- Navy Contract Writing Guide
- Past Performance in Source Selection
- Strategic Contingency Contracting
- Transforming DoD Contract Closeout
- USAF Energy Savings Performance Contracts
- USAF IT Commodity Council
- USMC Contingency Contracting

Financial Management

- Acquisitions via Leasing: MPS case
- Budget Scoring
- Budgeting for Capabilities-based Planning

- Capital Budgeting for the DoD
- Energy Saving Contracts/DoD Mobile Assets
- Financing DoD Budget via PPPs
- Lessons from Private Sector Capital Budgeting for DoD Acquisition Budgeting Reform
- PPPs and Government Financing
- ROI of Information Warfare Systems
- Special Termination Liability in MDAPs
- Strategic Sourcing
- Transaction Cost Economics (TCE) to Improve Cost Estimates

Human Resources

- Indefinite Reenlistment
- Individual Augmentation
- Learning Management Systems
- Moral Conduct Waivers and First-tem Attrition
- Retention
- The Navy's Selective Reenlistment Bonus (SRB) Management System
- Tuition Assistance

Logistics Management

- Analysis of LAV Depot Maintenance
- Army LOG MOD
- ASDS Product Support Analysis
- Cold-chain Logistics
- Contractors Supporting Military Operations
- Diffusion/Variability on Vendor Performance Evaluation
- Evolutionary Acquisition
- Lean Six Sigma to Reduce Costs and Improve Readiness
- Naval Aviation Maintenance and Process Improvement (2)
- Optimizing CIWS Lifecycle Support (LCS)
- Outsourcing the Pearl Harbor MK-48 Intermediate Maintenance Activity
- Pallet Management System
- PBL (4)
- Privatization-NOSL/NAWCI
- RFID (6)

- Risk Analysis for Performance-based Logistics
- R-TOC AEGIS Microwave Power Tubes
- Sense-and-Respond Logistics Network
- Strategic Sourcing

Program Management

- Building Collaborative Capacity
- Business Process Reengineering (BPR) for LCS Mission Module Acquisition
- Collaborative IT Tools Leveraging Competence
- Contractor vs. Organic Support
- Knowledge, Responsibilities and Decision Rights in MDAPs
- KVA Applied to AEGIS and SSDS
- Managing the Service Supply Chain
- Measuring Uncertainty in Earned Value
- Organizational Modeling and Simulation
- Public-Private Partnership
- Terminating Your Own Program
- Utilizing Collaborative and Three-dimensional Imaging Technology

A complete listing and electronic copies of published research 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