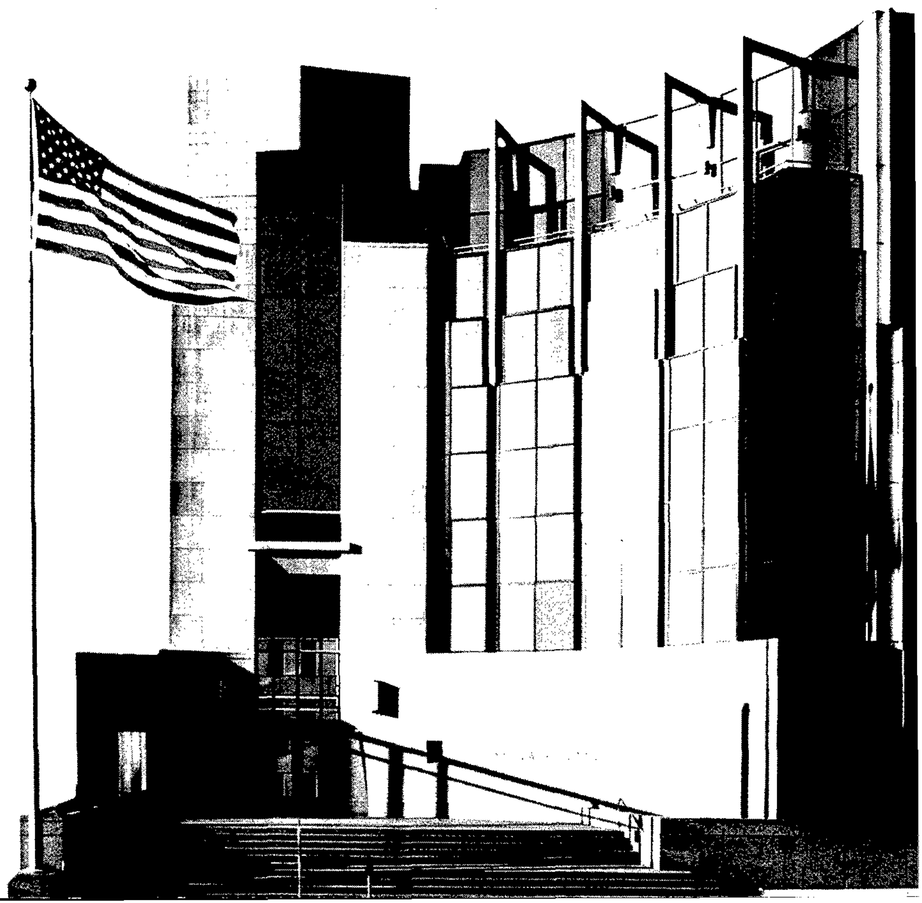
CarnegieMellon
Software Engineering Institute

Software Product Lines: Experiences from the Seventh DoD Software Product Line Workshop

John K. Bergey
Sholom Cohen
Patrick Donohoe
Lawrence G. Jones

March 2005



TECHNICAL REPORT
CMU/SEI-2005-TR-001
ESC-TR-2005-001

Carnegie Mellon
Software Engineering Institute
Pittsburgh, PA 15213-3890

Software Product Lines: Experiences from the Seventh DoD Software Product Line Workshop

CMU/SEI-2005-TR-001
ESC-TR-2005-001

John K. Bergey
Sholom Cohen
Patrick Donohoe
Lawrence G. Jones

March 2005

Product Line Practice Initiative

This report was prepared for the

SEI Joint Program Office
ESC/XPK
5 Eglin Street
Hanscom AFB, MA 01731-2100

The ideas and findings in this report should not be construed as an official DoD position. It is published in the interest of scientific and technical information exchange.

FOR THE COMMANDER

Christos Scondras
Chief of Programs, XPK

This work is sponsored by the U.S. Department of Defense. The Software Engineering Institute is a federally funded research and development center sponsored by the U.S. Department of Defense.

For information about purchasing paper copies of SEI reports, please visit the publications portion of our Web site (http://www.sei.cmu.edu/publications/pubweb.html).

Table of Contents

Abstract

The Carnegie Mellon® Software Engineering Institute held the Seventh Department of Defense (DoD) Product Line Practice Workshop in September 2004. The workshop was a hands-on meeting to share DoD product line practices, experiences, and issues and to discuss ways in which specific product line practices are accomplished within the DoD. Participants reported encouraging progress on DoD software product lines. This report synthesizes the workshop presentations and discussions.

1 Introduction

1.1 Product Line Practice

An increasing number of organizations are realizing that they can no longer afford to develop multiple software products one product at a time: they are pressured to introduce new products and add functionality to existing ones at a rapid pace. They have explicit needs to achieve large-scale productivity gains, improve time to market, maintain a market presence, compensate for an inability to hire, leverage existing resources, and achieve mass customization. Many organizations are finding that the practice of building sets of related systems together can yield remarkable quantitative improvements in productivity, time to market, product quality, and customer satisfaction. Those organizations are adopting a product line approach for their software systems.

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 02].

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 a conceptual framework for product line practice. The SEI *Framework for Software Product Line Practice*[SM] (henceforth referred to as "the framework") describes the foundational product line concepts and identifies the essential activities and practices that an organization must master before it can expect to field a product line of software or software-intensive systems successfully. The framework organizes product line practices into practice areas that are categorized as software engineering, technical management, or organizational management. (These categories represent disciplines rather than job titles.) 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 02], and the latest version is available on the SEI Web site [Clements 04].

® Carnegie Mellon is registered in the U.S. Trademark and Patent Office.

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

The framework's contents are based on information-gathering workshops,[1] extensive work with collaboration partners, surveys and investigations, and continued research. The SEI has also incorporated practices reported at its international Software Product Line Conferences [Donohoe 00, Chastek 02, Nord 04] and collected from the community.

In March 1998, the SEI hosted its first Department of Defense (DoD) product line practice workshop, *Product Lines: Bridging the Gap—Commercial Success to DoD Practice.* Product line practices, DoD barriers and mitigation strategies, and similarities and differences between DoD product line practice and commercial product line practices were discussed and documented [Bergey 98]. Subsequent workshops were held in successive years [Bergey 99, Bergey 00a, Bergey 01, Bergey 03, Bergey 04a]. At all six DoD workshops, the SEI was encouraged to continue to hold other DoD workshops and to continue to share best commercial and DoD practices through these forums.

One of the key outcomes of these workshops was the identification of product line practices that were particularly important to DoD acquisition organizations. This information supported development of a companion to the framework, titled *Software Product Line Acquisition: A Companion to A Framework for Software Product Line Practice* [Bergey 04b] (henceforth referred to as "the companion"). The companion, like the framework, is a living document with the latest version available on the SEI Web site at http://www.sei.cmu.edu/productlines/companion.html.

1.2 About This Workshop

The goals of the Seventh DoD Product Line Practice Workshop in September 2004 were to

- share DoD product line practices, experience, and issues regarding both development and acquisition
- discuss ways to motivate product line efforts in support of DoD systems
- explore ways to initiate software product line adoption in the DoD community

All participants in this workshop were from the DoD acquisition and contractor community. They were invited based on our knowledge of their experience with and commitment to software product lines as either DoD system acquirers or DoD system contractors. Together, we discussed the issues that form the backbone of this report.

The format of this workshop differed somewhat from our past workshops. Participants were invited to make brief presentations about their organizations' activities and interests in software product lines. The intent was to limit discussion during these presentations and to save detailed discussion for later breakout sessions. However, given the richness of the presentations and associated discussions, the group asked to extend the presentation activities in lieu of the breakout sessions. Because of the small group size, the facilitators could accommodate

---

[1] The results of some of these workshops are documented in SEI reports [Bass 97, Bass 98, Bass 99, Bass 00, Clements 01].

this request. After the workshop, the group agreed that this more inclusive participation worked well.

The workshop participants included

- Thomas Baker, Austin Information Systems
- James Baxter, U.S. Army Program Executive Office (PEO) Aviation
- John Bergey, Product Line Systems Program, SEI
- Peter Blankenship, Northrop Grumman
- Stephanie Burns, Rockwell Collins
- Matthew Cerroni, U.S. government systems engineer
- Sholom Cohen, Product Line Systems Program, SEI
- David DeKing, United Defense LP
- Patrick Donohoe, Product Line Systems Program, SEI
- Dave Drum, Argon Engineering Associates
- Edward Dunn, U.S. Navy Naval Undersea Warfare Center (NUWC)
- Scott Harris, Argon Engineering Associates
- William Johnson, PEO Deputy for Open Architecture, U.S. Navy
- Lawrence Jones, Product Line Systems Program, SEI
- Linda Northrop, Director, Product Line Systems Program, SEI
- Frank Polster, U.S. Army Training Support Center (ATSC)
- Dean Runzel, U.S. Army PEO for Simulation, Training, & Instrumentation (STRI)
- Stuart Ware, Argon Engineering Associates

1.3 About This Report

This document summarizes the presentations and discussions from the workshop. This report is written primarily for those in the DoD who are already familiar with product line concepts, especially those working on or initiating product line practices in their own organizations. Acquisition managers and technical software managers should also benefit from this report. Those who desire further background information are referred to the following publications:

- *Basic Concepts of Product Line Practice for the DoD* [Bergey 00b]
- *A Framework for Software Product Line Practice, Version 4.2* [Clements 04]
- *Software Product Line Acquisition: A Companion to A Framework for Software Product Line Practice, Version 3.0* [Bergey 04b]
- *Software Product Lines: Practices and Patterns* [Clements 02]

The next section of this report contains a digest of the presentations. The report concludes with a brief summary.

2 DoD Software Product Line Experiences: A Digest of Participant Presentations

2.1 Introduction

Peter Blankenship of Northrop Grumman gave a keynote presentation. In addition, all participants were invited to give a short presentation about their organizations' product line interests and experiences. A summary of these presentations follows.

2.2 Force XXI Battle Command Brigade and Below (FBCB2)—Peter Blankenship, Northrop Grumman

Force XXI Battle Command Brigade and Below (FBCB2) is an Army tactical command and control system designed and built for on-the-move operations. The primary development contractor is Northrop Grumman Mission Systems (NG) with acquisition services provided by the FBCB2 Program Office (PO), Fort Monmouth, NJ.

A deployed FBCB2 system consists of a network of operational platforms (e.g., tanks, helicopters, operations centers), each with its own FBCB2 configuration of hardware and software. The Tactical Internet (TI) provides the network communications infrastructure. The TI utilizes line-of-sight tactical radios augmented with beyond-line-of-sight satellite communications.[2] The hardware for a platform configuration includes computers, displays, Global Positioning System (GPS) receivers, installation kits, and equipment interfaces. FBCB2 exists on diverse platforms, including the M1A1 Abrams, M2A3 Bradley, M109A6 Paladin, a variety of High-Mobility Multipurpose Wheeled Vehicle (HMMWV) configurations, and Army Aviation UH-60, AH-64, OH-58, and EH-60 helicopters.

FBCB2 fulfills two primary missions for tactical maneuver troops:

1. situational awareness (SA), including
   - Blue Force Tracking (BFT) (e.g., friendly position reports)

---

[2] The tactical radios include Enhanced Position Location Reporting System (EPLRS) and Single Channel Ground and Airborne Radio System (SINCGARS). Satellite communication links include both Military Satellite Communications (MILSATCOM) assets and commercial L-Band links. The Tactical Internet Management System (TIMS) provides network monitoring and management services.

- enemy locations (correlated and uncorrelated spot reports)
- geo-referenced battlefield objects (e.g., minefields, obstacles, supply points)
- battlefield graphics (overlays)

2. command and control via digital messaging between platforms/units and linked to higher echelon systems for handling intelligence, logistics, maneuver control, free text, and other command-and-control functions. Examples include
   - field orders
   - spot reports
   - status reporting
   - audio and visual message alerts (Nuclear, Chemical, Biological [NBC]] alerts and other high-priority messages)

Simply put, FBCB2 provides soldiers with answers to these basic questions:

- Where am I?
- Where are my buddies?
- Where is the enemy?
- What is the environment?

With a clear picture of the battlefield, commanders can make decisions faster and communicate them faster than the enemy can react.

FBCB2 began in 1994 as a prototype demonstration system. The Army has since adopted FBCB2 for widespread use, including fielding the system across a vast number of Army units. The program office has proposed cooperation with the U.S. Marine Corps (USMC) to adopt FBCB2 as its standard BFT and messaging system. FBCB2 must also interface to a variety of Army, other military, and coalition systems. The system matured through a series of tests, including the Task Force XXI Advance Warfighting Experiment (AWE) at Fort Irwin and Limited User Tests. The Army initially fielded FBCB2 with the 4th Infantry Division (Mechanized)—the Army's "First Digitized Division"—in September 2000. The Balkan Digitization Initiative proved the ability of FBCB2 to support BFT, and the system was widely deployed in Afghanistan as part of Operation Enduring Freedom (OEF) and in Iraq as part of Operation Iraqi Freedom (OIF). FBCB2 is currently in use by forces in the Balkans, Afghanistan, and Iraq. In support of OIF alone, the Army has expedited delivery of more than 8,800 systems on three continents, deployed with two services.

FBCB2 is the Army's most successful entrée into battlefield digitization. The program has received several national awards including

- 2002: Top Five Quality Software Project (from the Software Technology Support Center, Hill Air Force Base)
- 2003: Network-Centric Warfare Most Innovative U.S. Government Program (from the Institute for Defense and Government Advancement)

- 2004: Monticello Award as "the information system having the most direct and meaningful impact on human lives" (from *Federal Computer Week*)

As FBCB2 enjoyed widespread acceptance and accolades, the program started to become a "victim of its own success." With the release of each new system, the developers incorporated more capability. Although each of those systems is called FBCB2, they were all derived from the initial operational version and exist in slightly different configurations. Each configuration is a monolithic system, loaded on individual platforms, with parameterization that determines capabilities and interfaces for an individual platform. In reality, a number of very similar systems are maintained and released as separate builds. Weaknesses in this approach and the existing FBCB2 software architecture became apparent. These weaknesses limited the ability of developers to continue evolving FBCB2 both technically and affordably. Maintenance in the field required a dedicated support staff, and many functions could not be used because of a lack of communications bandwidth or other limitations. The PO succeeded in maintaining the funding stream but could not delay fielding systems that addressed some of these deficiencies.

In 2001, under the direction of General Paul J. Kern, a study team from the SEI began a yearlong investigation to assess the long-term sustainability of FBCB2 in light of new emerging threats and missions. The SEI study team concluded that FBCB2 software was not on the appropriate architectural path for current operational use or for longer term use by the Objective Force. The study team recommended course corrections to address these deficiencies. One of the recommendations was to rearchitect the system to support a software product line approach.

The NG–PO–SEI team began a series of activities to address the recommendations, which included a redesigned software architecture, known as the Objective Architecture. It included

- isolation of application software from specific hardware
- development of a uniform, scalable interprocess communication mechanism known as the Node Data Broker
- design and implementation of support for backward compatibility
- isolation of the applications from the operating system through a façade
- isolation of the software from changes in message formats through message agents
- use of a display/map server to allow non-FBCB2 applications to display images on an FBCB2 map

Peter Blankenship reported that initial results have been generally positive. FBCB2 software can be moved more readily to a wide range of platforms. Moving core capabilities, such as communications processing, has been relatively straightforward. The Node Data Broker concept has facilitated creation of different configurations without undue burden. The use of message agents provides better isolation from changes over which the FBCB2 program has no control.

Blankenship also reported a number of technical and organizational challenges. Their unifying theme is how to maintain a balance between the cost- and schedule-driven nature of the program with a focus on individual systems and maintenance of the "pure" approach to implementing a system based on the Objective Architecture. The program must be vigilant in ensuring that it does not compromise the product line approach if it takes shortcuts to gain a specific product function. Also, the effective reuse of existing software assets requires considerable effort. The pros and cons of the decision to mine a particular legacy asset must be considered carefully.

In summary, the FBCB2 program is still transitioning to a product line approach. From a technical standpoint, the Objective Architecture appears to provide a viable means for achieving the desired flexibility, reliability, and stability for a family of products. From an organizational standpoint, project personnel are more universally informed and project practices are stabilizing around the product line approach. The FBCB2 product line offers substantial improvement and growth opportunities for the program and the Army.

2.3 Austin Info Systems and Software Product Lines—Tom Baker

Tom Baker of Austin Info Systems (AIS), Inc., presented an overview of the AIS strategy for software product lines. AIS is a small (100-plus people) company that provides analysis, design, development, and support of complex information and battle management systems [AIS 05]. The software in such systems spans command, control, and communications (C3), intelligence data fusion and tracking, fire control and targeting, and digital mapping and display. The company was driven to adopt a product line approach for much the same reason as CelsiusTech Systems [Bass 03]: it would go out of business if it kept doing business the old way.

AIS is adopting software product lines to exploit the intelligence business segment. The company intends to leverage its core capabilities in support of DoD, national, and international intelligence programs. The basis for the AIS product line is a service-oriented architecture for automated intelligence analysis known as the Joint Intelligence Toolkit. The product line will provide increasing levels of capability to programs such as DCGS (Distributed Common Ground Systems) and Future Combat Systems (FCS). In FCS, for example, the product line will provide support for situation understanding, intelligence data fusion, and sensor planning. The software runs on Windows platforms (with support for UNIX/Linux for targets such as FCS). Currently there are about 1.5 million lines of code in the unclassified code base and about 0.5 million in the classified code base.

The Joint Intelligence Toolkit has a layered architecture that includes an application services layer with command, analysis, and intelligence support. These services are engineered to be compatible with other DoD service-oriented architectures such as the System of Systems Common Operating Environment (SOSCOE). In fact, the application services are a product line within the overall product line.

AIS envisions a four-step transition path to software product lines:

1. Refactor existing software modules into subsystems and layers that are service-based architecture (SBA) compatible. (This was the focus of activities in 2004.)
2. Structure individual subsystems for participation in a service-based operating environment. (This is the 2005 focus.)
3. Extend individual subsystem interfaces for compliance with targeted SBA instances. (The first of these is SOSCOE; the second is the DCGS integration backbone.)
4. Create a production plan for targeted environments. (The first system delivery is in 2004, with six more planned for 2005.)

The goal is to provide early fielding of capability for FCS and DCGS. AIS has restructured the company to support these and other key programs. AIS has also been able to get customers to provide some of the first-year seed money for the product line effort.

2.4 The Common Avionics Architecture System (CAAS)—Stephanie Burns, Rockwell Collins

Rockwell Collins is a recognized leader in the development and fielding of open systems solutions, data links and self-organizing network communications, advanced display solutions, and precision geo-location and navigation. These key elements are integral to enabling the military's transformational objectives. Rockwell Collins plays a key role in developing new technology solutions such as the Common Avionics Architecture System (CAAS) [Rockwell 04].

The CAAS software architecture is being used to support the creation of a software product line of cockpit avionics software for special operations helicopters: the MH-47G Chinook, the MH-60M Black Hawk, and the MH/AH-6M Little Bird. The software product line will enable the introduction of unparalleled capabilities for situational awareness and connectivity into these cockpits and will help the special operations aircrews better manage cockpit workload in a uniform manner. Moreover, it will facilitate the integration of these new technologies with existing systems, allow for easier upgrades in the future, and simplify integration of equipment and software from third-party vendors. A Rockwell Collins Research and Development (R&D) project derived the software architecture from the Rockwell Collins Flight2 avionics architecture used in military and commercial aircraft such as the KC-135 and Boeing 767.

The vision for CAAS was to create a scalable system that would address obsolescence and modernization issues in multiple helicopter cockpits, and would reduce total cost of ownership by using a single, open, common avionics architecture system for all platforms. The drivers for adopting the CAAS approach were an open system architecture, variability isolation, connectivity, supportability, and modularity, all of which are essential to providing economical total cost of ownership.

Rockwell Collins is the software integrator for CAAS and the provider of multifunction displays, control display units, and general-purpose processing units shared by all three helicopters. Multiple isolated applications can run on a single processor, and the open system architecture simplifies connectivity and support. The scalable CAAS uses one Power PC 750 processor in each Control Display Unit (CDU) and two Power PC 750 processors in each Multifunction Display (MFD). The open system architecture is meant to simplify connectivity and support, including the ability to swap line-replaceable units between platforms in the field. The common avionics hardware and software in all three aircraft will reduce the logistics demands on aviation units and simplify training. Additionally, the commonality of software and hardware components is expected to provide the Special Operations Forces a lower total life-cycle cost and lower costs for technology insertion and supportability.

Some of the strengths of the software product line approach are that it supports insertion of new technology, enables multifunction displays and other avionic equipment units to be swapped out from one helicopter avionics system to another with automatic reconfiguration, and accommodates integration of subsystems by third-party developers through well-defined application program interfaces (APIs). The system complies with the Joint Technical Architecture-Army (JTA-A) requirements, using industry-standard, system-level interfaces: MIL-STD-1553 and ARINC 429 for legacy equipment, SMPTE 292 HDTV video standard for system video, and IEEE 802.3 for Ethernet connectivity.

CAAS embodies an open system architecture designed using a layered approach with widely accepted interfaces, separating hardware from software, and software services from system functional logic. The system functionality itself is partitioned into separate software components and isolated through published and controlled interface definitions. This approach allows hardware and software components to be replaced or upgraded independently and helps to lessen integration issues. The approach also allows problem fixes to be migrated to all systems. The layered architecture facilitates product development by application teams having specific domain expertise.

The product line scope, while bounded by the aviation platforms it will support, is now changing. More variability is being introduced because of

- plans to add more avionics platforms that will draw from the same logistics pool of hardware and software components
- an introduction of new system features traditionally used by service-oriented architectures
- the Army's desire to reuse software components across dissimilar platforms. One driver for this is that the conventional Army expects to upgrade its CH-47D helicopters to CH-47Fs, and some of them will be converted into MH-47Gs for special operations.

The architecture's impact on total life-cycle cost is that it will reduce non-recurring engineering costs to upgrade or add new functionality, and it will allow use of all available processing reserves when the system is upgraded, without requiring drastic rerouting of system data. A

Software Application Developer's Toolkit is available to third parties so that they can migrate and integrate their software applications on CAAS processing platforms or incorporate third-party equipment using dissimilar processing platforms.

2.5 RangeWare—Ed Dunn, NUWC

Ed Dunn of the Naval Undersea Warfare Center (NUWC) presented an update of product line activities within his organization. As a development lab component of the Naval Sea System Command (NAVSEA)–Division Newport, NUWC is the Navy's full-spectrum research, development, test and evaluation, engineering, and fleet support center for submarines, autonomous underwater systems, and offensive and defensive weapons systems associated with undersea warfare [NUWC 04].

Product lines are a reality within NUWC. Currently, the product line includes 10 projects. There has been local acceptance within the test ranges currently supported by Dunn's branch, but the product line approach is not yet institutionalized across NUWC. Organizationally, Dunn's branch does not have a complete infrastructure in place to support widespread sharing of processes or architectures across NUWC.

The software branch that Dunn represents deals with variations in testing systems that cover live test ranges, augmented with simulations. In other words, the organization builds test software to manage, execute, and report test exercises that are conducted with live, operational assets (e.g., submarines, torpedoes, and sonobuoys). But the test software must also deal with test exercises where simulation software is substituted for one or more of these assets. A current project is working on a widely expanded architecture, built on the RangeWare asset base described in previously referenced DoD workshop reports and the work of Cohen, Dunn, and Soule [Cohen 02].

Together with the SEI, NUWC has produced a series of case studies and a product line business case. As reported at the Sixth DoD Workshop [Bergey 04a], NUWC has launched a measurement program to address the need for better measurement standards to validate business case results. The NUWC staff developed measurement goals, identified and collected data to validate that goals are being met, and produced a plan to use results to show management the effectiveness of the product line approach.

The hope within NUWC is that documented product line results will overcome the difficulty of justifying strategic funding to support product line adoption. Another goal is to know when the product line approach is not appropriate for a new area or when a specific product falls outside of the scope of an existing product line. For example, while an experimental system may be built quite easily using product line assets, turning that experimental system into a fielded product may be outside the product line scope or exceed product line capabilities. Measuring the effectiveness of the product line for a specific product could be a useful risk management tool.

2.6 Argon Engineering Associates and Software Product Lines—Dave Drum, Argon

Argon Engineering Associates is a rapidly growing systems engineering and development company providing full-service information solutions to a wide range of customers. The Argon business vision is to grow by providing creative state-of-the-art technology solutions to difficult system problems. Argon's current challenges include the design and development of communication systems that search, identify, and capture signals. This work includes sensor development, data collection and decision support, and analysis and design of information retrieval and visualization techniques [Argon 04].

Argon uses a product line approach to develop and deploy many of its systems. Among the benefits it has seen from the product line approach are

- shorter development and delivery schedules
- lower development and upgrade costs
- lower total ownership costs
- support for an incremental development model
- shared development and technology costs
- production of best-in-class commercial off-the-shelf (COTS) and government off-the-shelf (GOTS) products
- continuous technology insertion

Argon's product line, which is called Lighthouse, is based on a true product line approach that enables the company to avoid developing unique point solutions by fielding a family of related systems. Use of a common architecture and common asset base enables Argon to meet its customer's cryptologic needs on a variety of platforms using current technology. These platforms include

- manned and unmanned airborne
- submarine and unmanned undersea vehicle (UUV)
- ship
- land and land mobile

In developing its product line, Argon had to define common capabilities from all its various customer requirements to achieve broadly capable software that could execute on general-purpose hardware. In effect, Argon has pulled its customers together (as a user group) based on needed capabilities because there is no desire to merge deliveries among programs and no means to make it happen. Customers typically have little motivation to align their schedules to benefit a product line approach, as they are only interested in their particular communication and intelligence (COMINT) systems. Modularity was stressed to permit adaptation of capability over time. Lighthouse relies heavily on COTS hardware and software; in some cases, the use of COTS/GOTS hardware and software is required.

Argon's production model is driven by a system requirements document and includes parallel paths for individual product developments through a common asset base to facilitate asset integration and synchronized build processes. Production involves a matrix of assets, asset groups, producers, enhancers and users. The assets themselves are developed, enhanced, or used by three product line groups: (1) signal processing, (2) real-time infrastructure, and (3) operability. While programs are the primary contributors to the development and enhancement of the asset base, independent research and development projects also contribute to the asset base. New missions are met primarily with new software that is then mined for assets that can be folded back into the product line.

Argon is currently delivering systems to the U.S, United Kingdom, and Australian military services to satisfy their cryptologic needs. The time to field a new system has typically been reduced from three or four years to one year.

Argon has addressed these product line challenges in the DoD environment :

- imposed size, weight, and cost constraints from hardware
- accommodating different funding and development cycles across customers and releases
- having a model for cost estimation and funding that supports various contract types and development and maintenance
- managing technology evolution for long duration programs (20-plus years) for major architecture changes, technology refresh, and technology insertion

2.7 Army Training Support Center (ATSC)—Frank Polster

Bringing the group up to date on last year's workshop presentation, Frank Polster reported that further significant organizational improvements were made following the use of the SEI Product Line Technical Probe℠ (PLTP℠), the subsequent planning sessions to address the PLTP findings, and the use of the SEI Architecture Tradeoff Analysis Method® (ATAM®). The ATSC staff members see product line technology as the way to embed training systems in Army combat systems. They are involved in the Army's FCS planning and see this as the likely mechanism to carry forward many of their product line ideas.

℠ Product Line Technical Probe and PLTP are service marks of Carnegie Mellon University.

® Architecture Tradeoff Analysis Method and ATAM are registered in the U.S. Patent and Trademark Office by Carnegie Mellon University.

2.8 Army Program Executive Office (PEO) Aviation—James Baxter

The Program Executive Office (PEO) Aviation is the Army manager for combat, cargo, and utility helicopters. It is also responsible for unmanned aerial vehicles (UAVs) and other Army aviation programs. The program executive officer reports directly to the Army acquisition executive. "PEO Aviation is the responsible management official who provides overall direction and guidance for the development, acquisition, testing, systems integration, product improvement and fielding of assigned programs" [PEO Aviation 04].

PEO Aviation sees three primary goals in its software improvement programs:

1. Achieve better integration within platform architectures, eliminating major rewrites for each upgrade.
2. Identify and exploit commonality.
3. Support integration and interoperability across Army aviation platforms.

The PEO has identified several approaches to address these goals through adoption of a product line approach:

- Eliminate the need for multiple versions of software. Achievement of this goal would eliminate the current platform-centric development, which looks at each aircraft as a separate build. Instead, a product line approach would involve moving to an architecture-centric approach that supports software reuse across platforms.
- Eliminate closed/proprietary architectures via migration to mission-centric approaches. Currently, each manufacturer/developer has its own architecture. All upgrades and modifications are tied to that single organization. Two plans are underway that would create non-proprietary solutions:
    1. Common Aviation Architecture System (CAAS) for special operations, cargo, and utility
    2. Manned/Unmanned Common Architecture Program (MCAP) for Apache helicopters

A further step would be to merge these two architecture solutions.

Army Aviation is also participating in the new Army FCS program. This program will provide new UAVs, but more importantly will introduce the concept of network-centric warfare to support interoperability. A System of Systems Common Operating Environment (SOSCOE) is envisioned for FCS to support all future platforms. From a product line perspective, there are questions about the suitability and variability of SOSCOE to support real-time aviation needs, as opposed to more conventional command and control.

2.9 The Common Gunnery Architecture—Dean Runzel, Army PEO for Simulation, Training, and Instrumentation

Dean Runzel of the U.S. Army Program Executive Office for Simulation, Training, and Instrumentation (PEO STRI) presented an overview of an effort to standardize gunnery training systems. The Army currently has eight different trainers built by three different contractors for the Abrams, Bradley, and Stryker platforms. There is no interoperability among the three different contractor baselines; even trainee records cannot be moved across the systems, because the databases are different. The envisioned solution to the interoperability problem—and to the related stovepiped life-cycle support problems—is the common gunnery architecture (CGA).

The CGA is an extension of the One Semi-Automated Forces (OneSAF) Objective System (OOS) product line architecture framework. The CGA provides a common baseline for all gunnery trainers; it incorporates requirements from existing systems in a core set of requirements. The common baseline will include universal student records, universal terrain databases, universal dynamic terrain effects, and universal entity behaviors. Among other things, the CGA is intended to permit training in contemporary operating environments and urban environments, minimize the cost of concurrency and obsolescence, and bridge the gap with the FCS. (Coordination between PEO STRI and FCS is needed for the training portion of FCS so that FCS does not develop its own solution outside of the CGA.) The CGA will also leverage the current investment in the Stryker Mobile Gun System (MGS) training system.

The government owns the CGA. Contracts will be awarded to develop specific components (e.g., a crew station interface), while other components will be provided to contractors as government-furnished information (e.g., the after-action review portion and the training management system). Because of the commonality, there is a potential to extend the CGA to every training device produced by PEO STRI.

The anticipated benefits of the CGA include

- cost savings in software maintenance and flexibility in responding to changes in tactics, techniques, and procedures
- cost avoidance because of no redundant development
- better products with added functionality

A 10-year cost–benefit analysis of the CGA was conducted, with the following assumptions:

- All eight training devices migrate to the CGA in fiscal year 2007–2008.
- Migration occurs in conjunction with each system's scheduled technical refresh efforts.
- A single integrated development environment and support organization is established.

The analysis showed a potential $66 million in cost avoidance.

3 Summary

The SEI's Seventh DoD Product Line Practice Workshop explored the product line practices of organizations in the DoD community in light of best commercial practices and government experience. This workshop demonstrated a continuance of the trend revealed during the sixth workshop: namely, software product line practice is becoming a reality in the DoD. Almost all the presentations were based on experience rather than plans or speculation. The discussions validated the pivotal pieces of the framework and provided valuable feedback on the companion. Challenges and experience-based solutions were discussed.

As in previous workshops, the empirical and anecdotal evidence that the workshop participants brought to the discussion significantly enhanced mutual understanding of the practices and issues as they apply to the DoD. Traditional DoD acquisition strategies are not naturally conducive to software product lines. However, it is clear that product line practice is possible within the DoD, and more DoD organizations are taking a product line approach.

In an effort to expand both the information base and the DoD community interested in software product lines, the SEI was encouraged by the participants to continue to hold similar workshops.

As before, results of this workshop will be incorporated into the acquisition companion to the framework, which will continue to be refined and revised as the technology matures and as the SEI continues to receive feedback [Bergey 04b]. If you have any comments on this report or are using a product line approach in the development or acquisition of software-intensive systems for the DoD and would like to participate in a future workshop, please send email to Linda Northrop at lmn@sei.cmu.edu.

References

URLs are valid as of the publication date of this document.

[AIS 05] Austin Info Systems, Inc. http://www.ausinfo.com (2005).

[Argon 04] Argon Engineering Associates. *Introduction.* http://www.argoneng.com/intro.html (2004).

[Bass 97] Bass, L.; Clements, P.; Cohen, S.; Northrop, L.; & Withey, J. *Product Line Practice Workshop Report* (CMU/SEI-97-TR-003, ADA327610). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 1997. http://www.sei.cmu.edu/publications/documents/97.reports/97tr003/97tr003abstract.html.

[Bass 98] Bass, L.; Chastek, G.; Clements, P.; Northrop, L.; Smith, D.; & Withey, J. *Second Product Line Practice Workshop Report* (CMU/SEI-98-TR-015, ADA354691). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 1998. http://www.sei.cmu.edu/publications/documents/98.reports/98tr015/98tr015abstract.html.

[Bass 99] Bass, L.; Campbell, G.; Clements, P.; Northrop, L.; & Smith, D. *Third Product Line Practice Workshop Report* (CMU/SEI-99-TR-003, ADA361391). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 1999. http://www.sei.cmu.edu/publications/documents/99.reports/99tr003/99tr003abstract.html.

[Bass 00] Bass, L.; Clements, P.; Donohoe, P.; McGregor, J.; & Northrop, L. *Fourth Product Line Practice Workshop Report* (CMU/SEI-2000-TR-002, ADA375843). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2000. http://www.sei.cmu.edu/publications/documents/00.reports/00tr002.html.

[Bass 03] Bass, L.; Clements, P.; & Kazman, R. *Software Architecture in Practice, second ed.* Reading, MA: Addison-Wesley, 2003.

[Bergey 98] Bergey, J.; Clements, P.; Cohen, S.; Donohoe, P.; Jones, L.; Krut, R.; Northrop, L.; Tilley, S.; Smith, D.; & Withey, J. *DoD Product Line Practice Workshop Report* (CMU/SEI-98-TR-007, ADA346252). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 1998. http://www.sei.cmu.edu/publications/documents/98.reports/98tr007/98tr007abstract.html>.

[Bergey 99] Bergey, J.; Campbell, G; Clements, P.; Cohen, S.; Jones, L.; Krut, R.; Northrop, L.; & Smith, D. *Second DoD Product Line Practice Workshop Report* (CMU/SEI-99-TR-015, ADA375845). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 1999. http://www.sei.cmu.edu/publications/documents/99.reports/99tr015/99tr015abstract.html.

[Bergey 00a] Bergey, J.; Campbell, G; Cohen, S.; Fisher, M.; Jones, L.; Krut, R.; Northrop, L.; O'Brien, W.; Smith, D.; & Soule, A. *Third DoD Product Line Practice Workshop Report* (CMU/SEI-2000-TR-024, ADA387259). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2000. http://www.sei.cmu.edu/publications/documents/00.reports/00tr024.html.

[Bergey 00b] Bergey, J.; Fisher, M.; Gallagher, B.; Jones, L.; & Northrop, L. *Basic Concepts of Product Line Practice for the DoD* (CMU/SEI-2000-TN-001, ADA375859). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2000. http://www.sei.cmu.edu/publications/documents/00.reports/00tn001.html.

[Bergey 01] Bergey, J. et al. *Fourth DoD Product Line Practice Workshop Report* (CMU/SEI-2001-TR-017, ADA399205). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2001. http://www.sei.cmu.edu/publications/documents/01.reports/01tr017.html.

[Bergey 03] Bergey, J. et al. *Fifth DoD Product Line Practice Workshop Report* (CMU/SEI-2003-TR-007, ADA416733). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2003. http://www.sei.cmu.edu/publications/documents/03.reports/03tr007.html.

[Bergey 04a] Bergey, J.; Cohen, S.; Jones, L.; & Smith, D. *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, 2004. http://www.sei.cmu.edu/publications/documents/04.reports /04tn011.html.

[Bergey 04b] Bergey, J.; Campbell, G; Cohen, S.; Fisher, M.; Gallagher, B.; Jones, L.; Northrop, L.; & Soule, A. *Software Product Line Acquisition: A Companion to A Framework for Software Product Line Practice, Version 3.0.* http://www.sei.cmu.edu/productlines /companion.html (2004).

[Brownsword 96] Brownsword, L. & Clements, P. *A Case Study in Successful Product Line Development* (CMU/SEI-96-TR-016, ADA315802). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 1996. http://www.sei.cmu.edu/publications/documents/96.reports /96.tr.016.html.

[Chastek 02] Chastek, G, ed. *Proceedings of the Second Software Product Lines Conference (SPLC2)* (Lecture Notes in Computer Science [LNCS] volume 2379). San Diego, CA, Aug. 19-22, 2002. New York, NY: Springer-Verlag, 2002.

[Clements 01] Clements, P. et al. *Fifth Product Line Practice Workshop Report* (CMU/SEI-2001-TR-027, ADA272355). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2001. http://www.sei.cmu.edu/publications/documents/01.reports /01tr027.html.

[Clements 02] Clements, P. & Northrop, L. *Software Product Lines: Practices and Patterns*. Reading, MA: Addison-Wesley, 2002.

[Clements 04] Clements, P. & Northrop, L. *A Framework for Software Product Line Practice, Version 4.2.* http://www.sei.cmu.edu/productlines/framework.html (2004).

[Cohen 02] Cohen, S.; Dunn, E.; & Soule, A. *Successful Product Line Development and Sustainment: A DoD Case Study* (CMU/SEI-2002-TN-018, ADA407788). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2002. http://www.sei.cmu.edu/publications/documents/02.reports /02tn018.html.

[Cohen 04] Cohen, S.; Zubrow, D.; & Dunn, E. *Case Study: A Measurement Program for Product Lines* (CMU/SEI-2004-TN-023). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2004. http://www.sei.cmu.edu/publications/documents/04.reports/04tn023.html.

[Donohoe 00] Donohoe, P., ed. *Software Product Lines: Experience and Research Directions, Proceedings of the First Software Product Line Conference (SPLC1).* Denver, CO, Aug. 28-31, 2000. Boston, MA: Kluwer Academic Publishers, 2000.

[Nord 04] Nord, R., ed. *Proceedings of the Third International Software Product Lines Conference (SPLC 2004)* (Lecture Notes in Computer Science [LNCS] volume 3154. Boston, MA, Aug. 3-Sept. 2, 2004. New York, NY: Springer-Verlag, 2004.

[NUWC 04] Naval Undersea Warfare Center. http://www.nuwc.navy.mil/ (2004).

[PEO Aviation 04] Program Executive Office, Aviation. https://www.peoavn.redstone.army.mil/ (2004).

[Rockwell 04] Rockwell Collins, Inc. http://www.rockwellcollins.com/news/background/page3072.html (2004).

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (Leave Blank)	2. REPORT DATE March 2005	3. REPORT TYPE AND DATES COVERED Final
4. TITLE AND SUBTITLE Software Product Lines: Experiences from the Seventh DoD Software Product Line Workshop		5. FUNDING NUMBERS F19628-00-C-0003
6. AUTHOR(S) John K. Bergey, Sholom Cohen, Patrick Donohoe, Lawrence G. Jones		
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Software Engineering Institute Carnegie Mellon University Pittsburgh, PA 15213		8. PERFORMING ORGANIZATION REPORT NUMBER CMU/SEI-2005-TR-001
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) HQ ESC/XPK 5 Eglin Street Hanscom AFB, MA 01731-2116		10. SPONSORING/MONITORING AGENCY REPORT NUMBER ESC-TR-2005-001
11. SUPPLEMENTARY NOTES		
12A DISTRIBUTION/AVAILABILITY STATEMENT Unclassified/Unlimited, DTIC, NTIS		12B DISTRIBUTION CODE 31

13. ABSTRACT (MAXIMUM 200 WORDS)

The Carnegie Mellon® Software Engineering Institute held the Seventh Department of Defense (DoD) Product Line Practice Workshop in September 2004. The workshop was a hands-on meeting to share DoD product line practices, experiences, and issues and to discuss ways in which specific product line practices are accomplished within the DoD. Participants reported encouraging progress on DoD software product lines. This report synthesizes the workshop presentations and discussions.

14. SUBJECT TERMS DoD product line practice, Framework for Software Product Line Practice, software product line, software product line acquisition			15. NUMBER OF PAGES 31
16. PRICE CODE			
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL

NSN 7540-01-280-5500 Standard Form 298 (Rev. 2-89) Prescribed by ANSI Std. Z39-18 298-102