2010-04-30

Comparing Acquisition Strategies: Open Architecture versus Product Lines

Guertin, Nicholas; Clements, Paul

Monterey, California. Naval Postgraduate School

https://hdl.handle.net/10945/33466

NPS-AM-10-033

EXCERPT FROM THE
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

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

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.

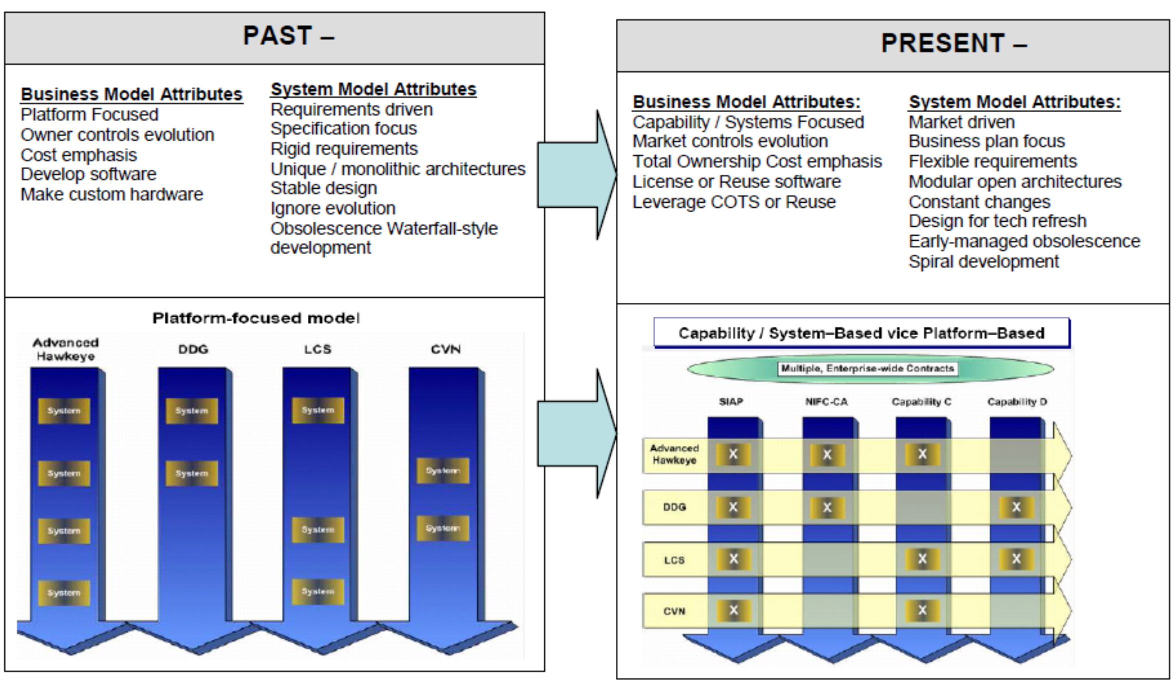

Figure 1. Traditional versus Open Architecture Development Approaches

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.**)

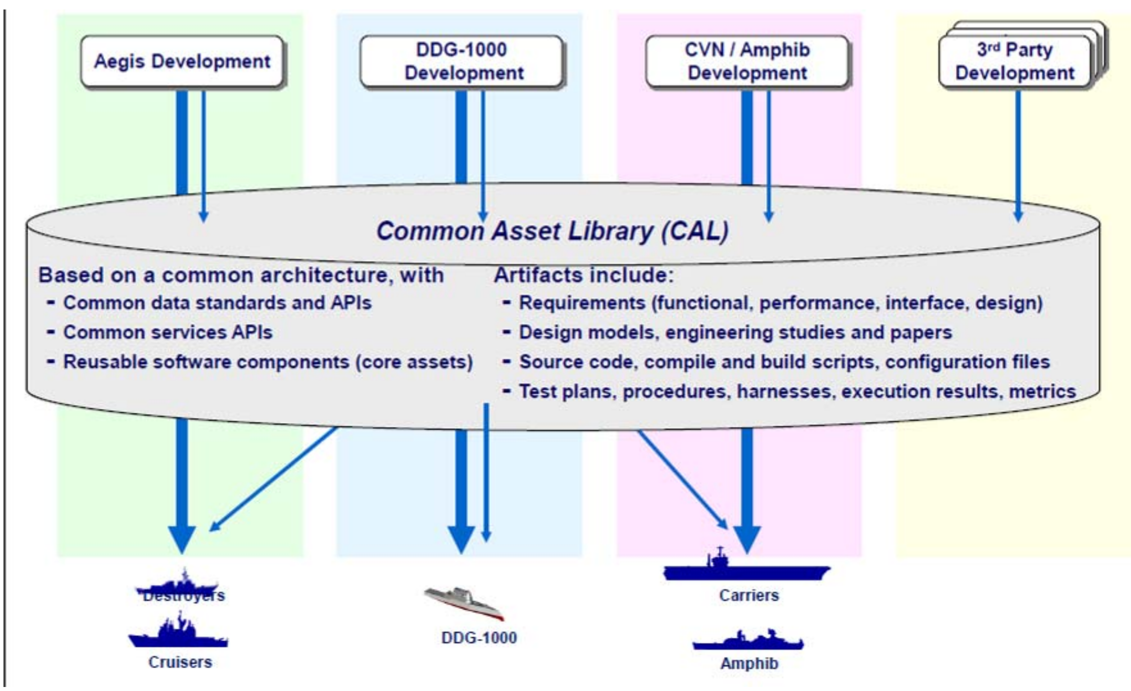

Figure 3. PEO IWS Product Line Approach for Surface Combat Systems

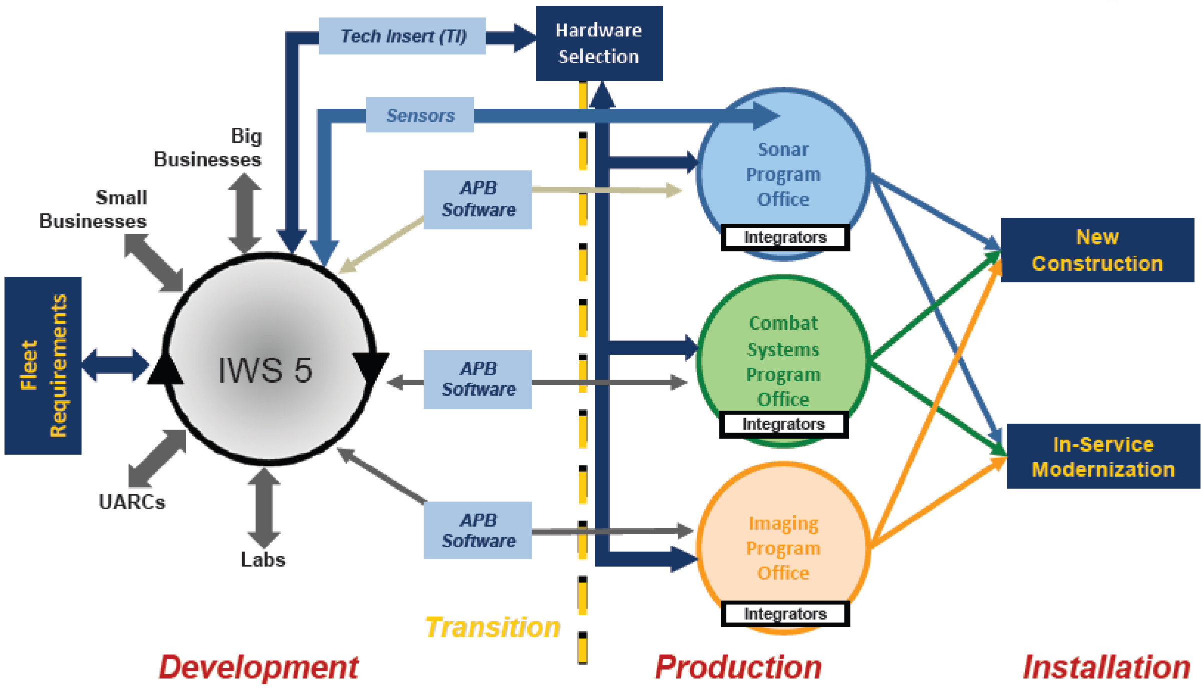

Figure 4. PEO Submarines SWFTS Model for Cross-platform Product Commonality

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):

2. 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.

THIS PAGE INTENTIONALLY LEFT BLANK

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

THIS PAGE INTENTIONALLY LEFT BLANK

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

www.acquisitionresearch.org