

# The Chief Software Architect in U.S. Army Acquisition

**Stephen Blanchette, Jr. and John Bergey**  
Software Engineering Institute

**Abstract:** The U.S. Army is aggressively pursuing software architecture practices as a means of reducing risk in its acquisition programs. Central to this strategy is creating an appropriately skilled workforce capable of overseeing software development activities in its innovative programs. The latest development in the Army's long-standing pursuit of improving the software talents of its acquisition workforce is the establishment of Chief Software Architects in its program executive offices. This article discusses this latest demonstration of the Army's commitment to adopting an architecture-centric acquisition approach and its focus on developing the software architecture skills of its acquisition workforce.

In May of 2009, Lieutenant General Ross Thompson, then the Military Deputy to the Assistant Secretary of the Army for acquisition, logistics and technology (ASA[ALT]), issued a memorandum directing each Program Executive Office (PEO) to designate a Chief Software Architect (CSWA). The directive was another step in the Army's aggressive efforts to instill architecture-centric practices across its acquisition programs. Since late 2002, the ASA(ALT) has been working with the Carnegie Mellon® Software Engineering Institute (SEI)—a federally funded research and development center—in a strategic partnership known as the Army Strategic Software Improvement Program (ASSIP). The aim of this partnership is to improve the Army's ability to acquire software-reliant systems (Figure 1)—i.e., systems whose behavior (e.g., functionality, performance, safety, security, interoperability, and so forth) is highly dependent on software in some significant way. Through this partnership, the Army is enhancing its ability to be a “smart buyer” of software-reliant systems.



*Figure 1: A typical software-reliant system: the M1 Abrams tank relies on software for navigation, targeting, precision fires, and more.<sup>1</sup>*

Early ASSIP investigations into Army acquisition programs indicated, among other things, that while software-architecture practices were deemed important for software-reliant systems programs, the methods and skills to carry out those practices were perceived to be inadequate. Hence, the ASSIP formulated an initiative to raise the organic capabilities of the Army acquisition workforce in the area of architecture-centric software development. This article discusses the Army's software architecture initiative and examines the human factor behind the technology: the Chief Software Architect.

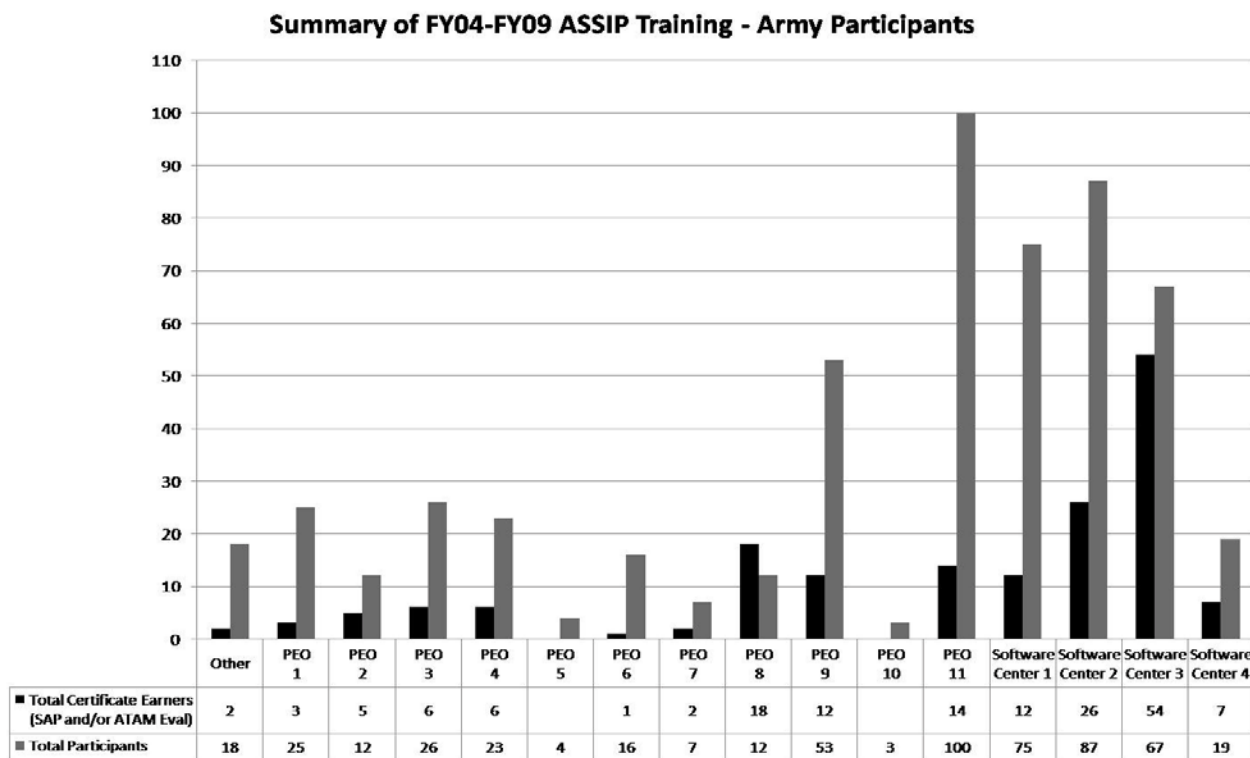
## The Importance of Software Architecture

When viewed in terms of program impact, the reason for focusing on software architecture becomes obvious. Experience confirms that the quality and longevity of a software-reliant system is largely determined by its architecture. The software architecture underpins a system's software design and code; it represents the earliest design decisions, ones that are difficult and costly to change later [1]. Further, the software architecture supports, or impedes, the desired system qualities that are manifest in the software, so getting the architecture “right” has enormous implications both for the software and for the parent system that is reliant upon that software to deliver any part of its functionality. The right software architecture will facilitate user acceptance of a system; the wrong one will do quite the opposite. As confirmed by a number of studies in the last decade [2, 3, 4, 5], sound software architectural practices are essential to successful software-reliant systems programs.

However, history has shown that the linkage between software architecture practices and successful acquisition of software-reliant systems has not been sufficient motivation to incorporate such practices in acquisition programs. According to a 2009 NASA study on flight software complexity, “Good software architecture is the most important defense against incidental complexity in software designs, but good architecting skills are not common” [6]. Indeed, reports repeatedly cite poor architectural practices and a general lack of understanding of the need for software architecture as a source of acquisition program difficulties [7, 8, 9, 10, 11].

Thus, while an architecture-centric development approach is an acknowledged good practice in software-reliant systems programs, it is rarely executed effectively or rigorously.

Figure 2: Summary of ASSIP Architecture Training – Army Participants



### The ASSIP Software Architecture Initiative

Recognizing that software architecture is still one of the key technical challenge areas facing its Project Management Offices (PMOs), the Army devoted a significant part of its ASSIP resources to address the problem by creating a software architecture initiative. Initially, a training component formed the core of the initiative.

The SEI already had available a formal training curriculum for software architecture,<sup>2</sup> and the ASSIP elected to use it as the basis of the software architecture initiative's training element. The curriculum consists of six courses:

- >> **Software Architecture: Principles and Practices**
- >> **Documenting Software Architectures**
- >> **Software Architecture Design and Analysis**
- >> **Software Product Lines**
- >> **SEI Architecture Tradeoff Analysis Method® (ATAM®) Evaluator Training**
- >> **ATAM Leader Training**

The SEI delivered the curriculum at the Army Software Engineering Centers (SECs) using the same materials and instructors as in its publicly offered classes. The SECs provided the most central location for many participants since most of the Army's PMOs are located in close proximity to one of the SECs. Students who completed the prescribed course sequences earned certificates just as if they had attended the regular public offerings.<sup>3</sup>

The training program enjoyed strong participation, a good indication of both need and interest within the Army acquisition community. In fact, demand exceeded expectations and forced the waving of class size restrictions in a few instances. Additionally, participation was broad, with representation from all 11 PEOs<sup>4</sup> and all of the Army's software centers. Well over 500 Army technical professionals have attended at least part of the curriculum, with more than 25% having earned at least one certificate. Figure 2 summarizes these results.<sup>5,6</sup>

In addition to training practitioners, the ASSIP builds awareness at higher levels: A rotating list of Army senior leaders, personally invited by the MILDEP, gain exposure to software architecture and other important software engineering concepts three times a year during the ASSIP senior leader education program.

Beyond training, the ASSIP software architecture initiative grew to include a skill-building component. The initiative sponsored several ATAM-based software architecture evaluations, with the proviso that trained Army evaluators would participate as evaluation team members. (Projects that had not yet developed a software architecture conducted Quality Attribute Workshops, or QAWs, usually as a precursor to an ATAM evaluation.) Table 1 shows the projects that have participated to date. The evaluations allowed trained Army personnel to practice their skills and also introduced architecture-centric practices across a variety of Army projects.

Table 1: Projects Employing Architecture-Centric Practices

Army Project (in alphabetical order)	ATAM	QAW
Aerial Common Sensor	✓	✓
Army Battle Command System		✓
Command Post of the Future	✓	
Common Avionics Architecture System	✓	
Distributed Common Ground Station – Army	✓	✓
Force XXI Command Brigade-and-Below	✓	
Future Combat Systems	✓	✓
Integrated Fired Control	✓	✓
Joint Tactical Common Operational Picture Workstation	✓	
Manned/Unmanned Common Architecture Program	✓	
Network Operations Data Product Development Environment		✓
One Semi-Automated Forces	✓	
Sequoyah		✓
Warfighter Information Network – Tactical	✓	

According to a recent study, these architecture-centric practices have had a positive impact [12]. As shown in Figure 3, most projects reported significant improvement in their architecturally significant artifacts (including system quality attributes, software architectures themselves, and architecture-related risks). The architecture teams achieved an understanding of stakeholder expectations and the implications of architectural decisions on user needs [12]. Additionally, almost all projects experienced very substantial or significant improvement in stakeholder communication (see Figure 4). Stakeholders, collectively, achieved a common understanding of the systems under development, which increased the likelihood that those systems would address expectations and user needs (and, consequently, improved the chances for program success) [12].

### The Role of the Army’s CSWAs

Having trained a cadre of acquisition professionals capable of implementing architecture-centric practices, the next step for the Army was to begin the institutionalization of software architecture practices throughout its acquisition offices. LTG Thompson decided that the best way to accomplish that goal was to establish Chief Software Architects in the program executive offices. Each PEO has oversight responsibility for a domain of related projects and products:<sup>7</sup>

- >> PEO Ammunition (Ammo)
- >> PEO Aviation (AVN)
- >> Joint PEO Chemical and Biological Defense (CBD)
- >> PEO Combat Support and Combat Service Support (CS&CSS)
- >> PEO Command Control and Communications – Tactical (C3T)
- >> PEO Enterprise Information Systems (EIS)
- >> PEO Ground Combat Systems (GCS)
- >> PEO Integration
- >> PEO Intelligence, Electronic Warfare and Sensors (IEW&S)
- >> PEO Missile and Space (MS)
- >> PEO Simulation, Training, and Instrumentation (STRI)
- >> PEO Soldier

Each CSWA is responsible for providing guidance for software issues across a PEO’s portfolio of programs. The scope of responsibility is broad; the CSWAs are accountable for oversight and management of all software being developed or acquired within their respective PEOs. Consequently, the position requires strong software competence and pertinent training. Particularly notable in the CSWA directive is the specific requirement for training. The intent is that the position is not just another task in someone’s job jar; the CSWAs are expected to possess or obtain skills relevant to the position. Each CSWA must complete training equivalent to the SEI course series for Software Architecture Professionals. A subset of the architecture curriculum, the Software Architecture Professional series consists of a foundational course in software architecture principles and practices (including a compulsory competency examination), as well as in-depth courses covering essential concepts for effectively designing and analyzing software architectures, effective documentation methods, and an introduction to software product line concepts. These are advanced topics; the coursework assumes attendees already are practicing software professionals with responsibility for designing, developing, or managing the construction of software-reliant systems.

Figure 3: Architecture-Centric Practices Improve Artifacts

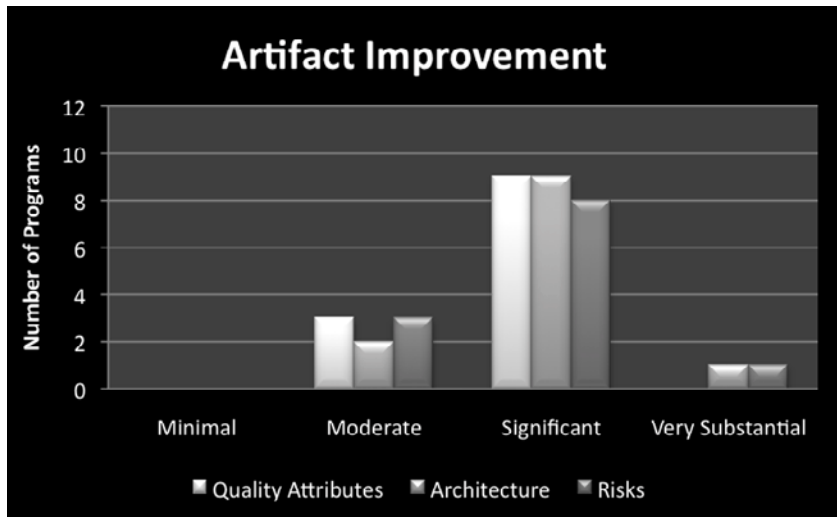
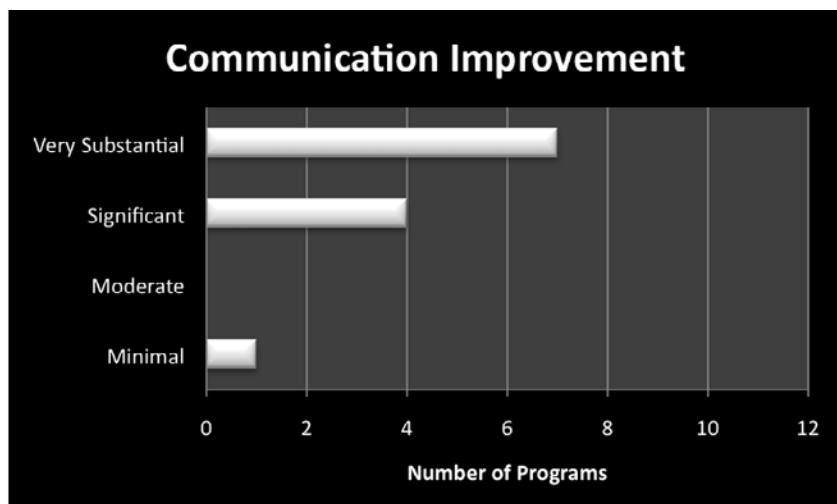


Figure 4: Architecture-Centric Practices Improve Communication



In August 2009, the CSWAs met together for the first time during an ASSIP Action Group meeting. There, they fleshed out their collective responsibilities in more detail. They identified their primary task as providing support to project managers (PMs) with their software processes, including monitoring software architecture development from initial design decisions throughout the acquisition life cycle in order to identify and mitigate software risks, linking architectural components to mission drivers, and focusing on stakeholder requirements. The CSWAs will help ensure every PM has an appropriately documented software architecture and will help to evaluate how well individual systems meet the appropriate quality attributes. Beyond the architecture, the CSWAs will assess and evaluate software cost estimates in a system life

cycle context for portfolio programs as well as review and endorse system engineering plans with their respective Chief System Engineers to ensure those plans leverage appropriate standards<sup>8</sup> and appropriate architecture-centric practices.

A second task for the CSWAs is to establish the necessary infrastructures within their PEOs to support software objectives, including issuing guidance to the PMs on software architecture requirements, identifying and enforcing any PEO-specific system quality attributes that will be implemented in software, and providing guidance for software architecture design and reviews to ensure consistent implementation of best practices.

The CSWAs' third task is to decide the best ways to leverage software architecture to mitigate program risks, especially with regard to analysis in response to integration and interoperability challenges. In particular, they will ensure development of software architectures in a system of systems context to address the interoperability requirements that are becoming more common across all Army systems.

Lastly, the CSWAs will participate in the ASSIP and other Army-wide communities of interest to exploit opportunities for commonality across the PEO portfolios.

### Way Ahead

The Office of the DoD Chief Information Officer issued a white paper [13] on a competency framework for the DoD Architect that noted three root causes for shortcomings in architecture practices across the DoD:

- >> **Inability to leverage the benefits of an architecture due to inadequate training on the part of stakeholders or inadequate communication on the part of architects**
- >> **Lack of incentives to encourage the professional growth of architects in the DoD**
- >> **Lack of visibility into the existence or value of architecture training**

All the services have made some strides with respect to system-level architecture (the Navy's Open Architecture initiative, for example, instituted relevant policy supported by a model and a corresponding tool [14]). However, through the ASSIP and the CSWAs, the Army has leapt ahead with a comprehensive strategy for software architecture that addresses not just technical issues but also these non-technical aspects, which are essential to institutionalization and achieving maximum benefit from software architecture practices. The goal now is to help ensure that the new Army CSWAs are positioned for success. To that end, the FY10 ASSIP plan focuses on supporting them with continued training and awareness opportunities as well as technical assistance.

In working with the CSWAs to develop execution plans, one non-technical theme recurs: How can a CSWA direct and influence within organizational constraints? Since the CSWAs exercise no direct authority over the projects within their respective PEO portfolios, the question is a crucial one. As a solution, most CSWAs are taking a relationship-building approach, teaming with PMO software architects and engineers to work on problems collaboratively. In so doing, they will be able to leverage early adopters of software architecture practices to achieve initial successes and build publicity within their organizations. In addition, some CSWAs are seeking formal endorsement from their PEOs or Chief System Engineers as a means of putting more weight behind their objectives.

From a technical perspective, feedback from the CSWAs indicated some challenges. One challenge is using software architecture to help understand, validate, and improve software cost estimation. Intuitively, a better understanding of a software architecture should lead to a better understanding of the software to be built, which in turn should lead to a better estimate of software cost. However, CSWAs need tools and methods to formalize the relationship between architecture and cost estimation. Another challenge is developing a standard means of determining appropriate technology readiness levels (TRLs) for software, and determining which phases of the acquisition lifecycle require which software TRLs.

Overall, the positioning of the CSWAs at the PEO level is advantageous in that it enables them to take a portfolio perspective on such important issues, as well as on architecture sub-specialties such as data architecture and security architecture, instead of developing solutions project by project. Data and security architectures, particularly, are vital for implementing robust and reliable networked solutions for the warfighter, and such solutions are becoming increasingly commonplace. Further, and perhaps more importantly, the CSWAs are able to collaborate with each other through the ASSIP forum to address these software architecture matters at the system of systems level, which will facilitate the development of truly interoperable capabilities for a modernized Army and for joint and coalition forces.

### Summary

The creation of a Chief Software Architect role in each PEO has been a significant step in the Army's efforts to institutionalize architecture-centric practices in its software-reliant system acquisition programs. Through the ASSIP, the Army has focused on developing the software architecture skills of its acquisition workforce and building awareness of architecture-centric practices among its leadership. The CSWAs can leverage the cadre of software architecture professionals and qualified ATAM evaluators to realize the benefits of architecture-centric practices across the Army's acquisition projects and set the standard for improvement across the DoD.

The ASSIP continues to support the CSWAs as they work to establish and champion architecture-centric practices within their PEOs. ♦

## ABOUT THE AUTHORS



**Stephen Blanchette, Jr.** is Chief Engineer for Army Programs at SEI. He has more than 23 years experience in the defense industry as a software engineer and manager. He is an associate fellow of the American Institute of Aeronautics and Astronautics and a senior member of the Institute of Electrical and Electronics Engineers. Mr. Blanchette earned a BS in Computer Science from Embry-Riddle Aeronautical University and an MA in Diplomacy from Norwich University.

E-mail: [sblanche@sei.cmu.edu](mailto:sblanche@sei.cmu.edu)



**John Bergey** is a senior member of the technical staff at SEI, specializing in transitioning SEI product line and architecture-centric practices (e.g., Quality Attribute Workshop, Architecture Tradeoff Analysis Method®) into acquisition practice across the Armed Services. Prior to joining SEI, he served over 25 years as a software division manager with the U.S. Navy. Mr. Bergey is a graduate of Pennsylvania State University and the University of Michigan.

E-mail: [jkb@sei.cmu.edu](mailto:jkb@sei.cmu.edu)

**Software Engineering Institute**  
**4500 Fifth Avenue**  
**Pittsburgh, PA 15213**



## REFERENCES

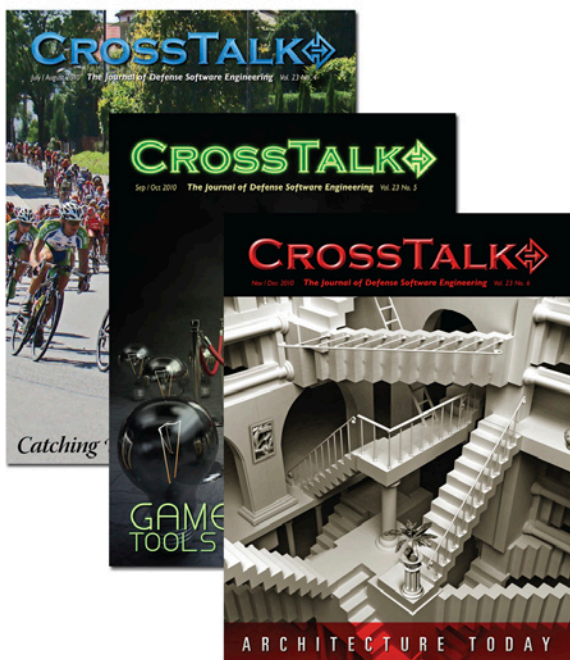
1. Bass, Len, Paul Clements, and Rick Kazman. *Software Architecture in Practice*, (2nd ed.). Boston: Addison-Wesley, 2003.
2. Defense Science Board Task Force. "Acquiring Defense Software Commercially." June 1994. 25 December 2009 <<http://www.acq.osd.mil/dsb/reports/commrcialdefensesoftware.pdf>>.
3. ---. "Defense Software." November 2000. 25 December 2009 <<http://www.acq.osd.mil/dsb/reports/defensesoftware.pdf>>.
4. Anderson, William, et al. *Army workshop on lessons learned from software upgrade programs (CMU/SEI-2001-SR-021)*. Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2001.
5. Government Accountability Office. *Defense Business Transformation, Sustaining Progress Requires Continuity of Leadership and an Integrated Approach (GAO-08-462T)*. Washington: Government Accountability Office, 2008.
6. Dvorak, Daniel L. "NASA Study on Flight Software Complexity." 2009.
7. Charette, Robert, John McGarry and Kristen Baldwin. *Tri-service Assessment Initiative Phase 2 Systemic Analysis Results*. January 2003.
8. Bass, Len, Robert Nord and William G Wood. "Risk Themes from ATAM Data: Preliminary Results." April 2006. 25 December 2005 <<http://www.sei.cmu.edu/library/abstracts/presentations/basssaturn.cfm>>.
9. Kasunic, Mark. *Army Strategic Software Improvement Program (ASSIP) Survey of Army Acquisition Managers (CMU/SEI-2004-TR-003)*. Pittsburgh, PA: Software Engineering Institute/Carnegie Mellon University, 2004.
10. Keeler, Kristi L. *U.S. Army Acquisition - The Program Office Perspective (CMU/SEI-2005-SR-014)*. Pittsburgh, PA: Software Engineering Institute/Carnegie Mellon University, 2005.
11. Blanchette, Jr., Stephen. *U.S. Army Acquisition - the Program Executive Officer Perspective (CMU/SEI-2005-SR-002)*. Pittsburgh, PA: Software Engineering Institute/Carnegie Mellon University, 2005.
12. Nord, Robert L., John Bergey, Stephen Blanchette, Jr., and Mark Klein. *Impact of Army Architecture Evaluations (CMU/SEI-2009-SR-007)*. Pittsburgh, PA: Software Engineering Institute/Carnegie Mellon University, 2009.

## REFERENCES (continued)

13. Office of the DoD CIO. *White Paper Phase I: A Competency Framework for the DoD Architect*. Washington: Department of Defense, 2008.
14. Shannon, Jim. "Naval Enterprise Open Architecture: What Program Managers Need to Know." January 2006.

## NOTES

1. DoD photo By Staff Sgt. Aaron Allmon, U.S. Air Force. (Released). Retrieved 12/23/2009 from <<http://www.defense.gov/dodcmsshare/newsphoto/2006-02/060203-F-7823A-008.jpg>>.
2. Interested readers will find additional information about the SEI software architecture curriculum on the SEI website: <<http://www.sei.cmu.edu/training/find/architecture.cfm>>.
3. Three certificates, Software Architecture Professional, ATAM Evaluator, and ATAM Lead Evaluator, are available to students who complete the required courses. Beginning in 2009, individuals seeking one of these certificates were required to pass a validation exam in addition to completing the coursework.
4. Data for PEO Missiles and Space include its predecessor organizations PEO Tactical Missiles and PEO Air Space and Missile Defense; PEO Integration, created in mid-2009, is not represented in the data.
5. PEOs and software centers are shown in random order.
6. In addition to Army personnel, 62 representatives from other services and support contractors have been trained through the conclusion of FY09.
7. In addition to the Army PEOs noted in the list, the Joint PEO for the Joint Tactical Radio System (JTRS), currently transitioning from the Navy to the Army, also will participate.
8. Examples of such standards are International Standards Organization/International Electrotechnical Commission (ISO/IEC) standards 15288 for system engineering and 12207 for software development.



## CALL FOR ARTICLES

If your experience or research has produced information that could be useful to others, **CROSSTALK** can get the word out. We are specifically looking for articles on software-related topics to supplement upcoming theme issues. Below is the submittal schedule for three areas of emphasis we are looking for:

**People Solutions to Software Problems**

May/June 2011

Submission Deadline: December 10, 2010

**DoD Gaming and Virtual World Applications**

July/August 2011

Submission Deadline: February 11, 2011

**Protecting Against Predatory Practices**

September/October 2011

Submission Deadline: April 8, 2011

Please follow the Author Guidelines for **CROSSTALK**, available on the Internet at <[www.crosstalkonline.org/submission-guidelines](http://www.crosstalkonline.org/submission-guidelines)>. We accept article submissions on software-related topics at any time, along with Letters to the Editor and BackTalk. To see a list of themes for upcoming issues or to learn more about the types of articles we're looking for visit <[www.crosstalkonline.org/theme-calendar](http://www.crosstalkonline.org/theme-calendar)>.