

Test and Evaluation for Agile Information Technologies

Steven Hutchison, Ph.D.

Defense Information Systems Agency, Falls Church, Virginia

Section 804 of the FY2010 National Defense Authorization Act directed the secretary of defense to develop and implement a new acquisition process for information technology (IT) systems. The law requires the Department of Defense (DoD) to base the new acquisition process on recommendations of the March 2009 Defense Science Board (DSB) Report on DoD Policies and Procedures for the Acquisition of Information Technology. The DSB recommended an agile model for acquiring IT similar to successful commercial practices. Agile software development is a high optempo process that delivers working software at “speed of need.” It is highly collaborative, documentation light, and change resilient. Agile focuses short development iterations on priority needs of the customer; in the DoD, the customer is the warfighter. In this model, an iteration is typically 8 weeks or less in duration. This article proposes a means to adopt the DoD IT test, evaluation, and certification (TE&C) process to an Agile model that will ensure TE&C continues to be an enabler of rapid acquisition of enhanced IT for the warfighter.

“Agile Information Technologies”—what does that mean? Agile (with a capital “A”) refers to a software development practice that follows the principles of the Agile Manifesto, of course. At this point, everyone with a smart phone should launch the browser and try that slick voice-command feature and check out what comes up. With a little luck, you’ll find yourself at agilemanifesto.org. Looking down at the fine print at the page bottom, you’ll notice that Agile is not a new idea—at least not new to industry; signed back in 2001, the principles of the Manifesto have been shaping software development for nearly 10 years. If that were only true of software development in the Department of Defense (DoD), I probably wouldn’t be writing this article! By the way, there’s a good chance that the apps you so readily find to enhance the capabilities of your smart phone were developed using Agile processes; I say that only because if they were developed using more traditional “waterfall” processes, they might not have been there for you to download when you needed them.

And that’s the point, right—the capability is there when you need it. In this author’s opinion, Agile is about delivery of capability at “speed of need.” Agile focuses

short development iterations on the *priority needs* of the customer. For those of us in the DoD acquisition arena, the customer is the warfighter, and there should be no doubt that our objective must be rapid fielding of enhanced capabilities to the warfighter. Hence, Agile would seem to be a “no-brainer” for the new DoD information technology (IT) acquisition process.

What *new* DoD IT acquisition process? By the time this article is published, it will have been over a year since the Congress directed the DoD to develop a new acquisition system for IT. The National

Defense Authorization Act for FY2010, Section 804, directed the secretary of defense to implement a new acquisition process for IT and report back to Congress in 270 days (which would have been July 2010) with the Department’s plans to implement the new process. Section 804 had some remarkably specific language, citing Chapter 6 of the Defense Science Board Report on Acquisition of IT (DSB-IT) (Defense Science Board 2009), published in March of 09, as the model to follow.



Steven J. Hutchison, Ph.D.

Report Documentation Page				Form Approved OMB No. 0704-0188	
Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.					
1. REPORT DATE 2010		2. REPORT TYPE		3. DATES COVERED 00-00-2010 to 00-00-2010	
4. TITLE AND SUBTITLE Test and Evaluation for Agile Information Technologies				5a. CONTRACT NUMBER	
				5b. GRANT NUMBER	
				5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S)				5d. PROJECT NUMBER	
				5e. TASK NUMBER	
				5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Defense Information Systems Agency, Test and Evaluation, Falls Church, VA, 22041				8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)				10. SPONSOR/MONITOR'S ACRONYM(S)	
				11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution unlimited					
13. SUPPLEMENTARY NOTES					
14. ABSTRACT					
15. SUBJECT TERMS					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT Same as Report (SAR)	18. NUMBER OF PAGES 7	19a. NAME OF RESPONSIBLE PERSON
a. REPORT unclassified	b. ABSTRACT unclassified	c. THIS PAGE unclassified			

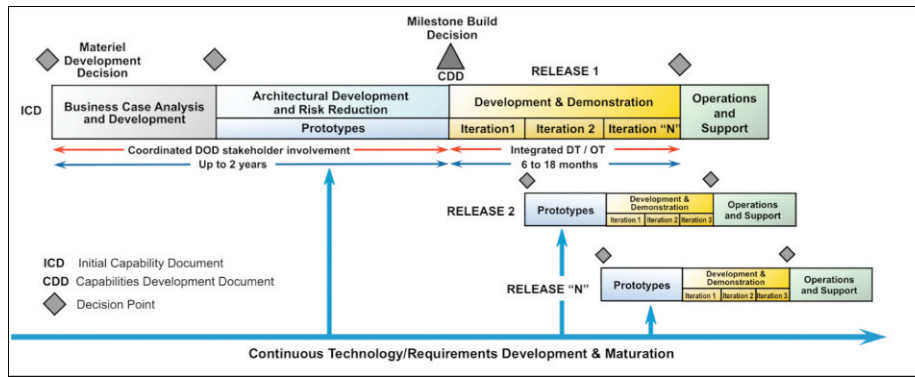


Figure 1. New acquisition process for information technology (Defense Science Board 2009).

So what did the Defense Science Board have to say? The DSB-IT concluded that current acquisition policies and processes (as defined in the DoD 5000 series directive and instruction) “cannot keep pace with the speed at which new capabilities are being introduced in today’s information age—and the speed with which potential adversaries can procure, adapt, and employ those same capabilities against the United States” (Defense Science Board 2009). As we marvel at the pace at which new electronic gadgetry shows up in stores, in our cars, and even in our living rooms, it is clear that technological advancements are far more readily available in the commercial sector than in the DoD. Let’s face it, if we could push blue force tracking data to the iPhone®, there would already be “an app for that™,” and our digital generation soldiers, sailors, airmen, and marines would be using it on the battlefield right now. The DoD *can* improve agility in delivery of IT products. To that end, the DSB-IT recommended a new IT acquisition process that “... is agile, geared to delivering meaningful increments of capability in approximately 18 months or less, and leverages the advantages of modern IT practices” (Defense Science Board 2009). *Figure 1* depicts the DSB-IT model.

The DSB-IT model features are as follows:

- multiple, rapidly executed releases of capability,
- early and continual involvement of the user, and
- integrated testing.

These are all good and necessary features of an IT acquisition system and are at the core of Agile processes. But change is never easy in the DoD, so before we jump in with both feet and say “let’s do Agile,” we should first take measure of the potential obstacles, so we can successfully overcome them on the road to Agile IT.

Rapidly executed releases of capability are the objective. We hear a lot about rapid acquisition these

days; in fact, the wars have been the source of greatest pressure to speed the process, since nothing can get to troops in harm’s way fast enough. Our acquisition system today is characterized by cumbersome processes beginning with lengthy, over-specified requirements, which require lengthy, complex development efforts, followed by long, complex test events. We can’t just substitute “rapidly executed releases” into the middle of this sequence and expect to have fixed the problem. To achieve rapid releases, we must have a requirements process that acknowledges and fosters evolving user priorities, and an equally agile test process. In other words, we can’t focus only on the middle; we have to fix the whole process, end-to-end. Rapidly executed releases must have an underpinning in an agile requirements process; likewise, evolving requirements (read “user priorities”) will demand more from our testers than we are currently structured to support. For IT capabilities, getting to Agile will stress the existing testing processes; in fact, the current approach will not work in the Agile IT environment. More on that later.

Early and continual involvement of the user is essential. However, this can be problematic for a Department at war—we simply may not be able to routinely task operating forces to support testing. We are going to have to be imaginative in how we conduct testing; leveraging exercises, experiments, and other venues. We will have to find ways to overcome the tension between testing and training to ensure mutual achievement of objectives. For Agile IT, we will need a user base (beta testers) from each IT community of interest that we can routinely draw from to conduct testing. With a sufficiently large pool of users to draw from, and leveraging other nontraditional test venues, including virtual testbeds, we should be able to overcome the challenges of high optempo deployments and test support.

Integrated testing has been a topic of discussion for decades. Some argue that we’ve been doing integrated

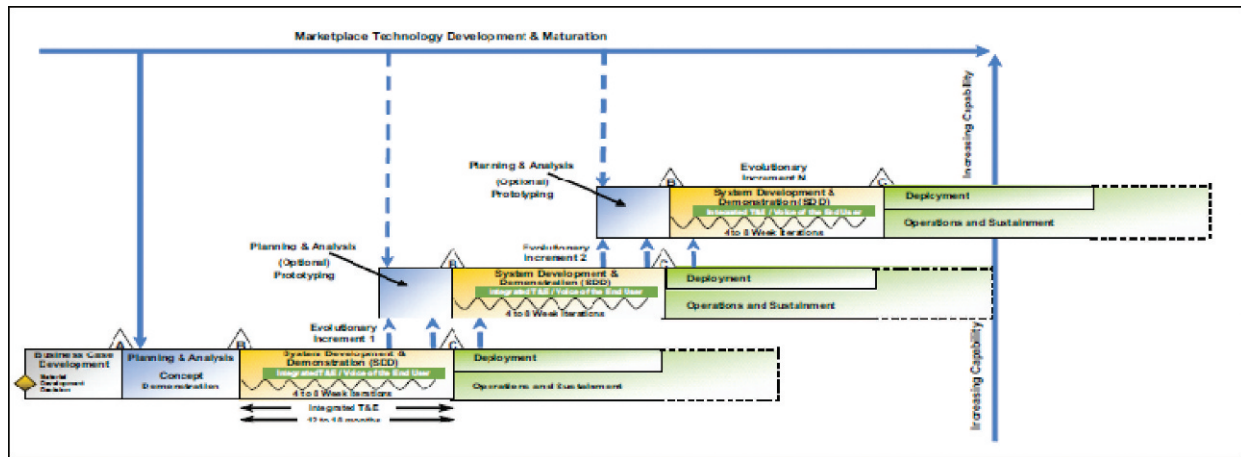


Figure 2. Information technology acquisition management approach (National Academies of Sciences 2010).

T&E all along, others that we need to start doing it. Unfortunately, we have not defined what integrated testing means for IT capabilities. In early 2008, the DoD defined “integrated testing” as, essentially, collaboration between the developmental test (DT) and operational test (OT) communities (<https://acc.dau.mil/CommunityBrowser.aspx?id=215765>). For IT, that’s only half the testers needed; integrated testing of IT involves not only DT and OT but also must include joint interoperability testing and security testing (information assurance). But why do we place all this emphasis on integrated testing? The motivation behind integrated testing is “early involvement” of the OT community; as the perception is that the OT folks begin T&E planning late in the process, so developers don’t understand how their product is going to be tested once OT starts, and this often results in late discovery of key failure modes, causing further cost and schedule delay. Early involvement is the key to reversing this trend; hence, the mandate for “integrated testing.” Integrated testing is really about *testing the capability as it is intended to be used*, and the sooner this starts, the better. In Agile software development, understanding how the capability will be used and tested is the motivation behind the practice known as “test driven development” (Beck 2002). For the DoD to adopt this approach, all of the test, evaluation, and certification (TE&C) organizations (DT, OT, interoperability, and security) will have to bring their needs to the table and make every test event a shared resource. There are, however, strong cultural barriers to this in the DoD, and it is clearly one of the obstacles we must remove to be successful at Agile.

The National Academies study

The DSB wasn’t the only group looking at acquisition of IT. DISA sponsored a study by the National Academies of Sciences who released their

final report in June 2010 (National Academies of Sciences 2010). *Figure 2* is the study committee’s version of an acquisition management approach for IT. The study committee refers to the overarching process as “iterative, incremental development,” and their model is generally consistent with the DSB-IT, including the three central points just reviewed: rapid release of capability, continuous user involvement, and integrated T&E. Yet there are also some notable differences. *Figure 3* shows the central part of this model in detail. Notice the green banner “integrated T&E/Voice of the End User.” The committee is making an important distinction between integrated testing (as described in the DSB-IT report) and integrating testers and users; that is, it is not enough to know that the system meets requirements, it is equally important to know whether the user thinks the iteration delivers militarily useful capability. Another distinguishing feature of the model is the “sine wave” with the words “4 to 8 Week Iterations” written beneath. Each peak-to-peak transit of the wave represents a complete software development iteration, or “sprint.” These sprints are obviously considerably shorter than the DSB-IT’s nominal 6-month iterations, and a lot closer to commercial Agile practices. *Figure 4* shows the details of the wave, and as described in the report, “Each iteration will include analysis, design, development, integration, and testing to produce a progressively more defined and capable,

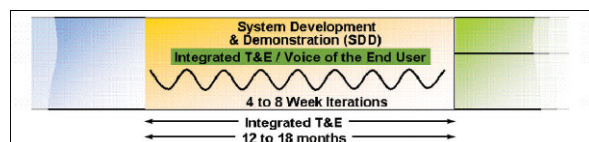


Figure 3. Capability increment in detail (National Academies of Sciences 2010).

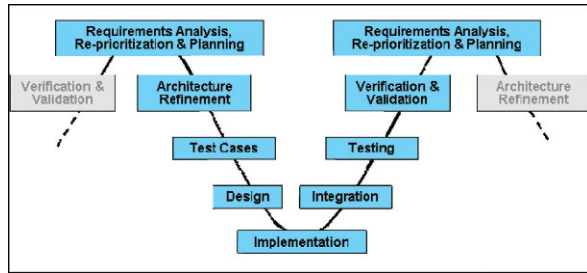


Figure 4. Key elements of the iteration (National Academies of Sciences 2010).

fully integrated and tested product” (National Academies of Sciences 2010). The wave is obviously very similar to what we know as the systems engineering “V,” but with several key differences. As we begin following the process down the left-hand side of the V, the iteration begins with requirements analysis and architecture refinement—the latter being an essential consideration for IT. This model then inserts “Test Cases.” Note its placement on the left-hand side of the V, before design and coding begins. Here testers translate “user stories” (a description of how the capability is used) into executable test cases. This is “test driven development” and is by definition “early involvement” when *all* testing stakeholders participate in writing the test cases.

Continuing through the iteration, design and build begin, and then “Testing” occurs on the right side. This is *independent* testing with users, not developer testing, but should be understood to be a team effort of all TE&C stakeholders. In the words of the study committee (National Academies of Sciences 2010),

“Therefore, an integrated approach to T&E to include the voice of the end user; traditional [DT&E]; [OT&E]; interoperability certification; and information assurance certification and accreditation equities is a fundamental element of this modified acquisition management approach for IT programs. As was the case with the requirements process, this implies a profound change in the T&E process used for such programs.”

Complete integration is the key to T&E at the speed of need.

The current DoD information technology TE&C environment

Our current test and certification process does a good job at helping users and decision makers understand capabilities and limitations, but it can be lengthy, costly, and duplicative. It is not agile. *Figure 5* depicts a high-level view of the Plan-Test-Report (PTR) cycle for IT TE&C. This PTR cycle can take 6 months, although it can be shorter or longer. As the diagram indicates, DT, OT, interoperability, and security testing can and often do occur as separate events, with their respective test teams performing separate analyses and producing separate reports. The process concludes as the various reports inform the milestone decision authority’s acquisition (procurement) decision, the Joint Staff J6 interoperability certification, and the designated approving authority’s information assurance accreditation. It is a kludge of IT considerations overlaid on a weapons-based acquisition system—but—just as for weapons and major platforms, when it takes years to develop and deliver a new IT capability, this process works. It is just not well suited for Agile IT. What we need is a TE&C model that is fully integrated, less duplicative, less costly, and ultimately one that fuses all test information into a coherent evaluation, so that decision makers better understand capabilities and limitations when making decisions about deploying the capability. What we need is an Agile testing model.

Agile for DoD

So what might an Agile IT acquisition process look like, aside from the DSB-IT’s notion of “18-month releases subdivided into iterations”? Agile software development is a high optempo process that delivers working capability at speed of need. It is highly collaborative, documentation light, and change resilient. *Figure 6* depicts an Agile capability development life cycle adapted from the “Scrum” framework for iterative, incremental development. There are many

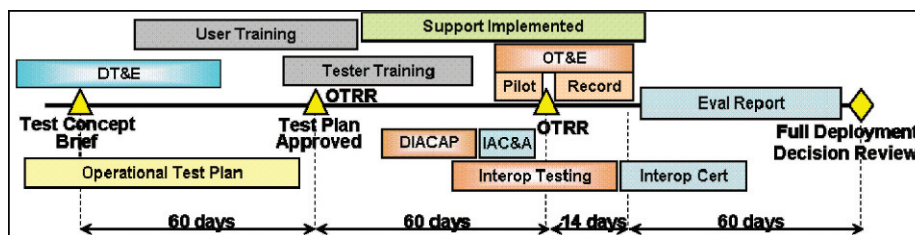


Figure 5. Test, evaluation, and certification of Department of Defense information technology.

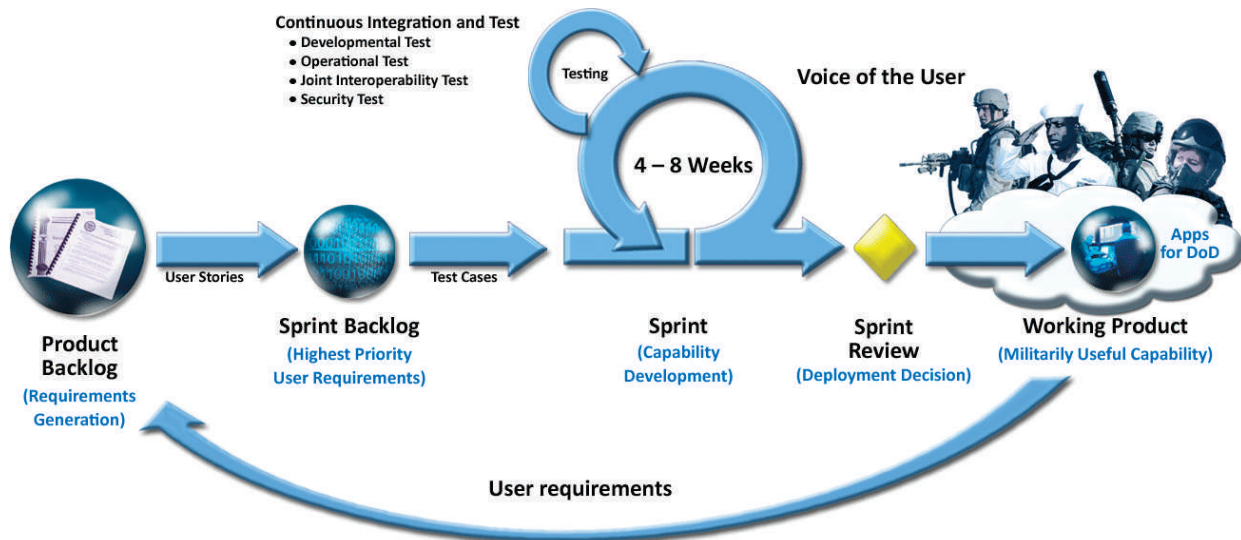


Figure 6. An agile development life cycle adapted for the Department of Defense.

sources of information on Scrum (www.scrum.org) and the Agile life cycle (www.ambyssoft.com). Scrum succeeds through team member commitment and by removal of impediments; it enables The Team (a cross-functional group with necessary expertise to deliver a potentially deployable product at each sprint) to self-organize and achieve “hyper-productive” results.

In the model depicted in Figure 6, the key stages are “time-boxed,” so that development can be accomplished at a sustainable pace. The “product owner” is responsible for articulating the product vision and identifying features in priority order (the commercial sector refers to this list of features as the product backlog). In the DoD, the operational sponsor would likely fill the role as product owner. In Agile, the product backlog evolves over time with priorities updated as features are added and removed to reflect the emerging needs of the customer. This is a critical distinguishing characteristic of Agile software development; resilience to change means that a change in the warfighter’s priorities or needs could be just *one sprint away* from delivery.

The Agile process values working software over lengthy documentation (per the Agile Manifesto); therefore, to follow this development practice, we will need to revise the DoD requirements generation process to shift away from rigid requirements definition expressed in capability development documents written years before a product is delivered,¹ to a flexible, priority-driven process responsive to the changing needs of the warfighter. Our interoperability and information assurance certification processes also have to be revised for Agile IT. Likewise, since test

activities will be responding to prioritized requirements at each sprint, it is unlikely that we can adequately describe test objectives, scope, and resources as we currently do in a Test and Evaluation Master Plan (TEMP), so we will need to shift the emphasis on detailed descriptions in the TEMP (objectives, scope, and resources) to well-crafted test cases in each sprint.

In the next step, The Team, not the product owner, selects the features from the product backlog that they can commit to develop during the sprint (keeping in mind that the duration of the sprint is a fixed period of time), taking the highest priority items and working down the list. Before The Team can make the commitment, they have to translate user stories into tasks and test cases to better understand the level of effort required to deliver each feature in the product backlog. In this way, The Team takes ownership of the development effort, while assuring the product owner that the highest priority items are included. This short list of priority features constitutes the sprint backlog.

A user story can be described by the simple statement, “As a *role*, I want to *what*, so that *why*.” For example, “As an operator, I want to display current blue force locations, so that I have better situational awareness.” In the DoD, a “mission thread” is likely to contain numerous user stories. The user story is further decomposed into tasks, and test cases are written before the sprint begins. This is the “test driven development” practice referred to earlier. Test driven development has shown that when developers understand how the capability will be tested, the resultant code has fewer defects. For the DoD, this is the type of early involvement we have been struggling

to achieve; if we can get the complete team of government testers (developmental, operational, interoperability, security) involved this early, we should be able to significantly improve the quality of the product and reduce time to deployment.

In this model, a sprint is typically 8 weeks or less in duration. Once the sprint begins, the product owner cannot change the priorities; any changes will be addressed in the next sprint. During the sprint, items in the sprint backlog are developed and continuously integrated and tested. In the commercial sector, this typically includes unit testing, acceptance testing, and exploratory testing. For the DoD, “Agile Testing” must accommodate the functions performed by government developmental testers, operational testers, joint interoperability testers, and information security testers—but these efforts are integrated and continuous, not separate and serial. When the sprint is complete and working software is ready, a sprint review is conducted at which all stakeholders are present, the capability is demonstrated, and the decision made whether or not to deploy the product.

Agile testing

To shift the DoD IT test and certification paradigm to be responsive to Agile IT programs, we need to move away from the “who does what, when” process (e.g., program manager does DT, the OTA does OT) to a collaborative model built upon shared data and reciprocity of test results that is ultimately an enabling process for delivering working capability. Let’s take what’s good from our process shown in *Figure 5* and collapse it into a responsive, on-demand, “testing as a service” construct. In other words, let’s test smart.

To set the conditions for success of Agile Testing, we must first move away from the linear, serial processes that characterize development and test today. The Agile environment is iterative and collaborative; it exploits the principles of the Manifesto to achieve desired effects. An empowered team can reduce lengthy coordination cycles for document approvals, readiness reviews, etc. Likewise, a team approach will reduce duplication during test execution and publish more comprehensive findings on capabilities and limitations. Empowerment is critical to rapid development and deployment of working capability.

Next there are three key elements in our current (*Figure 5*) process that we must make *persistent* resources in the Agile life cycle; these include user training, tester training, and support structure (help desk). The help desk, as it is intended to support operations, must be in place during every development iteration. Also, since early and continuous involvement of the users is fundamental to success in the Agile

environment, we will need to establish a pool of knowledgeable users (beta testers) from each community of interest (C2, business, intel, etc.) to ensure that we can obtain an adequate number of users to test. Likewise, to support the high test optempo, we must be able to draw from a cadre of testers knowledgeable in the systems and services in the capability area, representing all TE&C disciplines. This cadre must be able to engage early, be responsive to evolving user priorities, and execute the PTR cycle in highly compressed time lines.

Not shown in *Figure 5* are additional factors required to support Agile projects, including training our acquisition workforce, providing an enterprise knowledge management capability, and implementing a persistent integration and test environment. As part of improved training for the IT workforce, we need to update our curriculum in the Defense Acquisition University to better prepare our program managers and testers for IT programs in general, and Agile practices in particular. We need a project dashboard for IT programs that provides comprehensive and transparent knowledge management capabilities for all stakeholders. The DoD has spent considerable dollars funding programs in a way that allows them to build their own program-specific system integration labs (SILs). This strategy has failed; in fact, the plethora of SILs has only aggravated the Department’s interoperability crisis. A new approach is needed. For example, instead of funding new programs to build more SILs, let’s fund a select few SILs across the DoD to serve as a common development, integration, and test environment, and federate them together to ensure access as a shared resource. DISA is providing one such environment in Forge.mil (www.forge.mil), and within this virtual environment, the TestForge.mil will provide robust capabilities for users and testers to ensure capabilities perform as desired. The degree to which we can provide a common environment, common test tools, common methods, data collection, etc., will help all phases of the development process become more agile. A common development, integration, and test environment may eventually provide the foundation for “apps for DoD,” similar to the app stores we see supporting our favorite gadgets.

The traditional PTR activities depicted in *Figure 5* can be adapted to the Agile environment, and each has a role; we don’t sacrifice rigor in Agile testing. The Capability Test Team (CTT)² merges and consolidates these PTR activities but does so in a manner that enables each stakeholder to accomplish their evaluation objectives. The CTT is engaged from the outset; so as requirements are prioritized for each sprint, the team translates user stories into test cases. Test cases are risk

based and mission focused, and they address relevant technical parameters, operational issues, interoperability measures, and security measures. In Agile processes, test execution relies more heavily on automation, such as load simulators. Defects that cannot be corrected during the course of the sprint are returned to the work stack; working software is eligible to be fielded. Following the test, the CTT posts the evaluation report to the dashboard, with findings that state whether the capability is effective, suitable, interoperable, and secure. In 8-week iterations, the PTR cycle should be completed in 6 weeks. A single evaluation report could support the acquisition decision, interoperability certification, and the information assurance certification and accreditation. Last, we should modify the deployment decision. Rather than a “full deployment decision review,” we should adopt one where we “start small and scale rapidly,” with testers in a continuous monitoring role. In this way, we can ensure the capability effectively supports operations at scale, or take corrective actions should a problem arise.

Summary

A new IT acquisition system is coming to the DoD that will feature much higher optempo in development, testing, and fielding. As we evolve our acquisition process to deliver capabilities at the speed of need, test, evaluation, and certification will need to adapt processes to this new environment. The Agile environment will require a capability test team that is empowered to execute the plan-test-report cycle and provide objective assessments of key technical, operational, interoperability, and security metrics necessary for decision makers to understand capabilities and limitations. Key to the approach is to treat all test

activities as a shared resource, while being mindful of each test organization’s roles and responsibilities. Continuous user involvement, combined with appropriate risk-based, mission-focused testing will ensure TE&C is an enabler of rapid acquisition of enhanced information technologies for the warfighter, and this in turn will help ensure the critical apps that warfighters need are there when they need them. □

STEVEN HUTCHISON, Ph.D., is the Test and Evaluation Executive, Defense Information Systems Agency. He is a certified ScrumMaster. E-mail: steven.hutchison@disa.mil

Endnotes

¹The Defense Science Board Report on Policies and Procedures for the Acquisition of Information Technology, March 2009, reported “... an average of 48 months to deliver useful functionality from the Milestone B decision...”

²The capability test team members are empowered representatives of all test and certification organizations and the user community.

References

- Beck, Kent. 2002. *Test driven development: by example*. Boston, MA: Addison-Wesley Professional.
- Defense Science Board. 2009. *Department of Defense policies and procedures for the acquisition of information technology*. <http://www.acq.osd.mil/dsb/reports/ADA498375.pdf> (accessed October 7, 2010).
- National Academies of Sciences. 2010. *Achieving effective acquisition of information technology in the Department of Defense*. Washington, D.C.: The National Academies Press. http://www.nap.edu/catalog.php?record_id=12823 (accessed October 7, 2010).