

Agile in Government

Written Testimony of Software Engineering Institute's Agile in Government Team to House Ways and Means SSA Subcommittee



Authors

Suzanne Miller, Principal Researcher

William Hayes, Principal Engineer and Testimony Speaker

Eileen Wrubel, Program Manager

SEI Agile in Government Team

Provided by:

Software Engineering Institute

A Federally Funded Research and Development Center
operated by Carnegie Mellon University

Date:

July 14, 2016

1 SEI Testimony to House Ways and Means Subcommittee: Bottom Line Up Front

In the following testimony, the Software Engineering Institute (SEI) Agile in Government team addresses these points:

- Agile is not a “silver bullet” that solves all the complex management and engineering problems of government IT Transformation efforts, but it has contributed to successful IT Transformation efforts.
- The benefits achievable when development organizations use Agile methods only manifest when the development (government or contractor developer) and oversight (government acquisition) efforts are aligned.
- The approach and mindset to government obligations in oversight must change when Agile is the focus of development and the SEI has observed negative consequences in organizations that do not address these changes.
- Changing the oversight approach in Agile settings means asking different questions on a new cadence than oversight organizations have in the past. This leads to different measurement and reporting approaches as well.
- A focused government workforce development effort is required to enable the knowledge, skills, and abilities needed for effective oversight and interaction in Agile settings.

2 Software Engineering Institute’s Agile in Government Team Perspective on Agile in IT Transformation

2.1 Introduction

The Software Engineering Institute’s (SEI) Agile in Government team is pleased to provide testimony to the House Ways and Means Committee on the topic of Agile opportunities and challenges in IT transformation oversight. For over three decades, the SEI has been helping government and industry organizations to acquire, develop, operate, and sustain software systems that are innovative, affordable, enduring, and trustworthy. We serve the nation as a Federally Funded Research and Development Center (FFRDC) sponsored by the U.S. Department of Defense (DoD) and are based at Carnegie Mellon University, a global research university annually rated among the best for its programs in computer science and engineering.

Overseeing the acquisition of large software intensive systems in government settings brings many challenges; however, we believe our observations and experiences with government programs using Agile methods in IT transformation can elucidate the issue. Our observations focused on senior oversight aspects of these projects reveal practical application and policy implications.

2.2 Major Observations

Agile is an iterative approach to software delivery that builds and delivers software incrementally from the start of the project, instead of trying to deliver it all at once near the end (for more detail, see the Appendix). In the case of an Agile life cycle, requirements and solutions evolve through collaboration among self-organizing teams and project sponsors to encourage rapid and flexible response to change. The following observations reflect over seven years of SEI involvement in research and field support of the application of Agile methods and principles in government settings.

The SEI’s Agile in Government team was chartered in 2009 to initially answer the question posed by a senior US Air Force official: “Can Agile methods actually be used in government settings?” That question led to researching multiple dimensions of the application of Agile methods and principles in a wide variety of government scopes and settings. [Ward & Miller 2016]

Our first observation is one that may seem obvious—Agile cannot solve all your IT Transformation problems. IT Transformation is a multi-dimensional challenge. Agile approaches, although not a silver bullet, can contribute to successful transformations when appropriate conditions are met. In a recent article titled “Embracing Agile,” the *Harvard Business Review* asserts several conditions that reflect an appropriate setting for Agile success [Rigby 2016]:

- The problem to be solved is complex.
- Solutions are initially unknown.
- Product requirements will most likely change.
- Work can be modularized.
- Close collaboration with end users (and rapid feedback from them) is feasible.

We have worked in several government settings that meet all of these criteria and have often found Agile to provide benefits that are valued by the sponsors of the projects as well as the end users of the software products, albeit not without challenges. The SEI is familiar with many of the particular adoption hurdles of Agile development and management, especially in a contracted setting. [Lapham 2011]

Table 1: SEI Observations on the Significant Benefits of Agile

Benefit Summary	Notes
Early opportunity for course correction, especially when the environment changes after a program has begun	It is amazing how perspectives change on what a system should do or contain once users actually see the working software. Several times the SEI Agile in Government team has heard statements like “Thank goodness we implemented that aspect of the system early. We never would have known to change X if we had just been working with the requirements and design documents.”
Early risk reduction, especially in user-facing areas of the system	Agile’s emphasis on early learning by implementing allows a program to focus risk reduction via implementation, rather than speculation through documents that might or might not be correct.
Shorter “idea to realization” cycle which results in fast user feedback for future increments of functionality	In the Defense arena we call the life cycle “concept to capability.” The ideas are the same. The faster we get feedback from real users on what we have provided, the faster we can improve the product to meet evolving needs.

However, the benefits are only achievable when oversight and development are aligned in several areas to include both development execution and mindset changes around the topics listed in Table 2. [Miller 2014]

Table 2. Factor Categories for Focusing Alignment in Agile Settings

Business and Acquisition	Organizational Climate
Team, project, and customer environment	System attributes
Technology environment	Development practices (particularly small batches, collaboration and engagement, and continuous integration)

Lastly, particular challenges relevant to oversight are catalogued in Table 3.

Table 3: Summary of Challenges Observed by the SEI in Overseeing Agile Programs in Government

Challenge Summary	Challenge Notes
The need for requirements to be allowed to evolve, and the need for ongoing government involvement in their prioritization	Treating initial requirements as final requirements has never led to successful IT projects. [Nielsen 2009] This is primarily due to the volatile technical and operational environments our systems are expected to robustly work within.
Lack of communication and alignment of program goals and strategic initiatives	Although this is a challenge in any IT Transformation setting, the effect of poor communication and alignment is more visible in settings where the delivery cadence is on the order of weeks rather than months. Frequent deliveries allow the course correction that may be needed to mitigate this issue.
Lack of workforce knowledge, skills, and abilities related to overseeing Agile projects	This has been a consistent challenge across multiple programs adopting Agile and lean methods. Early adopters struggle to find training to prepare them for new duties. Invariably the second wave of adopters gets more training than the

	first, after the experience of the first wave in adjusting to government oversight obligations.
Difficulties in aligning acquisition policy with small batch approaches like Agile	Slowly the acquisition policy environment is changing; until recently, enabling incremental delivery, review, and production of documentation and software products was challenging. We have seen multiple programs who had to deal with the overhead of large-scale document reviews at the same time they were trying to implement small batches of the software described in the documents.

2.2.1 Agile in Software Acquisition

Agile, as an alternative to traditional project management, can be a response to common acquisition failures. Large technology projects are fundamentally uncertain—time and technological advancement ensures that what you want today, is almost never what you want tomorrow or what you will get in a week. Agile’s inherent flexibility can reduce some of the adaptability risks ingrained into large software projects.

For example, requirements creep is a universally acknowledged problem, and dealing with changes in requirements is considered the bane of program managers; yet, regrettably “Previous experience shows that changes within an SIS [software-intensive system] are inevitable, whether or not there are changes in requirements or technology.” [Kennedy 2011]

A strength of Agile approaches is to explicitly anticipate and factor in the effects of inevitable change; however, oversight of systems development is often not consistent with Agile principles and methods. (See the Appendix)

2.2.2 Oversight for Agile has Different Questions

Organizations that are successful ask different questions—not the ones typically found in oversight guidance. We observe the following themes in effective organizations:

- emphasis on mission needs priority vs. cost
- continual focus on product quality
- evolving systems based on learning vs. big bang
- oversight cadence aligned with delivery of small batches of work

3 Overseeing Agile Settings in a Government Context

In most cases, government organizations are not doing Agile development themselves; more often they engage with Agile development contractors. Even in the case where organic development is occurring within a government organization, senior level oversight of Agile should take on a different cadence and focus than in traditional, large batch development settings.

3.1 Focus Areas of Oversight Approach in Agile Settings

The responsibility for oversight and due diligence does not change, but the approach to oversight in an Agile setting does. Topics relevant for senior government oversight personnel include, but are not limited to:

- level of understanding government personnel have for the complex tradeoff space involved when making decisions to replace, evolve, or fix an aging legacy system
- level of understanding government personnel have of Agile concepts and their relevance to program performance; examples to discuss might include:
 - identifying clashes between Agile timelines and traditional long-lead time expectations
 - managing technical debt and appropriately investing in modernizing legacy system components
 - assuring investment in technical infrastructure required to enable rapid delivery of quality products (e.g., test environments)
- level of understanding of government program about how to partition system deliveries to gain incremental value on a regular, short-term basis while maintaining alignment with enterprise architecture vision

- level of willingness of government program office to engage in small batch interactions on a frequent (as much as every two weeks) cadence with the developer organization(s).

Although these areas of focus are not all directly amenable to traditional measurement, the discussion of the topic with the government program office is likely to lead the senior oversight staff to greater or lesser confidence in the government program’s ability and commitment to execute in an Agile fashion.

3.2 Progress Measurements that are Relevant for Oversight of Agile Settings in Government

The following suggestions are offered with the caveat that relevant leading indicators of progress are highly specific to a particular context. The adage “what gets measured, gets done” is both fortunately and unfortunately true. Measures often lead to unintended consequences that are in opposition to the goals of those who established the measures.

Many of the typical measures you will see with Agile methods in the literature are focused on development team measures; however, those are not relevant for senior oversight, and their use by senior-level oversight is likely to skew the focus of the work being done. [Austin 1996] [Hayes 2013]

We offer two types of suggestions with regard to measures:

- Categories of measures that should be discussed with the government program in terms of how things relevant to the particular effort are being measured. Agreement on measures to send forward to the senior oversight group would be a result of this discussion.
- Measurement examples and their connection to the categories discussed above.

Table 4. Categories of Measures to Consider in Senior Oversight of Agile Settings

Category	Description
Flow	Flow measures come out of the lean engineering and management environment. They focus on understanding the “idea to realization” cycle time. Flow measures for senior oversight focus on the development organization’s ability to consistently meet timelines for deployment of IT functions according to a roadmap. These are cycles measured in weeks and months, rather than quarterly or annual cycles seen traditionally.
Engagement	Engagement measures help oversight organizations understand the level of collaboration that has been achieved. Timely involvement of stakeholders from the workflow supported by the IT system results in a deeper understanding of intended usage. Evolution of the workflow to better utilize technology results from engagement with the correct decision makers.
Quality	Quality measures at senior oversight levels have less to do with software defect rates than they do with the quality of the services supported by the IT systems. For example, improvements in wait times for key services, or percentage of “made it through in one pass” attempts to use a service are potential quality measures. These measures, in turn, drive the priorities for quality measures among software teams.
Risk	Risk measures for senior oversight can focus on the development organizations’ performance in managing threats to their success, more than those threats themselves. When using Agile methods, confidently asserting the expected success of a program is no longer based on the comprehensiveness of up-front specification documents. Therefore, an oversight approach for Agile cannot rely on review and approval of such projective documents as the primary mode of risk identification. The short and steady cadence of Agile promotes rapid learning.

The following are examples of measures we have seen used in executive-level dashboards for services similar to those provided by government agencies.

Table 5. Examples of Executive Dashboard Measures Related to Agile that SEI Has Seen

Measurement	Description
Deployment according to roadmap (flow)	An indicator of reliable delivery (into production) of software capabilities in accordance with the roadmap for identified features and technologies.
Predictable delivery volume (flow)	The percentage of planned capabilities successfully released (whether into a staging area or production) at the conclusion of a set of iterations.
Deployment speed (flow)	The typical cycle time to deploy IT capabilities into production can be dramatically reduced using concepts in DevOps.
Stakeholder involvement (engagement)	Participation of stakeholders, as well as subject matter experts involved in technical work, in the events and communication mechanisms intended for them.
Synchronization of cadence across the supply chain ()	Agile prime contractors work with subcontractors to synchronize incremental delivery plans. Agile methods drive the synchronization points to occur more frequently than is typical in the past, and with a wider array of contributors.
IT-enabled workflow performance (quality)	Measurements of the service system performance before and after deployment of new IT capabilities (e.g., speed, accuracy, and capacity)
Defect backlog (quality)	Defect severity, concentration in particular system components, and defect age measures help to describe the backlog of defects managed by an organization.
First pass fix rate (quality)	An indicator of efficiency in rework, this is the percentage of defects that are correctly repaired and pass testing on the first attempt.
Deferred complexity (risk)	Many Agile organizations choose to solve the hard problems early, rather than deferring them in favor of low-risk work. Appropriate focus on enablers including architecture and infrastructure enable this pattern to a greater extent as well.

4 IT Transformation Challenges Beyond Agile

Agile approaches in major IT transformation efforts must successfully account for key success/failure drivers that persist regardless of the development cadence and methodology chosen.

Table 6. Important IT Transformation Considerations Beyond Agile

Important IT Transformation Area	Explanation of Importance
Managing technical debt	At its simplest, technical debt is the sum of both intentional and unintentional (usually via defects) deferral of technical work on a system to a later point in its evolution. Intentional deferral — sometimes related to postponing high difficulty work or avoiding architecture changes until a later point — also affect the resilience and evolvability of the system. Practices related to technical debt in Agile settings are suggested by the SEI's Software Solutions Division. http://www.sei.cmu.edu/architecture/research/arch_tech_debt/arch_tech_debt_library.cfm
Effectively dealing with cybersecurity	Cybersecurity and its interaction with IT systems has led to NIST's Risk-based Management Framework. [NIST 2010] Understanding and responding to the security risks in any IT transformation is a priority. Practices like secure coding and SQUARE (a method for eliciting relevant security requirements) are supported by the SEI's CERT Division. Agile methods may change the implementation pattern for some best practices — potentially amplifying their effectiveness with more frequent iterations.

	http://www.cert.org/secure-coding/ and http://www.cert.org/cybersecurity-engineering/products-services/square.cfm
Understanding the quality attributes	A focus on quality attributes (non-functional requirements), such as security, safety, dependability, performance, affordability, achievability, effectiveness, and flexibility, allows us to categorize a subset of system requirements that tend to pervade multiple elements, or sometimes the entire, system. For example, security of an IT system is not typically satisfied within a single element of the system, just as air worthiness of an aircraft does not rest in a single component. The prioritization of quality attributes is usually visible via the system's architecture. Not all Agile methods explicitly include quality attribute requirements in their approach. Methods for understanding the interactions and tradeoffs among quality attributes are available from the SEI's Architecture Practices team. http://www.sei.cmu.edu/architecture/tools/establish/qaw.cfm
Architecting for evolution and incremental delivery	Architecting a system for evolvability and sustainability is an ongoing process within a system development. Concepts like architecture runways are used in some Agile methods to ensure that sufficient framework for making design decisions is in place for teams to work effectively. Incremental development and delivery also encourages an evolvable architecture that is sustainable throughout the long life of many government IT systems. The SEI's Architecture team provides concepts and practices to support evolvable, sustainable software architectures. http://www.sei.cmu.edu/architecture/
Communicating needs effectively to system developers	Particularly early in development, when system concepts are still being formulated, communicating across the developer/user organizations can be difficult. Agile methods contain approaches (like user stories) to help address this challenge. Frequent interaction, using working software products to demonstrate implementation choices, helps address this topic as well. SEI's Measurement and Analysis team also offers methods like QUELCE (Quantifying Uncertainty in Early Lifecycle Cost Estimation) that help stakeholders specifically identify and assess ambiguities in early life cycle activities. https://www.sei.cmu.edu/measurement/research/quelce/index.cfm

5 Call to Action

If asked for recommendations, the SEI Agile in Government Team would emphasize two areas:

- Ask different questions of those using Agile as part of their IT Transformation strategy
- Support development of Agile oversight knowledge, skills, and abilities for the government workforce

5.1 Ask Different Questions

Any successful IT transformation involves more than the dedication of talented development staff. Supervisors must ask about alignment of people, processes (Agile or otherwise), technology support, or business context. There are two particular contexts for which different questions could be relevant—evaluation of an agency's IT transformation strategy, and oversight of an agency's ongoing IT transformation effort.

About the (Agile) IT Transformation Strategy

- If iterative and incremental approaches to modernization are intended, how does the cadence of work done by the contributing elements of the organization align to the modernization roadmap? How will leadership keep the effort synchronized, when the pace of deployment is expected to increase dramatically?
- Depending on the scope of workflow activities served by the IT transformation effort, how large and diverse is the cohort of service delivery staff who will see an IT-driven change? How deep into the flow of normal work activities will this IT-driven change permeate?
- Is there evidence of consideration for resolving high-uncertainty or high-risk areas at appropriate times in the timeline—rather than deferring undue amounts of risk?
- With a roadmap for incremental deployment of these IT-driven changes in hand, stakeholders can ask focused questions about the value returned to those served by the organization. Questions can focus on the

timeframe and/or scope of deploying certain new technologies (or the retirement of old subsystems) in light of the scope of anticipated impact.

About the ongoing (Agile) IT Transformation Effort

- How successful is the organization in defining small batch approaches to the work? Are the development teams able to sustain a short iteration length (measured in weeks, not months)?
- What is the trend in the time it takes to mature a concept all the way through development activities to the point where it is an accepted element of a service or maintenance workflow?
- How consistently does the organization deploy capabilities in accordance with the transformation roadmap?
- Are planned major technology updates/introductions occurring according to the roadmap?
- How well is the balance of routine system maintenance, defect repair, and new deployments being managed? Does the pattern of activity match the plan, and is the organization's capacity adequate?
- What patterns of feedback are seen from the users of IT systems involved in the transformation? User experiences associated with the IT systems as well as the Agile approach should be sought.

No matter what specific measures are chosen to provide oversight, it is vitally important to avoid limiting visibility to measures expressed in units of resource consumed (be it time or money). Most Agile methodologies call for steady resource loading and fixed iteration lengths. Schedule and cost overruns are lagging indicators, which can be anticipated by examining the flow of delivered capabilities in Agile development.

5.2 Workforce Development of Government Staff Working in Agile Settings

Increasing access to Agile-related training for government staff has already started in some areas. OMB Digital Service has sponsored development of customized learning paths and related courses to help contracting officers and others involved in managing Agile efforts in government to understand the concepts and to offer guidance in fulfilling agency needs with Agile projects. [OMB 2016] The U.S. Digital Service has also sponsored several outreach and awareness building communication activities targeted at government agencies. [USDS 2016]

Several federal agencies, including Department of Homeland Security and the Internal Revenue Service, have written guidance or policies related to enabling Agile IT projects to productively occur.

In the Department of Defense arena, the Defense Acquisition University is in the process of enhancing its IT acquisition curriculum to include content related to Agile. A continuous learning course on Agile considerations in government acquisition is in work, for which the SEI Agile in Government team is an author.

One caution: There are many certifications available in the commercial community related to Agile, and almost all of them focus on roles directly involved in software development. Although many have useful content for government staff to know, a certification program that focuses on the commonly available Agile methodologies would not serve the knowledge and skill needs of the staff who write the contracts and do the oversight of government IT projects.

Appendix: Agile Tenets and Principles

The basis for all Agile methods is a set of four tenets and 12 principles that were developed by a group of software methodologists in 2000. They were concerned that more and more software projects in the commercial space (their primary area of work) were becoming larger, more unwieldy, and less successful than smaller, more agile projects they had observed. Throughout two days of collaboration, they established the Manifesto that is reproduced here. Any method that claims to be Agile should be able to discuss how its practices and methods express these tenets and principles.

With each principle, we have included a short statement of adaptations that we see being needed to express the principle in government settings.

Manifesto for Agile Software Development

<http://www.agilemanifesto.org/>

“We are uncovering better ways of developing software by doing it and helping others do it.” Through this work we have come to value:

- **Individuals and interactions** over processes and tools
- **Working software** over comprehensive documentation
- **Customer collaboration** over contract negotiation
- **Responding to change** over following a plan

That is, while there is value in the items on the right, we value the items on the left more.

Table A1. Twelve Agile Principles Derived from the Agile Manifesto.

Agile Principle	Useful Interpretations in Government Settings
The highest priority is to satisfy the customer through early and continuous delivery of valuable software.	In government, the customer is not always the end user. The customer includes people who pay for, people who use, people who maintain, as well as others. These stakeholders often have conflicting needs that must be reconciled.
Welcome changing requirements, even late in development. Agile processes harness change for the customer’s competitive advantage.	Rather than saying “competitive” advantage, we usually say “operational” advantage. This principle causes culture clash with the “all requirements up front” perspective of many large, traditional approaches.
Deliver working software frequently, from a couple of weeks to a couple of months, with a preference for the shorter timescale.	What it means to “deliver” an increment of software may well depend on context. With large embedded systems, we are sometimes looking at a re-release into a testing lab. Also, for some systems, the operational users are not able to accept all deliveries on the development cadence—because there are accompanying changes in the workflow supported by the software that require updates.
Business people and developers must work together daily throughout the project.	In government settings, we interpret “business” people to be end users and operators, as well as the other types of stakeholders mentioned in Principle 1, since in many government settings, the business people are interpreted as the contracts and finance group.
Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.	A frequent challenge in government is to provide a suitable technical and management environment to foster the trust that is inherent in Agile settings. Allowing teams to stay intact and focused on a single work stream is another challenge.
The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.	In today’s world, even in commercial settings, this is often interpreted as “high bandwidth” rather than only face-to-face. Telepresence via video or screen-sharing allows more distributed work groups than in the past.
Working software is the primary measure of progress.	Our typical government system development approaches use surrogates for software -- documents that project the needed requirements and design -- rather than the software itself, as measures of progress. Going to small batches in short increments allows this principle to be enacted, even in government settings, although delivery may well to be a test environment or some internal group other than users themselves.
Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.	This principle is a caution against seeing agility just as “do it faster.” Note that this principle includes stakeholders outside of the development team as part of the pacing.
Continuous attention to technical excellence and good design enhances agility.	This is a principle that often is cited as already being compatible with traditional government development.
Simplicity—the art of maximizing the amount of work not done—is essential.	One issue with this principle in government settings is that our contracts are often written to penalize development organization if they don’t produce a product that reflects 100% of the requirements. This principle recognizes that not all requirements we think are needed at the onset of a project will necessarily turn out to be things that should be included in the product.

The best architectures, requirements, and designs emerge from self-organizing teams.	Note that the principle does not suggest that the development team is necessarily the correct team for requirements and architecture. It is, however, encouraging teams focused in these areas to be allowed some autonomy to organize their work. Another complication in many government settings is that we are often re-architecting and re-designing existing systems.
At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.	This principle is an attempt to ensure that “lessons learned” are actually learned and applied rather than just being “lessons written.”

References

[Austin 1996] Austin, R. <i>Measuring and Managing Performance in Organizations</i> . New York: Dorset House Publishing. 1996.	In addition to providing excellent guidance, this book offers coherent advice on avoiding the misuse of measurement.
[Hayes 2013] Hayes, W. et al. <i>Agile Metrics: Progress Monitoring of Agile Contractors</i> . CMU/SEI-2013-TN-029. Software Engineering Institute, Carnegie Mellon University. 2014. http://resources.sei.cmu.edu/library/asset-view.cfm?assetid=77747	This SEI report provides initial guidance on issues to address when designing a program of Agile progress metrics in a government setting.
[Kennedy 2011] Kennedy, M. “Achieving Better Acquisition Outcomes in Austere Times.” Defense Acquisition University Symposium. April 2011.	This presentation from (at the time) DAU professor Matthew Kennedy, reviews relevant aspects of Agile development for government program offices.
[Lapham 2011] Lapham, M.A. et al. <i>Agile Methods: Selected DoD Acquisition and Management Concerns</i> . CMU/SEI-2011-TN-002. Software Engineering Institute, Carnegie Mellon University. 2011. http://resources.sei.cmu.edu/library/asset-view.cfm?assetid=9769	This SEI report addresses several topics of concern to acquisition professionals considering or already taking on adoption of Agile methods and principles. This report includes a chapter on cultural issues that commonly must be addressed in Agile adoption.
[Miller 2014] Miller, S.M. “The Readiness & Fit Analysis: Is Your Organization Ready for Agile?” Software Engineering Institute, Carnegie Mellon University. 2014. http://resources.sei.cmu.edu/library/asset-view.cfm?assetid=90977	This white paper summarizes a method that the SEI has developed to help government and other regulated organizations understand the adoption risks they may face when they undertake adoption of Agile methods and principles.
[Nielsen 2009] Nielsen, P. “Challenges to Effective Acquisition and Management of Information Technology Systems.” Invited testimony for the U.S. Senate Armed Services Committee. 2009. https://www.gpo.gov/fdsys/pkg/CHRG-111hrg51959/pdf/CHRG-111hrg51959.pdf	Dr. Nielsen’s testimony outlines the major challenges that the SEI has seen in supporting major software acquisitions over the previous twenty years.
[NIST 2010] Various authors. “Guide for Applying the Risk Management Framework to Federal Information Systems.” SP-800-37. National Institute of Standards and Technology. 2010. http://csrc.nist.gov/groups/SMA/fisma/framework.html	This guide provides organizations in the federal space with guidance on applying NIST’s Risk Management Framework, designed to provide a risk-based approach to cybersecurity threat management.
[OMB 2016] Various authors. Announcement of Agile Contracting Curriculum. 2016. http://www.icfi.com/contracts/featured/digital-service-contracting-professional-training-and-development-program	This program is meant to help acquisition professionals understand various aspects of Agile and lean concepts so as to better prepare contract materials in Agile settings.
[Rigby 2016] Rigby, D. et al. “Embracing Agile.” <i>Harvard Business Review</i> . May 2016.	This article provides a high-level summary of issues in leadership of an organization moving towards Agile. Although the examples go beyond IT transformation settings, they are still relevant to IT transformation.

<p>[USDS 2016] Various authors. United States Digital Service blog. https://www.usds.gov/blog</p>	<p>This site provides ongoing information about the work that U.S. Digital Service sponsors and engages in.</p>
<p>[Ward & Miller 2016] Ward, D. and Miller, S. <i>Update 2016: Considerations Using Agile in DoD Acquisition</i>, 2016-TN-001, Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA 15213</p>	<p>This is an update to the original TN that started the SEI Agile Adoption research. It adds in a case study and references 2015 version of DoD 5000.02 and its changes that better support Agile use in DoD settings.</p>

Selected SEI Agile in Government Resources

<p>Lapham, M. A. et al. "How Air Force Acquisition Can Effectively Adopt Agile Methods", Joint Mitre and SEI briefing to General E. Pawlikowski, April 8, 2016</p>	<p>This briefing, a joint effort between Mitre and SEI, provided General Pawlikowski with the organizations' perspective on barriers and enablers to Agile adoption in the USAF.</p>
<p>Carney, D., Miller, S. and Korzec, K. <i>Agile in Government: Myths, Monsters, and Fables</i>. Software Engineering Institute, Carnegie Mellon University, 2015.</p>	<p>This booklet provides acquisition professionals new to Agile methods with a set of common myths related to Agile methods and basic information to help them understand the potential effects of Agile on their work.</p>
<p>Lapham, M. and Miller, S. "Applying Agile in the DoD", podcast series, Software Engineering Institute, 2014-2015. http://www.sei.cmu.edu/podcasts/agile-in-the-dod/</p>	<p>This series of podcasts relates the experiences of the SEI researchers working with DoD programs using Agile to the twelve Agile principles.</p>
<p>SEI Home Page for Agile Adoption Research http://www.sei.cmu.edu/acquisition/research</p>	<p>This home page provides links to all the publicly available resources related to SEI's Agile Adoption in Regulated Settings research and transition efforts.</p>
<p>Palmquist, S. et al. <i>Parallel Worlds: Agile and Waterfall Differences and Similarities</i>, 2013-TN-021, Software Engineering Institute, Carnegie Mellon University, October 2013.</p>	<p>This report helps acquisition professionals accustomed to common US government acquisition language with a "cross-walk" between Agile terms and more common terms used in waterfall-style acquisitions.</p>

Acknowledgements

The Agile in Government team would like to provide a special thank you to two individuals at the SEI who have constantly championed and shepherded our Agile adoption research since its inception in 2009: John Foreman authorized the original USAF project that began our research in Agile adoption, and has been a consistent, stalwart champion of the work both inside and outside the SEI. Mary Ann Lapham was the original leader of the research and continued in that role until her upcoming retirement from the SEI. She has shaped the research and provided uncompromising support to the team of researchers involved in the Agile in Government research and transition work. Our sincere thanks to you both from the Agile in Government team.

For more information, please contact:

Eileen Wrubel

Senior Member of the Technical Staff

SEI Software Solutions Division

P: 412.268.9976

E: eow@sei.cmu.edu

Suzanne Miller

Senior Member of the Technical Staff

SEI Software Solutions Division

P: 412.268.9143

E: smg@sei.cmu.edu

William Hayes

Senior Member of the Technical Staff

SEI Software Solutions Division

P: 412.268.6398

E: wh@sei.cmu.edu

Alyssa Le Sage

Legislative Analyst

P: 703.247.1346

E: 412-297-9023

About

For four decades, the Software Engineering Institute (SEI) has been helping government and industry organizations to acquire, develop, operate, and sustain software systems that are innovative, affordable, enduring, and trustworthy.

Contact Us

Software Engineering Institute
4500 Fifth Avenue, Pittsburgh, PA 15213-2612

Phone: 412.268.5800 | 888.201.4479

Web: www.sei.cmu.edu | www.cert.org

Email: info@sei.cmu.edu