

SEI Virtual Learning Package 4

Getting Insight (not just Oversight!) into Contractor Progress

May 2020
SEI Continuous Deployment of Capability Directorate

Copyright 2020 Carnegie Mellon University.

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8702-15-D-0002 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

The view, opinions, and/or findings contained in this material are those of the author(s) and should not be construed as an official Government position, policy, or decision, unless designated by other documentation.

References herein to any specific commercial product, process, or service by trade name, trade mark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by Carnegie Mellon University or its Software Engineering Institute.

NO WARRANTY. THIS CARNEGIE MELLON UNIVERSITY AND SOFTWARE ENGINEERING INSTITUTE MATERIAL IS FURNISHED ON AN "AS-IS" BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT.

[DISTRIBUTION STATEMENT A] This material has been approved for public release and unlimited distribution. Please see Copyright notice for non-US Government use and distribution.

This material is distributed by the Software Engineering Institute (SEI) only to course attendees for their own individual study.

Except for any U.S. government purposes described herein, this material SHALL NOT be reproduced or used in any other manner without requesting formal permission from the Software Engineering Institute at permission@sei.cmu.edu.

Although the rights granted by contract do not require course attendance to use this material for U.S. Government purposes, the SEI recommends attendance to ensure proper understanding.

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

DM20-0409

Meeting Conventions for Today

Please stay on mute for the lecture portion of the course module. If all goes according to plan, you will be muted automatically when you come into the Skype meeting (both web and dial-in participants).

If you are “in” the Skype meeting via web or app, please ask questions via the Chat window.

- A facilitator will collect the questions and either pass them to the facilitator if something immediate, or organize them for the Q&A portion of the course module

Those on dial in will enter questions via email to Jon Wood -- Jonathan.Wood.ctr@jsf.mil

Instructor will call for participation and discussion at various points. Please remember to come off mute before talking.

When you are done talking, before going back on mute, please say “Over” so others know you are finished.



Topics the SEI will address in this course module include:

- **A Notional Manifesto for Agile Acquisition**
- **Traditional Things We Think About Differently**

Bottom Line Up Front

There is a great deal of OVERSIGHT activity that is required by the traditional DoD acquisition eco system to progress a program, software or otherwise

- Many of the mechanisms used for acquisition oversight could be seen as substitutes for the communication that naturally occurs in a trust-based relationship typical of Agile settings
- Regardless of the informal communication on the program, required oversight has to be accomplished
 - Recent developments like Adaptive Acquisition Pathways change some of those requirements

The other goal for contract monitoring is to achieve INSIGHT into the program

- Acquisition CDRLS and required events are not always the best way to achieve insight
- Agile development settings, in particular, promote transparency and have built in mechanisms for achieving ongoing insight
 - These mechanisms, however, require proactive participation from the acquirer to be effective

Oversight/Insight

Oversight and insight are closely coupled concepts, in the same manner as verification and validation.

- Oversight works to continually assure that the project (internal or external) is:
 - following an appropriate process to the work at hand
 - providing sufficient evidence that the process, as executing, can feasibly deliver target capabilities within available resources
 - providing evidence that the appropriate standards are being met
 - managing the capability requirements and that they remain needed and feasible
- Insight works to continually assure that the project (internal or external) is:
 - correctly understanding the requirements and specifications associated with the environment
 - providing adequate user participation to support validation activities and prevent rework
 - providing valid feasibility evidence for oversight
 - executing an appropriate process
 - periodically reviewing its process to establish appropriateness, and evolves as necessary
 - managing the relationships and internal/external communications and interoperability within the project and with other projects

How We See the Need for Insight Depends on our World View

We have sought compliance, in the past, because we want to hold the contractor accountable for every last thing they said they were going to do;

- We had an adversarial mindset and believed we were getting insight because we saw compliance with the schedules.
- In part, that was enabled by the assumption that the world (contract) didn't change except through carefully reviewed processes (ECPs) that enabled us to adjust to a new stable world view.

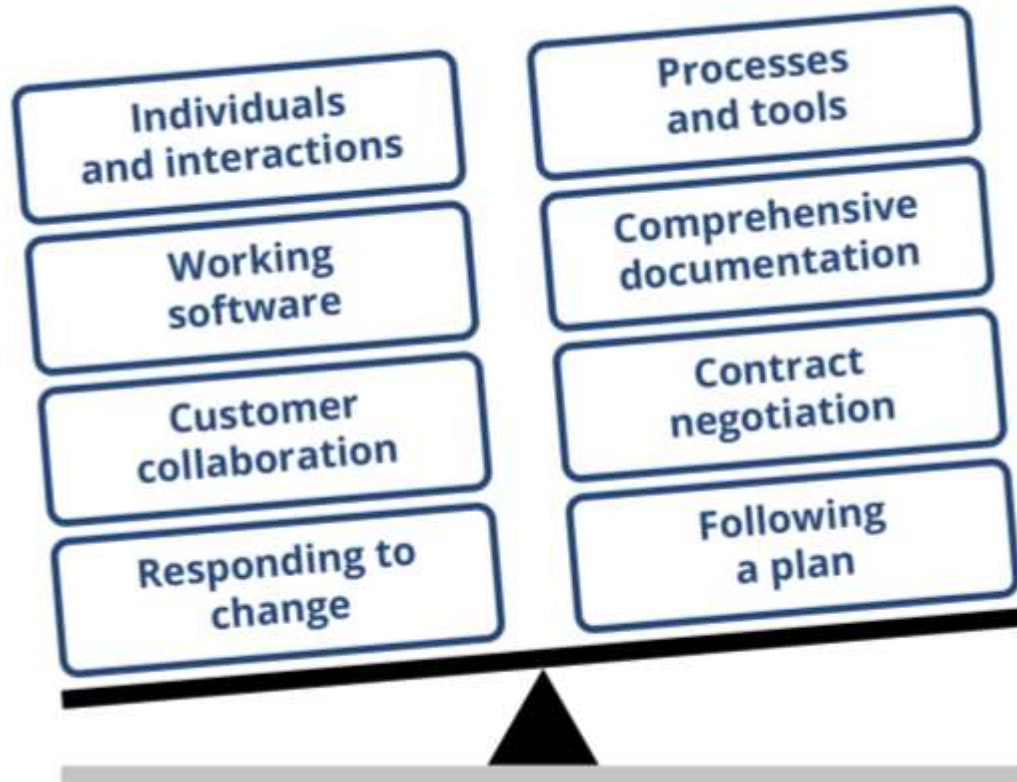
Today, we're looking for insight and understanding of the program largely because

- Compliance has never worked
- We're trying to be more collaborative
- Our understanding is that the only constant in the world is change
 - If the ground is continually shifting then there's nothing the contractor can comply to, so, we seek insight into what's happening so that we know things are proceeding in a direction that is productive

Back to the Principles...Agile and Lean

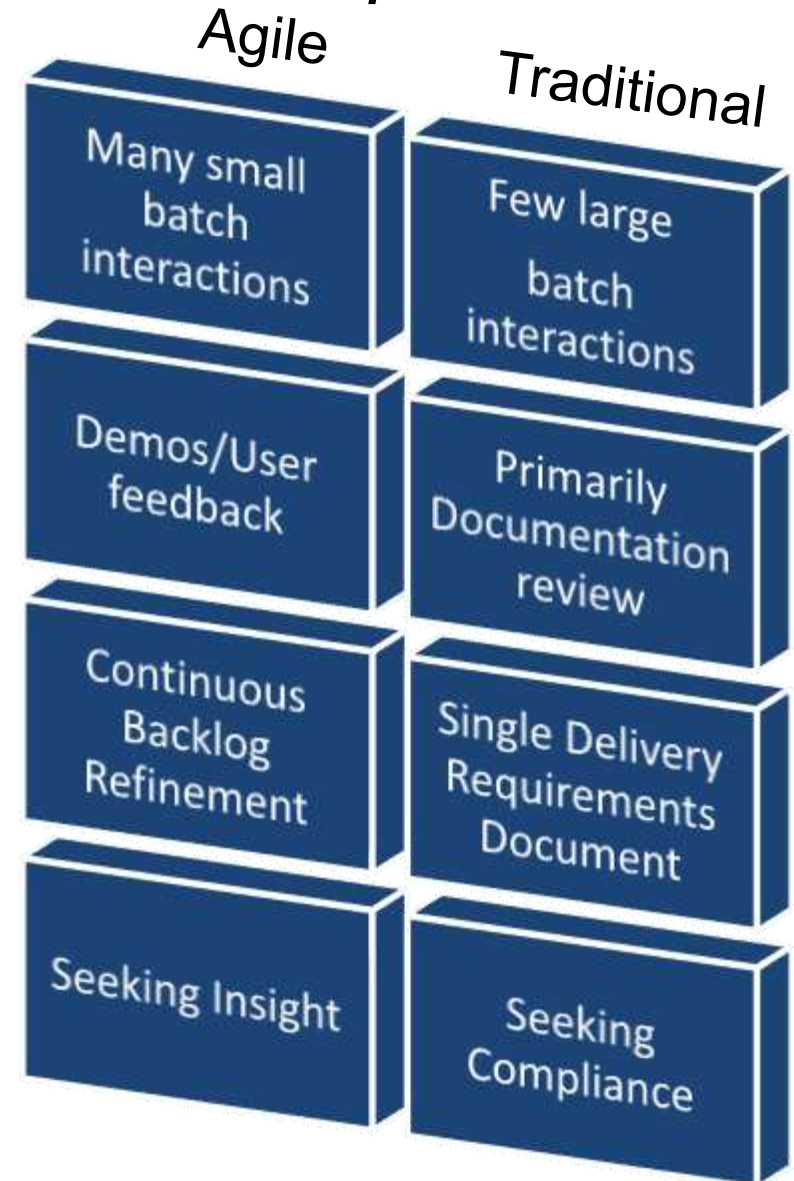
Reorienting the Manifesto for Agile *Software Development* Toward *System Acquisition*

Through this work we have come to value:



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

<https://agilemanifesto.org/history.html>



Principles from the Agile Manifesto for Developers-1

1. Highest priority is satisfy the customer through early and continuous delivery of software.
2. Welcome changing requirements, even late in development...
3. Deliver working software frequently, from a couple of weeks to a couple of months...
4. Business people and developers must work together daily throughout the project.
5. Build projects around motivated individuals. Provide environment and support they need...
6. The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.

Principles from the Agile Manifesto for Developers – 2

7. Working software is the primary measure of progress.
8. Agile processes promote sustainable development...a constant pace indefinitely.
9. Continuous attention to technical excellence and good design enhances agility.
10. Simplicity—the art of maximizing the amount of work not done—is essential.
11. The best architectures, requirements, and designs emerge from self-organizing teams.
12. At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

Adapted from <http://agilemanifesto.org/principles.html>

Don't forget the Lean Principles – For the Entire (govt + contractor + stakeholder) Enterprise

Apply SAFe Lean-Agile Principles

#1 Take an economic view

#2 Apply systems thinking

#3 Assume variability; preserve options

#4 Build incrementally with fast, integrated learning cycles

#5 Base milestones on objective evaluation of working systems

#6 Visualize and limit WIP, reduce batch sizes, and manage queue lengths

#7 Apply cadence, synchronize with cross-domain planning

#8 Unlock the intrinsic motivation of knowledge workers

#9 Decentralize decision-making

#10 Organize around value

SCALED AGILE™ © Scaled Agile, Inc.

29

The Oversight/Insight Challenge for the Lean Program Office

*How can the Program Office gain
Insight into Contractor Expression
Of the Agile & Lean Principles?*

As with last week's materials, we are just hitting the tip
of the iceberg on these concepts!

From Large Batch to Small Batch

Typical large batch behaviors/mindsets: Typical small batch behaviors/mindsets:

- “Nothing is done until everything is done”
 - More Work in Progress is good
 - 100% utilization of resources is a goal
 - Tendency to hide bad news
 - -False/optimistic reporting of progress in order to justify incentive/progress payments
 - Integration events are riddled with defects and are pushed out “until we think we have it right”
 - Increases number of potential defects that affect multiple areas of the system
 - Reduces confidence in system robustness
 - Harder for engineers to find sources of defects
 - Tendency toward “test quality in”
- We can learn from even small pieces being implemented/done
 - “Stop starting, start finishing”
 - Work in Progress is limited to enhance flow through the system
 - 100% utilization of resources is recognized as limiting flow, flexibility, and work accomplishment
 - Short time between when a defect is found and when it was created
 - Easier for engineering/developer to find source of defect
 - LOTS of integration happening across entire system, building confidence
 - Tendency to “build quality in”

From Primarily Documentation Review to Demos and other Mechanisms for User Feedback

Typical “Primarily Document Reviews” Behaviors/Mindsets:

- Preference for larger, more infrequent demos
- Spotty participation in demos
- Requirements documents seen as “ground truth” for user needs, even when known to be superseded
- Few opportunities for feedback
- Incomplete, rushed feedback on documents
- More emphasis on “to be” documents than “as built” documents
 - Using documents to “lock down” design

Typical “Demos/Other Feedback Mechanisms” Behaviors/Mindsets:

- Recognition that demo doesn’t EQUAL test, but INFORMS it
- Active participation in demos of small pieces of functionality
- Open, continuous feedback about both the fact of and the meaning of progress or lack thereof
- Info from demos is fed forward to testing and certification staff to ensure alignment
- -Uses def of done that includes certification criteria (cyber, DT/OT, ATC, ATO, etc.)
- Participation on continuous integration team by govt staff seen as a high priority

From Single Delivery of Requirements Document to Continuous Backlog Refinement

Typical “Single Delivery” Behaviors/Mindsets:

- Long lead time to get to the requirements document delivery reduces motivation to allow for refinement after delivery
- Task-switching from one large batch review to another
 - Hard to take in the large requirements set
 - Demotivates “digging in” on the need behind the requirements
- Get as far as we can with review in time allowed, but not expecting complete understanding in time allowed

Typical Continuous Backlog Refinement Behaviors:

- Mix of “push” and “pull” communication across govt/contractor interface on evolving refinements to requirements
 - Facilitated by workflow mgmt. tools like Jira, but both sides need to be on the same platform
- Frequent face to face/high bandwidth meetings to keep the relationship going, not just to do the refinement tasks
- Transparency among stakeholders that builds trust
- Frequent small batch prioritizations build a solid base of understanding of current state and progress

From Seeking Compliance to Seeking Insight

Typical “Seeking Compliance” Behaviors

- Deadlines that don’t have clear relationship to product evolution goals
- Silo’ed relationships
 - Independence=Isolation
 - Formal vs informal handoffs of information
- Agile events must conform to traditional Program events
 - Large batch SETR events
 - PMRs are disconnected from development cadence and cause value-based work to stop-start
- No/ too little sharing of test assets – “if you know the criteria, you’ll develop to it; I lose my independence!”
- Measures collected but not used for process improvement
- “Gotcha” mindset
- Work designed to pass the audit more than deliver value

Typical “Seeking Insight” Behaviors

- Don’t immediately react negatively to “bad news” – treat it as information that is meant to help make a different decision
- Informal handoffs of information where feasible/allowed
- Agile events allowed to preserve their cadence
- Lots of sharing of test/certification assets – “if you know the criteria, you’ll develop to it; that’s the goal!!!”
- Measures carefully selected and visibly used to solve problems in the process, not punish the contractor
- Collaborative mindset

Traditional Things We Still Need to Think About...but Differently

We Will Have Separate Modules on Each of These Topics

BLUF for each of these addressed briefly:

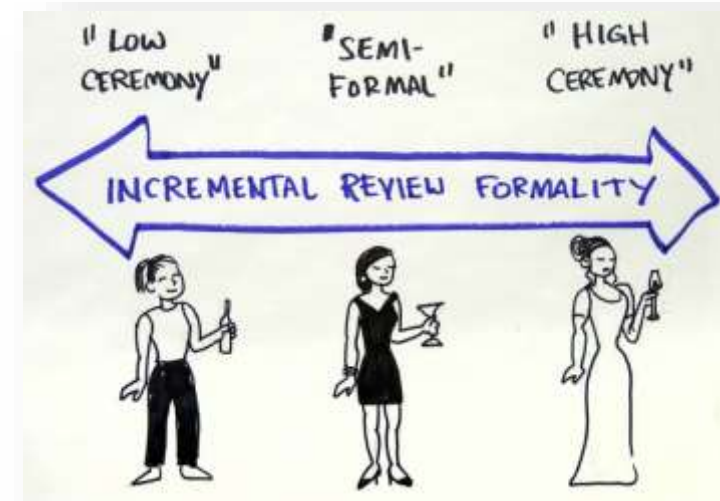
- Technical Reviews
- Requirements
- Systems Engineering
- Contracting
- Measures
- Testing
- Certification

Technical Reviews—How Do They Work in Agile Settings?

Agile View of Formal Technical Reviews

Agile in Government settings – have to recognize that changing the acquisition life cycle expectations doesn't happen overnight. Some accommodations have been successfully used

- Small batch events (around 12 weeks of work, typically) that reduce risk by producing working software/product, as well as required documentation
- Exit criteria for formal large-batch reviews incorporated into the small batch events
- Depending on system context, traditional System Requirements Review and even System Preliminary Design Review may occur more traditionally



Focus of gaining understanding by learning from as built software vs projective documents

Requirements in Agile Government Settings

From Single Large Batch Requirements Document to Roadmaps and Variable Detail Backlogs

“Fixed” intent assume stability

- Higher level of requirements baseline needed to allow learning to occur
- Roadmaps guide what needs to be specified in more detail
- Roadmaps guide when higher level requirements need to be detailed

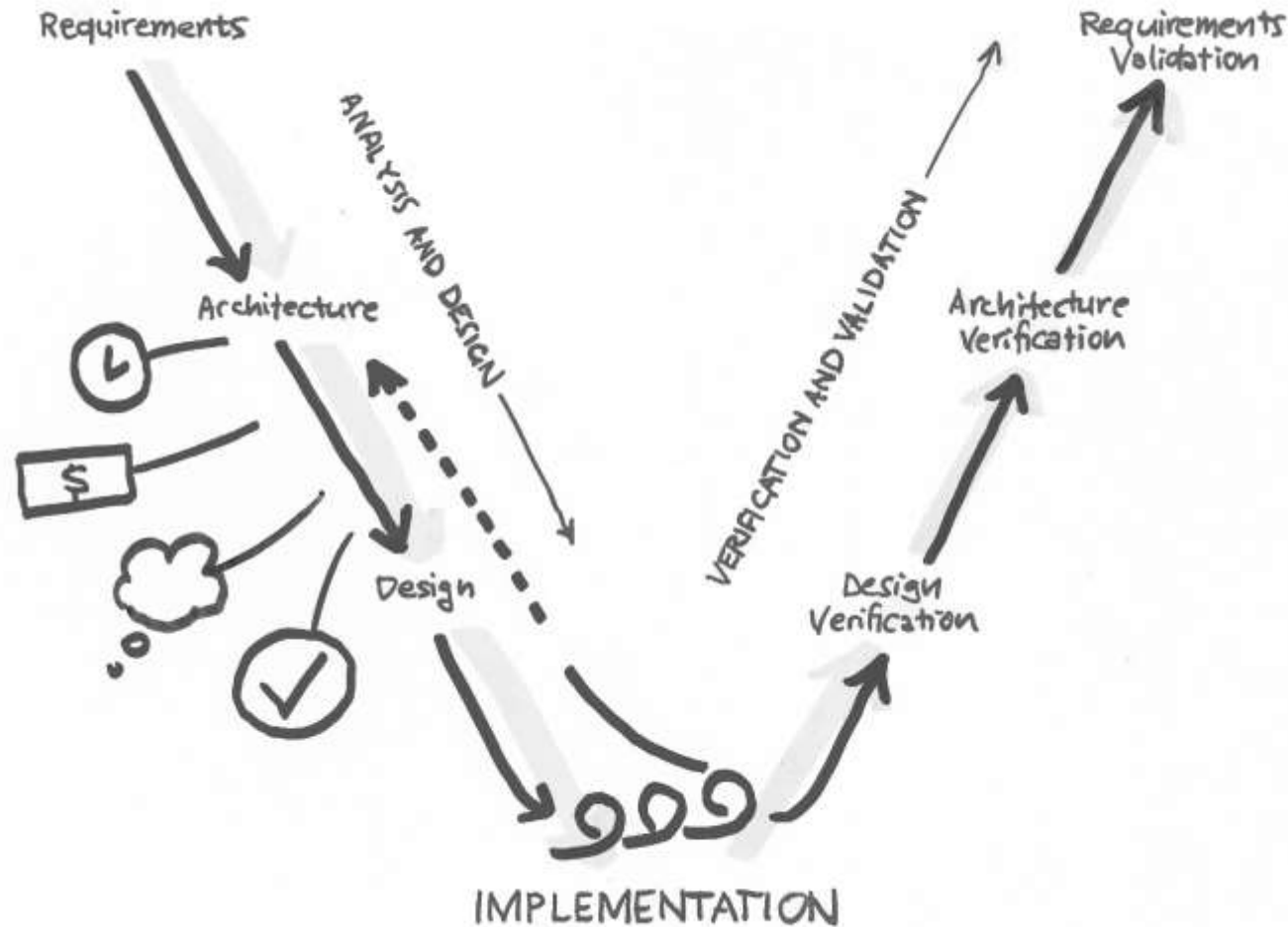
“Variable” intent assumes learning and evolution of the environment

- The trade space where requirements backlogs are reprioritized continually
- Allows focusing on the highest risk, most valued items earlier



Systems Engineering in Agile Government Settings

What Happens When “Large Batch” Systems Engineering Meets “Small Batch” Agile SW Development?



Contracting in Agile Government Settings

Properties of Successful Agile Project Teams

1. Frequent delivery
2. Reflective improvement
3. Close/Osmotic Communication
4. Personal Safety (free to speak without fear of reprisal)
5. Focus (knowing what to work on, having time to work on it)
6. Easy access to expert users
7. Sunshine/Visibility (no dark places in the project) (courtesy Sam Person, Overstock.com)
8. Technical environment

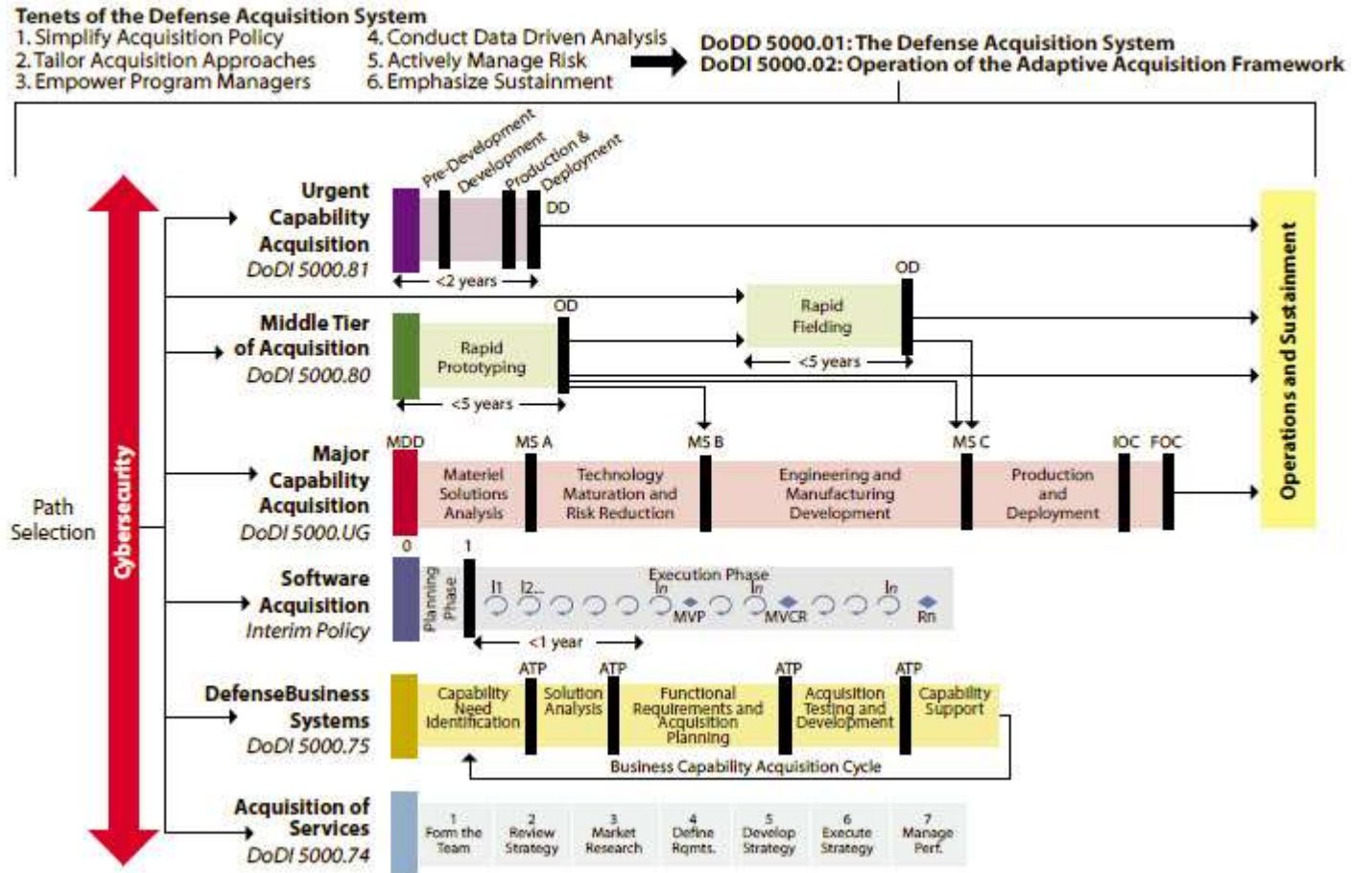
Crystal Clear: A Human-Powered Methodology for Small Teams, Alistair Cockburn, 2005 Pearson Ed cc 19 - 37

The contracting challenge is how to be aware of, and incentive these and other Agile/Lean attributes we've discussed

This is an Area Changing More than some Others

Adaptive Acquisition Pathways will be addressed in our Agile & Contracting module

Figure 3. Adaptive Acquisition Framework



Measuring Progress in Agile Government Settings

The Consistent Measurement Challenge is the Multi-Variable Nature of Causality

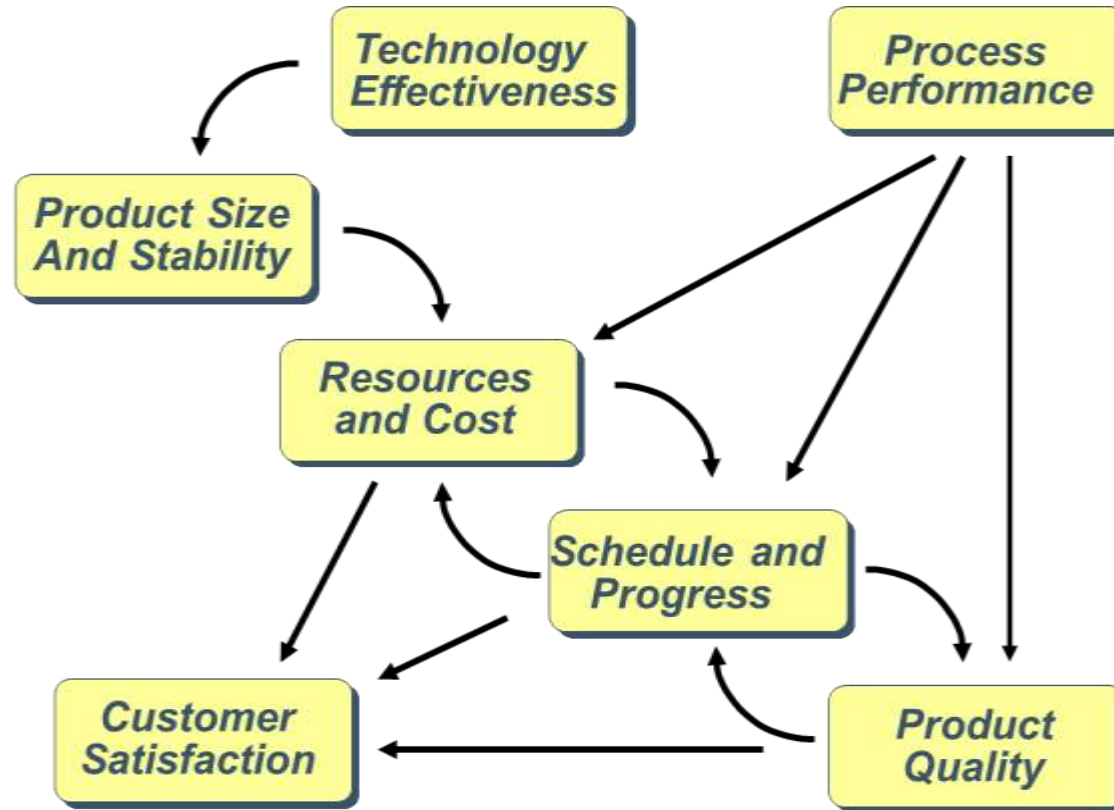
Model performance & tradeoffs that drive tactical choices.

Understanding cause-and-effect.

A pre-requisite to acting on progress measures.

Agile & Lean thinking drives to new areas of focus.

- Flow vs utilization
- Value
- Concept to Capability

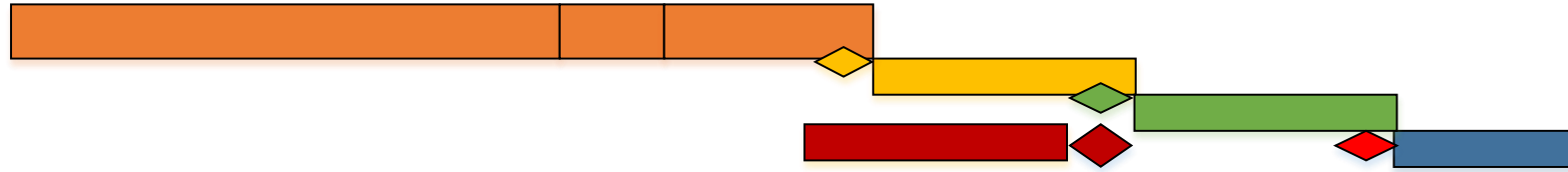


Graphic Adapted from: *Practical Software & Systems Measurement*
<http://www.psmc.com/>

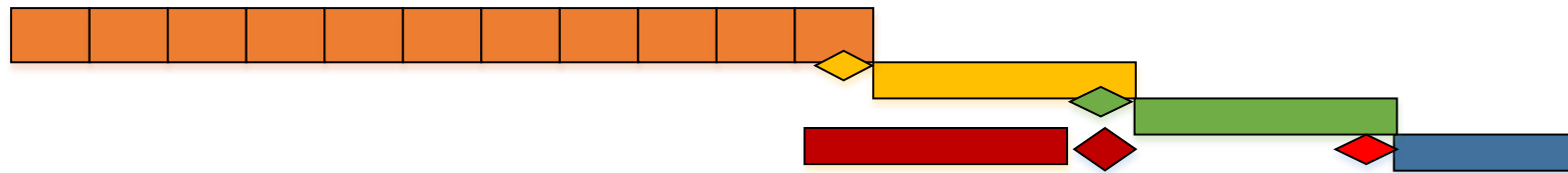
Testing in Agile Government Settings

Left-shift with Agile Testing

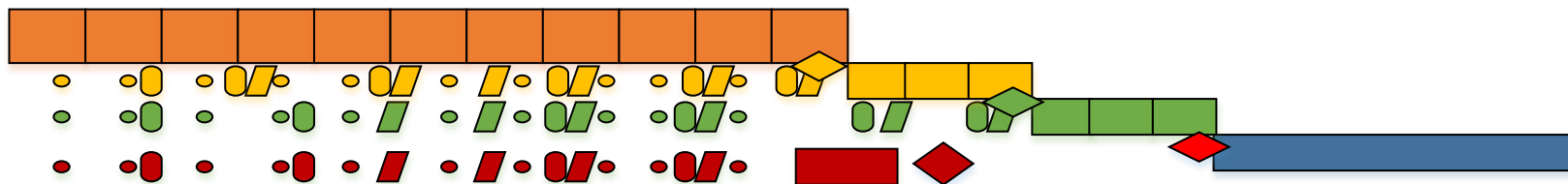
Traditional Vee-process



Agile development with traditional DT and OT (Hybrid)



Agile development with traditional DT and OT, early integration synch points



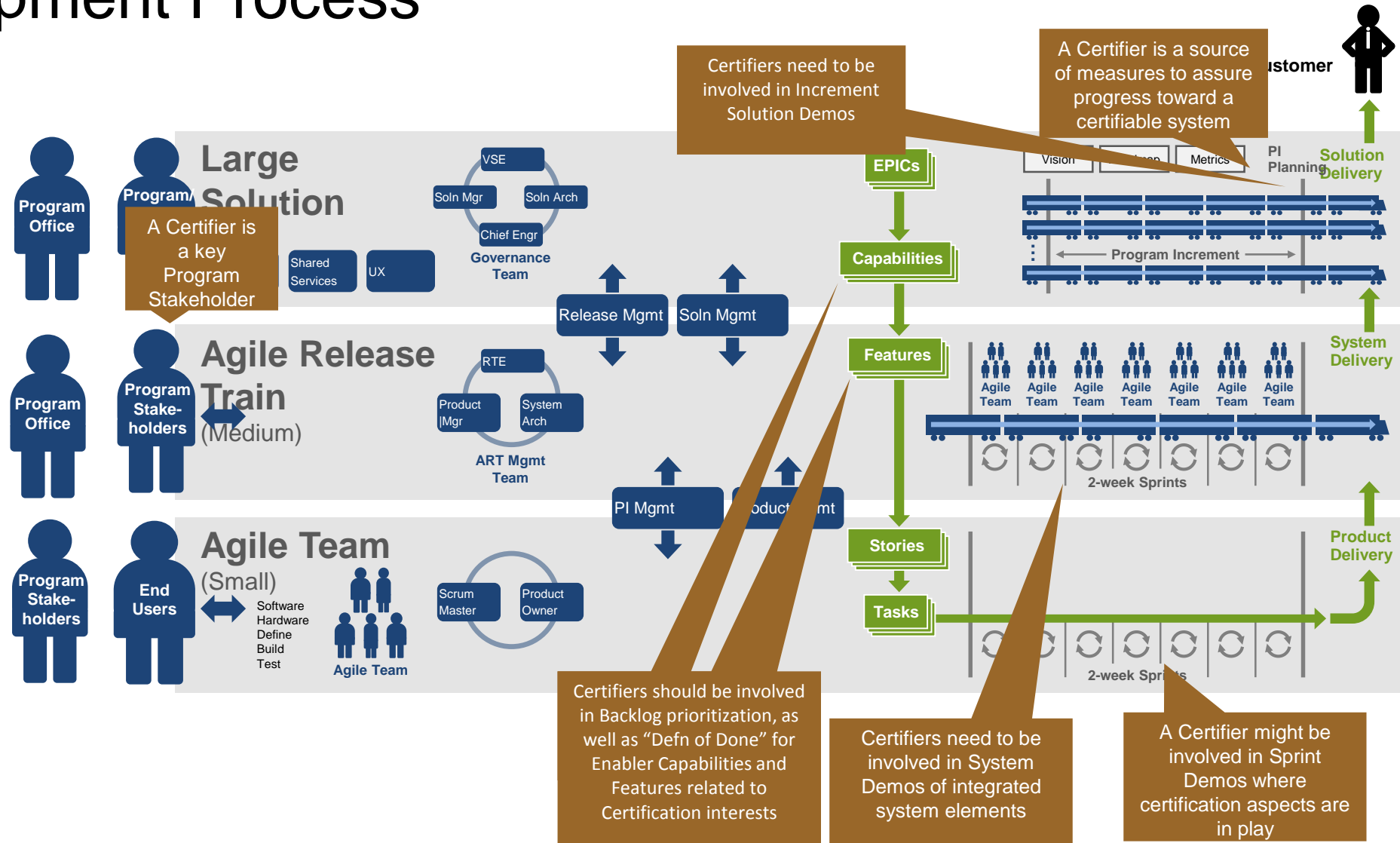
Moving from phased and siloed testing to Agile testing is the “Big Deal”

Integrating Agile cadence with DT/OT is a key challenge



Certification (Cyber, Airworthiness, Nuclear Surety...) in Agile Government Settings

Certifiers Need to be Involved Throughout an Agile/Lean Development Process



Summary

Moving from Oversight to Insight is a Big Shift for Many Program Offices

Changes in skill profiles

Changes in staffing curves

Changes in character of interactions with contractors and stakeholders

Changes in batch size

....

The payoff: faster delivery of certified, high value solutions to warfighters and our other stakeholders.

Contact Information

Suzanne Miller

Principal Researcher

SSD/CDC

Email: smg@sei.cmu.edu

U.S. Mail

Software Engineering Institute

Customer Relations

4500 Fifth Avenue

Pittsburgh, PA 15213-2612

USA

Will Hayes

Initiative Lead

SSD/CDC

Email: wh@sei.cmu.edu

Customer Relations

Email: info@sei.cmu.edu

Telephone: +1 412-268-5800

Fax: +1 412-268-6257