You're In My Space: Agile's Roles, Responsibilities and Competencies

Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213

Mary Ann Lapham / Suzanne Miller
March 4, 2015

© 2014 Carnegie Mellon University.
Published by The Aerospace Corporation with permission.

Copyright 2015 Carnegie Mellon University

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the United States Department of Defense.

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.

This material has been approved for public release and unlimited distribution except as restricted below.

This material was prepared for the exclusive use of GSAW/SPIN/INCOSE-LA Event Mar 4, 2015 and may not be used for any other purpose without the written consent of permission@sei.cmu.edu.

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

DM-0002135

© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

Agenda

Set the Stage

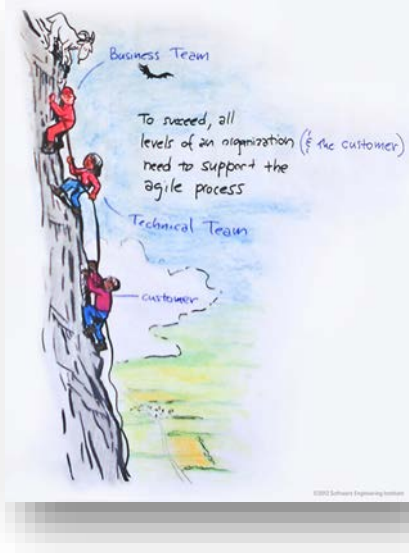
Common Questions

PMO Roles, Responsibilities, Competencies

New 5000.02

Eco-System

Practical Ideas



© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

SET THE STAGE

© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

Audience—Who are You?

What role do you have in software acquisition?

- ***Engineer***
- ***PM***
- ***Budget Staff***
- ***Contracting Staff***
- ***[your role here]…***

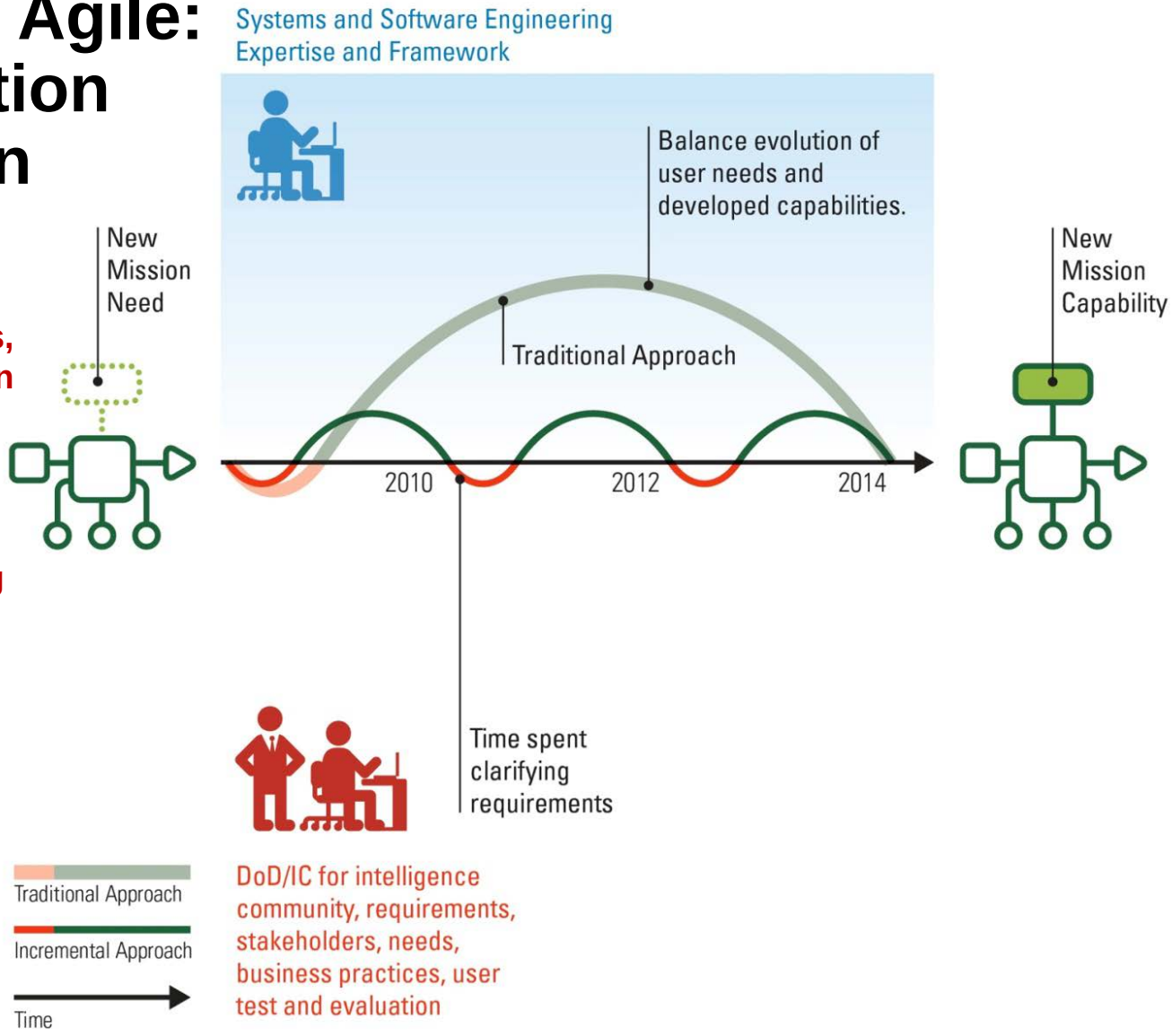
© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

© 2014 Carnegie Mellon University. Published by
The Aerospace Corporation with permission.

Motivation for Agile: Gov't Acquisition and Innovation

Many regulated environments, like the DoD, NEED innovation and NEED incremental improvements to their systems.

Many of them are now willing to consider changing their approach *if they can do it without getting in trouble with their governing statutes and regulations.*



© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

Agile Manifesto—Foundation of Agile Software Development

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**.

Common myth:

The manifesto is often misinterpreted to mean:

no documentation, no process, and no plan!

http://www.agilemanifesto.org/

© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

Agile Principles Accompanying the Manifesto$_1$ – All are important aspects of building an Agile culture

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.

© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

Agile Principles Accompanying the Manifesto$_2$ – All are important aspects of building an Agile culture

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

© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

What are the tenets and principles trying to enable?

1. Increase speed of development of high business value software
   - Speed=the time it takes to move ideas from one person to another
   - A reason that all Agile approaches emphasize face-to-face contact as much as possible
2. Take advantage of the benefits of small batch sizes
   - Fast feedback, fast learning, etc
   - What is the next smallest thing I can do that increases the business value delivered?
3. Reduce non-value-added work and rework
   - Reduce “time to fielding”
   - BUT understanding what is truly non-value added
     - Non-value for designer may be HIGH value for sustainer

© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

Shu Ha Ri

Shu – Learn a technique (beginners)

Ha – Collect techniques (intermediates)

Ri – Blend / invent techniques (experienced, masters)

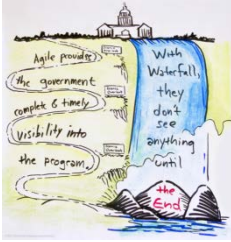
See http://alistair.cockburn.us/Shu+Ha+Ri

© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

COMMON QUESTIONS

© 2014 Carnegie Mellon University. Published by
The Aerospace Corporation with permission.

Selected Insights from Published Work[1]



The four top questions SEI is asked by government (potential) adopters:

- How do we accommodate traditional technical reviews like PDR, CDR?
- How do we effectively write solicitations that enable, but don't require, Agile?
- How do we write requirements at the right level to have a viable technical baseline while enabling the incremental learning that is important to Agile successes?
- How do we translate our Agile work products, metrics, etc into the traditional acquisition terminology we use to report upwards?

© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

Selected Insights from Published Work$_2$

Technical Reviews: *

- Identified three general approaches being taken
  - Stay with traditional use of milestones
  - Participate in progressive reviews throughout iterations
  - Technical staff participates in progressive reviews and milestones become management level review

Contracting:**

- Contract vehicle (FP, T&M, CPIF, IDIQ) not itself an impediment when contracting personnel aware of Agile methods

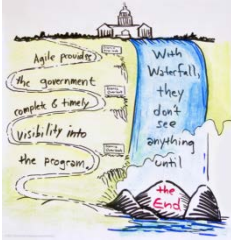
*** Agile Methods and Request for Change (RFC): Observations from DoD Acquisition Programs**

http://resources.sei.cmu.edu/library/asset-view.cfm?assetid=77732

**** Agile Methods: Selected DoD Management and Acquisition Concerns**
http://www.sei.cmu.edu/library/abstracts/reports/11tn002.cfm?DCSext.abstractsource=SearchResults

© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

Selected Insights from Published Work$_3$



Requirements Management:***

- Barriers and Potential solutions
  - DoD Guidance to track progress – waivers, higher level of abstraction
  - Translation of progress measures – Agile EVM
  - Risk adverse culture – move toward more trusting culture using IPTs
  - Work Breakdown Structure – consider alternate structure: task order perhaps
  - Effect of Requirement Change on Contracts – software is service to larger system
  - Reduced documentation – training, identifying what is "just enough"

Agile→Traditional Crosswalk:****

- A "Rosetta Stone" between Agile concepts and traditional acquisition life cycle concepts and terms

***** Potential Use of Agile Methods in Selected DoD Acquisitions: Requirements Development and Management**

http://resources.sei.cmu.edu/library/asset-view.cfm?AssetID=89158

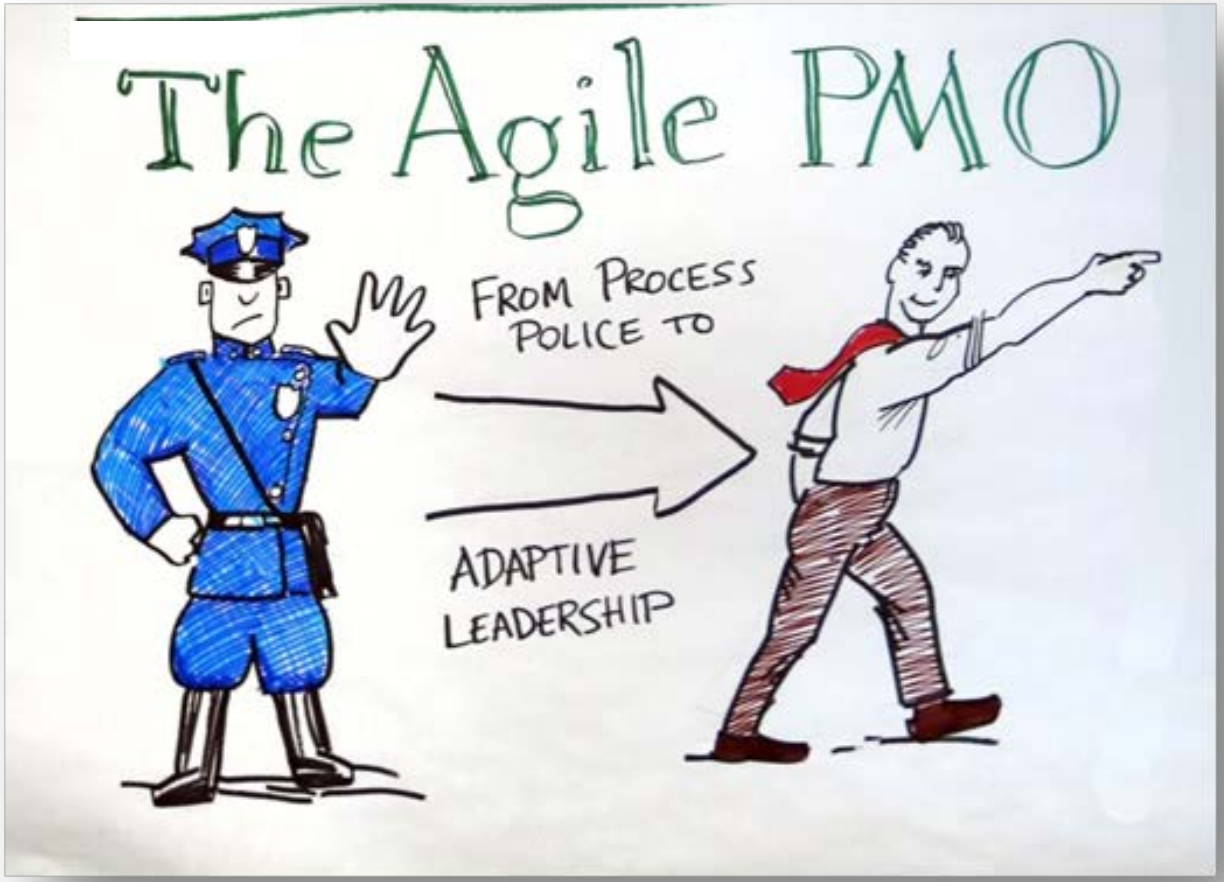
****** Parallel Worlds: Agile and Waterfall Differences and Similarities (CMU/SEI-2013-TN-021)**

http://resources.sei.cmu.edu/library/asset-view.cfm?AssetID=62901

© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

PMO ROLES, RESPONSIBILITIES, COMPETENCIES

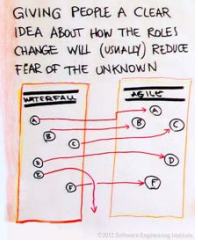
© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

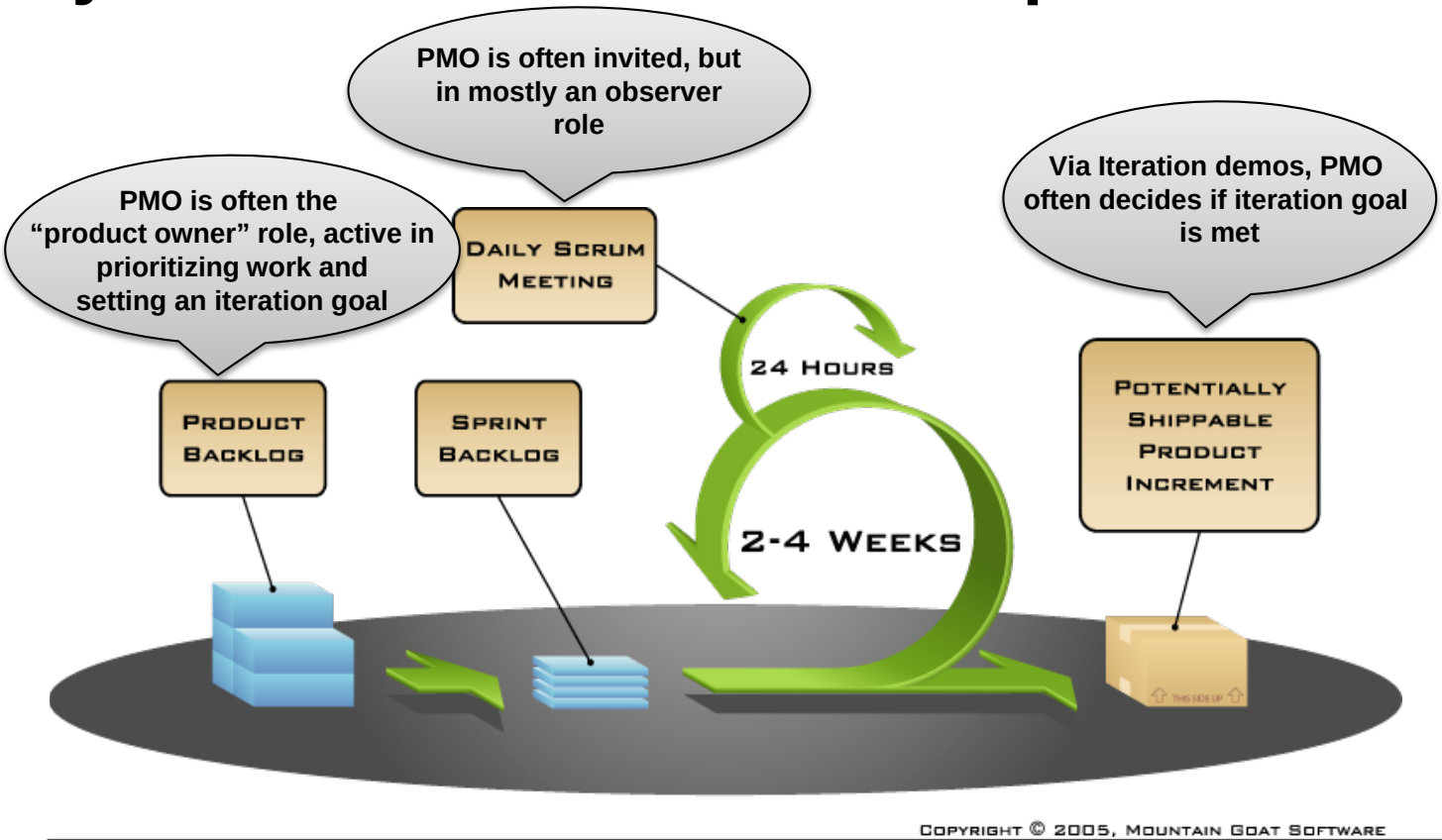


SEI Agile Collaboration Group Colloquium

© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

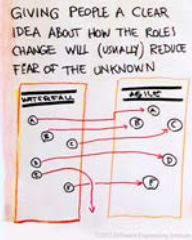
"Product Owner" is a New Role at Development Team Level—Not Used This Way in all Government Development Settings





Battle rhythm philosophy for Agile is: more frequent interaction, but on smaller chunks of content

© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.



The Big Challenges for Government Program Office as Product Owner

TIME!

- Moving from less frequent, higher content interactions to more frequent, lower content level interactions

Many program office staff support more than one program

- Engaging in an Agile program can upset the rhythm of interactions across others

Navigating the “constructive change” landscape when interactions are frequent can be challenging

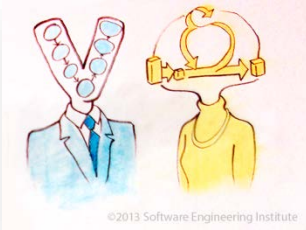
Representing all stakeholders as a single voice at all times (in real time)

On larger programs, having enough staff to interact with all the teams in play is challenging

The payoff: earlier course corrections lead to products with less rework

© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

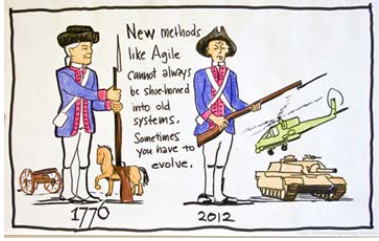
Several Traditional Viewpoints Change When Adopting Agile Methods



"Traditional at its Worst"	"Agile at Its Best"
Plan the work—especially the budget, schedule, and deliverables—to the maximum extent possible before beginning any design or code.	• Near-term plans contain more detail, while plans further out on the time horizon contain fewer details. • The overall vision is broken down into a roadmap, which is further broken down into release plans, which are further broken down into sprint or iteration plans, which are further broken down into daily plans. • Requirements are rank ordered, not just prioritized as Hi/Med/Lo. • Cost and schedule estimates are prepared for each capability at a high level. Relative estimation versus absolute estimation is employed. • Frequent planning sessions (at the beginning of each iteration) result in detailed, high-fidelity plans. • Risks are assessed and risk mitigation influences planning.
Lock down requirements to prevent gold-plating and scope creep.	• No requirements can be added to an iteration once it has started. • New requirements are evaluated by the stakeholders and prioritized thus preventing gold-plating and scope creep.
Institute multiple reviews to provide senior leadership oversight as well as to serve as gates for continued work.	• The customer is involved in all aspects of planning and testing. Customer (in the form of the product owner) is involved daily. • There are reviews at the end of each iteration that serve as gates to further work.
Move forward in a step-by-step, sequential manner and only when all parts of the previous steps were complete.	• The code base is integrated and tested daily. • The code base must pass all tests before and after integration. Regression testing is typically done each night.
Capture all details with extensive documentation.	• There is an overall plan. • There are requirements descriptions. • There are cost and schedule estimates. • There are risk assessments. • There is training material (as appropriate). • There is documentation (as appropriate). • There are lessons learned (based on retrospectives).

© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

Some Challenges: Agile in DoD Settings



DoD Guidance

- Interim DoD 5000.02 helps, but stops short of encouraging Agile

Translation of Requirements Progress Measures

- Progress=document completion is problematic in Agile settings

Risk Averse Culture

- "build to the requirements"-safe, but doesn't account for inevitable learning

Work Breakdown Structure

- Especially HW-centric WBS can result in software requirements at too low a level of detail

Effect of Requirements Changes on Contracts

- Assumption that change is an exception vs. change is expected and planned for

Perception that Reduced Documentation is Cause for Concern

- Letting go of "documents=progress" is difficult

Nidiffer, K. Miller, S. & Carney, D. *Potential Use of Agile Methods in Selected DoD Acquisitions: Requirements Development and Management* (CMU/SEI-2013-TN-0006), September 2013.

© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

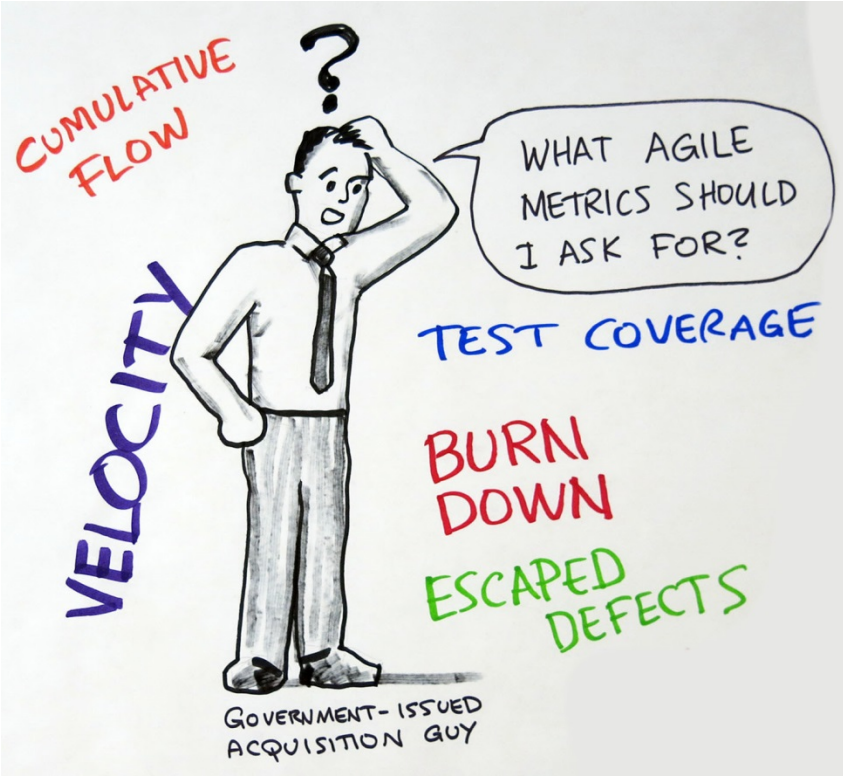
Integrating Agile Measurement into Your Overall Management Metrics

Agile/Iterative measures for teams leverage the approach of time-boxing (fixing the schedule for the iteration, and varying the target requirements to be fulfilled based on the team's capacity and capability)

- Opposite of traditional acquisition where schedule is based on estimates of what it takes to implement a fixed set of requirements

Agile/iterative iteration measures are meant primarily for the team's use

- External use of *iteration* metrics is generally discouraged
- *Release* metrics are generally used for monitoring/management purposes



© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

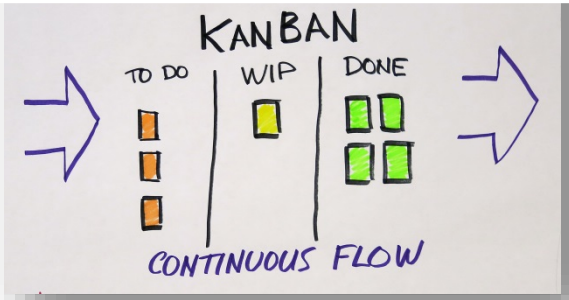
Typical Metrics for Agile Development

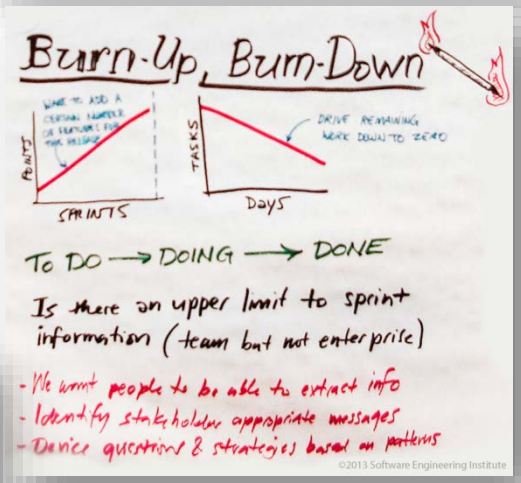
Metrics used by and for the development team

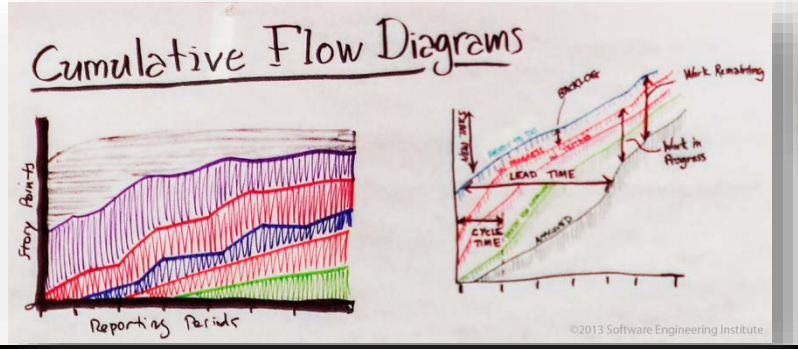
- Kanban Board for Task Tracking
- Sprint Burn-Down Charts
- Release Burn-Up Charts
- Velocity Tracking

Metrics for team and release level insight

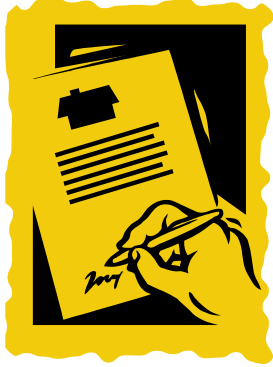
- Cumulative Flow Diagrams
  - Basic Construction
  - Tell-Tale Signals







© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

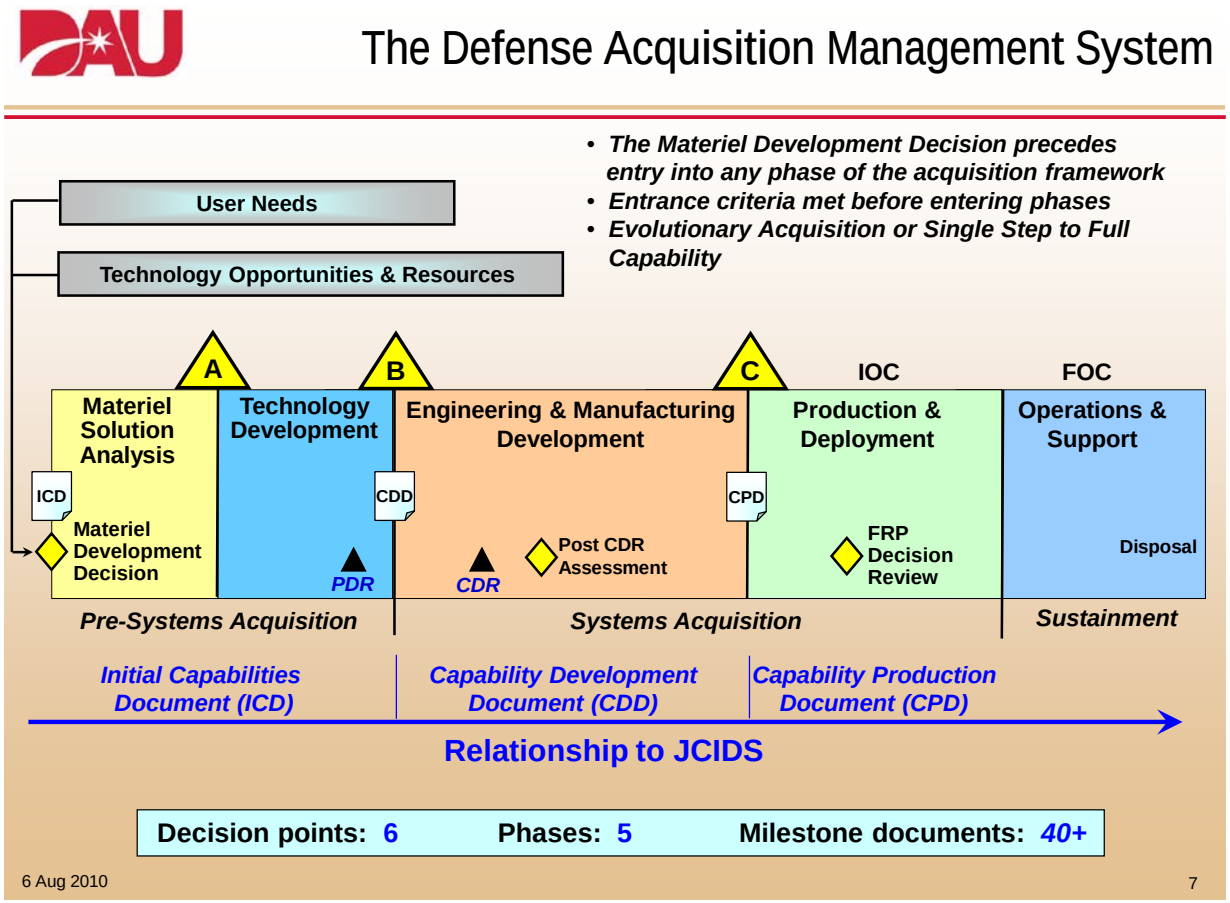


NEW 5000.02

© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

Overarching Framework for System Acquisition

Some things at the "next level down" have more explicit variety in the new DoD 5000.02 published in January 2015.

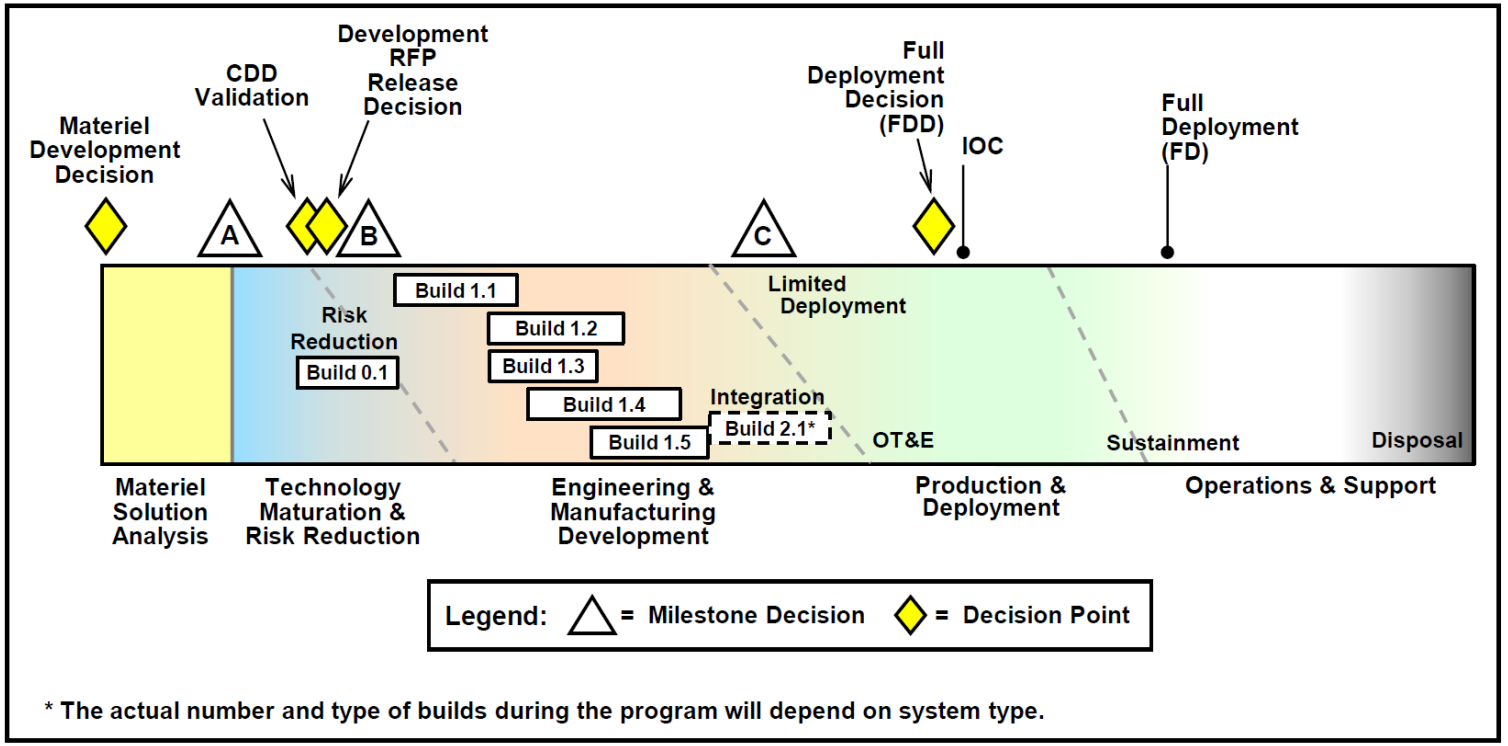


Source: Palmquist, Steve, et al. Parallel Worlds:

© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

Recent Changes in DoD Acquisition Guidance Explicitly Address IT Systems-1

Figure 4. Model 2: Defense Unique Software Intensive Program



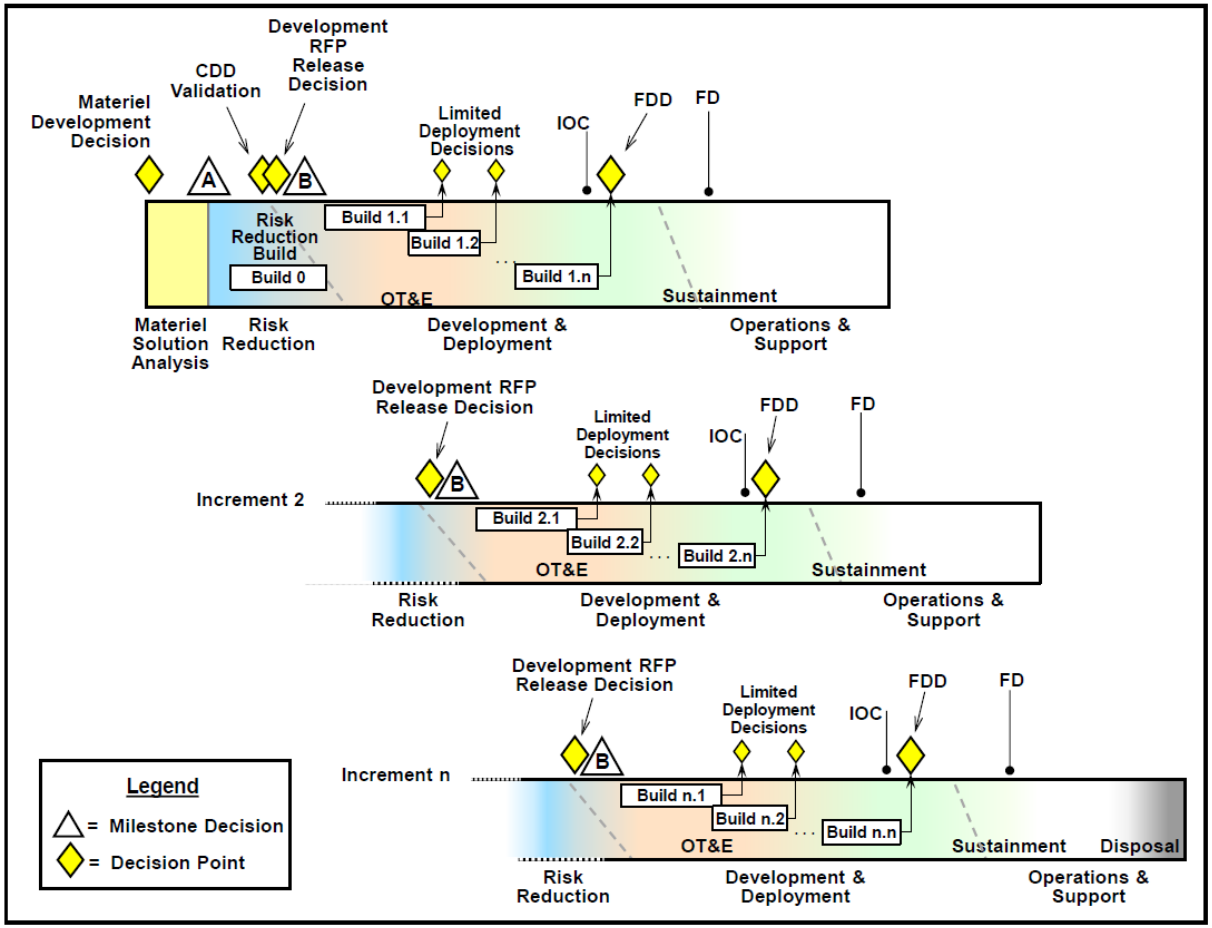
http://www.acq.osd.mil/docs/500002p.pdf

issued Jan 7, 2015

© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

Recent Changes in DoD Acquisition Guidance Explicitly Address IT Systems-2

Figure 5. Model 3: Incrementally Deployed Software Intensive Program

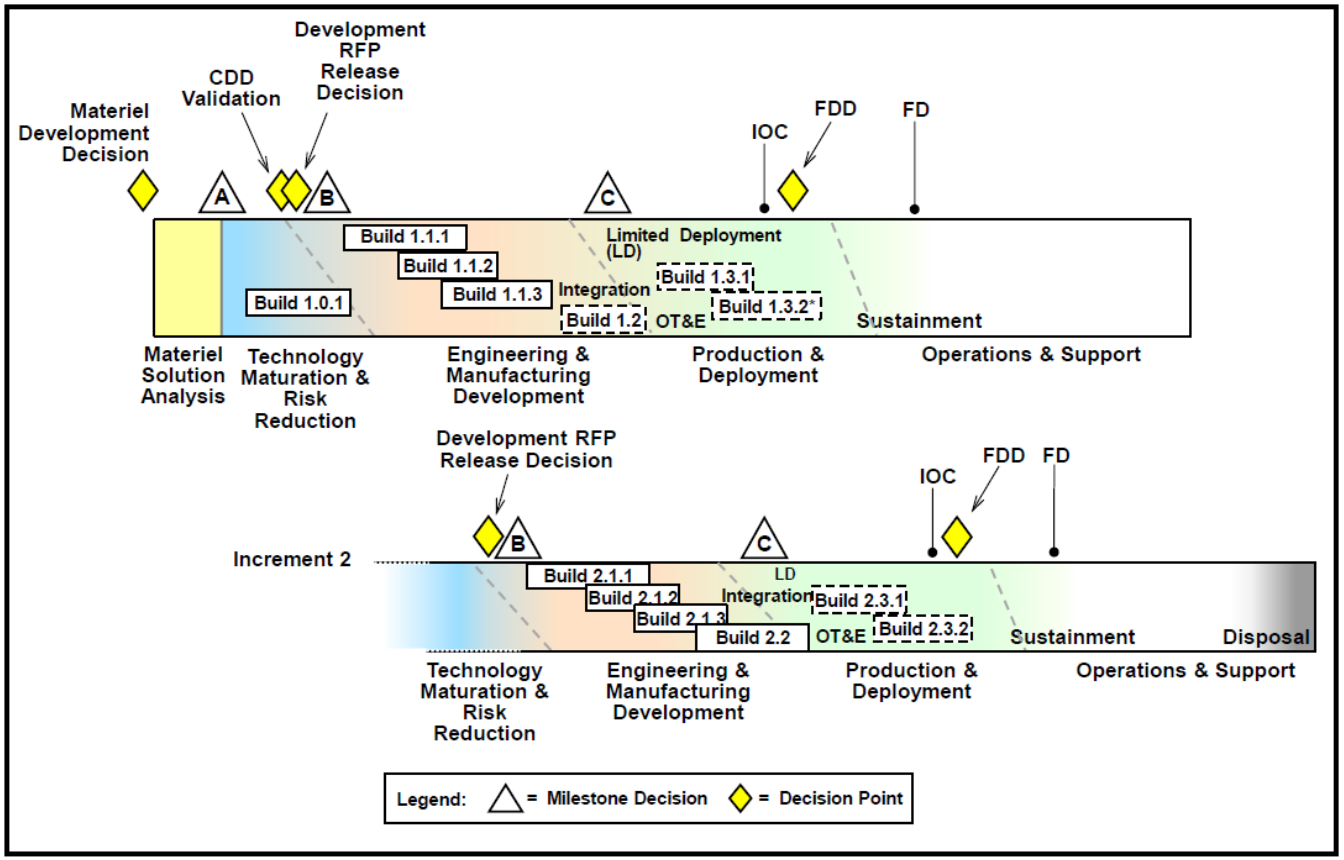


http://www.acq.osd.mil/docs/500002p.pdf issued Jan 7, 2015

© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

Recent Changes in DoD Acquisition Guidance Also Address SW-Dominant HW/SW Systems

Figure 8. Model 6: Hybrid Program B (Software Dominant)



http://www.acq.osd.mil/docs/500002p.pdf issued Jan 7, 2015

© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

Some Other Useful Insight

The TechFAR Handbook highlights the flexibilities in the Federal Acquisition Regulation (FAR) that can help agencies implement "plays" from the Digital Services Playbook that would be accomplished with acquisition support — with a particular focus on how to use contractors to support an *iterative, customer-driven software development process,* as is routinely done in the private sector.

https://playbook.cio.gov/techfar/

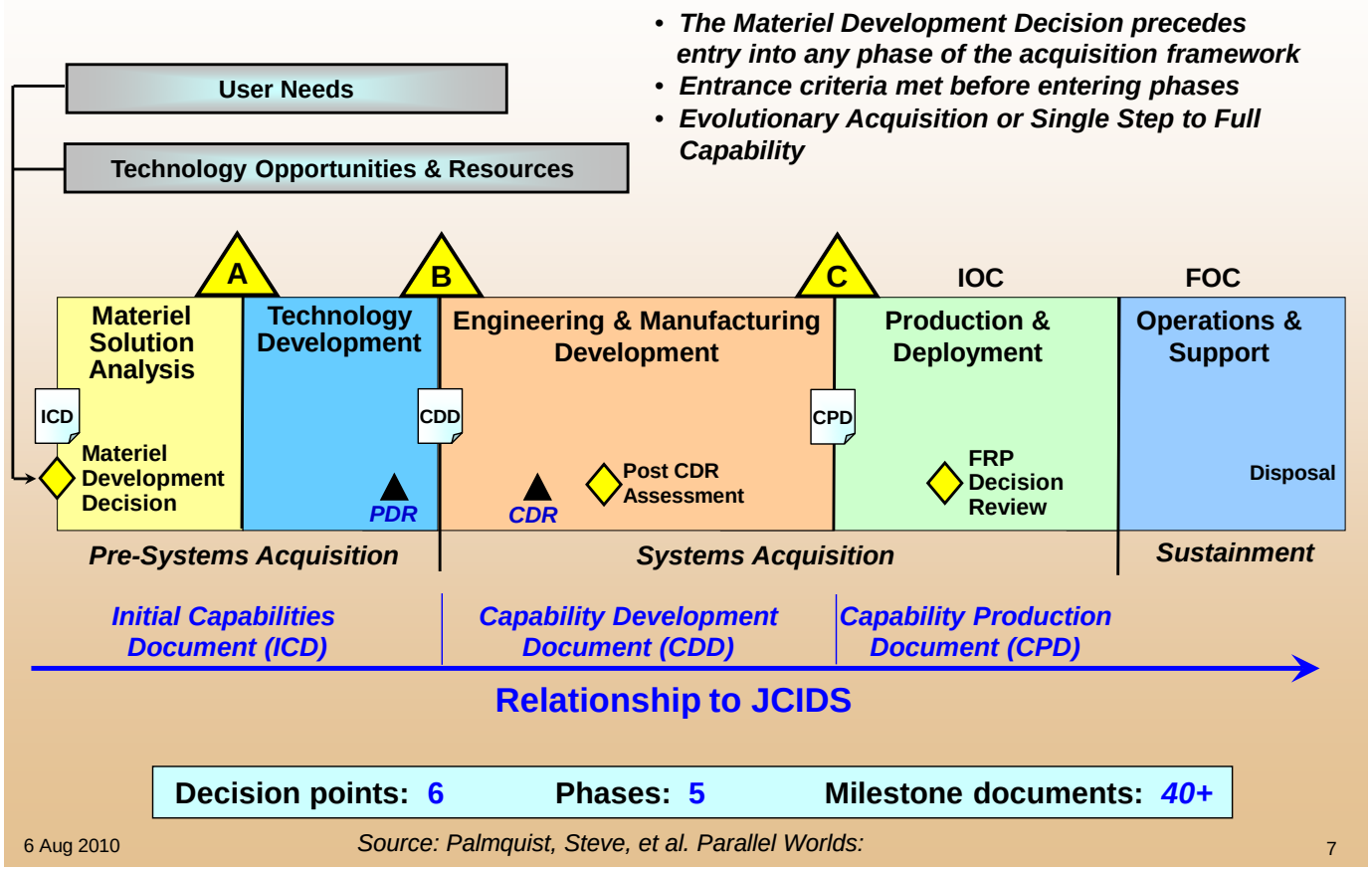
© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

ECO-SYSTEM

© 2014 Carnegie Mellon University. Published by
The Aerospace Corporation with permission.

Remember-Agile Methods are Used in Context of a Larger Program (More Often Than Not)

The Defense Acquisition Management System



6 Aug 2010

Source: Palmquist, Steve, et al. Parallel Worlds:

7

© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

Systems Engineering Mindset that Supports Agile Software Development

Product side

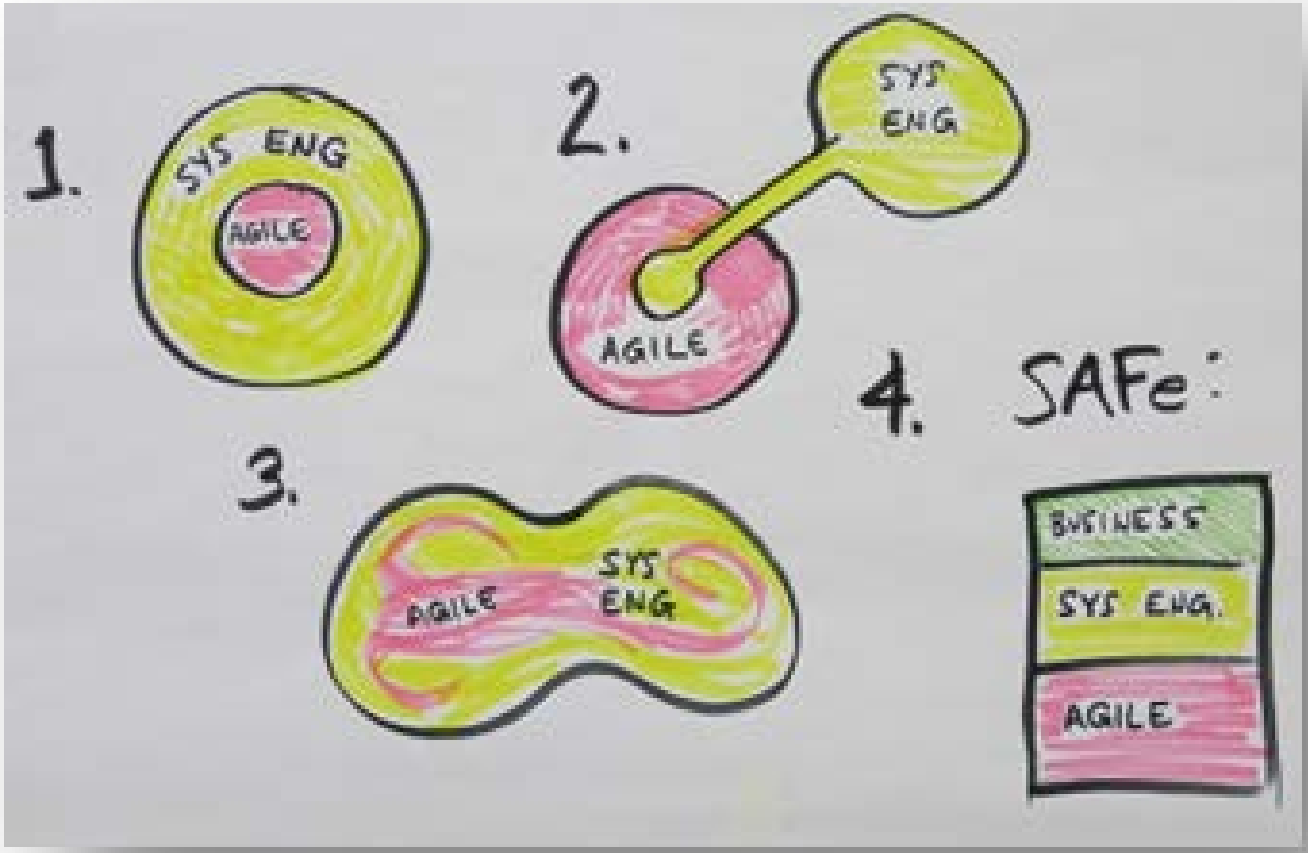
- Leverage incremental, iterative approach with heavy user involvement to increase speed of development of key requirement and design artifacts needed to implement different mission/system threads.
- Incorporate acceptance-test-driven development into activities of systems engineering to increase connection between two sides of systems engineering "V"

Service side.

- When program scale requires a separate systems engineering function, the coordination, communication and conflict resolutions services that are provided could translate into product owner surrogate role, Scrum of Scrums facilitator role or other specialty roles that show up in Agile scaling approaches.
  - Note: Scaled Agile Academy is working on a "continuous systems engineering" addition to SAFe

© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

Three Approaches for Systems Engineering and Agile



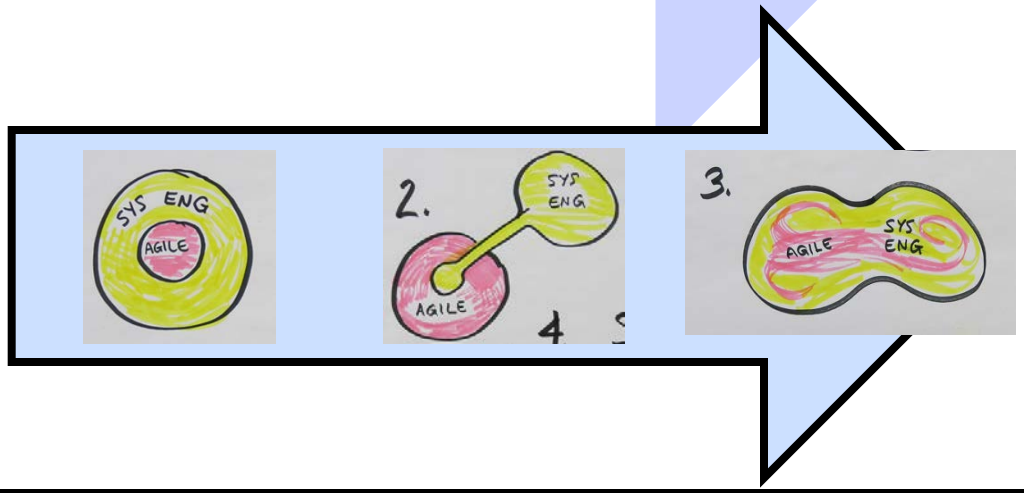
© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

Three Approaches We Observed of Systems Engineering Interacting with or Being Part of Agile Teams

Agile software teams interacting with traditional systems engineering

Systems engineers acting as Agile team members

Systems engineers applying Agile methods to their own work



In reality it is a continuum:

- As software teams demonstrated/continued to demonstrate success, systems engineering teams and leaders got engaged with software processes
- Successful activities led, in at least one case, to application of Agile methods to part of systems engineering process

© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

Components of the Eco-System

Components include but not limited too:

- Program Management Office
- Contracts
- Finance
- System Engineering
- User (end, interfacing system(s) users, etc)
- Stakeholders
- Information Assurance
- Supply Chain
- Developers
- Testers
- …….

© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

PRACTICAL IDEAS

© 2014 Carnegie Mellon University. Published by
The Aerospace Corporation with permission.

Properties of Successful Agile Project Teams

Frequent delivery

Reflective improvement

Close / Osmotic Communication

Personal Safety (free to speak without fear of reprisal)

Focus (knowing what to work on, having time to work on it)

Easy access to expert users

Sunshine / Visibility (no dark places in the project) (courtesy Sam Person, Overstock.com)

Technical environment

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

© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

Gimmes & Gotchas

Gimmes are a list of behaviors that give you confidence that your program office and/or contractor is embracing an agile process.

Gotchas are a list of behaviors that may indicate problems currently exist or on the horizon in your agile program.

These Gimmes & Gotchas **<u>are not</u>** intended to be all inclusive nor are they a checklist. The goal of these is to help identify areas to investigate further and focus your energy toward a successful program.

© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

First Steps Gimmes 1

- Your program and contractors understand the system is software reliant
- *Your motivation, trade-offs, benefits, and expectations for using an agile approach is clearly documented and understood*
- There is an explicit understanding that the requirements are expected to evolve
- Automated testing is planned for and budgeted
- Multiple acquisition approaches are being considered

Italics – this item also impacts external stakeholders

© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

First Steps Gimmes 2

- Contract allows flexibility and incremental delivery
- The concept of incremental delivery of content is included in CDRLs
- Entire program team is aware that the interim DoD 5000.02 has two new lifecycle descriptions that support more "agile" approaches
- Hindrances for agile implementation are acknowledged and paths to success are identified
- Readiness for agile adoption is determined by using a formal method such as the SEI Readiness and Fit Analysis (RFA)

© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

First Steps Gotchas

- *Senior managers and stakeholders reluctantly agreed to use agile or are unaware*
- Software development is constrained to a hardware architecture
- *Mindset that document completion equals progress*
- Program exists in a "risk adverse culture"
- Integration testing isn't planned until just before final delivery
- Testing isn't budgeted until much later in the program
- *Agile is being considered a silver bullet*

Italics – this item also impacts external stakeholders

© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

Readiness Gimmes 1

- Your agile approach has been tailored to best meet your program's needs
- Program office staff including systems engineers understand the agile process you're using
- The agile manifesto and principles are understood throughout the organization
- Appropriate training has been provided for the entire organization
- *Expectations and artifacts necessary for milestone decisions have been agreed to and documented*

Italics – this item also impacts external stakeholders

© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

Readiness Gimmes 2

- Agile roles and responsibilities have been clearly assigned
- *The definition of done has been established and includes what documentation is required*
- The contracting team has been trained and understands the agile process
- The program office is open to changing roles
- Systems engineers are an integral part of the agile process

Italics – this item also impacts external stakeholders

© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

Readiness Gimmes 3

- The schedule identifies when emulators/simulators are needed by the software development team
- Leadership and staff are educated on differences from the way they are used to doing business
- Program team has answers for most agile "myths"
- Program utilizes adoption support mechanisms

© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

Readiness Gotchas

- Your testing function/organization has not been integrated into the day-to-day activities
- Requirements stability, operating environment, and the evolution of the technology base has not been fully assessed
- *Constraints are imposed for the sake of tradition*
- *Contract progress payments are based on "earned value" for the accounting period*
- Testing organization is not involved in the agile process
- Regulations are cited as a reason not to embrace agile approaches

Italics – this item also impacts external stakeholders

© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

Application Gimmes 1

- *Your users and stakeholders can accommodate incremental deliveries*
- *Necessary and beneficial documentation has been identified*
- *Requirements can be prioritized without pushback*
- User stories conform to the "INVEST"* concept
- *Technical reviews are structured to understand technical issues and mitigate technical risks*

*** INVEST = Independent, Negotiable, Valuable, Estimable, Small, Testable**

Italics – this item also impacts external stakeholders

© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

Application Gimmes 2

- *Iteration and release reviews are used to build a case to demonstrate readiness to pass milestone reviews*
- Agile measurements are integrated into your overall management metrics
- Measurements are focused on "are we producing sufficient value fast enough?“
- *User requirements are validated during the creation of user stories*
- The program office responsibilities haven't changed but how they perform them has

Italics – this item also impacts external stakeholders

© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

Application Gotchas

- Your program or contractor is proposing agile as a quick fix for existing failures on the program
- Team metrics are used for comparisons
- *Users and stakeholders are not actively engaged in the agile process*
- Oversight activities are abandoned
- Focus is on compliance rather than mission success

Italics – this item also impacts external stakeholders

© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

GETTING OFF THE STAGE - IN SUMMARY

© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

Overall Messages

Agile is not a silver bullet for software acquisition, but it can be used in a government setting when appropriate

"Agile" isn't one approach or method – it's an umbrella term and is best thought of as reflecting the Agile tenets and principles of the Agile Manifesto

How the program office tasks for oversight are performed in a program using Agile is different from a traditional acquisition, but the responsibility of the PMO remains

I need to find out about the acquisition "ecosystem" my Agile program lives in before I can make decisions about what elements of Agile I can productively use here

© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.

Contact Information

Mary Ann Lapham
Principal Engineer
SSD/CTSD
Telephone: +1 412-268-5498
Email: mlapham@sei.cmu.edu

Suzanne Miller
Principal Researcher
SSD/CTSD
Telephone: +1 412-268-5498
Email: mlapham@sei.cmu.edu

U.S. Mail
Software Engineering Institute
Customer Relations
4500 Fifth Avenue
Pittsburgh, PA 15213-2612
USA

Web
www.sei.cmu.edu
www.sei.cmu.edu/contact.cfm

Customer Relations
Email: info@sei.cmu.edu
Telephone: +1 412-268-5800
SEI Phone: +1 412-268-5800
SEI Fax: +1 412-268-6257

© 2014 Carnegie Mellon University. Published by The Aerospace Corporation with permission.