



Agile and Incremental Software Development in the Defense Acquisition System

Mr. Sean Brady

**Office of the Deputy Assistant Secretary of Defense
for Systems Engineering**

**18th Annual NDIA Systems Engineering Conference
Springfield, VA | October 29, 2015**



Overview



- **Agile at Massive Scale in DoD (ACAT I programs)**

- DoD's demand for software
- ACAT I environment and complexity
- Exploiting the benefit of Agile values
- Adoption tipping point & ROI

- **Challenges in Our Environment**

- SE Rigor: Reconciling Agile SW Development w/ SE Technical Reviews
- More Predictable Delivery
 - DoD's reality: long-range commitments & estimates
 - Agile Metrics

- **Conclusion**



The Future of War

“...where soldiers and machines join forces in a multidimensional ‘informationalized’ zone to fight...
...combined in combat teams [with] new advances in robotics and autonomy and unmanned systems,
...we can create super-empowered squads with enhanced situational awareness and lethality.”



“...our technological superiority is slipping...we want to achieve an overmatch over any adversary from the operational theater level all the way down to the fighter plane, Navy ship or infantry squad...Battlefield advantages in the future are going to be very short-lived because the amount of technology that is out there right now is unbelievable.”

– Honorable Robert Work, Deputy Secretary of Defense
April 2015, US Army War College

2nd Cone of Uncertainty

Uncertainties in competition, technology, organizations, mission priorities



Image: Future Challenges for Software Data Collection and Analysis, 2009, USC-CSSE

“Simply delivering what was initially required on cost and schedule can lead to failure in achieving our evolving national security mission — the reason defense acquisition exists in the first place.”

– Honorable Frank Kendall
Under Secretary of Defense (AT&L)

2015 Performance of The Defense Acquisition System



DoD must be responsive to change.



Demand for Software

- **Software (SW) provides the decisive edge to our forces**
 - the SWE is the modern day swordsmith; critical to future battlefield dominance
- **SW acquisition for major programs poses some of our toughest SE challenges**
- **Compounding the challenge: the only constant for DoD systems is change**
 - Evolving threats | Strategic and tactical innovation | Resource and demand uncertainty
 - Rapid technological change | Increased Defense leverage of commercial systems
- **These factors all demand increased agility for military systems**
 - designs that afford capacity to adapt and adjust
 - maintaining operational advantage in an environment of change
 - ever increasing functionality controlled by software: SW can enable system change
- **Is DoD acquisition aligned with rapid and Agile methodologies?**
 - Can we ensure adequate SE rigor while delivering capability to the fight? evaluate rapid acquisition?
- **DoD wants: more predictable SW delivery | SE rigor | reduced cost- & time-to-Warfighter**

SW development agility: key contributor to program success



Software in the DoD Acquisition Environment



- **Governmental statutory and regulatory requirements**

- Upfront: significant analysis and needs justification prior to the decision to fund
- In-progress: numerous decisions points for moving to next phase
- Handoffs: test and evaluation by separate organizations

- **Broad spectrum of applications and environments**

- Limited demonstration opportunities for large, complex systems
- High test (e.g. full up live fire; survivability) and deployment costs may limit number of releases

Attributes of rigorous waterfall

- **DoD Leaders increasingly appreciate Agile SW practices & growing adoption**
- **DoD seeks practical approaches that mesh Agile with DoD's statutory, regulatory, operational and closed-scope environment**
- **DoD applauds any methodology that can improve SW acquisition & SE**
- **DoDI 5000.02 supports tailoring for adoption of Agile SW development**

“We are not in an easy business.” — Hon Frank Kendall, USD(AT&L)



Agile Affinity? Preponderance of DoD Acquisition: ACAT* ID and IC Programs



• **MAIS:** IT/SW-oriented; may engender less conflict adopting Agile

• **97% of acquisition funding is in MDAP ID and IC programs**

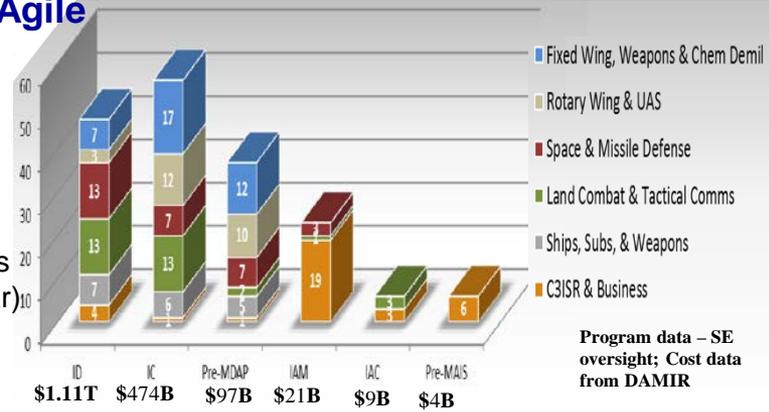
• **MDAP:** HW/SW-oriented w/ mix of SW:

- MDAP SW must sync w/ SW *and HW* requirements, schedules
- Diverse mission systems, safety-critical, real-time, embedded weapon systems
- Can't easily timebox/omit integral requirements (can't fight w/ "2 years" of radar)

• **MDAP cyber-physical and logical dependencies**

- complex ground/air/space dependencies

• **MDAP SW only a *subsystem* of massive, complex systems**



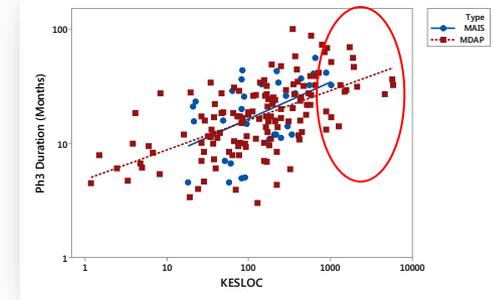
DASD(SE) Oversight 178 programs \$1.7T in acquisition

• **...yet SW efforts for MDAP subsystems can surpass those of entire MAIS**

- mass complexity & scale of SW development in MDAPs

• **Comparison of Top 5 MDAP and MAIS SW Efforts**

- \sum 5 Largest MAIS < Each of top 3 MDAP alone
- Avg of Top 5 MDAP > 7M ESLOC | Avg of Top 5 MAIS ~ 1M ESLOC
- Average of Top 5 MDAP Peak Staff is 5x > than MAIS counterparts
- Average of Top 5 MDAP Effort is 10x > than MAIS counterparts



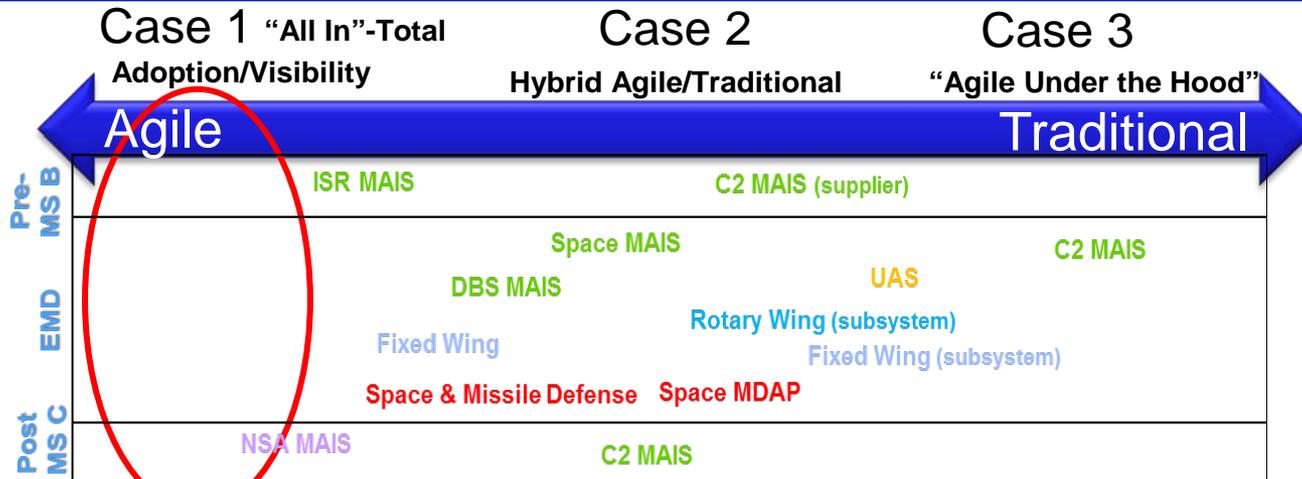
SW acquisition on ACAT ID and IC poses some of our toughest SE challenges

*Acquisition Category (ACAT) I programs are Major Defense Acquisition Programs (MDAPs) designated by the Under Secretary of Defense for Acquisition, Technology and Logistics - estimated to require RDT&E > \$365 million or procurement > \$2.19 billion (FY 2000 constant dollars) for all increments. ACAT ID: the Milestone Decision Authority (MDA) is USD(AT&L). "D" - Defense Acquisition Board (DAB), which advises the USD(AT&L) at major decision points. ACAT IC: the MDA is the DoD component head or the DoD component acquisition executive (CAE). "C" -component. Src: ACQuipedia <https://dap.dau.mil/acquipedia/>



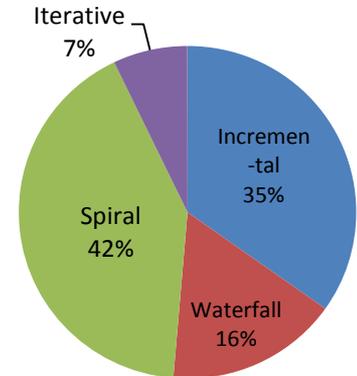
MDAP / MAIS

Notional Agile 'Population'



Position along horizontal axis are relative and notional.

- No ACAT I program is fully Agile
 - o Total adoption: suppliers, PMO, SE, test & customer participation);
 - o "badge-less society" between prime/suppliers; common metrics;
 - o continuous E2E delivery; working SW at every cadence;
 - o lack of big integration test events; separate test teams
- Is DoD gaining the full benefit of Agile theory?
- How do we become more Agile at scale and take advantage?



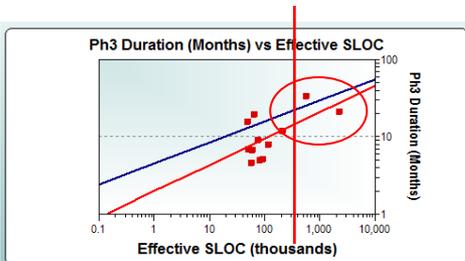
Majority of SW life cycles since 1995: not "traditional" waterfall.

Recency points at smaller waterfall footprint giving credence to "Strawman" Waterfall.

224 DoD SRDR SW Reports

- "Final" SRDRs (SRDR-3s); all Services, mix of MDAPs and MAIS; Current SRDR does not denote "Agile" SDLC

False Dichotomy: no ACAT I Program is fully Agile. Few are "classic" waterfall; most incremental; no single SDLC used in DoD



**Quant Trend for Agile in DoD at scale not established yet...
DoD studying emerging Agile adoption at ACAT I scale to determine benefit & ability**



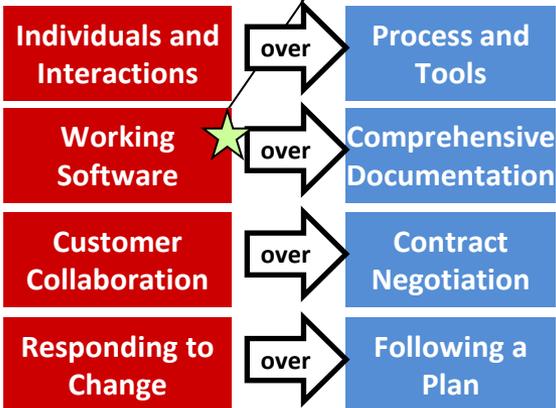
Agile Principles & Pillars... Can DoD Fully Realize the Benefit?

Hope for Benefit

• Principles & Pillars

- Inspect – Adapt – Transparency (Make Work Visible)
- Vertical Slice Development; Clean, Integrated Code
- Urgency & Motivation to Crush Impediments & Deliver on a Cadence w/ Common Purpose

• Agile Values



DoD Observations

★ How Agile are We:

- **“Agile-in-Name-Only” key discriminator: working SW on cadence**
 - (esp. w/ clean code pattern)
- **Sample DoD Experience on MDAPs**
 - **Inherent limits to mass “vertical slice” integration**
 - “Waterfalling” the sprint(s); **Impeccable mapping of Scrum onto WF**
 - Definition of Done: sprint **earning points for parts of SDLC**
 - Hardening Sprints | sprints leading to **big integration event @ end**
 - *“Aggressive Scrum”* can discriminate mutant implementations
- **Range of Agile Adoption Can Constrain Program Agility**

SE Rigor

- **Deliverables: going Agile can drive contract changes**
 - and drive just-in-time SE; metrics-needs

Predictability

- **Unstable & Unsustained Velocity; Slow-Start Scaling;**
 - not using “Yesterday’s Weather” to plan; ignoring team-level metrics

Polymorphous A(a)gile

- **Ranging/changing meaning**/claims: Agile in one context; Modified-Agile in another
- Can malign process & confuse stakeholder expectations

False Dichotomy:

- Traditionalist sees “Agile Zealot” || Agilest sees “Agile Denier”
- “Strawman Waterfall” – rarely encounter classic “waterfall”; incremental de facto approach

In its environment, to what degree can DoD exploit the benefit of Agile values?



Key Enabler for Agile at Scale Already Benefitting Programs



Continuous Systems Integration & Automated Testing

- SW tools & environments as productivity & force multipliers
- Integration environments giving DoD ACAT I PMOs
 - on-demand, deep visibility into their programs
 - reducing “integration” risk & cost of late defect discovery / refactoring
- Extraordinarily difficult to scale without



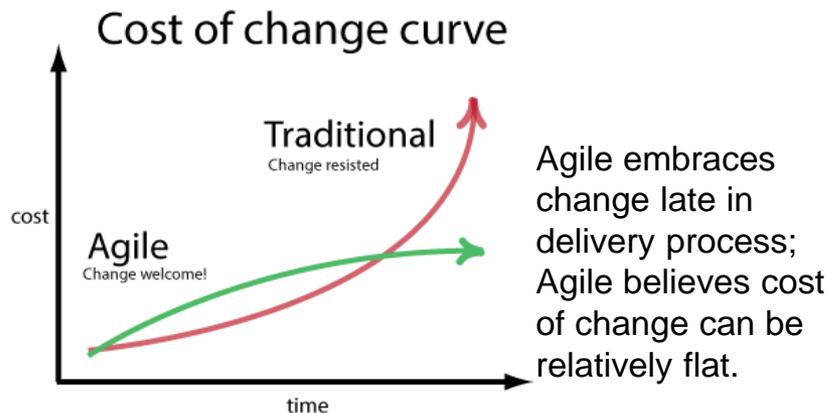
THE U.S. DIGITAL SERVICE

PLAY 10 - Automate testing and deployments

...verify 1000s of scenarios in minutes & then deploy updated code into production environments multiple times a day...manual tests still necessary...automated tests provide consistent protection against regressions

Building on immature code-base; Late learning & discovery and correction of defects is expensive;

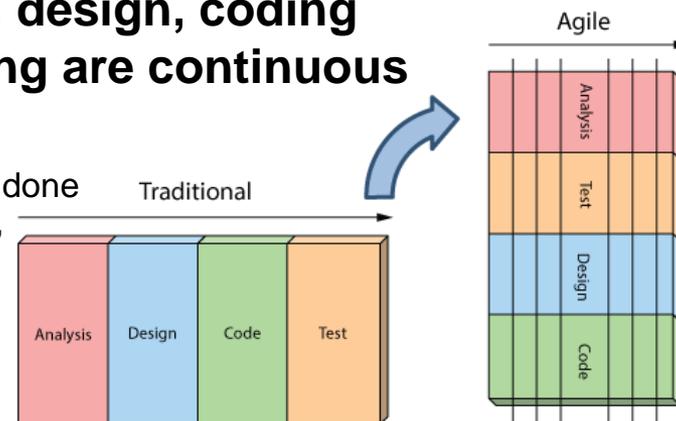
Environment enables mitigation of integration nightmares (fix-break cycles) & enables “vertical slice” delivery on cadence



Graphics & Annotation: http://www.agilenutshell.com/what_is_agile

Analysis, design, coding and testing are continuous activities

You're never done with analysis, design, code and test with Agile.



Continuous Integration & Automated Testing enables predictable, consistent delivery on a cadence.



Key Challenge: Reconciling Adequate SE Rigor and Tech Reviews w/ Agile SW



• Statute: 10 U.S.C. 2366b - MDAP certification required before Milestone B

- approval requires PDR and formal post-PDR assessment,
- certifies [entire] program “demonstrates a high likelihood of accomplishing its intended mission”

• Regulation: DoDI 5000.02 - PDR/CDR Assessments

- MDAPs/MAIS: a **post-PDR & -CDR assessment** to the MDA
- Assess design maturity, and the program’s readiness to begin ... software coding with acceptable risk.

• DAG Guidance for PDR:

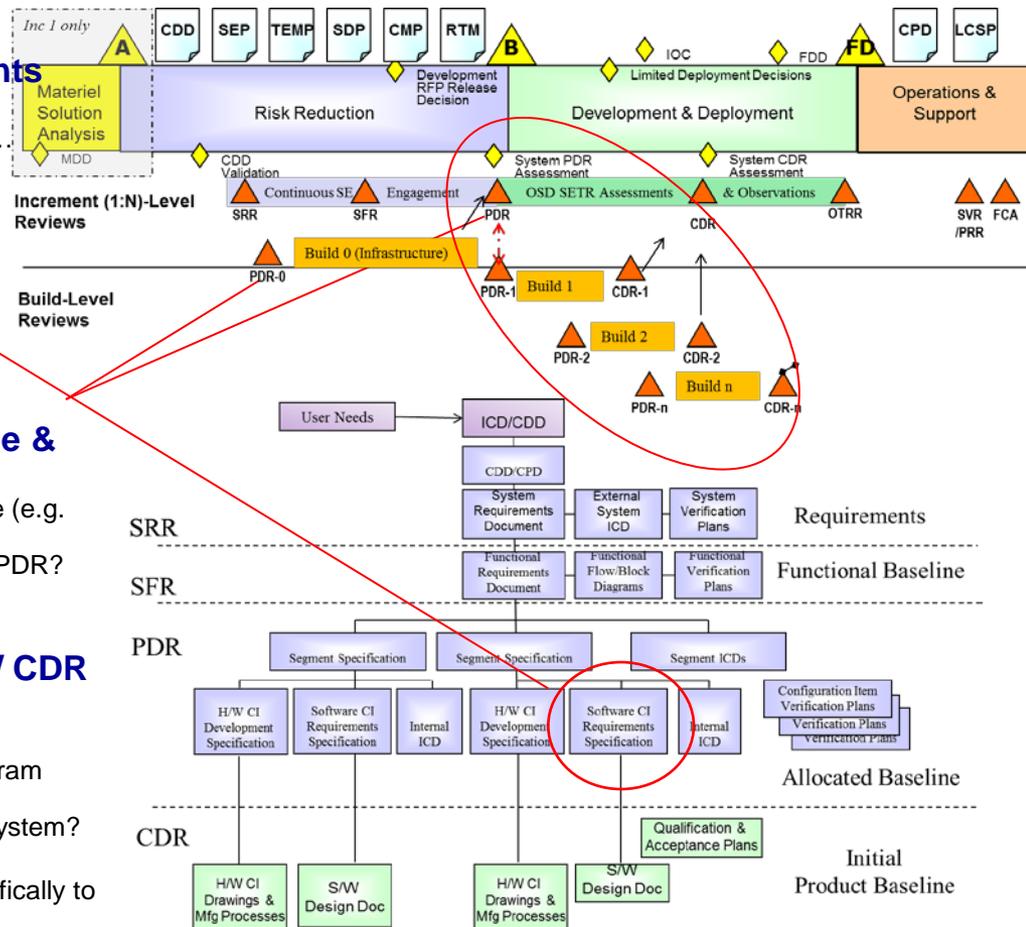
- ..software architecture designs have been established; all Computer Software Units (CSUs) have been defined.
- Software Requirements Specifications (SRSs) and Interface Requirement Specifications (IRSs), are complete for all CSCs.

• Incremental SW development, Technical Baseline & Technical Debt

- may not produce a fully established baseline at SETR milestone (e.g. allocated baseline gradually established)
- How many requirements and design decisions completed post-PDR? In an IT Box? In an MDAP? 25%? 80%?

• How does DASD(SE) formally assess post-PDR / CDR assessments given

- evolving requirements & design maturity
- evidence of enough **SE Rigor** to ensure “high likelihood of program accomplishing its intended mission?”
- ... when we have allocated baseline for a small fraction of the system?
- acceptable tailoring decisions including minimums related specifically to
 - PDR/CDR , related documentation/artifacts
 - allocated and product baseline content, and delivery points?





What are Potential Expectations for Adequate SE Rigor, e.g. PDR?



- **Minimum Viable Requirements**

- high level requirements covering the full scope of the effort.
- architecturally significant requirements (including non-functional, quality attributes)
- lower level requirements to meet critical functions & key quality attributes

- **Minimum Viable Architecture**

- initial software architecture and design; evidence of architectural evaluations
- may still be multiple decisions on the architecture which will emerge
- E.g., numerous candidate COTS products, but haven't made final selection yet (have at least one viable candidate solution)

- **Risk assessment covers full scope of the effort.**

- Some design decisions (non mission-critical?) will not be defined at PDR.
- Track architectural technical debt as a system-level risk
- Risk related to dependencies
- Infrastructure that enables mission significant requirements

*“It is not down in any map;
true places never are.”
-Herman Melville*

- **Useful progress and product metrics to ensure you can track plans**

- **Evidence artifacts that are coarse grain (big difference from traditional).**

- Admit we don't have all the detailed requirements, and these will come incrementally in the future and enable better trades?
- Document (e.g., SDP) a minimum set of characteristics for each that would make a coarse grain level artifact acceptable
- We expect updates to these artifacts to fill in the gaps (fine grain solutions).

- **Admit that when we learn more we may discover other risks.**

- **DoD wants adequate SE Rigor regardless of SW methodology.**

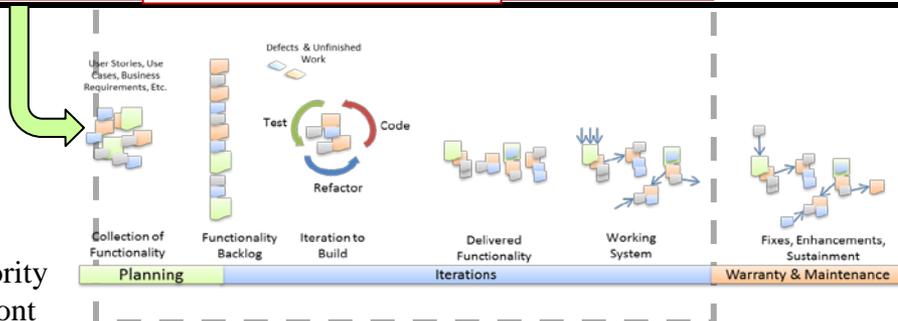


DoD's Reality: Long-Range Commitments and Estimates



Issue Framework: OSD CAPE MAIS Estimation WG

ISSUE 1: WHERE DOES AGILE ACTUALLY "FIT" IN THE COST WBS / MODEL?



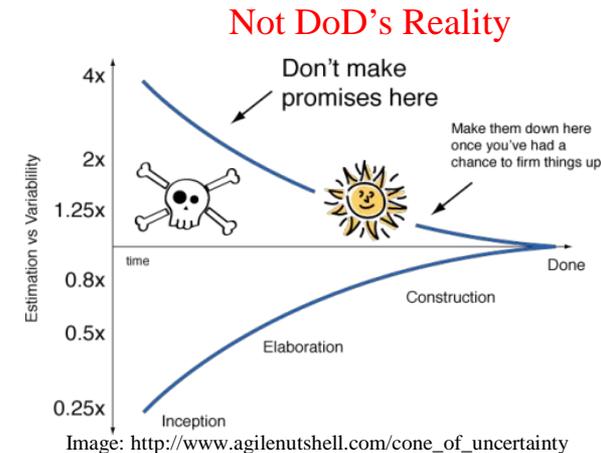
Responsibility, accountability and authority for multi-billion \$ program drives upfront analysis commitment to Congress, taxpayer and Warfighter

Traditional SW Estimation & Benchmarks

Agile Execution Monitoring – Progress to Plan

How can Agile Teams on contract tomorrow ... help compulsory long-range estimation today?

Agile is hostile to long-range estimation. DoD billion dollar programs require upfront significant analysis & needs justification prior to the decision to fund.



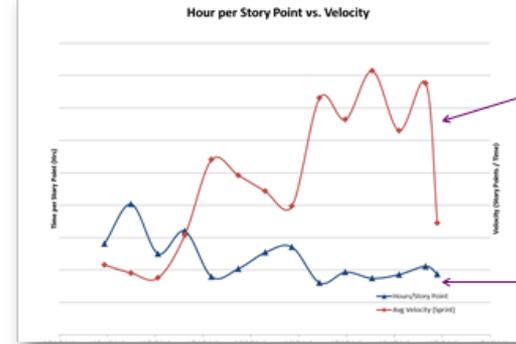


Agile Metrics and Quantitative SW Engineering Vital for Predictable Delivery



- **Meaning of SP (Done) must be understood**
 - Are system integration, DT & maturity factors baked in per Agile expectation
- ★ **Predictability — how well do we estimate?**
 - Sustainable development; can we sustain delivery pace?
 - Ignoring “Yesterday’s Weather” to plan; ignoring team-level metrics
- **Scaled metrics continued area of study — Normalization & Aggregation:**
 - Can safely monitor predictability, acceleration (& percentages) in aggregate
 - Can we meaningfully aggregate if the reference story is the same?
 - Aggregate velocity can hide Team velocity critical path risk
- **Daily, Sprint and Release cadence insights**
 - Sprint metrics optimized for team delivery;
 - At scale, measure effectiveness of synchronization and ability to deliver E2E thread
- **Lack of E2E Value Delivery — [does it] “Do Something” — Metric**

★ Unstable, Aggregate Agile Velocity



After 12 sprints, there is no predictable average velocity in aggregate.

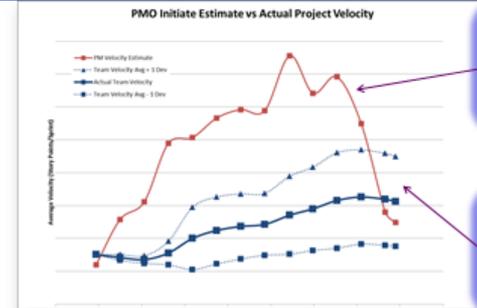
Impairs ability to predict capability shipments.

The average hours per story point by sprint indicates a possible trend

Without stability in metrics, these measures are difficult to use for future estimation.

18th NDIA SE Conference October 26-29, 2015 Page 47 Distribution Statement A – Approved for public release by DOPSR on 10/09/2015. SR Case # 16-S-0164 applies. Distribution is unlimited.

★ Challenges to Agile Estimation Aggregation & Normalization



PMO estimate differed substantially from reality

Standard Deviation is 30% of average velocity

“There is a difference in how estimations can be done at the iteration, release, and enterprise levels. At the iteration level, the team should always be involved. However, as the project gets bigger, the need for release- and eventually enterprise-level estimates may look more like those seen in Waterfall.” (CMU/SEI-2010-TN-002)

18th NDIA SE Conference October 26-29, 2015 Page 48 Distribution Statement A – Approved for public release by DOPSR on 10/09/2015. SR Case # 16-S-0164 applies. Distribution is unlimited.



Conclusions



- **Software is vital:** - providing the decisive edge to our forces
- **DoD systems and acquisition must be responsive to change**
 - Technology availability: will shorten Battlefield advantages in the future
 - Evolving threats & National Security mission
- **DoD seeks practical approaches that mesh Agile with DoD's statutory, regulatory, operational and closed-scope environment**
- **DoD is studying degree to which DoD can exploit the benefit of Agile values**
- **DoD ACAT I programs are increasingly adopting but none are fully Agile**
 - Limits observed in DoD for delivering working SW on a cadence
- **Polymorphous A(a)gile: communicate Agile expectations in SEP/SDP**
 - Avoid confusing stakeholder adoption expectations which can cause metrics dysfunction
- **Continuous Integration and Automated Testing is a Key Enabler at Scale**
 - Start at realistic level...gradually increase capability for inexperienced teams
- **Key Challenges**
 - Tailoring and communicating SETR decision point expectations w/o compromising SE rigor
 - Estimation and metrics –vital to understand SW delivery predictability; avoid the pitfalls
- **DoD at tipping point of understanding WRT benefit of ACAT I Agile adoption**
- **DoD applauds any methodology that can improve SW acquisition and SE**
- **DoD wants: more predictable SW delivery | adequate SE rigor | reduced cost- and time-to-Warfighter**

"There's a lot of promise in agile; I think a lot of us are excited about it...The question is going to be how agile is our acquisition system to take advantage of it."

- Hon William LaPlante,
Assistant Secretary of the Air Force
for Acquisition
Defense One Summit
October 6, 2015

"We find ourselves in a position where until you bring that software to bear against the systems that you're integrating, you don't really have a good sense for the quality and completeness of the SW..."and then you get into a test and fix mode." The adoption of agile methods could mean less uncertainty throughout the lifecycle of SW development"

- Hon Sean Stackley
Assistant Secretary of the Navy for
Research, Development,
Acquisition and Test
Defense One Summit
October 6, 2015



For Additional Information



Sean Brady

Program Support Team Lead for Software Engineering

Major Program Support

ODASD, Systems Engineering

(571) 372-6144

sean.p.brady.civ@mail.mil



Systems Engineering: Critical to Defense Acquisition



Defense Innovation Marketplace
<http://www.defenseinnovationmarketplace.mil>

DASD, Systems Engineering
<http://www.acq.osd.mil/se>



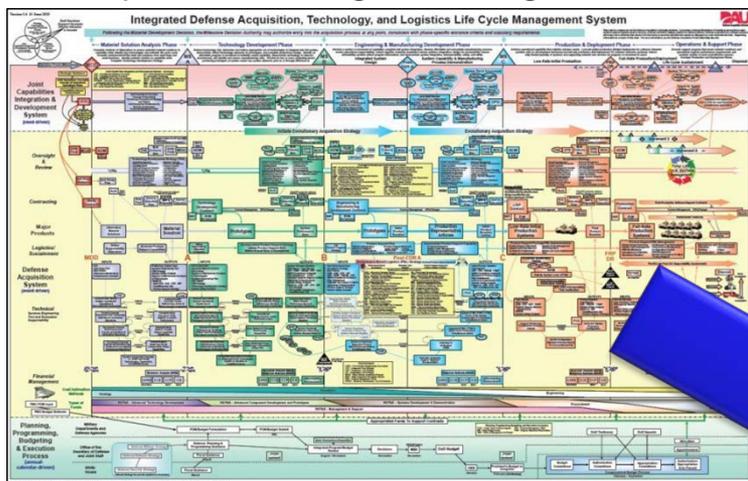
Additional Resources





DoD Framework Affords Flexibility and Critical Thinking

DoD Acq System (circa 2008)
 - showed waterfall System and SW development & the classic Systems Engineering “V” ...

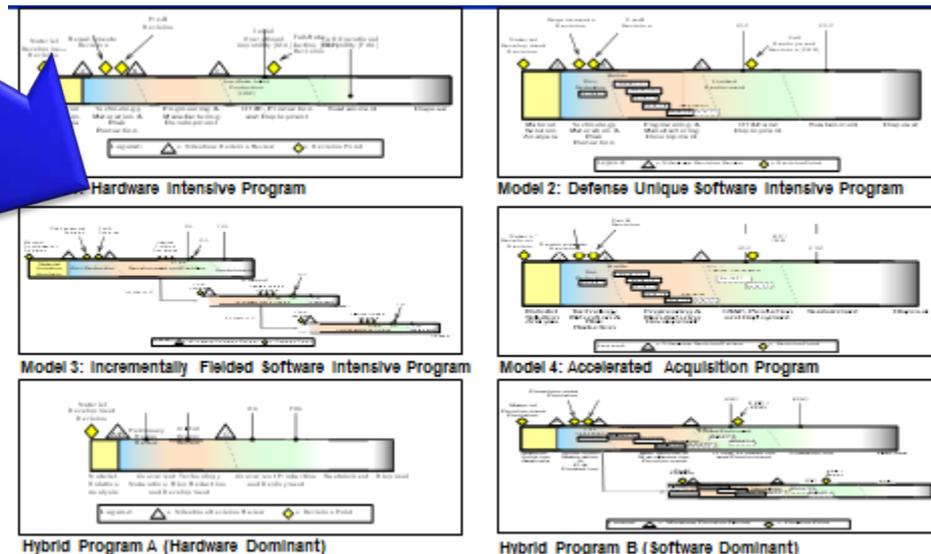


“the first responsibility of the acquisition workforce is to think... not to automatically default to a perceived ‘school solution’...”

...there's not just one size or one way to set up a program...examples are intended to serve as starting points, not a set of alternatives from which to choose ...”

- Honorable Frank Kendall, USD(AT&L)

... the new 5000.02 - emphasis is on tailoring and offers example models for structuring programs





Attributes of Agile Development

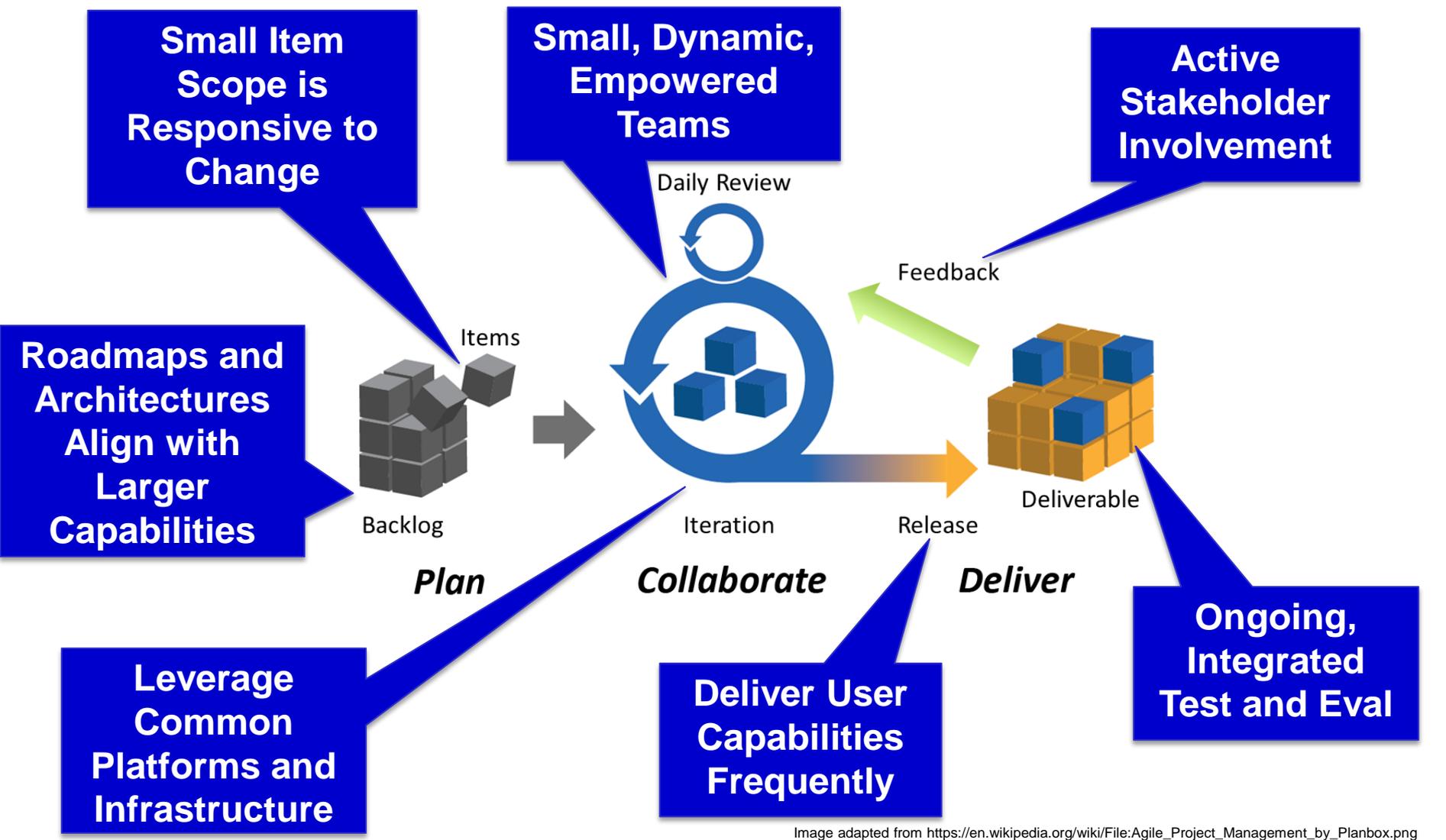


Image adapted from https://en.wikipedia.org/wiki/File:Agile_Project_Management_by_Planbox.png



Expectations for Adoption and Success Measures

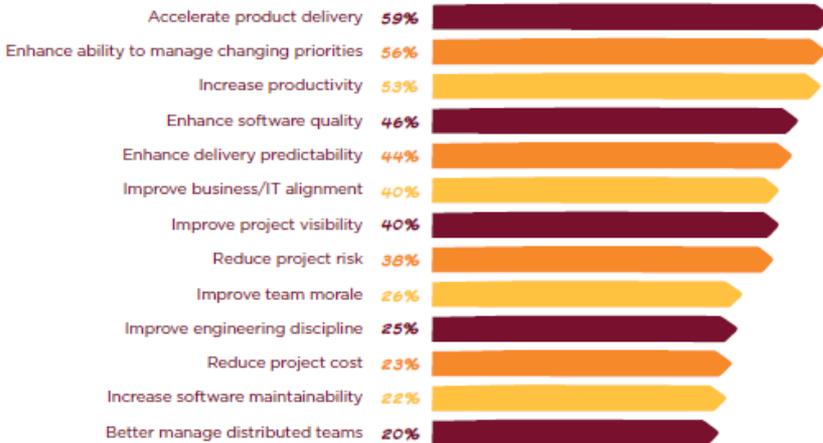


HOW IS SUCCESS MEASURED... WITH AGILE INITIATIVES?

When asked how respondents gauge the success of their agile initiatives, the most-cited value indicator was on-time delivery of projects – followed by product quality and customer /user satisfaction.

*Respondents were able to make multiple selections.

REASONS FOR ADOPTING AGILE



Source: VersionOne 9th annual State of Agile survey of 3,925 agile practitioners



Agile is... Agile isn't...

Highest priority is:

satisfy the customer

-- through early and continuous --
delivery of valuable software.

- Useful, Working Software
- User Driven Development
- Iterative Development
- Sustainable, Predictable SW Delivery Rates

Goal of Agile is not
purely speed!

It's focus is:

learning | early risk reduction |
frequent, quantitative feedback on
progress and process & product quality

Agile Myths...	Agile actually is...
...is just ad hoc, cowboy coding	...a set of proven best engineering practices that require significant process discipline to follow
...is just spiral or iterative waterfall	... iterative , but its practices also require developers to self-manage their work; both a process and a culture focused on communication and flexibility
...means we don't need to plan anymore	... planning done continuously and at multiple levels including a coarse grain and a fine grain level. The coarse grain level is critically important in DoD environments where up front approvals are needed. The fine grain level is critical for continuous risk assessment and design maturity assessment
...you're either purist agile or you are not agile at all	...there may be degrees of agility...the right degree for a DoD program depends on many factors. Because there is not one size that fits all, Agile programs require more up front planning and tailoring than traditional programs
Agile won't work in DoD or government environments	...DoD 5000.02 encourages program tailoring: there may be an appropriate level of agility for any DoD program

Sources: Introduction to Agile Engineering, 2013, MITRE | Agile 101, 2015, CMU/SEI | DASD(SE)/MPS, 2015



“Traditional” Waterfall vs. Agile: General Mindsets and Practices



“Traditional” Waterfall		Agile
<u>Scope</u> ; not Time and Cost	★ Fixed	Time and Cost; <u>Not scope</u>
Completely defined in detail up-front; resists change; Tries to be predictable	★ Requirements & Estimation	Iteratively refined during development; welcomes change; Accepts that total predictability in software requirements is impossible
Detailed cost estimates and full funding	Risk Reduction	Incremental releases and sprints
Early, large, and document-intensive	Reviews	Small, frequent, and often informal
Process and documentation over people	★ Emphasis	Knowledgeable, empowered teams/people and delivered code
Detailed plans freeze solution early	Baselines	Adapted to new info learned throughout development
At end of an increment (years)	Delivery	At end of a release (months)
Earned value measures and conformance against plan	Measurement	Frequent capability deliveries; value to the customer
Independent, following development	★ Testing	Daily development, integration, test
Acceptance at end of increment	★ Users	Active for continual review and feedback
Hierarchical, command-and-control, formal structures; difficult to change; IPTs	Org Structure	Flexible and adaptive structures,
Leader as keeper of vision; primary authority to act	Leadership Style	Facilitative; champion and team advocate
Top-down; documentations; indirect	Communication / Decision Making	Daily stand-up; face-to-face; retrospectives; info radiators

★ Key DoD Friction point

Agile misinterpretations: **no documentation, no process, no plan...**
 ...successful Agile implementations: **look for *balance*, not extremes**

Sources: Introduction to Agile Engineering, 2013, MITRE | SEI |DASD(SE)/MPS, 2015 | http://mike2.openmethodology.org/wiki/Agile_Business_Transformation

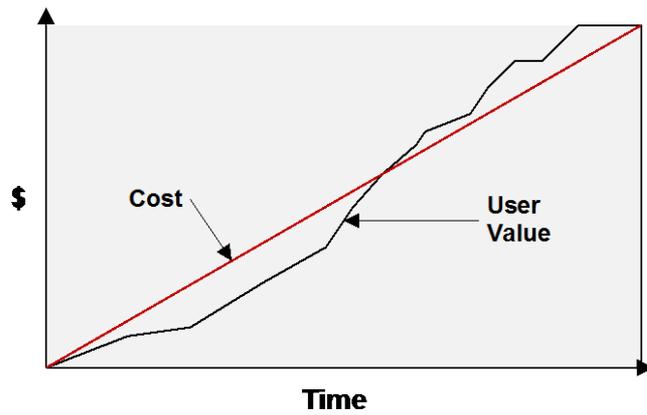
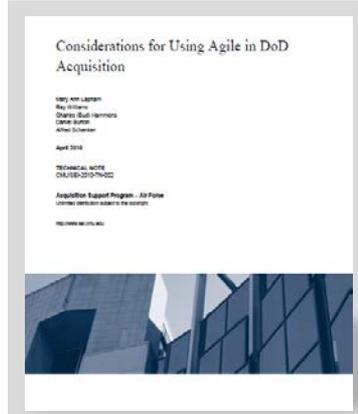
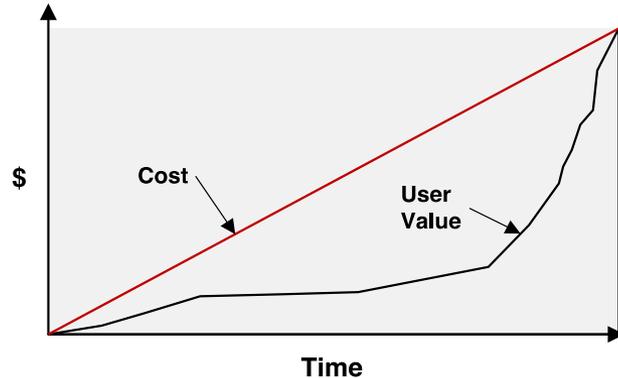


Software Development Life Cycles



• Comprehensive Planning

- Models like Waterfall
- Early baseline and lock down of design slows requirements change and cost growth.
- Capability appears at the end



• Incremental Planning

- Agile Models like Scrum and TDD
- Evolving plans can adapt to changes without causing rework, waste, and development cost growth
- Capability appears uniformly through the program

Major DoD Programs will likely select different life-cycle models for specific portions of the development effort



Combatting Misperceptions

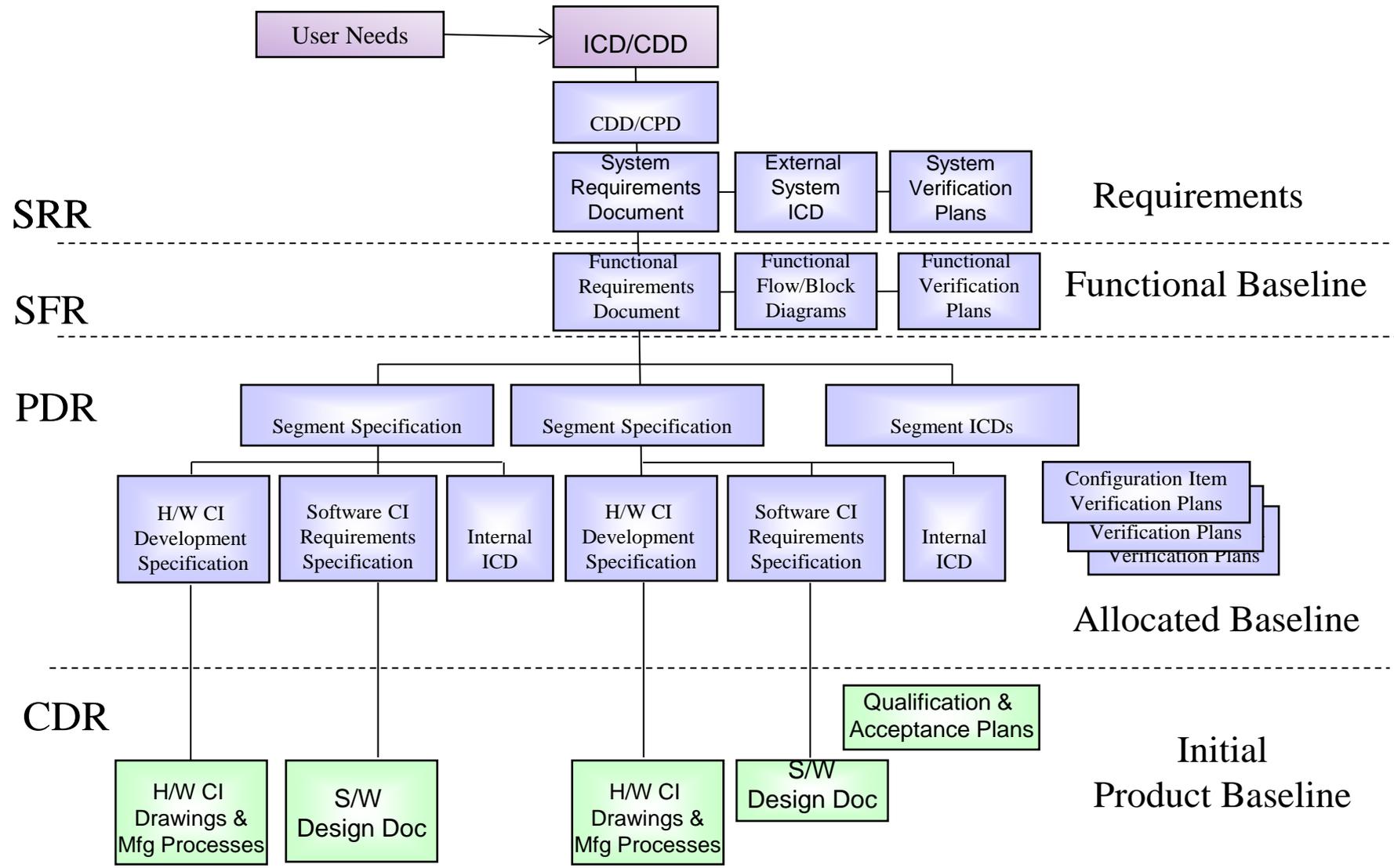


- **DoD Management is risk averse**
 - DoD applies significant effort to identifying, quantifying, and managing risk
- **New processes and methods don't mix with existing techniques**
 - This has never been the case. Projects routinely pilot new techniques on small tasks and expand their use gradually. One size does not fit all
- **Limited available workforce to conduct Agile software development**
 - Like any emerging technique, training and experience must be built into the workforce incrementally as the approaches are applied to more programs
- **Agile methods only apply in open-scope commercial settings**
 - DoD has been successful in applying Agile methods in closed-scope programs where all software requirements must be satisfied for the program to succeed
- **Agile software development means you don't have a long-term plan**
 - Software methods only apply after capabilities have been defined and allocated to software. Long-term project planning is done at a systems engineering level
- **Agile is difficult to contract for**
 - Contracts provide systems or capabilities, they define work at the systems engineering level
- **Metrics for agile technologies are hard**
 - Process-driven metrics for agile are different, but no harder to calculate or track
- **Agile means no documentation or artifacts**
 - On the contrary, Agile techniques provide design and implementation/test artifacts throughout the development process



Technical Reviews, Baselines and Documents

CDR





Performance Measures

Recommended Measurement Categories for DoD Acquisition Programs to Tailor for Domain & SELC



Quantitative Process Measures

Software	Production
Demographics Effort Productivity, Agile Velocity Schedule Staff Test	Build-to-Package Completions Traveled Work Supplier/Sub Quality Tests Scrap, Rework and Repair Hours Touch Labor Hours Yield
Staffing *	Technology Maturity
Quantity Effort Hours Experience Turnover Rate	Manufacturing
Schedule *	Design/ Development
Requirements Management	Architecture
Cost	% DoDAF drawings complete Quality Attributes Flexibility, Stability
Affordability Resources Dollars/Funding CPI	Integration
Risk Management	COTS/GOTS/NDI Components Interface Definition Interface Verification Interface Stability
Exposure Burndown	System Assurance
	Infrastructure

Product TPMs

Software	Mission Performance
Defects Quality Size	Mission Threat & End-to-End Performance e.g. Probability of Detection
System Performance	Net Ready KPP
Accuracy Lethality Bandwidth System Latency System Throughput System Response Time Utilization—Data bus, CPU, Memory SWAP-C Range	Network Management Time to enter network Time to exchange data
System Quality *	User Acceptance
Reliability	User questionnaire scores User acceptance scores
# unscheduled reboots Time between reboots (MTBCF) Time to reboot (MTRCF) MTBF, MTTF	Supportability/ Maintainability
	Maintainability Characteristics Mean time to repair

Legend

Category	MDAP-centric
Sub-category 1 Sub-category 2 ... Sub-category N	Included on SRDR

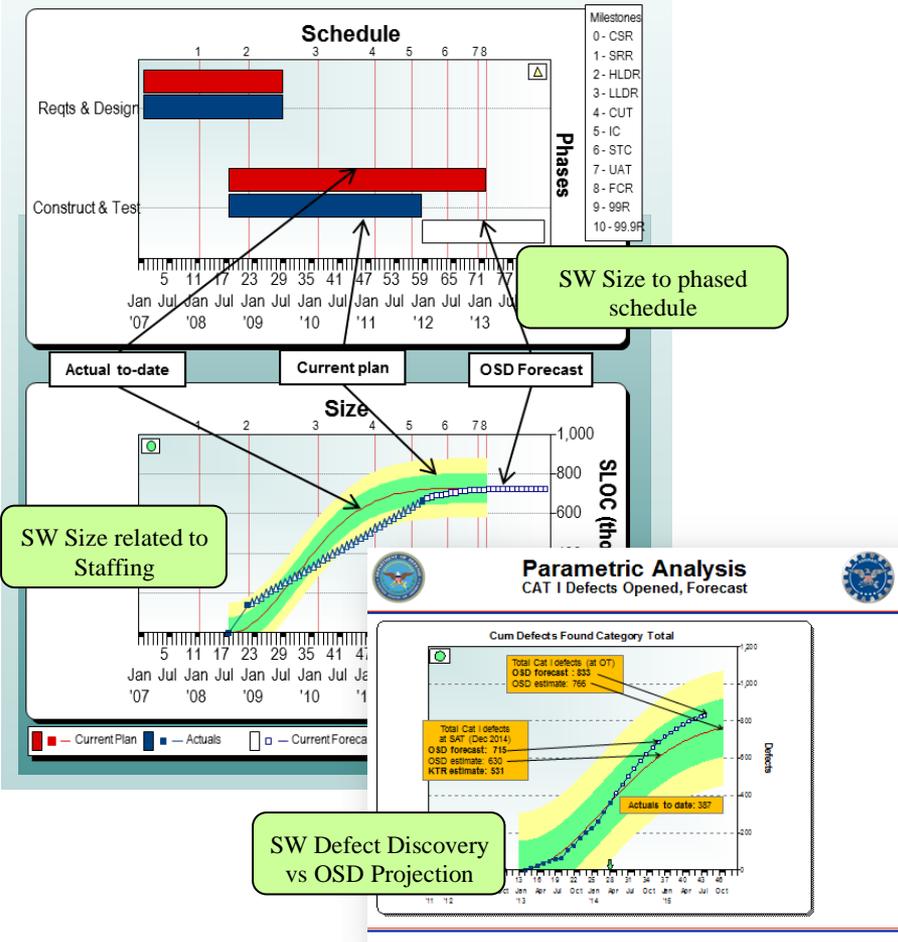
* Staffing, Quality & Schedule are also included in the Software Category



Software Metrics and TPMs

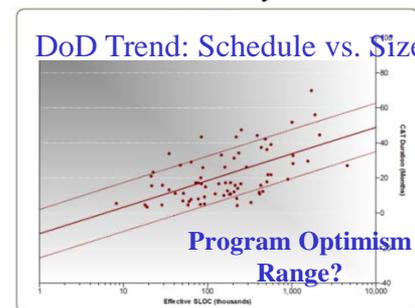
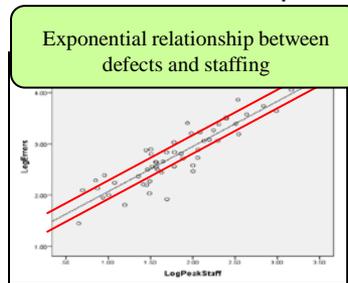


Performance to Plan

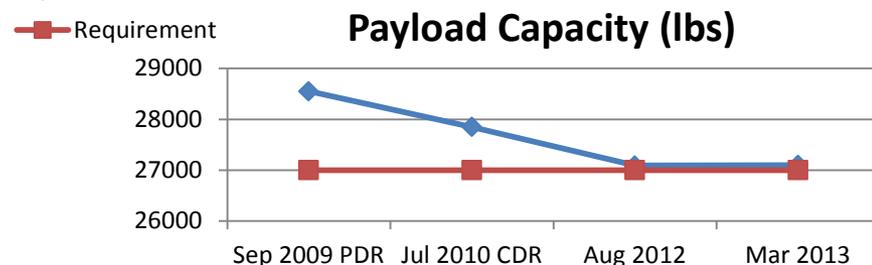
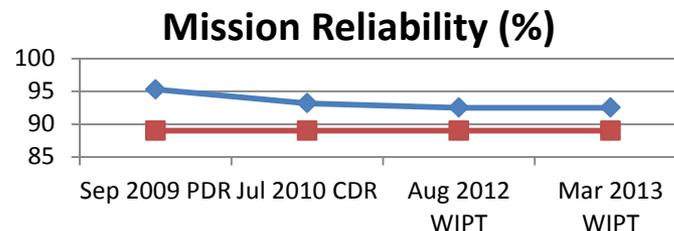


Analysis & Benchmarking

- SE models/assesses SW schedule and maturity
 - benchmark performance / statistically anomalous behavior



Performance to Plan track KPPs, KSAs, and TPM



Traditional SE/SW metrics critical to assess performance, schedule realism and SW maturity