Ground System Architectures Workshop

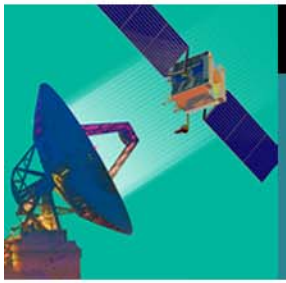
Session 11E

Adopting Agile Ground Software Development

Supannika Mobasser

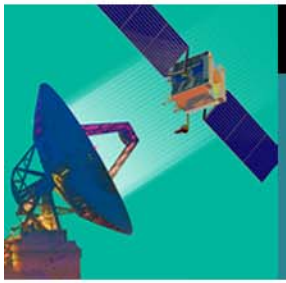
The Aerospace Corporation

AEROSPACE



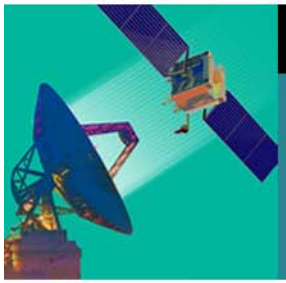
Overview

- *To look beyond the horizon and to embrace the rapid rate of change in ground software system development, it is crucial to be flexible, resilient, and robust.*
- *Seems like there is nothing new under the sun, but Agile provides different perspectives.*
- *Many programs started to adopt Agile, many programs are hesitant.*
- *Several success stories, yet several struggling stories*
- *Share your Agile adoption experiences and learn from others*
  - *Participants with all levels of agile expertise are welcome.*



Introduce ourselves

- What is your name?
- Where are you from?
- One good thing about your experiences in Agile adoption
- One pain point about your experiences in Agile adoption
- What's your expectation about this working group?

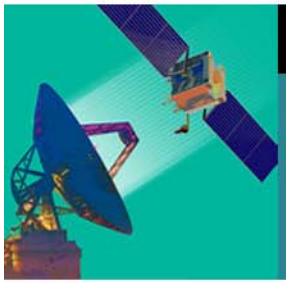


Schedule

Time	Presentation and Discussion
1:00 – 1:30pm	Session Overview
1:30 – 2:15pm	“Smashing the Stovepipe” Paul Swenson, ASRC Federal Space & Defense
2:15 – 3:00pm	“GMSEC Services Suite - An Agile Development Story” Vuong T. Ly, GSFC, NASA
3:00 – 3:30pm	Break
3:30 – 4:00pm	“Re-visit Agile Lessons Learned from previous GSAW” Supannika Mobasser, The Aerospace Corporation

4:00 – 5:00pm	General discussions
	• Agile Acquisition Models
	  • Milestones and Deliverables; Roles and Responsibilities
	• Agile and other disciplines
	  • MBSE, hardware-intensive subsystem, accreditation, etc.
	• How to measure success
	• Trust management

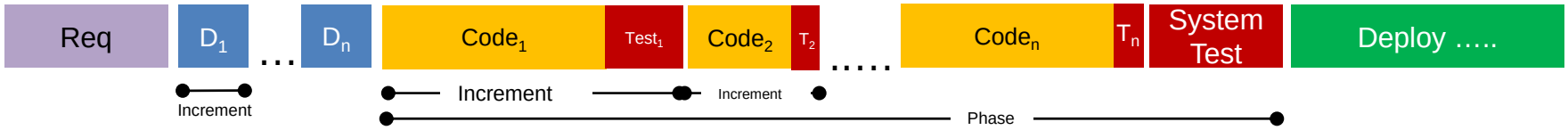
* Notional timeline



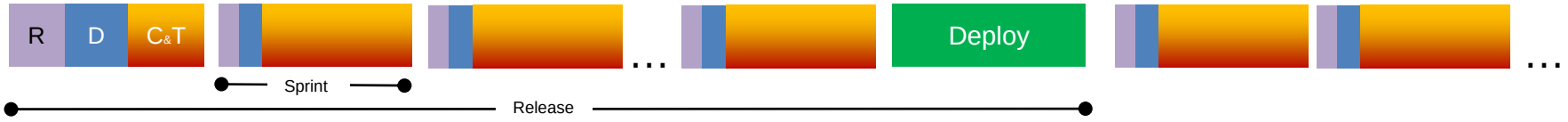
Process Models Comparison

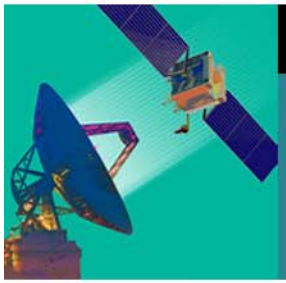
Waterfall / V-Model

Iterative and Incremental



Agile





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:

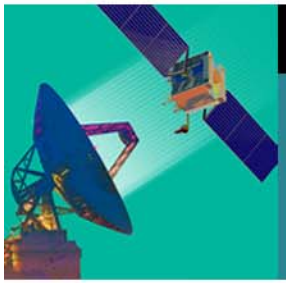
Individuals & interactions	over	**Processes** & tools
Working **software**	over	Comprehensive **documentation**
Customer **collaboration**	over	**Contract** negotiation
Responding to **change**	over	Following a **plan**

[Ref: Agile manifesto http://www.agilemanifesto.org/]

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

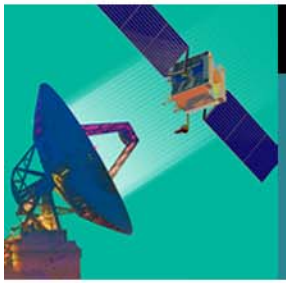
Agile development promotes

- Adaptive planning
- Evolutionary development and delivery
- Time-boxed iterative approach
- Rapid and flexible response to change



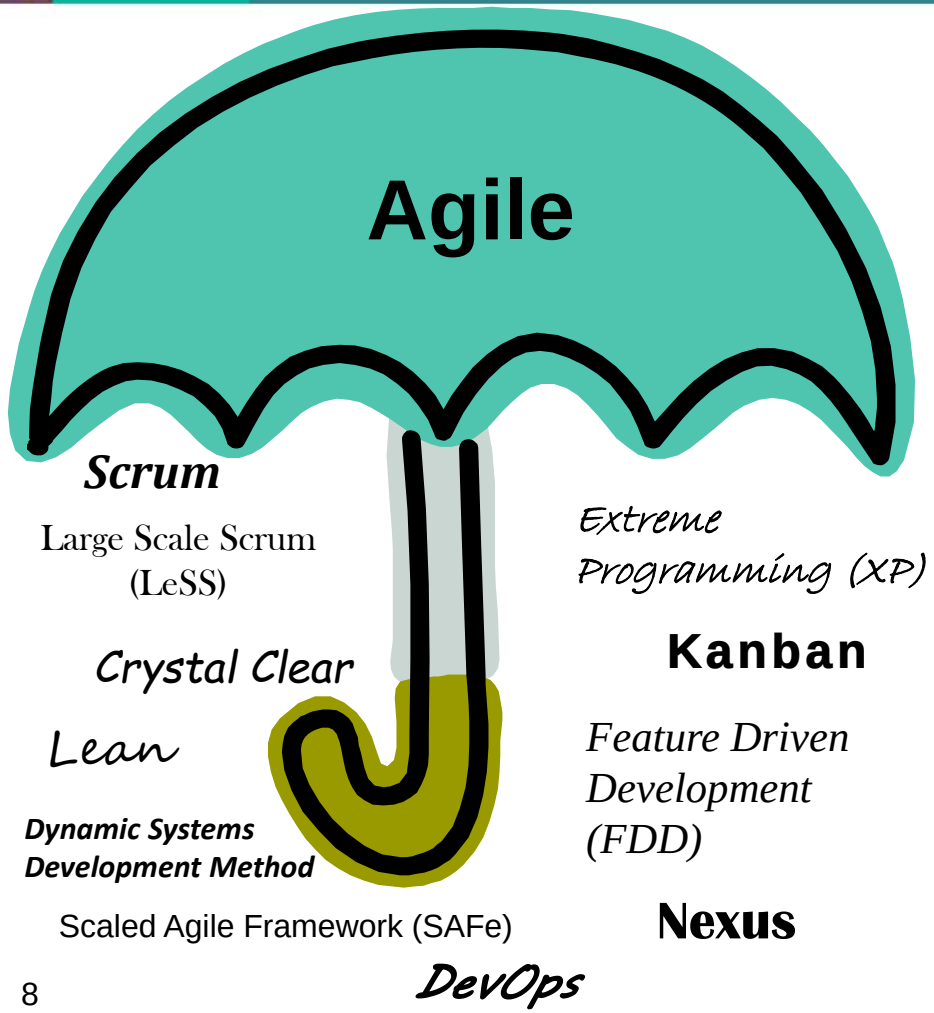
12 principles of Agile software development

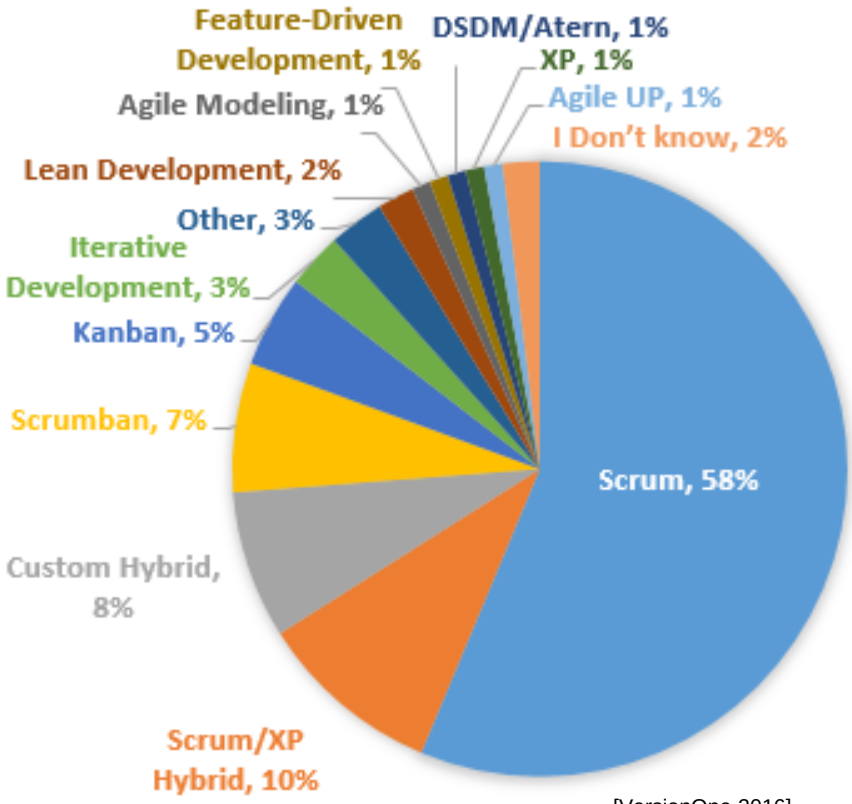
1. Our highest priority is to **satisfy the customer** through early and continuous delivery of valuable software.
2. **Welcome changing requirements**, even late in development. Agile processes harness change for the customer's competitive advantage.
3. **Deliver working software frequently**, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
4. Business people and developers must **work together daily** throughout the project.
5. Build projects around **motivated individuals**. Give them the environment and support they need, and trust them to get the job done.
6. The most efficient and effective method of conveying information to and within a development team is **face-to-face conversation**.
7. **Working software** is the primary measure of progress.
8. Agile processes promote **sustainable development**. The sponsors, developers, and users should be able to maintain 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.



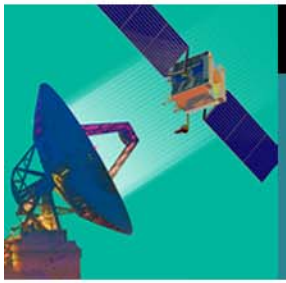
Agile Methodologies

Scrum: the most popular agile methodology in the commercial sector

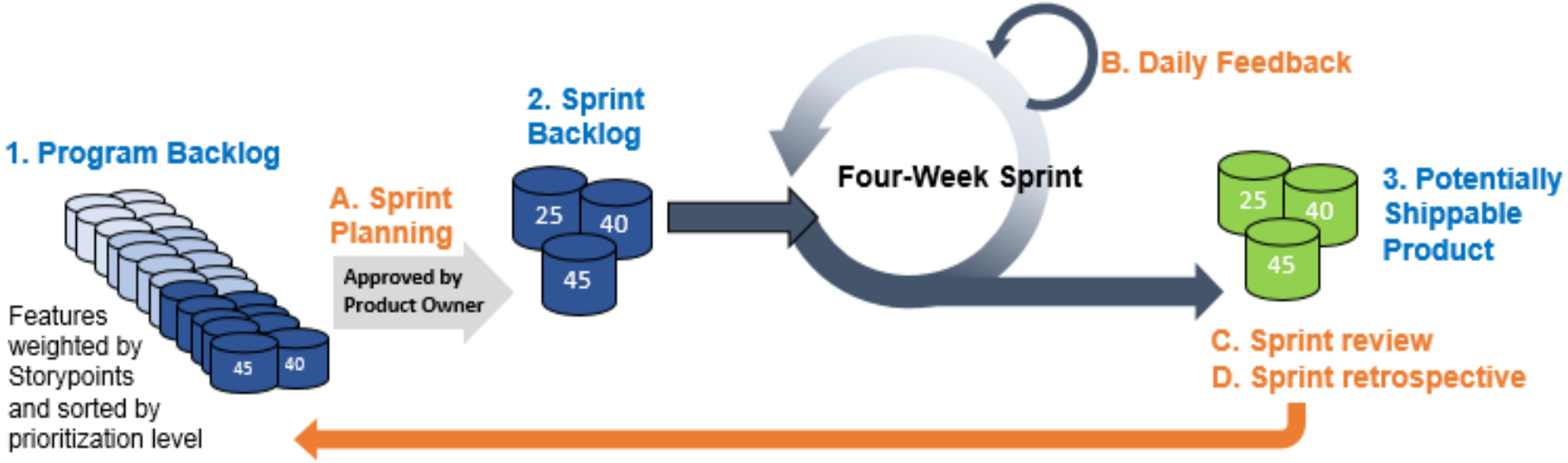




[VersionOne 2016]



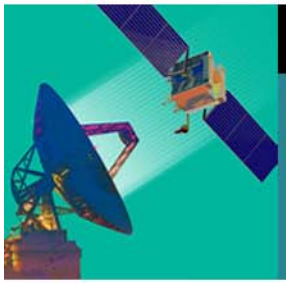
Scrum Process Model



Roles:
Scrum Master, Scrum Product Owner, Development Team

Artifacts:
Product Backlog, Sprint Backlog, Potentially Shippable Product

Activities:
Sprint Planning, Daily Scrum, Sprint Review, Sprint Retrospective



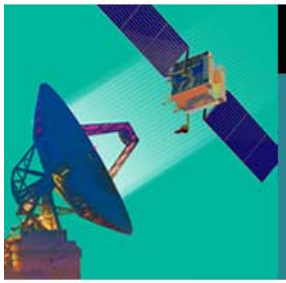
Adopting Agile: Two sides of the same coin

Benefits of Agile:

- Welcome changing requirements
- Quick turnaround time
- No big design up front
- Just in time and just enough

Concerns about Agile:

- Not “business as usual”, what do I have to do differently ?
- But the processes and regulations require …….
- So, everyone can do whatever they like?, what should I expect?
- Trust but Verify, how to do that?



Positive Feedback from the industry

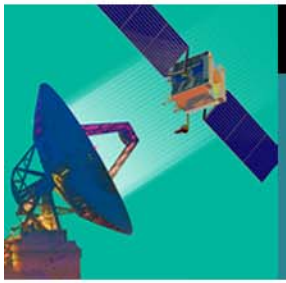
From VersionOne 2016 Report

Top three benefits of Agile

- 87% Ability to manage changing priorities
- 85% Increased team productivity
- 84% Improved project visibility

Reasons for adopting Agile

Accelerate product delivery	62%
Enhance ability to manage changing priorities	56%
Increase productivity	55%
Enhance software quality	47%
Enhance delivery predictability	44%
Improve business/IT alignment	44%
Improve project visibility	40%
Reduce project risk	40%
Improve team morale	29%
Improve engineering discipline	24%
Reduce project cost	23%
Increase software maintainability	22%
Better manage distributed teams	21%

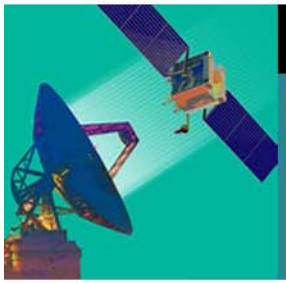


Negative Feedback from the industry

From VersionOne 2016 Report

Leading Causes of Failed Agile Projects

Company philosophy or culture at odds with core agile values	46%
Lack of experience with agile methods	41%
Lack of management support	38%
Lack of support for cultural transition	38%
Inconsistent agile practices and process	38%
External pressure to follow traditional waterfall processes	36%
Ineffective management collaboration	34%
A broader organizational or communications problem	30%
Unwillingness of team to follow agile	30%
Inability to continuously prioritize work	28%
Insufficient training	27%
Ineffective collaboration	25%
Don't know	5%

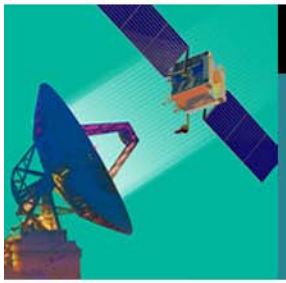


Negative Feedback from the industry

From VersionOne 2016 Report

Barriers to Further Agile Adoption

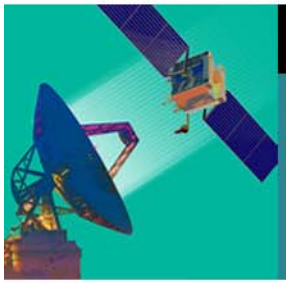
Ability to change organizational culture	55%
General organizational resistance to change	42%
Pre-existing rigid/waterfall framework	40%
Not enough personnel with the necessary agile experience	39%
Management support	38%
Business/user/ customer availability	28%
Concerns about a loss of management control	27%
Management concerns about lack of upfront planning	25%
Confidence in ability to scale agile methodologies	18%
Concerns about the ability to scale agile	18%
No barriers	17%
Perceived time and cost to make the transition	15%
Development team support	14%
Regulatory compliance	13%



Schedule

Time	Presentation and Discussion
1:00 – 1:30pm	Session Overview
1:30 – 2:15pm	"GMSEC Services Suite - An Agile Development Story" Vuong T. Ly, GSFC, NASA
2:15 – 3:00pm	"Smashing the Stovepipe" Paul Swenson, ASRC Federal Space & Defense
3:00 – 3:30pm	Break
3:30 – 4:00pm	"Re-visit Agile Lessons Learned from previous GSAW" Supannika Mobasser, The Aerospace Corporation

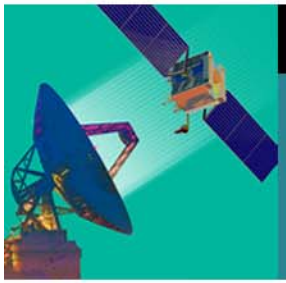
4:00 – 5:00pm	General discussions
	• Agile Acquisition Models
	  • Milestones and Deliverables; Roles and Responsibilities
	• Agile and other disciplines
	  • MBSE, hardware-intensive subsystem, accreditation, etc.
	• How to measure success
	• Trust management



Schedule

Time	Presentation and Discussion
1:00 – 1:30pm	Session Overview
1:30 – 2:15pm	"Smashing the Stovepipe" Paul Swenson, ASRC Federal Space & Defense
2:15 – 3:00pm	"GMSEC Services Suite - An Agile Development Story" Vuong T. Ly, GSFC, NASA
3:00 – 3:30pm	Break
3:30 – 4:00pm	"Re-visit Agile Lessons Learned from previous GSAW" Supannika Mobasser, The Aerospace Corporation

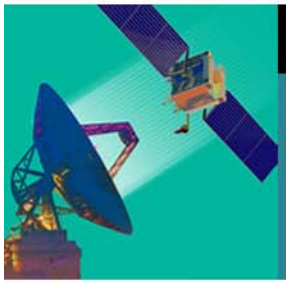
4:00 – 5:00pm	General discussions
	• Agile Acquisition Models
	  • Milestones and Deliverables; Roles and Responsibilities
	• Agile and other disciplines
	  • MBSE, hardware-intensive subsystem, accreditation, etc.
	• How to measure success
	• Trust management



Schedule

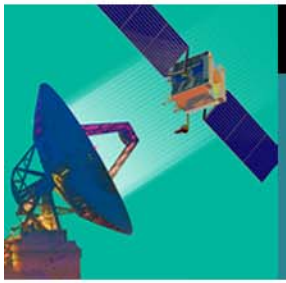
Time	Presentation and Discussion
1:00 – 1:30pm	Session Overview
1:30 – 2:15pm	"Smashing the Stovepipe" Paul Swenson, ASRC Federal Space & Defense
2:15 – 3:00pm	"GMSEC Services Suite - An Agile Development Story" Vuong T. Ly, GSFC, NASA
3:00 – 3:30pm	Break
3:30 – 4:00pm	"Re-visit Agile Lessons Learned from previous GSAW" Supannika Mobasser, The Aerospace Corporation

4:00 – 5:00pm	General discussions
	• Agile Acquisition Models
	  • Milestones and Deliverables; Roles and Responsibilities
	• Agile and other disciplines
	  • MBSE, hardware-intensive subsystem, accreditation, etc.
	• How to measure success
	• Trust management



Agile Working Groups in the past GSAWs

- **2015 : Lessons learned on Agile Ground Software Development**
  - Interactive discussions on various topics
- **2016 : Challenges and Experiences of Adopting Agile Ground Software Development**
  - Two presentations
    - Embracing Change to Overcome Challenges with Large-Scale Agile Software Development
    - Agile Fit Check
  - Interactive discussions on 3 main topics
- **2017 : Adopting Agile Ground Software Development**
  - Two presentations
  - Re-visit the lessons learned
  - Interactive discussions



2015 : Lessons learned on Agile Ground Software Development

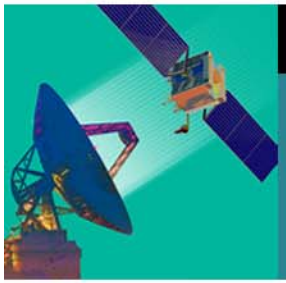
Key Takeaways

Interactions between contractors and government

- Additional roles : a task manager, a process police, a technical proxy
- Interaction frequency : As frequent as needed
- Do not mix between agile and waterfall teams
- Ratio of government to developers : 5 – 10%

Requirements volatility and scope management

- Use product roadmap to map system level requirements
- Use capability-based (vertical), not requirement-based development
  - Throw away and rebuild if needed
- Use early sprints for infrastructure and just enough architecture
- Balance “–ilities” (performance) and architecture



2015 : Lessons learned on Agile Ground Software Development

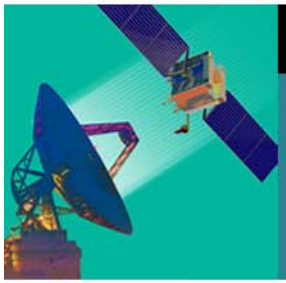
Key Takeaways

Useful Metrics

- Weekly quality / delivery matrix
- Planned vs. Actual requirements
- Projected vs. Actual burn down rate
- Agile EVM by release
- Cumulative Flow diagram
- Story aging

Testing and QA

- Incorporate definition of done and acceptance testing
- Keep track of technical debt
- Regression testing, automated testing with continuous integration
- Operational testing at the release level
- Code review at least at half way point
- System testing plan and strategy
  - use tools, test throughout the lifecycle



2015 : Lessons learned on Agile Ground Software Development

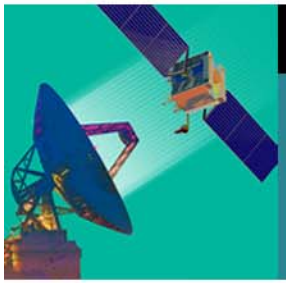
Key Takeaways

Contract types

- In the contract, instead of specifying the process, such as agile or scrum, that the contractor needs to follow, the contract should rather address agile principles or required practices
- Be careful on what to incentivize, pick the right goals such as performance objectives to get to the next contract
- In various cases, products can be delivered but not deployable

Source Selection

- Check past performance
- Scrum master should be certified
- Potential contractor must be able to describe how they are going to apply agile
- Let the contractor pick the process
- Evaluate based on the practices that the government required
  - such as continuous integration, test-driven development
- Avoid putting inhibitors to agile approach
  - such as Preliminary Design Review (PDR), Critical Design Review (CDR)



2015 : Lessons learned on Agile Ground Software Development

Key Takeaways

Schedules and milestone decisions

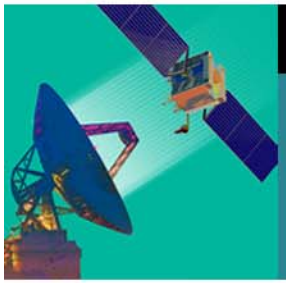
- Sprint length depends on how long can you freeze the requirements
- Popular sprint lengths
  - 4-6 weeks with weekly quality/delivery matrix
  - 2-3 weeks with close-to-as-built documents at 12 weeks-release
- Dedicate a sprint to fix bugs and/or to resolve technical debt

Deliverables

- Document : automate as much as possible to reduce repetitiveness
  - CDRL : Need to be incremental and less formal, focus on as-built deliverables
- Focus on code sustainment
  - Test cases, test reports, test analysis
  - Environments and infrastructure

Personnel turnover

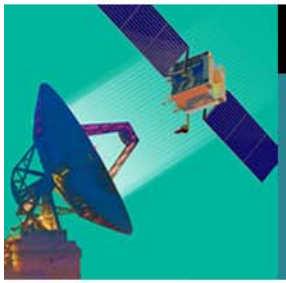
- Encourage on-going training
- Create team culture to ensure sustainable process and environments
- Use project workbook to share knowledge



2016 : Challenges and Experiences of Adopting Agile Ground Software Development

Key Takeaways

- Teach Agile in ACQ 101
  - Have a proper agile training for both acquirer and developer
- Need a supportive collaboration infrastructure
  - All about people: people have to work together and trust each other
- Plan for evolving requirements
- Having open discussions
  - Contractors tell the Government exactly what you want
  - Increase face time between Government and Contractor
- During source selection
  - Ask the right questions on how the contractor applies agile practices
- CDRLs: consider tailoring up instead of tailoring down
- Better understand what we want to measure and how to measure them

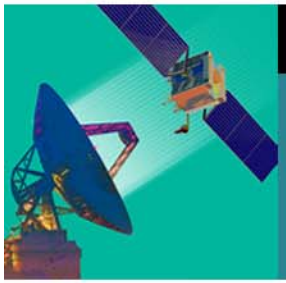


2016 : Challenges and Experiences of Adopting Agile Ground Software Development

Key Takeaways

Failed Agile Attempts

- Learn Agile on the fly
- Solely rely on individuals and tacit/tribal knowledge
- Lack of integration infrastructure to support agile teams
- Not enough system and software engineering to begin development
- Developing overlapping capability threads
- Use selection criteria such as lowest-cost technically acceptable
- Handle performance requirements as a user story
  - Assign performance requirement to a sprint
  - Tacking on Information Assurance at the end
- User stories not ready when sprint planning begins
- Not having a definition of done

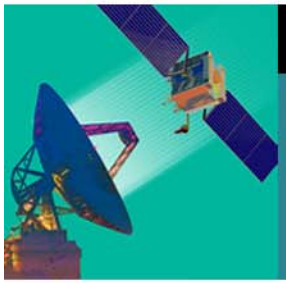


2016 : Challenges and Experiences of Adopting Agile Ground Software Development

Key Takeaways

Pains, struggles, and barriers in adopting agile ground software development

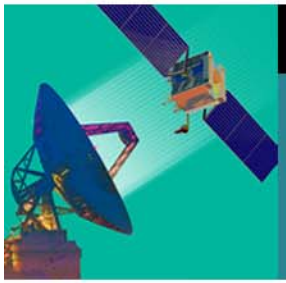
- Disconnect between Government processes (acquisition, contracts, security A&A) vs. the tempo of Agile
- Agile doesn't scale well using current best practices, “A Bridge Too Far”
- The government business model doesn’t always align with the principles and values of the agile manifesto (predefined scope and delivery)
- Understanding how to measure “done” and the metrics to determine an equivalent to earned value
- High personnel turnover can impede effectiveness of agile teams
- Rigidity of user story
  - Some user stories can’t really fit in a sprint, need a flexible sprint
- Not enough of continuous integration and testing



Schedule

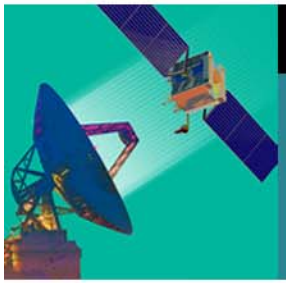
Time	Presentation and Discussion
1:00 – 1:30pm	Session Overview
1:30 – 2:15pm	"Smashing the Stovepipe" Paul Swenson, ASRC Federal Space & Defense
2:15 – 3:00pm	"GMSEC Services Suite - An Agile Development Story" Vuong T. Ly, GSFC, NASA
3:00 – 3:30pm	Break
3:30 – 4:00pm	"Re-visit Agile Lessons Learned from previous GSAW" Supannika Mobasser, The Aerospace Corporation

4:00 – 5:00pm	General discussions
	• Agile Acquisition Models
	  • Milestones and Deliverables; Roles and Responsibilities
	• Agile and other disciplines
	  • MBSE, hardware-intensive subsystem, accreditation, etc.
	• How to measure success
	• Trust management

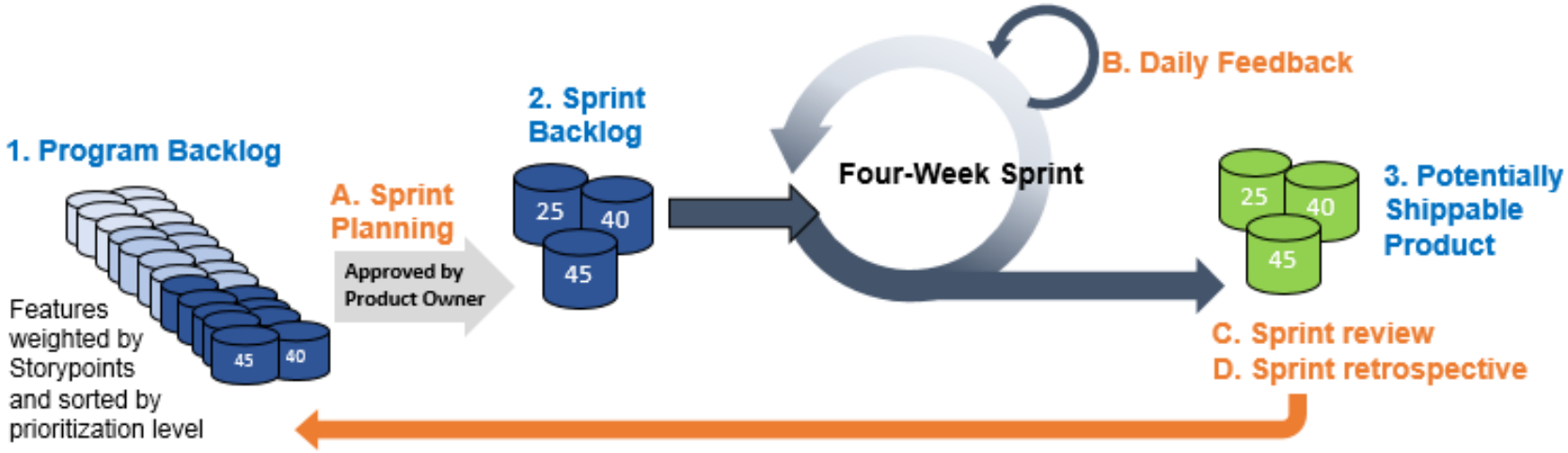


Open discussions

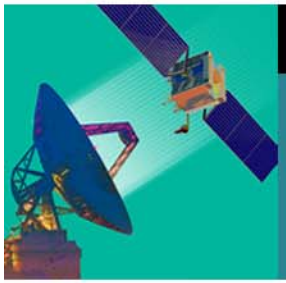
- Time to share your experiences about agile adoption
- 4 topics
  - Agile Acquisition Models
    - Milestones and Deliverables; Roles and Responsibilities
  - Agile and other disciplines
    - MBSE, hardware-intensive subsystem, accreditation, etc.
  - How to measure success
  - Trust management



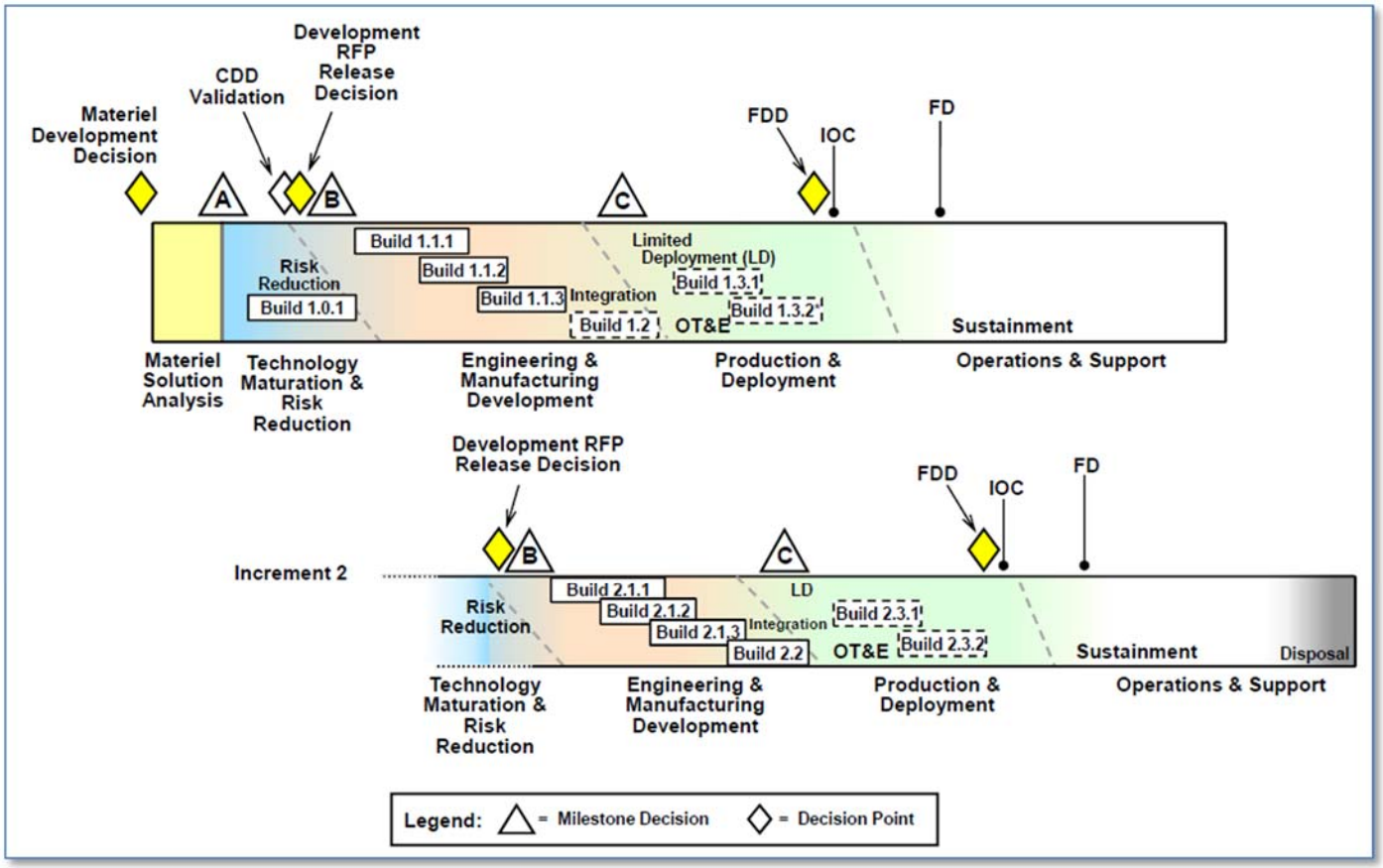
Agile Acquisition Models



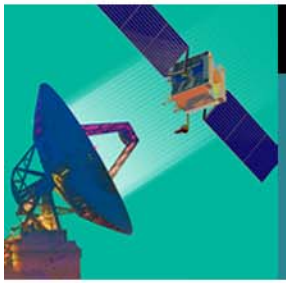
Scrum



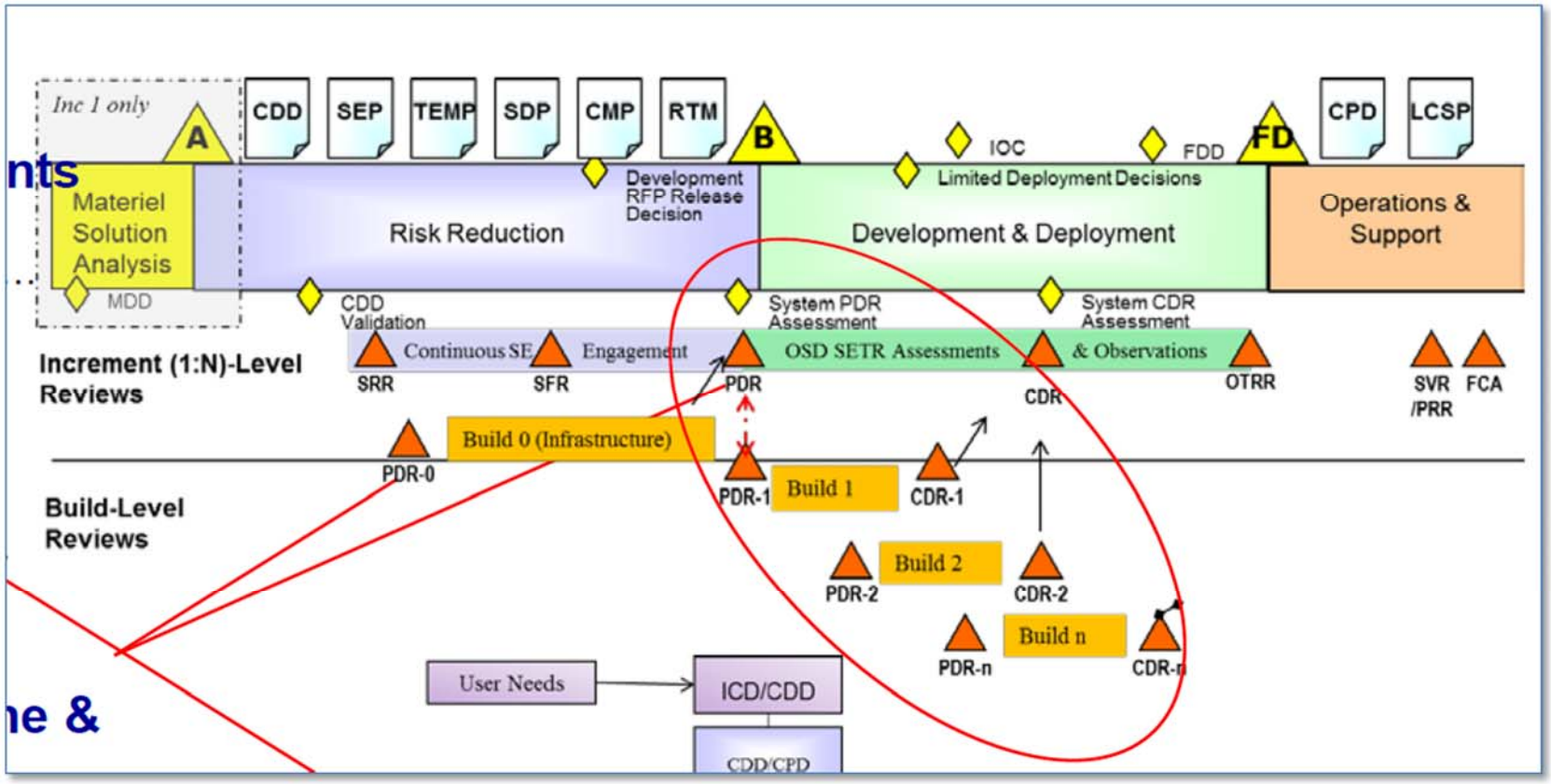
Agile Acquisition Models



DoD 5000.0 Hybrid Program B (Software Dominant)

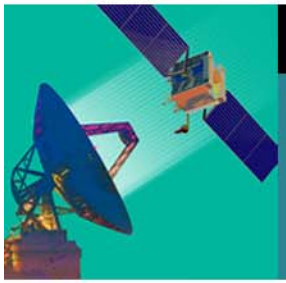


Agile Acquisition Models

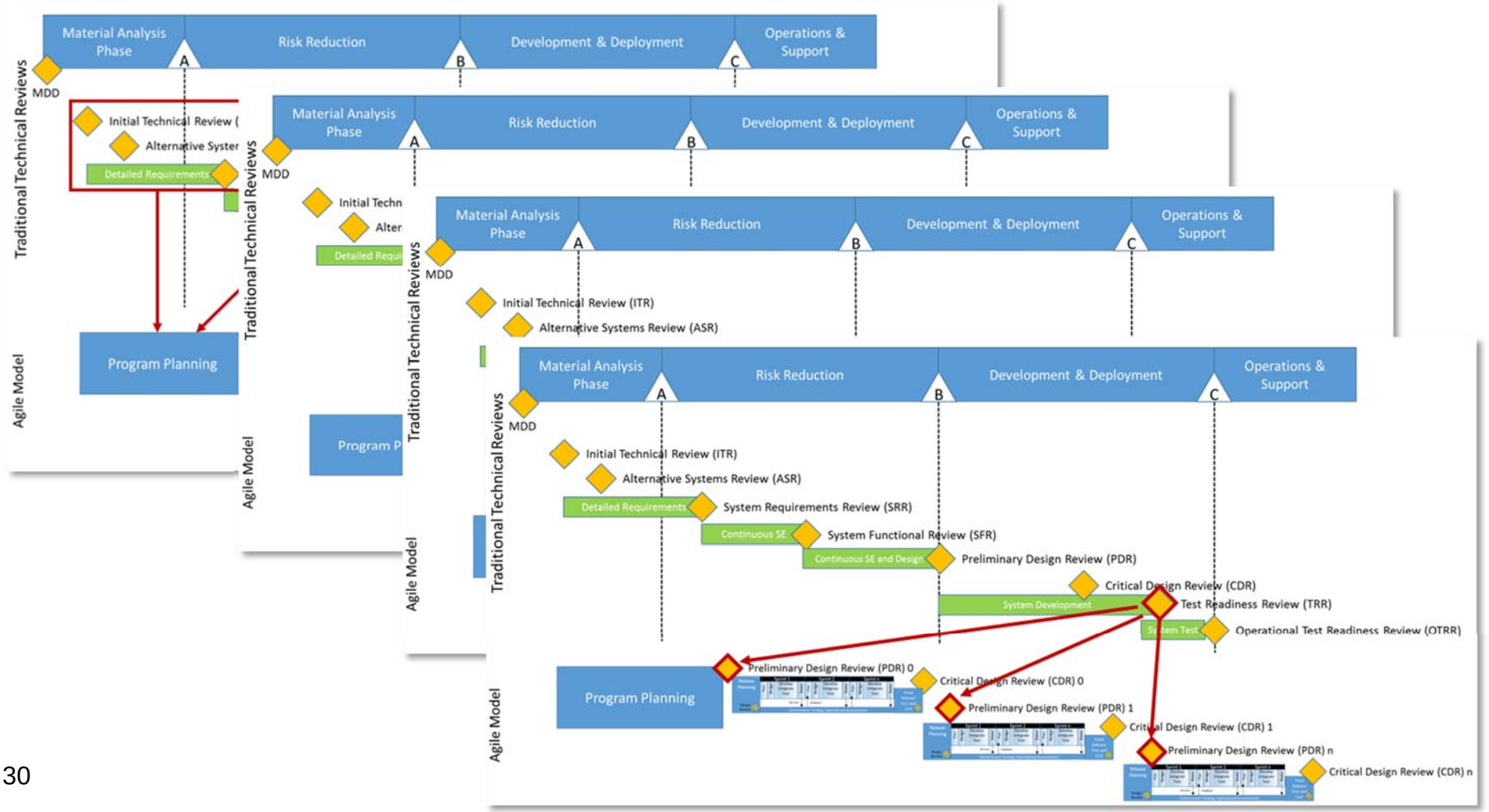


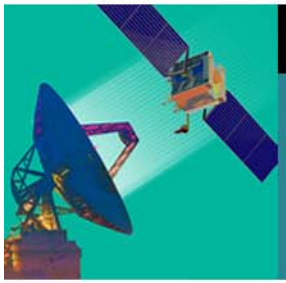
Ref: Brady, Brady, " Agile and Incremental Software Development in the Defense Acquisition System", NDIA, Oct 29, 2015

29



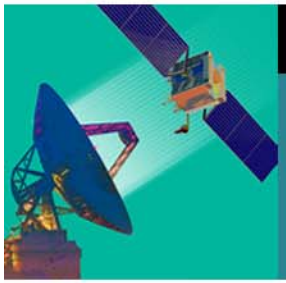
Agile Acquisition Models





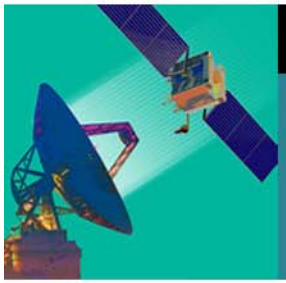
Agile Acquisition Models

- Any other acquisition models you have seen?
  - Good and not good
- What are the concerns or challenges?



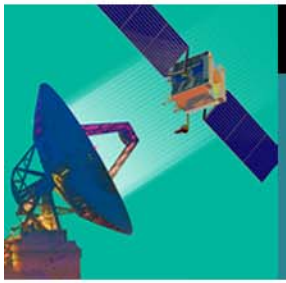
Roles and Responsibilities

- Scrum roles: Scrum Master, Product Owner, Development Team
- GSAW 2015 suggested: a task manager, a process police, a technical proxy
- Who should be the product owner in this process, the contractor, the government team, or a chosen representative?
  - Pros & cons
- Any other essential roles?
- What is the role of testing in Agile? How does this fit into testing of acquired space systems? What if you have an independent tester?
- What does agile leadership look like? How is this similar to / different from traditional leadership?



Agile and other disciplines

- MBSE – Model-based Systems Engineering
  - Such as requirements, diagrams, simulations, prototype
  - How can we apply MBSE in an agile program?
  - Any challenges?
  - What do you have to do differently?
- Hardware-intensive development
  - Do you have to complete the requirements and design before coding?
  - How do the milestones or synchronization points look like?
  - Any challenges?
  - What do you have to do differently?
- Accreditation / Certification
  - Require additional processes, documents?
  - Any challenges?
  - What do you have to do differently?

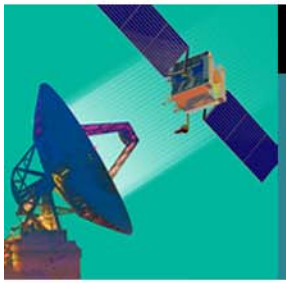


Metrics

- What's the right metrics to use in an Agile team on day-by-day basis?

Velocity	57%
Iteration burndown	51%
Release burndown	41%
Planned vs. actual stories per iteration	37%
Burn-up chart	34%
Work-in-Process (WIP)	30%
Planned vs. actual release dates	29%
Customer/user satisfaction	29%
Defects in to production	28%
Defects over time	23%
Budget vs. actual cost	23%
Business value delivered	21%
Defect resolution	20%
Cycle time	19%
Estimation accuracy	18%
Individual hours per iteration/week	18%
Test pass/fail over time	17%
Scope change in a release	17%
Cumulative flow chart	15%
Earned value	10%
Customer retention	7%
Revenue/sales impact	7%
Product utilization	7%

[VersionOne 2016]

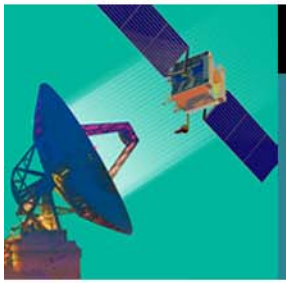


How to measure success?

- Which metrics that the upper management should look at?
- How to measure success?

On-time delivery	58%
Product quality	48%
Customer/user satisfaction	46%
Business value	46%
Product scope (features, requirements)	36%
Productivity	31%
Project visibility	30%
Predictability	26%
Process improvement	24%

[VersionOne 2016]



Trust Management

- What should the government team do to get the project visibility but not to step on the contractor's toes?
- How can Agile help in increasing transparency between the government team and the contractor?
- What would the contractor expect from the government? And what would the government expect from the contractor?

Ground System Architectures Workshop

Session 11E

Adopting Agile Ground Software Development

Supannika Mobasser

supannika.k.mobasser@aero.org

The Aerospace Corporation

AEROSPACE