Agile Innovation for Enterprise Integration

GSAW 2022
Working Group H

Jodene Sasine, Jason McKenney,
Donald Denham, Michael Roberts, Eric McCary
The Aerospace Corporation

Approved for public release. OTR 2022-00370.

Schedule

Time	Topic
11:00 – 11:10am	Overview and Introductions
11:10 – 11:40am	Digital Engineering (DE) and Agile
11:40 – 12:20pm	Hardware Development with Agile
12:20 – 12:55pm	Cybersecurity with Agile and DevSecOps
12:55 – 1:30pm	Continuous Integration with Agile / DevSecOps Development

Overview and Introductions

- Agile acquisition has significant challenges for the Government sector as opposed to the commercial software intensive industry
  - *How do we adapt Agile concepts to the Ground System Acquisition lifecycle?*

- This working group provides an opportunity for Agile practitioners to share their experiences and learn from others on several topics concerning integration
  - *Digital Engineering (DE) and Agile*
  - *Hardware Development with Agile*
  - *Cybersecurity with Agile and DevSecOps*
  - *Continuous Integration (CI) with Agile / DevSecOps Development*

- Utilize the Chat to tell us about yourself
  - *Your name*
  - *Your company/organization*
  - *An expectation you have for this working group*

Feel Free to Take a Break when Needed!

Digital Engineering (DE) and Agile

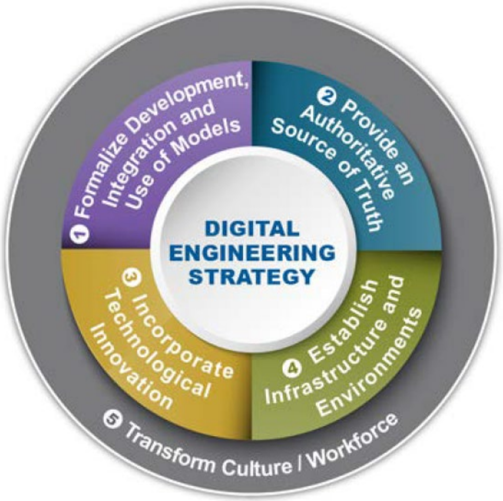
For Ground Software systems, how to integrate Agile methods with Digital Engineering?

Jodene Sasine

DE and Agile

DoD Digital Engineering Strategy

Integrated digital approach that uses authoritative sources of systems data and models as a continuum across disciplines to support lifecycle activities from concept through disposal


DIGITAL ENGINEERING STRATEGY
1 Formalize Development, Integration and Use of Models
2 Provide an Authoritative Source of Truth
3 Incorporate Technological Innovation
4 Establish Infrastructure and Environments
5 Transform Culture / Workforce

Department of Defense Digital Engineering Strategy June 2018

Agile Manifesto

"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		Comprehensive **documentation**
Customer **collaboration**		**Contract** negotiation
Responding to **change**		Following a **plan**

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

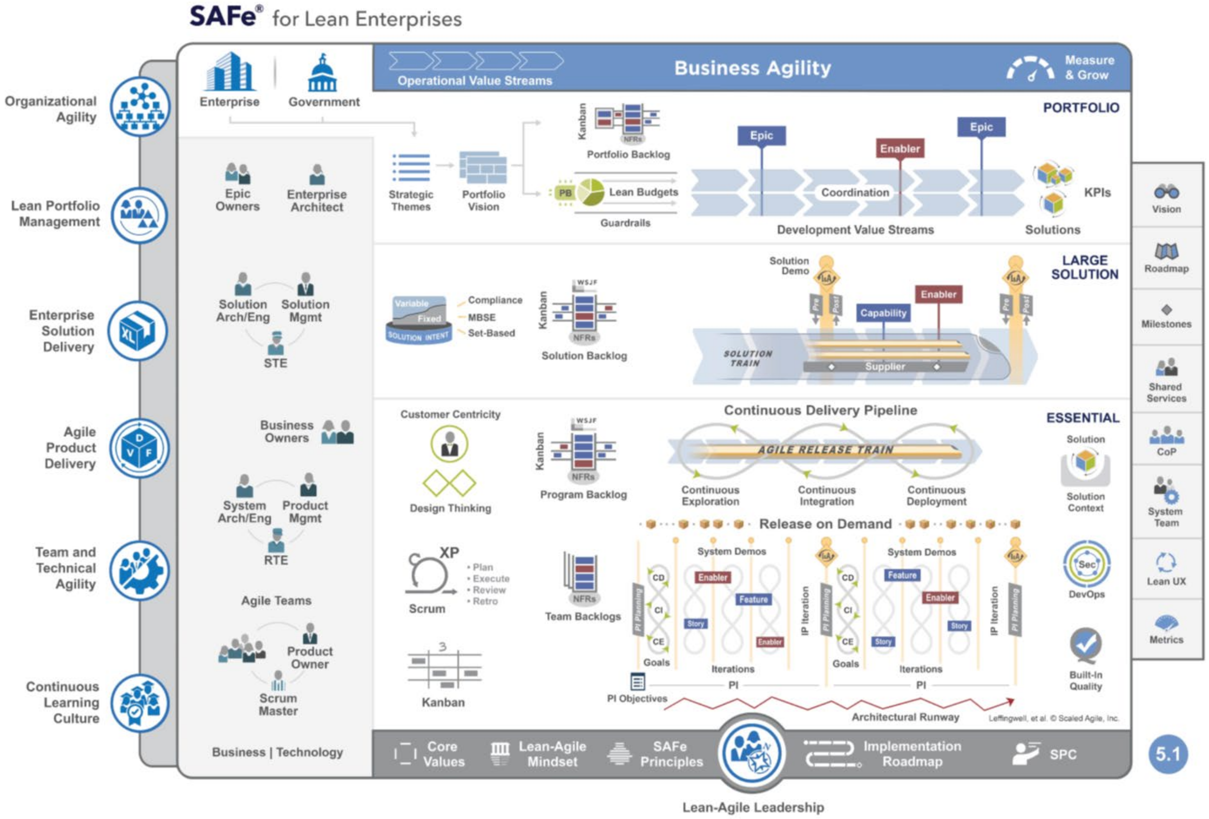
http://www.agilemanifesto.org

The big question is how do we coalesce Digital Engineering with Agile Processes?

DE and Scaled Agile Framework (SAFe)

Large Solution Configuration

- MBSE, an enabler of DE, integrated at Solution Train level
- Model information recorded as part of Solution Intent via Enablers
- Emphasize continuity of use of models across the lifecycle



Source: ***https://www.scaledagileframework.com/posters/***

SAFe® | PROVIDED BY SCALED AGILE

Integrating DE on an Agile Software Program

One Option: Independent DE-specific Agile Teams

- What is the role composition of the DE team (i.e., SE, Modeler, etc…)?
  - ***Working Group Inputs:***
- How many DE teams are there?
  - ***Working Group Inputs:***
- Are DE Teams Sprinting? Kanbaning?
  - ***Working Group Inputs:***

Working Group Participation greatly encouraged!

Integrating DE on an Agile Software Program

One Option: Independent DE-specific Agile Teams

- How do DE Teams get ahead of development teams?
  - *How much architectural runway is created before development starts?*
  - *What artifacts should be available?*
  - ***Working Group Inputs:***

- How do DE Teams collaborate/communicate/stay in sync with other Agile Teams (i.e., Development, I&T)?
  - ***Working Group Inputs:***

Working Group Participation greatly encouraged!

Integrating DE on an Agile Software Program

Another Option: DE resources included in other Agile Teams

- How many DE resources are on a team?
  - ***Working Group Inputs:***
- How does the architectural runway get ahead of development?
  - *What artifacts should be available?*
  - ***Working Group Inputs:***
- Any lessons learned, good and bad? Best practices?
  - ***Working Group Inputs:***

Working Group Participation greatly encouraged!

Integrating DE on an Agile Software Program

Conclusions

- What are your experiences on integrating Agile with DE? What was painful? What made you happy? What surprised you?
  - ***Working Group Inputs:***

Working Group Participation greatly encouraged!

Hardware Development with Agile

Integrating Agile Methodologies into Hardware Development Projects

Donald Denham

Background

Donald Denham

- *Education*
  - BSAE – SDSU in aerospace structures
  - MSEE – USC in control theory
- *Senior Engineering Specialist in the Electromechanical Controls Department*
- *30 Years in the industry (21 Years Real-Time Embedded SW)*
  - Hughes Missile Systems / Raytheon – 8 years
  - General Dynamics Ordnance and Tactical Systems – 11 Years
  - Enphase Energy – 2 years
  - Crane Aerospace – 4 years
  - Aerospace – 5 years

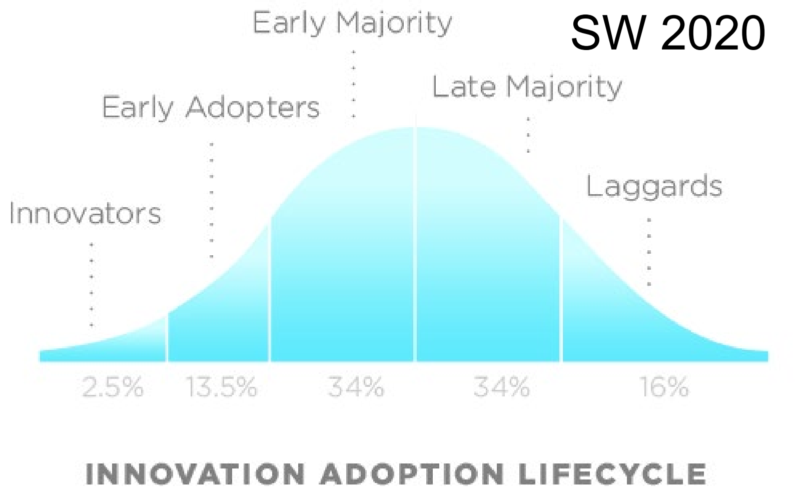
First used iterative short design cycle SW development in 1997

- *Followed a major SW development project failure*
- *Predated Agile by 7 years*

HW Development and SAFe

- In the early 2000's, software development was the value bottleneck
- In the 2020's, hardware development has become the value bottleneck
- Companies that embrace rapid prototyping, test to failure, continuous production agility, and other techniques will bring value to the market.
  - *SpaceX Falcon 9 and Starship development versus NASA/Boeing SLS*
  - *SpaceX Dragon Capsule versus Boeing Starliner*

HW 2020
SW 2000

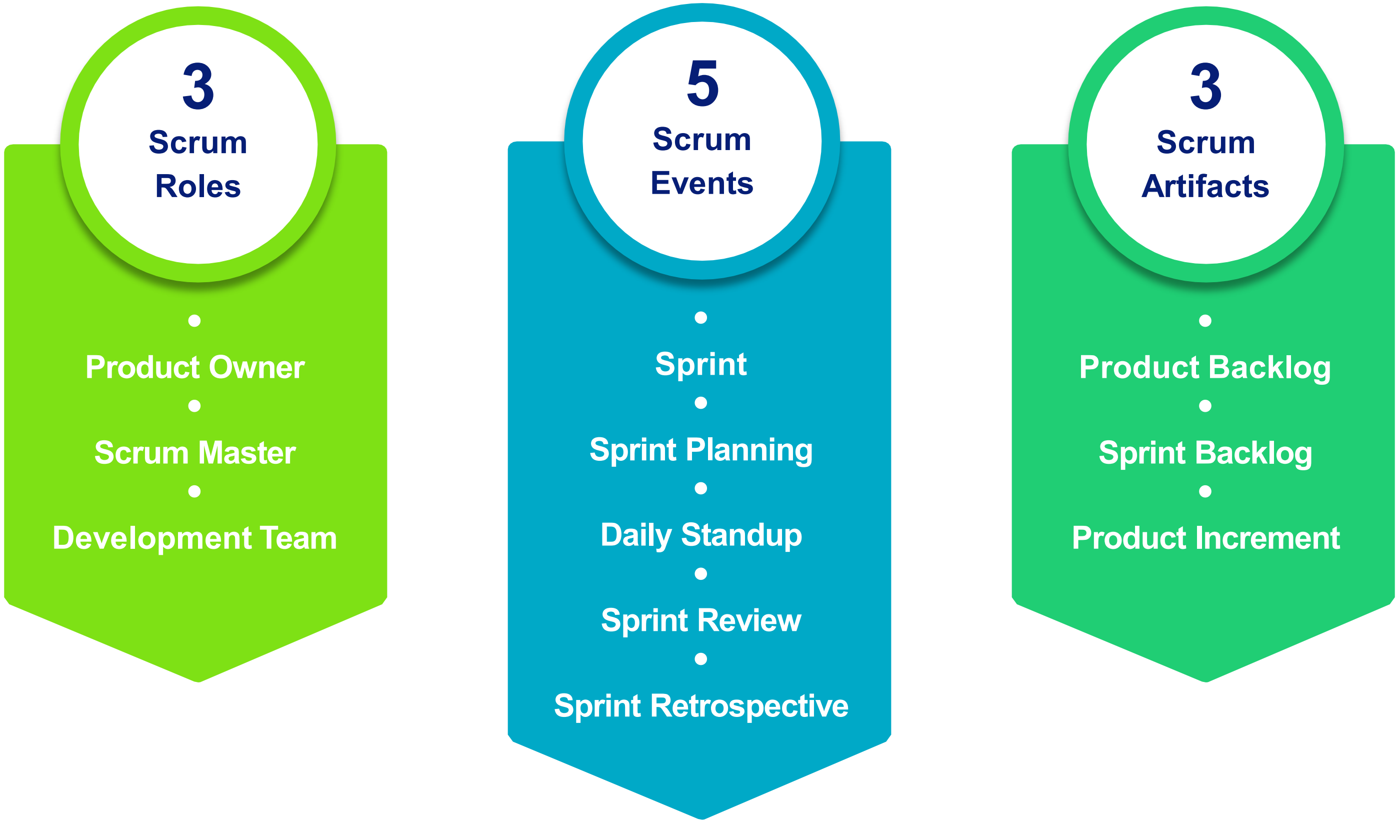


SW 2020

Source: https://commons.wikimedia.org/wiki/File:DiffusionOfInnovation.png

HW Agile/Scrum 3-5-3

Is it really all-or-nothing…



3 Scrum Roles

- Product Owner
- Scrum Master
- Development Team

5 Scrum Events

- Sprint
- Sprint Planning
- Daily Standup
- Sprint Review
- Sprint Retrospective

3 Scrum Artifacts

- Product Backlog
- Sprint Backlog
- Product Increment

Most teams struggle at the start to fill in all the boxes. Don’t despair! Use the retrospective process to Scrum the Scrum! Add tasks to improve the Scrum.

Comparing HW and SW Agile Processes

The short and incomplete list

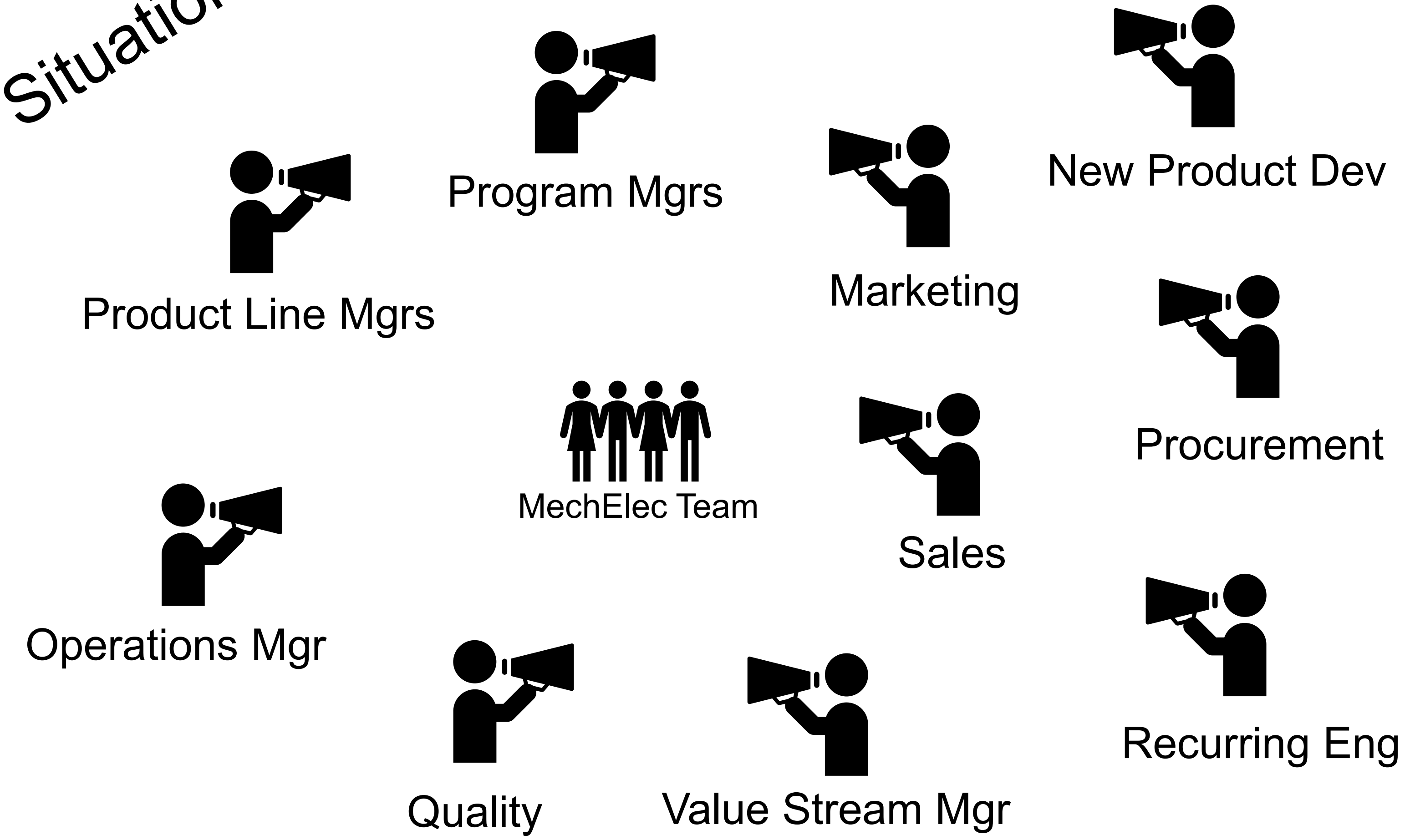
Agile Element	Software	Hardware
User Stories	Directly added to backlog	Must be broken down for cross-functional team tasking
Backlog	The current list of Epics, User stories, and Tasks	Current list of tasks and their interrelationships
Sprints	1-4 Weeks with end of sprint demos	1-4 weeks with a path to near-term prototypes, demos, mockups
Team	SW Dev Team	Cross-functional team
Process Owner	Scrum Master	Scrum Master
Customer Rep	Product Owner	Product Owner

The Agile process is not the same for HW and SW projects. It is still Agile!

Case Study #1 – Mechanical and Electrical Engineering (2016)

“How to get more done with the same team”

The Situation



Program Mgrs

New Product Dev

Product Line Mgrs

Marketing

MechElec Team

Procurement

Sales

Operations Mgr

Recurring Eng

Quality

Value Stream Mgr

Every actor felt empowered to redirect the team at any moment!

Case Study #1 – Mechanical and Electrical Engineering

"How to get more done with the same team"

KC Green

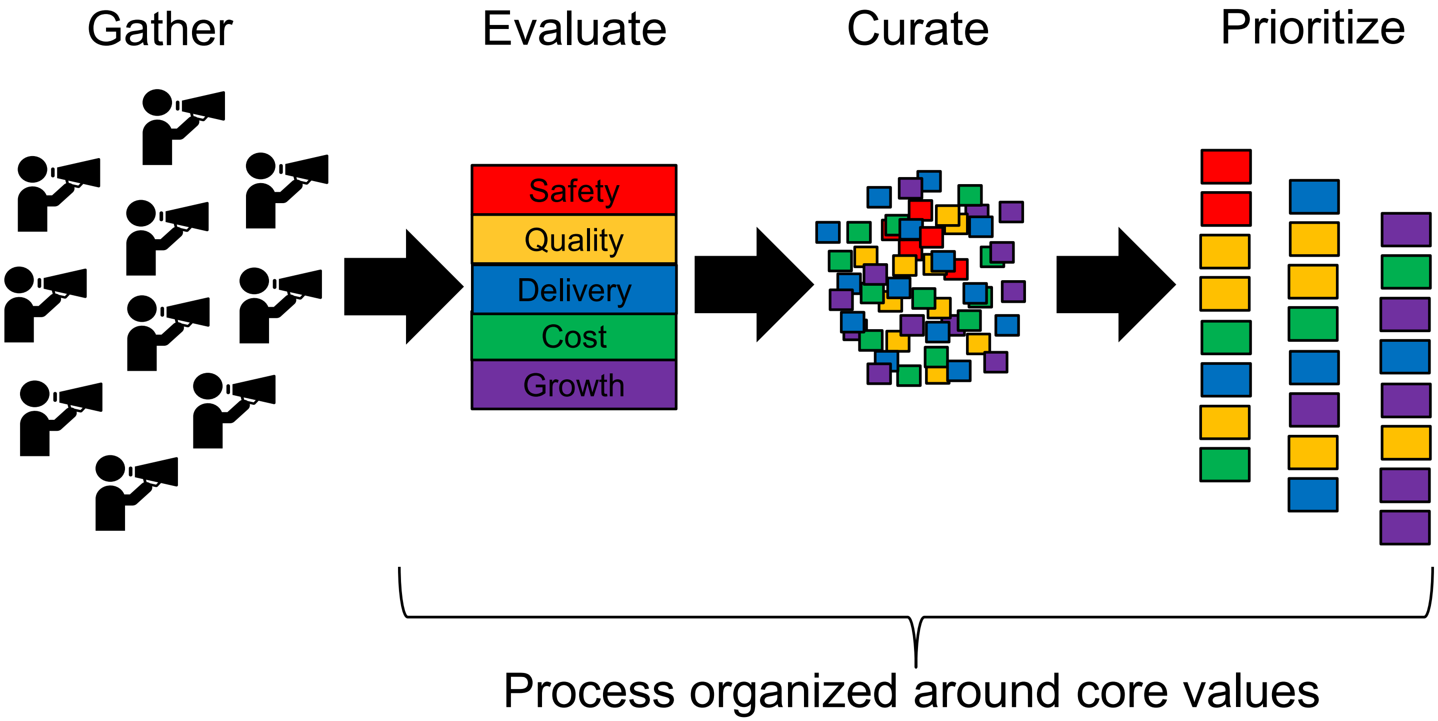
No prioritization of the issues that were coming to the team.

Case Study #1 – Mechanical and Electrical Engineering

“How to get more done with the same team”

Okay! What to do about it?

1) Generate a comprehensive list of everything the team is being asked to do.

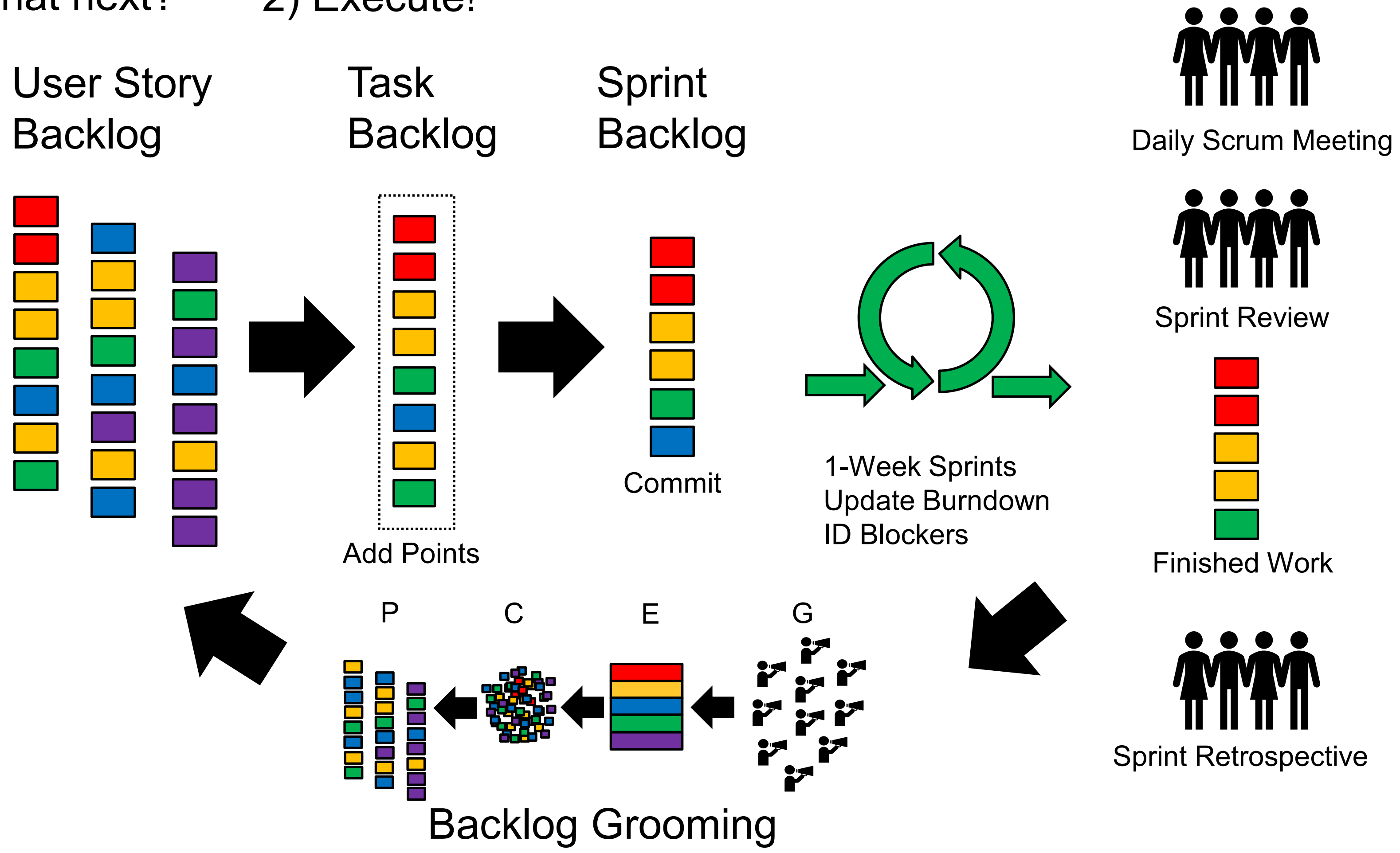


This initial process is the same for SW or HW projects.

Case Study #1 – Mechanical and Electrical Engineering

"How to get more done with the same team"

What next? 2) Execute!



User Story Backlog

Task Backlog

Sprint Backlog

Add Points

Commit

1-Week Sprints
Update Burndown
ID Blockers

Daily Scrum Meeting

Sprint Review

Finished Work

Sprint Retrospective

P C E G

Backlog Grooming

Sprint work M-1/2F. Sprint Review, Retrospective, Backlog Grooming, Task Breakout, and Sprint Backlog on Fridays.

Case Study #1 – Mechanical and Electrical Engineering

How did we do?

Problems	Start	End
Safety Issues	3	0
Open ECOs	89	17
ECO Closure	91 Days average	21 Days Average
ECO Oldest	300 days	300 days
Work Cell 1		50% Takt
Work Cell 2		75% Takt
Cost Reductions	~$50K	~$1M
I18N Drawing Backlog	~250	~200
I18N Drawings updated	100/year	400/Year
Next Gen Products	Gen 2	Gen 3
Continuous Life Testing	0 UUT	6 UUT
Kaizens Supported	1	5
Lab 5S Program	0S	3S

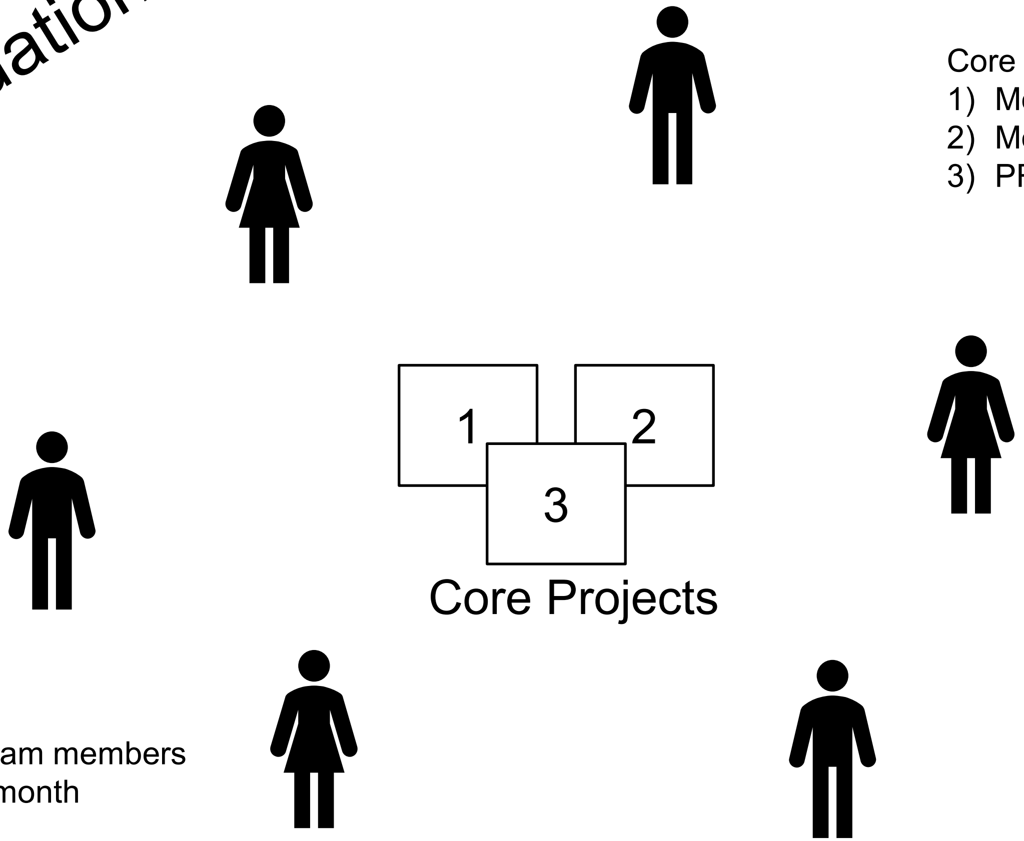
The mechanical and electrical teams were fully engaged, highly motivated, and morale was high!

Case Study #2 – SERPA Cross Functional Team

“How to get anything done with an ever-changing team”

The Situation

Core Projects:
1) Motor Test Bed
2) Motor Design
3) PRT/ITS Test Bed



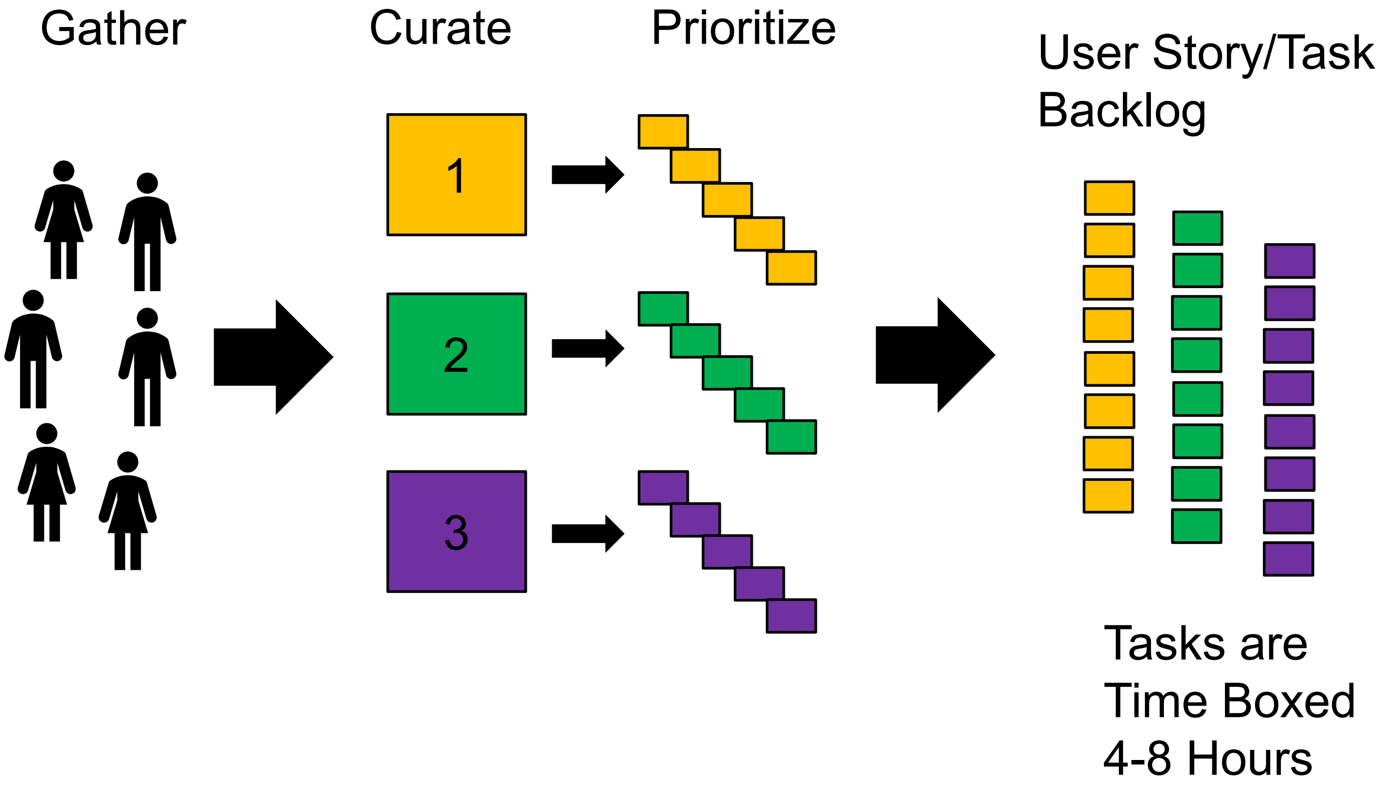
6 – 10 Active team members
80 STE Hours/month

SERPA = Sustained Experimentation and Research for Program Applications

Limited availability of team 0-10 Hours per sprint to work on core projects.

Case Study #2 – SERPA Cross Functional Team

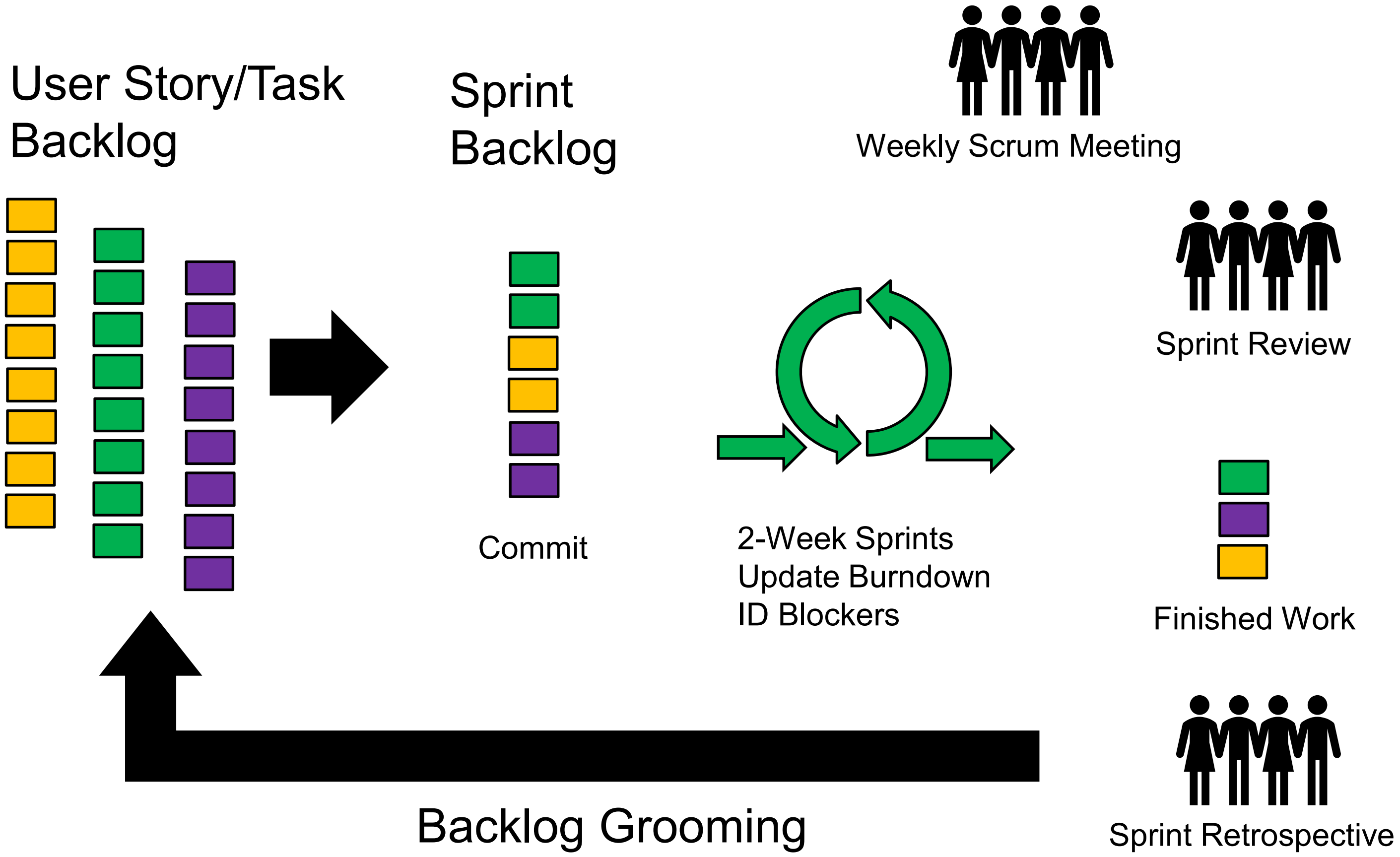
“How to get anything done with an ever-changing team”



Limited availability required the team to breakdown user stories to single step task lists.

Case Study #2 – SERPA Cross Functional Team

"How to get more done with an ever-changing team"



User Story/Task Backlog

Sprint Backlog

Weekly Scrum Meeting

Sprint Review

Commit

2-Week Sprints
Update Burndown
ID Blockers

Finished Work

Backlog Grooming

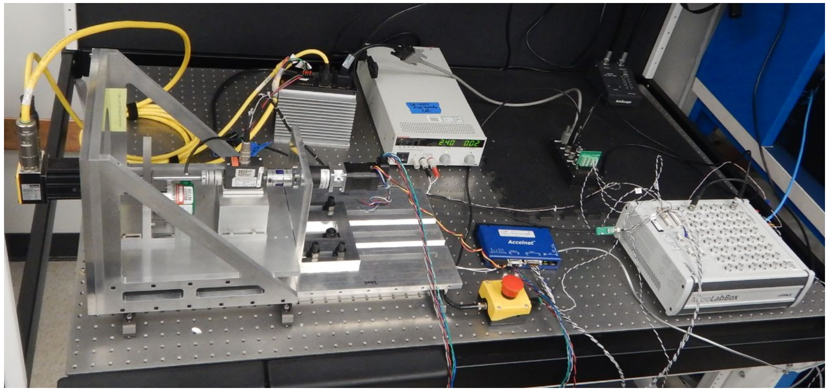
Sprint Retrospective

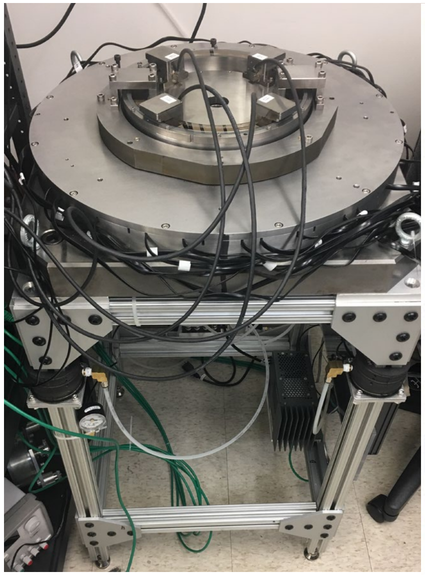
The highly dynamic nature of the team's availability typically leads to overcommitting.

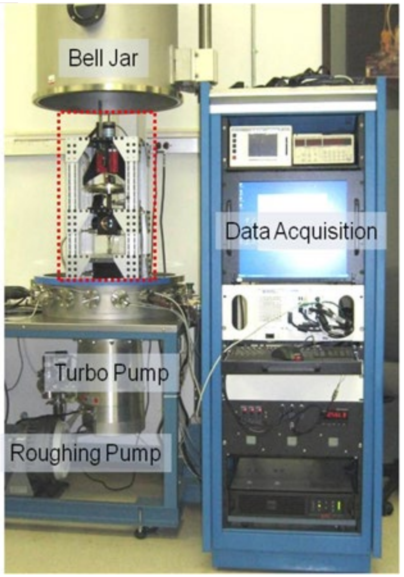
Case Study #2 – SERPA Cross Functional Team

How are we doing?

- Multi-year developments providing engineering expertise and capabilities
- Testbeds
  - *Bearing Thermal TB*
  - *Rotational Sensor TB*
  - *Harmonic Drive TB*
  - *Precision Rotation Table TB*
  - *Inductosyn TB*
  - *Motor TB*
- Modeling
  - *Bearing Thermal Conductance Model*
  - *Harmonic Gearbox Model*
  - *High Fidelity Motor Simulation*







Over 7 years the SERPA has provided hands on experience to more than 50 MTS in the Guidance and Controls Subdivision.

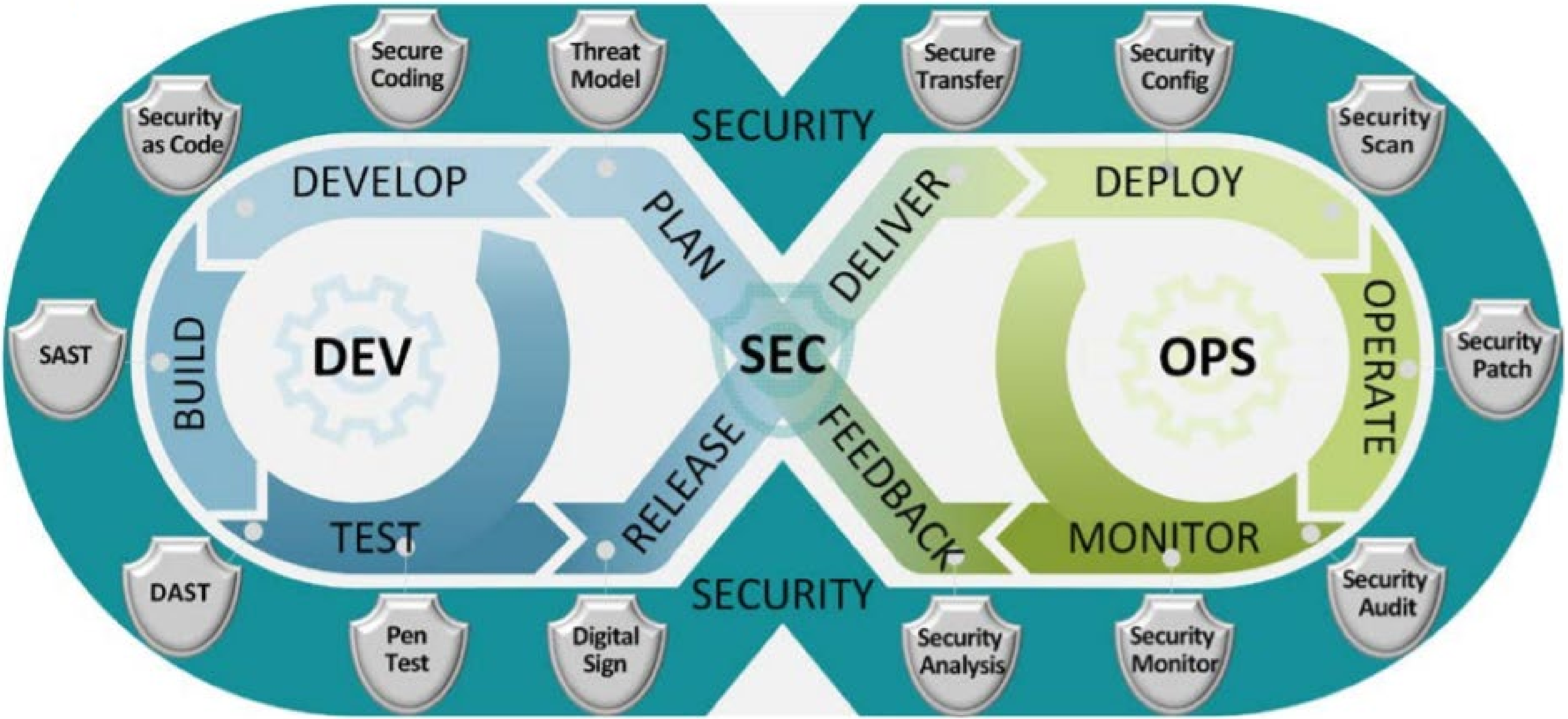
Resources and References

- Jeffrey Liker (2004), *The Toyota Way.* McGraw-Hill.
- Jeff Sutherland (2014), *Scrum: The Art of Doing Twice the Work in Half the Time.* Crown Business.
- J. J. Sutherland (2019), *The Scrum Fieldbook: A Master Class on Accelerating Performance, Getting Results, and Defining the Future*, Currency.
- https://www.scaledagileframework.com/applying-safe-to-hardware-development/
- https://www.scruminc.com/the-3-5-3-of-scrum/
- https://agileforhardware.org/wp-content/uploads/2018/07/An-Intro-to-MAHD-Ebook-Final-7_25_18.pdf

Cybersecurity with Agile and DevSecOps

Mike Robert, Eric McCary

DevSecOps Security Activities



DoD Enterprise DevSecOps Reference Design

Example Security Best Practices

- Security scan and test automation (e.g. static analysis, dynamic analysis, penetration testing)
- Create a continuous feedback loop of threat intelligence and report generation/remediation
  - *Enable developers to remediate vulnerabilities as they're discovered*
- Monitor and assess the effectiveness of security policies
  - *Build checks into the process*
- Convert taxing security tasks into automated scripts, with human sign-off on results if necessary
- Implement validation cycles which test the accuracy and efficiency of test scripts
  - *Configure scripts so that they can be replicated across different projects.*
- Test new code within a staging environment
  - *Allow low-impact failure (remediate afterwards)*
  - *Test on every code commit*
- Monitoring should produce logs and transcripts of applicable security and anomalous events create logs (or red flags)

Security Authorization

- ATO: Authorization granted by a AO/Designated Approving Authority (DAA) for a DoD IS to process, store, or transmit information; an ATO indicates a DoD IS has adequately implemented all assigned cybersecurity controls to the point where residual risk is acceptable to the DAA. ATOs may be issued for up to three (3) years.

- Continuous ATO: Authorization is granted, similar to a traditional ATO, while shifting the paradigm from assessing only the final product to assessing the process used to develop the product.

Discussion Topics for Cyber Security

- What are some key challenges you’ve faced in applying security to the Agile process?
  - *E.g. developer resistance, lack of secure coding standards, failing to cyber “test like you fly”*
  - ***Working Group Inputs:***

- ATO Compliance vs Security
  - *What do developers/stakeholders typically see as the main value of securing a system?*
    - Achieving ATO compliance?
    - Prevention of cyber attacks?
  - *Do developers have confidence that meeting RMF controls will prevent cyber attacks?*
  - ***Working Group Inputs:***

Working Group Participation greatly encouraged!

Discussion Topics for Cyber Security

- How does continuous ATO impact the Agile process?
  - ***Working Group Inputs:***

- Team composition
  - *How should Agile teams incorporate security expertise?*
    - Dedicated security staff member(s)?
    - Security trained developers?
  - *How does team composition affect development?*
  - ***Working Group Inputs:***

Working Group Participation greatly encouraged!

Discussion Topics for Cyber Security

- How could security tooling in the DevSecOps pipeline be more effective for Agile developers?
  - *E.g. Do high false positive rates undermine confidence in findings?*
  - *E.g. are IDE security plugins typically easy to use?*
  - ***Working Group Inputs:***

Working Group Participation greatly encouraged!

Continuous Integration with Agile and DevSecOps Development

What best practices from commercial industries can the Government take advantage of?

Jason McKenney

Best Practices for Continuous Integration

What role does Agile play in CI?

- **Agile** = A set of values defined in the Agile Manifesto (2001).
- **Scrum** = A framework to implement those values.
- CI is a critical practice for Agile teams (traditional or scaled)
- Scrum model best supports the goal of CI/CD with sprints, co-location, incremental delivery, frequent builds, fast feedback loops, and emphasis on automated testing
- What other Agile frameworks support CI? Kanban? Scrum-but?
- Are there Ground System programs that won't work with Agile or CI?

Best Practices for Continuous Integration

Are these methods feasible for Ground System projects?

- Well-defined code branching strategies
  - *Keep use of branches to a minimum.*
  - *Have a **mainline**: a single branch of the project currently under development everyone should work off this*
  - *Daily commit to the mainline, preferably using a CI Server*
    - Developer first updates their working copy to match the mainline
    - Resolves any conflicts with the mainline
    - Builds on their local machine.
    - If the build passes, then they are free to commit to the mainline

Best Practices for Continuous Integration

Are these methods feasible for Ground System projects?

- Automate the Build
  - *Automated environments for builds have been around for years but can we improve how they are used?*
  - *Keep the build fast (under 10 minutes)*
  - *Include everything needed in the automated build*
  - *Self-testing build: include automated tests in the build process*
  - *Critical test failures should cause the build to fail*

Best Practices for Continuous Integration

- Use of Test-Driven Development (TDD) or Extreme Programming (XP)
  - *Write the tests first, then write the code to make them pass*
- Automated Test Suites and a Test Management System
  - *How to organize test cases*
- Fix broken builds immediately
  - *Nobody has a higher priority task than fixing the build (within reason)*
  - *Fastest way to fix a build is usually to revert to the latest commit from the mainline*

Best Practices for Continuous Integration

- Keep the build fast
  - *Use of a deployment pipeline with 2 stages (David Farley):*
    - Quick tests for stability when someone commits to the mainline
    - Longer tests when shared to additional teams/machines
    - Rapid vs Full pipeline (Google)
- Test in a clone of the Production Environment
  - *This has become easier with the use of VMs and containers or with Blue/Green Deployments*

Best Practices for Continuous Integration

- Define a Release and Rollback Strategy
  - *How can the use of Infrastructure as Code or GitOps add value?*
    - Provision, configure, manage infrastructure resources
  - *Define and follow a rollback strategy for all teams*
- Metrics and reporting
  - *What metrics provide the most value?*
  - *Lead time, MTTR, Failed builds, Test success ratios, bug churn, etc.*
- What other CI methods have shown benefits for Enterprise Integration?

Thank you for participating!