



SERC RESEARCH REVIEW 2024 | NOVEMBER 12, 2024

Space Systems Command (SSC) Military Communications & Positioning, Navigation and Timing Directorate – Mission Engineering and Integration of Emerging Technologies

WRT-1069

Funding Agency: US Space Force

PI: Michael Orosz (USC/ISI)

Collaborators: Lt Col John M. Turner (USSF), Maj Steven Beyer (USAF), Brian Duffy (USC/ISI) and Craig Charlton (USC/ISI)



DISTRIBUTION A: Approved for public release; distribution unlimited

Agenda

- Project Objectives and Timeline
- Current Project Status and Observations
- Next Steps

Project Overview

- **Funding agency:** U.S. Space Force and Space Systems Command, Military Communications & Positioning, Navigation, and Timing Program Executive Office (SSC/CG)
- **Team:**
 - USC Information Sciences Institute (USC/ISI)
 - Systems Engineering Research Center (SERC)
- **Period of Performance:** August 2016 – January 2025 + Three One-Year Options

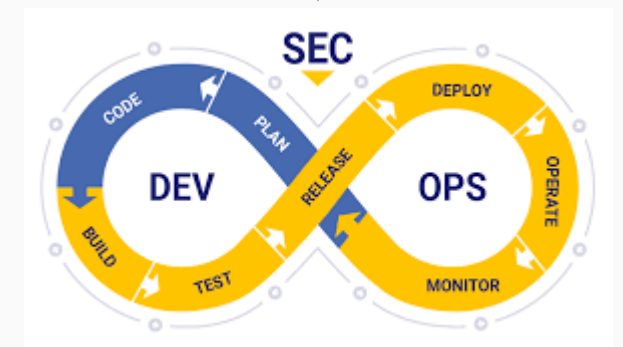


UNITED STATES
SPACE FORCE



Objectives

- **Improve U.S. Department of Defense (DoD) competitiveness:** Specifically - improve existing DoD space-based *software* system acquisition processes (i.e., the Software Acquisition Pathway)
- **Goals:**
 - Determine the mission engineering methods, analysis, and metrics to transition from traditional DoD 5000 waterfall development environments to agile/DevSecOps processes
 - Includes integration of emerging technologies and related education for the future workforce



Process

1. Understand the current acquisition environment
 - Immerse into environment (become part of the team)
2. Develop approaches to transition waterfall acquisition elements from DoDI 5000.02 to Agile/DevSecOps
...including workforce training
3. Incorporate processes and “lessons-learned” into a transition process to apply to other domains

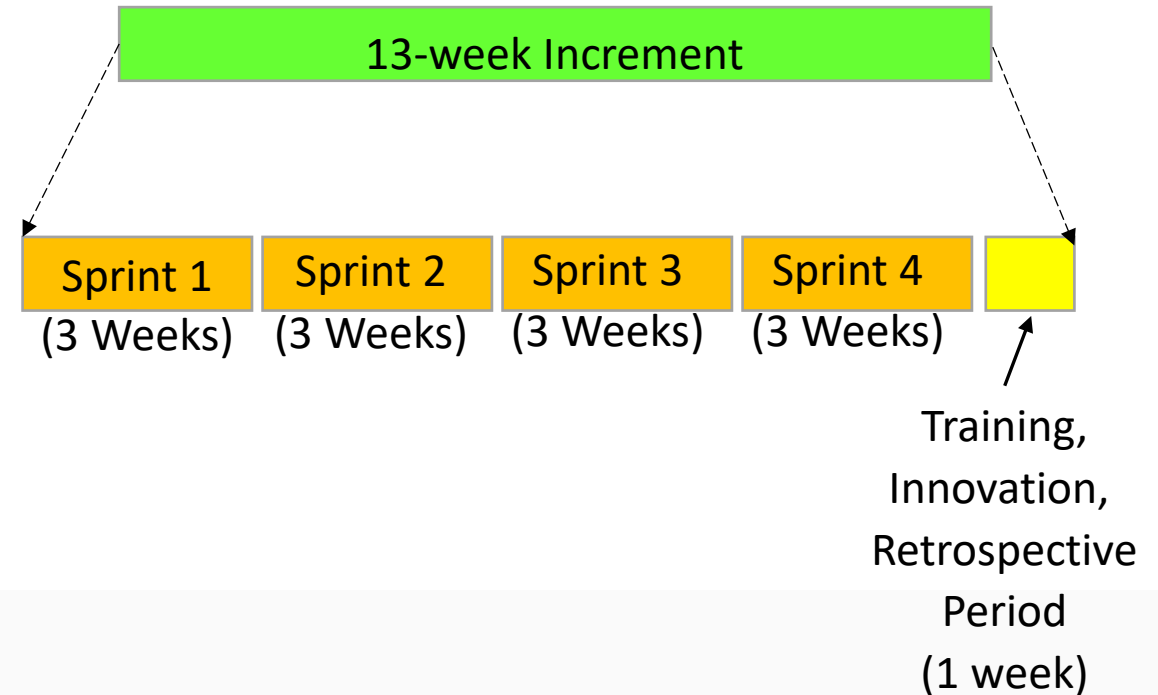
Four DoD Acquisition Projects

Baseline

- **Project A:** Traditional waterfall method used (**completed**)
 - Duration: 39 months (includes schedule extension)
 - Software lines of code (SLOC): 178K
- **Project B:** Hybrid composed of both waterfall and agile/near continuous integration processes (**completed**)
 - Duration: 25 months
 - Software lines of code (SLOC): 113K
- **Project C:** Undertake technical explorations and stand up agile/DevSecOps environment in preparation for Project D (**completed**)
 - Duration: 15 months
 - Software lines of code (SLOC): None
- **Project D:** Agile/DevSecOps (**In Progress for 40 months**)
 - Duration: Approximately 52 months (but potential for extended duration)
 - Software lines of code (SLOC): ~100K

Project D - Details

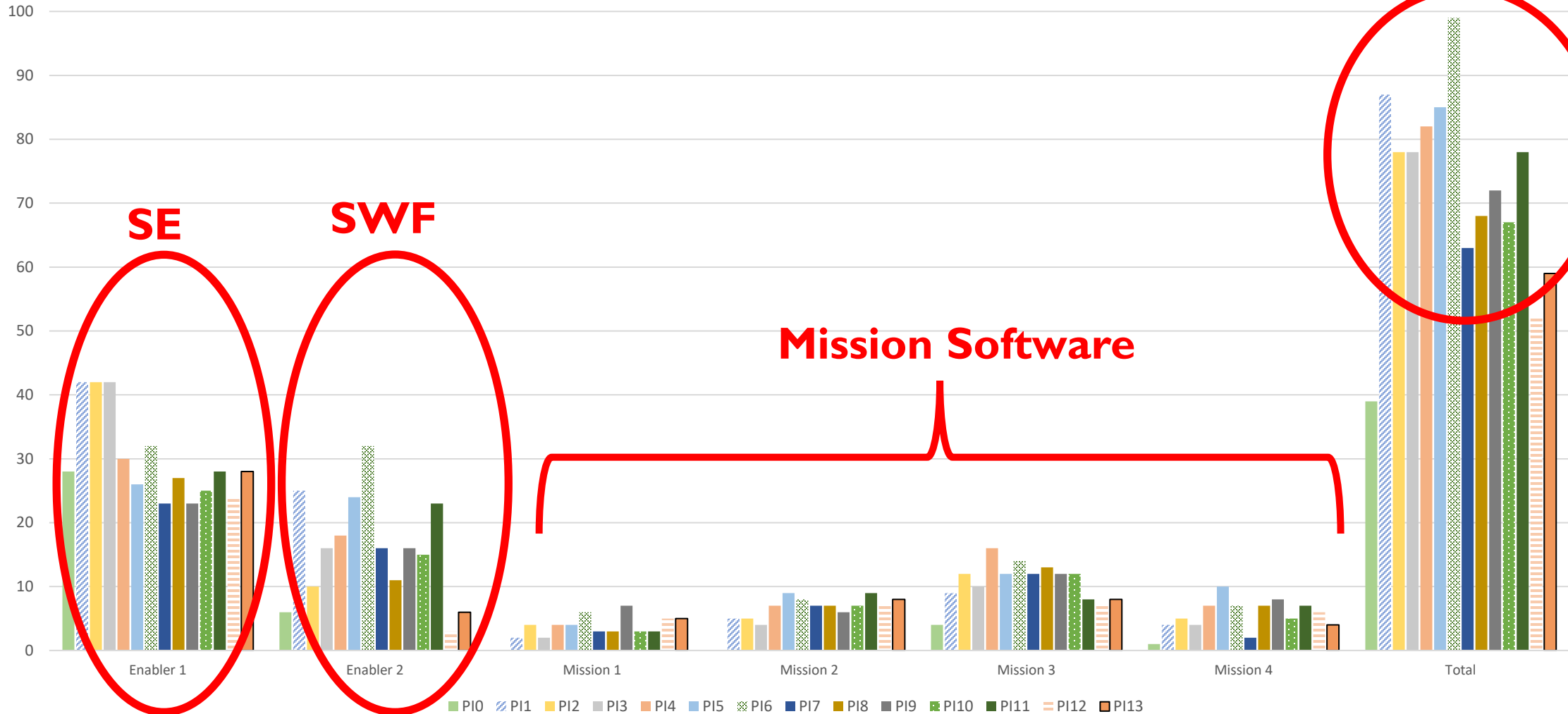
- Hybrid project
 - Roughly 70% agile / 30% waterfall (mainly in the programmatic area)
 - Duration: 52 months (currently in month 40)
 - Software lines of code (SLOC): Not yet known
- Agile implementation
 - Method: Modified SAFe® implementation
 - Program Increment (PI): 13 weeks in duration with four 3-week sprints
 - Last week of PI reserved for demonstrations, training, innovation and if necessary, “catching –up”
 - Six scrum/sprint teams (4 are mission-focused teams, 2 are enabler teams)



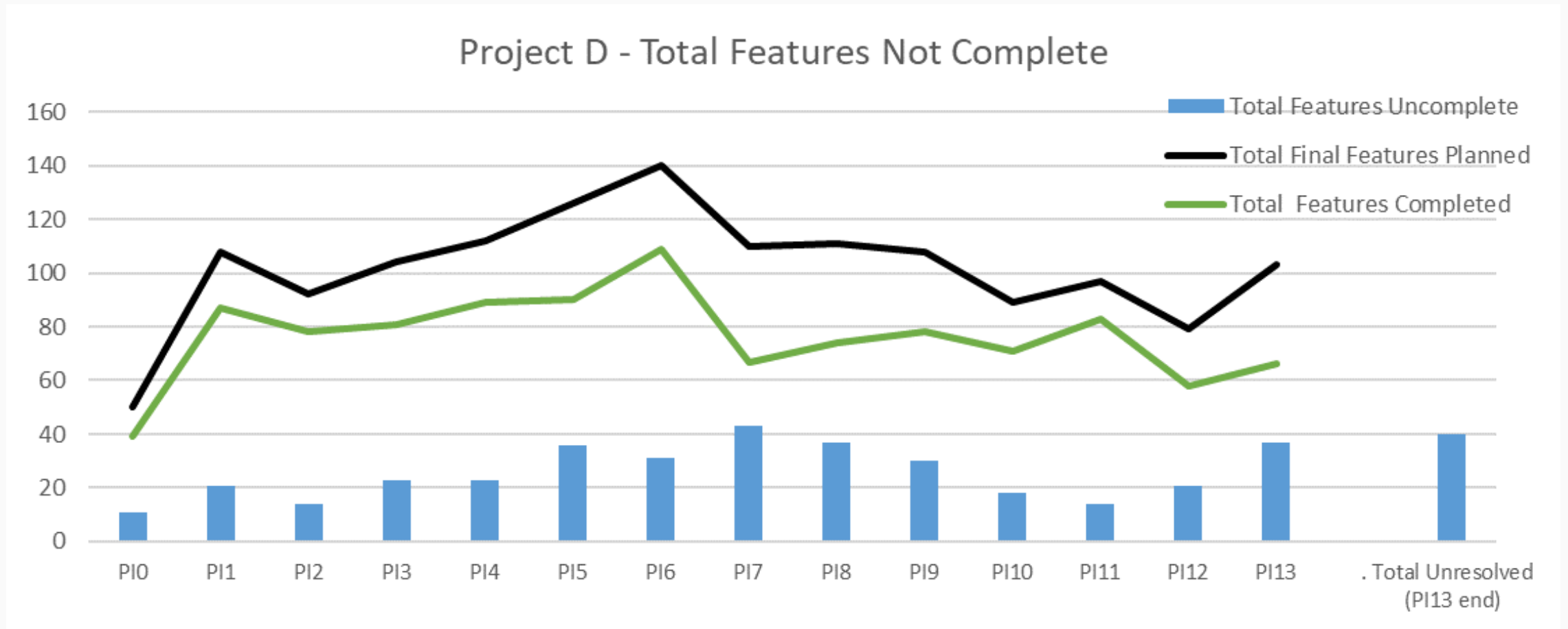
From PI-5 – PI-11, one of the enabler (SWF) teams was split into three teams – producing 8 total teams.

Feature Velocities

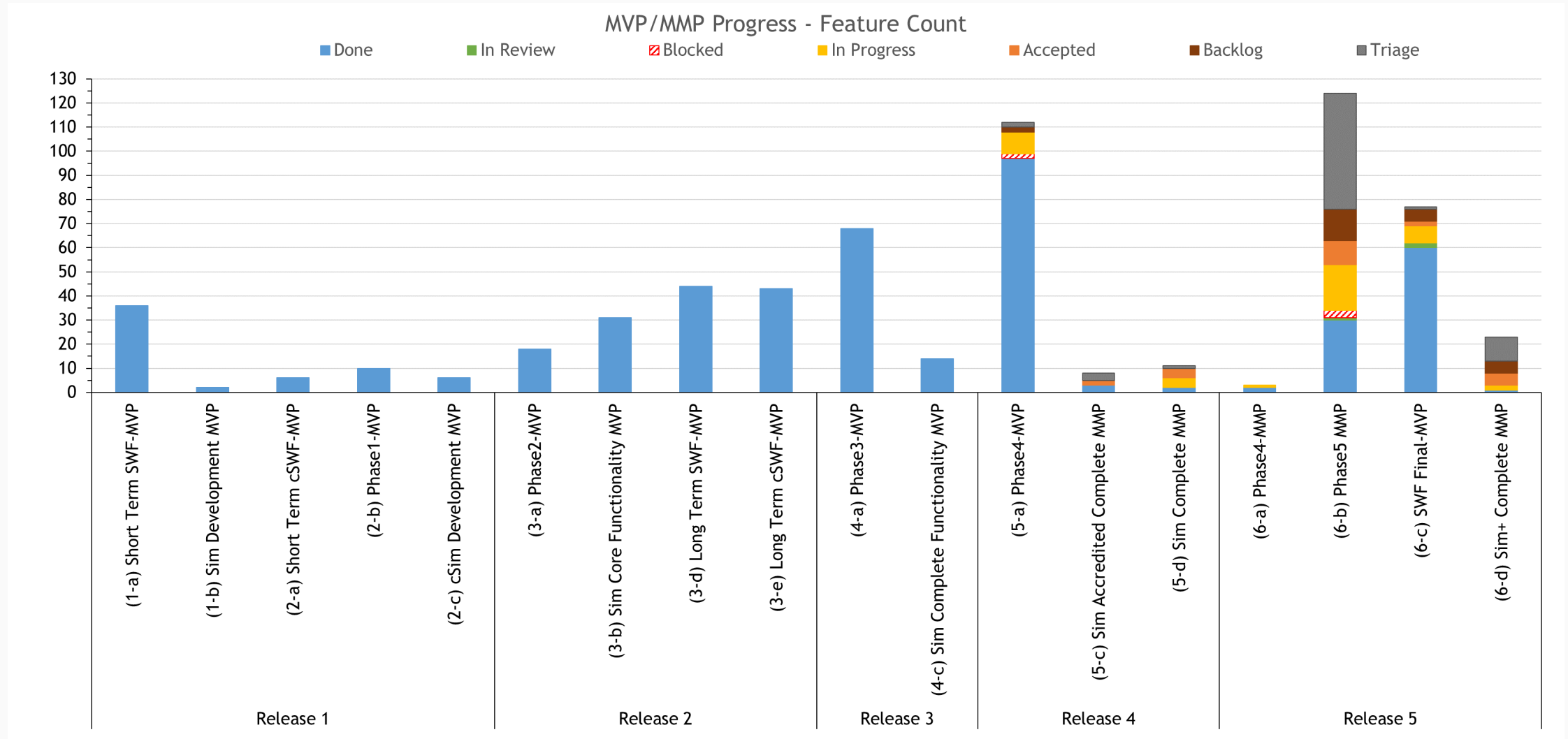
Project D - Feature Velocities per Team



Features Slipping from PI to PI



Are We Delivering Value Despite Features Slipping to Future PIs?



What about EVM – Does EVM Show If Value Is Being Delivered?

- Features that don't complete within a PI (i.e., 13-week increment) are moved into the next or future PI. This will result in a change in the Integrated Master Schedule - IMS (a Baseline Change Request (BCR)). This will impact Earned Value Management (EVM) for the current PI.
- Baseline Execution Index (BEI), Forecasted Execution Index (FEI) and Current Execution Index (CEI)
 - BEI measures how much work was completed against the baseline.
 - FEI measures how much forecasted work was completed. The percentage of forecasted features completing on or earlier than their baseline finish dates / periods.
 - CEI measures the number of tasks completed in a period compared to their previously forecasted completion in that period.

NOTE: EVM is based on monthly performance reporting. Program Increments (PI) work on a 13-week period of performance. EVM data is typically delivered a few weeks after the end of the monthly performance period due to the time required to process it.

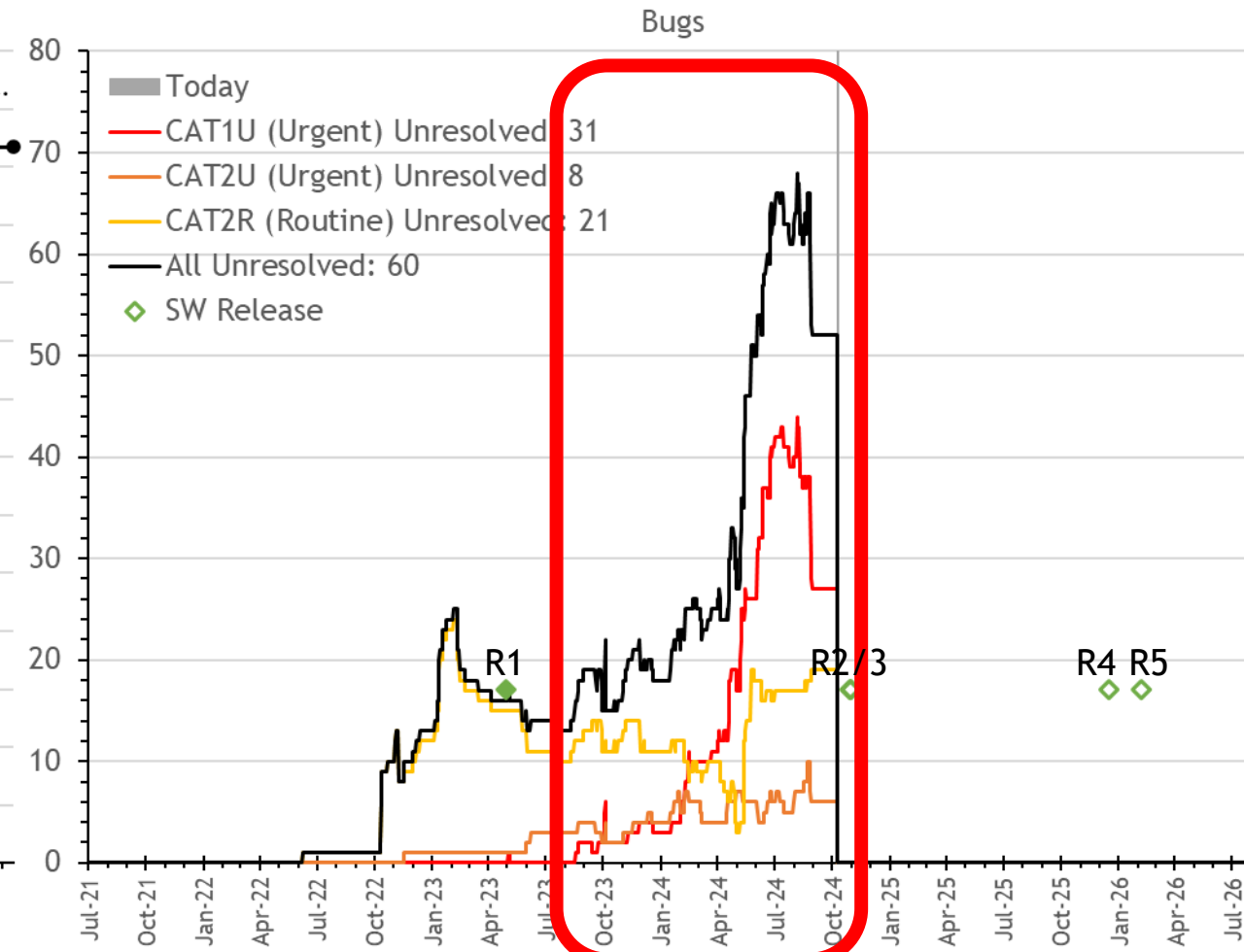
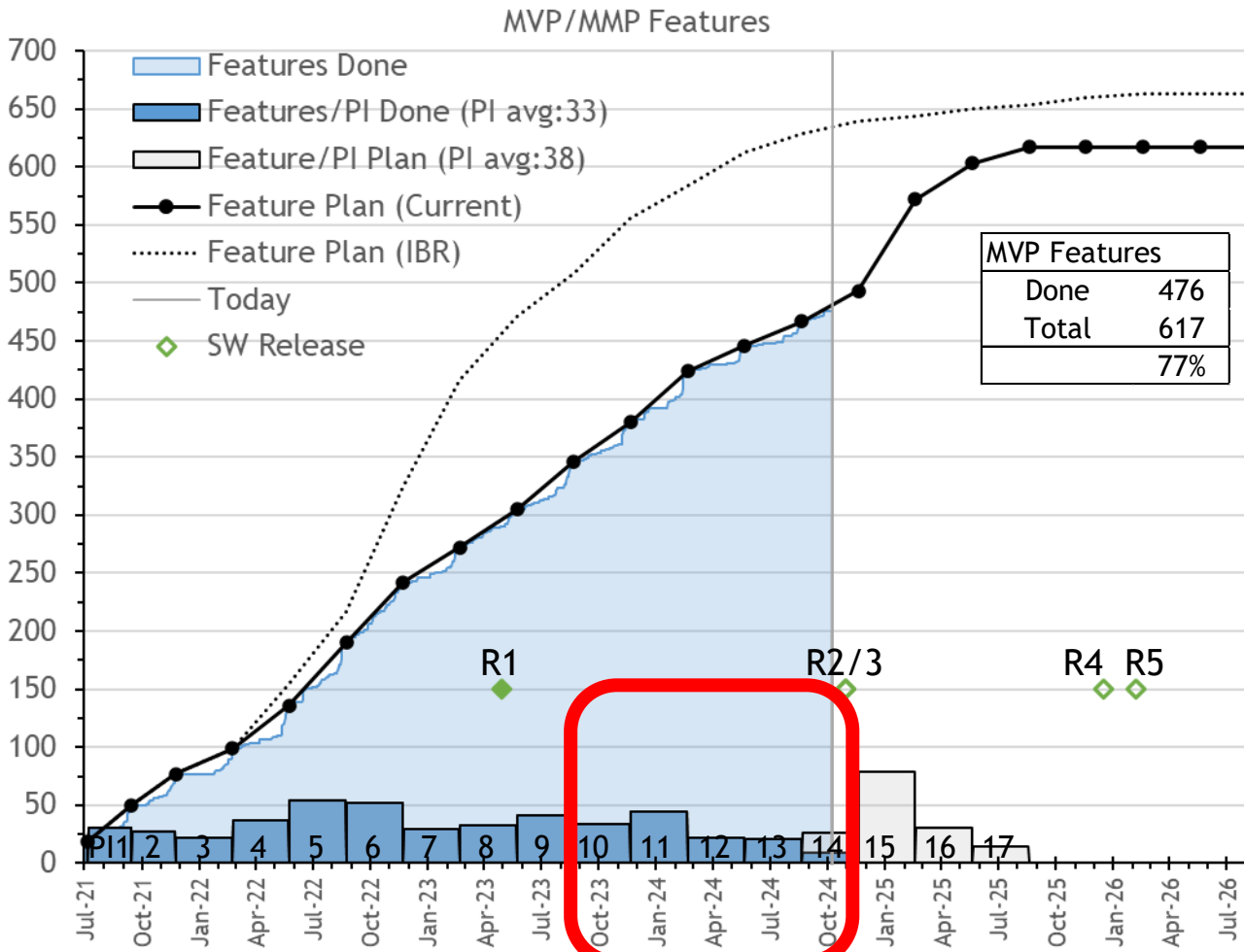
EVM – Can Give an Indication That Work Was Completed...but “Value?”

IPT	Current BEI	Cum BEI	FEI	CEI
IPT #1	0.00	0.90	N/A	N/A
IPT #2	0.33	0.85	N/A	N/A
IPT #3	1.00	0.91	2.00	0.0%
IPT #4	1.00	0.97	1.17	83.3%
IPT #5	N/A	0.40	N/A	N/A
IPT #6	1.00	0.96	N/A	N/A
IPT #7	N/A	0.92	N/A	N/A
IPT #8	1.17	0.99	0.88	75.0%
Overall	0.95	0.95	1.33	73.3%

IPT #3: The team planned to complete two features during the reporting period, and they did complete two features, however, neither of them were planned to be worked in that period (i.e., they were “pull forwards”)

Bottom Line: Tools from previous slides add value to the EVM metrics.

Bow Waves in Agile



A bow wave and we're running out of runway!

"Bugs" discovered during testing

Some Reasons for the Building of a Bow Wave

- Blockages
 - Lack of resources (software license issues, external dependencies, test facilities etc.)
- Competition for workforce
 - In many cases, team members work multiple projects and can be “pulled” depending on project priorities
- Underestimating code complexity
 - Some of this can be attributed to “discovery”
 - Can also be attributed to a lack of understanding of the system requirements
- **Bugs & DRs due to delayed integration and testing of the system**
 - The high priority bugs & DRs consume workforce effort and crowd out mission features which are pushed to the “right.”

Feature Verification

- In many projects, system requirements are decomposed into multiple features/stories
- Often, it is assumed that if the acceptance criteria for each of the features/stories decomposed from a requirement is met, the parent requirement is satisfied (i.e., each feature/story is tested in isolation (i.e., vertical integration & testing).
 - (i.e., the sum of the parts equals the whole part...)



- We need the equivalent of a Final Qualification Test (FQT) in the development pipeline.

Analysis

- **Observation:** Full integration and testing (i.e., end-to-end or horizontal testing) of a system is often delayed – even in an agile environment – until late into the effort, often near the delivery of the MVP
- **Some reasons:**
 - **Lack of upfront systems engineering:** End-to-end use cases and test scripts may be developed late in the development cycle due to the previously noted assumption that simply satisfying the acceptance criteria of a feature/story also satisfies the requirement.
 - **Lack of test resources:** often access to an integrated test environment is delayed due to delayed development of critical (internal/external) systems (simulated or the actual systems) or the test environment is in contention with other programs (i.e., a shared resource)

Number One Recommendation: Do Some Upfront Systems Engineering

- This doesn't mean a detailed design (e.g., waterfall approach).
- We need sufficient upfront engineering to:
 - Populate the project backlog (and understand initial priorities)
 - Determine the workforce needed and when certain skillsets are required
 - Understand external dependencies and the risks of particular elements not being available when they are needed. This is particularly important when supporting continuous integration and testing.
 - **Develop end-to-end tests upfront to facilitate integration and testing (I&T) and understand what resources will be needed for I&T.**
 - Even if (initially) lookup tables are used in lieu of simulators and actual external systems



SYSTEMS
ENGINEERING
RESEARCH CENTER

Thank you!

Stay connected with SERC Online:



Email the presenter: Michael Orosz

 mdorosz@isi.edu

Email the research team:

 duffy@isi.edu
mdorosz@isi.edu