SYSTEMS ENGINEERING RESEARCH CENTER

Space and Missiles Center (SMC) Production Corps - Mission Engineering and Integration of Emerging Technologies

WRT-1041

PI: Michael Orosz

Sponsor: US Space Force Space Systems Command Military Communications and Position, Navigation and Positioning Directorate (SSC/CG)

USC Viterbi
School of Engineering
Information Sciences Institute

ANNUAL RESEARCH REVIEW 2022

Agenda

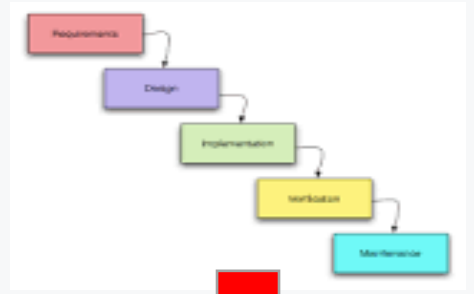
- Project Objectives and Timeline
- Brief Overview of Previous Results
- Current Project Status and Observations
- Next Steps

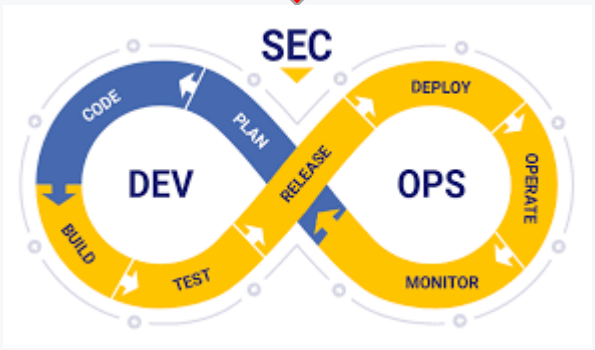
Project Overview

- **Team**:
  - —USC Information Sciences Institute (USC/ISI)
  - —SERC
- **Funding agency**: U.S. Space Force and Space Systems Command Military Command and Positioning, Navigation and Timing Directorate (SSC/CG)
- **Period of Performance**: August 2016 – April 2023 (via multiple SERC projects)

Objectives

- **Improve DoD competitiveness**: Specifically - improve existing DoD space-based *software* system acquisition processes
- **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 acquisition elements from DoD 5000 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

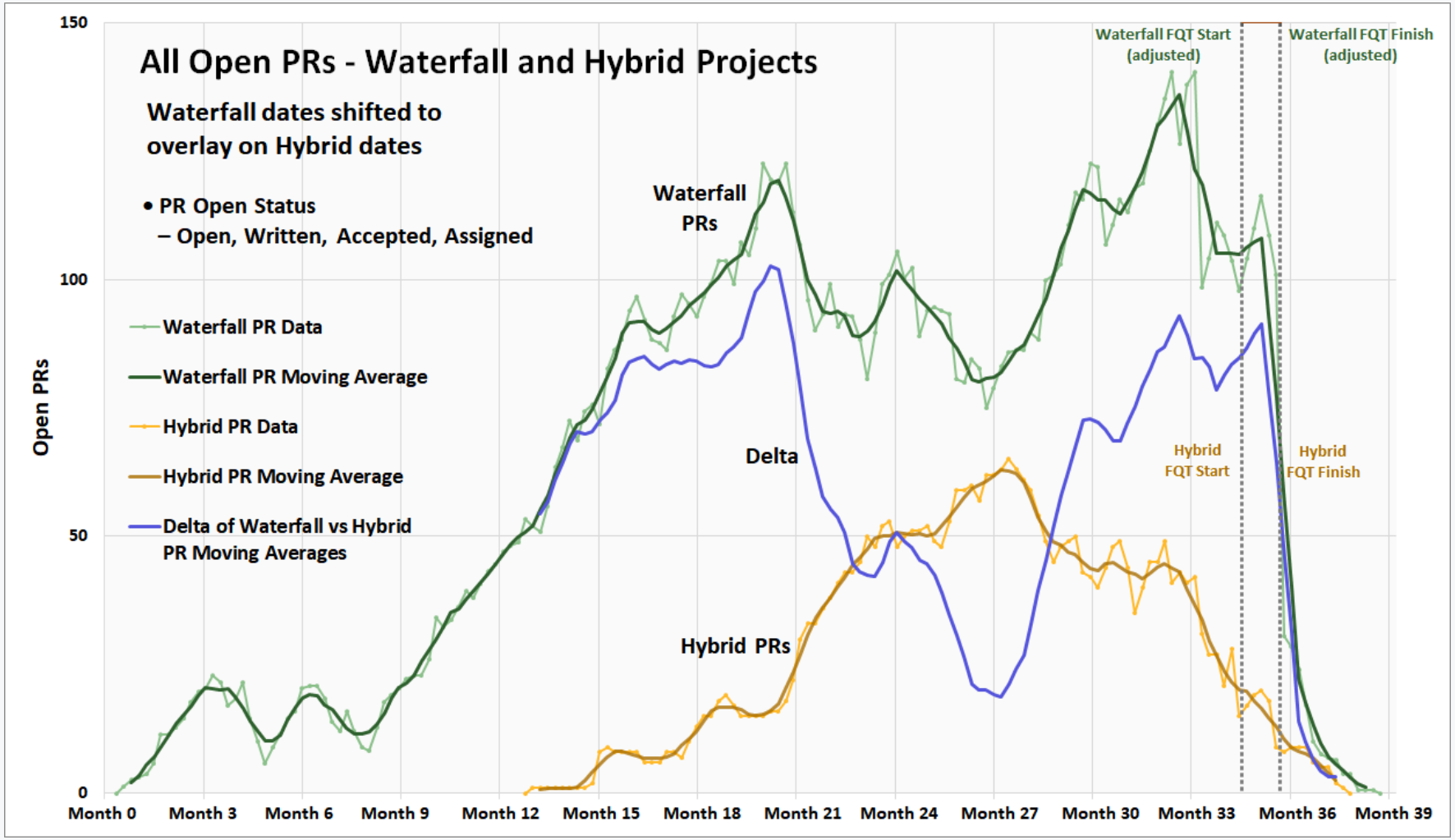
- **Project A:** Traditional waterfall method used (completed)
  - —Duration: 39 months (includes schedule extension)
  - —Software lines of code (SLOC): 178K

Baseline

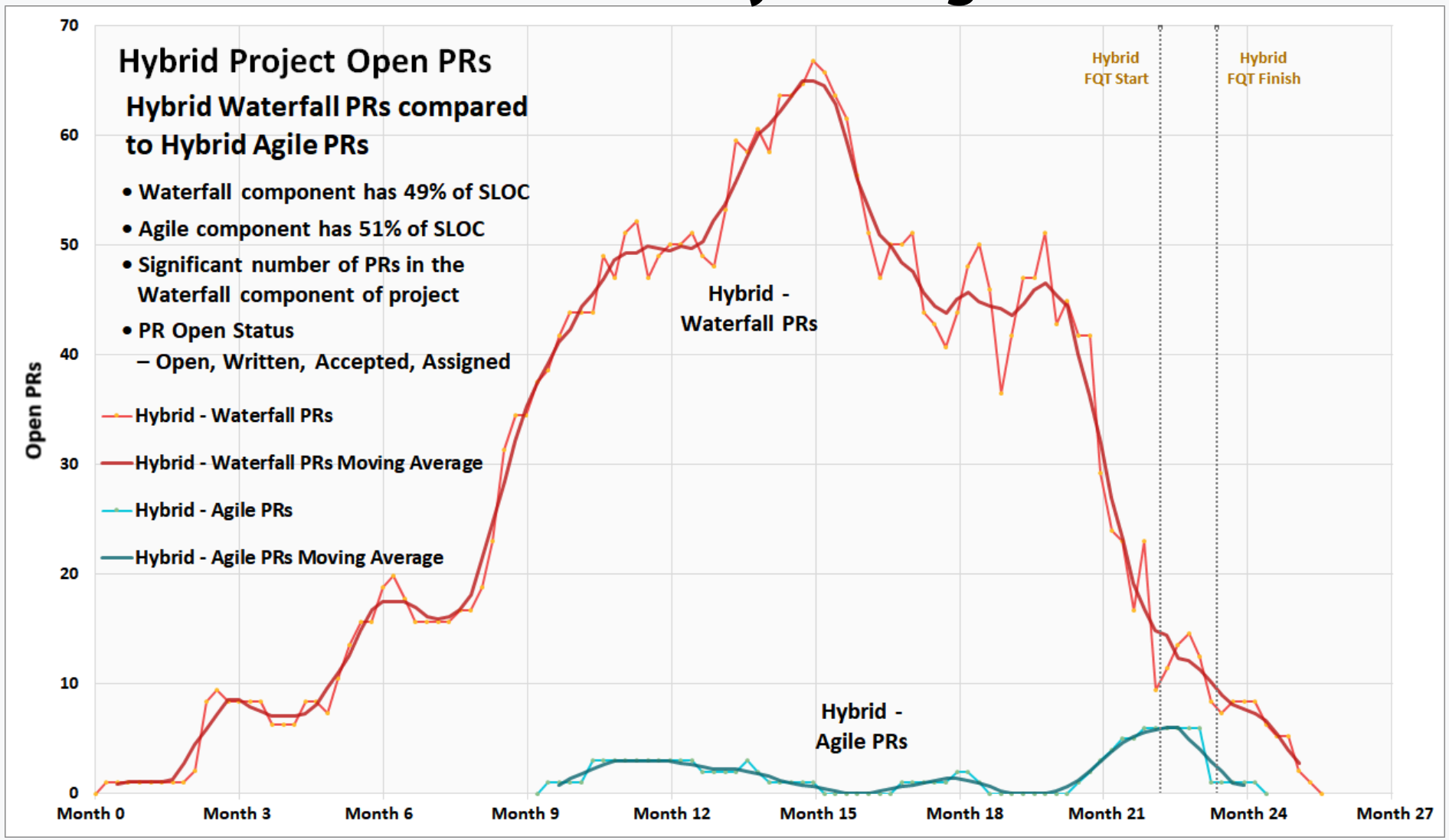
- **Project B**: Hybrid composed of both waterfall and agile/near continuous integration processes (completed)
  - —Duration: 25 months
  - —Software lines of code (SLOC): 113K
- **Project C**: Undertake technical explorations and stand up agile/DevSecOps environment in preparation for Project D (completed)
  - —Duration: 15 months
  - —Software lines of code (SLOC): None
- **Project D**: Agile/DevSecOps (**In Progress for 17 months**)
  - —Duration: Approximately 52 months
  - —Software lines of code (SLOC): TBD

Projects A & B – Example of the Benefits of Agile and Continuous Integration

Problem Report (PR) Comparison of Project A (Waterfall – *The Baseline*) and Project B (Hybrid)



Project B Only: Comparison of PRs between hybrid waterfall vs hybrid agile



Project C – A Study (No Software Development)

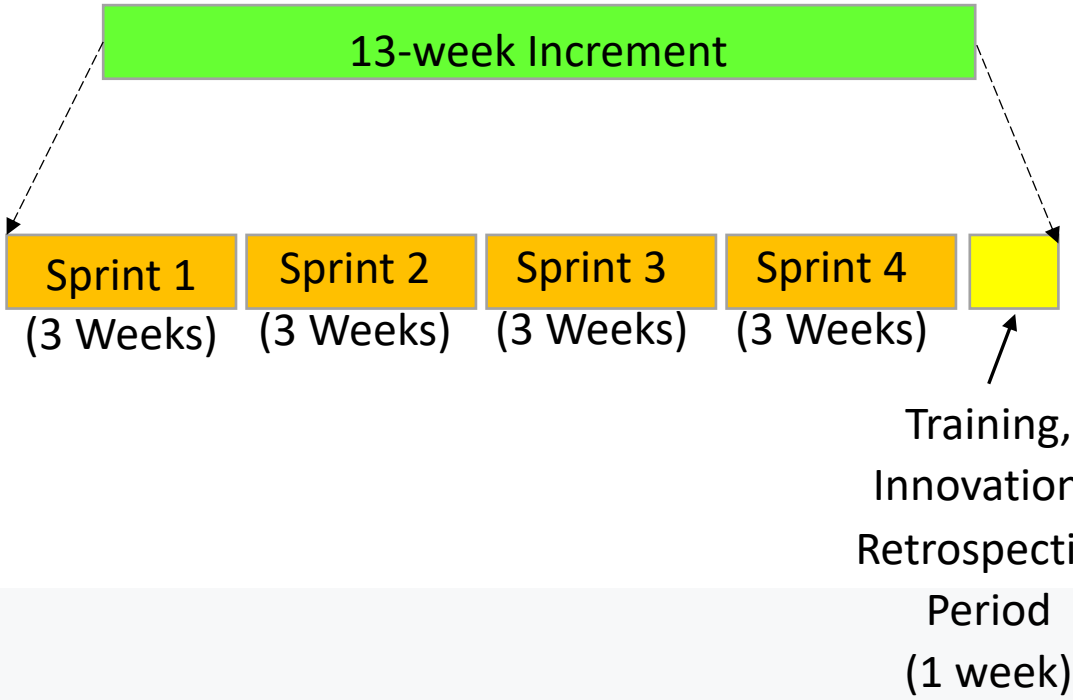
- **Study goal**: Undertake initial research into technical challenges, populate a project backlog and stand up an Agile/DevSecOps software factory (SWF) environment in preparation for **Project D**.

  – **Project D**: A new project to extend an existing waterfall-developed platform. Code complexity is very similar to projects A and B.

- Like Project B (hybrid), Project D exists within an acquisition management system that continues to rely on waterfall metrics (lines of code written/tested, number of PRs reported and worked off, EVM, IMS, etc.).

Project D

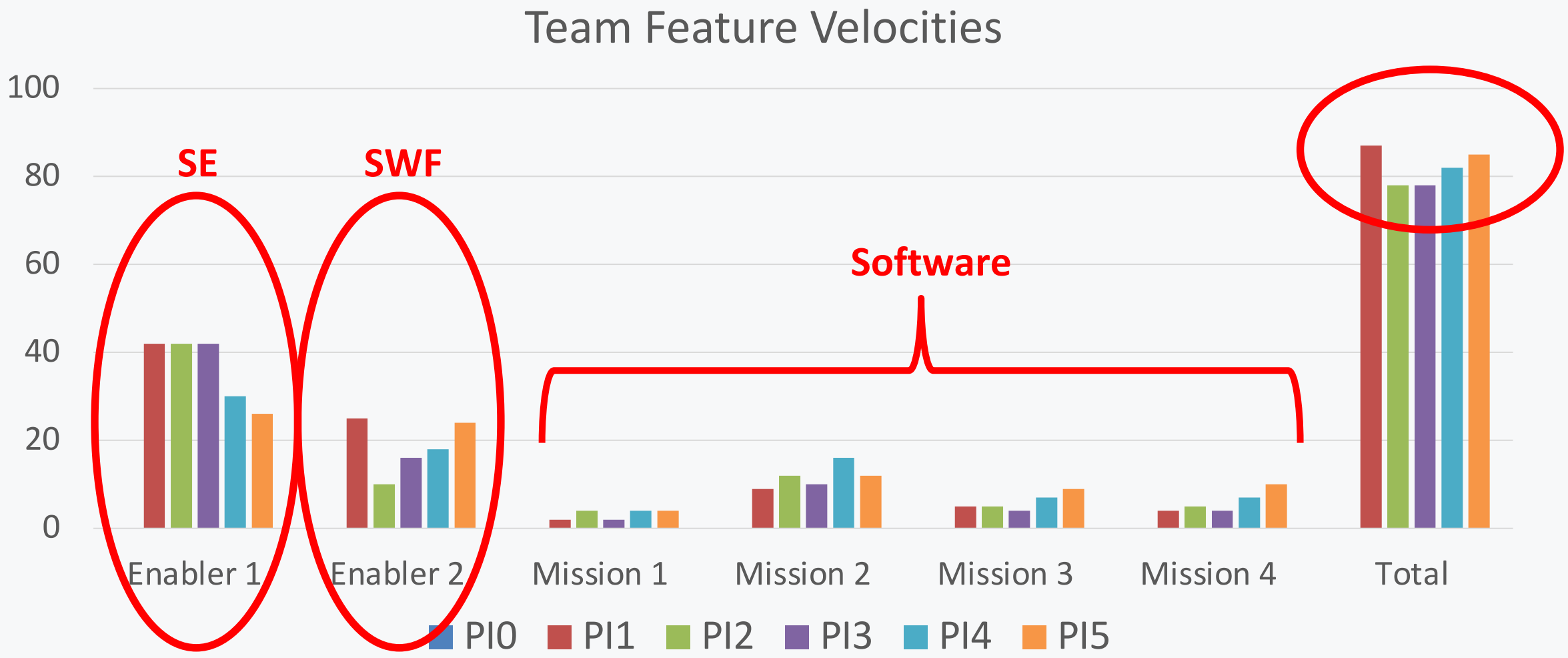
- Hybrid project
  - —Roughly 70% agile / 30% waterfall (mainly in the programmatic area)
  - —Duration: 52 months (currently in month 17)
  - —Software lines of code (SLOC): Not yet known
- Agile implementation
  - —Method: Modified SAFe® implementation
  - —Program Increment (PI): 13 weeks in duration 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)

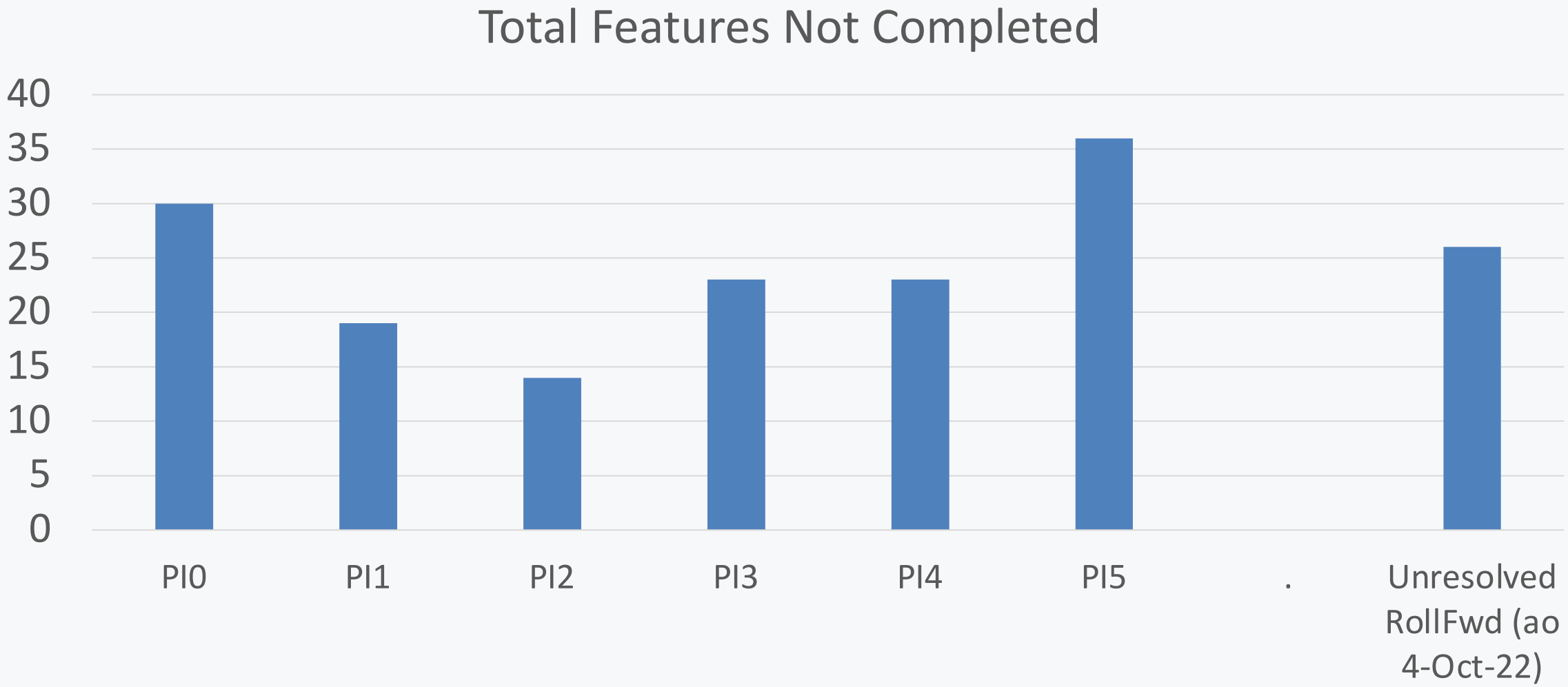


Starting with PI5, one of the enabler teams was split into three teams – producing 8 total teams

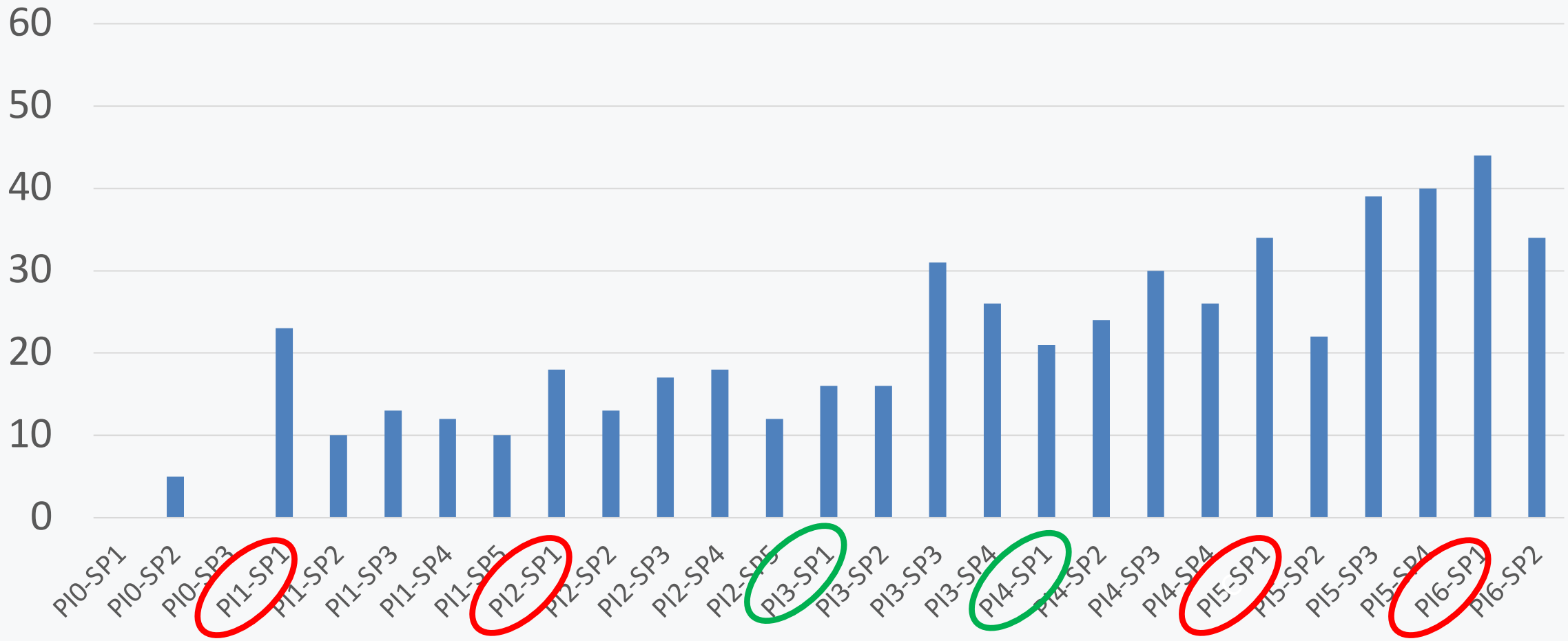
Early Results

- First 17 months focused mainly on systems engineering tasks, build out of DevSecOps pipelines and initial software development.
- Performance metrics being tracked:
  - Requirements → Features → Stories
  - Features completed and not completed within a PI (goal: complete features within a PI)
  - Stories completed and not completed within a sprint (goal: completed stories within a sprint)
- Currently starting to see stories and features spill over boundaries (spillage)
- Velocities have stabilized

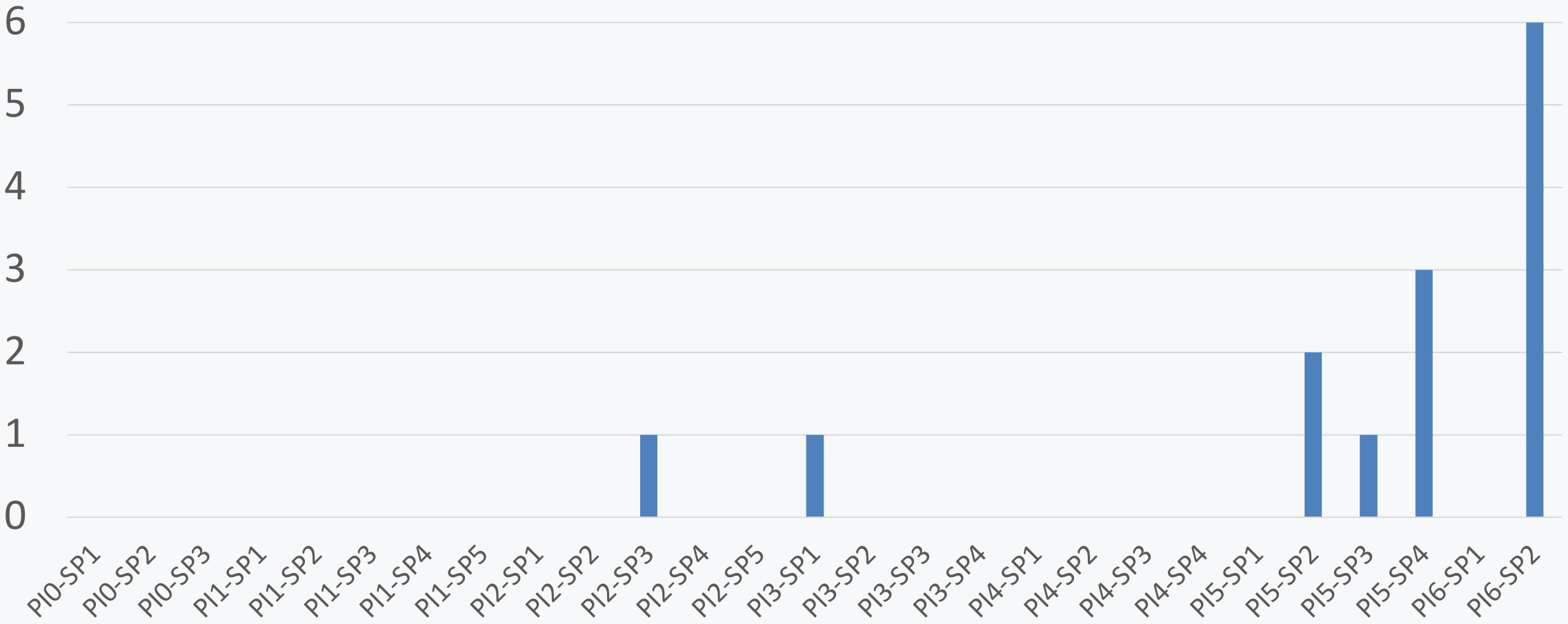

Team Feature Velocities
100
80
60
40
20
0
SE
SWF
Software
Enabler 1
Enabler 2
Mission 1
Mission 2
Mission 3
Mission 4
Total
PI0
PI1
PI2
PI3
PI4
PI5


Total Features Not Completed
40
35
30
25
20
15
10
5
0
PI0
PI1
PI2
PI3
PI4
PI5
.
Unresolved RollFwd (ao 4-Oct-22)

All Teams – Stories Incomplete by PI and Sprint



All Teams – Incomplete Non-Replanned Stories



Some Reasons for the Spillage

- Blockages
  - Lack of resources (software license issues, external dependencies, test facilities etc.)
- Competition for staff
  - 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
- Changing priorities of features and stories due to various issues

Observations and Recommendations

1. Perform upfront engineering to help populate the project backlog and to identify dependencies as early as possible in the program
2. The need to establish (early in the program) a near operational environment (for horizontal I&T)
3. Allocate stories to sprints at the beginning of a program increment
4. Plan margin into the sprints to handle unexpected events such as new technology insertion and/or unexpectedly complex stories
5. Focus on MVP/MMPs during Program Increment (PI) planning
6. Challenges with implementing Agile in a hybrid environment (i.e., IMS, EVM, etc.)
7. Programmatic issues: get licensing, IP, accreditation, certification and other programmatic issues resolved early

Observations and Recommendations

8. Need for on-board and continuous training to ensure team members (both the contractor and acquisition team) are on the same page
9. Be prepared to customize performance tracking tools
   - Applies to all teams…government and development contractor
   - Issues:
     - Software incompatibilities
     - Foreign ownership of tools
     - Access challenges (e.g., VPN, security concerns, etc.)

Next Steps

- Work with government team to continue to address observations and apply lessons learned from the study (Project C) and initial phases of Project D.
- Continue collection of performance metrics with a focus on velocity and related metrics.
- Explore mitigation strategies to the challenges of using EVM, IMS and other cost and schedule performance tracking metrics within an Agile program.
- Continue developing/refining training materials and processes
- Join other projects to collect data and provide SME services

SYSTEMS ENGINEERING RESEARCH CENTER

THANK YOU

Stay connected with us online.

Michael Orosz

mdorosz@isi.edu
michael.orosz.ctr@spaceforce.mil