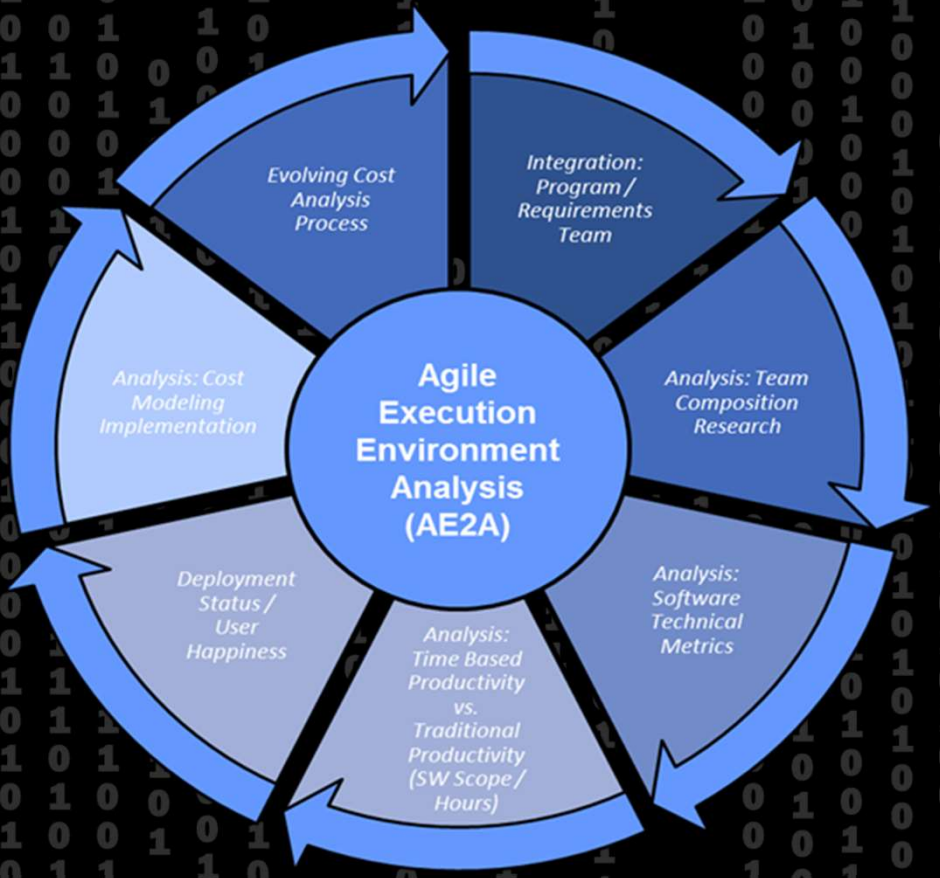
D I G I T A L D I R E C T O R A T E

Case Study:

Agile Execution Environment Analysis (AE2A) in AF DCGS



Matt Hoffman (AFLCMC/HBGF)

Kyle Davis (AFLCMC/HBG, Quantech Services)

Elizabeth Ashwood (AFLCMC/HBG, Quantech Services)

Alex Smith (AFLCMC/HBG, Quantech Services)

William Lane (AFLCMC/HBF, Quantech Services)

Presented at the 2019 ICEAA Professional Development & Training Workshop - www.iceaaonline.com

Case Study: AE2A in AF DCGS

Problem Statement

- Agile Software Development
- Highly integrated software, all applications modified simultaneously
- Requirements / Program Structure aligned with Scaled Agile Framework***…Continuous Development***
- Large Complex Portfolio with Numerous Contracts / Vendors and Government Organizations
- Shared “Acquisition Support” functions
- Inability to rely on traditional cost reporting / CSDR
- Need to provide meaningful real-time decision analysis support

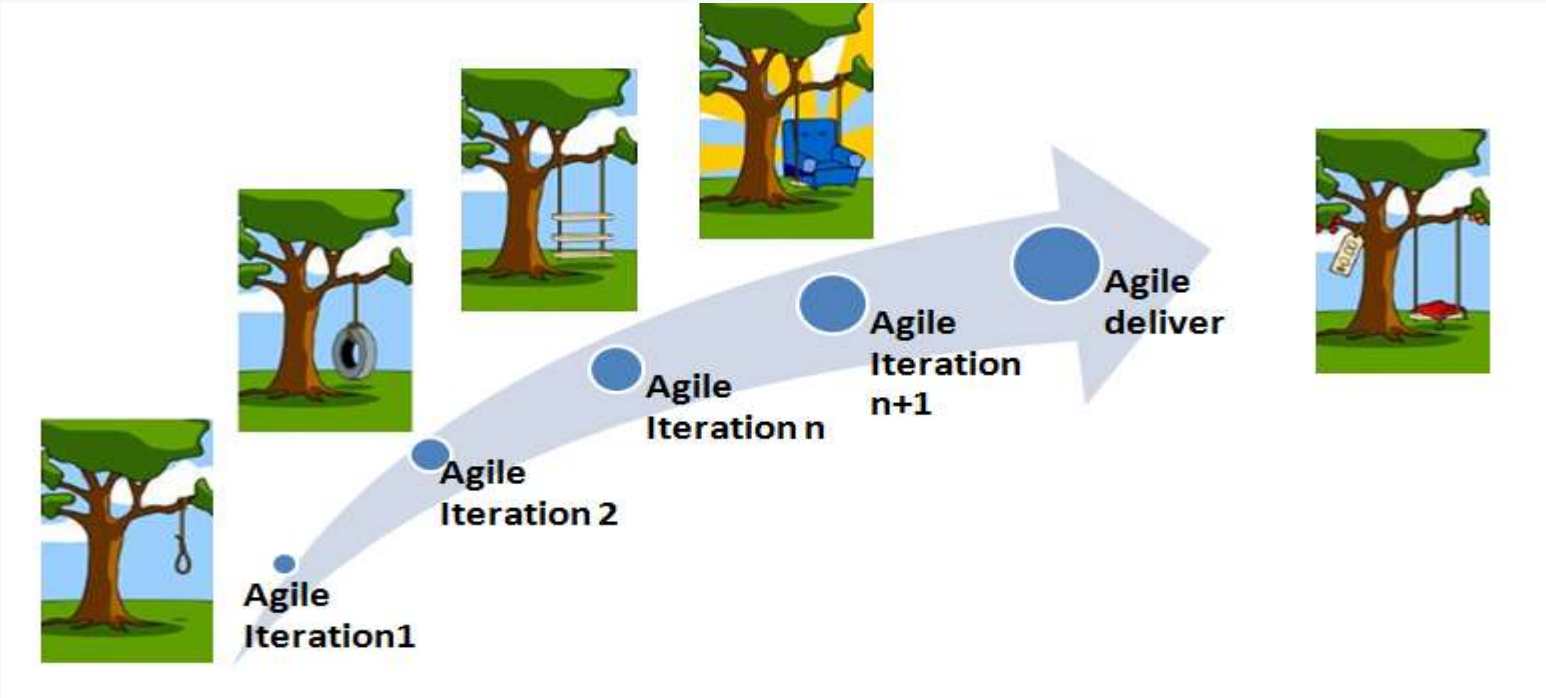
Case Study: AE2A in AF DCGS
Outline

- What is Agile Software Development?
- Understanding Requirements Process & Programmatic Structure
- Program Structure & Progress Tracking Tools (DI2E)
- Discrete vs. Continuous?
- Understanding Program Structure, Size & Complexity
- AE2A: Re-shaping the Cost Analysis Process
- Software Development Teams: Specific Tailored Usage of Agile Terms & SW Tracker Tool Configuration
- Software Metric Analysis: Initial Findings, Lessons Learned and Emerging Techniques
- Way Ahead: Process Integration

Case Study: AE2A in AF DCGS

What is Agile Software Development?

- Agile is the incremental delivery and development of software products with an emphasis on requirement evolution and user feedback
- Software is fielded in increments as new capability is developed, and each limited fielding provides a user with mature sub-elements of the overall capability



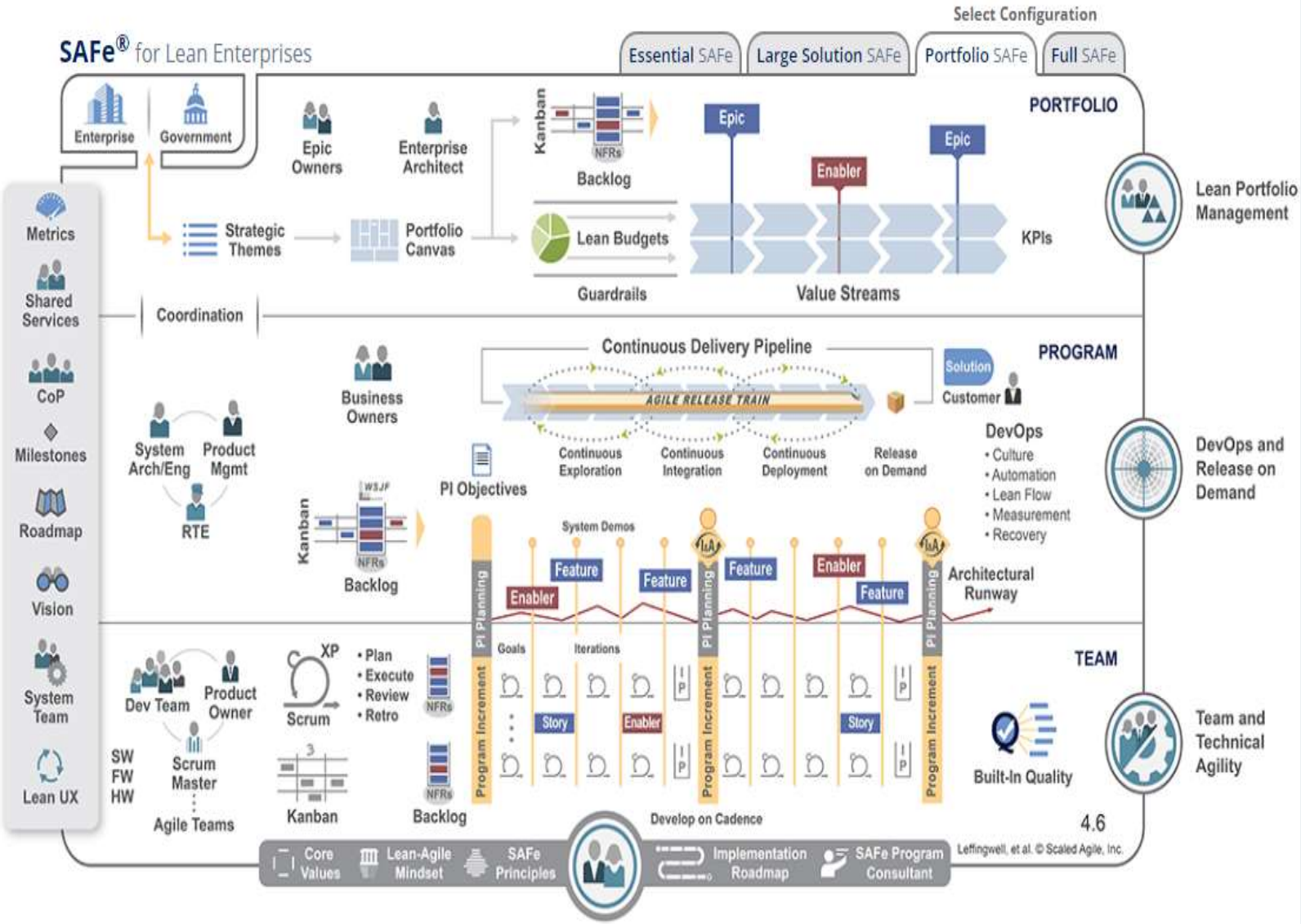
Case Study: AE2A in AF DCGS

What is Agile Software Development?

- The developmental process evolves solutions and products through the collaboration of cross-functional teams utilizing method concepts from Scaled Agile Framework (SAFe)
- The processes allow for flexible development and continuous integration of frequently tested software in order to deliver quality capabilities quickly
- Prioritizes satisfaction of the customer and the continuously changing and evolving requirements with fully involved users throughout the entire process

Case Study: AE2A in AF DCGS

Scaled Agile Framework (SAFe)



Case Study: AE2A in AF DCGS

New Environment Driving Process Change

Traditional Process

Scope / Discrete Requirements → Technical Baseline / Productivity Analysis → Estimate Range of Possible Cost and Schedule Duration

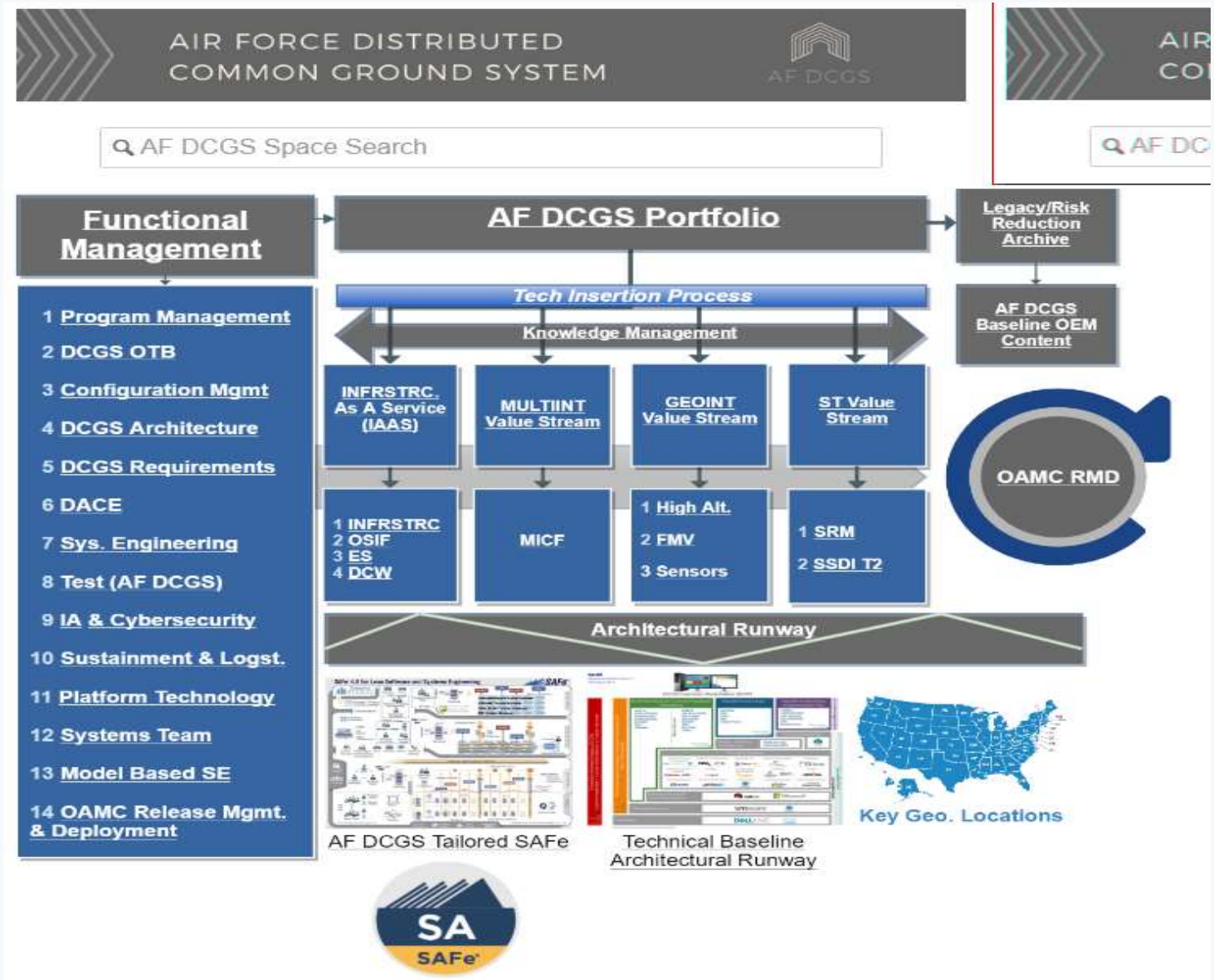
Evolving C2ISR Division Process

Known Team Size / Burn Rate → Productivity & Feature Size Analysis → Estimate Range of Possible Scope

- **Current C2ISR Division requirement's process & programs are structured in alignment with SAFe… continuous / no discrete endpoint**

Case Study: AE2A in AF DCGS

DCGS Transformation Structure



Case Study: AE2A in AF DCGS

Program Status Tracking Tools

Jira - Issue Tracking, Agile Planning

Issue and project tracking for traditional and agile projects.

Confluence - Wiki, Documentation

Per-project, collaborative document editing.

Jenkins - Continuous Integration, Automated Builds

Automated software build, test, integration, and performance profiling.

Nexus - Code Artifacts, Maven Repo Mirror

Stores software artifacts and resolves dependencies for maven builds.

Password Management

Reset and unlock your own DI2E password.

XMPP Chat server

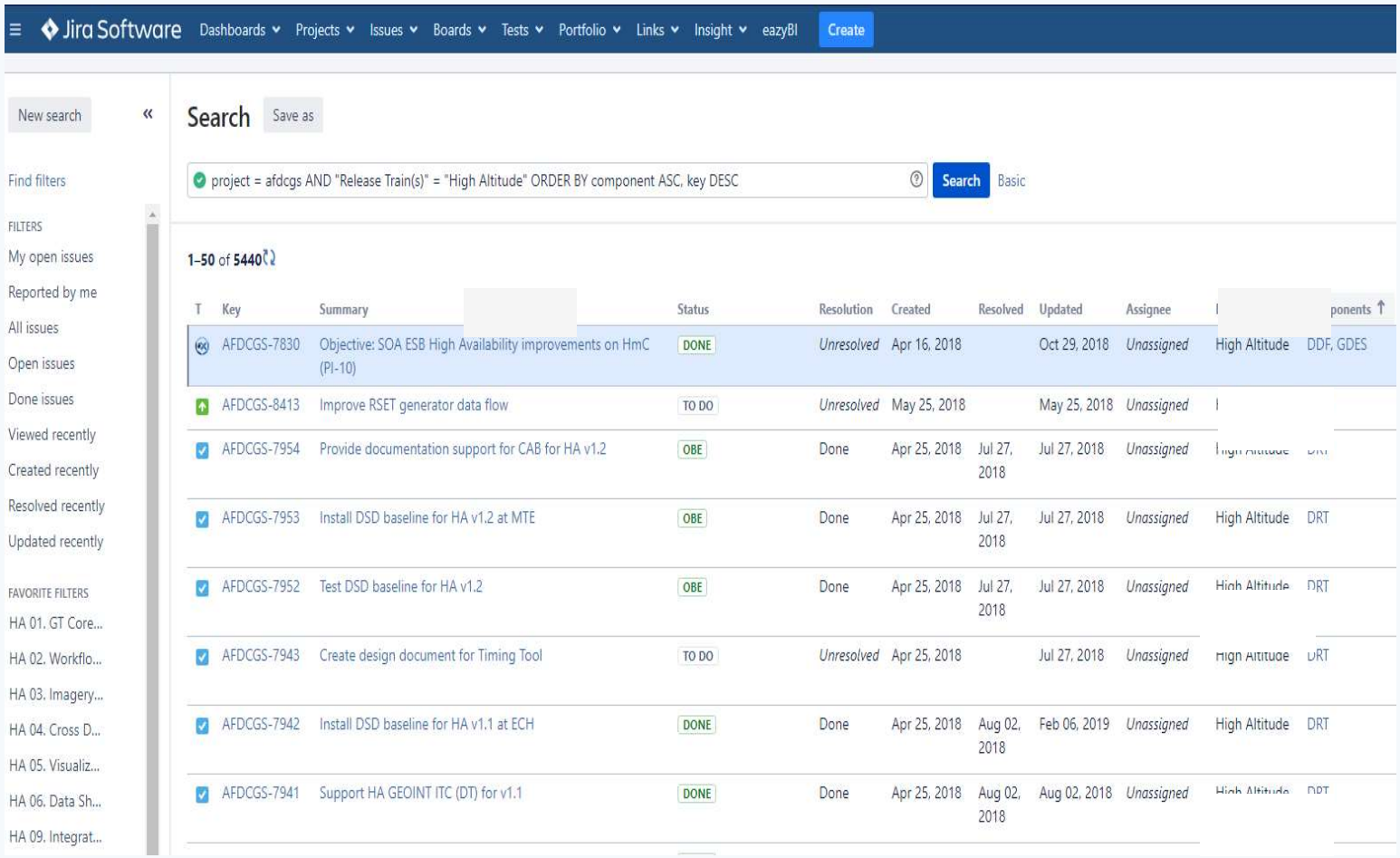
Real time chat to be used with any XMPP client (e.g. Psi, Pidgin, Trillian.) Can be configured with permanent rooms by request.

FileSender

Makes it easy to securely transfer large files using only a web browser. Files can be transferred to one or several recipients.

- **Confluence Used for Program Structure Documentation, JIRA Used for Software Effort Progress Tracking**

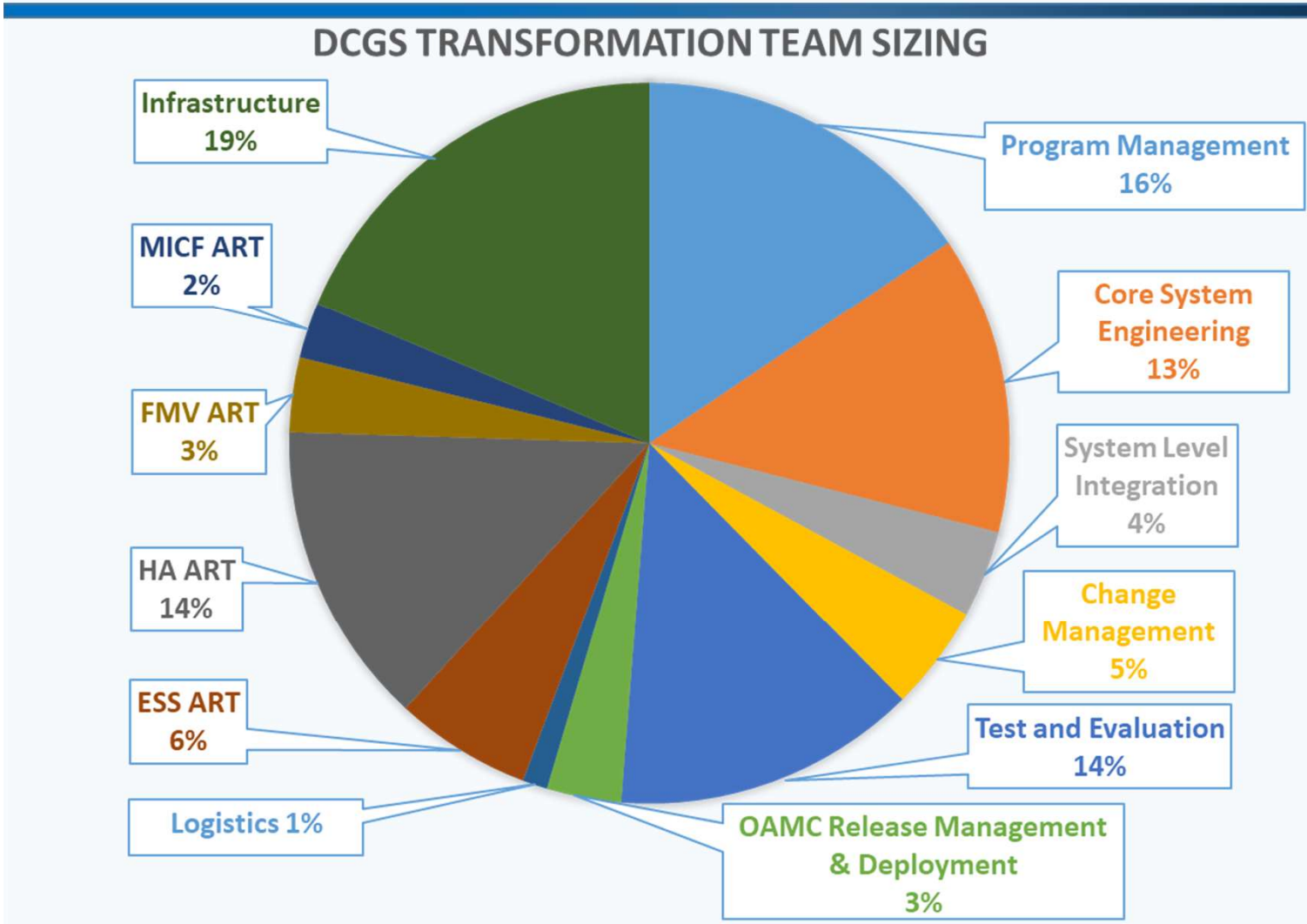
Case Study: AE2A in AF DCGS
Program Status Tracking Tools



- **JIRA Issues Navigator where programmatic tickets for Epics, Features, and Stories are able to be exported**

Case Study: AE2A in AF DCGS

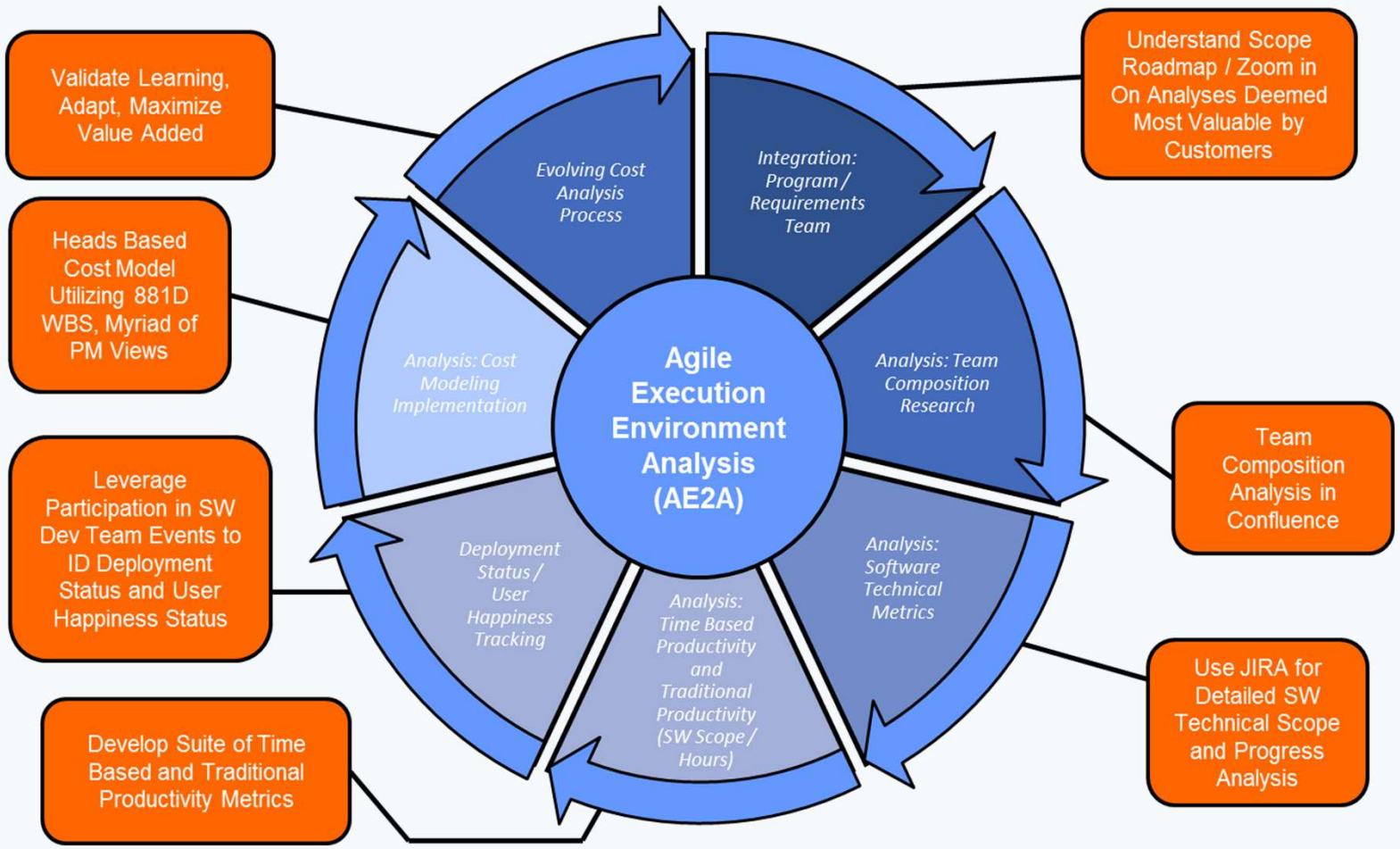
Size and Complexity of the DCGS Transformation Effort



- **Government Led Integration Leveraging 60+ Contracts**
- **~44% of Heads are in the ARTs / Value Stream; ~56% of Heads are within the Functional Management Teams**

Case Study: AE2A in AF DCGS

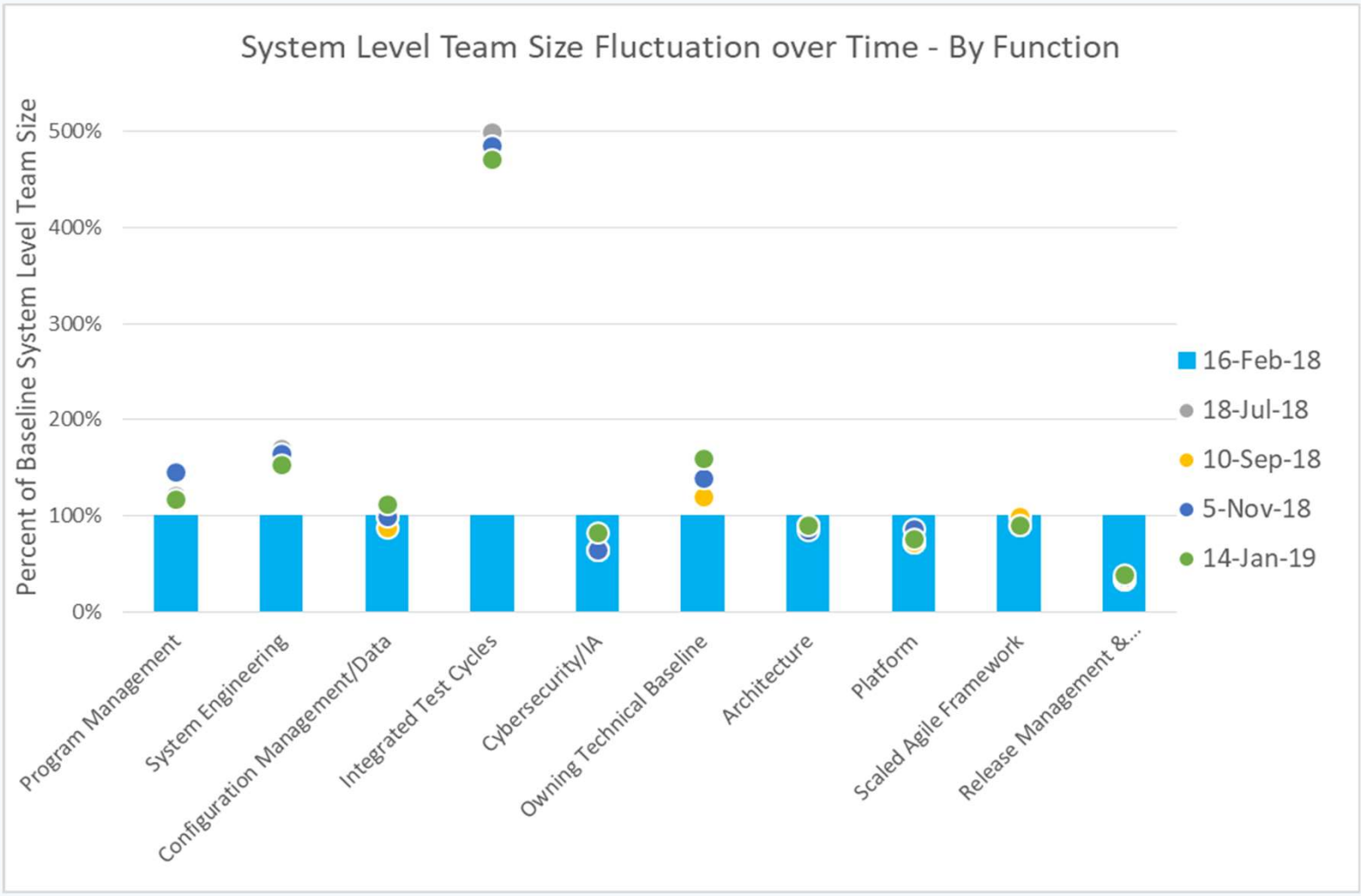
AE2A: Re-shaping the Cost Analysis Process



Our Own Evolving Continuous Process…Working to Integrate it Now

Case Study: AE2A in AF DCGS

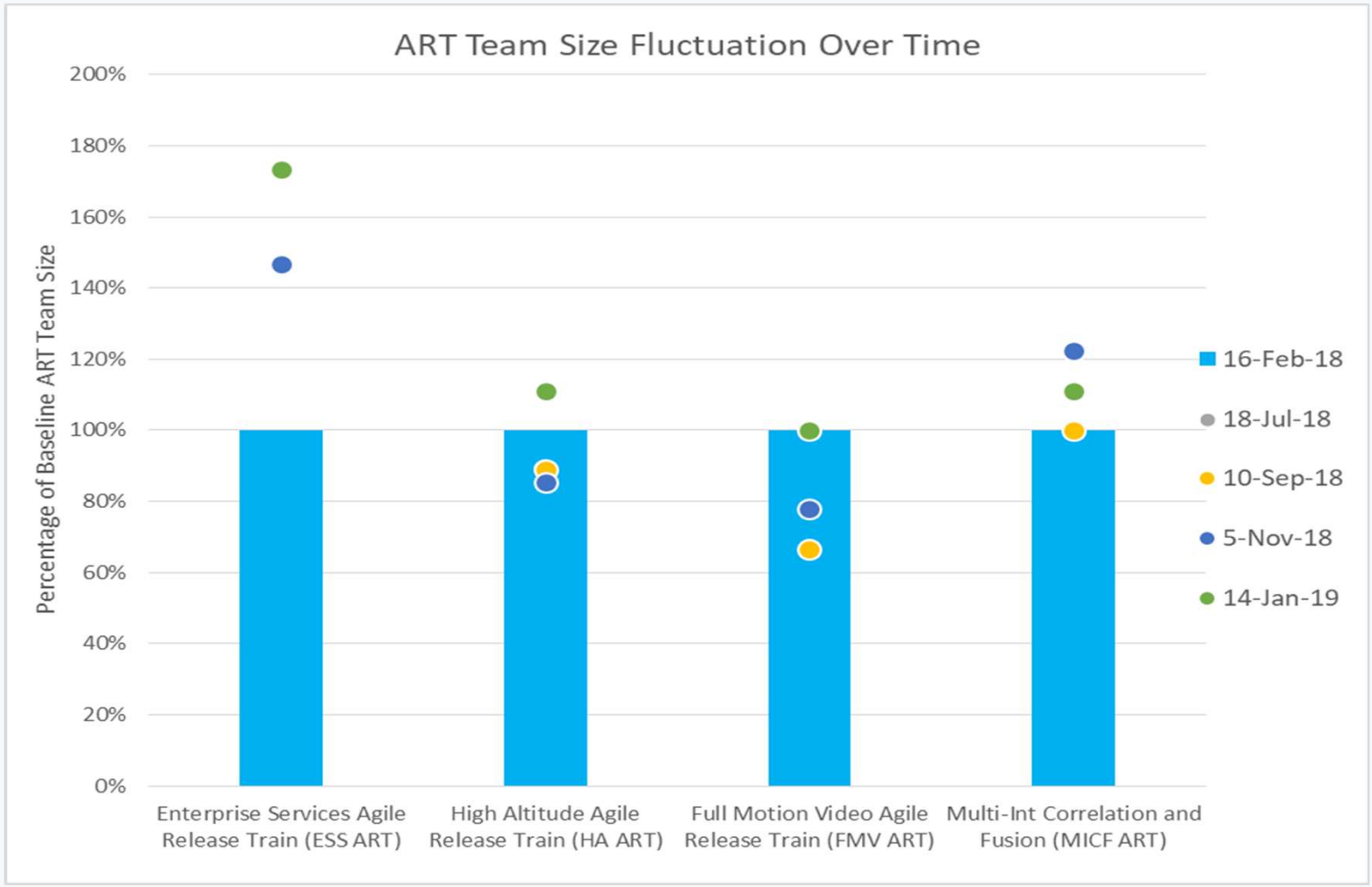
Lessons Learned from Initial Analysis



- **Understand Entire Workforce, Whether PEC Funded or Not**
- **Study Trends over Time; Overarching System Level head counts remain stable with an average delta of 4.4%**

Case Study: AE2A in AF DCGS

Lessons Learned from Initial Analysis

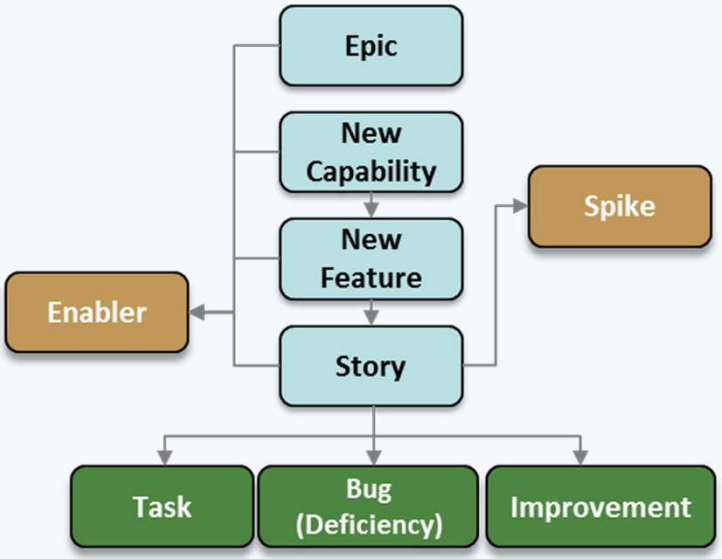


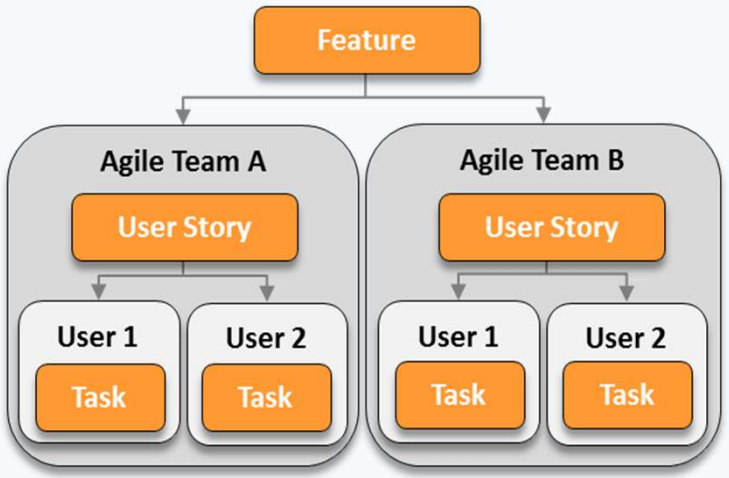
- **Understand Entire Workforce, Whether PEC Funded or Not**
- **Study Trends Over Time**

Case Study: AE2A in AF DCGS

Software Development Teams: Specific Tailored Usage of Agile Terms & Software Progress Tracker Tool Configuration

- **Feature – A service that fulfills a stakeholder need to be delivered by a single ART in a single PI**
- **User Story – Short descriptions of a small piece of desired functionality (written in the user's perspective to convey value) sized so they can be completed in a single iteration**
  - Defines system behavior / functionality
  - Completed in a single iteration
- **Task – An objective that must be achieved**
  - Tasks are smaller work items (can be completed in a day or so) that build a story and by itself, is devoid of business benefit
- **Bug – Problem the impairs product or service functionality**
- **Enabler – Activities needed to extend the Architectural Runway to provide future business functionality. These include exploration, infrastructure, compliance, and architecture development**
- **Improvement – An enhancement to an existing feature**
- **Spikes – A type of exploration Enabler Story that represent activities such as research, design, investigation, exploration, and prototyping**





- **Understand How Each Team / Area Utilizes their SW Progress Tracking SW**

Case Study: AE2A in AF DCGS

Software Metric Analysis: Initial Findings & Lessons Learned

	Epics	Features	Average Issues Per Feature
HA ART	11	216	4.0
FMV ART	9	65	4.2

	Components	Epics	Average Issues Per Epic
ESS ART	16	122	14.1

- **Understand How Each Team / Area Utilizes their SW Progress Tracking SW**
- **Determine how Issues are mapped in each Area and understand the complexity of each ART independent from one another**

Case Study: AE2A in AF DCGS

Software Metric Analysis: Initial Findings & Lessons Learned

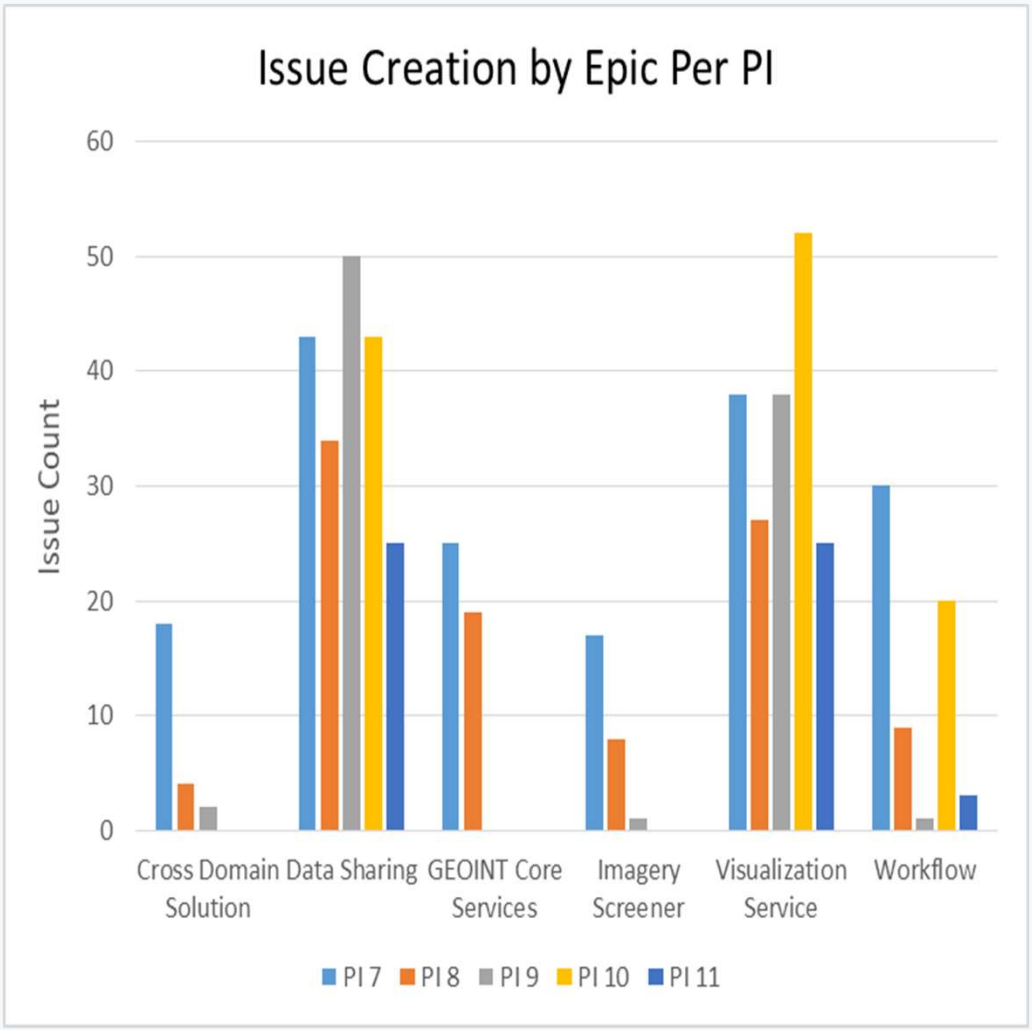
	Average Issues Per Feature	Average Issues Per Small Feature	Average Issues Per Medium Feature	Average Issues Per Large Feature
HA ART	4.0	2.0	6.1	14.5
FMV ART	4.2	1.5	6.7	15.8

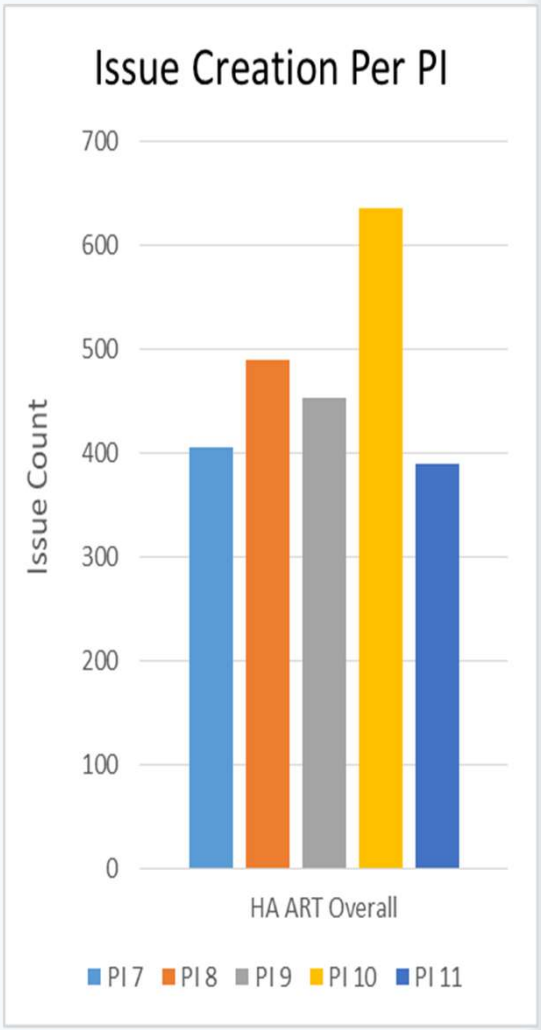
	Average Issues Per Epic	Average Issues Per Small Epic	Average Issues Per Medium Epic	Average Issues Per Large Epic
ESS ART	14.1	3.1	26.9	169.8

- **Understand detail in the ART components in order to map future requirements to historic actuals for each team**
- **Integrating with SPO Teams / Processes Early**
- **Socializing our new analysis abilities and establishing integrated process between user, XR, cost and PM/EN chains to utilize this data for POM / forecasting**

Case Study: AE2A in AF DCGS

Software Metric Analysis: Initial Findings & Lessons Learned





- **Monitor and Understand New Issue Creation. What Drives It?**

Case Study: AE2A in AF DCGS

Software Metric Analysis: Emerging Techniques

Agile Execution Environment Analysis (AE2A) Productivity Explorations
Time Based Productivity (Not Normalized for Team Size)
Equivalent Epics Completed During Each PI (Serial)
Cumulative Equivalent Epics Completed Per PI
Equivalent Features Completed During Each PI (Serial)
Cumulative Equivalent Features Completed Per PI
Issues Completed During Each PI (Serial)
Cumulative Issues Completed Per PI
Traditional Productivity (Normalized for Team Size)
Equivalent Epics Completed During Each PI (Serial)
Cumulative Equivalent Epics Completed Per PI
Equivalent Features Completed During Each PI (Serial)
Cumulative Equivalent Features Completed Per PI
Issues Completed During Each PI (Serial)
Cumulative Issues Completed Per PI

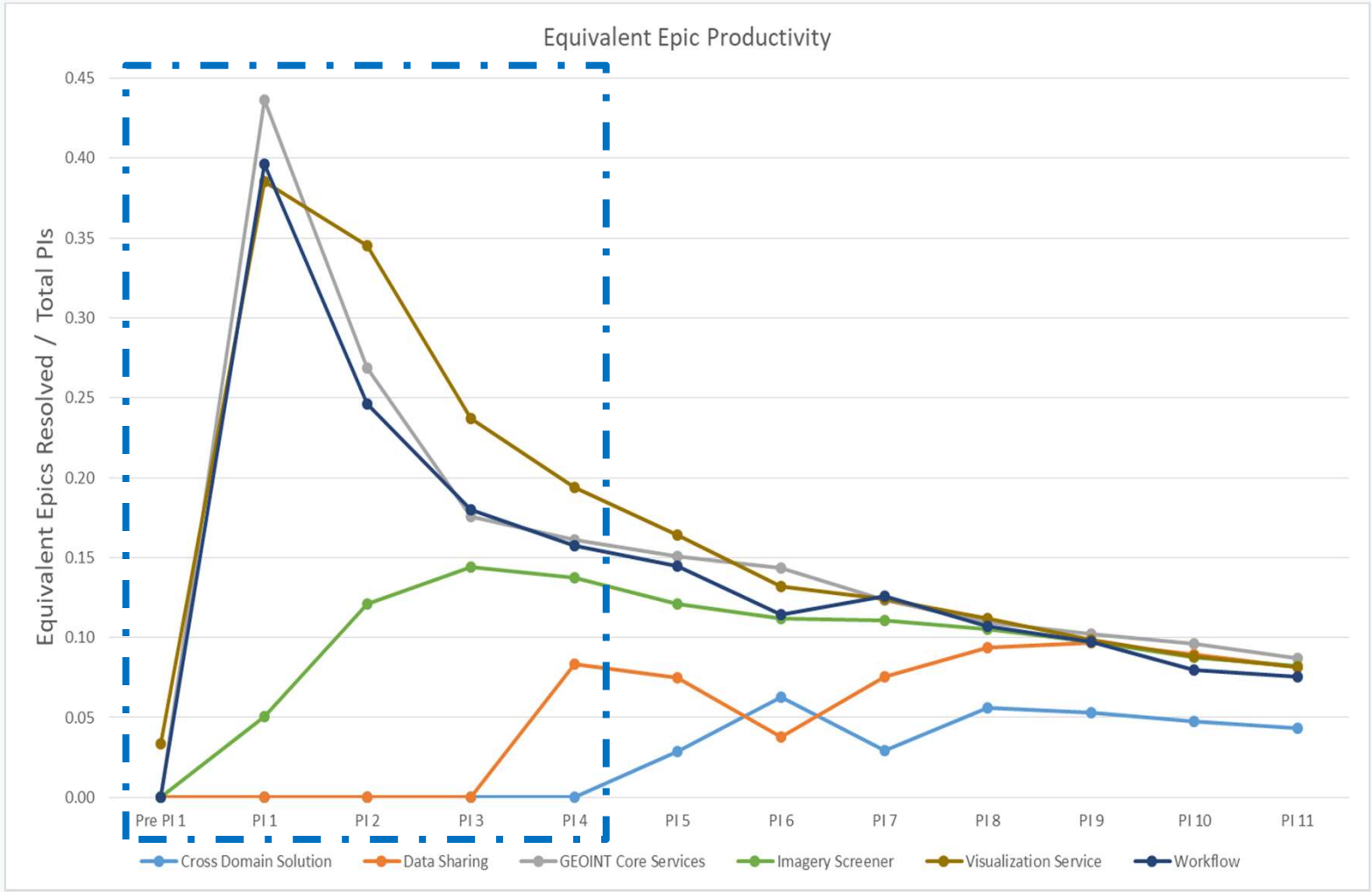
Assess Requirements & Conduct Out Year Planning @ Epic or Feature Level

Smart Usage of Issue Level Productivity Used for Behind the Scenes Conversions

- **Analyze Productivity From a Number of Angles**

Case Study: AE2A in AF DCGS

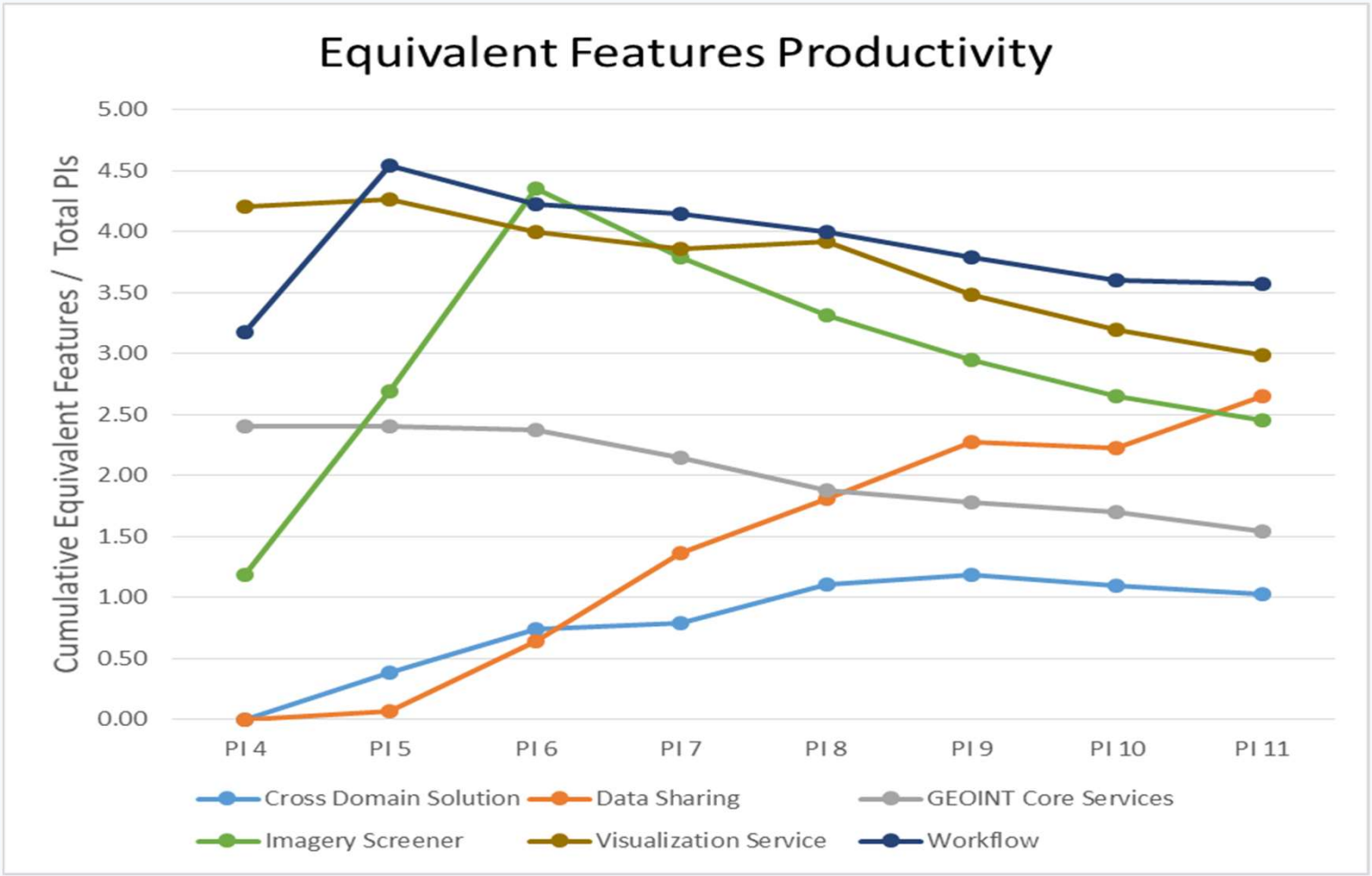
Software Metric Analysis: Emerging Techniques, Time Based Productivity



- **Be Careful When Utilizing Productivity Data "Too Early"**
- **Consider Time Based Productivity "Decline" Over Time as System Size and Complexity Increases**

Case Study: AE2A in AF DCGS

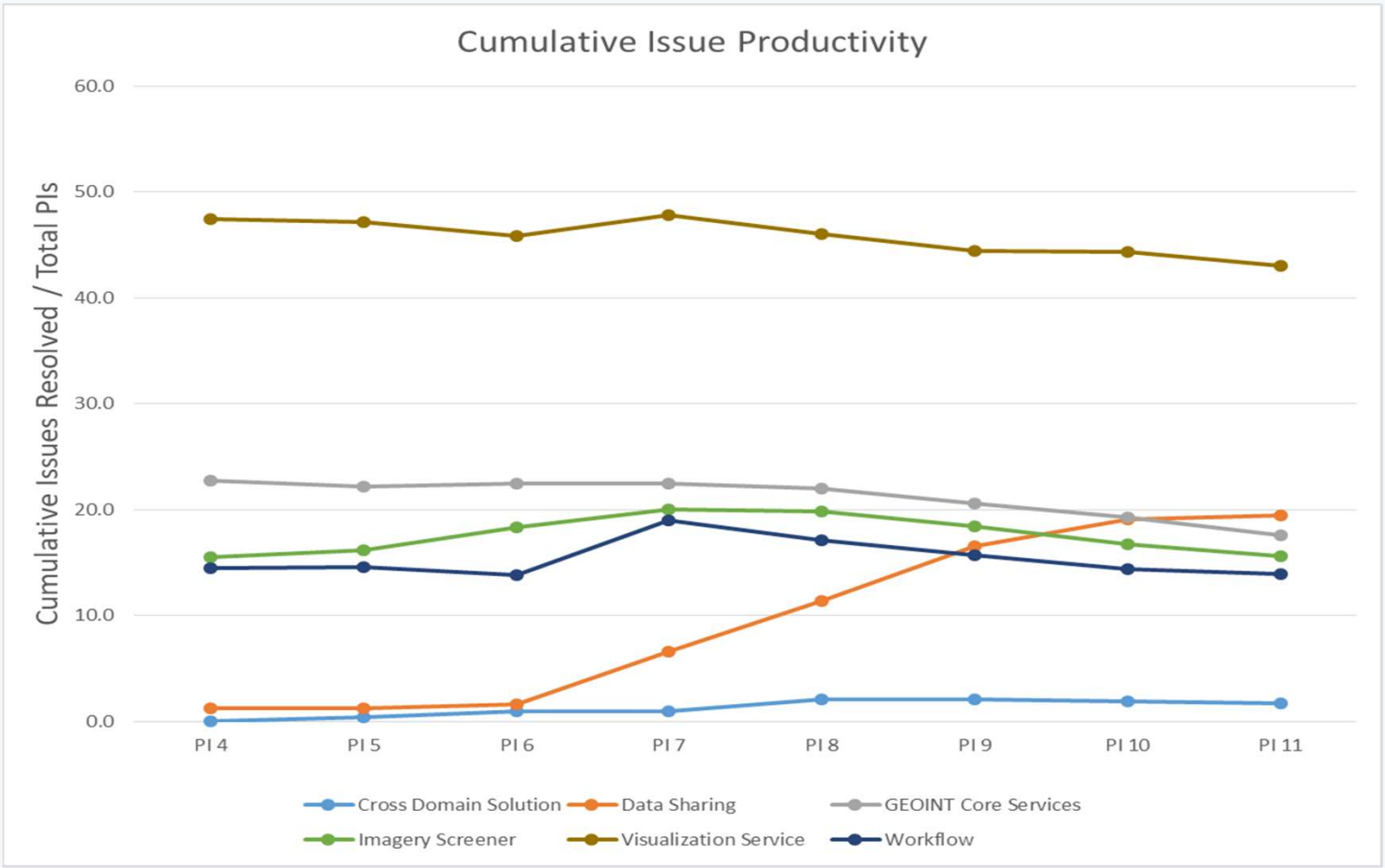
Software Metric Analysis: Emerging Techniques, Time Based Productivity



- **Analyze Productivity From a Number of Angles**
- **Shown Above: High Altitude ART, Cumulative Time Based Productivity of "Equivalent Feature" by Epic**

Case Study: AE2A in AF DCGS

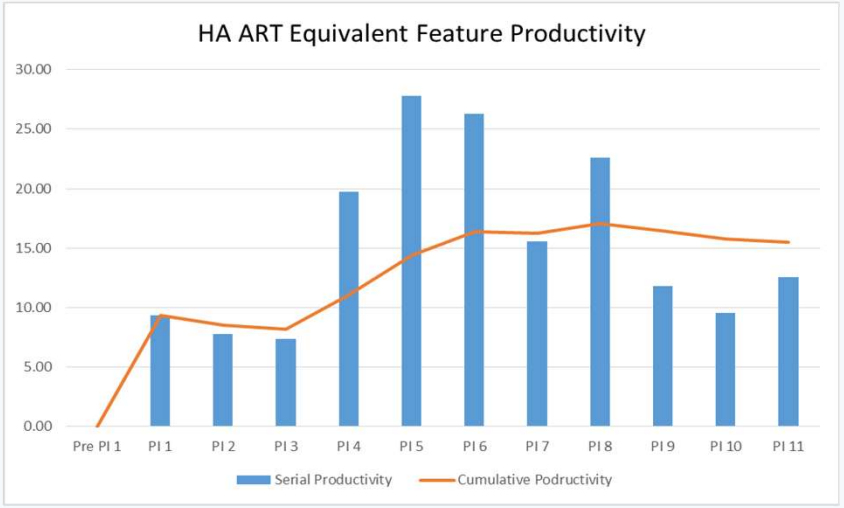
Software Metric Analysis: Emerging Techniques, Time Based Productivity

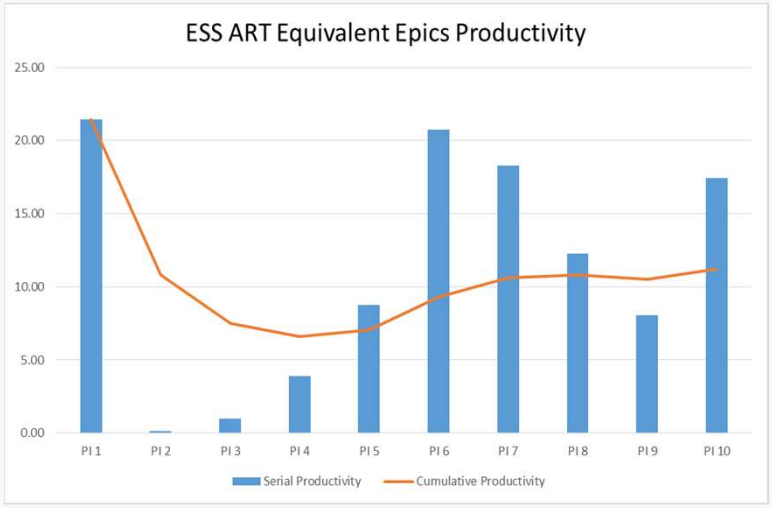


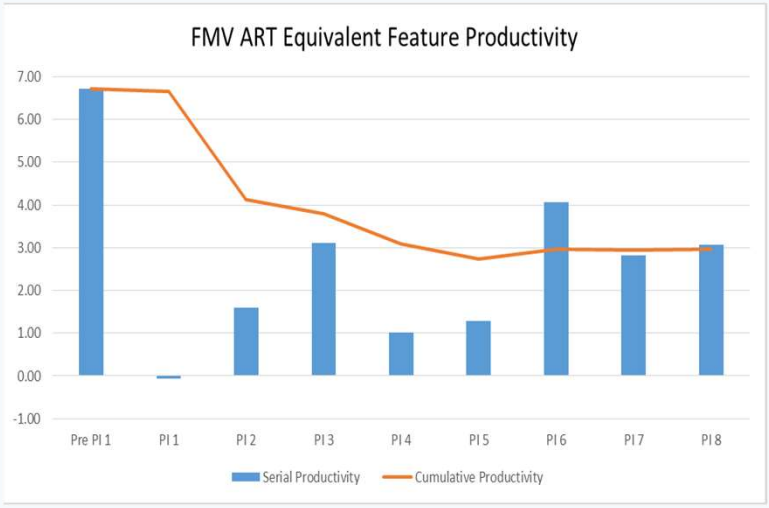
- **Smaller the unit of measure, the more consistent…but reqts do not come down at this small of a unit…**
- **Shown Above: High Altitude ART, Cumulative Time Based Productivity of Issues by Epic**

Case Study: AE2A in AF DCGS

Software Metric Analysis: Emerging Techniques, Time Based Productivity



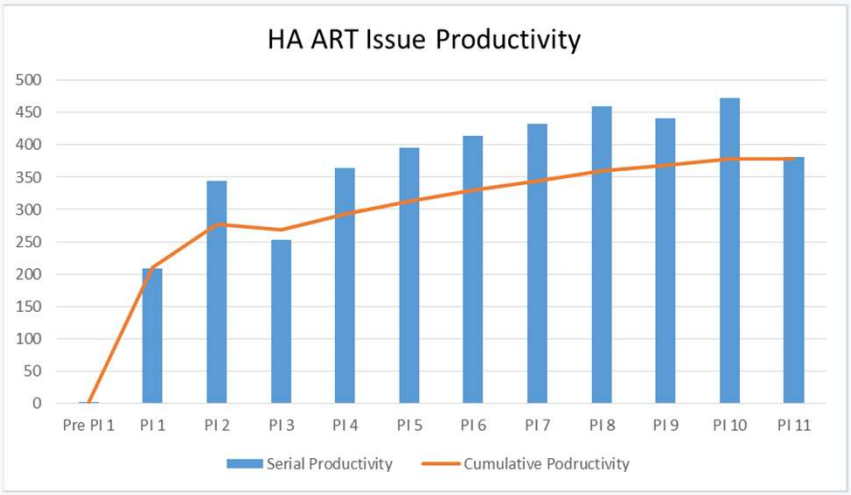


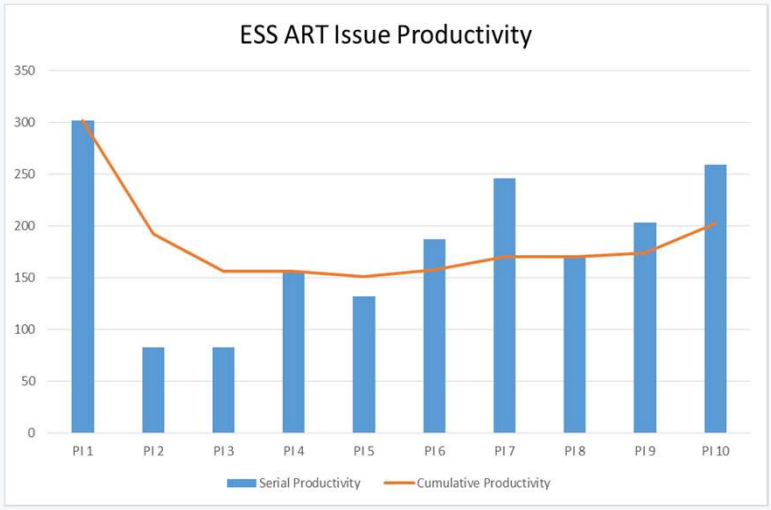


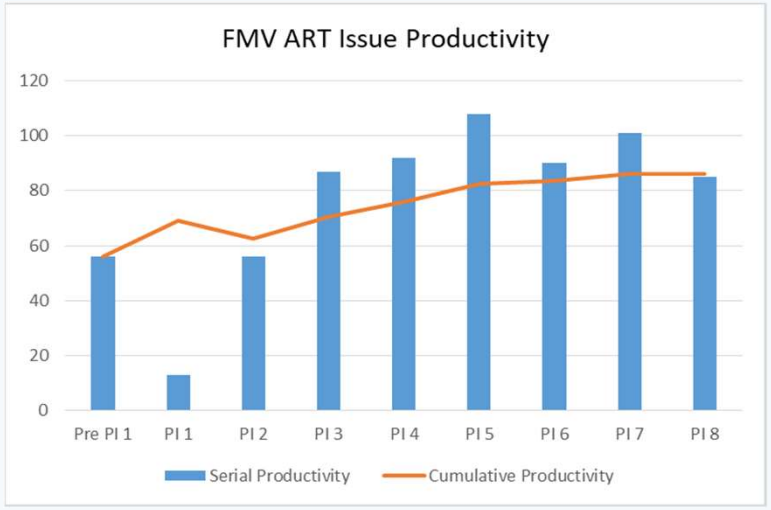
- **Time Based Productivities for Equivalency in each of the ARTs**
- **Be Careful When Utilizing Productivity Data "Too Early"**
- **Consider Productivity fluctuation over time as System Size and Complexity Increases**

Case Study: AE2A in AF DCGS

Software Metric Analysis: Emerging Techniques, Time Based Productivity



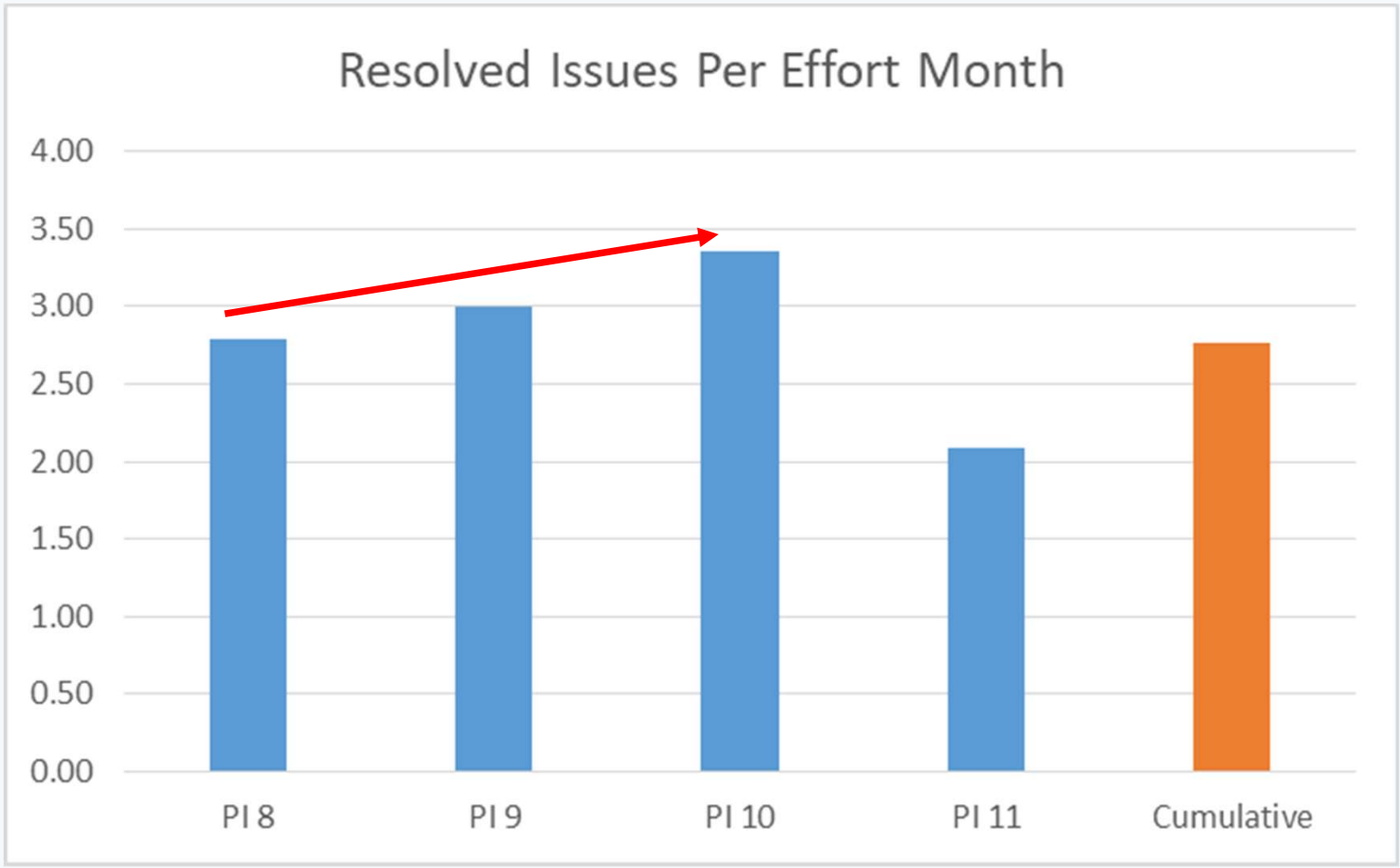




- **Time based Issue Productivity for each of the ARTs; lowest level of measurement**
- **Be Careful When Utilizing Productivity Data "Too Early"**
- **Consider Productivity fluctuation over time as System Size and Complexity Increases.**

Case Study: AE2A in AF DCGS

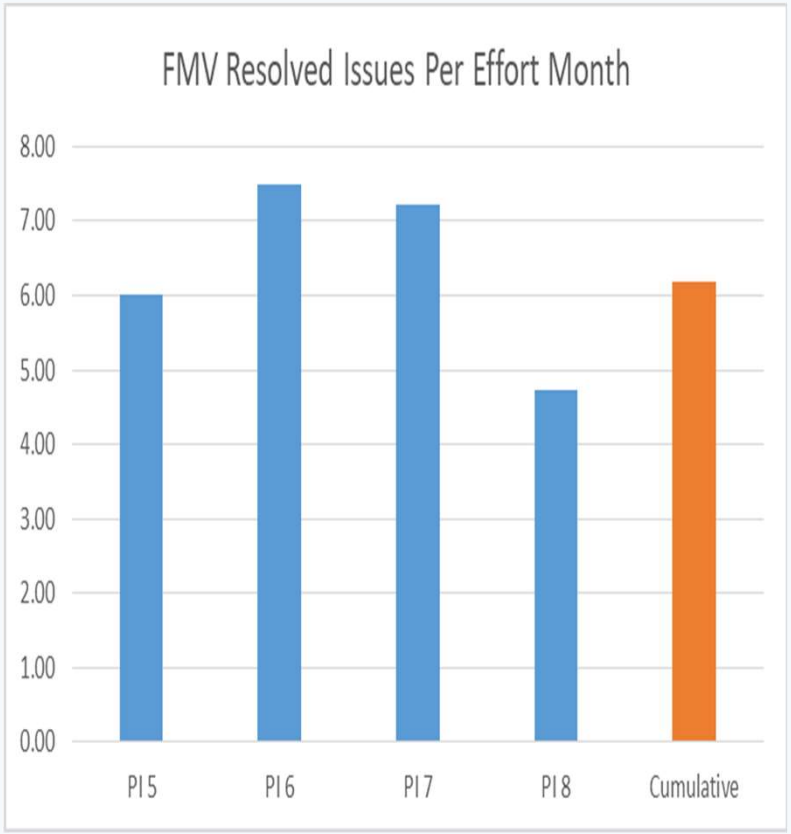
Software Metric Analysis: Emerging Techniques, Traditional Productivity

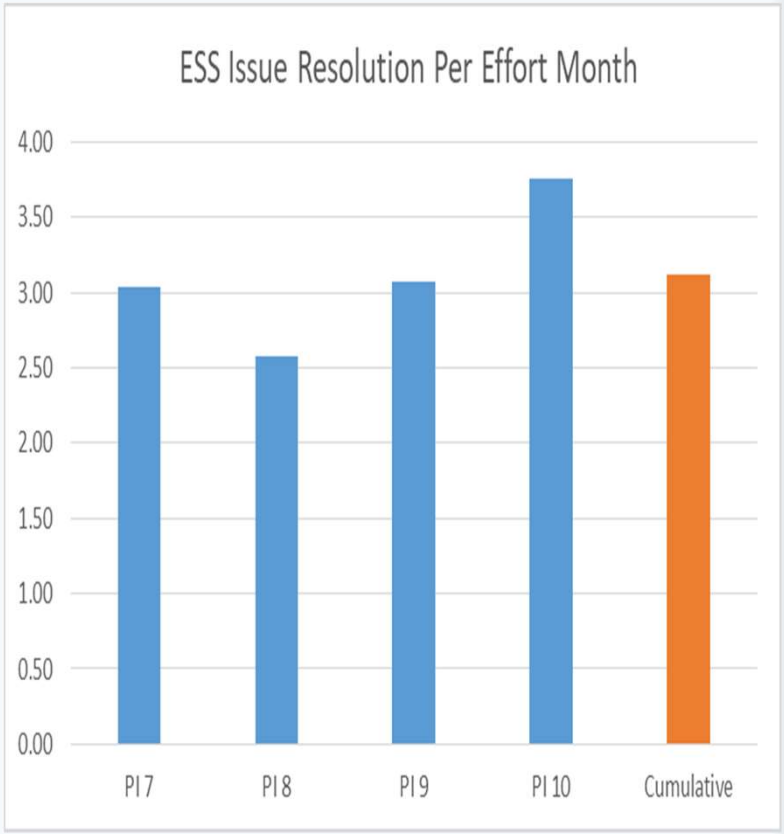


- **Analyze Productivity From a Number of Angles. Which data is most relevant? why? Should we fit a incline curve? Or should we utilize the cumulative average?**
- **Shown Above: High Altitude ART, Traditional Productivity, Issues Per Effort Month**

Case Study: AE2A in AF DCGS

Software Metric Analysis: Emerging Techniques, Traditional Productivity



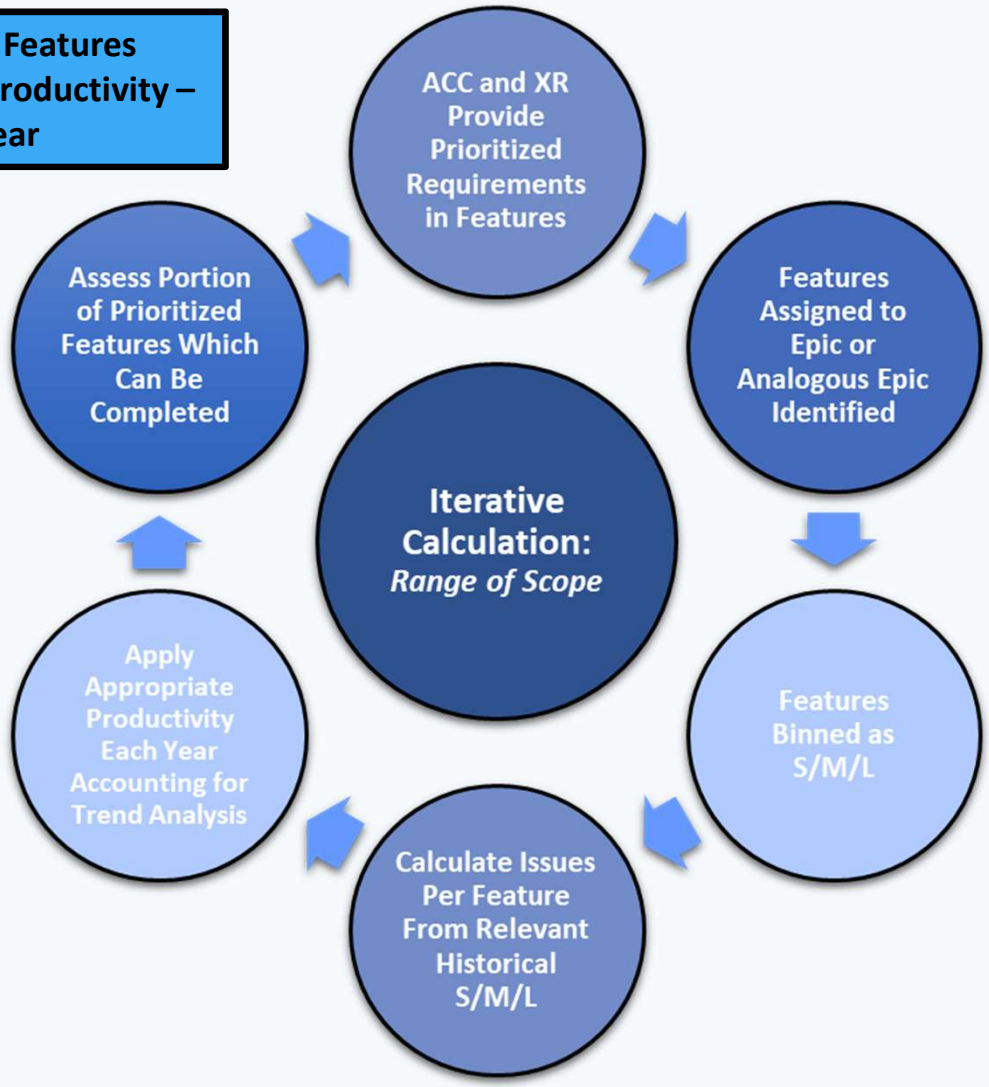


- **Analyze Productivity From a Number of Angles. Which data is most relevant? why? Should we fit a learning curve?**
- **Improvements are made every 3 months with new data from Jira and Confluence**
- **Shown Above: (left to right) FMV ART, ESS ART Traditional Productivity, Issues Resolved Per Effort Month**

Case Study: AE2A in AF DCGS

Estimate Result = Range of Possible Scope Each Year

HA ART spread of Features with Forecasted Productivity – 61 Features Per Year



- **Socializing the approach now, working to stand up the process in advance of Feb - Jun 2019 Portfolio 5 Year Plan Development Process**

Case Study: AE2A in AF DCGS

Software Metric Analysis: Initial Findings, Duration Analysis

	PIs to Date	Total Issues Created	Total Issues Resolved	Current Backlog Issues	Avg Issues Created Per PI	Cumulative Issues Productivity (Latest PI)	PIs to Complete (Current Backlog)
HA ART	11	4866	4170	696	405.5	379.1	1.84
FMV ART	8	951	688	263	105.7	86.0	3.06
ESS ART	10	2375	1822	553	215.9	202.4	2.73

	PIs to Date	Total Issues Created	Total Issues Resolved	Current Backlog Issues	Avg Issues Created Per PI	Cumulative Issues Productivity (Latest PI)	PIs to Complete (Increasing Backlog)
HA ART	11	4866	4170	696	405.5	379.1	Infinite
FMV ART	8	951	688	263	105.7	86.0	Infinite
ESS ART	10	2375	1822	553	215.9	202.4	Infinite

- **Duration Projections: Recommend Exploring from a Number of Angles**
- **How is your program structured? Finite or Infinite Requirements?**
- **What insight can you gain from the program engineering team to augment the data analysis findings? How is sustainment handled?**

Case Study: AE2A in AF DCGS

Way Forward: Integrating AE2A in the Program Office

MONDAY	TUESDAY	WEDNESDAY	THURSDAY	FRIDAY
WEEK THIRTEEN				
**Exporting Epic Hierarchies* **Extracting Automated Reports* **Epic Definitions & Descriptions*	Program Increment Solution Demo (Attend - Occurs End of Every 3 Months) Inspect and Adapt Event (Attend - Occurs End of Every 3 Months)	PI Planning Event (Attend - Occurs End of Every 3 Months)		
PI Planning				
WEEK ONE				
**Track creation of new Epics and Features as described at PI Planning Event (Week Thirteen)* **Track and Export Head Counts by ART*	**Track creation of new Epics and Features as described at PI Planning Event (Week Thirteen)* **Track and Export Head Counts by ART* **Add updated information to overall DCGS tracker*	**Track creation of new Epics and Features as described at PI Planning Event (Week Thirteen)* **Update Heads by ART ACE file with latest information*	**Track creation of new Epics and Features as described at PI Planning Event (Week Thirteen)*	**Track creation of new Epics and Features as described at PI Planning Event (Week Thirteen)* Daily Scrum Meeting - Attend
Sprint One (1)				
WEEK TWO				
				Daily Scrum Meeting - Attend
Sprint One (1)				
WEEK THREE				
		**PM tag up and feedback*		Sprint Summary Meeting - Attend
Sprint One (1)				

- **Align Process with SW Development Team Cadence**
- **Socialize Findings @ Key Events**
- **Work with User & XR in parallel to increase their desire….**

Case Study: AE2A in AF DCGS

Way Forward: Integrating AE2A in the Program Office

- Developed new approach to cost analysis in AF DCGS ARTs
  - Agile + Continuous Requirements + Continuous Development
- Working with Requirements Teams (XR/ACC) to implement AE2A process for next MYPEP
  - XR/ACC and the Value Stream Management Teams (VSMT) to decompose Requirements that will be aligned to an ART into Features by working with the individual ARTs
  - XR recommends implementing Story Point data when possible
- Working with HA, FMV, MICF ART to implement AE2A process for next MYPEP
  - Collaborating with ARTs at PI Planning events to collect necessary information and provide recommendations on DI2E usage to allow for further data analysis and metric collection
  - Developing project unique estimating methods based on the ART
  - ARTs recommend implementing Story Point data when possible
- Working with DCGS Agile Center for Excellence (DACE) to determine best practices for DI2E usage and associated metric collection
  - Program structure and FTE tracking
  - Epic / Feature / Issue sizing
  - Implementation of Hierarchies and Data Tagging / Linking (productivities associated with a specific capability, Story Point data, etc.)

Case Study: AE2A in AF DCGS

Way Forward: Wrap-Up

- Continue to collect and analyze new data every 3 months
- Continue to assess productivities over time
  - Time-Based and Traditional specific to the ARTs
  - Analyze causes of significant deltas in productivity
- Shift in focus of key product = scope range, no longer just the cost estimate range when applicable
  - Agile + Continuous Requirements + Continuous Development
- Implement AE2A process in practice for FY20 – FY24 MYPEP

Back Up

Case Study: AE2A in AF DCGS

Software Metric Analysis: Initial Findings, Duration Analysis

- If work were focused entirely on Features for a given year of time the teams could expect to complete 68 Equivalent Features, or a large sum of only small, medium, or large features

HA ART Feature Size	Average Issues Per Feature	Projected Features Completed with Forecasted Productivity (1 Year)	Projected Features Completed with Current Productivity (1 Year)
Small	2.03	996.3	745.9
Medium	6.09	332.6	249.0
Large	14.53	139.4	104.4
Equivalent Features	-	61.9	61.9

- **Duration Projections: Recommend Exploring from a Number of Angles**
- **How is your program structured? Finite or Infinite Requirements?**
- **What insight can you gain from the program engineering team to augment the data analysis findings? How is sustainment handled?**

Case Study: AE2A in AF DCGS

Software Metric Analysis: Initial Findings, Duration Analysis

- Using a ratio of issues aligned with Features compared to total issues within an ART allows for a more realistic percentage of Issue Productivity to be used to predict Features resolved

HA ART Feature Size	Average Issues Per Feature	Projected Features Completed with Forecasted Productivity (1 Year)	Projected Features Completed with Current Productivity (1 Year)
Small	2.03	175.9	131.7
Medium	6.09	58.7	44.0
Large	14.53	24.6	18.4
Equivalent Features	-	61.9	61.9

- **Duration Projections: Recommend Exploring from a Number of Angles**
- **How is your program structured? Finite or Infinite Requirements?**
- **What insight can you gain from the program engineering team to augment the data analysis findings? How is sustainment handled?**

Case Study: AE2A in AF DCGS

Software Development Teams: Specific Tailored Usage of Agile Terms & Software Progress Tracker Tool Configuration

- Even within the same program, approach to requirements and associated “agile” terminology may be different between Release Trains, for example…
- Enterprise Services Segment (ESS) ART:
  - Component is used to represent a large service area
  - Epics represent more specific capabilities within the Component
  - Features are rarely used within ESS JIRA
  - Epics are built up with average of 14.0 issues
- Example ESS ART Component:
  - ES – Identification, Authentication and Authorization (IAA)
- Example ESS ART Epic:
  - OADCGS-7091: One IM – IRP
  - Description: As a data/system owner, I want to restrict access so that I can protect the confidentiality and integrity of the system
    - As a user, I want to only be able to access the resources for which I’m authorized so that the confidentiality and integrity of systems are maintained
    - Focus is to create a central point of access control. All web interfaces will flow through the reverse proxy.

- **Understand How Each Team / Area Utilizes their SW Progress Tracking SW**

Case Study: AE2A in AF DCGS

Software Development Teams: Specific Tailored Usage of Agile Terms & Software Progress Tracker Tool Configuration

- Even within the same program, approach to requirements and associated “agile” terminology may be different between Release Trains, for example…
- High Altitude ART:
  - Components at the high level are not used
  - Epics generally represent larger areas which are not expected to “end”
  - Features represent more specific capability/functionality within the Epic
  - Features are built up with an average of 3.9 issues
- Example HA ART Epic:
  - 01. GEOINT Core Services Epic: For AF DCGS GH Block 30/40 GEOINT hub-based exploitation system that supports centralized enterprise level storage of GH Block 30/40 GEOINT data and delivery
- Example HA ART Feature:
  - Hub Based Map Server: Provides centralized access to foundation data via OGD-compliant web services

- **Understand How Each Team / Area Utilizes their SW Progress Tracking SW**