Technomics
Better Decisions Faster

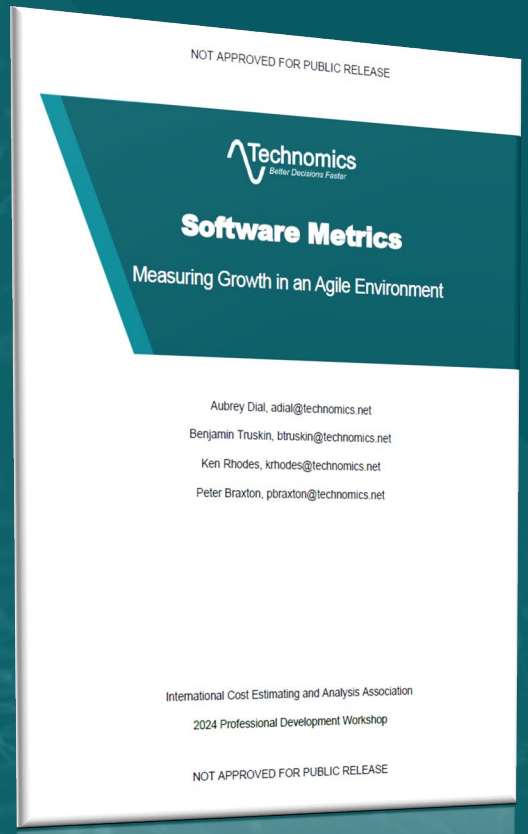
Software Metrics:
Measuring Growth in an Agile Environment

ICEAA Professional Development Workshop

May 2024

Agenda

Check out our Long-Form Research Paper!



- Background
- How Agile Actually Works
- Our Hypothesis – Change Traffic
- Descriptive Analyses – The Iron Triangle
- Predictive Analyses
- Conclusion and Next Steps

Our Team

Aubrey Dial

Lead Analyst

Aubrey is a CCE/A cost analyst with over 6 years of experience performing cost analysis, data manipulation, estimate development, acquisition decision support, and program execution support across several DoD customers. Aubrey is a 2023 ICEAA Washington Chapter Junior Analyst of the Year winner. She holds a BS in Mathematics and a MS in Data Analysis and Applied Statistics from Virginia Tech.

Ben Truskin

Service Area Manager

Ben is a CCE/A cost analyst with 11 years of experience spanning numerous commodities to include spacecraft, software, and defense systems. Other analytical experience includes, methods development, regression analysis, source selection support, data collection/normalization and data analytics. Ben holds a B.S. and M.S. in Aerospace Engineering from Pennsylvania State University

Ken Rhodes

Account Manager

Ken Rhodes is a CCE/A and Employee Owner at Technomics, Inc. He has seventeen years of experience performing cost analysis and acquisition decision support for DoD customers. Currently, Mr. Rhodes develops life-cycle cost estimates, cost / price assessments, and data visualization products for software and IT programs. He holds a BS in Industrial and Systems Engineering and MS in Systems Engineering from Virginia Tech.

Peter Braxton

Subject Matter Expert,
Director of Learning

Peter Braxton is a Subject Matter Expert and Employee Owner at Technomics, Inc. He has over 20 years of experience performing cost and risk analysis and delivering associated training for a broad spectrum of federal government clients. The inaugural VP for Professional Development and a multiple ICEAA Educator of the Year winner, he has shown a long-standing commitment to knowledge sharing within the community. He holds an AB in Mathematics from Princeton and an MS in Operations Research from the College of William and Mary. He is a semi-retired game show contestant and avid cruciverbalist.

BLUF

- Agile methods are designed to deliver capability to users sooner, not to mitigate cost/schedule growth
- Current agile metrics and approaches focus on delivery to user and not measures of productivity or value for acquisition
  - How much of the plan is accomplished each Program Increment (PI)?
  - Are we getting as much value as we planned (even if the scope continuously evolves)
- Our case study shows that program offices are not making informed decisions or conclusive measurements of development progress
- Our analysis results in a framework for leveraging already collected agile metrics to measure developer performance
  - Instead of just focusing on user-centric metrics

Agile – Coming Right at You!

- Profound changes in SW development affecting cost estimators
- Agile adoption drivers are well understood
  - Satisfies the customer with early and continuous delivery of valuable software
  - Welcomes changing requirements
  - Delivers working software frequently
- Agile does not mitigate all performance issues
  - 57% schedule growth, 24% scope growth
  - Average completion of only 77% of planned scope

SW Development Project Outcomes

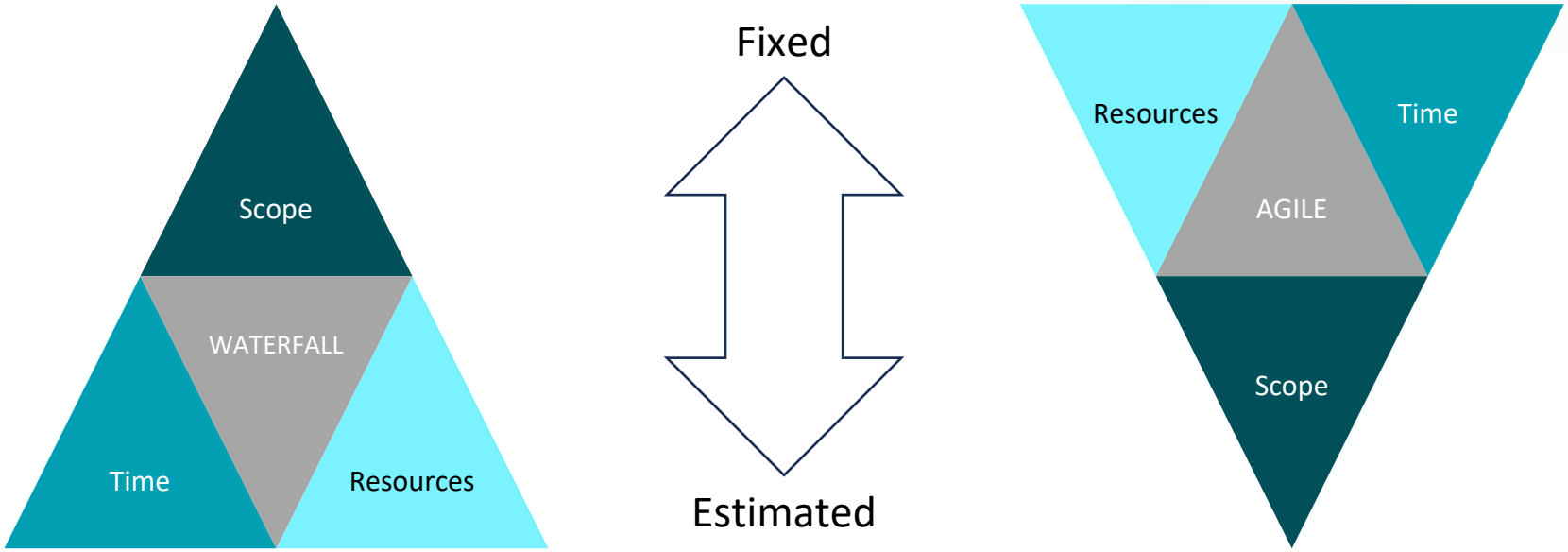
SIZE	METHOD	SUCCESSFUL	CHALLENGED	FAILED
All Size Projects	Agile	39%	52%	9%
	Waterfall	11%	60%	29%
Large Size Projects	Agile	18%	59%	23%
	Waterfall	3%	55%	42%
Medium Size Projects	Agile	27%	62%	11%
	Waterfall	7%	68%	25%
Small Size Projects	Agile	58%	38%	4%
	Waterfall	44%	45%	11%

The Standish Group. (2015). 2015 CHAOS Report. Retrieved from https://standishgroup.com/sample_research_files/CHAOSReport2015-Final.pdf

Agile – The Dilemma

- Agile approach prioritizes responsiveness and replanning over baselining
- Appears agile would lend itself most to estimate via Extrapolation from Actuals…but!
  - USG is way behind in implementing policy to enable collection of development performance
  - USG development contracts (and budget) extend far into the future
- Situation may get better with development of the Software Acquisition Pathway (DODI 5000.87)
- In the meantime, we estimators must keep doing what we do best…make prognostications in the world as it is

What is important in Agile



- Estimators must adapt to what acquisition offices value

Why Traditional Methods Will Not Suffice

Sizing

- Often lacking a complete baseline
- The backlog can be “arbitrary”
  - Curtailed based on time or PoP
  - Continuously added to
- Lack of required sizing metrics
  - Especially in move towards FP type contracts
- Lack of consensus and high variability in sizing approaches
  - Further discussion below

SW Productivity

- Fixed schedule blocks can make measuring productivity a challenge
  - “We’ll get done 1 sprint worth of effort!”
- Cultural challenges of measuring and comparing across teams
- Impacts of customer interaction more pronounced
  - Delays in PI planning/closeout
  - Amount of re-prioritization activities
- Testing and Deployment approach can greatly impact developer focus

Metrics, Metrics Everywhere

- Team acknowledges that current agile adoption includes lots of metrics

- DevOps Research Association (DORA) set:
  - Delivery time: Average time to deliver a Feature into operations
  - Availability: Percentage of the time the system is available to users
  - Deployment frequency: How often new releases are pushed into operations

User focus

- However, there are no common standards for metrics related to development or project execution

- User focused metrics do not align to development schedule or performance

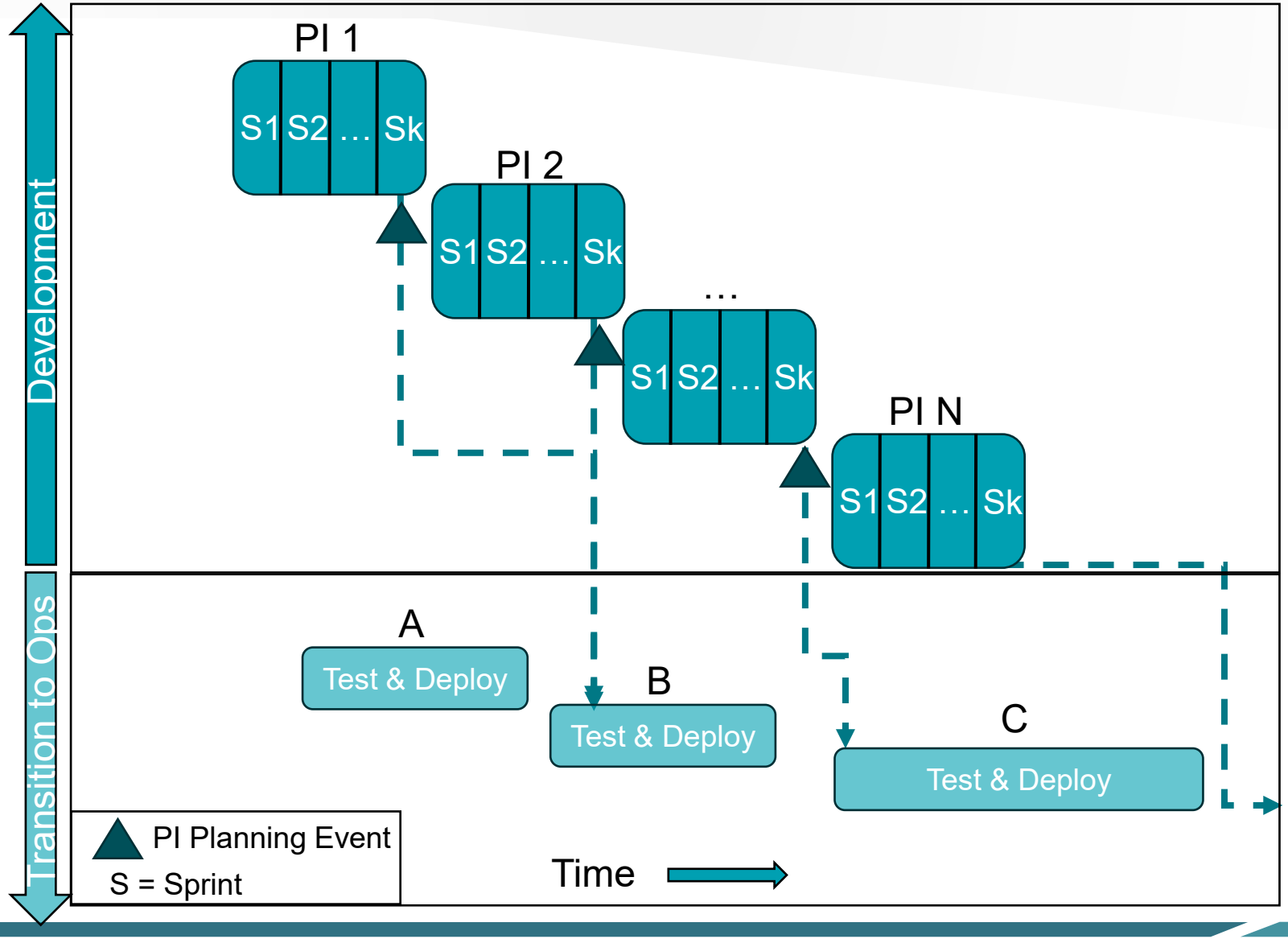
Dr. Nicole Forsgren, D. D. (2019). State of DevOps 2019. DevOps Research and Assessment (DORA).

Dashboards … Sands of Time

- The ethos of constant customer interaction lends agile development to leverage metrics dashboards

- Dashboards = Instant access to loads of data…what is the catch?!
  - Immediacy – Focus on recent history
  - Unclear completeness – Unclear what portion of data given user sees! Admin access not a given
  - Consistency
    - Dashboards are subject to change
    - If every program has unique dashboards, it presents a whole new learning curve
    - Data feeds over time subject to random variation

How Agile Actually Works…



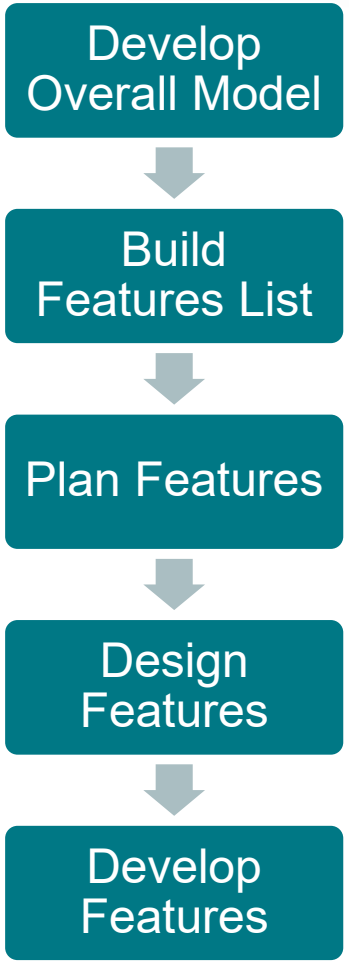
Features … Featureless

- Feature: a unit of functionality of a software system that satisfies a requirement, represents a design decision, and provides a potential configuration option

- Similar to other agile sizing approaches: T-shirt sizing, story point, etc.
  - Increments decompose to sprints, sprints decompose to agile sizing approach

- 'Goldilocks' sizing approach about what can be completed in a PI

Feature Driven Development Methodology



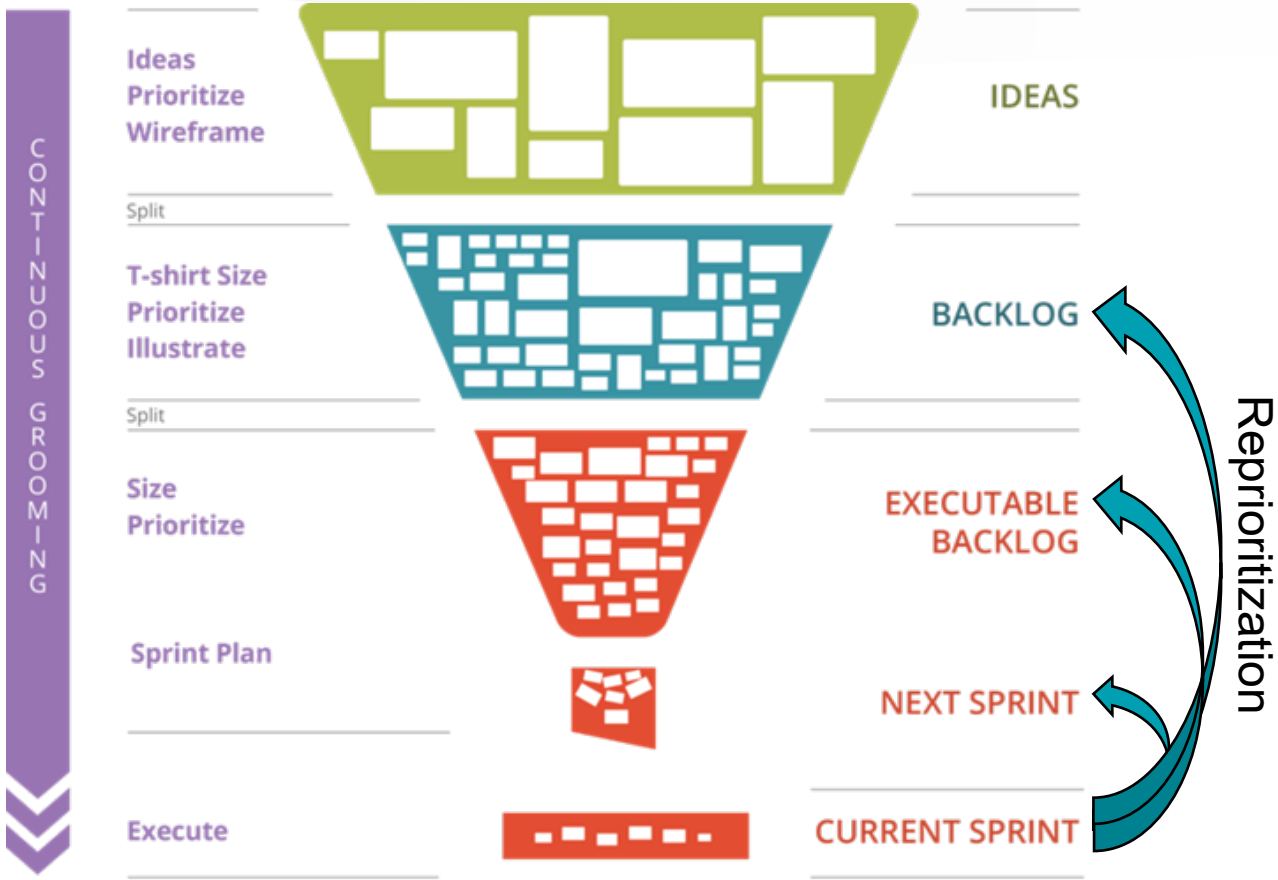
Apel, S., & Kästner, C. (2009). An Overview of Feature-Oriented Software. *Journal of Object Technology*, 1-36.

...but what is missing?

- The prior graphic leaves out a key piece of agile effort
  - How does progression through sprints work?!

- As the backlog is worked, features end up in one of five statuses

Type	Description
Complete	Feature successfully developed and ready for integration into software baseline
Add	Feature added for development in current PI that was not originally in PI
Delete	Feature slated for development in current PI is determined to be no longer relevant, and it's removed from the development queue
Move	Feature slipped from current PI to a later PI for future development
Split	Feature is broken up across multiple PIs, with some work done during the current PI on the feature and some work pushing to future PIs, usually the succeeding one.

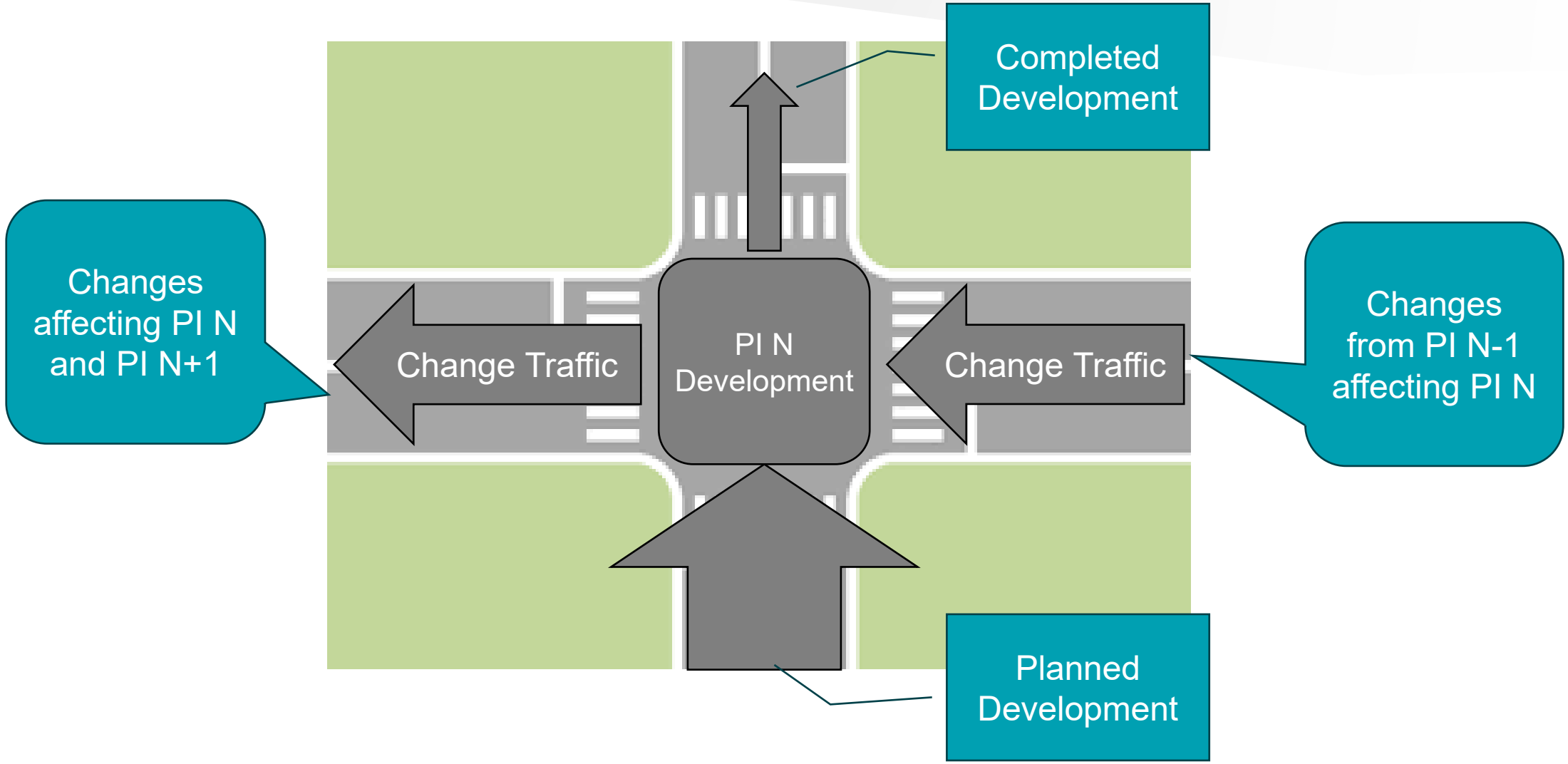


Graphic credit: ITX

Performance Analysis that must not be named

- Agile programs (and fixed price contracting) are afraid of EVM
- Because long term baselines are not made in agile, the importance of measurable performance over incremental periods is even more important
  - Plug for DoD SW Acquisition Pathway and DI 5000.87 for not eschewing data collection requirements
- Our team looked to adapt existing Agile Estimating Framework approaches
  - Our method is independent of sizing strategy and agile execution CONOPS

The 12-Lane Highway



Our Hypothesis

- Don’t just measure what’s completed…measure what capacity should be
  - Also know as: The Plan

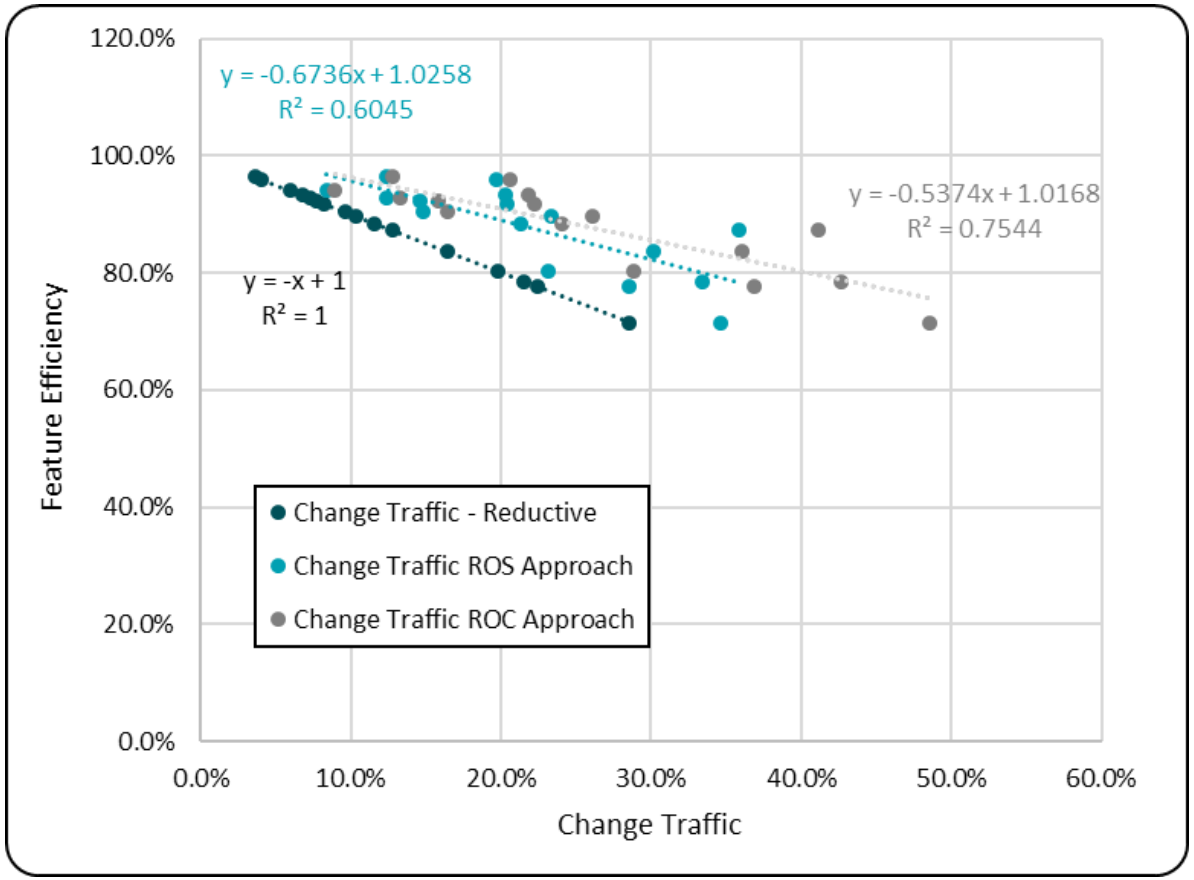
$$Feature\ Efficiency \propto Change\ Traffic$$

$$Feature\ Efficiency = \frac{Completed\ Features}{Planned\ Features}$$

$$Change\ Traffic = \frac{Added\ Features\ +\ Removed\ Features}{Planned\ Features}$$

$$Change\ Traffic = \frac{\sum Features\ (Add, Move, Split, Delete)}{Planned\ Features}$$

But why Change Traffic?



1. We needed a unitless equation…but what denominator?
   - Return-on-Cost (ROC): Actual Features
   - Return-on-Sales (ROS): Planned Features
   - **ROS is predictive**, ROC is descriptive
2. We needed to measure traffic in both directions
   - Addition of *signed* traffic is no good … 0 adds and 0 deletes ≠ 100 adds and 100 deletes

Into the Data*

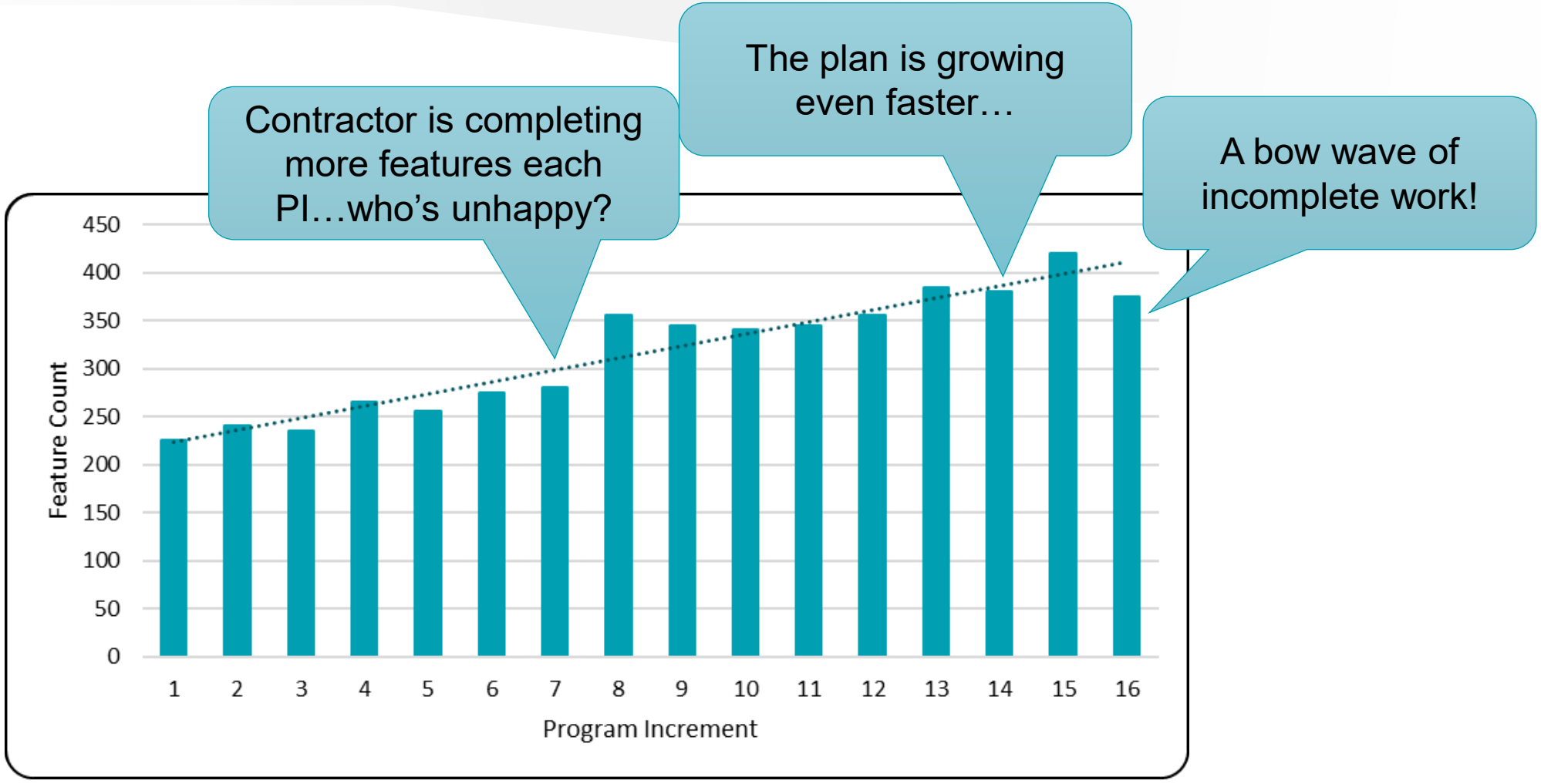
A Case Study

- Exemplar data, modeled after observed behavior
- 3-month Increments, comprising 4 sprints each
- Real development made up of numerous swim lanes ("12-lane highway")
  - Not considered for example data

Feature Count by Category and PI

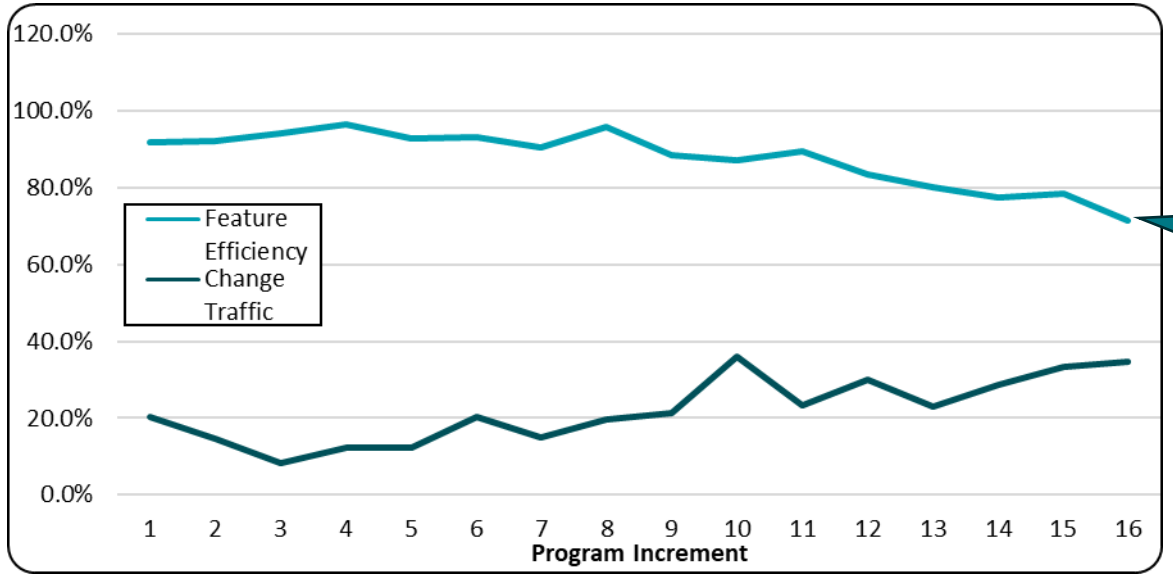
PI	Planned	Actuals	Shortfall	Added	Removed	Split	Moved	Deleted	Change Traffic
1	245	225	20	15	35	19	7	9	50
2	260	240	20	9	29	21	0	8	38
3	250	235	15	3	18	3	5	10	21
4	275	265	10	12	22	4	11	7	34
5	275	255	20	7	27	0	25	2	34
6	295	275	20	20	40	10	19	11	60
7	310	280	30	8	38	5	3	30	46
8	370	355	15	29	44	29	13	2	73
9	390	345	45	19	64	31	22	11	83
10	390	340	50	45	95	33	45	17	140
11	385	345	40	25	65	40	25	0	90
12	425	355	70	29	99	8	74	17	128
13	480	385	95	8	103	28	70	5	111
14	490	380	110	15	125	45	70	10	140
15	535	420	115	32	147	77	60	10	179
16	525	375	150	16	166	73	88	5	182

What is the problem?



Iron Triangle – Cost

- Correlation of Feature Efficiency and Change Traffic is significant ($\rho = -0.77$) and not spurious ($p - value = 3.9 * 10^{-5}$)

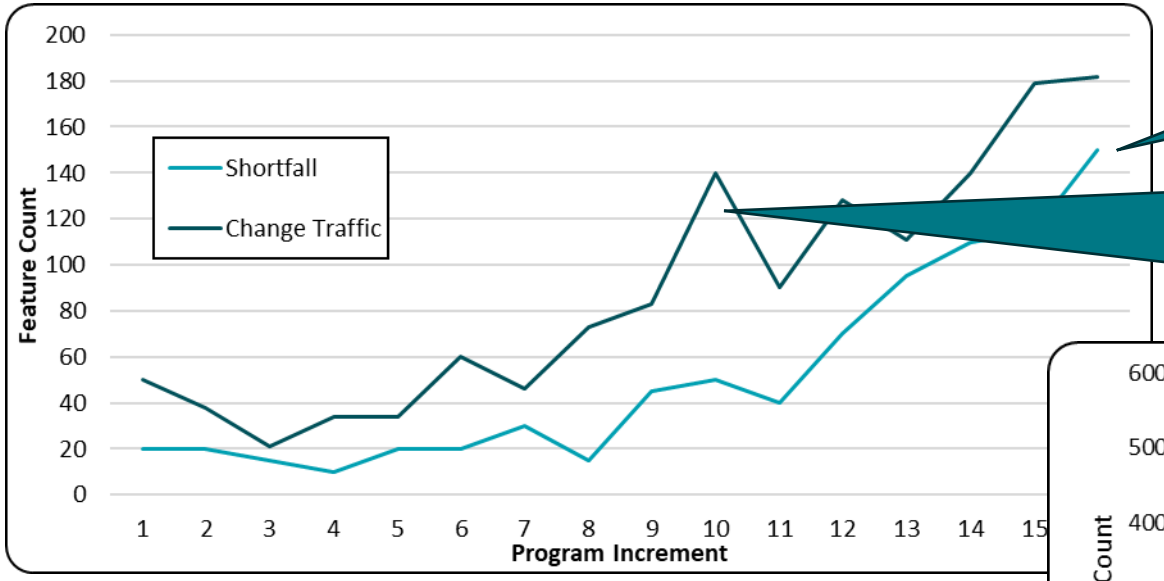


Is it ok to deliver only 77% of features if they're the 77% most important ones?

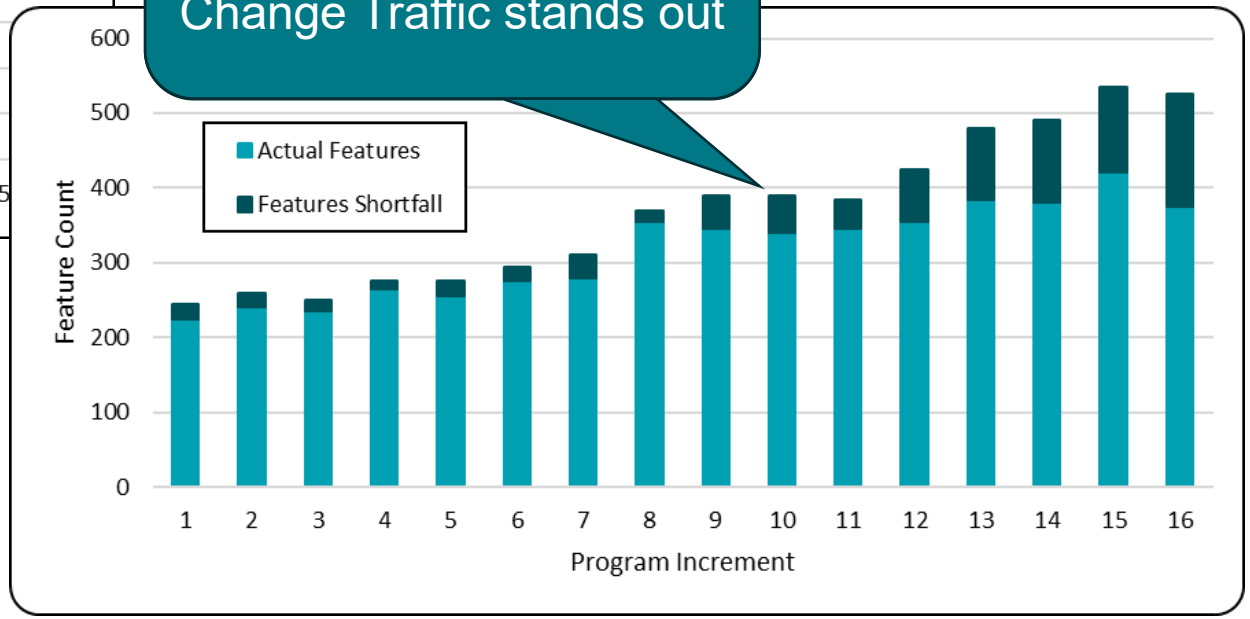
Iron Triangle – Schedule

Growing shortfall means delivery to user must slip

Contract extension!?

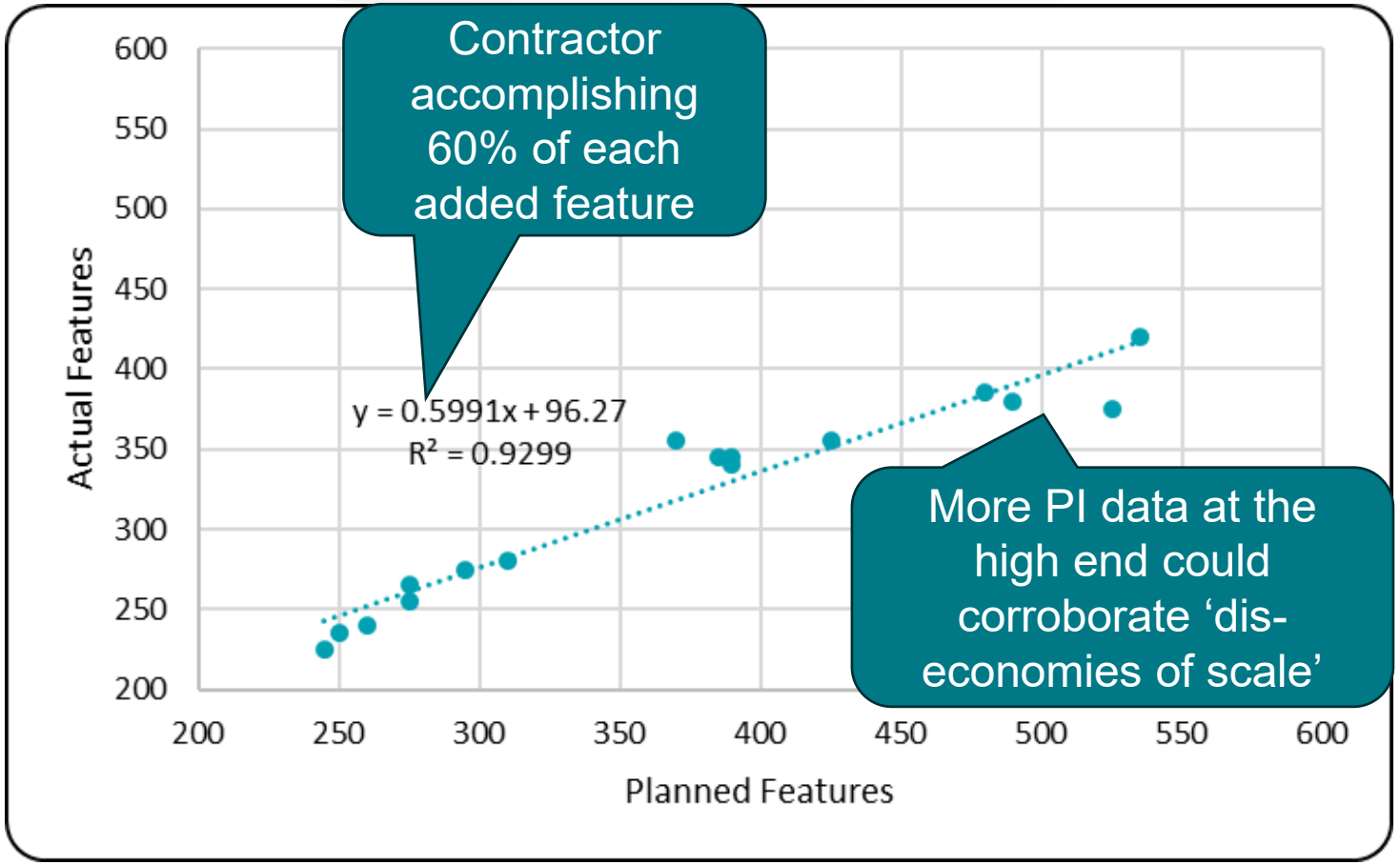


Shortfall at PI 10 looks nominal, however Change Traffic stands out

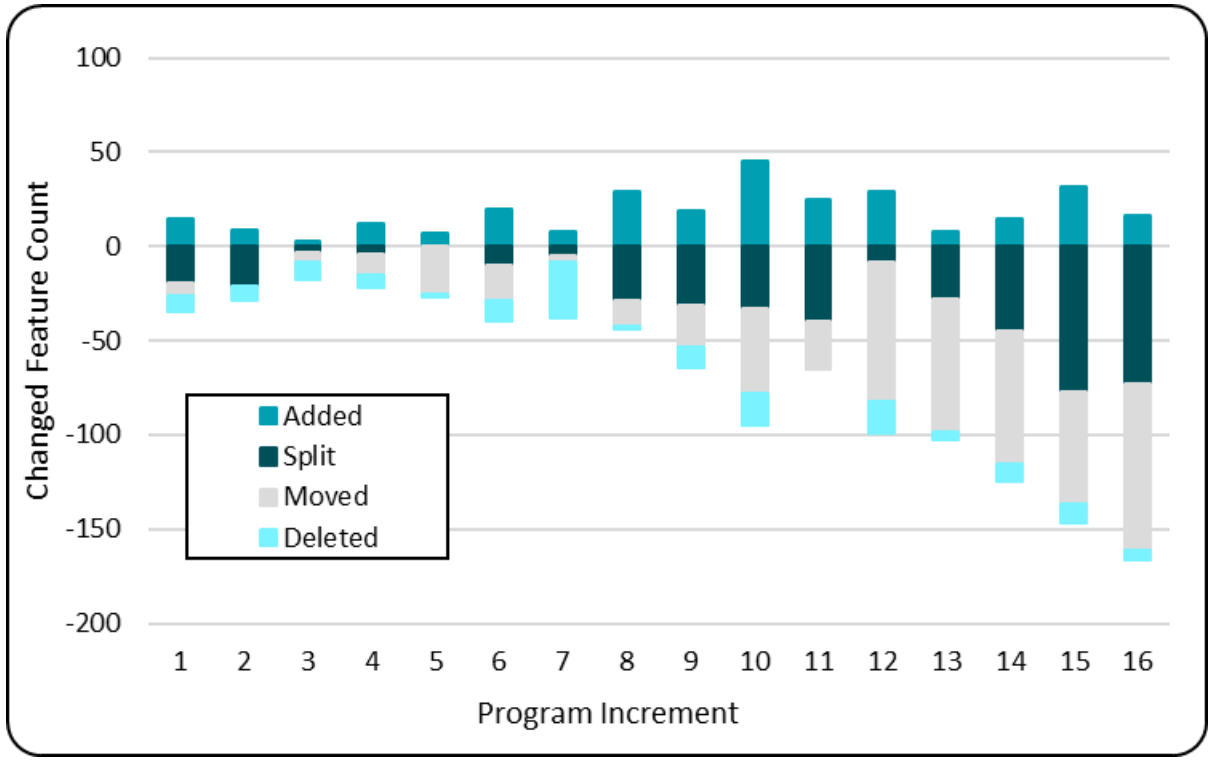


Iron Triangle – Technical (pt 1)

- It doesn't appear the contractor can just plan their way out of this via more capacity…the scalability isn't there



Iron Triangle – Technical (pt 2)



- Added features don't appear to move the baseline!
  - Initial team assumption not borne out in data
- Move/Split very impactful!
  - Double whammy of impacting current PI and re-prioritizing in near future
- Delete appears relatively uncorrelated
  - Score one for intuition!

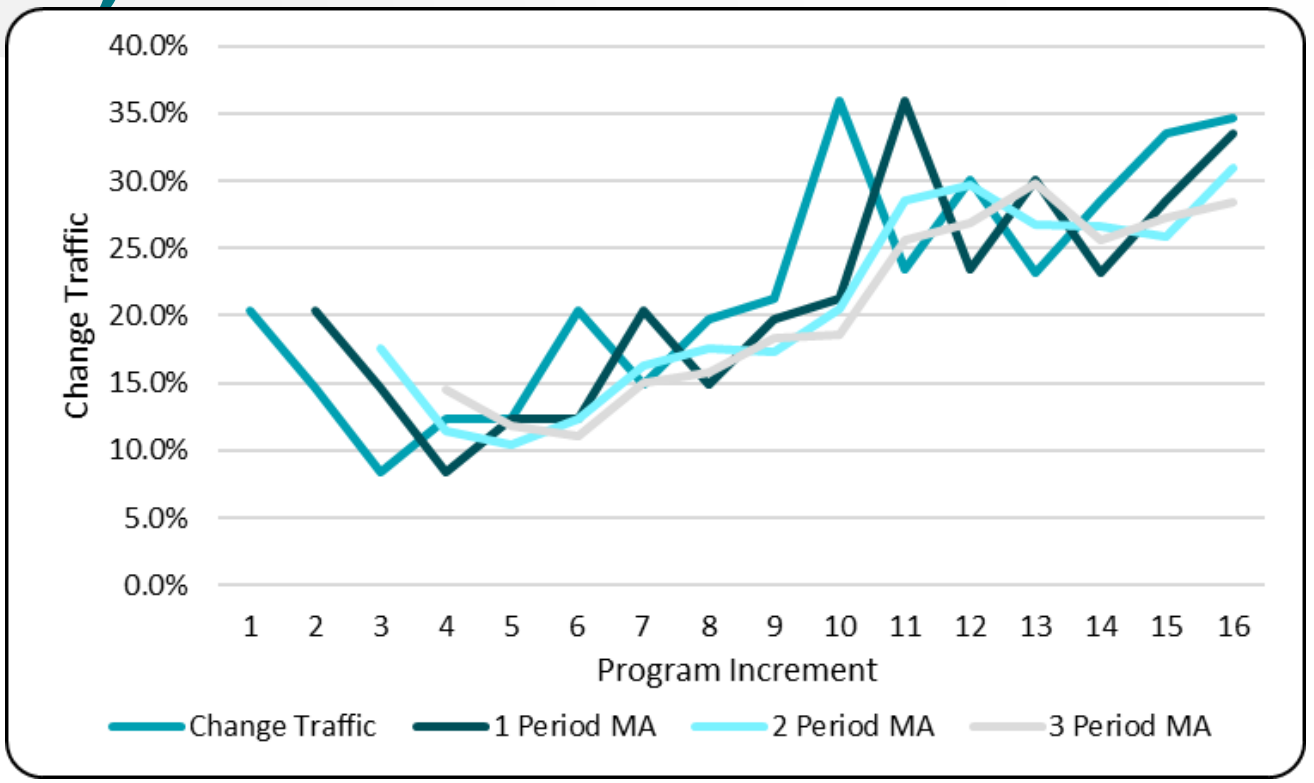
Predictive Analyses (pt 1)

Correlation of Prior PIs

	N-0	N-1	N-2	N-3
R^2	0.604	0.561	0.468	0.458
Correlation	-0.78	-0.75	-0.68	-0.68

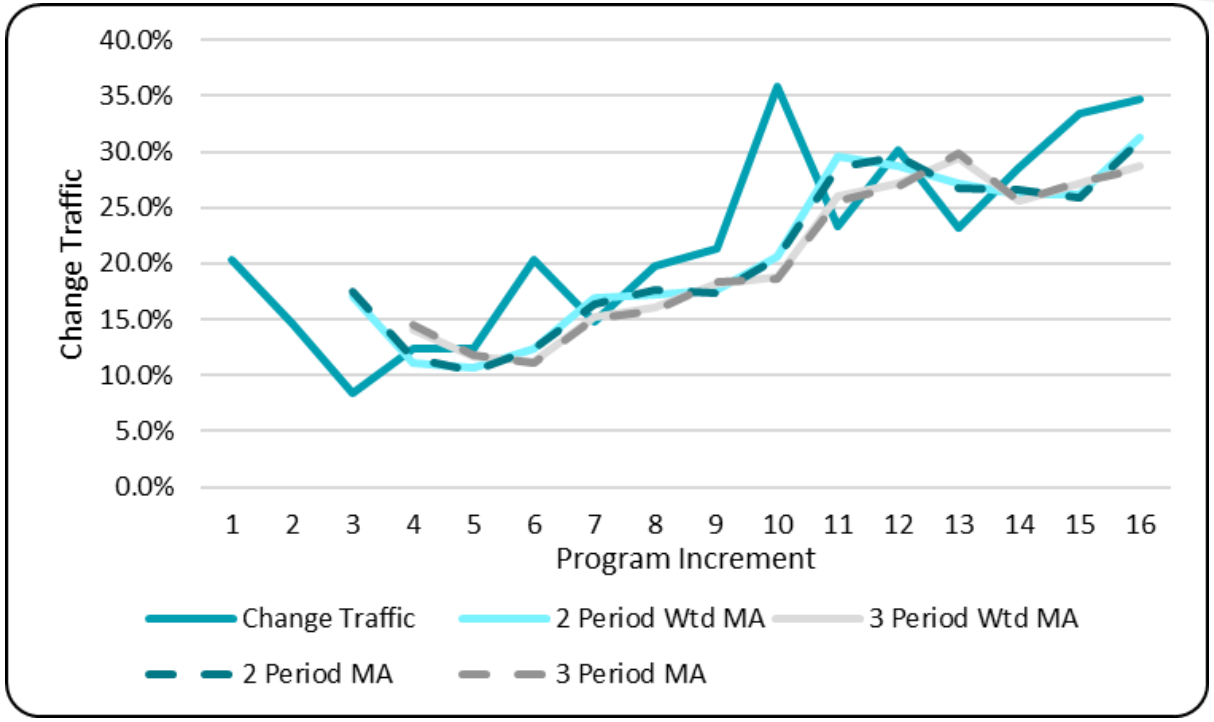
Correlation of Moving Average (MA)

	0 Period	1 Period	2 Period	3 Period
R^2	0.604	0.561	0.661	0.706
Correlation	-0.78	-0.75	-0.81	-0.84



- Strong correlation overall
- Persistent of correlation to prior experience should make resilient predictive models

Predictive Analyses (pt 2)



Correlation of Optimized Weighted MA PIs

	0 Period	1 Period	2 Period	3 Period
N-3 Weight	N/A	N/A	N/A	0.294
N-2 Weight	N/A	N/A	0.430	0.353
N-1 Weight	N/A	N/A	0.570	0.353
R^2	0.604	0.561	0.665	0.707
Correlation	-0.78	-0.75	-0.82	-0.84

- Very minimal improvement in performance for 2- and 3-period weighted Moving Averages

A little white lie…

- The earlier introduction of Change Traffic was a little simplistic
  - Why should all feature changes affect things equally?

$$Change\ Traffic = \frac{\sum Features\ (\alpha * Add, \beta * Moved, \gamma * Split, \delta * Delete)}{Planned\ Features}$$

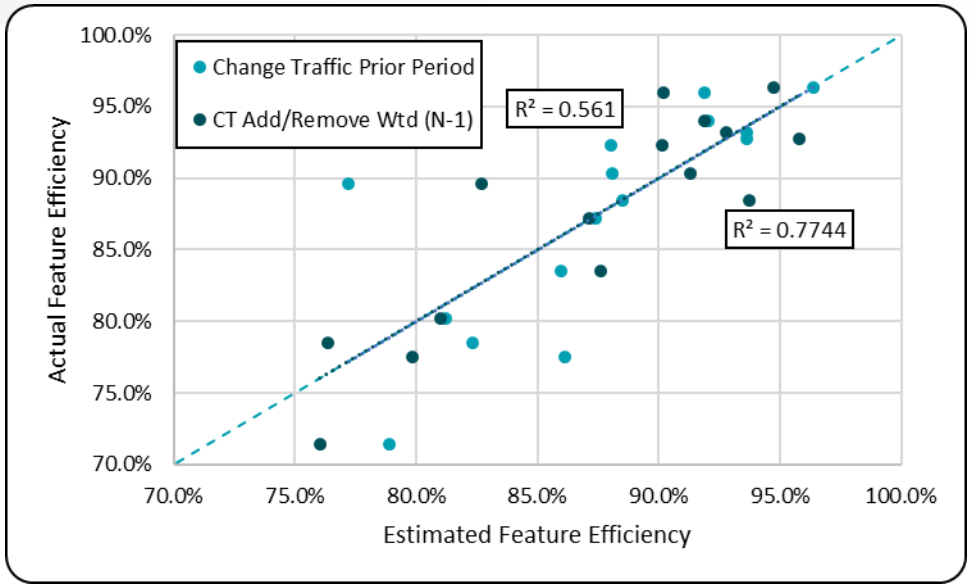
- We can specify or solve for α-δ
  - But beware multicollinearity!

Cross-correlation by Feature Type

	Add	Split	Moved	Delete
Add	100%			
Split	42%	100%		
Moved	31%	62%	100%	
Delete	8%	-24%	-8%	100%

Feature Weighting Optimization

- Our team attempted 24+ regressions using various measures of change traffic and feature types
- Select results presented below to showcase improved R^2 over "ideal"
  - Ideal being perfect knowledge of PI N Change Traffic



#	Name	Functional Form	R^2	SSE	Model F
1	Unweighted Change Traffic (N-0)	FE(N) = a*CT(N)+b	0.604	0.049	21.4
2	Unweighted Change Traffic (N-1)	FE(N) =a*CT(N-1)+b	0.561	0.045	16.6
3	Unweighted Change Traffic (N-2)	FE(N) =a*CT(N-2)+b	0.468	0.036	10.6
4	Add/Remove Wtd. Change Traffic (N-1)	FE(N)=(α*Add+β*(Split+Moved+Delete))/Planned+b	0.734	0.058	16.6
5	Add/Remove Wtd. Change Traffic (N-2)	FE(N)=(α*Add+β*(Split+Moved+Delete))/Planned+b	0.748	0.058	16.3
6	Delete Wtd. Change Traffic (N-1)	FE(N)=(α*Delete+β*(Split+Moved+Add))/Planned+b	0.672	0.054	12.3
7	Moved Wtd. Change Traffic (N-1)	FE(N)=(α*Moved+β*(Split+Add+Delete))/Planned+b	0.683	0.054	12.9

Conclusions

- Our results provide a **comprehensive framework to measure development performance** in agile systems
  - Micro-level: Performance in a contract across sub-projects or time
  - Macro-level: Performance health metrics and standards for estimating future systems
- **Clear and steady** inverse relationship between Change Traffic and Feature Efficiency
  - Change traffic is a prototypical regressive driver
- **Don’t throw away fundamentals!** It’s great to celebrate performance but if the trend keeps falling behind plan that’s still a problem
- **Manage proactively.** If poor performance occurs, look to re-baseline further than PI N
- **No such thing as a free lunch.** Any change in requirements should come with an expectation of changing output

The Importance of Policy and Contracts

- Our team had to hand-copy data from PPTs for two years to really get traction
- No policy about collecting, reporting, or making developer progress data available
- No requirement to align costs to developer activities (the “V” in EV)

- Goals
  - Inform metrics and dashboard for developer progress that can be shown to stakeholders just like user focused dashboards today
  - Use this data to identify contracting strategies to mitigate budget risk for government due to persistent growth

Next Steps

- Broaden the aperture – Working with other acquisitions in Civilian and Defense organizations
- The uncertainty principle – Is change traffic driving feature inefficiency or the other way around?
- Deeper technical dives – Do the type of feature/development (or other variables) infer multivariate relationships?
- Further enhancements – Continue to collect data and perform analysis on other factors to assess the correlation to team size/staffing over time, extensible to other agile sizing approaches, and SME-derived weights for feature types

Technomics
Better Decisions Faster

Questions?

adial@technomics.net
btruskin@technomics.net
krhodes@technomics.net
pbraxton@technomics.net