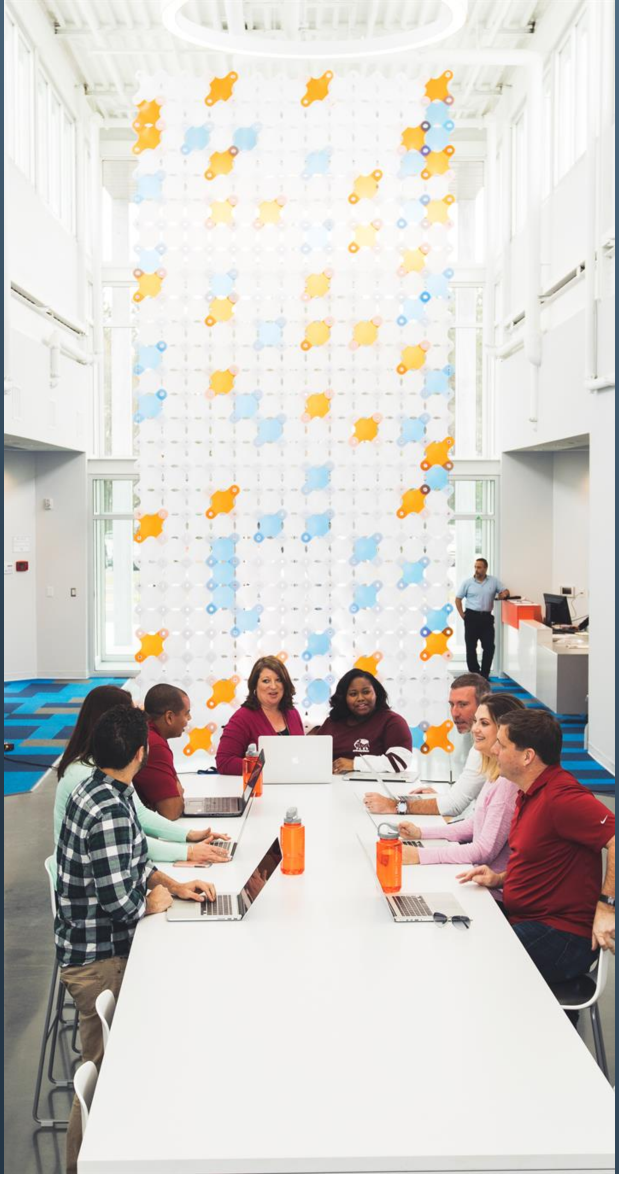
ICEAA International Cost Estimating and Analysis Association

Booz | Allen | Hamilton



AGILE DEVELOPMENT COST FACTORS CASE STUDY

Blaze Smallwood, CCE/A

ICEAA Conference 2018

JUNE 2018

AGENDA

INTRODUCTION

METRICS DETAILS

SUMMARY

PURPOSE

- Explore cost, schedule, performance metrics for a small collection of DoD agile software projects
- Determine if any trends exist and any rules of thumb can be derived
- Highlight major takeaways

CASE STUDY PROJECTS

- Completed DoD Automated Information System (AIS) software development/integration projects

Project / Marker	ACAT	Performer (GOV/KTR)	ALM Tool Used	Cost ($M) ***	Schedule (Months)
A Ⓐ	III	KTR	Forge	$5.6	20
B Ⓑ	III	KTR	Jira	$4.0	21
C Ⓒ	I	GOV	Jira	$21.2	18
D Ⓓ	III	KTR	TFS	$10.2	19
E Ⓔ	N/A**	KTR	Jira	$1.3	14
F* Ⓕ	N/A**	GOV	Jira	$7.4	11

* Project had no specific end date; schedule indicates # of months data was collected

** Pre-Acquisition risk reduction projects

*** Full cost of the software development/integration project; excludes non-PMP costs, like PMO costs

Acronyms: ACAT = Acquisition Category; ALM = Application Lifecycle Management; TFS = Team Foundation Server

PROJECT DATA SUMMARY

- Projects had varying levels of data available

Metric	A	B	C	D	E	F
Cost per Point	X	X	X	X	X	X
Hours per Point	X	X	X	X	X	X
Cost per Requirement	X	X	X	X	X	X
Hours per Requirement	X	X	X	X	X	X
Cost Variance	X	X	X	X	X	
Schedule Variance	X	X	X	X	X	
Scope Variance	X	X	X	X	X	
Team Composition			X	X		
Buffering Percentages	X	X	X	X	X	X

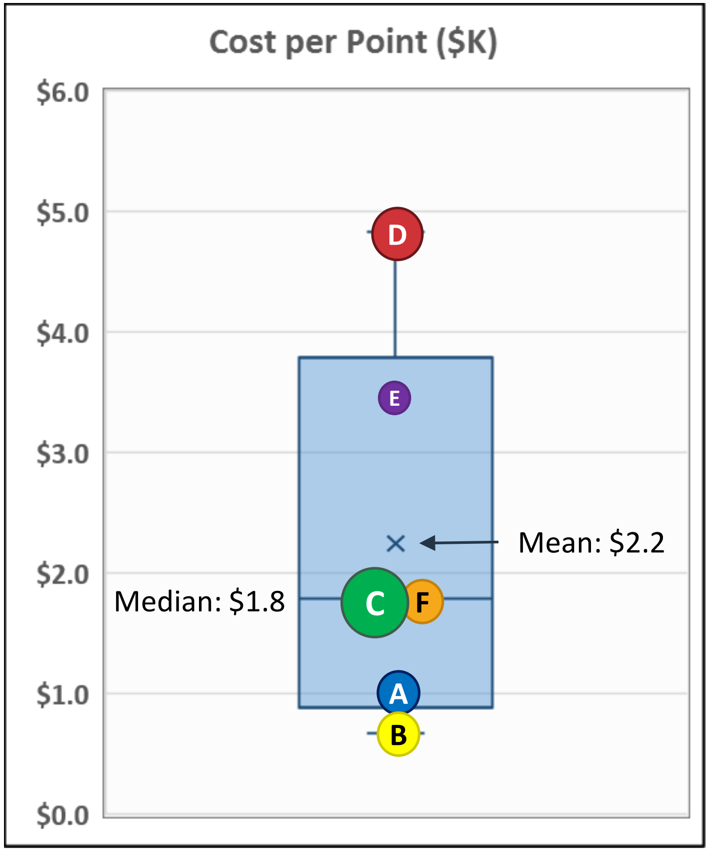
AGENDA

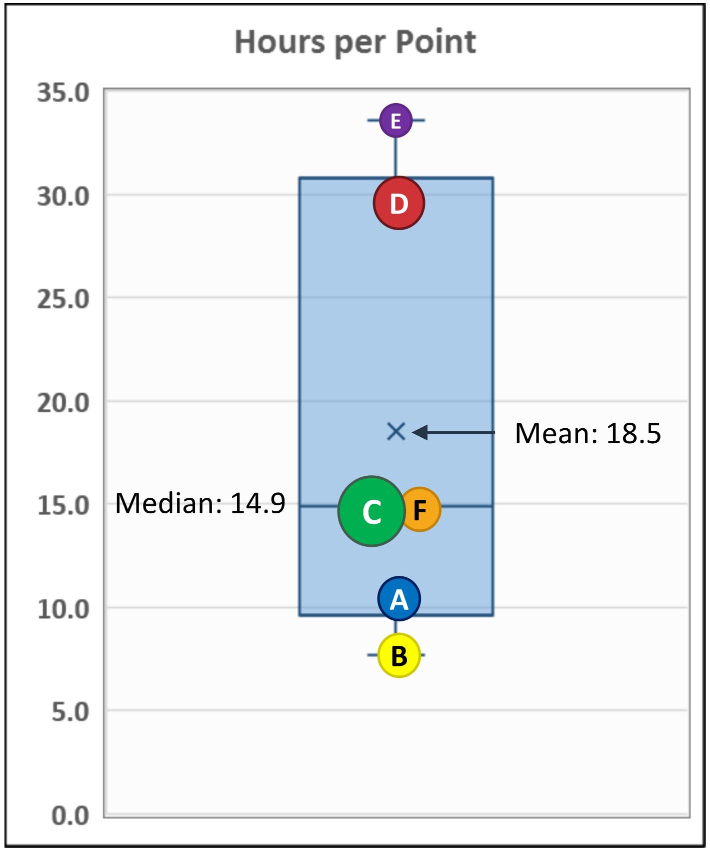
INTRODUCTION

METRICS DETAILS

SUMMARY

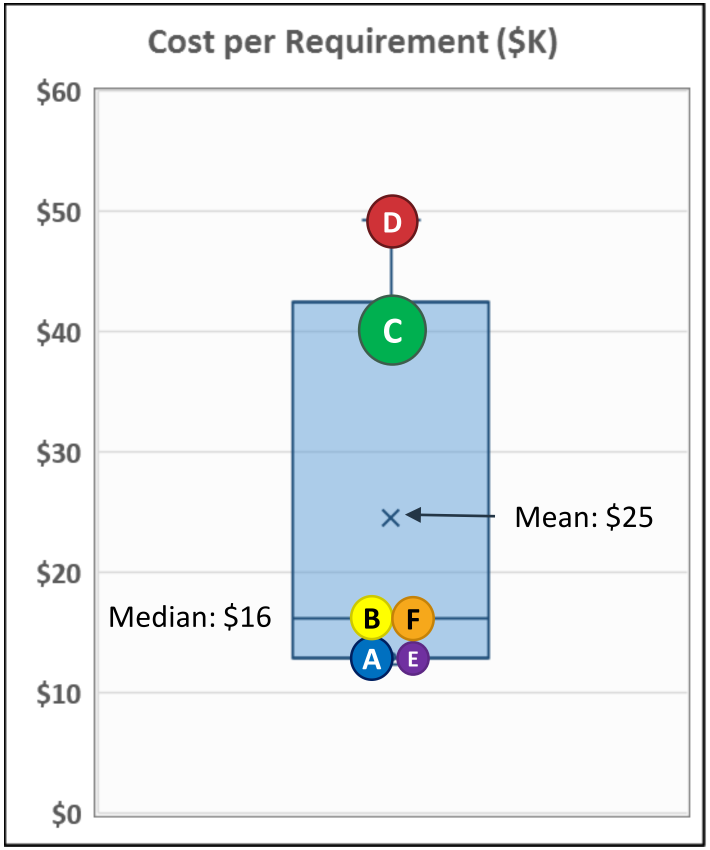
RESOURCES PER POINT

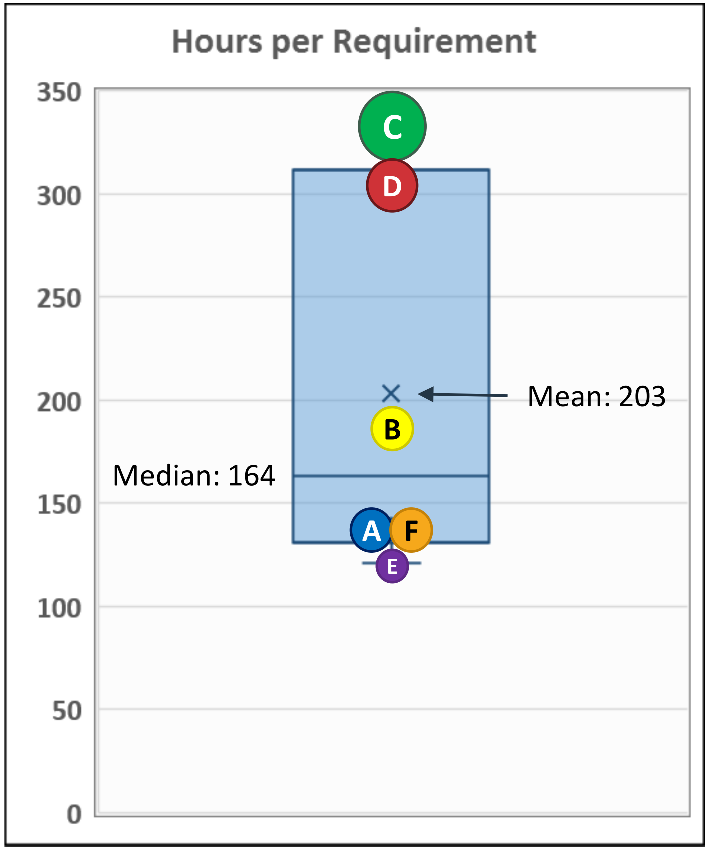




- Most projects defined a planned point as 8 developer hours
- Actuals indicate more cost/effort per point due to overhead and points taking more effort than expected to finish

RESOURCES PER REQUIREMENT





- Smaller projects (<$10M) tended to spend less resources per requirement – their requirements were generally less complex and defined at a more granular level

VARIANCES AT PROJECT END

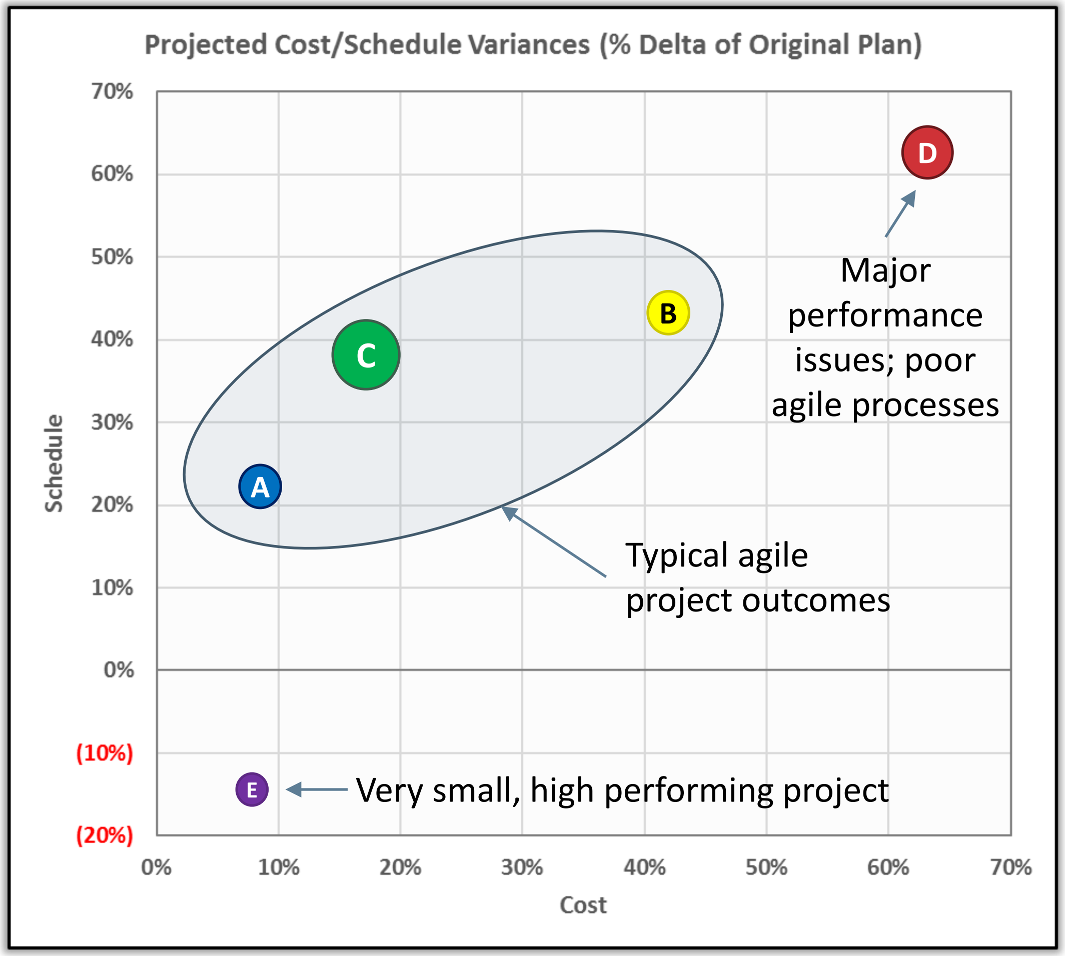
Project	Cost (% Delta – Plan minus Actual)	Schedule (% Delta – Plan minus Actual)	Scope (% of Planned Scope Not Completed)
Project A Ⓐ	0%	(11%)	8%
Project B Ⓑ	1%	0%	30%
Project C Ⓒ	27%	0%	41%
Project D Ⓓ	(2%)	(17%)	32%
Project E Ⓔ	1%	21%	2%

Negative numbers indicate cost/schedule overruns

All projects deferred at least some scope to later releases

- Project D had major performance issues, while Project E had atypically good performance; Projects A-C were fairly typical
- Most agile projects treat scope as variable with mostly fixed cost and schedule
  - Most finish at planned cost and schedule, but defer some scope to future releases, likely impacting future cost/schedule

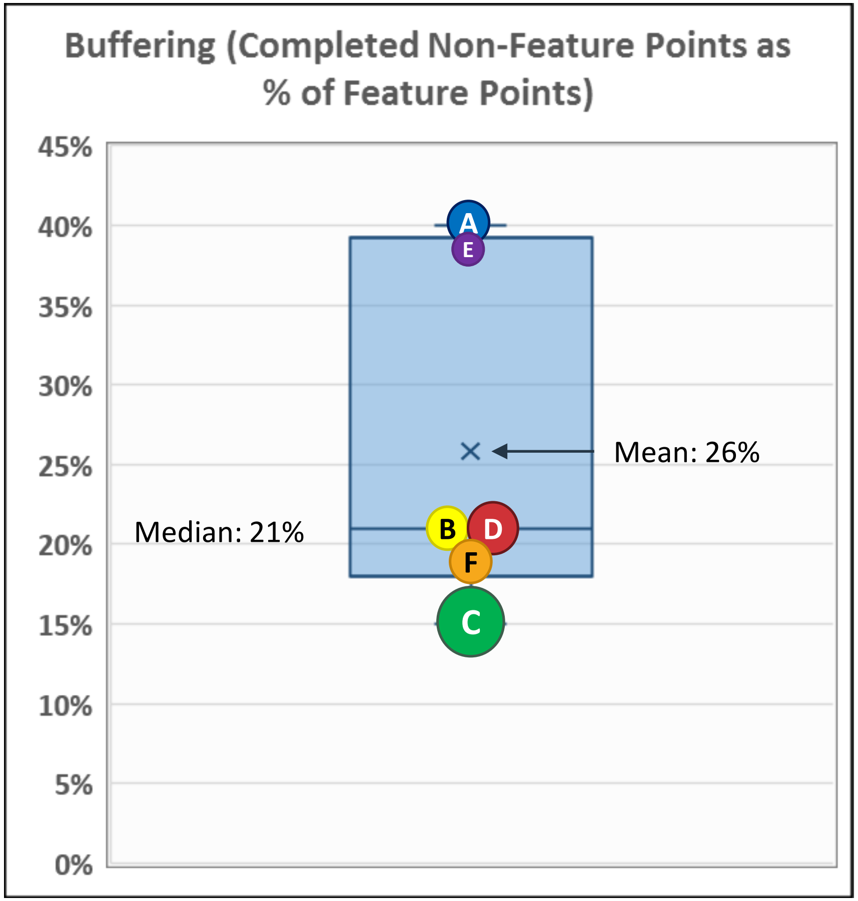
PROJECTED VARIANCES TO COMPLETE ALL REQUIREMENTS



- Projected to-complete cost/schedule overruns of 20-40% seem to be typical for agile projects
- Projected overruns caused by a combination of performance issues and prioritizing scope from agile activities
  - In-process testing
  - User evaluations
  - Requirements discovery

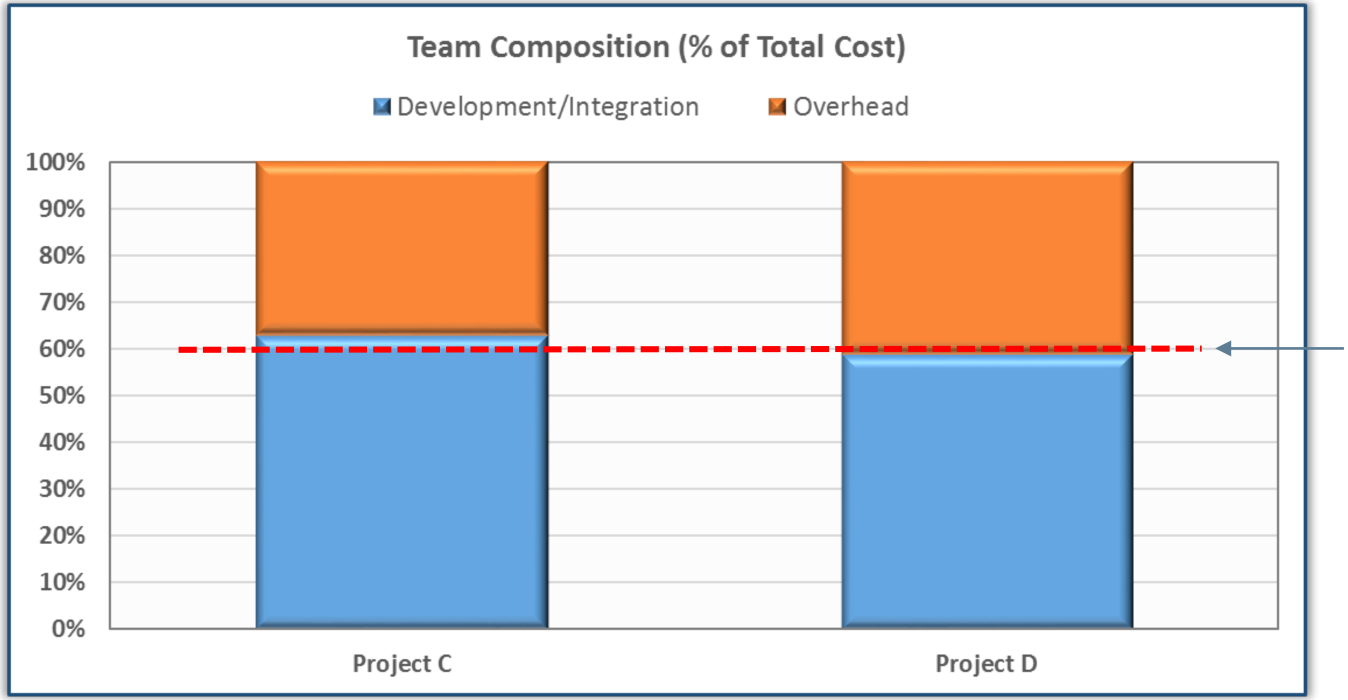
BUFFERING

- % of completed non-feature scope (bugs, usability, etc.) of completed feature scope (defined by functional requirements)



- Expected part of agile software development process, and should be included in estimates
  - Major component of expected cost/schedule overruns
- All projects in this case study between 15% and 40%
- Good rule of thumb: 20-30%

TEAM COMPOSITION



Average: ~60/40 split between development/ integration and overhead (PM, business support, functional SMEs, etc.)

- Two larger projects collected data needed for this metric
- Qualitative observation on other, smaller projects: they had more developers as % of total (less overhead), likely ~70-80%

AGENDA

INTRODUCTION

METRICS DETAILS

SUMMARY

TAKEAWAYS

- Monetizing points or requirement counts is difficult and entails large uncertainty ranges
- For most agile projects, scope is the variable
  - Most finish at planned cost and schedule, but defer some scope to future releases, possibly impacting future cost/schedule
  - Without scope deferral, our "normal" case study data points projected cost/schedule overruns at ~20-40% of original plan
- Good rule of thumb for buffering: Add 20-30% to requirements/feature-driven estimates for bugs, etc.
- Rules of thumb for team composition:
  - Project Cost > $10M: 60% development/integration; 40% overhead
  - Project Cost < $10M: 75% development/integration; 25% overhead

NEXT STEPS

- Further analyze existing data for other useful metrics
  - Impacts of team size changes on productivity
  - Correlation between cost/schedule/scope variances
  - Metric correlation to high-level project aspects (size, performer, etc.)
  - EVM-like metrics
- Collect/organize additional data points

SUMMARY

- Agile projects can be planned and measured
- Data analysis can yield useful metrics for cost estimating
- As usual, more data collection and analysis is needed

THANK YOU

For more information, contact . . .

Blaze Smallwood
Lead Associate

Booz | Allen | Hamilton

Booz Allen Hamilton Inc.
Office 309.359.3160
Mobile 619.850.6123
smallwood_blaze@bah.com