

Presented at the 2018 ICEAA Professional Development & Training Workshop - [www.iceaaonline.com](http://www.iceaaonline.com)



Agile Software Development Cost Risk  
for  
Information Technology Programs

# Today's Presenter

Presented at the 2018 ICEAA Professional Development & Training Workshop - [www.iceaaonline.com](http://www.iceaaonline.com)



**John McCrillis**

Senior Cost Estimator

John McCrillis has been working hardware and software cost estimating for 18 years as an operations research analyst at OSD, USN, ODNI, and most recently Technomics. He has 19 years' experience building flight simulators where he wrote software and led development teams. John has a BS in Aerospace Engineering and a MS in Mechanical Engineering. He has a patent application for Distributed I/O and Power and has written papers on cost estimating, simulators, and aircraft flight testing.

# Background on Agile Development

Presented at the 2018 ICEAA Professional Development & Training Workshop - [www.iceaaonline.com](http://www.iceaaonline.com)

- Agile software development is premised on evolving requirements
- Pushing product out the door as quickly as practical
- Getting user feedback early and continuously
- Updating requirements based on feedback
- Small increments frequently
- Small autonomous teams producing product
- Responding to unknowns quickly
- Unclear what the final product will look like
- Failing quickly and cheaply
- Agile is the antithesis of waterfall (maybe a little exaggerated)
- So if it's not known "what" it is, how can it be estimated and where's the risk?

How do I know when I'm done in an Agile development?

# Governing Equations

Presented at the 2018 ICEAA Professional Development & Training Workshop - [www.iceaaonline.com](http://www.iceaaonline.com)



- There are only two methods for costing software development -- capability or staffing

$$\text{Cost} = \text{Size} \times \text{Productivity}$$

or

$$\text{Cost} = \# \text{ Staff} \times \text{Duration}$$

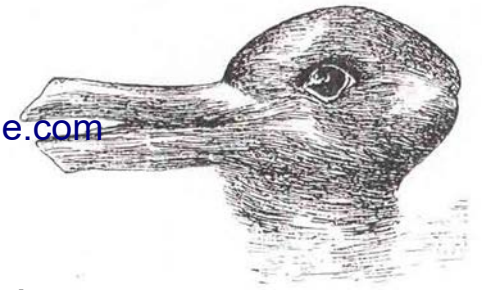
- These methods are appropriate regardless of software development process
- Risk is relevant to the capability method; not staffing method
  - There is risk of not delivering the intended capability
  - There is little-to-no risk in delivering staff for a defined duration

Productivity units: \$/size

The risk of not getting what was intended only occurs when costing capability

# Cost Estimators Paradigm

Presented at the 2018 ICEAA Professional Development & Training Workshop - [www.iceaaonline.com](http://www.iceaaonline.com)



- $DCT\ Cost(\$) = \text{Size of Effort (Function Point, ESLOC, RICE, \dots)} \times \text{productivity(hrs/FP, hrs/ESLOC, \dots)} \times \text{Labor rate}(\$/\text{hr}) \times \text{Growth factor}(\%)$ 
  - Function Point; derived from analysis of the design specifications using ISO standards
  - $ESLOC = \text{new} + \text{factor} \times \text{reuse} f(\text{mod, unmod}) + \text{factor} \times \text{auto-generated} + \text{factor} \times \text{prior build} f(\text{mod, unmod})$
  - Growth factor based on acquisition phase and maturity of program
- Add-ons include: system integration, SE/PM, SSA, facilities, IOT&E, SIL, ...
- Validate/compare the labor rate, size, and productivity with program historical and/or analogous programs, i.e. histograms, averages, median, ...
  - Data sources in order of precedent; program historical, development team historical, functionally similar programs, language-similar programs, other
- Identify risk ranges on each parameter
  - Identify Probability Density Function (PDF) and correlation from data
  - Calculate Confidence Level (CL) using Monte Carlo methods
  - Produce cost S-curve

DCT; Design, Code & Test (unit test)  
ESLOC; Equivalent Source Lines of Code  
HW; Hardware  
SW; Software  
SEPM; Systems Engineering & Program Management  
SSA; Software Support Activity  
SIL; Software Integration Laboratory  
IOT&E – Independent Operational Test & Evaluation

Risk is assessed in PDFs for a defined capability

# Agile Software Estimating Cost Challenge

Presented at the 2018 ICEAA Professional Development & Training Workshop - [www.iceaaonline.com](http://www.iceaaonline.com)

<u>Agile</u>		<u>Traditional</u>
Time	Versus	←Time→
Cost		←Cost→
←Scope→		Scope

Three legs of the "Iron Triangle"

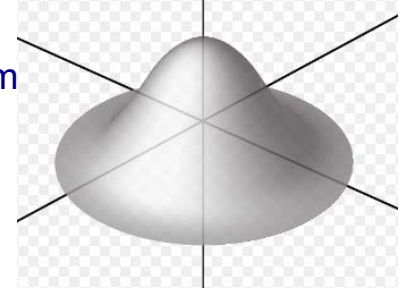


- Agile is different from traditional software development processes (spiral, waterfall, ...); it has fixed time & cost and variable scope increments
  - But what about capability; do users just get whatever is delivered?
  - When will the user say the program is complete?
  - If the time and cost are extended past the fixed period, is that cost growth?
  - Does this mean Agile should only be contracted via Time & Material (T&M) arrangements and estimated using staffing methods?
  - Variable scope means Agile cannot be contracted fixed price arrangements

What equations do I use to estimate cost for Agile development

# Function Points (FP)

Presented at the 2018 ICEAA Professional Development & Training Workshop - [www.iceaaonline.com](http://www.iceaaonline.com)



- Three types of requirements: technical, quality, functional
  - FP only apply to functional requirements
  - Functional requirements are representative the development effort
- FP are a "unit of measure" to express the amount of business functionality an information system (as a product) provides a user
- The cost (in dollars or hours) of a single FP is calculated from past projects
- Applicable to IT systems, but
  - Doesn't work well with embedded systems (does not cross boundaries)
  - Does not count "calculations" well; scientific applications are underestimated
- FP can be counted from program documentation like MNS, ORD, TEMP, CONOPS, XML prototypes, and other

Function points can be used to size Agile and other software development program scope

# FP Overview

Presented at the 2018 ICEAA Professional Development & Training Workshop - [www.iceaaonline.com](http://www.iceaaonline.com)

- A standard unit of measure that quantifies the size and complexity of a software application in terms of Elementary Processes (EP) a user performs

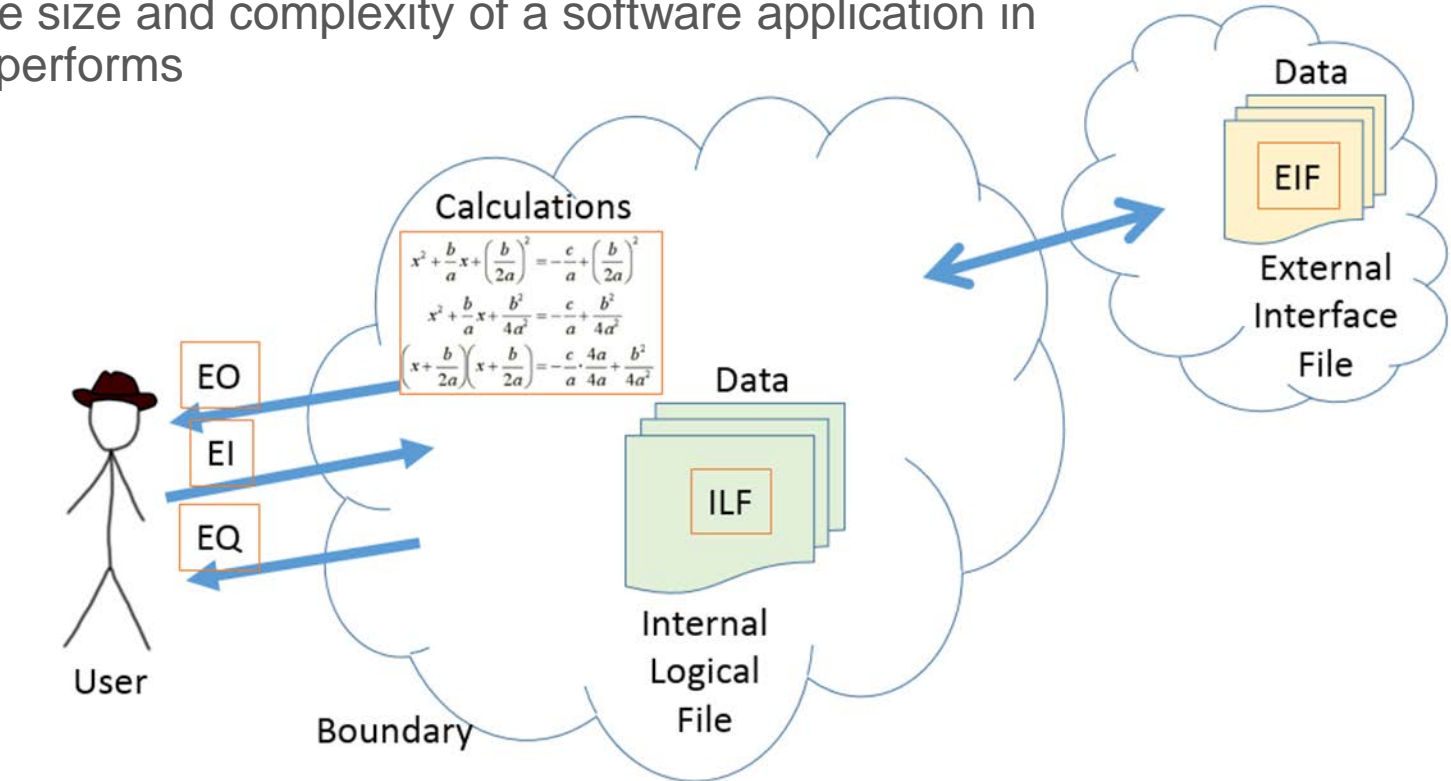
- For example an EP with:

- External Inputs (EI)
- External Outputs (EO)
- External Inquiries (EQ)
- External Interface Files (EIF)
- Internal Logical Files (ILF)
- Complexity for each based on count of EI/O/Q and E/ILFs
- Sum complexity X EI, EO, EQ, EIF, ILF

- 1 FP ~ 55 LOC & ~ 13 hrs development

- Alternate method

- Verbs (action) = EP
- Nouns (objects) = files



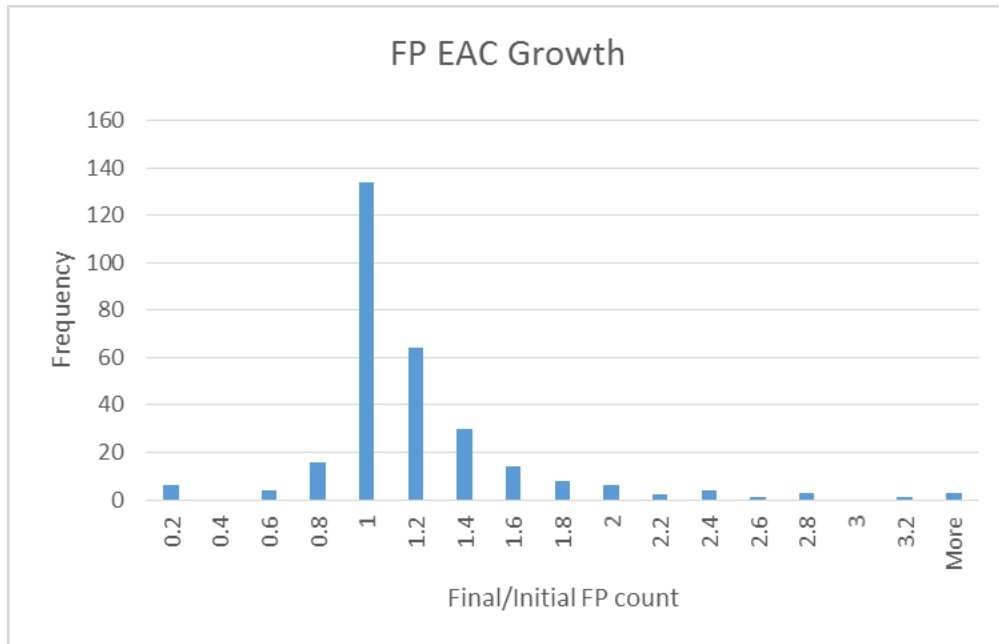
- Capability is broken into elementary process
- Count transactional functions that cross the application “boundary”

Counting FP “by the book” requires more detail than typically available in early stages of dev



# Size Growth (Function Point Count)

Presented at the 2018 ICEAA Professional Development & Training Workshop - [www.iceaaonline.com](http://www.iceaaonline.com)



EAC (Estimate at Completion)

	<i>Final/ Initial FP</i>
Mean	1.14
Standard Error	0.03
Median	1.00
Mode	1.00
Standard Deviation	0.48
Sample Variance	0.23
CV	42%
Skewness	3.05
Range	4.89
Minimum	0.01
Maximum	4.90
Count	296.00

- ISBSG (International Software Benchmarking Standards Group) data set
  - Initial estimate vs final actual
  - Mostly small projects < \$500k
  - Expect higher success rate in small projects
  - All counting methods

- The mean indicates 14% growth in FP count from the initial to the final
  - Historical SLOC growth is ~40% (DoD data)
  - Median indicates no growth

Histogram is basis of FP PDF in risk assessment

# Productivity

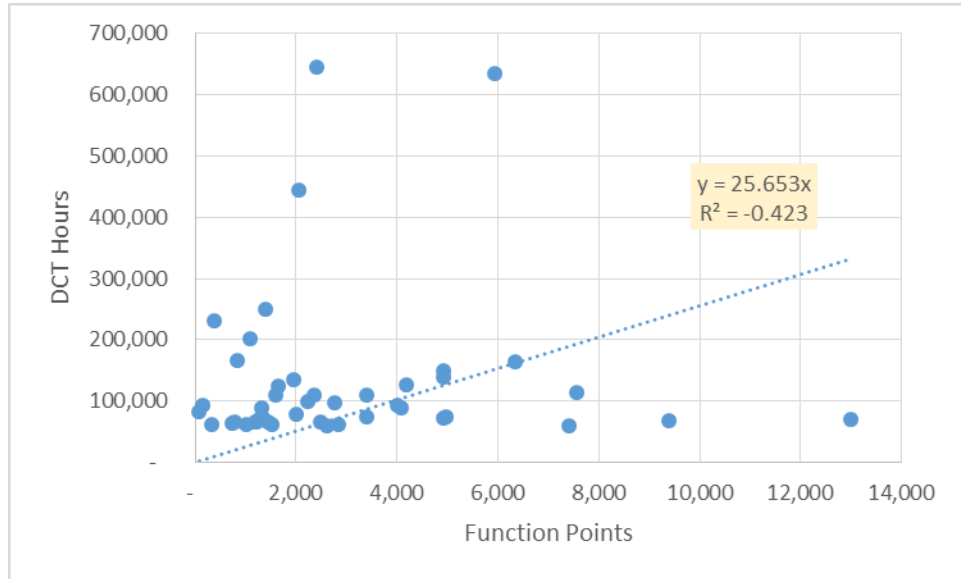
Presented at the 2018 ICEAA Professional Development & Training Workshop - [www.iceaaonline.com](http://www.iceaaonline.com)

- To estimate cost, both size and productivity are required
  - Productivity is dependent on the teams assembled
  - Always an issue for all software estimates, not just Agile estimates
- Data priorities: organization, program, commercial data sets
  - An organizations historical productivity for different commodities is best available data
  - Program data improves as teams mature and recedes during integration and test
  - Use commercial benchmarks if organization data not available initially
    - Replace with program data as it becomes available
- Establishing organization benchmarks using completed program actuals
  - Contracts data; dates, funding, ...
  - FP count based on initial docs and final actual
  - Program data; commodity, customer, dates, ...
  - System description; docs, SMEs, sketch
  - FP count documentation

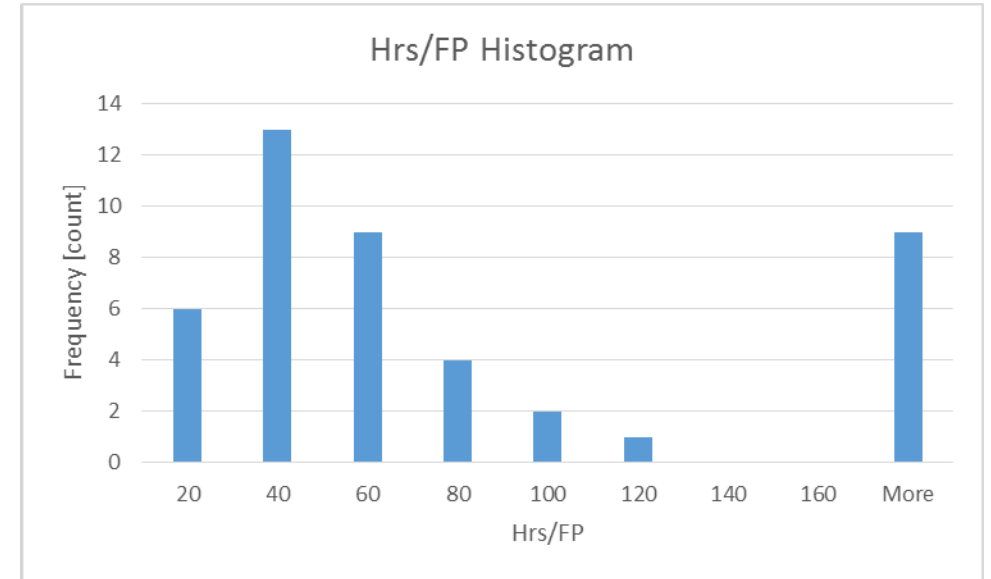
Organization benchmarks, not program productivity should be used whenever possible

# Commercial Productivity Benchmark for Large Programs

Presented at the 2018 ICEAA Professional Development & Training Workshop - [www.iceaaonline.com](http://www.iceaaonline.com)



Programs great than 30 FTEs	
Mean	112.1
Standard Error	29.7
Median	46.0
Mode	#N/A
Standard Deviation	196.8
CV	176%
Kurtosis	15.3
Skewness	3.7
Range	1,092.0
Minimum	5.4
Maximum	1,097.4
Count	44

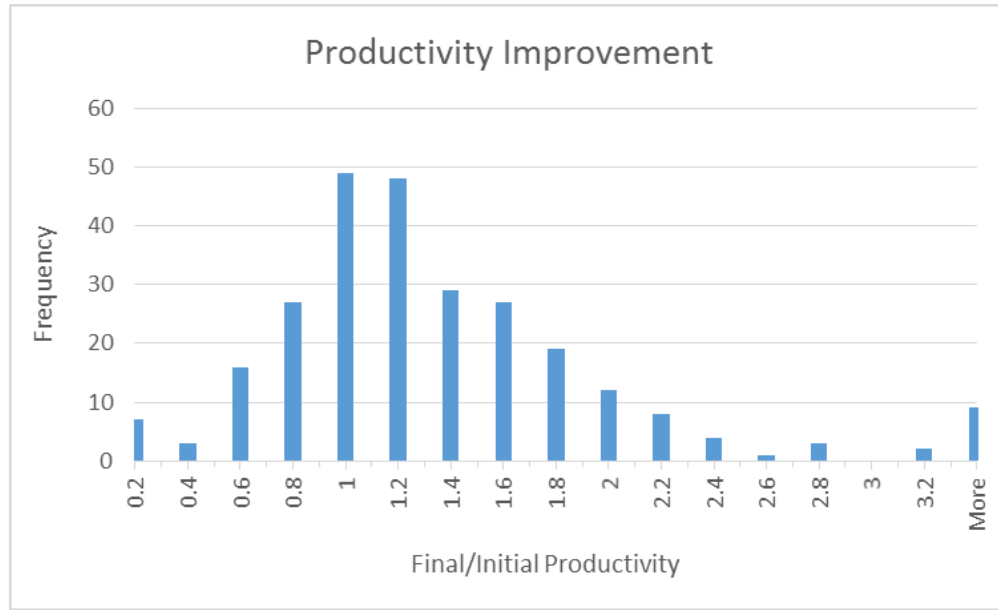


- Hours represents design, code, and test
- Significant variance; 9 programs with productivities greater than 140hrs/FP is skewing average to right
  - Median (46 hrs/FP) appears more representative of data than average (112 hrs/FP)
  - Regression, @ 26 hrs/FP (low  $R^2$ , high CV)
- Analysis did not identify any relationship with other parameters, i.e., dev language, measurement method, date, size, ...

This data indicates FP productivity data is insensitive to software development process

# Productivity Growth

Presented at the 2018 ICEAA Professional Development & Training Workshop - [www.iceaaonline.com](http://www.iceaaonline.com)



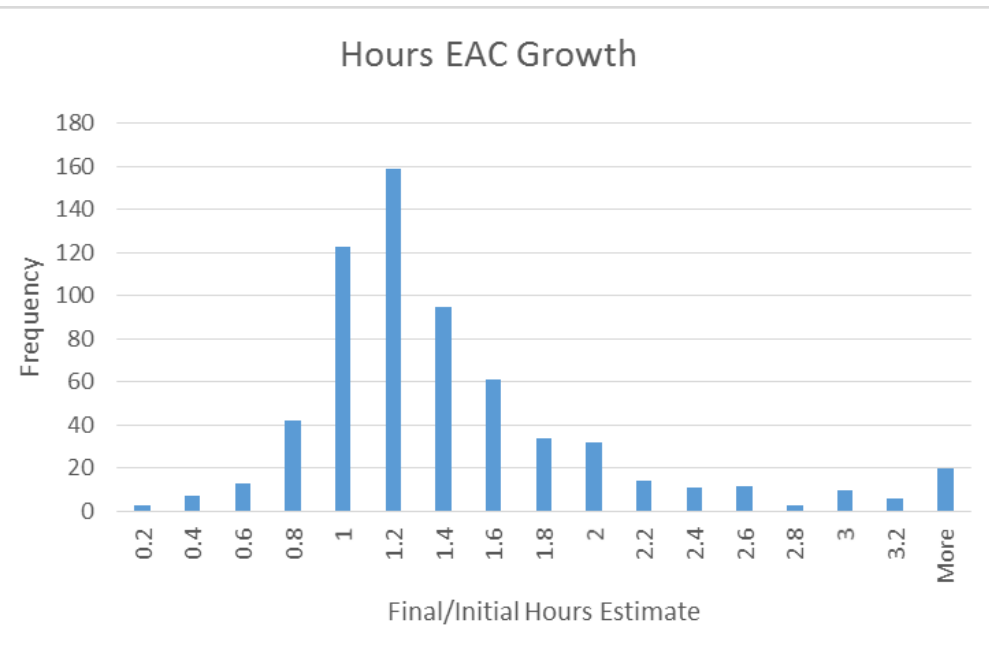
	<i>Final/ Initial productivity</i>
Mean	1.34
Standard Error	0.07
Median	1.11
Mode	1.00
Standard Deviation	1.09
Sample Variance	1.18
CV	81%
Skewness	4.91
Range	10.90
Minimum	0.04
Maximum	10.95
Count	264

- Data indicates productivity isn't as high as initially anticipated
  - Reflected in increased hours (see next slide)

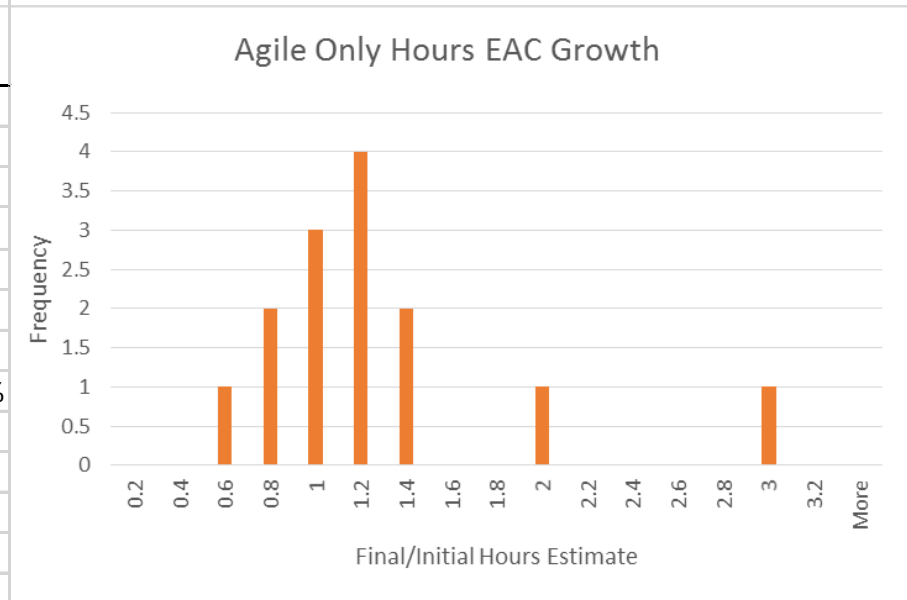
Histogram is basis for productivity PDF in risk assessment

# Effort (Hours) Growth Crosscheck

Presented at the 2018 ICEAA Professional Development & Training Workshop - [www.iceaaonline.com](http://www.iceaaonline.com)



	<i>Final/ Initial hrs</i>	<i>Agile only final/ initial hrs</i>
Mean	1.50	1.16
Standard Error	0.06	0.16
Median	1.16	1.06
Mode	1.00	#N/A
Standard Deviation	1.59	0.60
Sample Variance	2.54	0.35
CV	107%	51%
Skewness	7.60	2.16
Range	18.97	2.39
Minimum	0.03	0.52
Maximum	19.00	2.91
Count	645	14

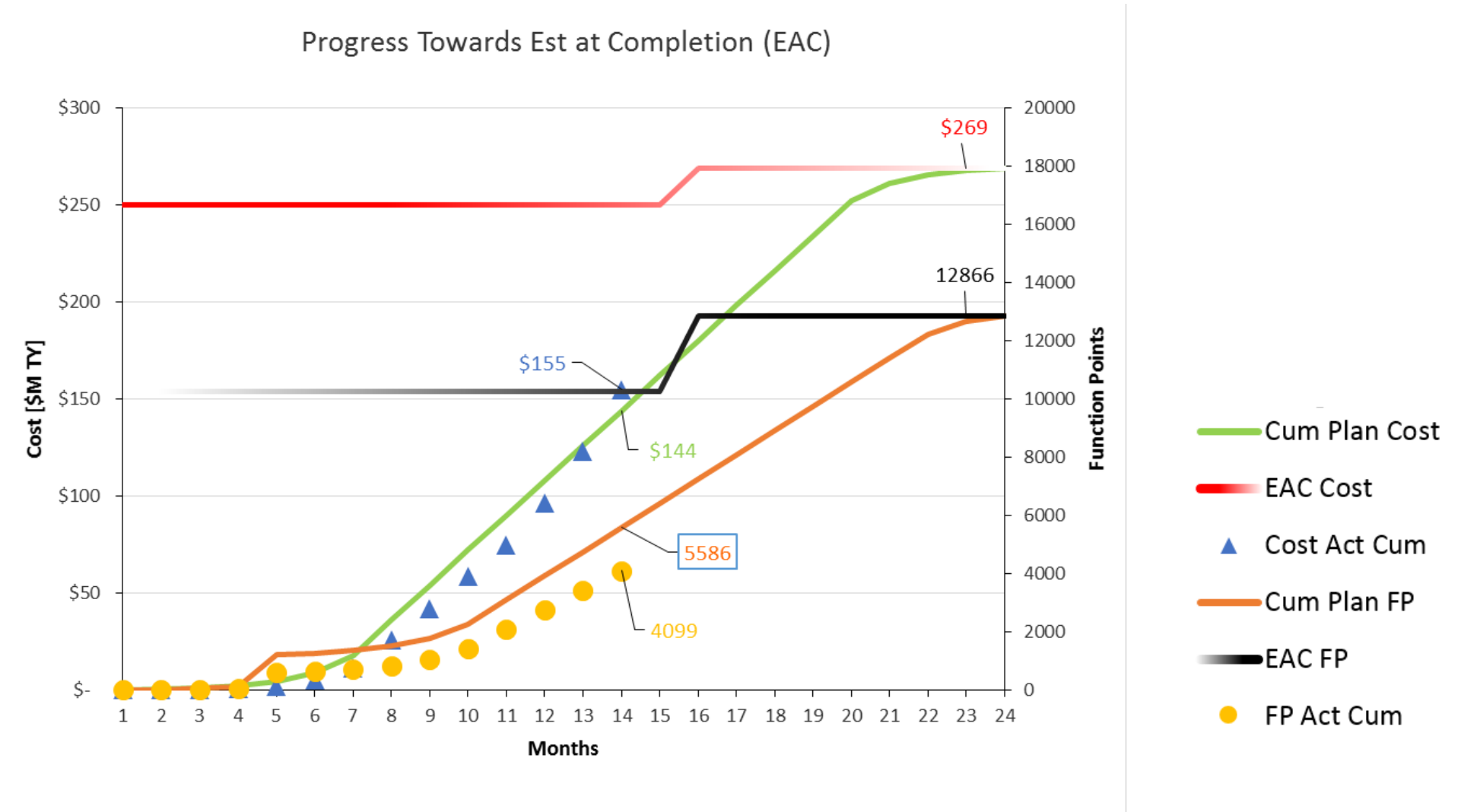


- Hours growth is significant, 50% mean
  - Histogram appears to be normal distribution around 20%
  - CV of 107% indicates the mean is not representative the data set
  - Median, 16% growth, appears most representative the data set
  - Consistent with productivity change and static FP growth data
- Agile only programs have insufficient number of data points to draw conclusions but histogram indicates similar growth

Initial estimates tend to be optimistic by the error of the productivity estimate

# EAC Facilitates Progress Assessment

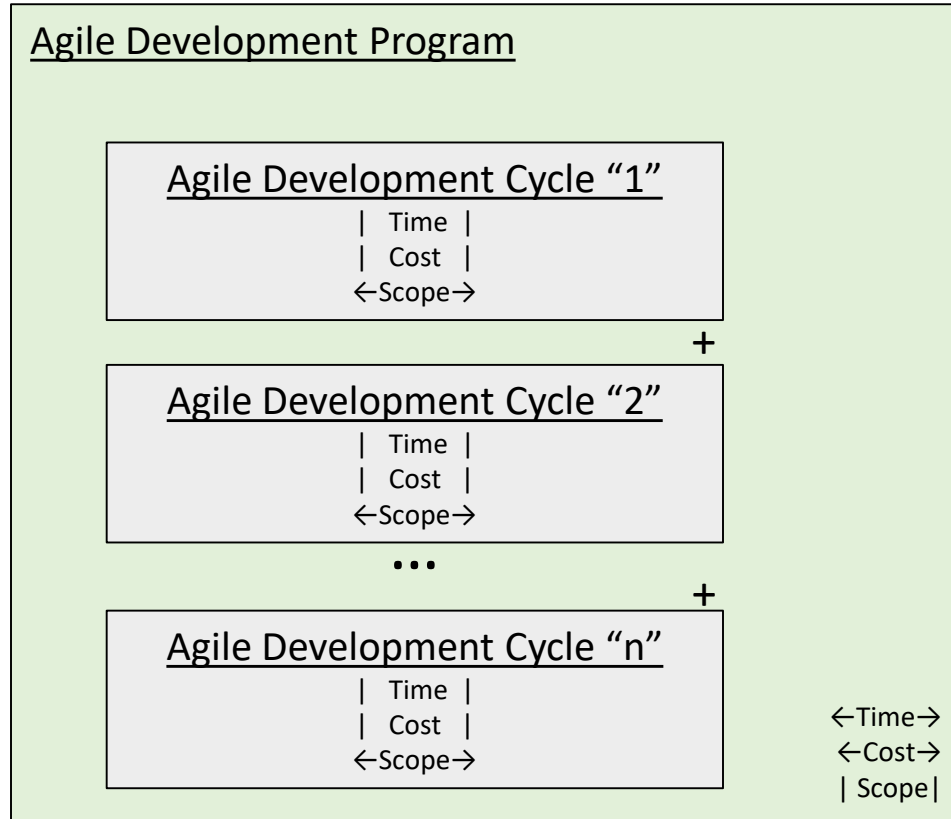
Presented at the 2018 ICEAA Professional Development & Training Workshop - [www.iceaaonline.com](http://www.iceaaonline.com)



Estimate at Completion (EAC) is an excellent measure of progress when applied to cost and size

# Where's the Risk?

Presented at the 2018 ICEAA Professional Development & Training Workshop - [www.iceaaonline.com](http://www.iceaaonline.com)



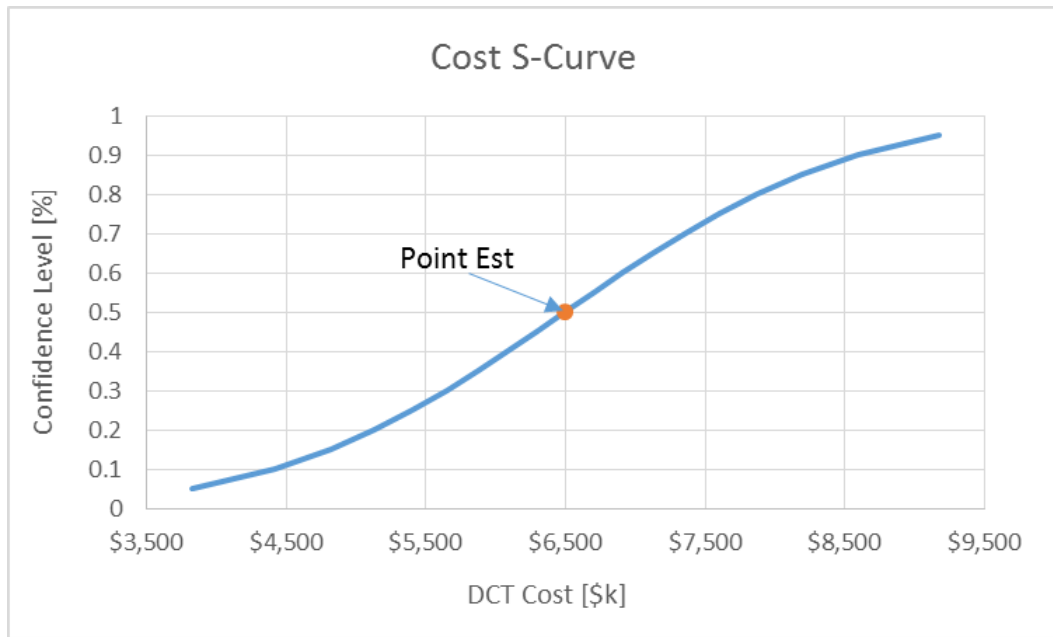
- Setting program scope is a discipline Agile programs support
- CONOPS, MNS, ORD, TEMP and other docs are being provided that describe program scope
  - Supported with interviews
  - The Program should agree with the count
- Program scope is being defined in functional terms
- Some Agile programs are reporting EVMS by increments which implies fixed scope.
- If program scope can be “sized” than traditional cost estimating and risk can be applied

The risk in Agile programs is the number of development cycles necessary to achieve the intended scope

# Assessing Risk With Cost S-Curve

Presented at the 2018 ICEAA Professional Development & Training Workshop - [www.iceaaonline.com](http://www.iceaaonline.com)

Example	
50% CL:	
FP [ct]	1,000
hrs/FP	50
\$/hr	125
\$/FP	\$ 6,250
Cost	\$6,250,000



- 1) Productivity  $f(\text{CL}) = \text{Cost } f(\text{CL}) / \text{Size}(\text{PE})$
- 2) Size  $f(\text{CL}) = \text{Cost}(\text{PE}) / \text{Productivity } f(\text{CL})$

Program productivity = all dev cost/size

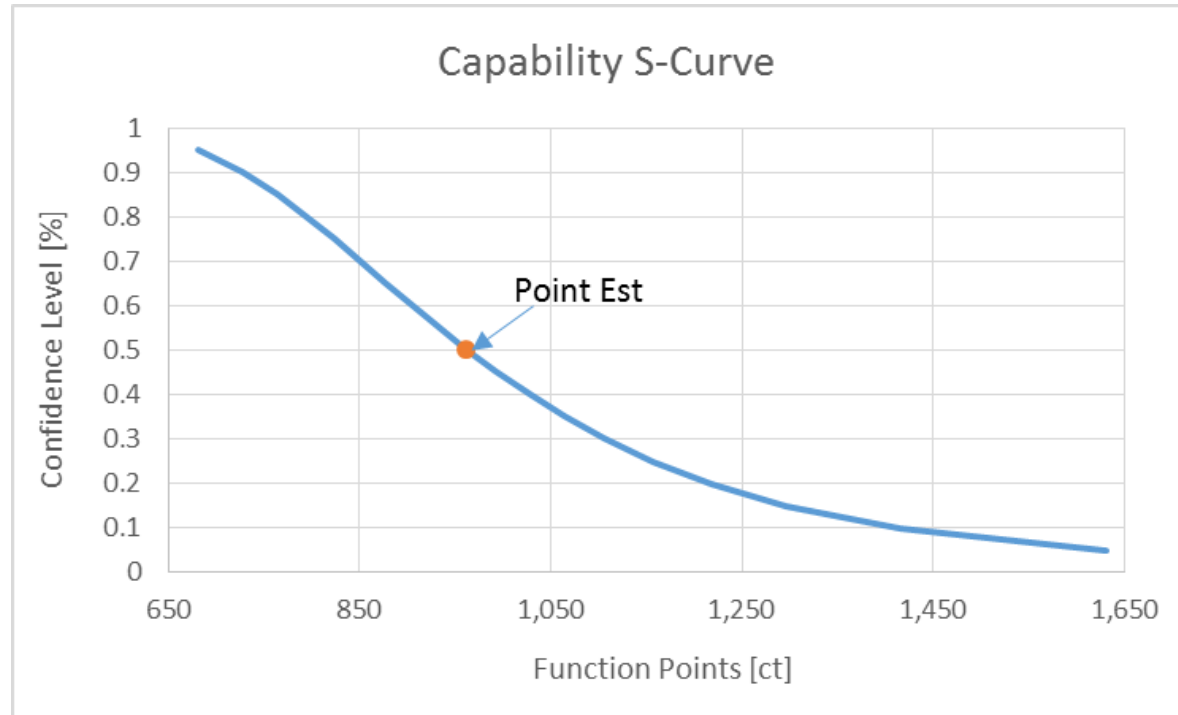
- With EAC cost known, the risk is getting less capability than intended for the cost intended
  - Same Monte Carlo S-curve process as classic analysis
  - Generate PDFs based on histogram data like previously shown
- Convert cost S-curve to capability S-curve
  - Using eq “1” to convert Cost  $f(\text{Confidence Level (CL)})$  to Productivity  $f(\text{CL})$  at the Point Estimate (PE) size
    - Program productivity is a notional concept (includes all cost) and not to be confused with DCT productivity
  - Using eq “2” convert program productivity  $f(\text{CL})$  by dividing it into the Cost(PE)
    - This is the concept of varying scope for a fixed price, i.e. the point estimate

Agile development risk is similar other software development process; not getting what was intended for the agreed cost and schedule



# Capability S-Curve

Presented at the 2018 ICEAA Professional Development & Training Workshop - [www.iceaaonline.com](http://www.iceaaonline.com)



This is the same data as the cost S-curve

- Confidence level decreases with increasing FP count for the current point estimate at the current productivity
- Tailoring the message to an Agile audience helps communicate the risk

Invert a classic S-curve to estimate and communicate the risk of achieving a desired scope

# Conclusions

Presented at the 2018 ICEAA Professional Development & Training Workshop - [www.iceaaonline.com](http://www.iceaaonline.com)

- Agile programs can be estimated and/or assessed like any other IT program as long as size EAC is established
  - Program is T&M in the absence of defined scope (i.e., size); as such, there is no risk because there is no capability commitment
- Functionality can be 'measured' using function points as the sizing metric
  - Existing program documents like CONOPS detailing different scenarios can be measured
  - Various counting methods can be used
  - The better the detail, the better the estimate
  - The variance in size decreases as the program matures
- Productivity metric data is required to estimate program cost
  - Commercial data readily available
  - Use program actual productivity after a few delivered increments
  - Use organizational data once process is established
- Iterative nature of Agile lends itself to an iterative cost/risk estimating process
  - Continual updates to a FP EAC facilitates discussion of THE primary issue – evolving requirements
  - The cone of uncertainty narrows as the program matures
- A traditional cost S-curve converted to a capability S-curve is an extremely effective approach to estimate and communicate the risk of achieving desired scope