Army Software Sustainment: Righting Our Cost Estimating Assumptions

Cheryl Jones

Presented to ICEAA Workshop

2021

Topics

- Purpose of this presentation
- Initiative details
- Data demographics
- Time phased analysis results
- Conclusions

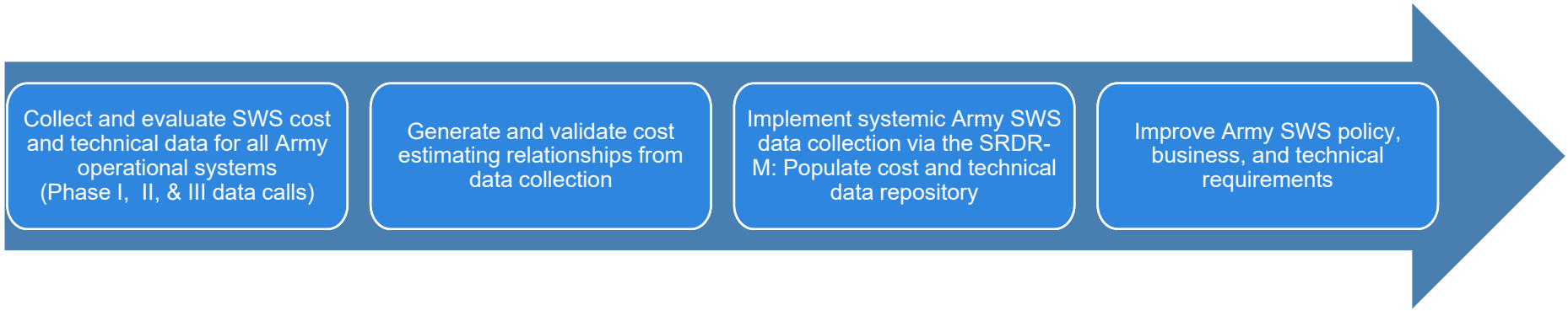
Purpose

- A challenge in estimating software Total Ownership Costs at the beginning of a software acquisition is estimating the cost of software after development.
- Rules-of-thumb are often used to estimate the annual change in the software during the sustainment lifecycle.
- This presentation will present data analysis on software sustainment profiles that show a rise and fall in the number of changes over long periods of time *contrary* to the practice of using a constant annual change

SWS Initiative Objective and Strategy

Accurately estimate Army system Software Sustainment (SWS) costs to:

- Effectively estimate and justify software and system life cycle costs
- Objectively evaluate Army system software sustainment execution costs
- Inform and optimize the allocation of available sustainment resources across the Army



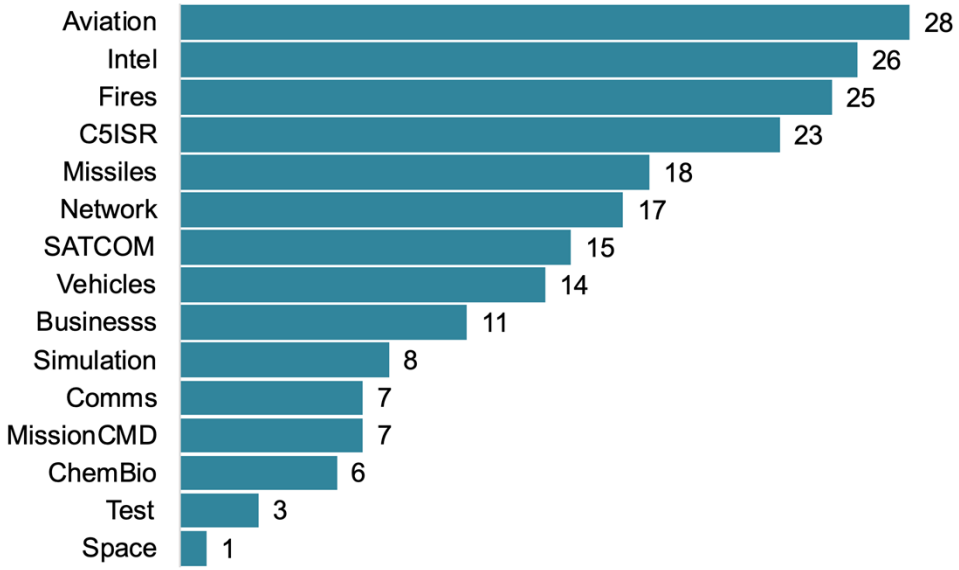
Effective software sustainment cost estimation is the basis for Army system software life cycle cost management

Summary of Accomplishments

- Established Software Sustainment Data Collection Mechanisms
  - Army Software Data Collection Questionnaire
  - Software Sustainment WBS Used to Collect Sustainment Costs
  - Annual Data Collection
- Created Comprehensive Software Sustainment Data Repository
  - 214 Unique Systems (including DBS systems)
  - 855 Capability Releases
  - 676 IAVA Releases
  - 3,840 records on software license data
- Established Robust Foundation for Software Sustainment Fact-Based Decisions
  - Allocations of Costs by WBS Elements
  - Continue to improve Software Sustainment Cost Estimating Relationships
- Data and Analysis Results provided to DoD Community

System Demographics

Number of Systems by Commodity



208
Unique Systems

41
Phase 1 Systems

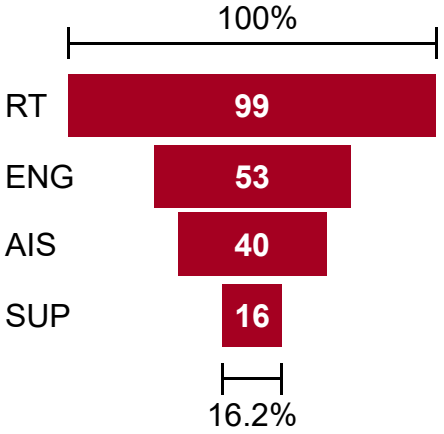
151
Phase 2 Systems

115
Phase 3 Systems

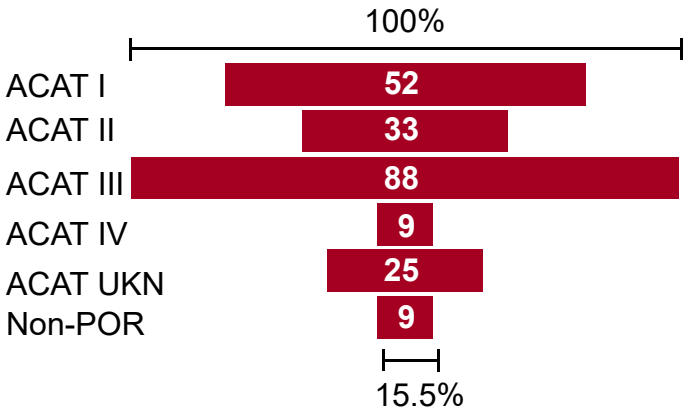
2,107
Data Fields

22 New Systems in Phase 3
94 systems in the previous database that are not in Phase 3

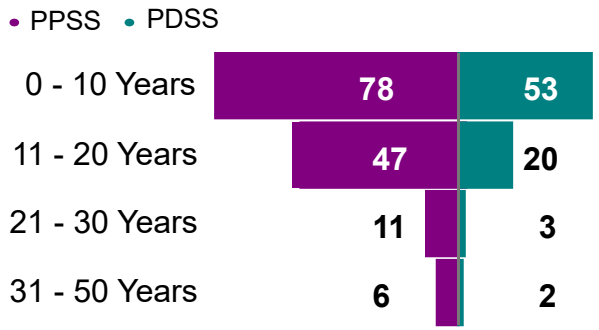
Super Domains



Systems by ACAT Level

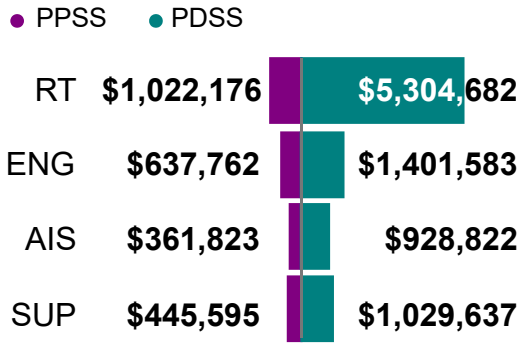


Distribution of System Age

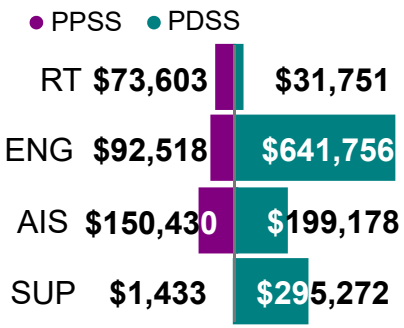


Software Releases

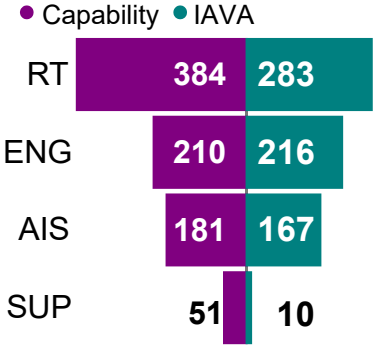
Average Capability Release Costs



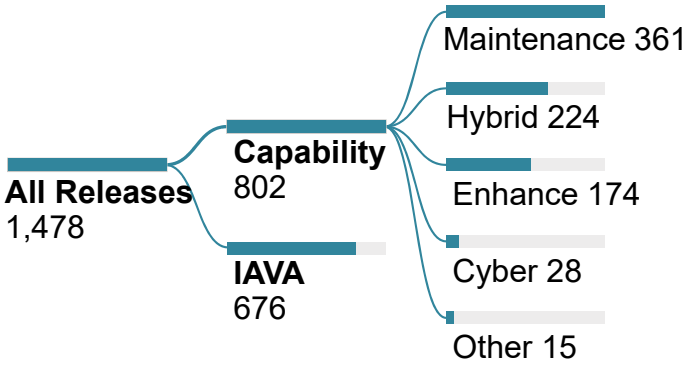
Average IAVA Release Costs



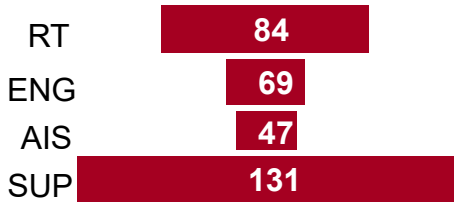
Releases by Super Domain



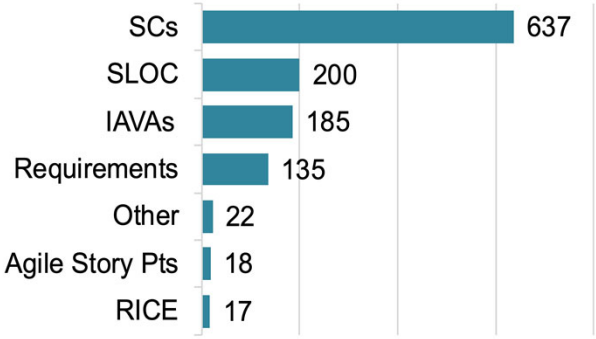
Change Type



Average Software Changes



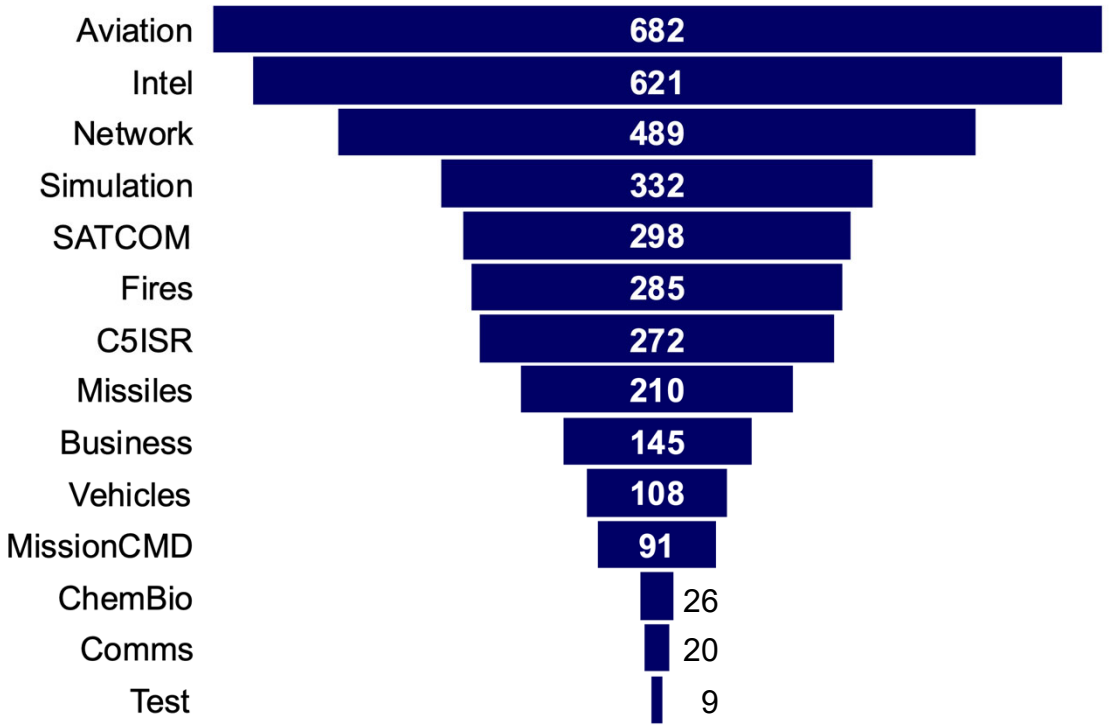
Releases by Size Measure



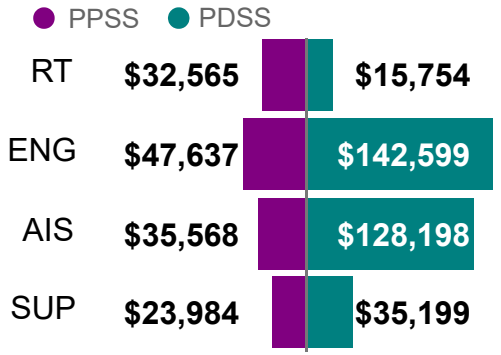
1/1

Software Licenses

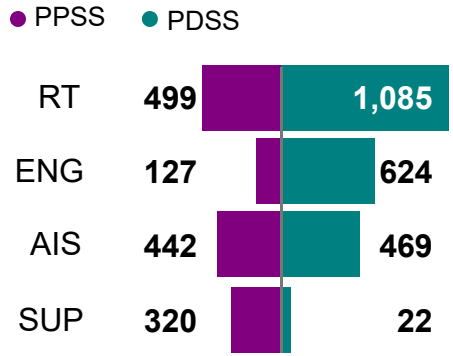
Licenses by Commodity



Average License Cost by Super Domain



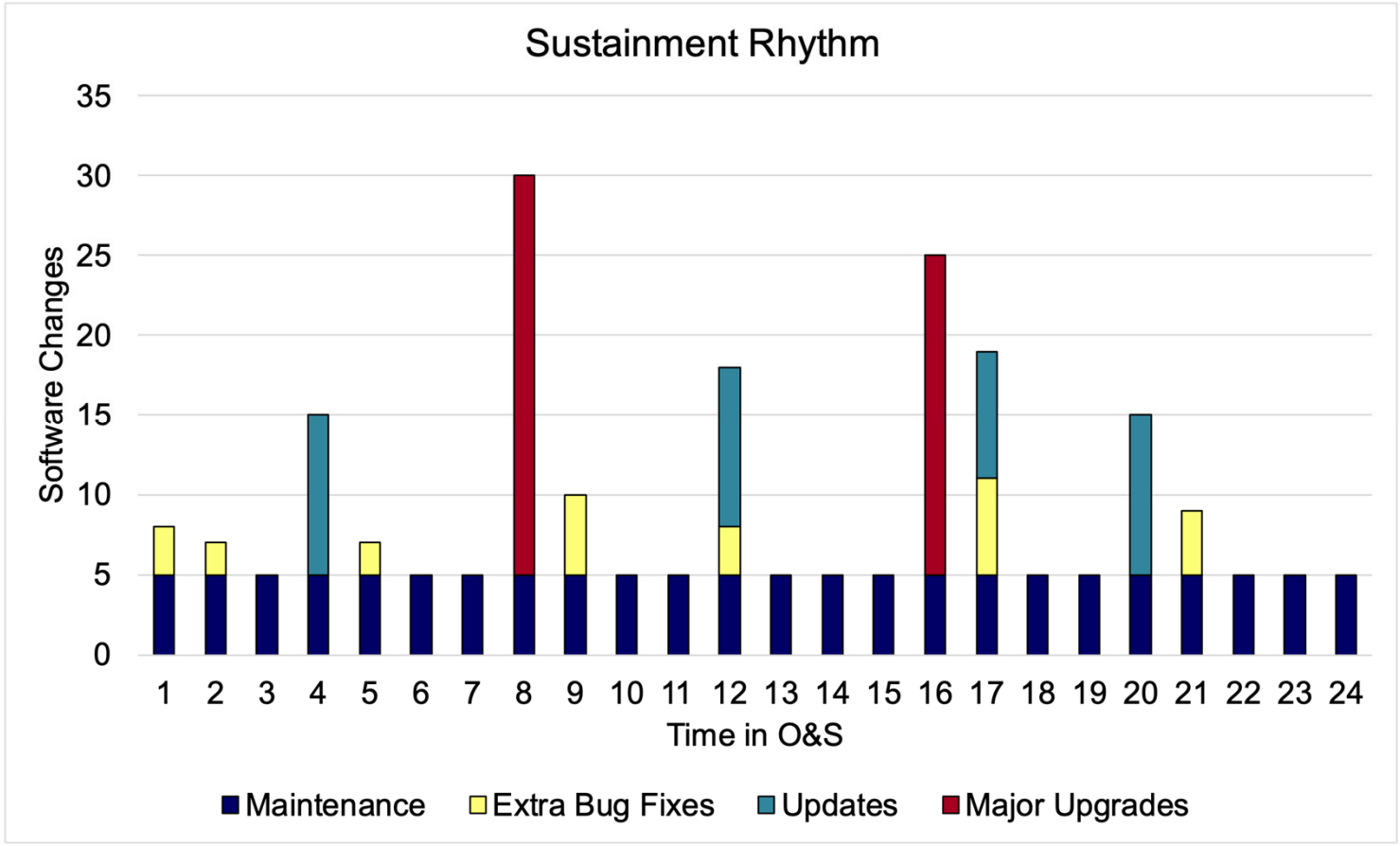
Number of Licenses by Super Domain



Previous Data Analysis

- Preparing data for analysis
  - Data collection
  - Data repository
  - Data normalization
  - Data imputation (backfill, online searches, averages)
  - Analysis
    - Outlier labeling rule
    - Histograms
    - Boxplots
    - Scatter plots
      - Loess smoothing
      - Non-linear Trend lines
    - Log Ordinary Least Squares regression
    - Non-linear regression
    - Regression trees
    - K-nearest neighbor
- Results
  - Cost distributions
  - Sensitivity analysis
  - Causal analysis
  - Cost Estimating Relationships (CER) using different segmenting strategies
- Current Analysis Focus
  - Changing release cost base over time in sustainment

Notional Software Sustainment Release Profile



Maintenance: normal repairs, continuously ongoing
Extra Bug Fixes: due to a major release
Updates: obsolescence, new functionality
Major Upgrades: architectural changes, new hardware, major functional changes

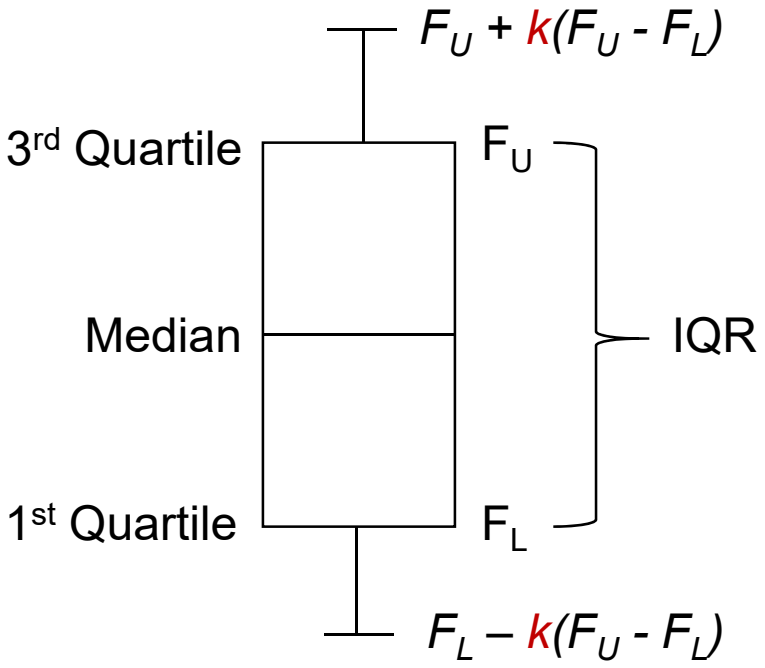
Analysis Data Selection

- Capability Release data records from data collection phases 1, 2, and 3.
- Only systems in the **O&S phases** were studied: 987 records
- Removed records with:
  - DBS: 16
  - No O&S phase start date: 39
  - No Release end data: 63
  - No Release Hours: 262
  - No SW Change (SC) counts: 452
- Records remaining: 345
- Time in the O&S Phase is defined as:

  2019– (Phase start date)
- Both All-data and Trimmed-data are presented
  - Data was trimmed using an *Outlier Labeling Rule*

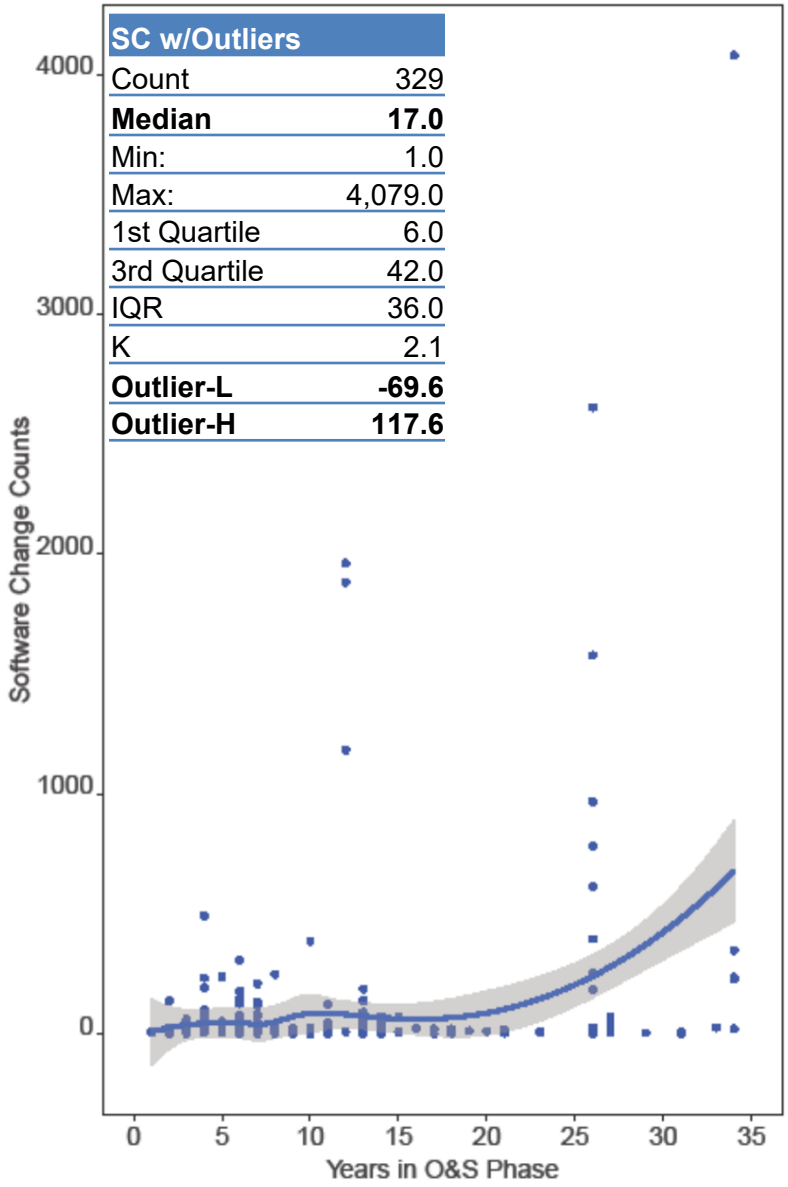
Outlier Labeling Rule

- In Exploratory Data Analysis [1], Tukey includes a rule of thumb for labeling (as "outside") observations that are extreme enough to be potential outliers. In terms of the lower fourth, F_L, and the upper fourth, F_U (approximate quartiles of the sample), an observation is "outside" if it does not fall in the interval between the cutoffs:
- $F_L - k(F_U - F_L)$ *and* $F_U + k(F_U - F_L)$
  - *(Note: $F_U - F_L$ is often referred to as the Inter-Quartile Range – IQR)*
  - Tukey's k is 1.5
- A subsequent study [2,3] shows that
  - For a low *n* of 5, k is above 3.0.
  - For an *n* above 10, the median value for **k is 2.1** for 90% and 2.3. for 95% probabilities that no observations are beyond the cutoffs.

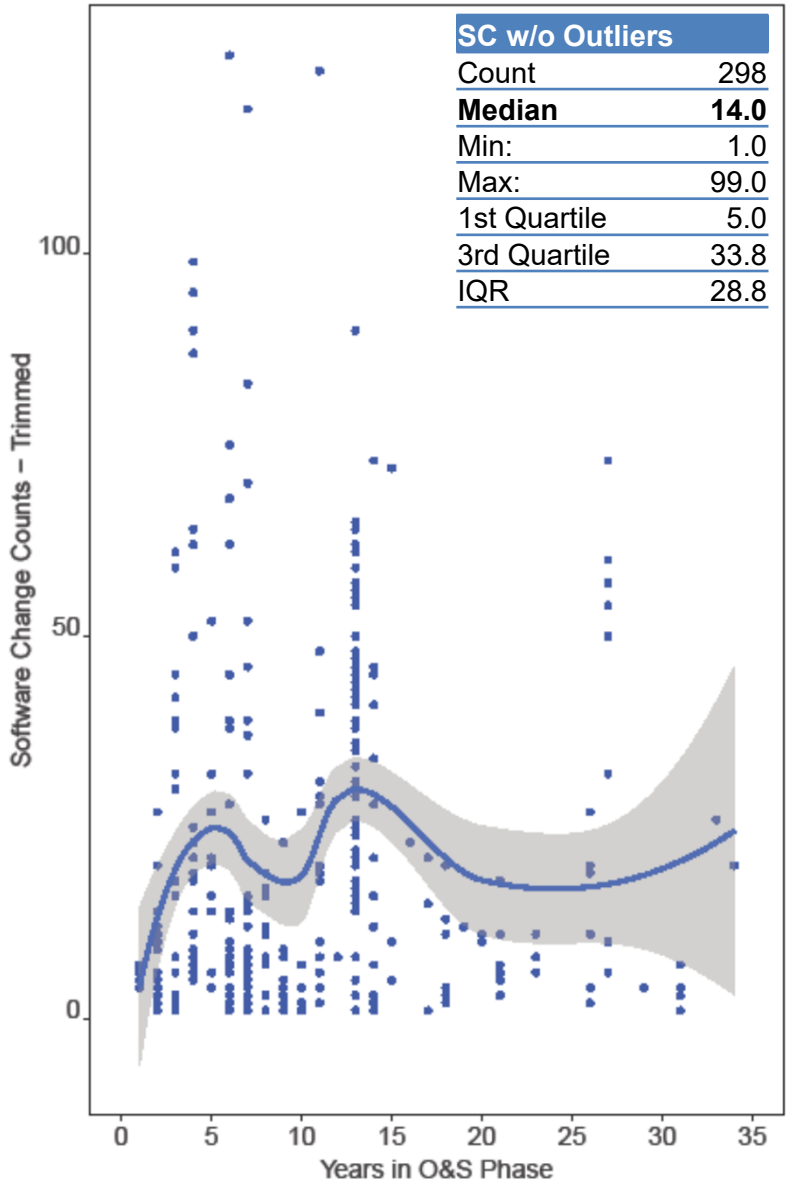


Software Changes over Time

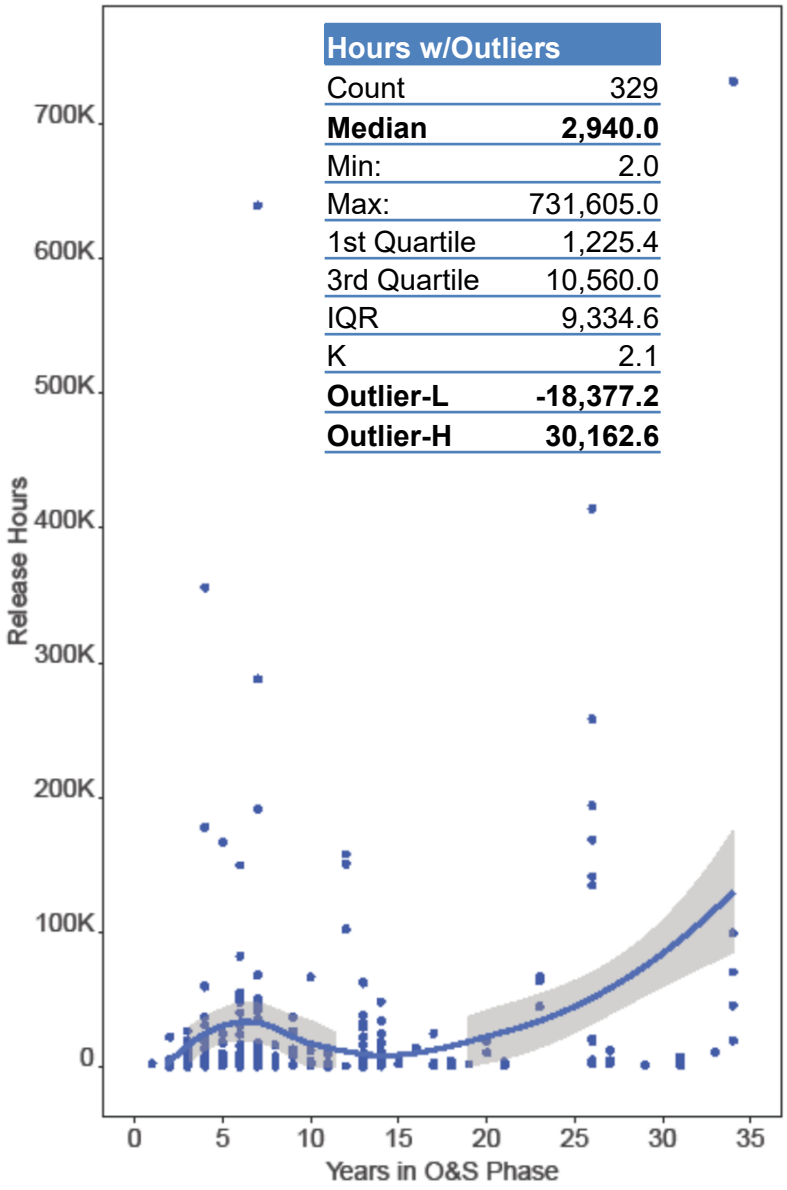
SC w/Outliers	
Count	329
Median	**17.0**
Min:	1.0
Max:	4,079.0
1st Quartile	6.0
3rd Quartile	42.0
IQR	36.0
K	2.1
Outlier-L	**-69.6**
Outlier-H	**117.6**



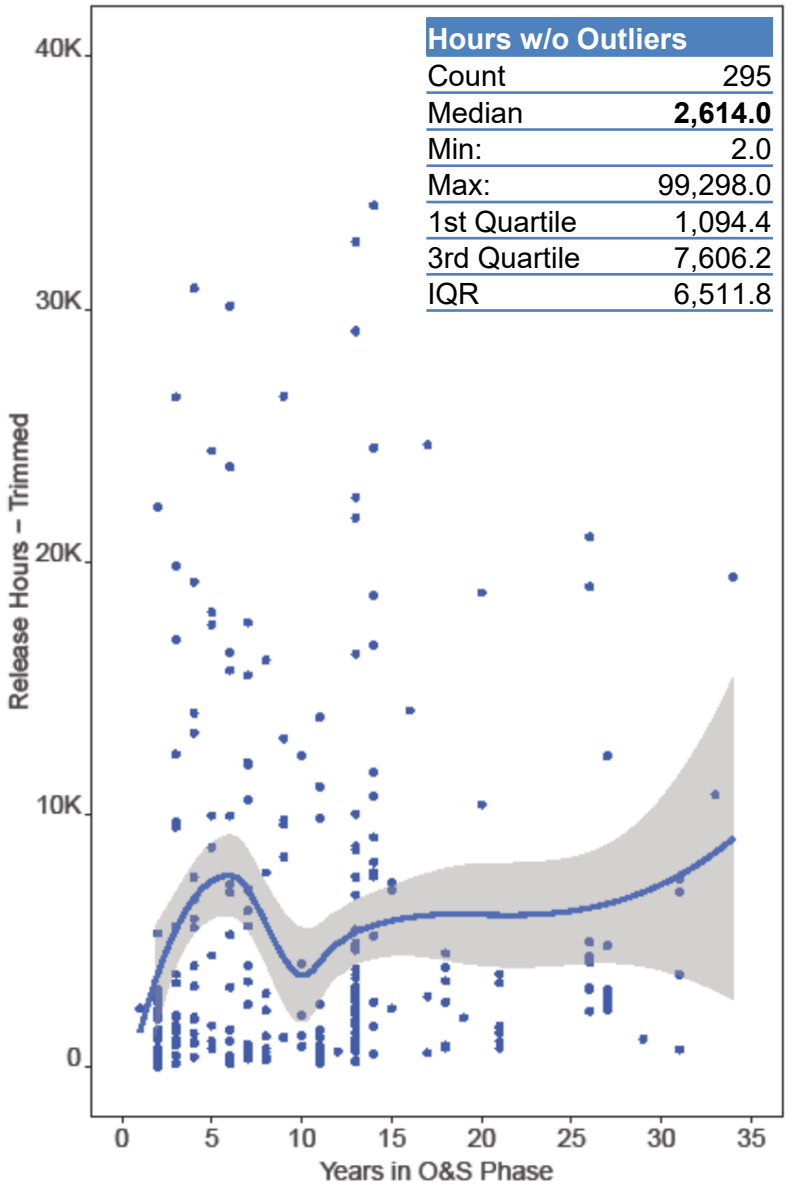
SC w/o Outliers	
Count	298
Median	**14.0**
Min:	1.0
Max:	99.0
1st Quartile	5.0
3rd Quartile	33.8
IQR	28.8



Release Hours over Time

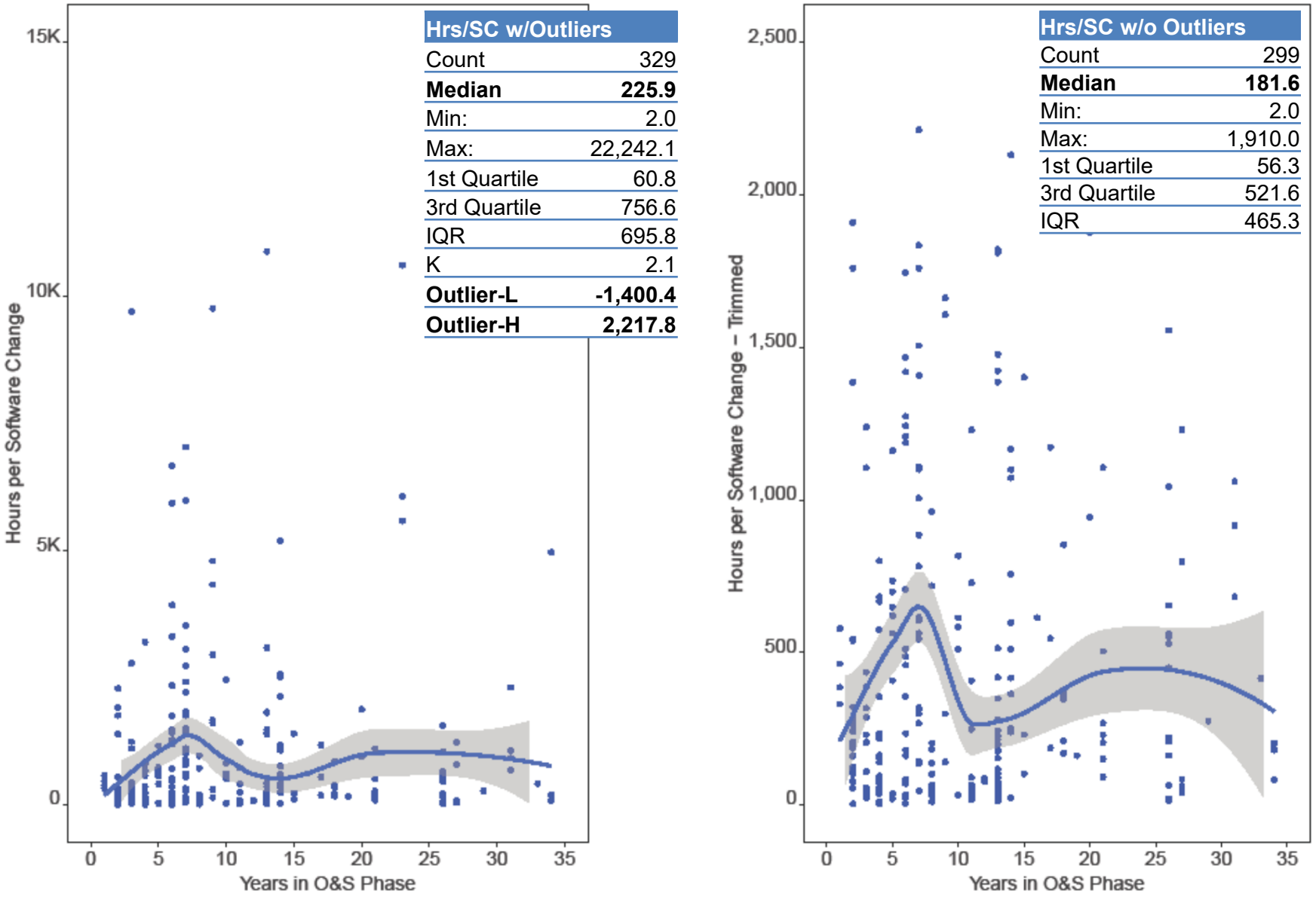


Hours w/Outliers	
Count	329
Median	**2,940.0**
Min:	2.0
Max:	731,605.0
1st Quartile	1,225.4
3rd Quartile	10,560.0
IQR	9,334.6
K	2.1
Outlier-L	**-18,377.2**
Outlier-H	**30,162.6**

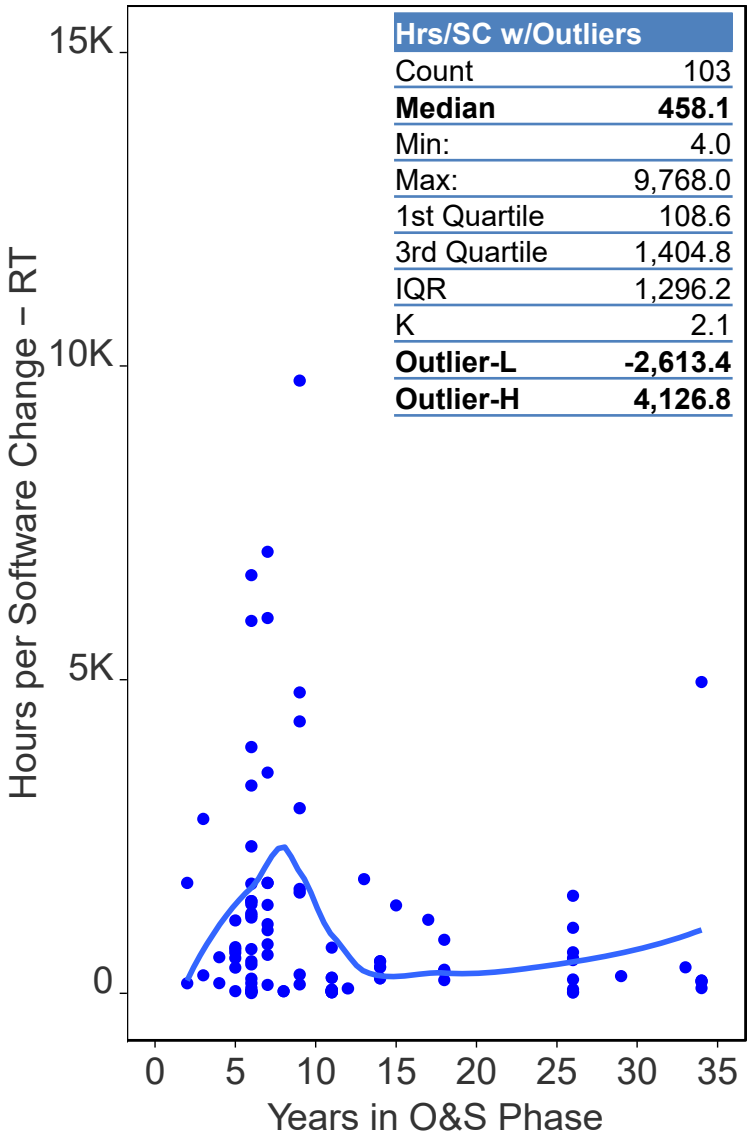


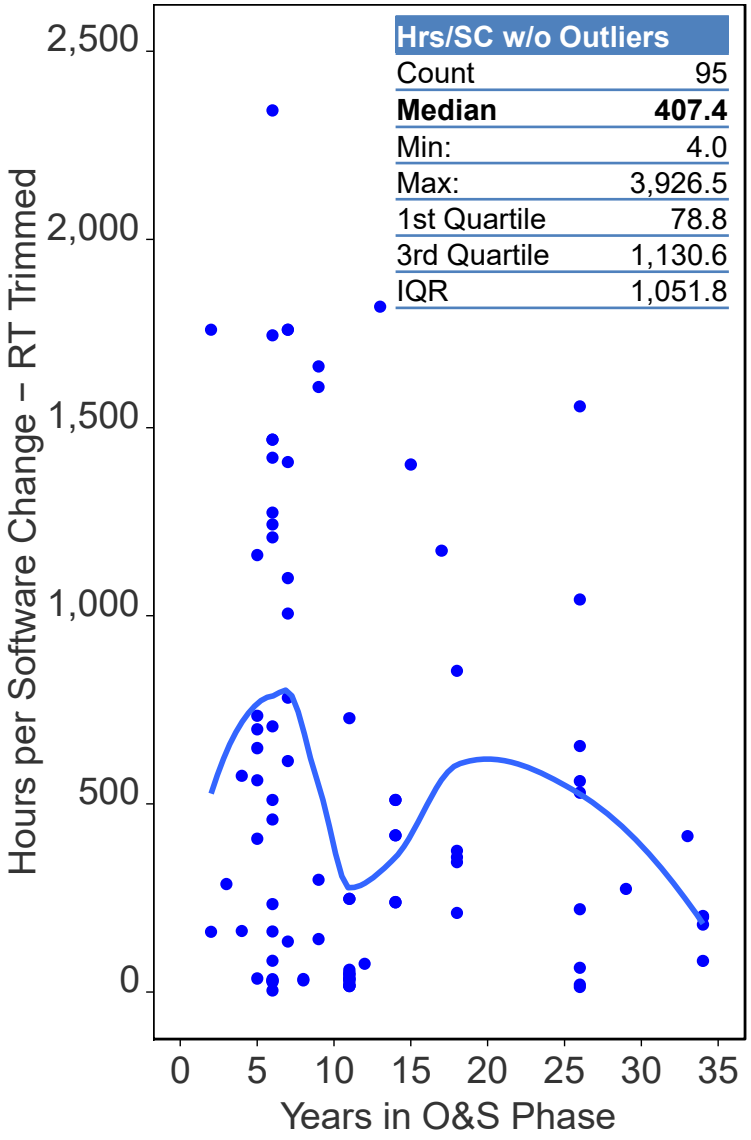
Hours w/o Outliers	
Count	295
Median	**2,614.0**
Min:	2.0
Max:	99,298.0
1st Quartile	1,094.4
3rd Quartile	7,606.2
IQR	6,511.8

Hours / Software Change over Time



Hours / Software Change over Time RT Domain





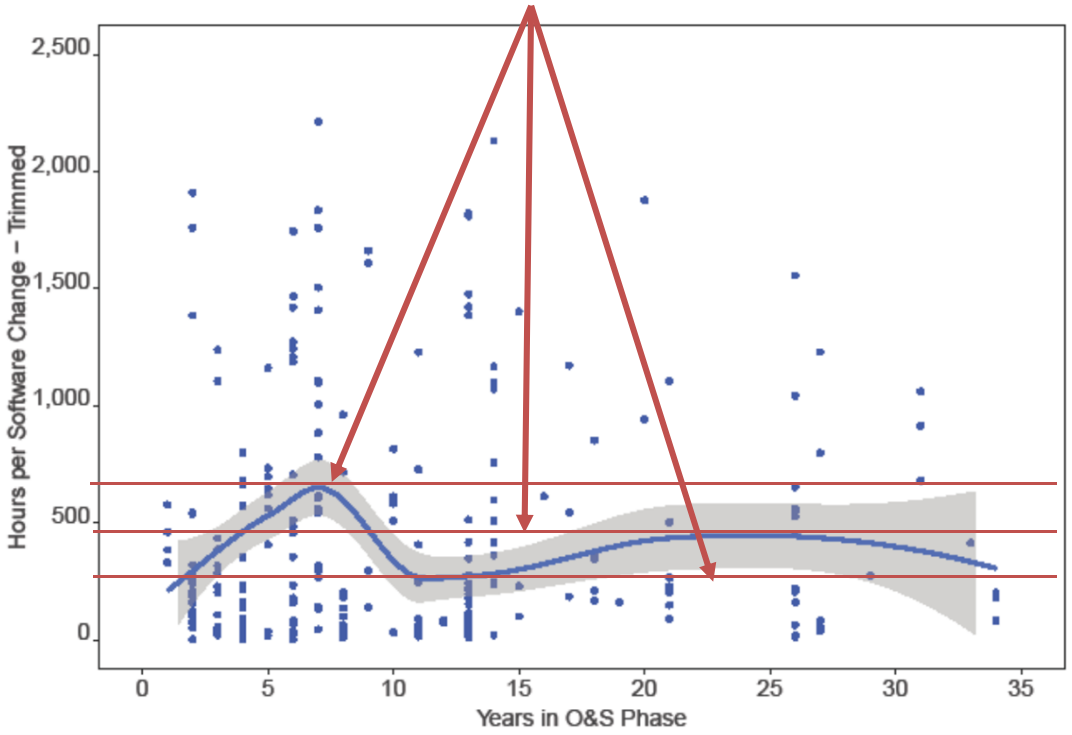
Implications for Software Cost

Unit cost changes over time

- Indicates that different CERs are needed at different times in the O&S phase

DevSecOps

- If everything in DevSecOps could be considered a Software Change (Enhancements & Repairs), at what level do you staff your Software Factory?



Conclusions

- The notional concept of the sustainment release profile appears to be valid although the rise and fall of sustainment action is different from the notional concept as shown by the data
- In the past, sustainment CERs were developed assuming sustainment actions were relatively constant over time, i.e., annual change traffic, but this appears to not be the case
- Budgeting approaches based on improved estimating methods need to be developed and applied
  - Otherwise, Program sustainment budgets will not reflect the actual work needed to be performed, which can lead to detrimental effects for the warfighter
- The implications of variable software changes during sustainment must be explored further for the DevSecOps environment

References

1. Tukey, J.S. Exploratory Data Analysis, Addison-Wesley, Reading MA, 1977
2. Hoaglin, D.C., Iglewiez, B., and Tukey, J.W. “Performance of some resistant rules for outlier labeling,” Journal of American Statistical Association, 1986, 81, 991-999. Labeling the technique “Outlier Labeling Rule.”
3. Hoaglin, D.C. and Iglewiez, B., “Fine tuning some resistant rules for outlier labeling,” Journal of American Statistical Association, 1987, 82, 1147-1149.

Contributors

Cheryl Jones

Measurement Analyst
US Army AFC
973-724-2644

James Judy

NISEC Division Chief
ODASA-CE
703-697-1612

Paul Janusz

Measurement Analyst
US Army AFC
973-724-4849

Dr. Bradford Clark

Vice President
Software Metrics, Inc.
(703) 754-0115

Dr. Robert Charette

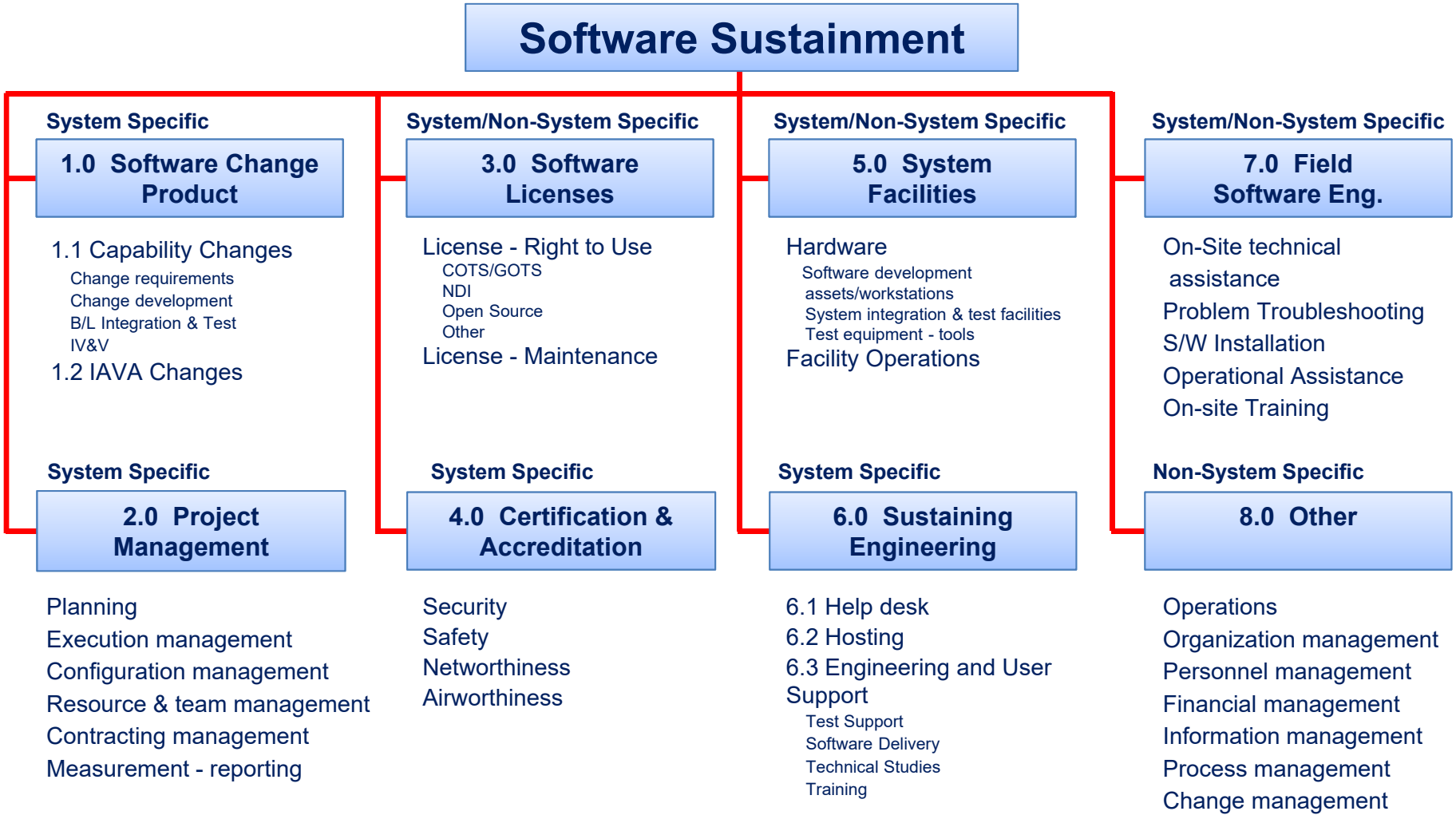
President
ITABHI Corporation
(540) 972-8150

Kimberly Roye

Senior Analyst
Galorath Federal
(703) 966-3192

Backup

DASA-CE SWS WBS



Version 5.0

Acronyms

- ACAT — Acquisition Category
- AIS — Automated Information Systems Super Domain
- ENG — Engineering Super Domain
- IAVA — Information Assurance Vulnerability Alert
- ICEAA — International Cost Estimating and Analysis Association
- IQR — Inter-Quartile Range
- O&S — Operations and Sustainment
- PDSS — Post-Deployment Software Support (PM management, development and/or production funding)
- PPSS — Post-Production Software Support (LCMC management, OMA funded)
- RT — Real-Time Super Domain
- SRDR-M — Software Resources Data Report for Maintenance
- SUP — Support Super Domain
- SWS — Software Sustainment
- THrs — Total Release Hours
- TSC — Total Software Changes
- WBS — Work Breakdown Structure