Software Cost Estimating: Friend or Foe? (to Agilists)

Carol Dekkers,
Quality Plus Technologies Inc.

PNSQC.ORG

PNSQC
OCTOBER 9-11 2023
AMP IT UP:
TRANSFORMING QUALITY

WHO AM I?
Carol Dekkers, PMP, CFPS (Fellow), CSM, SCEC, P.Eng.

Lead author of the International Cost Estimating and Analysis Association (ICEAA) Software Cost Estimating Body of Knowledge (CEBOK-S) and 2022 Educator of the year

ISO/IEC JTC1 SC7 project editor and member of USA delegation for software engineering standards since 1994

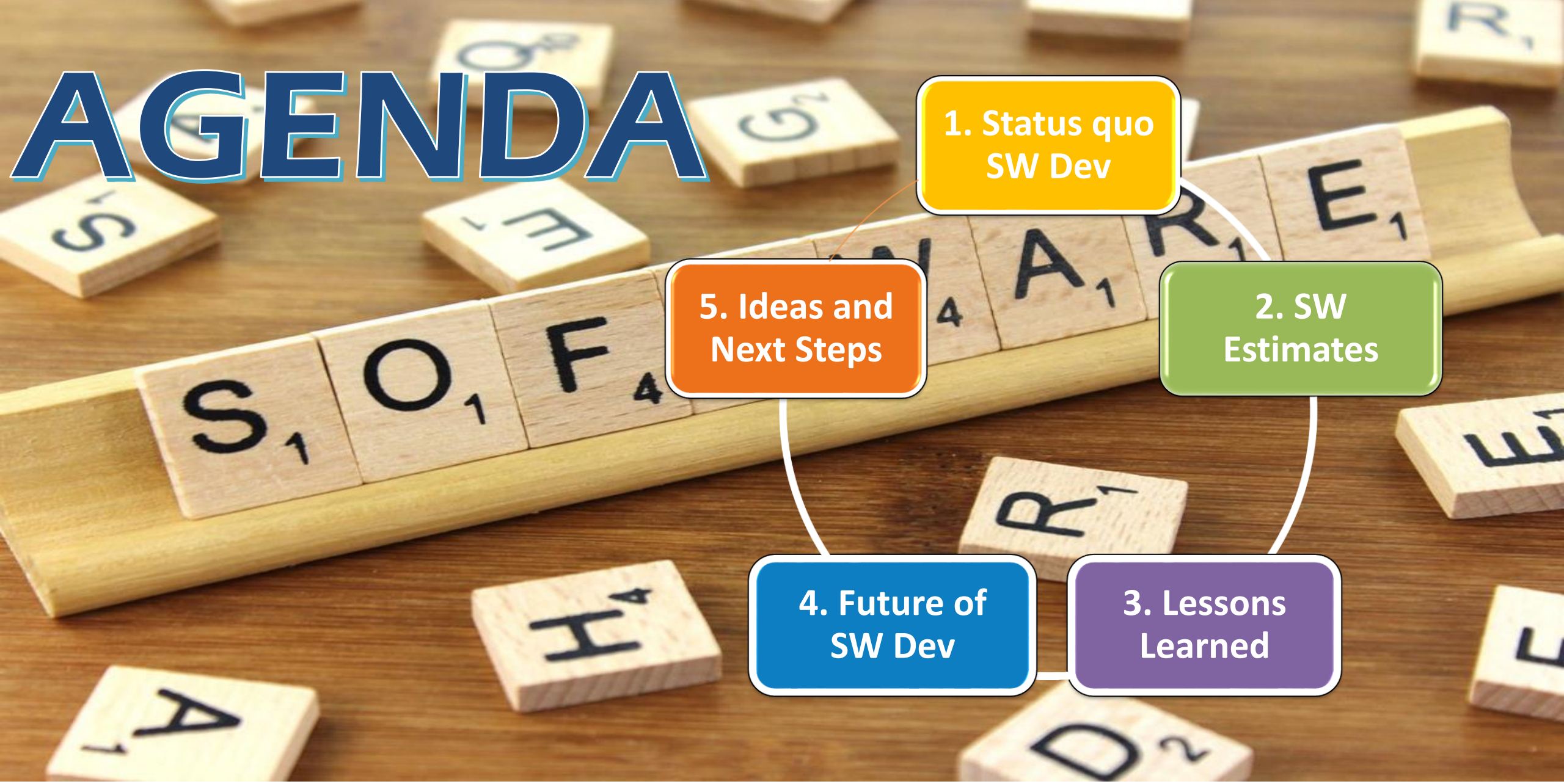
Founder, Quality Plus Technologies, Inc.

Past-president, International Function Point Users Group

Global consultant, published author, thought leader, speaker

© Carol Dekkers/Quality Plus Technologies, Inc.

AGENDA



© Carol Dekkers/Quality Plus Technologies, Inc.

1.Status quo

Current state of software development

© Carol Dekkers/Quality Plus Technologies, Inc.

Standish Group CHAOS Reports 1991-2020
Salient research on IT project success

- CHAOS = **Comprehensive Human Appraisal for Originating Software**. (Impact of human factors on project success.)
- Definitions:
  - **SUCCESS** = On-time, on-budget, meeting customer needs
  - **CHALLENGED** = Over schedule or budget or not meeting customer needs
  - **FAILURE** = Canceled or completed over-budget/late/not meeting customer needs
- 30 years of research spanned agile adoption and based on 50K projects
  → Success rates consistently 30-40%, (agile peaks at 42%), many attributable causes

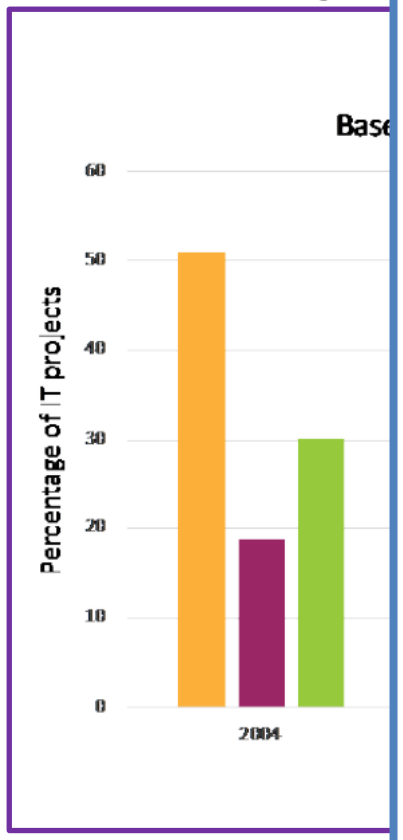
- **Research by others: PMI (Pulse of the Profession), SEI, GAO, support findings[1]**

1. PMI = Project Management Institute, SEI = Software Engineering Institute, GAO = US Govt Accountability Office

© Carol Dekkers/Quality Plus Technologies, Inc.

Software Development Status quo (2020)

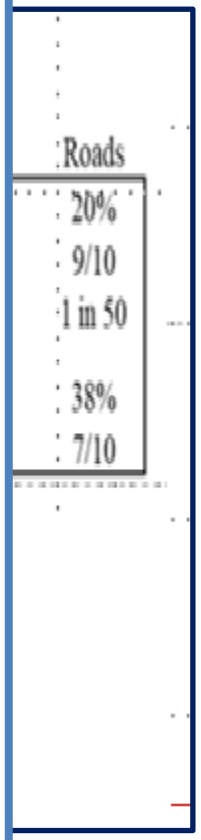
Software Projec



CHAOS Reports show Agile Progress ☺

2020 CHAOS Report "Beyond Infinity" ☹

- Agile projects = 2x successful as waterfall (42% to 26%) prev max = <40%
- Only 8% of agile projects = failures (vs 21% waterfall)
- 31.1% of projects are canceled before completion.
- 52.7% of projects cost 189% of their original estimates.
- 53% of projects were so worrying that they were challenged.



1. Standish Group CHAOS reports 2004-2020
2. Dr Christian Smart, *Solving for Risk Management: Understanding the Critical Role of Uncertainty in Project Management* (2021)

© Carol Dekkers/Quality Plus Technologies, Inc.

Status quo
Reasons for Project Failures & Challenges

Cause	Description	Business	Dev	Solutions (over the years)
"Poor" user input	Time, knowledge, not-my-job	X	X	Training, time, agile PO
Stakeholder conflicts	Shifting priorities, distance, communication, exclusion	X	X	Project mgmt, agile (backlog grooming, freq. delivery+++)
Vague and changing requirements	VUCA (volatility, uncertainty, complexity, ambiguity)	X	X	Terminology, prototyping, agile dev, collaboration
Skills (do not match the job)	Technology, SME, management	X	X	Training, experience
Failure to plan	Structure, SW without BPR, no PM	X	X	Structure, PM, sponsor, data
Communication breakdown	Blame, abdication, litigation	X	X	Training, agile standup/retro, VOC
Poor architecture	Obsolete/redundant tech/data/apps		X	Planning
Late failure warning signals	Few/no measures, punish mindset	X	X	Measurement, agile coaching
Poor cost & schedule estimates	**Over-optimistic, ad hoc, lacking in context & uncertainty**	**X**	**X**	**Formal approach, collaboration, better quality historical data**
Hidden $ of Lean and Mean	**Unrealistic goals and resources**	**X**	**X**	**Historical data, agile (fail early)**

Adapted from sources including : Standish Group CHAOS reports, (circa 2015), PMI Reports, author research, Loren May, CrossTalk,, https://herdingcats.typepad.com/my_weblog/2013/01/unrealistic-cost-and-schedule-estimates.html,

CHAOS - the Comprehensive Human Appraisal for Originating Software. Basically all about the human factor and how it influences project success.

© Carol Dekkers/Quality Plus Technologies, Inc.

2. SW Estimates

Typically composed of 4 parts:

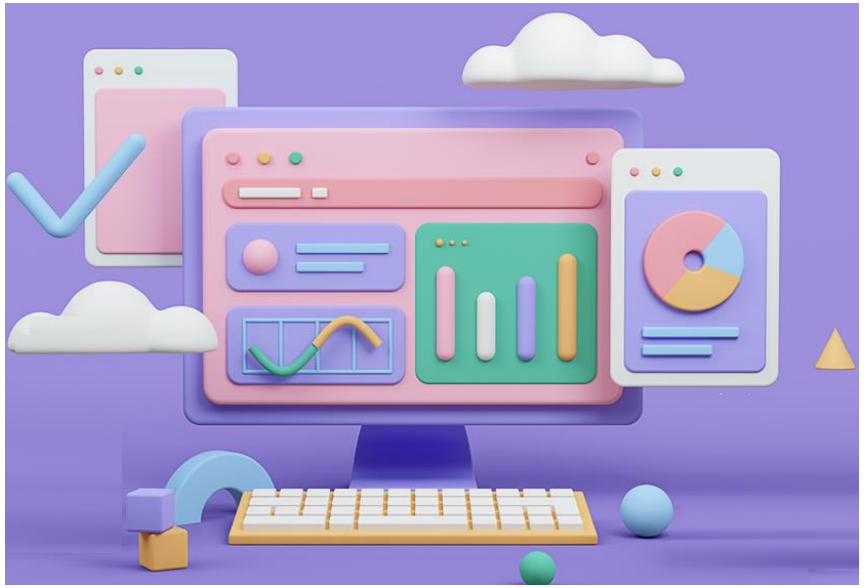
- Software size estimate
- Software effort estimate
- **Software cost estimate**
- **Software schedule estimate**

© Carol Dekkers/Quality Plus Technologies, Inc.

What are "Poor" cost and schedule estimates

- "Poor" = "Unrealistic" = Overly-optimistic = Indefensible

- Causes of poor software estimates:
  - Incomplete/premature source documents
  - Planning fallacy → ignore/override data & past performance factors; human = hard-wired optimist
  - Lack of quality data → incorrect/inconsistent historical "actuals" (scope, size, $, schedule, context…)
  - Lack of documented ground rules & assumptions
  - Lack of developer input (theory > experience) & review
  - Lack of adjustment for risk, uncertainty, growth or cross-checks
  - Lack of formal, mature software cost estimating approaches
  - Lack of range and confidence (uncertainty)



© Carol Dekkers/Quality Plus Technologies, Inc.

What is the impact of "poor" cost and schedule estimates?

Standish Group on U.S. government / business:

$81 B USD = canceled software projects

$59 B USD = budget overruns

Impact is compounded when estimates (best guess, given info at hand) become (unintended) budgets, timelines & targets

© Carol Dekkers/Quality Plus Technologies, Inc.

"Poor" estimates contribute to poor project outcomes

1. Few strong inputs, low estimating maturity leads to…

2. Weak (unreliable) estimates which lead to…

3. Unrealistic plans (cost, effort, duration) which lead to…

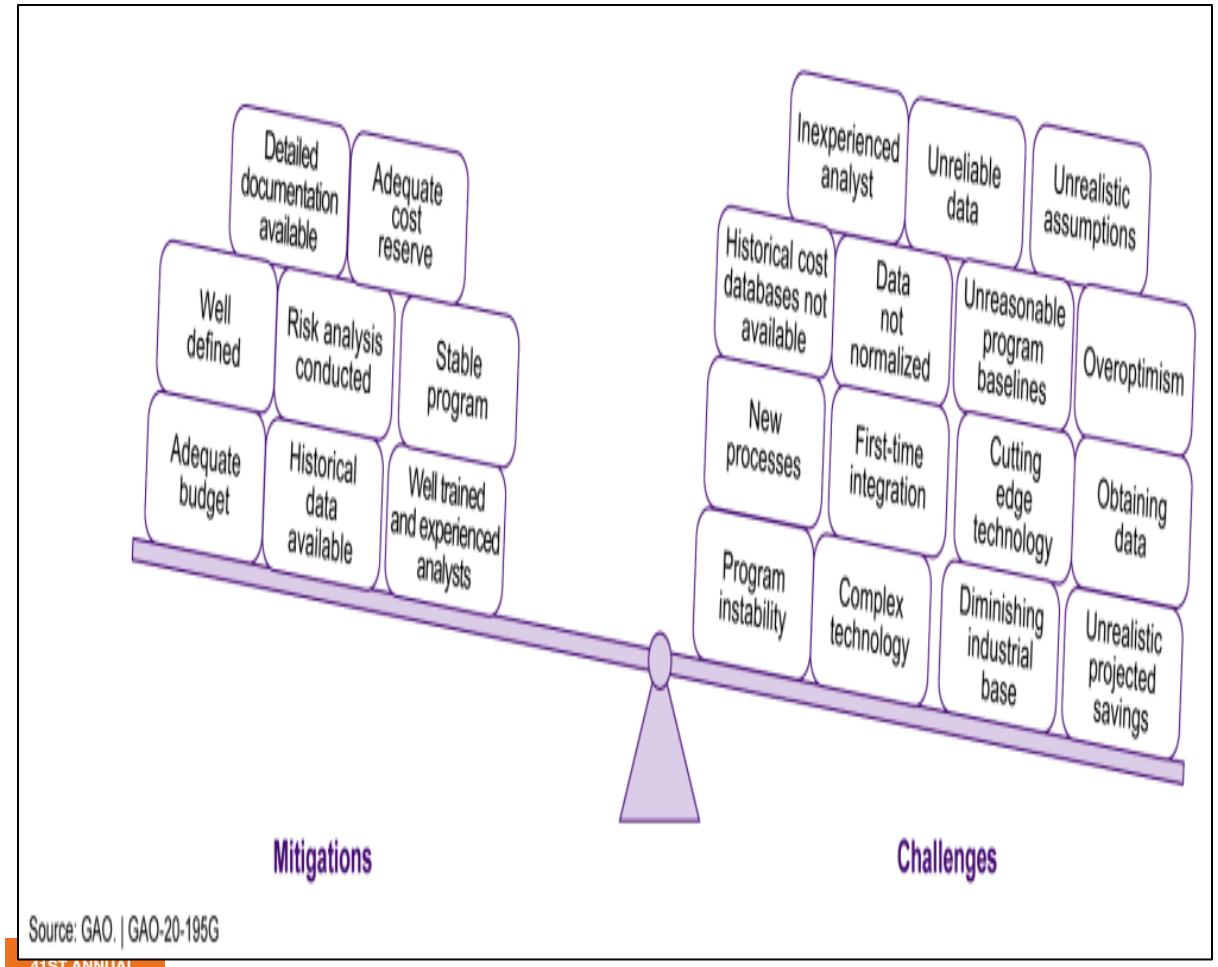
4. Potentially Impossible contracts/ internal projects resulting in…

5 Out-of-control projects with adverse outcomes …

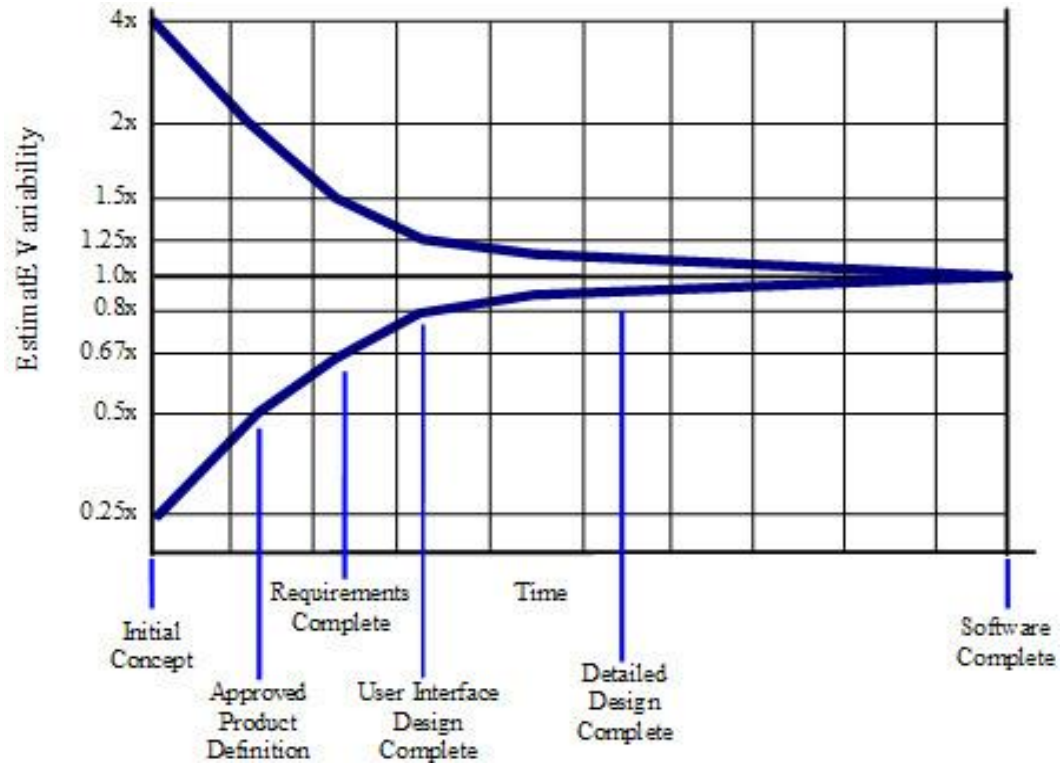
Source: CEBoK-S

© Carol Dekkers/Quality Plus Technologies, Inc.

Cost estimating
Challenges, mitigations and the Cone of Uncertainty



Cone of Uncertainty (www.construx.com)



© Carol Dekkers/Quality Plus Technologies, Inc.

Given the status quo + Agile uncertainty, Why estimate?

Mandatory US Government requirement: "Cost estimates are necessary for government acquisition programs for many reasons: to support decisions about funding one program over another, to develop annual budget requests, to evaluate resource requirements at key decision points, and to develop performance measurement baselines."

GAO-20-195G, Cost Estimating and Assessment Guide: Best Practices…

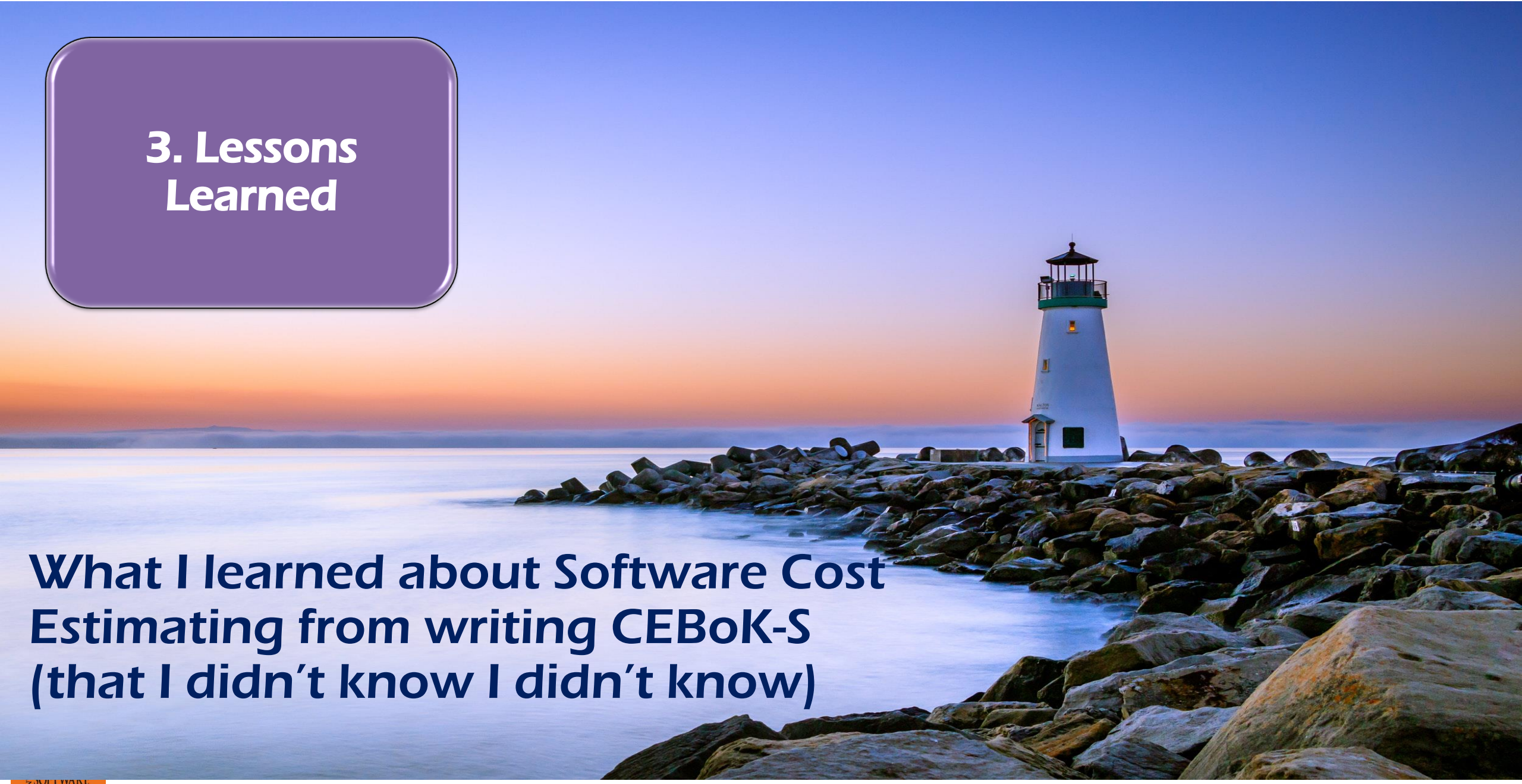
IT'S NOT REALLY A CHOICE

Corporate funding requirement: "An estimate helps to plan and coordinate product releases, synchronize work with other teams, ensure that resources are allocated properly to meet the needs of the product, and of course to enable accurate billing of clients when your team has been hired to do a job for an outside company."

Why Devs (Should) Like Estimates by Yaacov Ellis, Oct 2019

© Carol Dekkers/Quality Plus Technologies, Inc.

3. Lessons Learned



What I learned about Software Cost Estimating from writing CEBoK-S (that I didn't know I didn't know)

© Carol Dekkers/Quality Plus Technologies, Inc.

1. Data-founded estimates are more defensible than theory (or expert opinion)

- **Programmatic assumptions** → can be mitigated with data
- **Quality historical data** → real (and repeatable)
- **Data analysis & normalization are critical** → relevant, similar, comparable (units of measure, scope, who, what, OT, etc.)
- **Overcoming "Planning Fallacy"** → historical data facts > optimist opinion

© Carol Dekkers/Quality Plus Technologies, Inc.

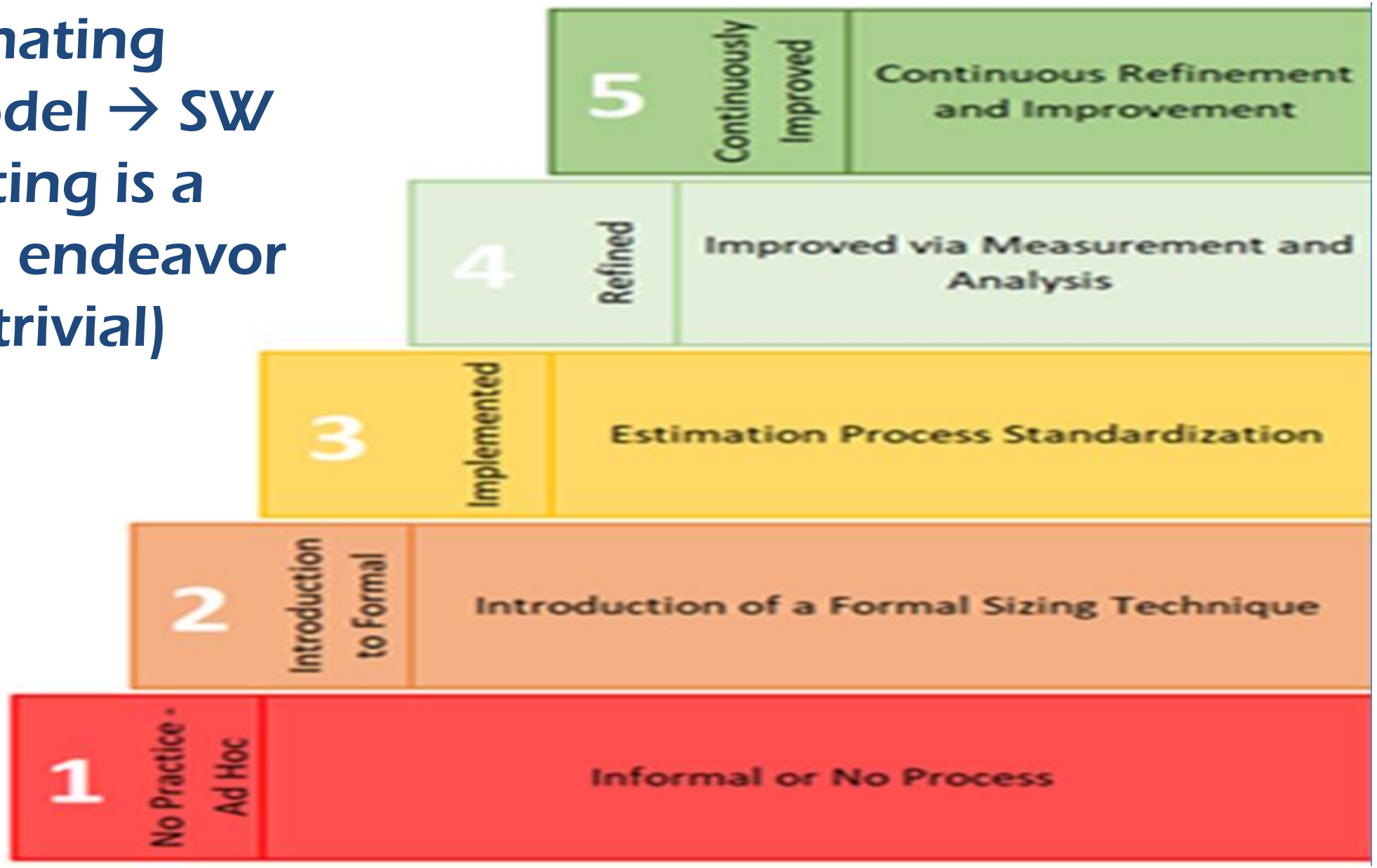
If you don't have historical data…

- **External available data:** Commercial tools or ISBSG[1] repository
- **Wide Band Delphi expert opinion:** SME experience in lieu of data
- **Cone of uncertainty & Rules of thumb**
- **Defensible estimate:** "Line in the sand"
- **T&M contracts with flexibility for change:** *See Acquisition & Management Concerns for Agile Use in Government Series Estimating in Agile Acquisition by the SEI*

1. ISBSG International Software Benchmarking Standards Group Development & Enhancement repository >10K projects

© Carol Dekkers/Quality Plus Technologies, Inc.

2. Cost Estimating Maturity Model → SW Cost Estimating is a professional endeavor (and is non-trivial)

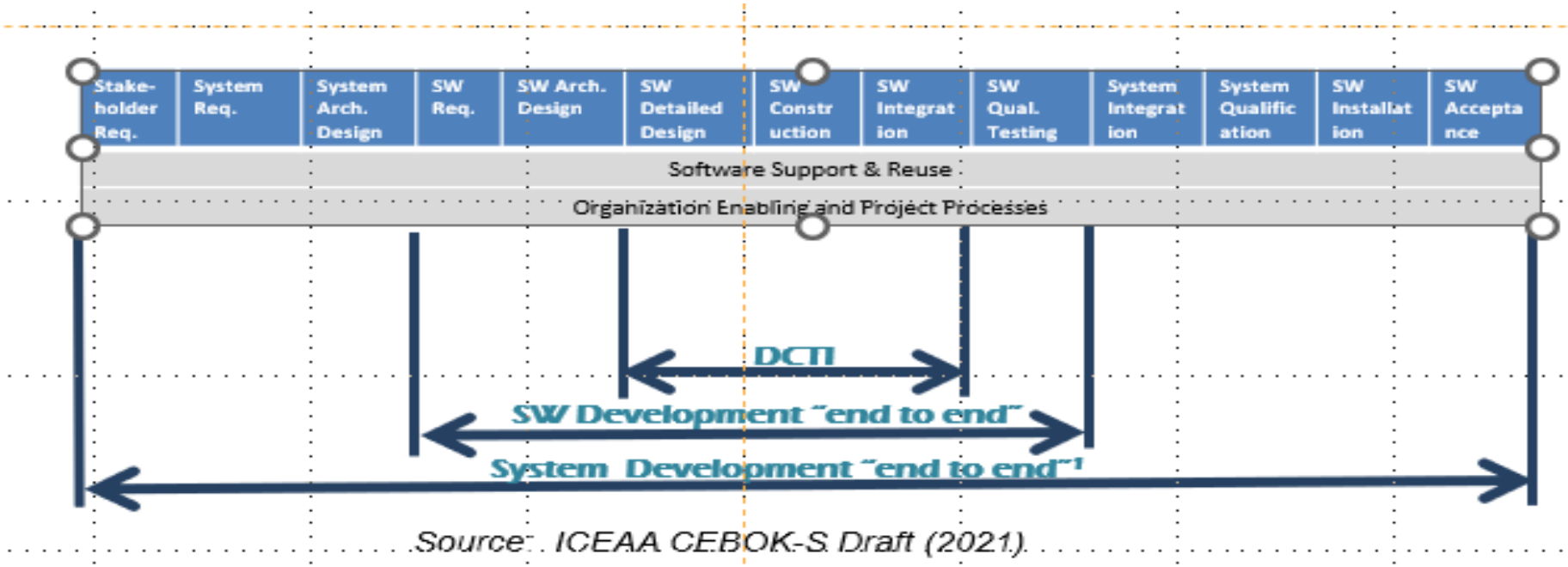


Most companies use informal estimating practices (Level 1)

Source: Adapted from Estimation Maturity Model by Dan Galorath and Esteban Sanchez, Galorath.com

© Carol Dekkers/Quality Plus Technologies, Inc.

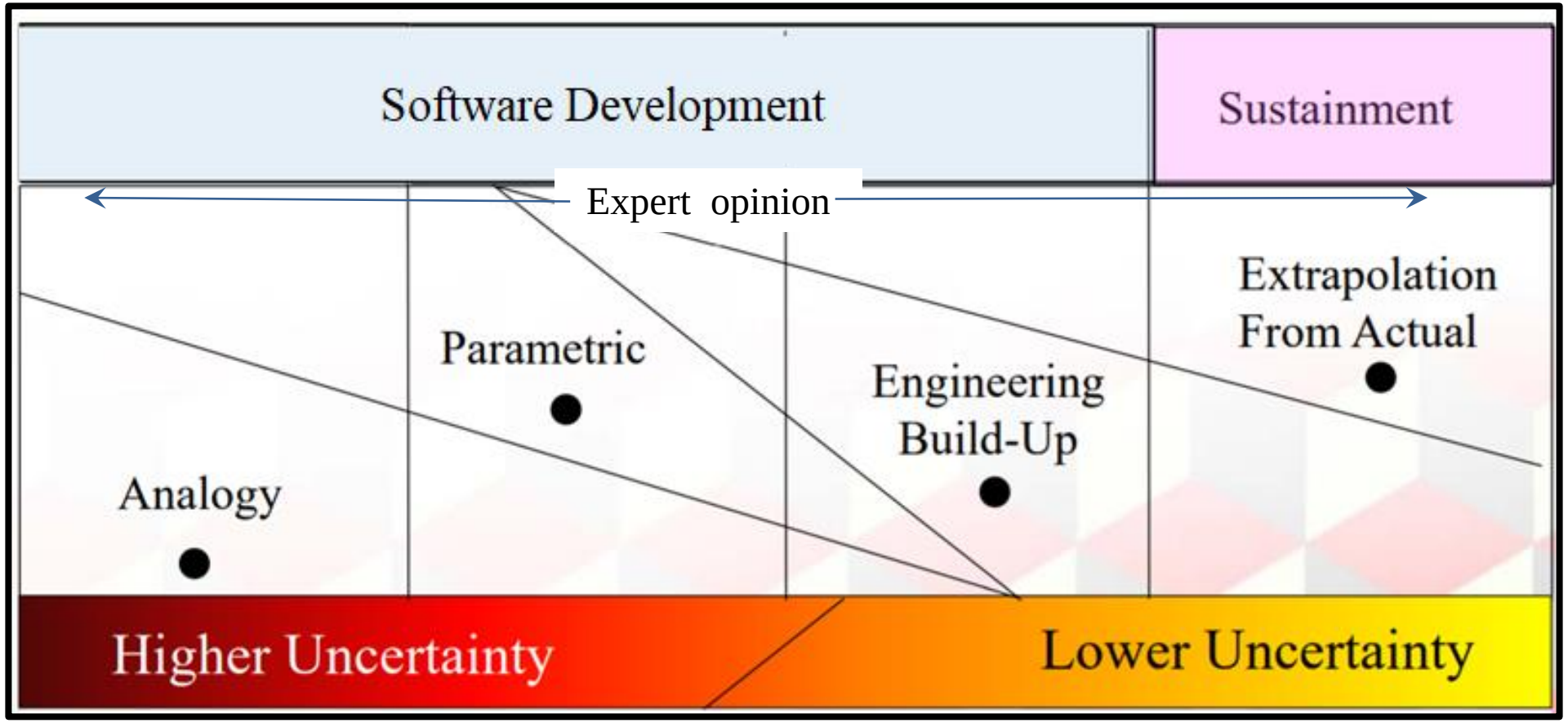
3. Scope of the Estimate is critical: Range of software activities



- **Design, Code, Test, Integration (DCTI)** factors cover the "core" parts of the process; other activities must be estimated separately or significant omissions will occur in the estimate
- **Software "end to end"** productivity attempts to cover all "software-specific" activities; higher level systems engineering activities must be estimated separately where relevant
- **SYSTEM END TO END (LCCE)** –full system life cycle cost estimate

© Carol Dekkers/Quality Plus Technologies, Inc.

4. There is no "One Size Fits All" software cost estimating approach

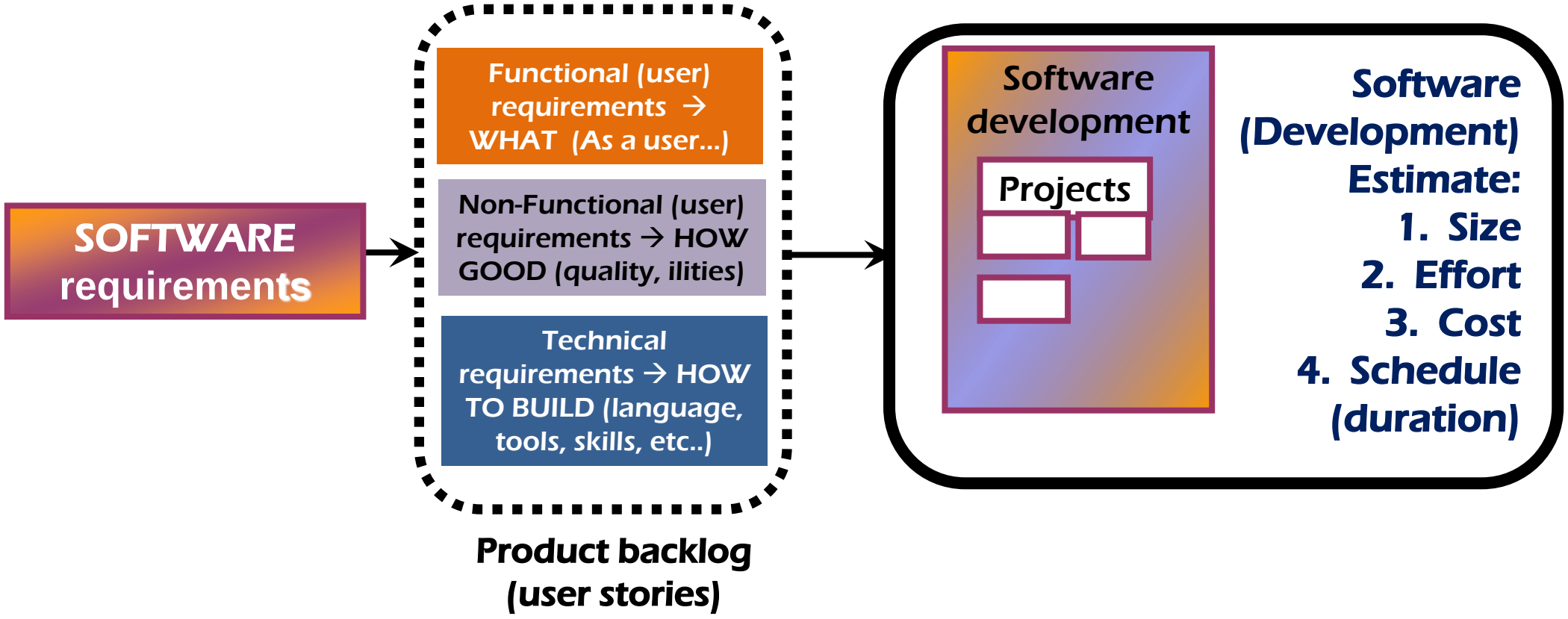


Source: ICEAA CEBOK-S Draft (2021)

© Carol Dekkers/Quality Plus Technologies, Inc.

41ST ANNUAL PACIFIC NW SOFTWARE QUALITY CONFERENCE OCT. 9-11,2023

5. Quantifying Software Size is fundamental to a good parametric estimate



Source: Quality Plus Technologies, Inc.

© Carol Dekkers/Quality Plus Technologies, Inc.

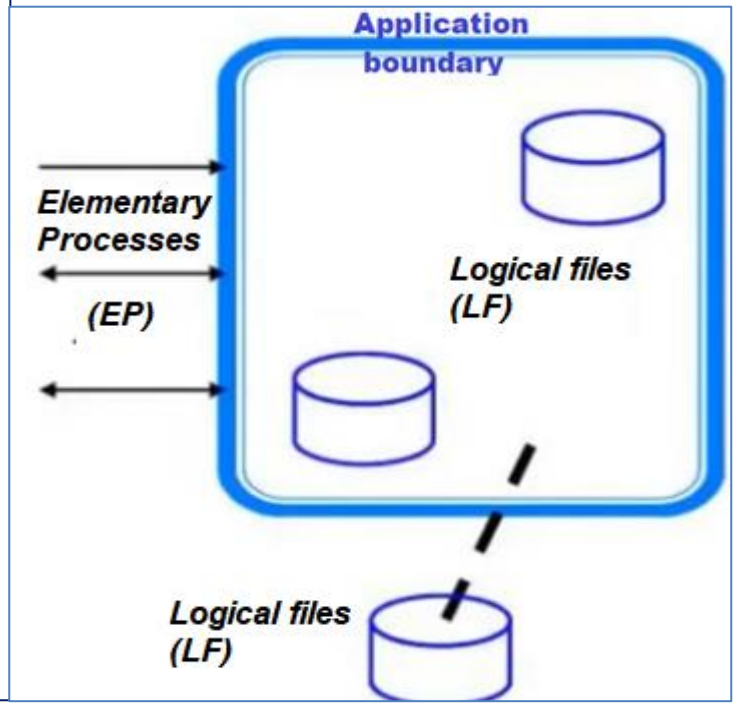
6. Software Size can be estimated using standard UOM (based on ConOps/EPICS/prelim backlog)

EXAMPLE - Simple Function Points[1] (SFP) is standardized Unit of Measure (International Function Point Users Group SFP v2.1):

- Software Size = # Logical Files * 7 SFP (each) +
  # Elementary Processes * 4.6 SFP (each)

Example ConOps requirements:

- Maintain customers = Entity + CRUD = 1 LF + 4 EP = 25.4 SFP
- 10 unique output reports = 10 EP = 46 SFP
- Receive file of customer data from other app = 1 EP = 4.6 SFP
- Notify administrator of security breach = 1 EP = 4.6 SFP
- Configure customer account limits = Entity + CRUD = 25.4 SFP

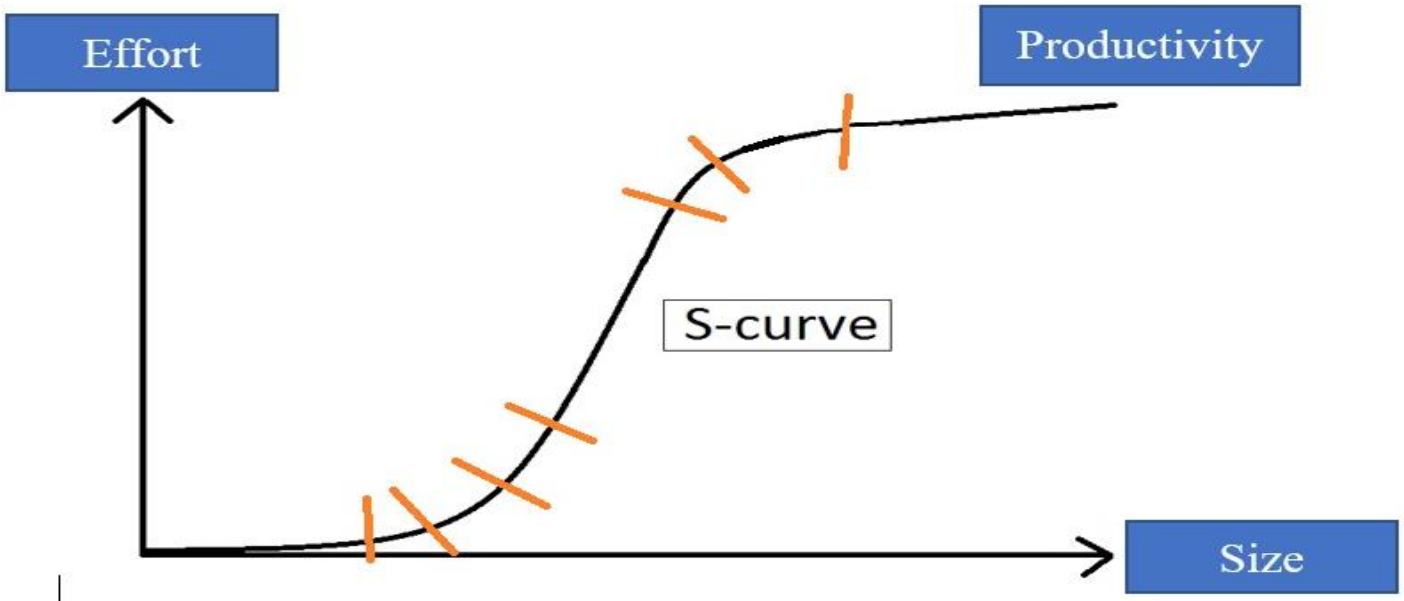


1. Simple FP details can be found at www.ifpug.org

© Carol Dekkers/Quality Plus Technologies, Inc.

7. Software development cost is non-linear (S-curve) and subject to Diseconomies of Scale (EXP > 1)

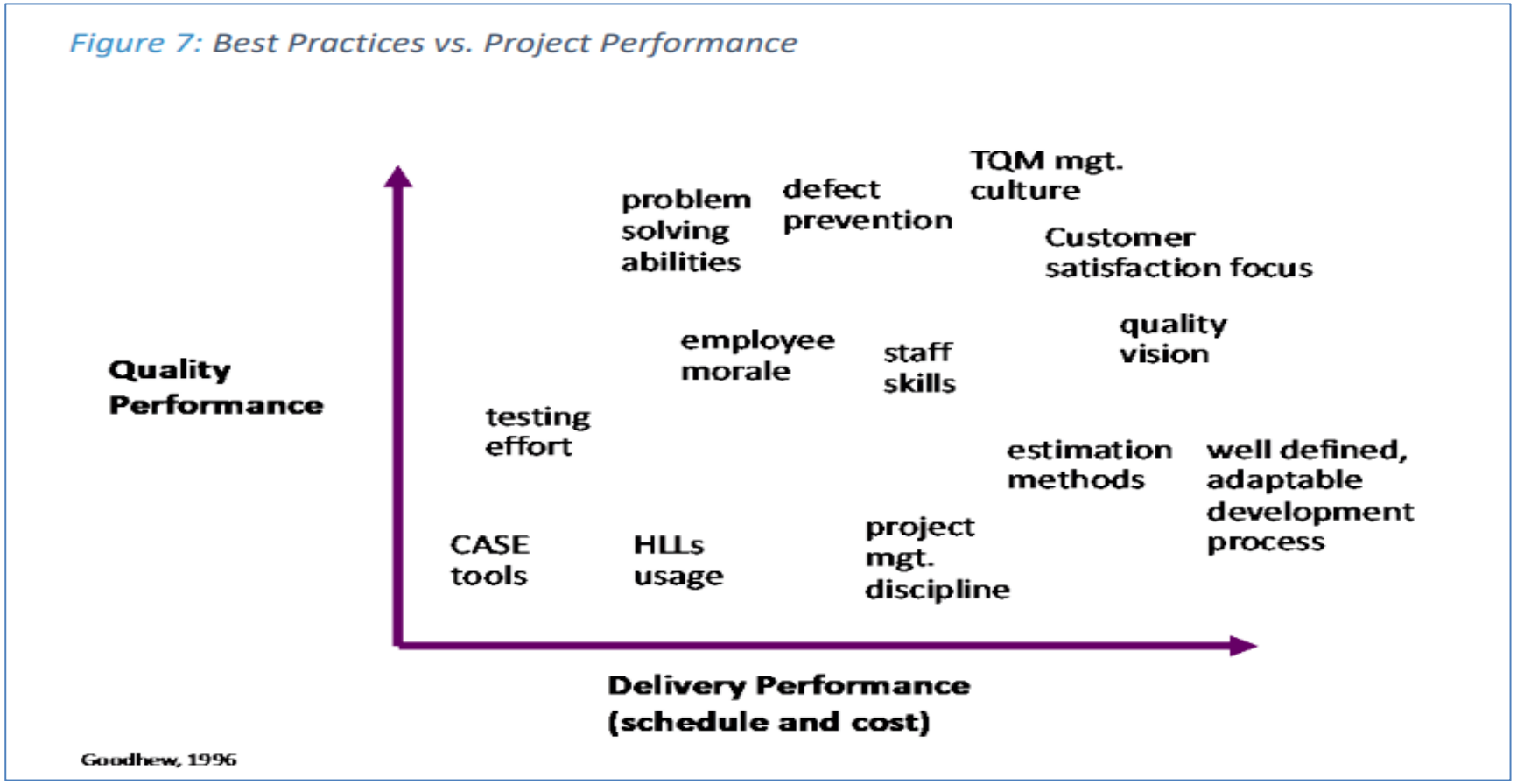
$Effort = Size^{EXP} * 1/Productivity$



Size and productivity are primary cost drivers of software development

© Carol Dekkers/Quality Plus Technologies, Inc.

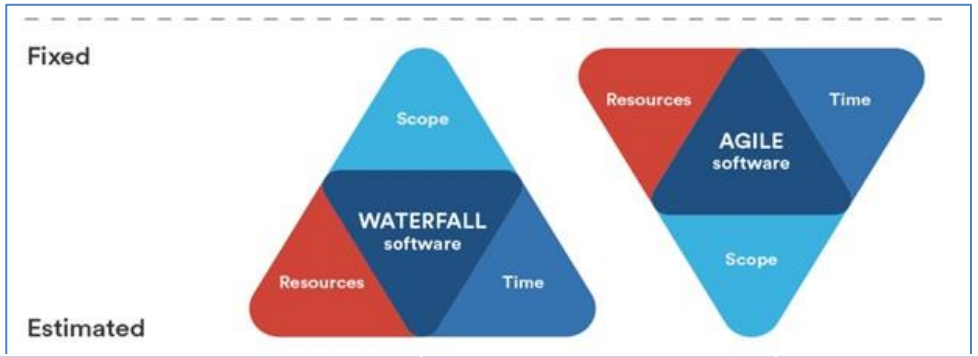
Software quality also has an impact[1]



1. The Cost of Poor Software Quality in the US: A 2020 Report by Consortium for Info and SW Quality: https://www.it-cisq.org/cisq-files/pdf/CPSQ-2020-report.pdf

© Carol Dekkers/Quality Plus Technologies, Inc.

8. Agile development approach → different cost (estimating) considerations

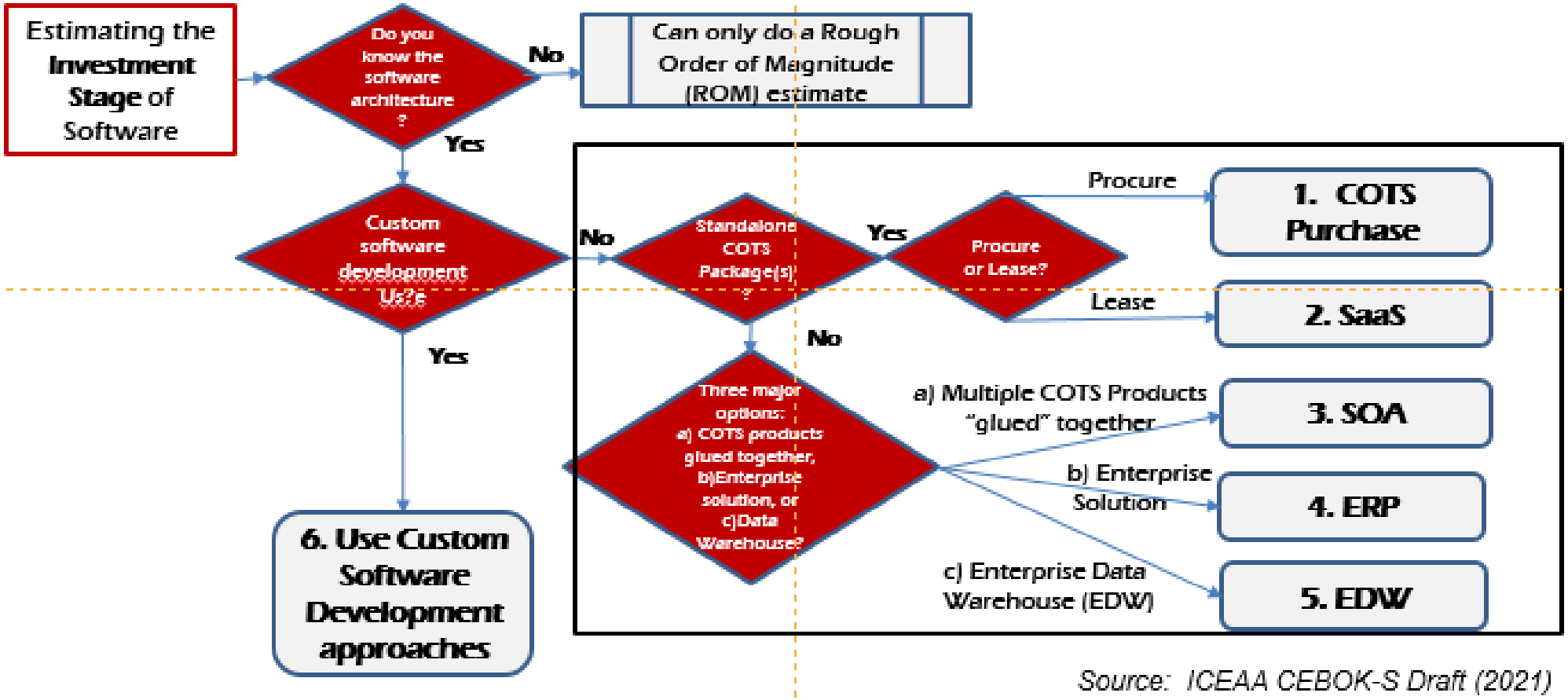


Adapted from
https://www.process.st/waterfall-vs-agile/

Topic	Agile	Predictive / waterfall	Hybrid-agile (Partly predictive/partly agile) TYPICALLY GOV"T
Fixed variables	Cost & schedule	Scope (requirements)	Initial high-level scope fixed, but is flexible to prioritization and change during development
Estimated	Scope (features)	Cost & schedule	Cost & schedule initially estimated, but becomes fixed during development
Driver	Change-driven	Plan-driven	Hybrid
Development risks[1]	Cost & schedule mostly fixed. Delivered size may fall short	Scope fixed → cost & schedule overruns	Cost & schedule mostly fixed during build. Delivered scope may fall short

© Carol Dekkers/Quality Plus Technologies, Inc.

9. Estimating hybrid software solutions require different approaches: Development vs Procurement

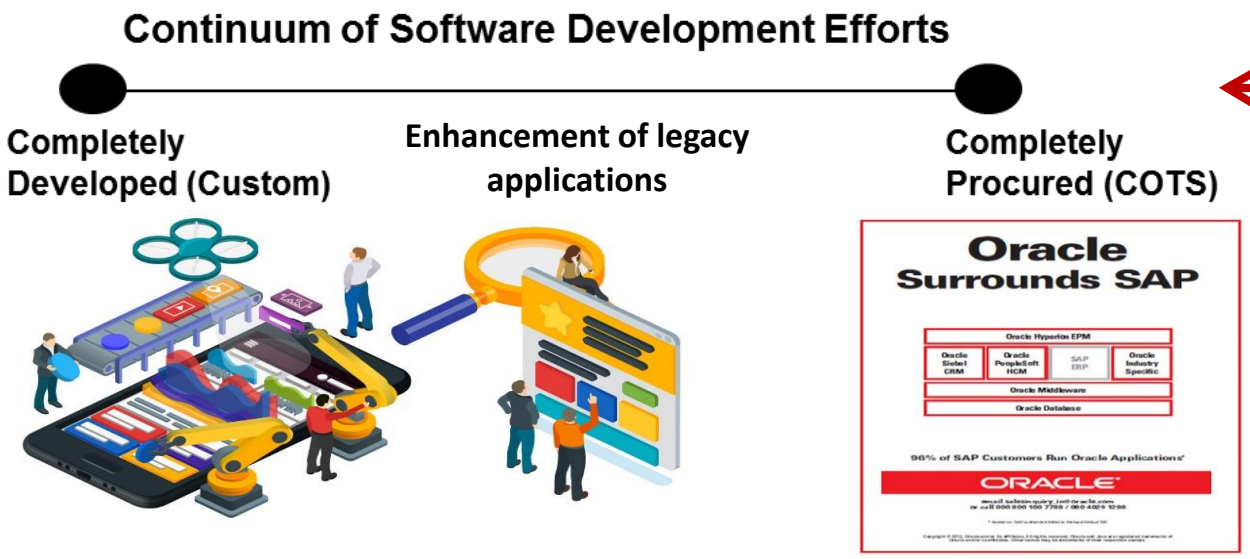


Hybrid solutions (combination) → create multiple cost estimates as needed

Source: ICEAA CEBOK-S Draft (2021)

© Carol Dekkers/Quality Plus Technologies, Inc.

10. Continuum of Development vs Procurement → Estimation can be complex



Software development paradigms

Configuration, licensing, integration, customization

All software efforts (with or without system activities) lie on a continuum of completely developed (custom) to completely procured[1]

Procured software involves effort to configure, integrate, and customize (see Lesson 6)

Source: CEBoK-S (2021)
1 Commercial Off-the-Shelf (COTS) and Information System (IS)/Business Systems software (packages) are covered in Lesson 6
This slide is used here to depict where software development paradigms fit in the context of software development

© Carol Dekkers/Quality Plus Technologies, Inc.

4. Future of SW Development

© Carol Dekkers/Quality Plus Technologies, Inc.

2020 CHAOS Report Beyond Infinity[1]

- **Biggest cause of software failures is "poor decision latency"**
  Decision latency is the amount of time it takes for a team to make a decision in response to a business change. For even the most sophisticated teams, the time between noticing a troubling change, formulating questions, reaching an answer, and taking action can be days, and in some cases weeks.
- **Redefined project maturity based on 3 success factors:**
  - **A good sponsor**
  - **A good team**
  - **A good place**
- **Success rates go up dramatically when using agile + DevOps in a highly mature way, → minimizing decision latency**
- **Introduced "Infinite Flow (Flow) is a non-project-based software development and implementation environment. Flow is a method to manage software development, implementation, and maintenance through a continuous process."**
- **No need for project budgets or estimates… continuous flow**

- **Sounds like mix of DevOps(Sec) with Kanban, Heart of Agile, *etc.***

1. Adapted from https://hennyportman.wordpress.com/2021/01/06/review-standish-group-chaos-2020-beyond-infinity/#comments and https://www.it-cisq.org/cisq-files/pdf/CPSQ-2020-report.pdf

© Carol Dekkers/Quality Plus Technologies, Inc.

Capers Jones on the impact of AI on Software Development[1]

"Within a few years artificial intelligence may be able to develop large systems in the 10,000- function point size range in less than 3 weeks instead of more than 3 calendar years which is the average for 2023. The main schedule driver would be requirements rather than design or coding."

"AI can also handle management tasks before software development begins including but not limited to:

- Predicting the size of the application in both lines of code and function points.
- Predicting the numbers of organizations that will want to use the application.
- Predicting probable updates for 3 years after deployment."

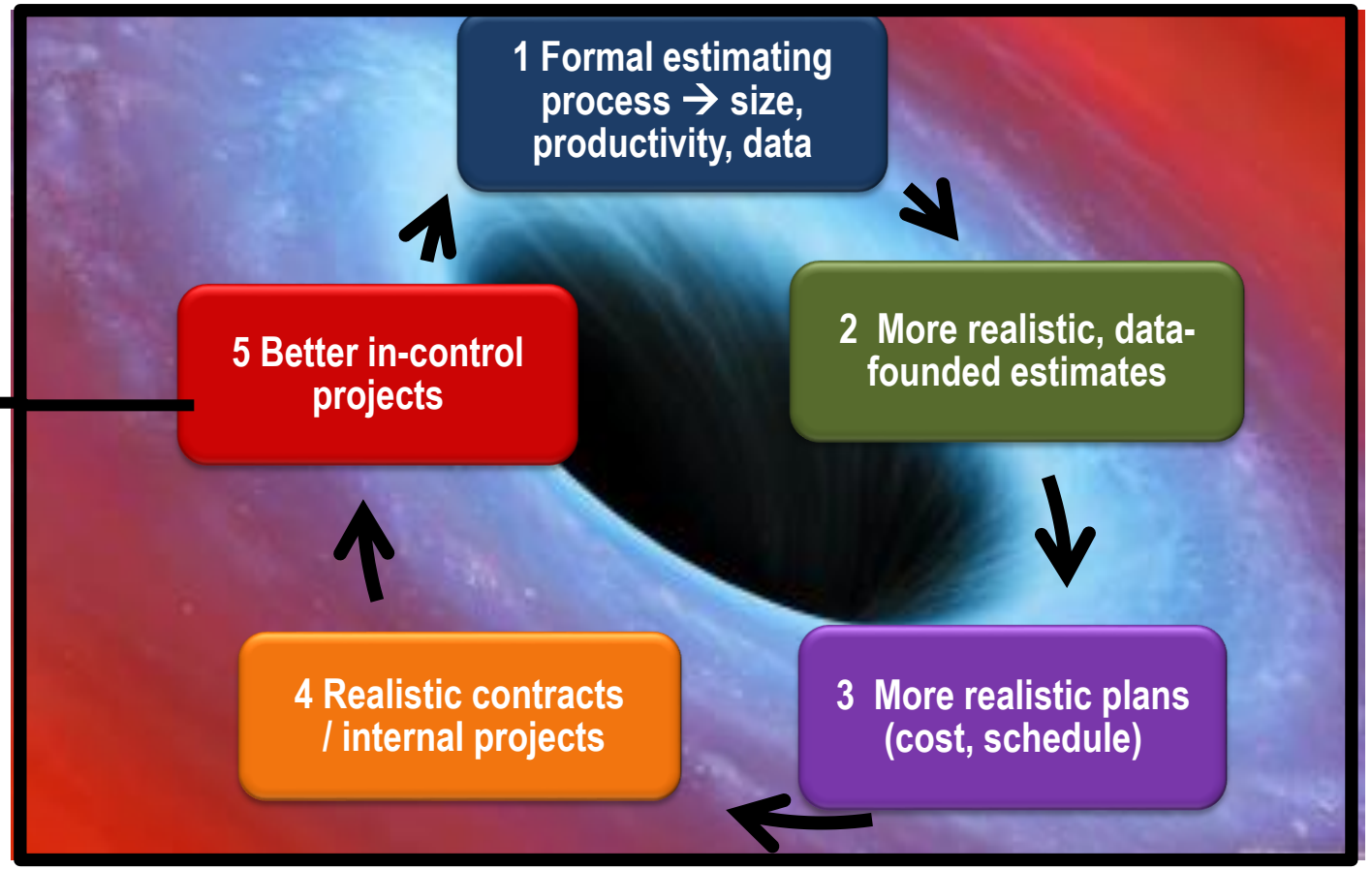
1. THE BENEFITS AND HAZARDS OF ARITIFICAL INTELLIGENCE (AI) by Capers Jones, Version 10.0, August 18, 2023

© Carol Dekkers/Quality Plus Technologies, Inc.

© Carol Dekkers/Quality Plus Technologies, Inc.

A (more) formal approach to software cost estimation can increase YOUR project outcomes

Improved project outcomes:
- Anticipated and tracked project "growth"
- Formal risk management
- Fewer surprises
- Increased potential to finish on-time and on-budget projects



© Carol Dekkers/Quality Plus Technologies, Inc.

Improve YOUR participation in the estimating process

- Software cost estimating is non-trivial
- **Software cost and schedule estimates** can be especially challenging … good data & inputs, formal approach, & cross-checks can reduce error
- Over-optimism and planning fallacy are natural tendencies
- Consider standardized software size estimates (SLOC, SFP)
- Better communication leads to better estimates
- Software cost estimation is a professional endeavor (ICEAA CEBOK-S)

© Carol Dekkers/Quality Plus Technologies, Inc.

Consider CEBoK-S as a Resource

LESSON 0: INTRODUCTION TO CURRICULUM

LESSON 1: IMPORTANCE AND MOTIVATION FOR CEBOK-S

LESSON 2: SOFTWARE DEVELOPMENT PARADIGMS

LESSON 3: CEBOK-S FIVE-STEP ESTIMATING PROCESS

LESSON 4: ESTIMATING CUSTOM SOFTWARE DEVELOPMENT

LESSON 5: SOFTWARE SUSTAINMENT

LESSON 6: ESTIMATING PROCURED SOFTWARE SOLUTIONS

LESSON X: SOFTWARE SIZE

LESSON Y: PRODUCTIVITY

LESSON Z: COMMERCIAL ESTIMATING MODELS

ICEAA International Cost Estimating and Analysis Association

Software cost estimating Body of Knowledge (CEBoK-S): materials, training and certification available now

www.iceaaonline.com

© Carol Dekkers/Quality Plus Technologies, Inc.

A Final Note…

'The software industry has the worst metrics and measurement practices of any industry in human history' – Capers Jones (2018)[1]

"Size- and data-based software estimates are the key to better project outcomes, and over time, better metrics." – Carol Dekkers, Dec 2021

1. Source: Capers Jones, **Quantifying Software – Global and Industry Perspectives**, 2018

© Carol Dekkers/Quality Plus Technologies, Inc.

Carol Dekkers
(+1) 813-816-1329
dekkers@qualityplustech.com
www.qualityplustech.com

© Carol Dekkers/Quality Plus Technologies, Inc.


PNSQC
OCTOBER 9-11 2023
AMP IT UP:
TRANSFORMING QUALITY
THANK YOU
Carol Dekkers
Software Cost Estimating –
Friend or Foe? (to agilists)
PNSQC.ORG