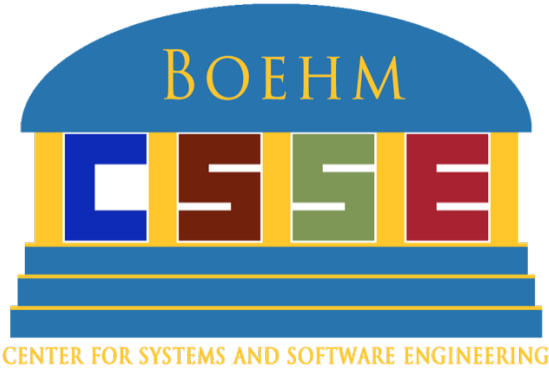


Boehm Center for Systems and Software Engineering (Boehm CSSE)

GSAW Evening Session

February 28, 2024

Agenda

- Overview of Boehm Center for Systems and Software Engineering
- COCOMO 3 Research
- COSYSMO: Present and Future
- Annual Research Review (ARR) 2024
- Interactive Panel Discussion

Who Are We?

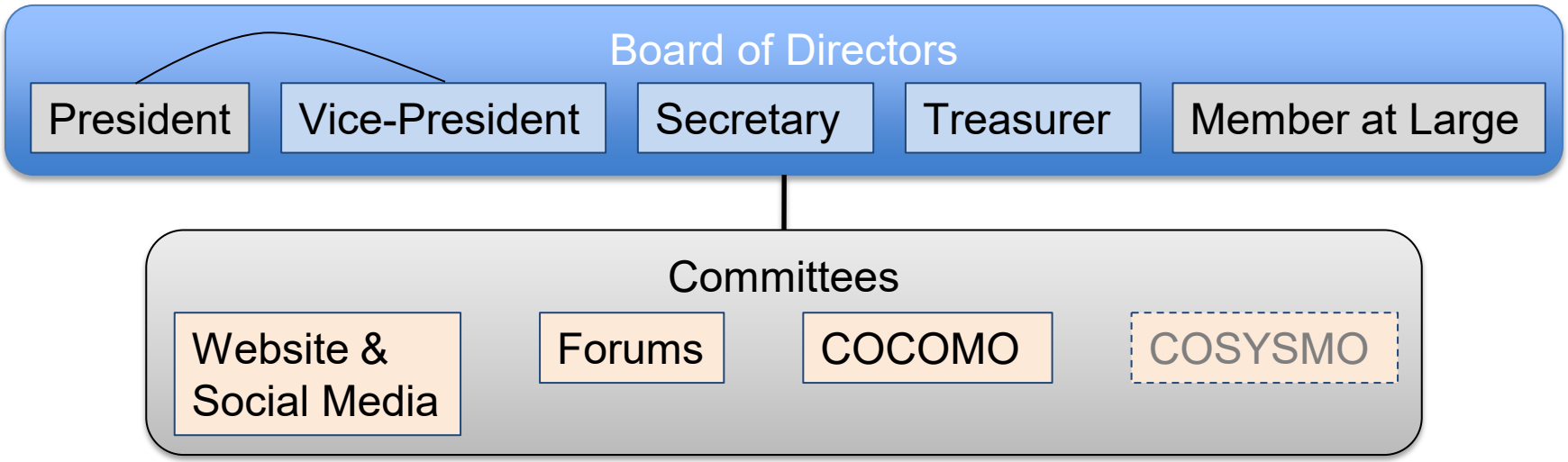
- Non-profit organization to continue Barry Boehm's pursuit for improving Systems and Software Engineering

Dedicated to pursuing the pioneering contributions of Professor Barry Boehm with an open and inclusive approach to related research

- Collaborate and promote improvements in Systems and Software Engineering:
  - Software Architecture and Quality
  - Systems and Software Economics
  - Software Tools
  - Software and Systems Processes
  - Software and Systems Management
  - Risk Management

Boehm CSSE Organization

- State of Arizona Non-Profit Corporation
- Internet presence with regular content updates (BoehmCSSE.org)
- Collaborate with other similar-minded organizations, e.g.,
  - Practical Software and System Measurement (PSM)
  - Ground Systems Architecture Workshop (GSAW)
  - Systems Modeling with Python (PyML)
- Offer memberships in Boehm CSSE



Newly Elected Board of Directors (starting January 2024)

Marilee Wheaton
President
president@boehmcsse.org

Brad Clark
Vice President
vicepresident@boehmcsse.org

Tenley Burke
Treasurer
treasurer@boehmcsse.org

Julie Sanchez
Secretary
secretary@boehmcsse.org

Accomplishments

- Incorporated in the State of Arizona
  - Board of Directors
  - Bylaws
  - Reporting requirements
- IRS 501(c)3 Non-Profit status
- Established an Internet presence with regular content updates (BoehmCSSE.org)
- Created a membership model (revenue stream)
- Established Committees to work in support of the Boehm CSSE mission
- Cooperating Organizations
- Now offer Boehm System and Software Engineering merchandise
  - Let us know if you have ideas for merchandise
- Hosted three events:
  - 2022 COCOMO Cost Forum (virtual)
  - 2023 Annual Research Review (virtual)
  - 2023 COCOMO Cost Forum with PSM (hybrid)

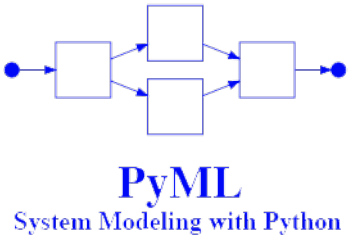
Cooperating Organizations

Homepage – right column
Images are hot-linked to Organization's website

Practical Software and System Measurement (PSM)

Ground Systems Architecture Workshop (GSAW)

Systems Modeling with Python (PyML)



- There is no cost to be a Cooperating Organization
- Please suggest other Organizations to join

Boehm CSSE Committees

- Website and Social Media
  - The Website and Social Media Committee will oversee the development and maintenance of the BoehmCSSE.org website and will promote Boehm CSSE on different social media platforms. (webmaster@boehmcsse.org)
- Forums
  - The Forums/Fora Committee will support the Boehm CSSE Board to organize bi-annual forums by managing scheduling, logistics, and coordination efforts. (forum@boehmcsse.org)
- COCOMO
  - This committee will preserve and advance Barry Boehm's COnstructive COst MOdel first published in 1981. (cocomo@boehmcsse.org)

Join a committee to support professional development and become subject matter experts (SMEs) on a topic!

Future Boehm CSSE Committees

- COSYSMO
  - This committee will preserve, and advance the COnstructive SYStems cost MOdel.
- Research Mentorship
  - This committee will give SMEs and industry professionals the opportunity to influence research to solve current, everyday challenges as well as mentor up and coming researchers.
  - Academic and industry researchers will get industry feedback for relevance and academic support for scientific rigor for FREE!
- Training and Certification
  - Become SMEs on software and systems cost estimation, process improvement, process models, risk management, and more!

Join a committee to support professional development and become subject matter experts (SMEs) on a topic!

Membership Benefits

- Free events, e.g., the Fall and Spring Forums
- Access to Forums' presentation slides and recordings
- Participation in committees, working groups, and activities
- Complete access to Boehm CSSE online artifacts and tools
- Free access to Boehm CSSE online training
- Voting rights for elections and future Boehm CSSE initiatives
- Inaugural membership recognized for the first year
- Lifetime membership recognized forever

Future Boehm CSSE Events

- ARR 2024: Apr 16 to Apr 18
- Cost Forum 2024: Nov 12 to Nov 14

Membership Fees

- Individual membership
  - Annual $50 (renew annually)
  - Lifetime $500
- Affiliate membership (with one person appointed as the representative - transferable)
  - Government $1000
  - University $1000
  - FFRDC $1000
  - Small business (under 500 people) $1000
  - Corporations $5000

Inaugural Members To Date

Boehm Family Foundation

Paula Baker
Lena Baird
Victor Basili
Jim Boehm
Linda Boehm
Romney Boehm
Ryan Boehm
Sharla Boehm
Tenley Burke
Cullen Burke
Mason Burke
Louise Fahey
Judith Johnson
Rachelle Lucyzinski
Gene Nebeker
Lisa Noxon
T.J. Noxon
Bill Roberti
David Schloss

Lifetime Members

Mauricio Aguiar
Jim Alstad
Jongmoon Baik
Pooyan Behnamghader
A Winsor Brown
Yue Chen
Brad Clark
Betsy Clark
Dan Galorath
Gary Hafen
Richard Halliger
Anandi Hira
Hoh In
Judy Kerner
Sue Koolmanojwong
Dan Ligett
Arlene Minkiewicz
Vu Nguyen
Leon Osterweil
Antony Powell
Doug Rosenberg
Julie Sanchez
Neil Siegel
Ricardo Valerdi
Bo Wang
Maribee Wheaton

Annual Members

Alain Abran
Rob Ashmore
Kim Bell
Richard Cann
Steven Cox
Babak Damadi
Sean Densford
Lonnie Franks
Patrick Hamon
Anthony Higginson
Paul Janusz
Safae Laqrichi
Chris McCauley
Vu Nguyen
Barry Papke
Jesus Rodriguez
David Seaver
Rosalind Singh
Julia Taylor
Russell Varnado
Tomeka Williams

Andrew Adams
Rick Battle
Ryan Bell
Murray Cantor
Rita Creel
Joseph Dean
Ryan Farrell
Cheryl Gray
Stephen Henry
Dan Houston
Cheryl Jones
Sariyu Marfo
Mac McDonald
William Nichols
Mauricio Pena
Esteban Sanchez
Pavel Shipillo
Dan Strickland
Kelly Timko
Urjaswala Vora
Paul Wilson

Charles Adams
Bridget Beamon
Salvatore Bruno
Kelly Cassidy
Paul Cymerman
Carol Dekkers
Robert Ferguson
Anil Gupta
Jeffrey Herrera
Robert Hunt
David Klappholz
Kevin McBride
Matt McDonald
Kenneth Nidiffer
Art Pyster
Benjamin Schumeg
Louis Silverstein
Ann Marie Stulik
Eric Topelian
Charles Wesolowski
Robin Yeman

Boehm CSSE Merchandise

(Homepage > Merchandise)

COCOMO III RESEARCH

Brad Clark
Boehm CSSE

The COCOMO III Project

COCOMO® (COnstructive COst MOdel) is the most widely used, free, open source software cost estimation model in the world.

But also

- Most widely referenced
- Most widely validated

And

- Most widely rebutted!

COCOMO III Project Scope

- COCOMO® III will produce estimates for:
  - Effort, Schedule, Cost, Defects
- COCOMO® III can be applied at various milestones in a project's lifecycle:
  - Early Estimation, Post-Architecture Estimation, Project Re-estimation
- COCOMO® III's functional vision
  - Top-level and Multiple component estimate
  - Alternative size measures
  - Analysis of alternatives
  - Analysis with Size-Effort-Schedule as independent variables
  - Support for different lifecycle processes
  - Lifecycle cost estimation
  - Legacy system transformation
  - Estimate using COCOMO® III and COSYSMO together
  - Local calibration

When is it appropriate to use COCOMO III to estimate cost & schedule?

MODEL SCOPE

Model – ICSM Common Cases

- Since 2000, a plethora of development processes have arisen
  - Non-Developmental Item such as COTS
  - Agile Development
  - Brownfield Development
- One size fits all estimation model is no longer feasible
- COCOMO III Use Cases
  - Hardware with Embedded Software Component
    - Concurrent hardware/software engineering; full lifecycle processes with milestone reviews, deployment
  - Indivisible Initial Operational Capability (IOC)
    - Determine minimum-IOC likely, conservative cost. Add deferrable software features as risk reserve
  - Hybrid Agile/Plan-Driven System
    - Full lifecycle model, encapsulated agile in low-medium criticality parts

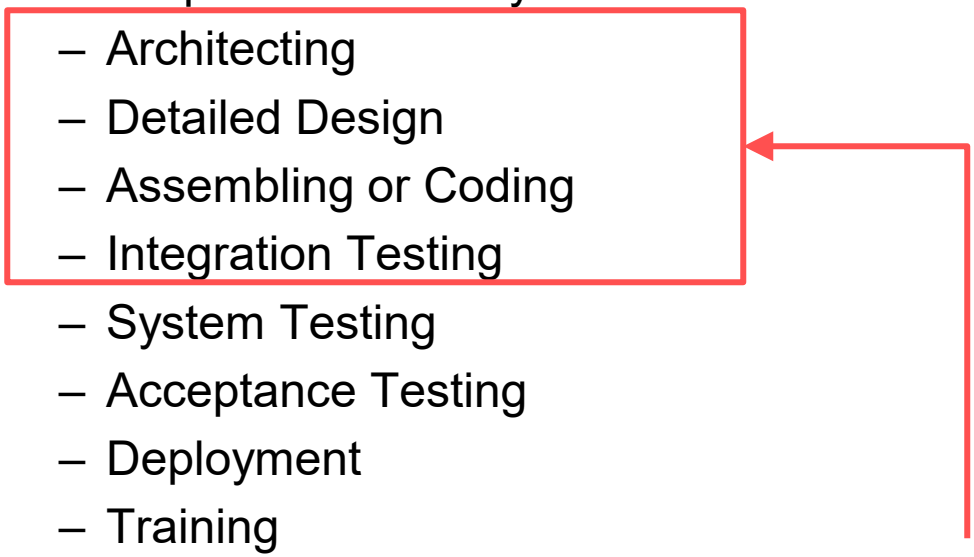
Use Cases Source: Boehm, B., Lane, J., Koolmanojwong, S. and Turner, R. "The Incremental Commitment Spiral Model", Addison-Wesley 2014, Chapter 11.

COCOMO III Modeling Focus

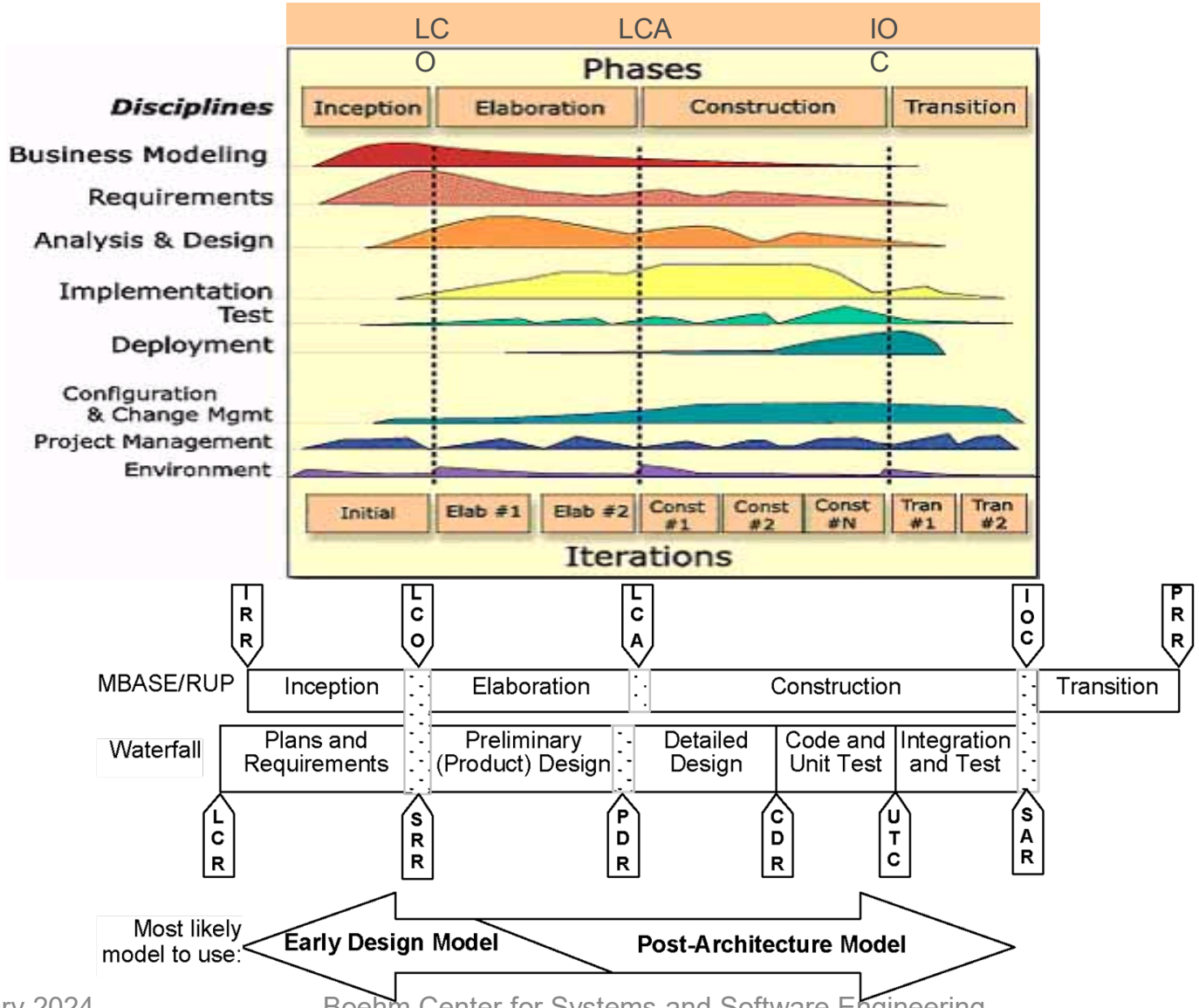
- In general, the Use Cases for COCOMO III are:
  - Plan-driven
  - Set of requirements “frozen” at the start of an iteration
  - High- and low-level design activities executed
  - Implementation followed by Integration activities
  - Ends with an acceptance activity

Model Breadth

- There are a number of different activities in software development:
  - Requirements analysis
  - Architecting
  - Detailed Design
  - Assembling or Coding
  - Integration Testing
  - System Testing
  - Acceptance Testing
  - Deployment
  - Training
- COCOMO III will cover these activities



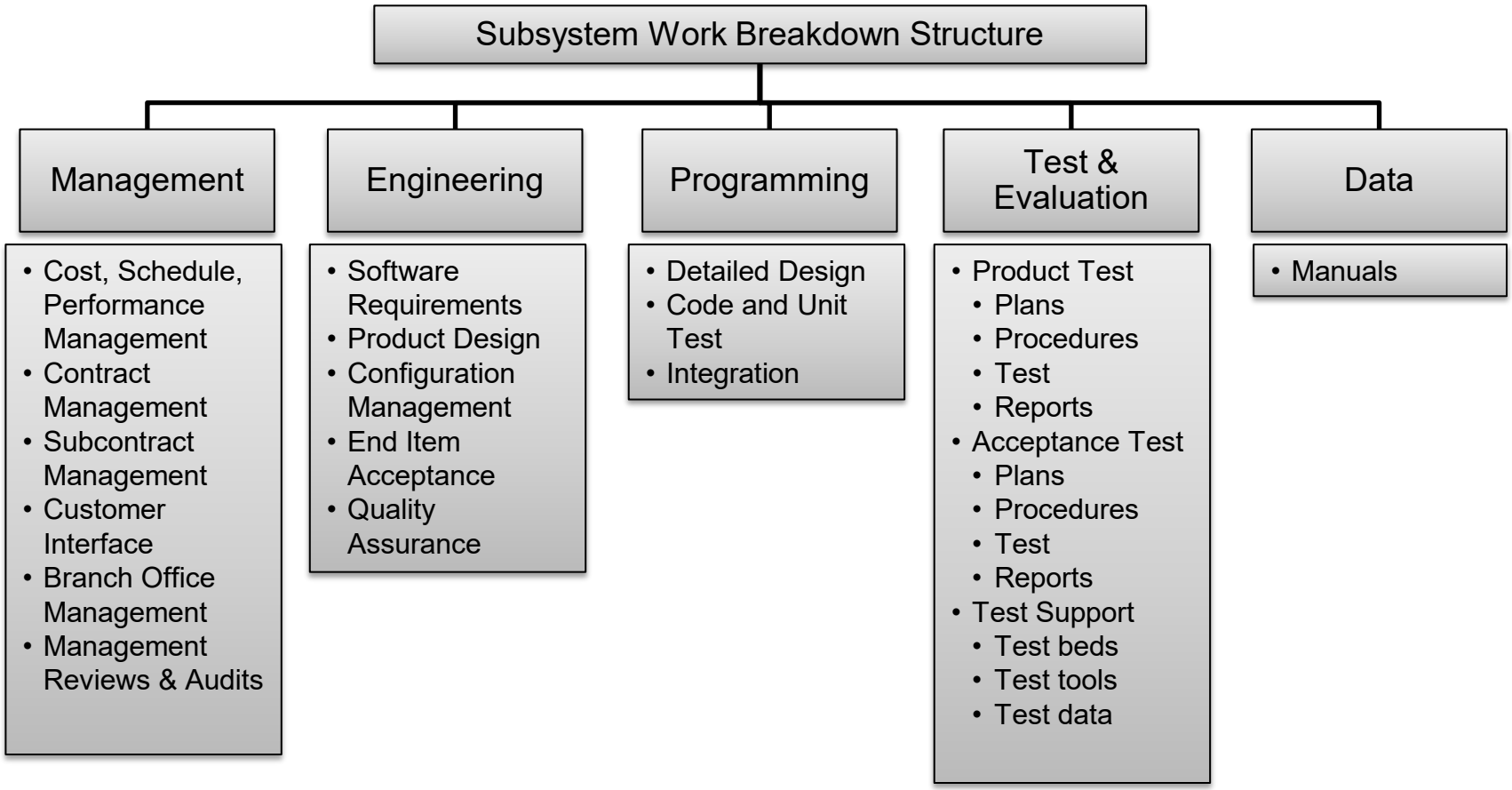
COCOMO II Model Phases



Model Depth

- Development activities include/exclude different types of work:
  - Management
  - Requirements analysis
  - Product design
  - Programming
  - Test and evaluation
  - Configuration Management / Quality Assurance
  - Documentation
- COCOMO III covers a number of work types (next slide)
  - The work covered is an indicator for whether the model is suitable for estimating a development process

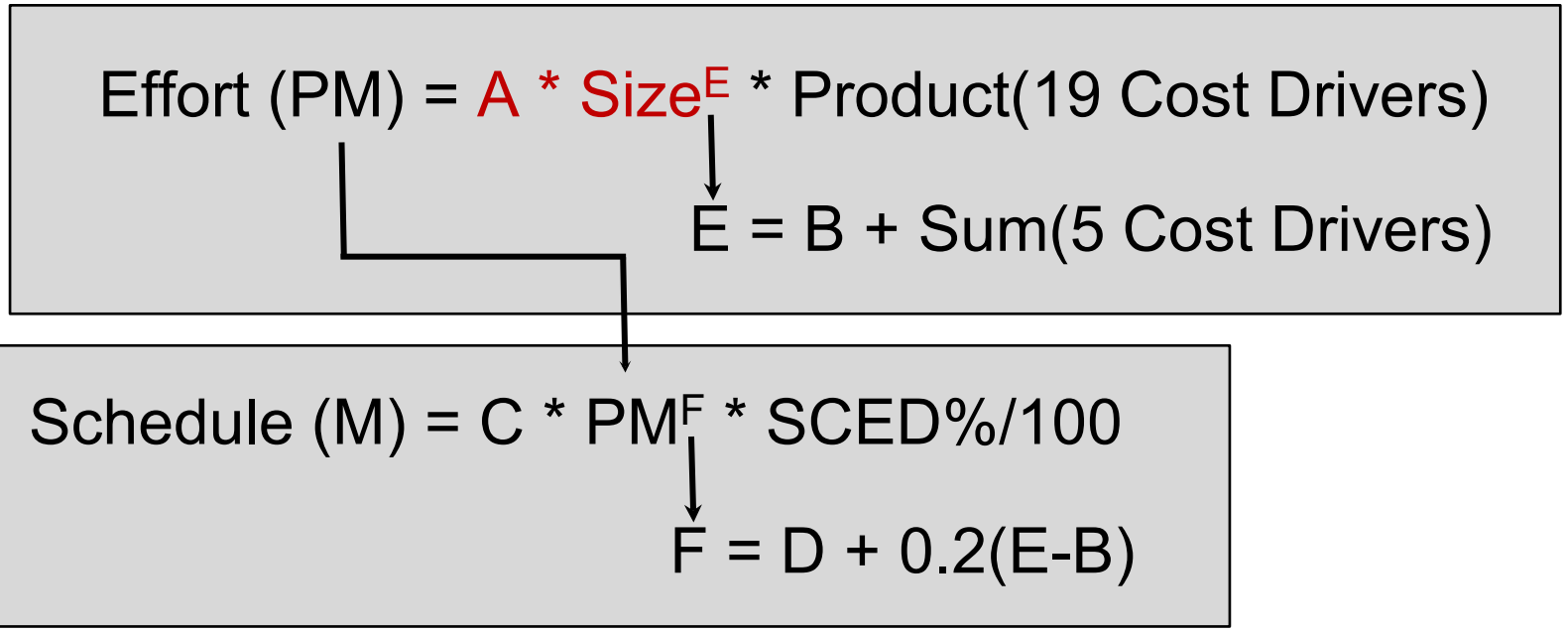
COCOMO III Depth



How are size and cost drivers used to create an estimate?

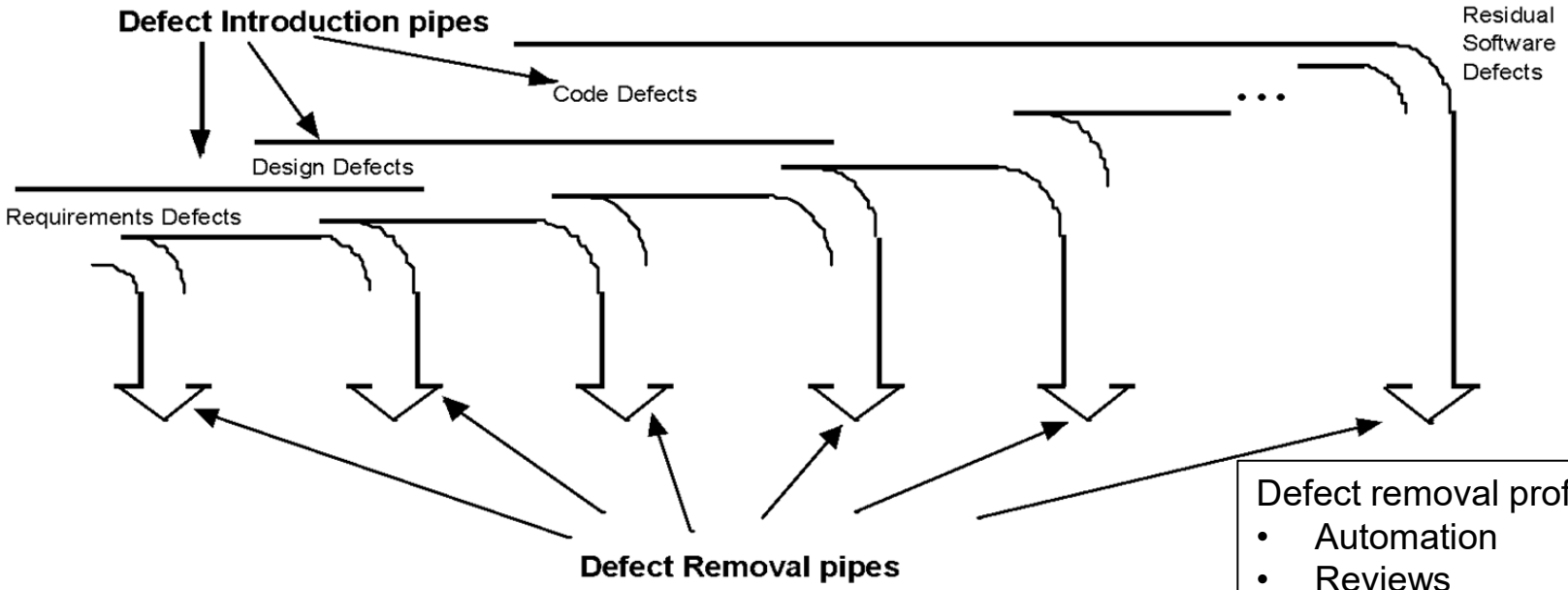
MODEL EQUATIONS

COCOMO III Effort & Schedule Estimation Model

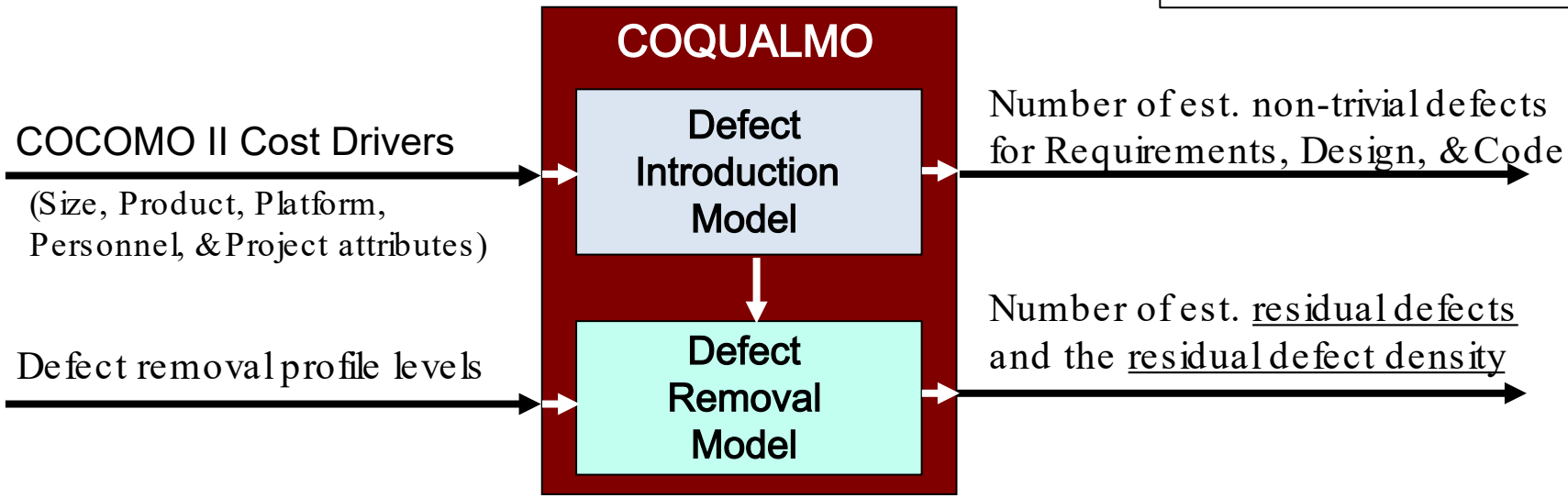


Where:
A, B, C, D are constants determined by calibration
E represents (dis)economies of scale and project-wide scale factors

COQUALMO



Defect removal profiles:
- Automation
- Reviews
- Testing



COCOMO III Defect Introduction and Removal Model

Defect Introduction (DI) = A * Size^{E} * Product(DI Drivers)

E = Initially set to 1.0

Residual Defects = C * DI * Product(1 – DRF)

DRF: Defect Removal Fraction from 3 profiles:

1. Automated Analysis
2. People Reviews
3. Execution Testing

How is the amount of work to be done estimated and how do you accommodate pre-existing artifacts?

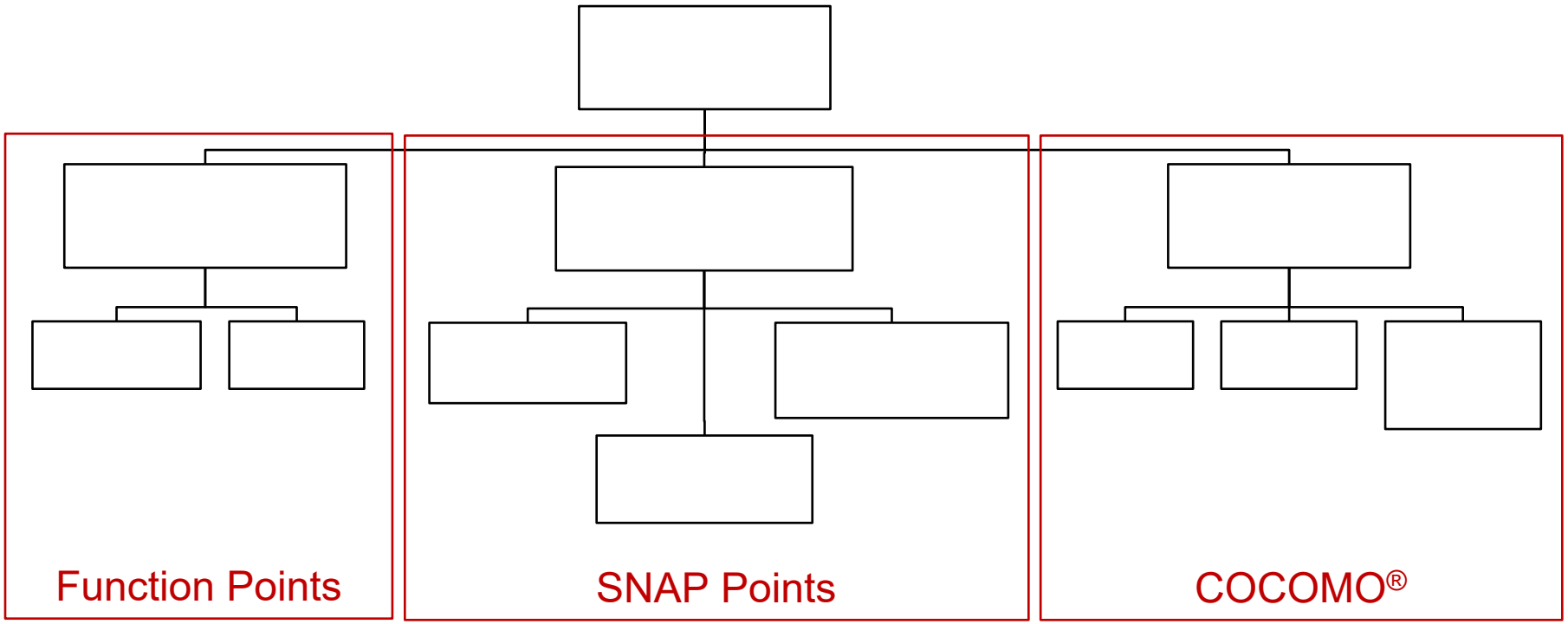
SIZE INPUTS

COCOMO III Size Inputs

- Intent is to produce an estimation model that takes different software size inputs directly
  - Current software size other than source lines of code (SLOC) is first converted to SLOC and used as "equivalent" size in the model
  - Dependent on the data collected for calibration
    - Software Requirements
    - Function Point
    - SNAP Points
    - Fast Function Points
    - COSMIC Points
    - Object / Application Points
    - Feature Points
    - Use Case Points
    - Story Points (*Agile Development*)
- How are Non-Functional Requirements handled?
- How is reused "Functionality" handled?

Motivation

Can COCOMO®, FPA, and SNAP be a comprehensive approach to estimate effort, considering all aspects of the software project?



Seems like a good marriage, right?

SNAP: Software Non-Functional Assessment Process
Slide source: Talmon Ben-Cnaan

(FP + 0.16SP) Power Model

$$PM = A * (FP + 0.16 * SP)^B$$

Formula = log(Effort) ~ log(FP16SP)

Residuals:

Min	1Q	Median	3Q	Max
-1.73293	-0.41761	-0.09431	0.40783	2.49941

Coefficients:

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	-2.59420	0.14434	-17.97	<2e-16 ***
log(FP16SP)	1.01619	0.04776	21.28	<2e-16 ***

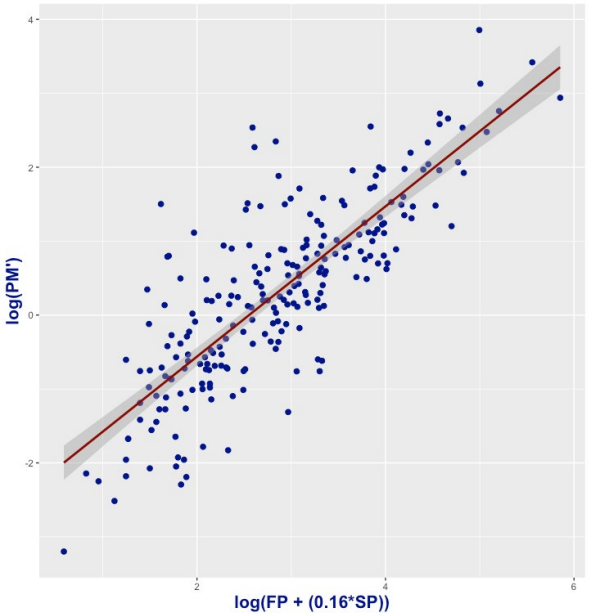
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

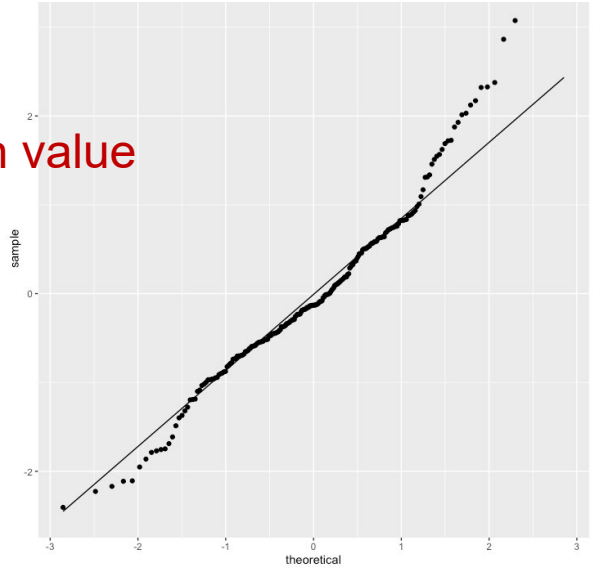
Residual standard error: 0.7216 on 229 degrees of freedom

Multiple R-squared: 0.6641, Adjusted R-squared: 0.6626 <= High value

F-statistic: 452.7 on 1 and 229 DF, p-value: < 2.2e-16

- Model:
  PM' = 0.075 * (FP + 0.16*SP)$^{1.02}$
- All coefficients are significant at a 95% confidence level, i.e., alpha = 0.05
- **1 to 1/6th supported by anecdotal evidence**





SLOC Reused Functionality -1

- Currently COCOMO III uses the reuse model from COCOMO II
  - Model is based on source lines of code

- AAF: Adaptation Adjustment Factor
  - DM: percent design modified
  - CM: percent of code and unit test modified
  - IM: percent of integration and test modified

- AAM: Adaptation Adjustment Multiplier
  - SU: Software Understanding
  - UNFM: Programmer Unfamiliarity
  - AA: Assessment and Assimilation

$$AAF = (0.4 \times DM) + (0.3 \times CM) + (0.3 \times IM)$$

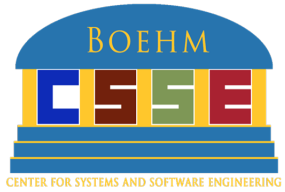
$$AAM = \begin{cases} \dfrac{[AA + AAF(1 + (0.02 \times SU \times UNFM))]}{100}, & \text{for } AAF \leq 50 \\ \dfrac{[AA + AAF + (SU \times UNFM)]}{100}, & \text{for } AAF > 50 \end{cases}$$

$$\text{Equivalent KSLOC} = \text{Adapted KSLOC} \cdot AAM$$

SLOC Reused Functionality -2

- Instead of DM, CM, and IM, maybe use a common approach based on percentages of different code types:
  - New SLOC: 100% of new code
  - Modified SLOC: 80% of modified code
  - Reused SLOC: 15% of reused code (unmodified code)
  - Auto-Generated SLOC: 30% of auto-gen code

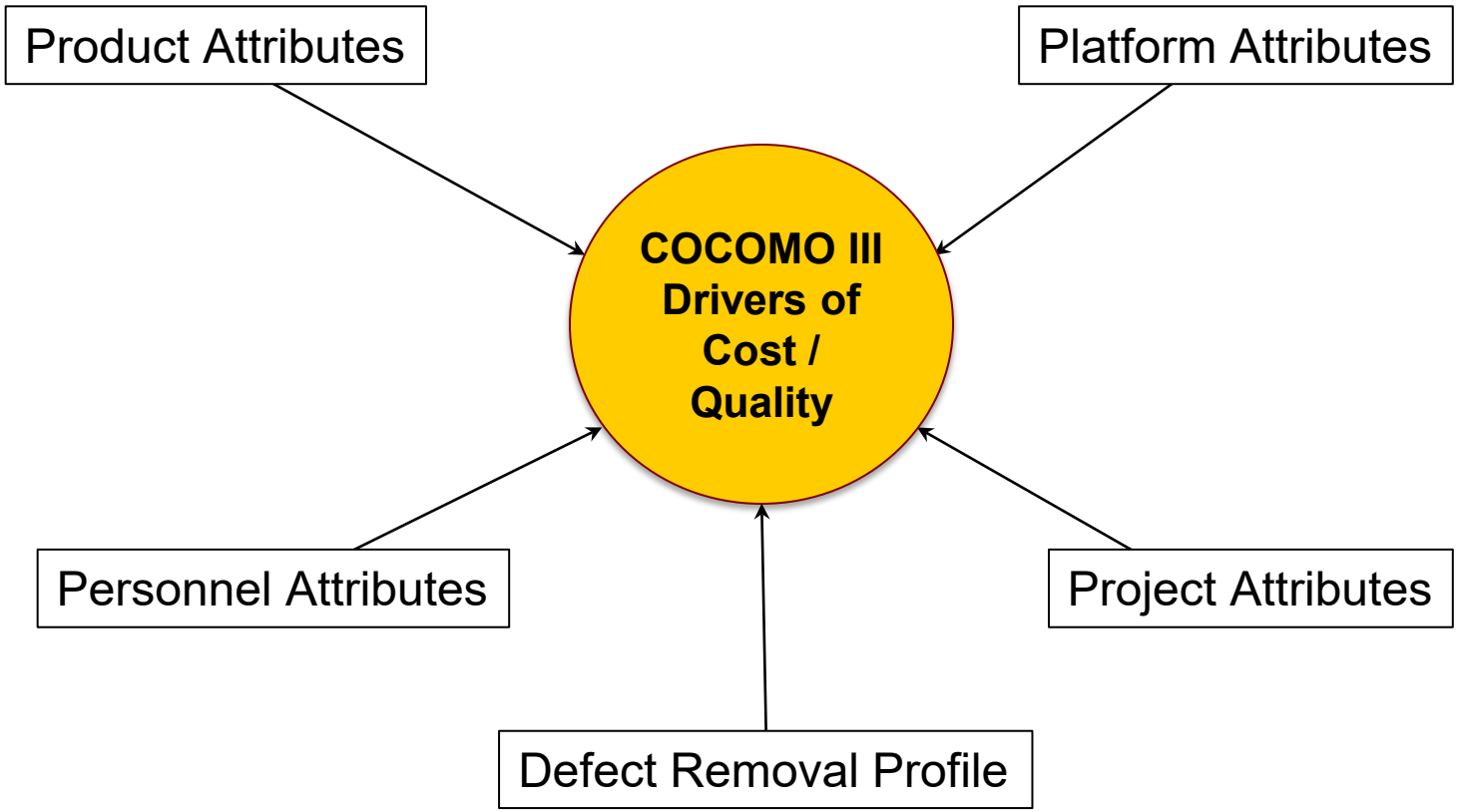
How do we handle Functional Size Measure **reuse**?



What are the drivers of cost?

COST DRIVERS INPUTS

COCOMO III Cost & Quality Drivers



New Feature: Application Domain Types

Real-Time
- Sensor Control and Signal Processing
- Vehicle Control
- Vehicle Payload
- Real Time Embedded
- Mission Processing

Engineering
- Systems Software
- Automation and Process Control
- Simulation Modeling

Automated Information Systems
- Mission Planning
- Training
- Test
- Data Processing

Selecting an Application Domain “pre-sets” model drivers

COCOMO III Cost Drivers -1

- Product Attributes
  - Impact of Software Failure (FAIL) (formerly Required Software Reliability)
  - Product Complexity (CPLX)
  - Developed for Reusability (RUSE)
  - Required Software Security (SECU)
  - Dropped:
    - Documentation Match to Lifecycle Needs
    - Database Size
- Platform Attributes
  - Platform Constraints (PLAT)
    - Combined Execution and Storage Constraints
  - Platform Volatility (PVOL)

COCOMO III Cost Drivers -2

- Personnel Attributes
  - Analyst Capability (ACAP)
  - Programmer Capability (PCAP)
  - Personnel Continuity (PCON)
  - Applications Experience (APEX)
  - Language and Tool Experience (LTEX)
  - Platform Experience (PLEX)

COCOMO III Cost Drivers -3

- Project Attributes
  - Precedentedness (PREC)
  - Development Flexibility (FLEX)
  - Risk/Opportunity Management (RISK)
  - Software Architecture Understanding (ARCH)
  - Stakeholder Team Cohesion (TEAM)
  - Process Capability & Usage (PCUS) (Formerly PMAT)
  - Use of Software Tools (TOOL)
  - Multisite Development (SITE)

Function Point Impact on COCOMO® III Cost Driver Values*

Five COCOMO III Cost Driver Effort Multiplier value comparisons

Cost Driver	Size Input	Very Low	Low	Nominal	High	Very High	Extra High	Productivity Range
RELY	SLOC	0.82	0.92	1.00	1.10	1.26		1.54
RELY	FP	0.76	0.89	1.00	1.14	1.38		1.82
CPLX	SLOC	0.73	0.87	1.00	1.17	1.34	1.74	2.38
CPLX	FP	0.75	0.88	1.00	1.16	1.31	1.66	2.21
PLAT	SLOC			1.00	1.08	1.23	1.54	1.54
PLAT	FP			1.00	1.04	1.12	1.28	1.28
PVOL	SLOC		0.87	1.00	1.15	1.30		1.49
PVOL	FP		0.92	1.00	1.09	1.17		1.27
TOOL	SLOC	1.17	1.09	1.00	0.90	0.78		1.50
TOOL	FP	1.41	1.21	1.00	0.80	0.58		2.43

$$Effort_{Total} = [A \times Size^{B}] \times \prod Cost\ Drivers$$

Depending on the size input, the Effort Multiplier values are different for each Cost Driver

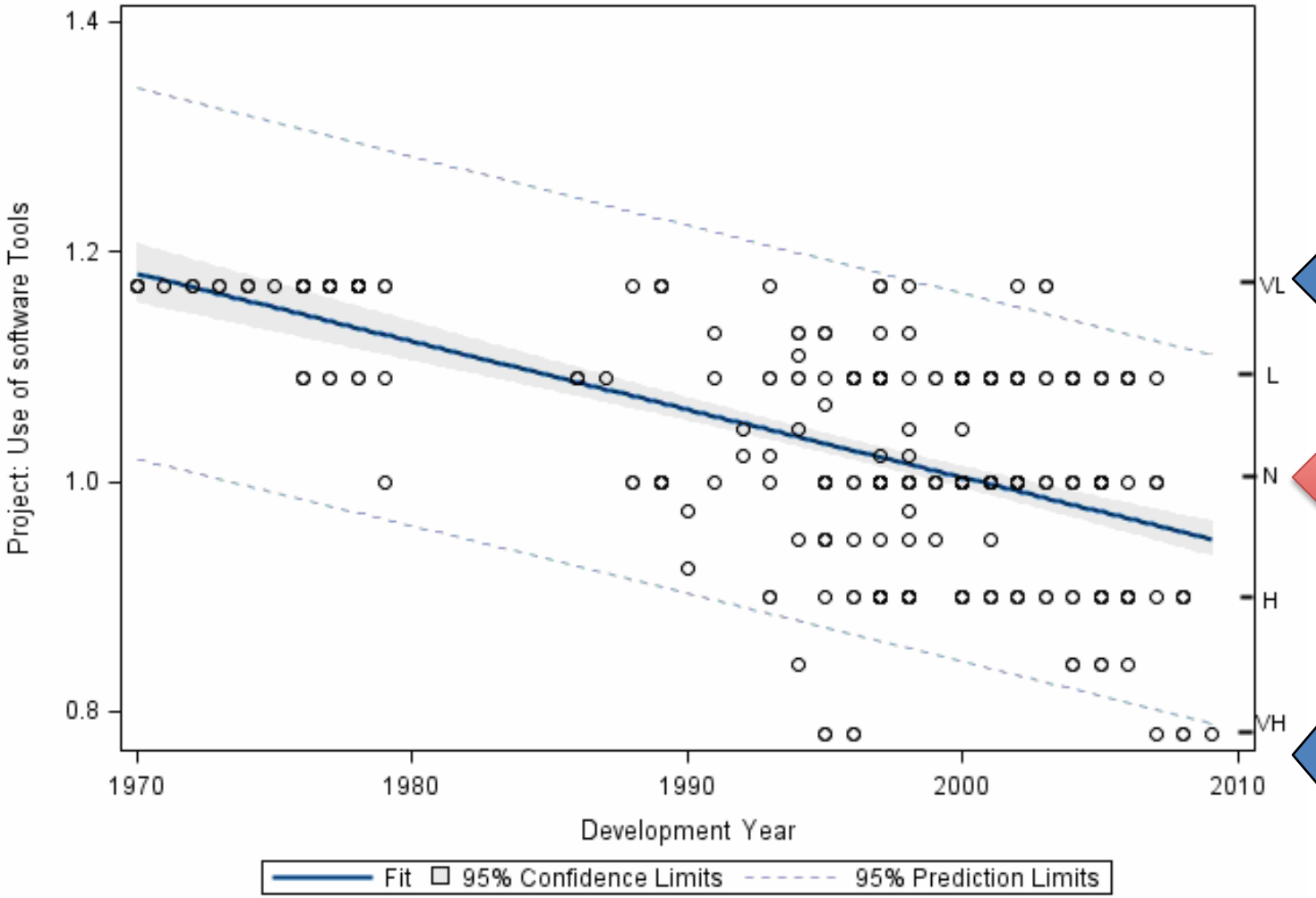
* Source: Venson, Elaine, "The Effects of Required Security on Software Development Effort," University of Southern California, PhD Dissertation, 2021, Table 5.12.

Impact of Productivity Trends

Kendall's Rank Correlation Coefficients between the Completion Year and COCOMO II Cost Drivers (sorted by degrees of correlation)

Cost driver Kendall's	τ	p-value
TOOL Use of Software Tools	-0.37	2.20E-16
PMAT Process Maturity (PCUS)	-0.30	1.22E-13
STOR Main Storage Constraint	-0.29	1.31E-11
TIME Execution Time Constraint	-0.26	6.62E-10
PLEX Platform Experience	-0.17	1.98E-05
PVOL Platform Volatility	-0.18	2.04E-05
APEX Applications Experience	0.17	4.88E-05
LTEX Language and Tool Experience	-0.15	2.84E-04
~~DATA Database Size~~	0.13	1.81E-03
RELY Required Software Reliability	-0.10	1.42E-02
CPLX Product Complexity	-0.10	1.58E-02
PREC Precedentedness of Application	-0.09	2.13E-02
ACAP Analyst Capability	0.08	4.87E-02

Use of Software Tools (TOOL)



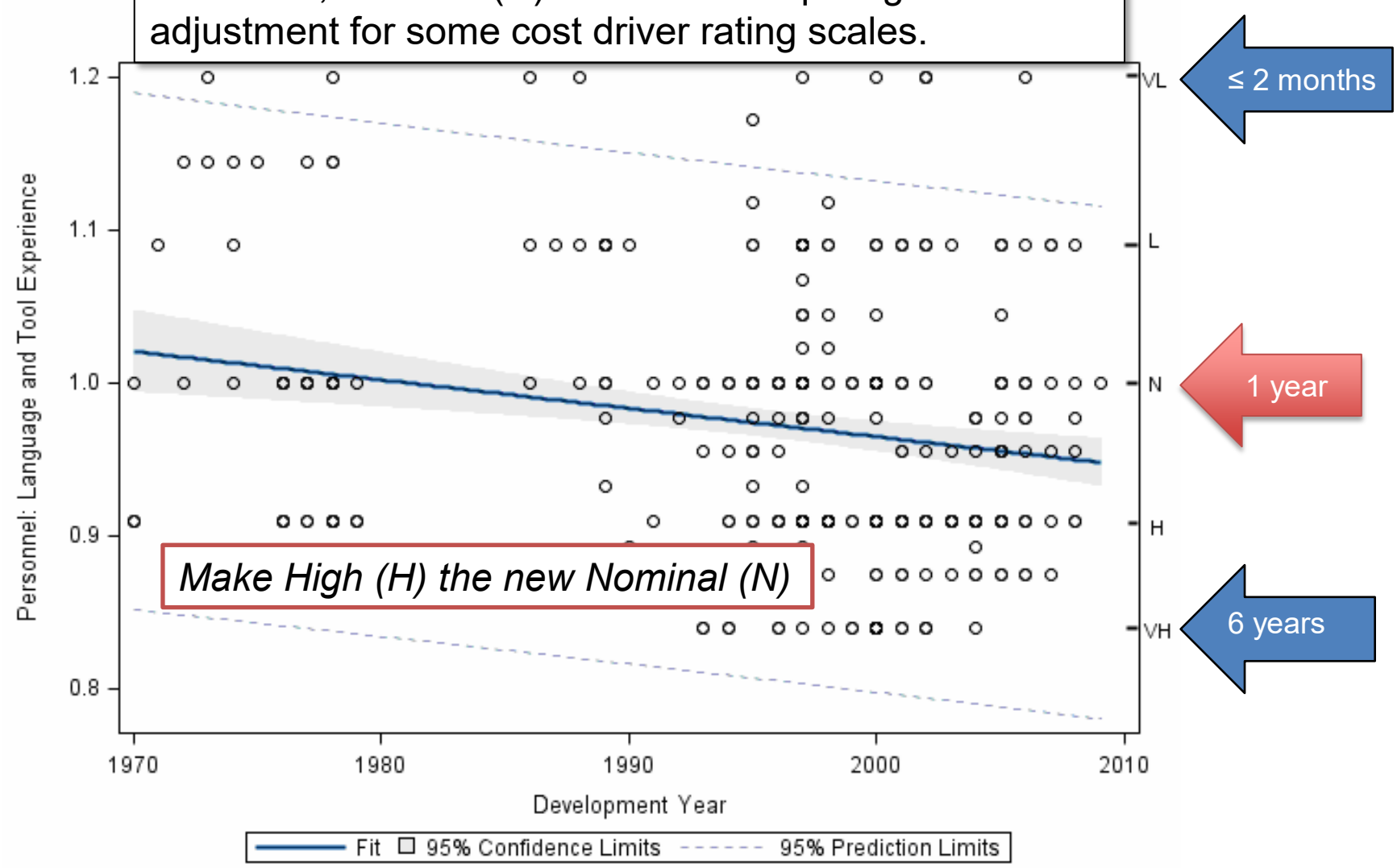
Edit, code, debug

Basic lifecycle tools, moderately integrated

Strong, mature, proactive lifecycle tools, well integrated with processes, methods, reuse

Language and Tool Experience (LTEX)

Over time, Nominal (N) has shifted requiring a future adjustment for some cost driver rating scales.



LTEX Rosetta Stone

This is a measure of the level of programming language and software tool experience of the project team developing the software system or subsystem. When rating this driver, consider the volatility of the development tools.

COCOMO II

LTEX Descriptors:	<= 2 months	6 months	1 year	3 years	6 year	
Rating Levels	Very Low	Low	Nominal	High	Very High	Extra High
Effort Multipliers	1.2	1.09	1	0.91	0.84	

Decrement COCOMO II rating by a level.

COCOMO III

LTEX Descriptors:	<= 6 months	1 year	3 years	6 year	6+ years	
Rating Levels	Very Low	Low	Nominal	High	Very High	Extra High
Effort Multipliers	1.2	1.09	1	0.91	0.84	

Other Cost Drivers

Given the scope of the model:

- Are there additional drivers of cost and effort?
  - AI Assistance?
- Are there some cost drivers that could be combined or eliminated?
  - Analyst Capability and Programmer Capability?

COCOMO Committee

INVITATION

COCOMO Committee -1

- Scope
  - This committee will focus on completing the COCOMO III project (the update to COCOMO II). There is a family of models that surround COCOMO II, and those models will be made available on the Boehm CSSE website but will not maintain them. The public will be encouraged to evolve those family models and return their version back to Boehm CSSE.
  - The committee will not produce any training related to COCOMO or the COCOMO family of models other than producing manuals describing the model operation and use.

Wanna be part of something big?

- COCOMO is the most widely used software cost estimation model in the world
  - Model will be open and free for anyone to use
  - Past models have been commercialized
- It has been 24 years since the model has been updated and calibrated to new Software Engineering data
- What we are looking for:
  - Your ideas on how the new model should be used and new input parameters to estimate software engineering development costs
  - Your chance to influence the new COCOMO III model

COSYSMO: PRESENT & FUTURE

Jim Alstad
Boehm CSSE

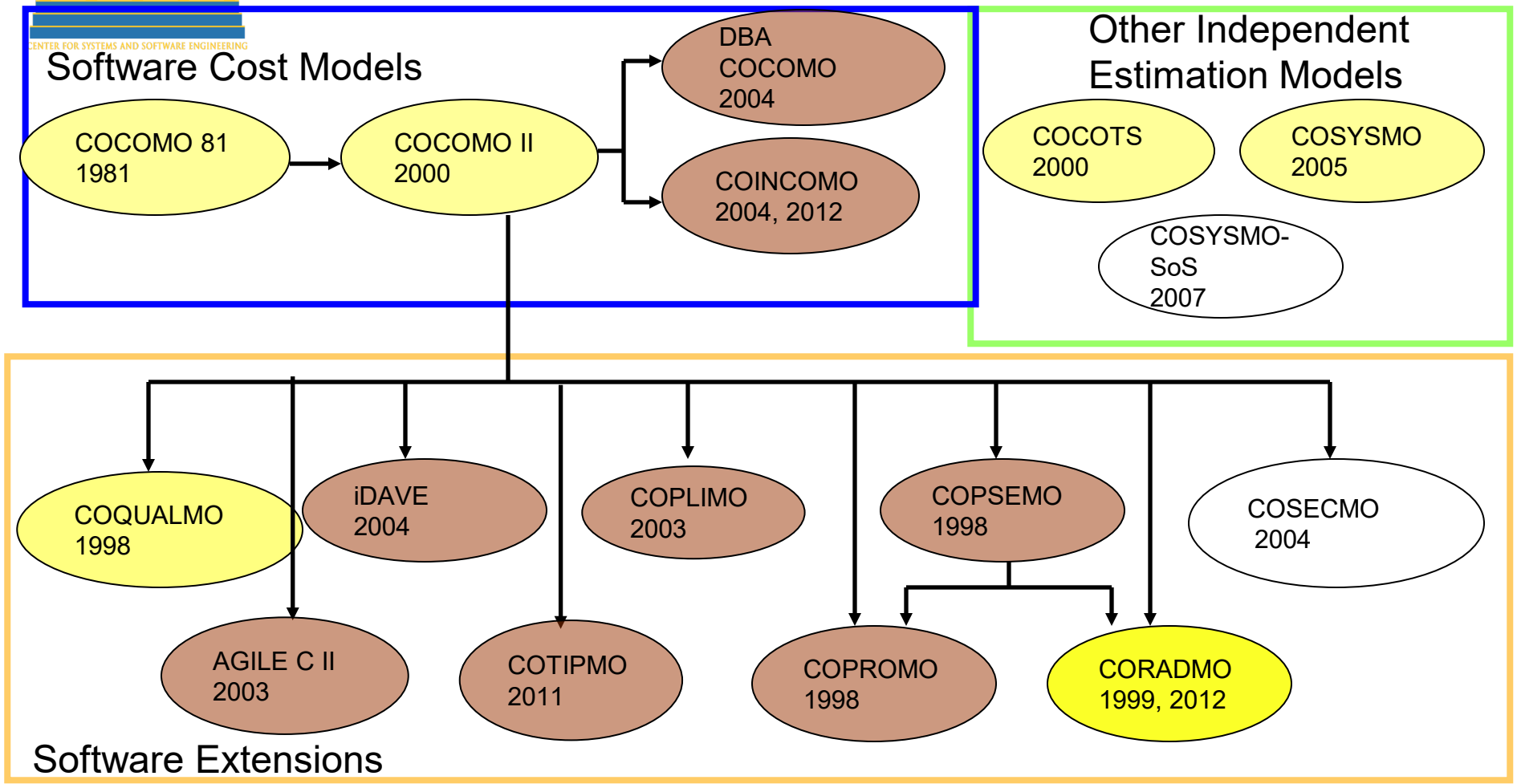
COSYSMO: Present & Future

- Introduction to COSYSMO
- Recent work toward a COSYSMO 3.0 Schedule Model
- Questionnaire

Introduction to COSYSMO

- COSYSMO History
- The Estimating Model
  - Top-Level Model
  - Size Model
  - eReq Submodel
  - Reuse (2 slides)
  - Exponent Model
  - Cost Drivers
- These slides are taken from my April 3, 2019 presentation at CSER 2019 “Development of COSYSMO 3.0: An Extended, Unified Cost Estimating Model for Systems Engineering”

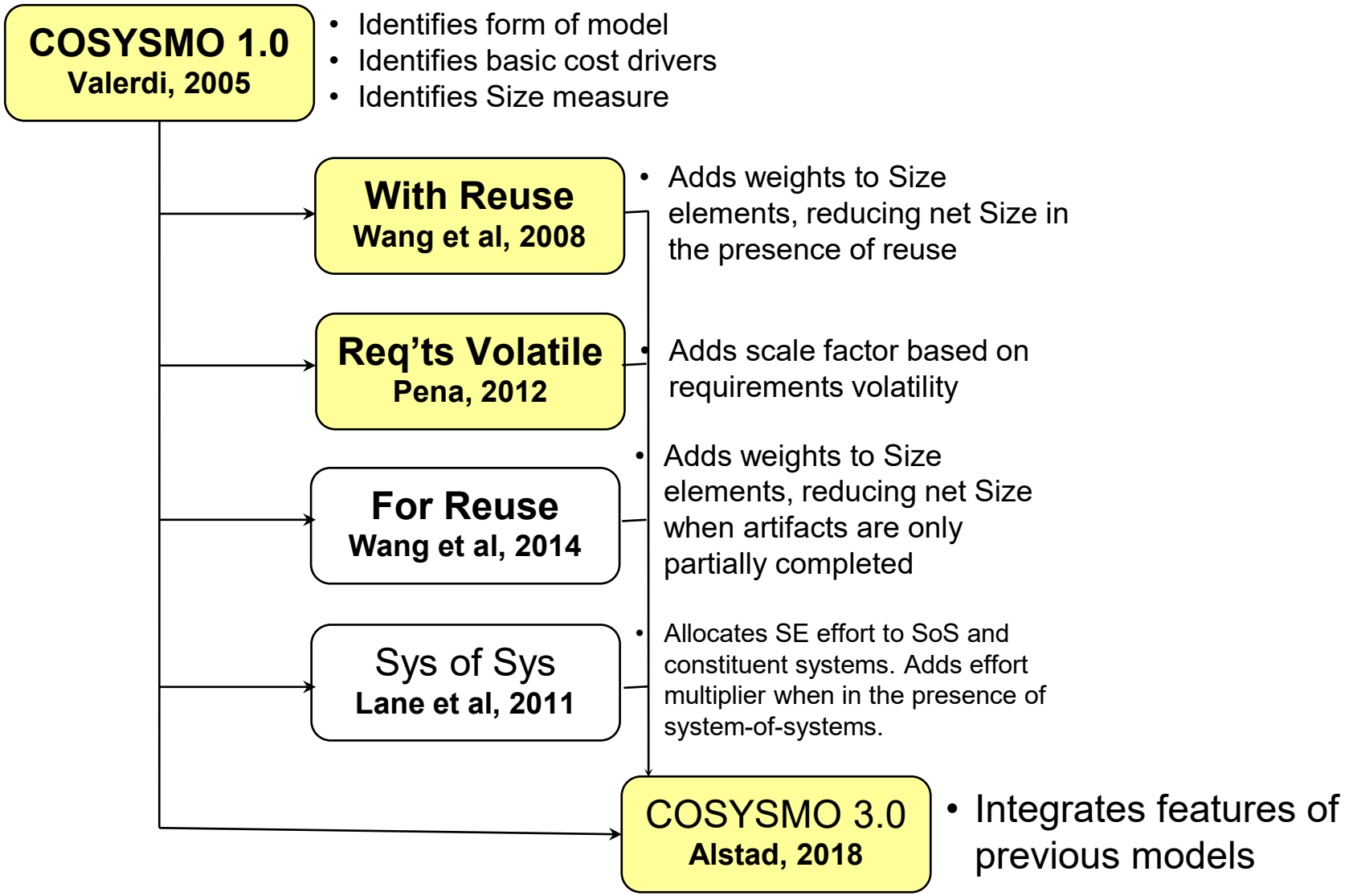
COCOMO Family of Cost Models



Legend:
Model has been calibrated with historical project data and expert (Delphi) data
Model is derived from COCOMO II
Model has been calibrated with expert (Delphi) data

Dates indicate the time that the first paper was published for the model

History of COSYSMO Models



COSYSMO 3.0 Top-Level Model

$$PH = A \cdot (AdjSize)^{E} \cdot \prod_{j=1}^{13} EM_j$$

Elements of the COSYSMO 3.0 model:

- Calibration parameter A
- Adjusted Size model
  - eReq submodel, where 4 products contribute to size
  - Reuse submodel
- Exponent (E) model
  - Accounts for diseconomy of scale
  - Constant and 3 scale factors
- Effort multipliers EM
  - 13 cost drivers

COSYSMO 3.0 Size Model

$$AdjSize_{C3} = \sum_{SizeDrivers} eReq(Type(SD), Difficulty(SD)) \times PartialDevFactor(AL_{Start}(SD), AL_{End}(SD), RType(SD))$$

- ***SizeDriver*** is one of the system engineering products that determines size in the COSYSMO family (per [2]). Any product of these types is included:
  - System requirement
  - System interface
  - System algorithm
  - Operational scenario
- There are two submodels:
  - Equivalent nominal requirements ("eReq")
    - Raw size
  - Partial development
    - Adjusts size for reuse

Size Model – eReq Submodel

- The eReq submodel is unchanged from [2].
- The submodel computes the size of a SizeDriver, in units of eReq ("equivalent nominal requirements")
- Each SizeDriver is evaluated as being easy, nominal, or difficult.
- The following table contains conversion factors for the conversion of a SizeDriver to a number of eReq:

Size Driver Type	Easy	Nominal	Difficult
System Requirement	0.5	1.0	4.5
System Interface	1.9	4.0	9.0
System Algorithm	1.9	3.8	9.8
Operational Scenario	6.4	13.6	26.3

How Reuse Is Addressed

Reuse operates in two directions [1]:

- Development with reuse (DWR): previously developed artifacts are reused on the current project
  - Addressed completely by the DWR partial development model
- Development for reuse (DFR): the current project is creating artifacts to be reused on other projects
  - One aspect of DFR development is that DFR costs more than ordinary development
    - Addressed by the DFR cost driver (covered there)
  - Another aspect of DFR is that the artifacts may be only partially completed, as during an IR&D project
    - Addressed by the DFR partial development model

Size Model – Partial Development Submodel

- (Concepts here are simplified a little)
- The basic DWR concept:
  - If a reused SizeDriver is being brought in, that saves effort, and so we adjust the size by multiplying the raw size by a PartialDevFactor less than 1.
  - The value of PartialDevFactor is based on the maturity of the reused SizeDriver, and is looked up in a table [24].
    - How fully developed was the SizeDriver?
  - If there is no reuse for this SizeDriver, then PartialDevFactor = 1 (no adjustment).

DWR Activity Level:	New	Design Modified	Design Im-plemented	Adapted for Integration	Adopted for Integration	Managed
DWR % for this AL through end	100.00%	83.00%	70.13%	56.88%	37.82%	17.50%

- The basic development-for-reuse (DFR) concept is analogous:
  - A product to be reused may be not be taken through the full development cycle (e.g., an IR&D project)

DFR Activity Level:	Conceptualized for Reuse	N/A	Designed for Reuse	Constructed for Reuse	N/A	Validated for Reuse
DFR % from start through this AL	31.96%		54.60%	78.06%		90.69%

COSYSMO 3.0 Exponent Model

- Exponent model is expanded from Peña [4, 9]

$$E = E_{Base} + SF_{ROR} + SF_{PC} + SF_{RV}$$

Where:

- E_{Base} = A minimum exponent for diseconomy of scale
- SF = scale factor
- *ROR* = Risk/Opportunity Resolution
- *PC* = Process Capability
- *RV* = Requirements Volatility

The effect of a large exponent is more pronounced on bigger projects

Cost Drivers

- **Here are the 13 COSYSMO 3.0 cost drivers:**

	Driver Name	Data Item
UNDR	CONOPS & requirements understanding	Subjective assessment of the CONOPS & the system requirements
UNDR	Architecture understanding	Subjective assessment of the system architecture
UNDR	Stakeholder team cohesion	Subjective assessment of all stakeholders
CMPX	Level of service requirements	Subjective difficulty of satisfying the key performance parameters
CMPX	Technology risk	Maturity, readiness, and obsolescence of technology
CMPX	# of Recursive levels in the design	Number of applicable levels of the Work Breakdown Structure
CMPX	Development for reuse	Is this project developing artifacts for later reuse?
OPRN	# and Diversity of installations/platforms	Sites, installations, operating environment, and diverse platforms
OPRN	Migration complexity	Influence of legacy system (if applicable)
PERS	Personnel/team capability	Subjective assessment of the team's intellectual capability
PERS	Personnel experience/continuity	Subjective assessment of staff consistency
ENVR	Multisite coordination	Location of stakeholders and coordination barriers
ENVR	Tool support	Subjective assessment of SE tools

- **Relative to COSYSMO 1.0 cost drivers, this model:**
  - **Drops Documentation**
  - **Adds Development for Reuse**
  - **Changes Process Capability to scale factor**

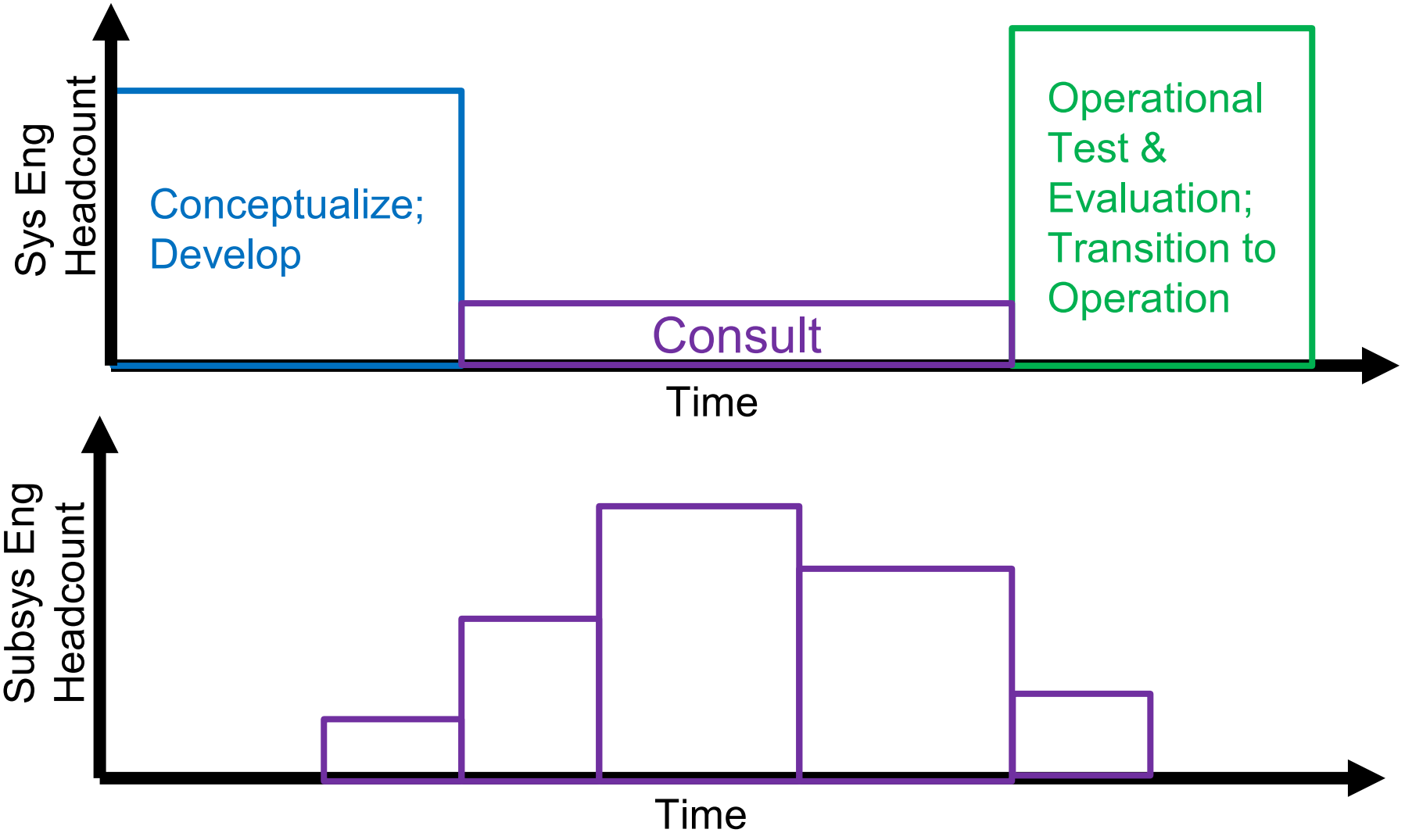
Recent Work toward a COSYSMO 3.0 Schedule Model

- Unlike COCOMO, COSYSMO presently doesn't have a schedule estimation model
  - This presentation is a step toward formulating such a model

- Is there a "Consult" phase?
  - My hypothesis about a Consult Phase
  - Systems Engineering vs Subsystem Engineering
  - Specific issues

- These slides are adapted from my November 15, 2023 presentation at the Boehm COCOMO Forum "Towards Estimating Schedule for COSYSMO: A Workshop"

My Hypothesis about a "Consult" Phase



Resulting Issues

Issues under this hypothesis:

- What duration is needed for Conceptualize + Develop? For OT&E + Transition? (These can be absolute, or %ages.) Alternatively, what average headcount?
- How should effort be divided among the three phases?
- What headcount is needed for Consulting? Zero?

Questionnaire

- I am handing out a questionnaire about schedule estimating for COSYSMO in the hope that you will fill it out and turn it in

Annual Research Review 2024

- Online: **April 16-18, 2024**
- https://boehmcsse.org/events-2/annual-research-review-2024/
- Topics for 2024 ARR:
  - Generative AI
  - Data analytics
  - Digital Engineering
  - AI, Autonomous Systems
  - Systems engineering
  - Systems Thinking as applied to business / organizational processes, particularly as related to technology and innovation
  - Topics related to quality of systems
  - Enterprise architecture, software processes
  - Software intensive systems
  - Software maintenance and quality
  - Innovative approaches to Cost Engineering
  - Moral and Ethical Boundaries of AI
  - Cybersecurity
  - Agile Processes / Agile Lessons Learned
  - DevOps
- Call For Abstracts: **Tuesday, March 5**
  - Send to abstracts@boehmcsse.org

Interactive Panel Discussion

- Identification of related areas of research
- Research gaps that are needed for industry
- Future topics for Boehm CSSE Forums

Questions?
Comments?

Marliee Wheaton
President@boehmcsse.org

Brad Clark
cocomo@boehmcsse.org

Jim Alstad
cosysmo@boehmcsse.org