



**UNIVERSITY OF
SOUTHERN CALIFORNIA**

COCOTS Tutorial

Chris Abts USC-CSE

Dr. Betsy Clark SMI

15th International COCOMO Forum

24 October 2000



USC Center for Software Engineering

Points of Contact at USC-CSE in Los Angeles

Mr. Chris Abts (primary graduate researcher).....(213) 740-6470
Ms. Ladonna Pierce (CSE Office Administrator).....(213) 740-5703
Dr. Barry W. Boehm (CSE Director).....(213) 740-8163
USC Center for Software Engineering FAX line.....(213) 740-4927
COCOTS E-Mail.....*cots-info@sunset.usc.edu*
World Wide Web page.....*http://sunset.usc.edu/COCOTS/cocots.html*

Additional Contact at Software Metrics, Inc. in Virginia (near D.C.)

Dr. Elizabeth (Betsy) Clark.....(703) 754-0115
FAX line.....(703) 754-0115
E-Mail.....*bkbailey@erols.com*



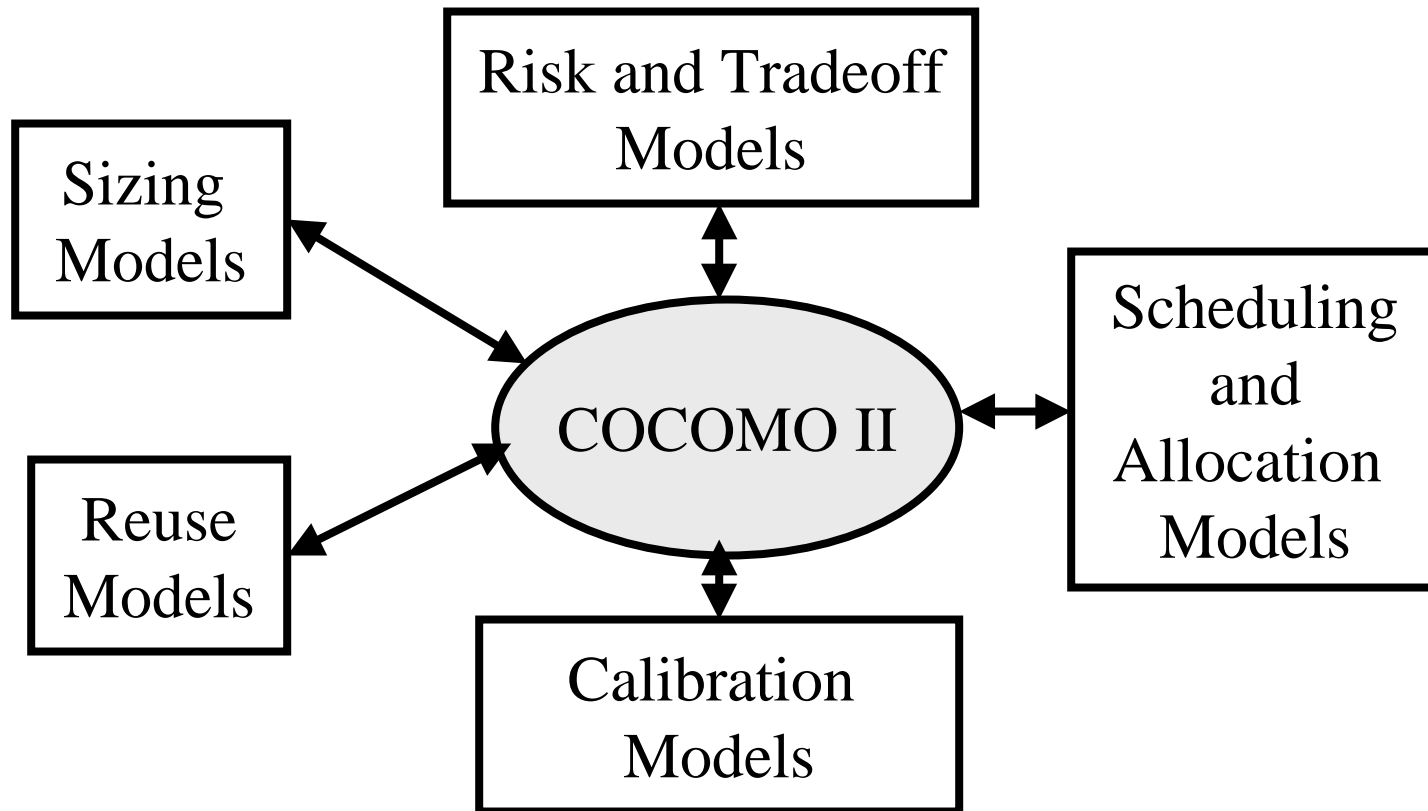
Outline

- COCOTS in Context of COCOMO Suite
- Rationale for using COTS approach
- General COTS Usage Pitfalls & Mitigation Strategies
- Lessons Learned from Source Code Unavailable
- Insights from COTS Data Collected to Date
- Approach to Schedule Estimation
- COCOTS Model Overview
- Latest Calibration Results
- Tool Demo

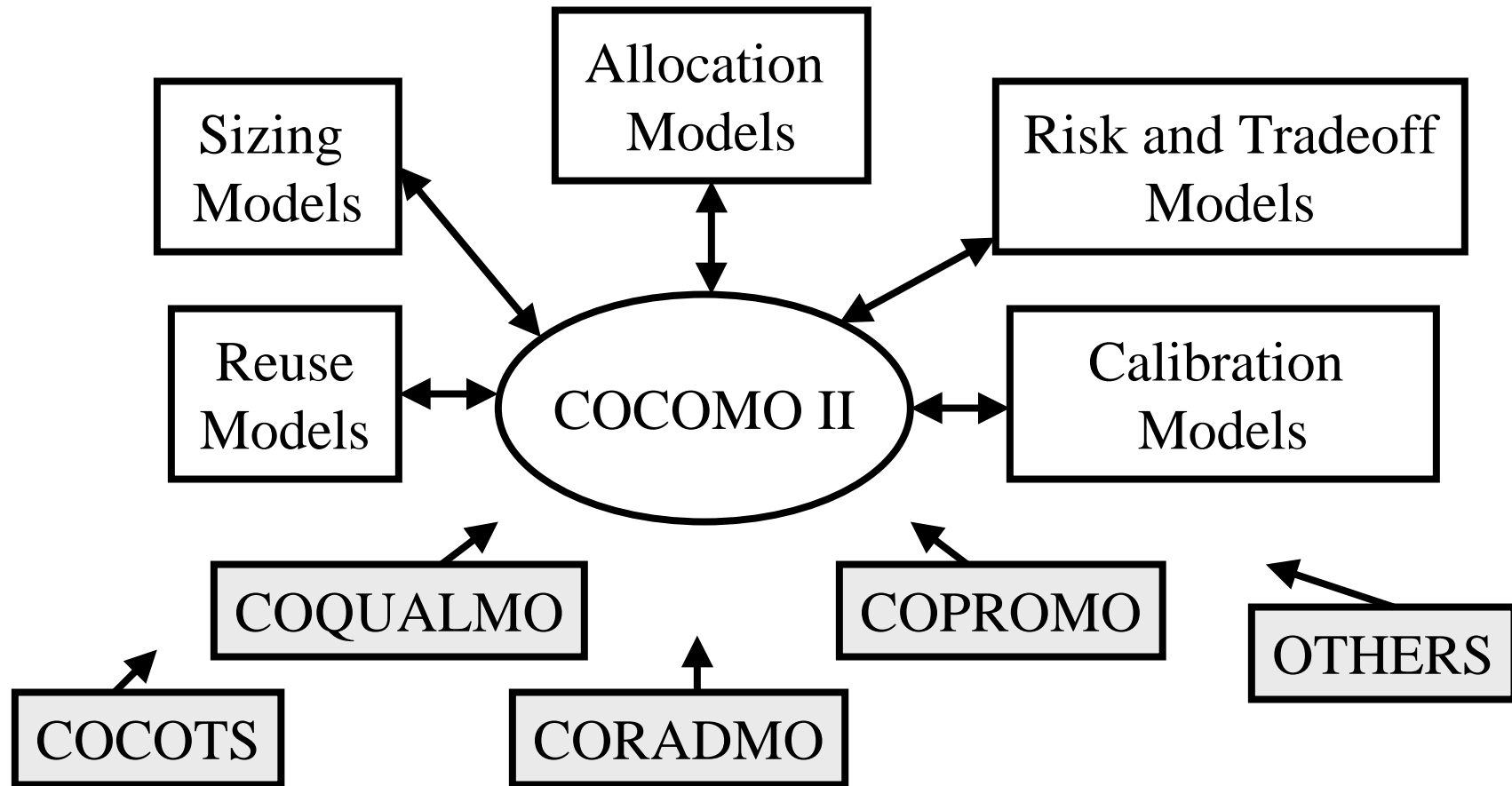


**UNIVERSITY OF
SOUTHERN CALIFORNIA**

COCOTS within the COCOMO Suite



(Figure courtesy of Mr. Don Reifer)

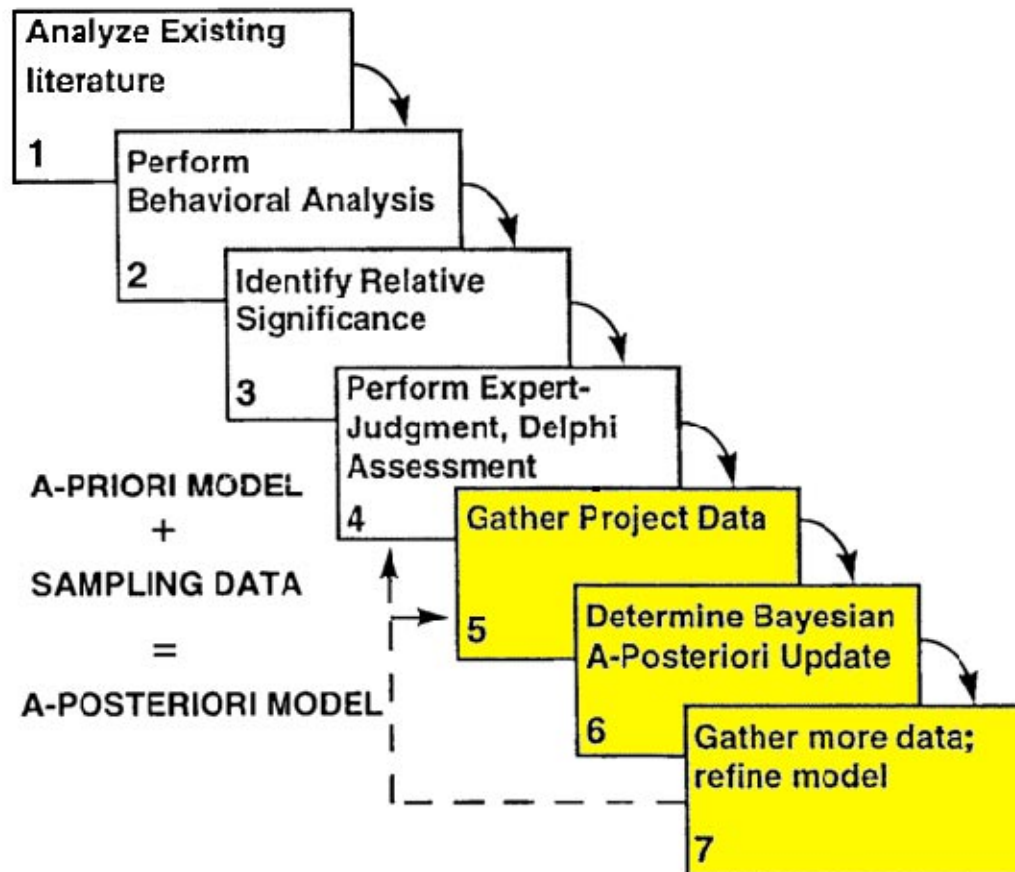


New Models Addressing Research Topics

(Figure courtesy of Mr. Don Reifer)



USC-CSE Seven Step Modeling Methodology





**UNIVERSITY OF
SOUTHERN CALIFORNIA**

Rationale for COTS Approach



COTS Definition

- “Commercial Off the Shelf” Software
- Commercial Software Products
 - sold, leased, licensed at advertised prices
- Source Code Unavailable
 - generally an application program interface (API)
 - frequently tailoring options
- Usually periodic releases with feature growth, obsolescence



Related Terms

- COTS “Commercial Off the Shelf”
 - Black Box (internal mods not allowed)
 - White Box (internal mods permitted - we treat as NDI)
- GFE “Government Furnished Equipment”
- GOTS “Government Off the Shelf”
- NDI “Non-developmental Item/Not Developed In-house”
- REUSE Code
 - source code originally written for some other project



Rationale for Using COTS Products

- Significant change in s/w development practice over past 20 years:
 - building systems with pre-existing software to keep development & maintenance costs as low as possible
 - One such source: COTS
- Rationale for COTS based systems:
 - involve less development time by taking advantage of existing, market proven, vendor supported products, thus lowering overall development costs

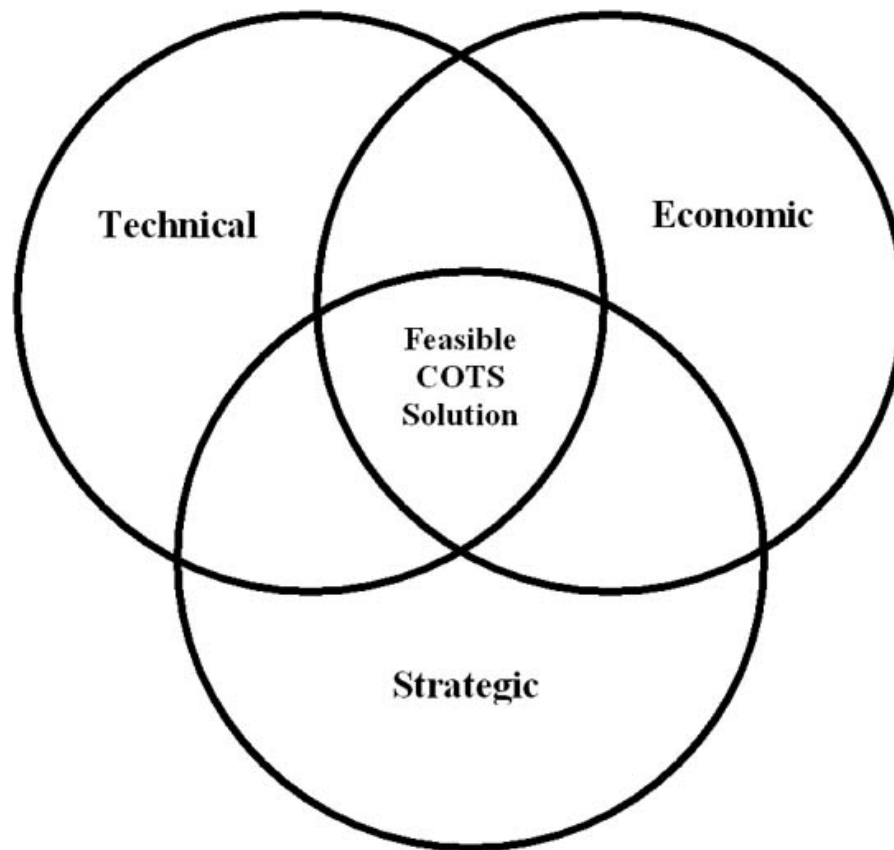


Caveat to Using COTS Products

- Two main characteristics of COTS:
 - source code not available to developer
 - evolution not under control of developer
- Results in trade-off:
 - development time can be reduced, but often at cost of increased s/w component integration work
- Unique risks associated with COTS:
 - cost of licensing and redistribution rights, royalties, effort needed to understand the COTS software, pre-integration assessment and evaluation, post-integration certification of compliance with mission critical or safety critical requirements, indemnification against faults or damage caused by vendor supplied components, and costs incurred due to incompatibilities with other needed software and/or hardware



When are COTS Products the “Right” Solution?



Copyright 1997 University of Southern California



When are COTS Products the “Right” Solution?

- When they lie at the intersection of the three determinants of feasibility, *and do so demonstrably better than could original code*:
 - technical, economic, and strategic constraints
- Technical
 - ability to supply the desired functionality at the required level of reliability
- Economic
 - ability to be incorporated and maintained in the new system within the available budget and schedule
- Strategic
 - ability to meet needs of the system operating environment--including technical, political, and legal considerations--now, and as environment is expected to evolve in the future



COTS Phenomena, Pitfalls and Practices

- You have no control over a COTS product's functionality or performance
- Most COTS products are not designed to interoperate with each other
- You have no control over a COTS product's evolution
- COTS vendor behavior varies widely



1. You have no control over a COTS product's functionality or performance.

Pitfalls to Avoid	Recommended Practices to Adopt
<ul style="list-style-type: none">• Using the waterfall model on a COTS integration project.• Using evolutionary development with the assumption that every undesired feature can be changed to fit your needs.• Believing that advertised COTS capabilities are real.	<ul style="list-style-type: none">• Use risk management and risk-driven spiral-type process models.• Perform the equivalent of a “receiving inspection” upon initial COTS receipt.• Keep requirements negotiable until the system’s architecture and COTS choices stabilize.• Involve all key stakeholders in critical COTS decisions.



2. Most COTS products are not assigned to interoperate with each other.

Pitfalls to Avoid	Recommended Practices to Adopt
<ul style="list-style-type: none">• Premature commitment to incompatible combinations of COTS products.• Trying to integrate too many incompatible COTS products.• Deferring COTS integration till the end of the development cycle.• Committing to a tightly coupled subset of COTS products with closed, proprietary interfaces.	<ul style="list-style-type: none">• Use the Life Cycle Architecture milestone as a process anchor point.• Use the Architecture Review Board (ARB) best commercial practice at the Life Cycle Architecture milestone.• Go for open architectures and COTS substitutability.



3. You have no control over a COTS product's evolution.

Pitfalls to Avoid	Recommended Practices to Adopt
<ul style="list-style-type: none">• “Snapshot” requirements specs and corresponding point-solution architectures.• Understaffing for software maintenance,• Tightly coupled, independently evolving COTS products.• Assuming that uncontrollable COTS evolution is just a maintenance problem.	<ul style="list-style-type: none">• Stick with dominant commercial standards.• Use likely future system and product line needs as well as current needs as COTS selection criteria.• Use flexible architectures facilitating adaptation to change.• Carefully evaluate COTS vendors’ track records with respect to predictability of product evolution.• Establish a pro-active system release strategy, synchronizing COTS upgrades with system releases.



4. COTS vendor behavior varies widely.

Pitfalls to Avoid	Recommended Practices to Adopt
<ul style="list-style-type: none">• Uncritically accepting COTS vendors' statements about product capabilities and support.• Lack of fallbacks or contingency plans.• Assuming that an initial vendor support honeymoon will last forever.	<ul style="list-style-type: none">• Perform extensive evaluation and reference checking of a COTS vendor's advertised capabilities and support track record.• Establish strategic partnerships or other incentives for COTS vendors to provide support.• Negotiate and document critical vendor support agreements.



Summary COTS Advantages and Disadvantages

Advantages

- Available now; earlier payback
- Avoids expensive development & maintenance
- Predictable license costs & performance
- Rich in functionality
- Broadly used, mature technology
- Frequent upgrades often anticipate organization's needs
- Dedicated support organization
- Hardware/software independence
- Tracks technology trends

Disadvantages

- Licensing and intellectual property procurement delays
- Up front license fees
- Recurring maintenance fees
- Reliability often unknown/ inadequate; scale often difficult to change
- Unnecessary features compromise usability, performance
- Functionality, efficiency constraints
- No control over upgrades/maintenance
- Dependency on vendor
- Efficiency sacrifices
- Integration not always trivial; incompatibilities among vendors
- Synchronizing multiple-vendor upgrades



**UNIVERSITY OF
SOUTHERN CALIFORNIA**

Lessons Learned from Projects using COTS Approach

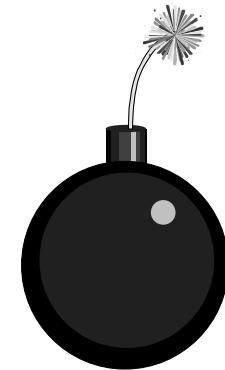
(Betsy Clark)



UNIVERSITY OF
SOUTHERN CALIFORNIA



IF YOU
ASSUME,
YOUR COTS
ARE DOOMED!

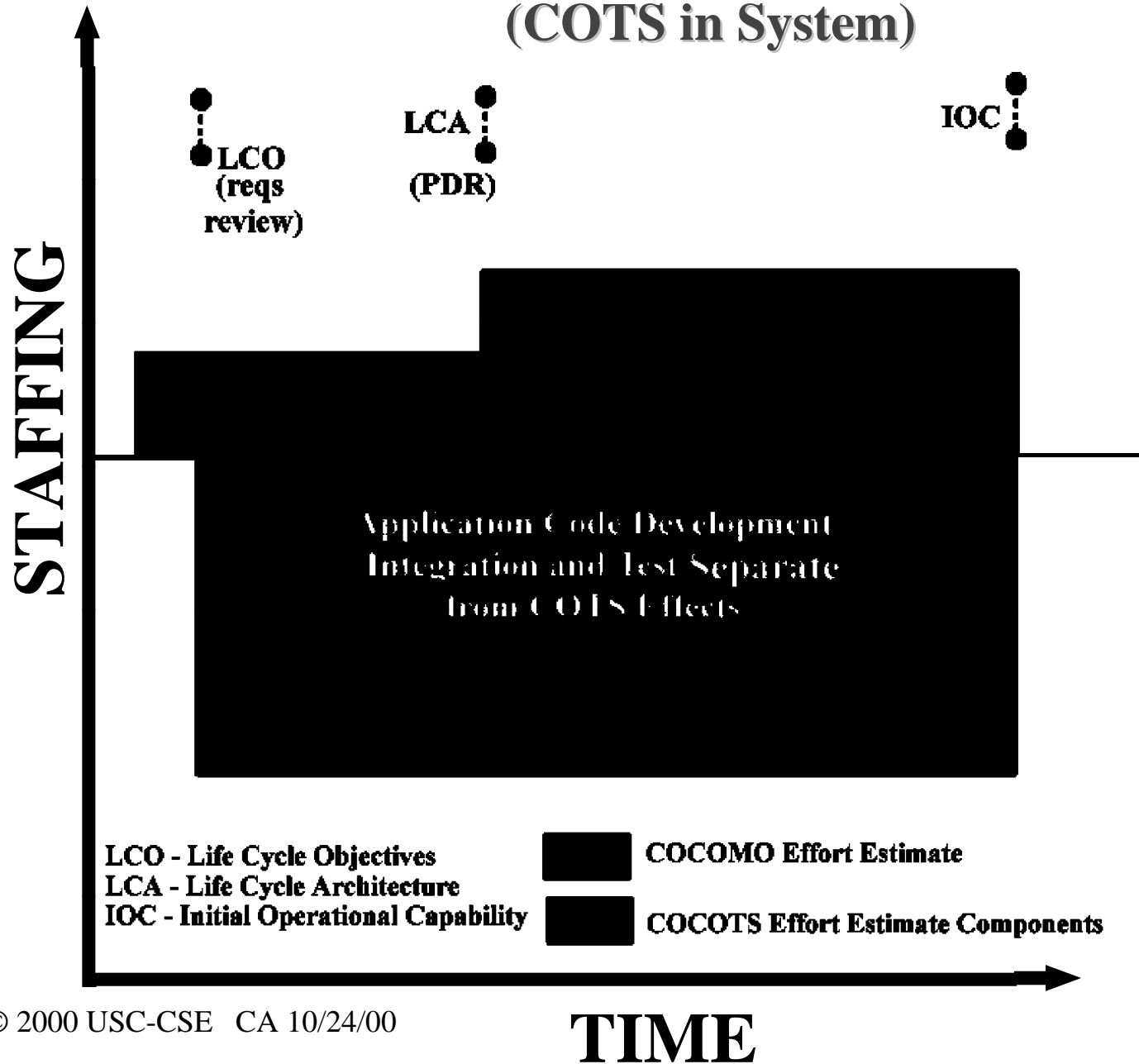




**UNIVERSITY OF
SOUTHERN CALIFORNIA**

Insights from COCOTS Calibration Data

COCOMO vs. COCOTS Cost Sources (COTS in System)



Current COCOTS Database

- 20 Industrial projects
- Data collection continuing
 - (COCOMO 81 debuted with 63 calibration data points)
- Following summaries based on those 20 points

Project Domains

(project sources: Army, Navy, FAA, CSE Affiliates)

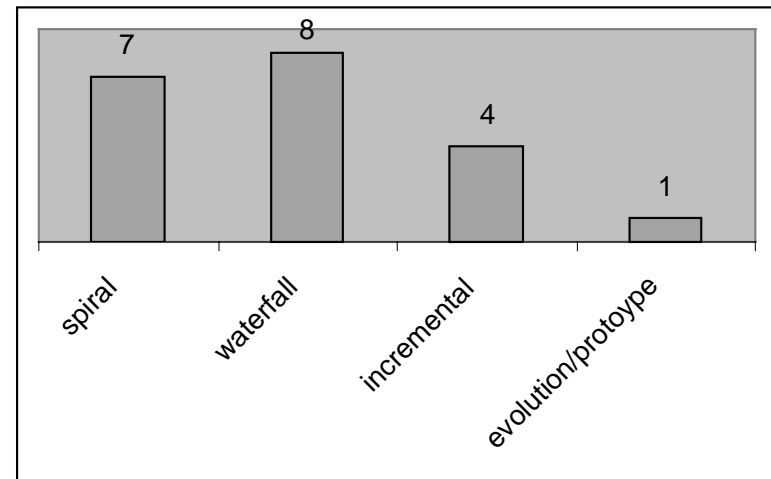
- Air Traffic Management 8
- Business (including databases) 3
- Communication, Navigation, & Surveillance 4
- Logistics 1
- Mission Planning 1
- Operations 2
- Web-based Maps 1

Classes of COTS Products Used

- configuration mgmt/build tools
- databases
- data conversion packages
- disk arrays
- compilers
- communication protocols/packages
- emulators
- engineering tools (reqs mgmt, design)
- software process tools
- GUIs/GUI builders
- graphic information systems
- middleware
- operating systems
- network managers
- device drivers
- problem mgmt
- report generators
- back office retail
- telecommunication & infrastructure
- telemetry analysis
- telemetry processing
- word processors

Development Processes

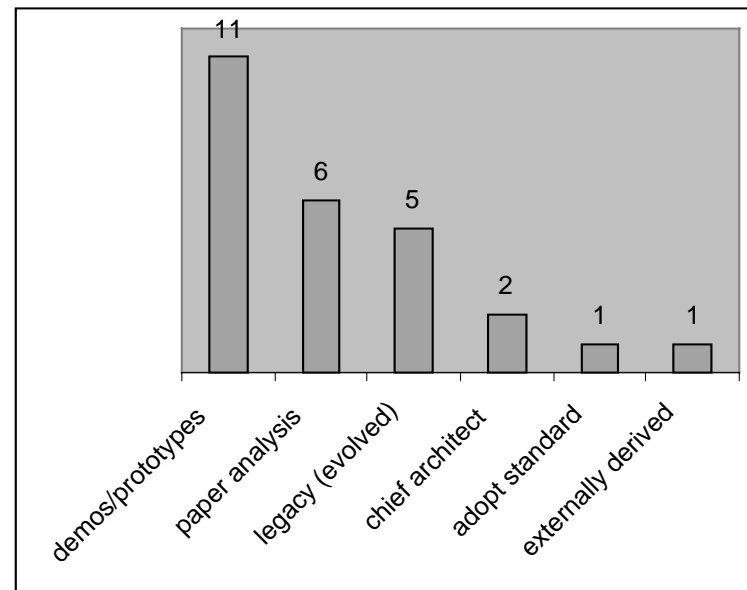
- Spiral 7
- Waterfall 8
- Incremental 4
- Evolution/Prototype 1



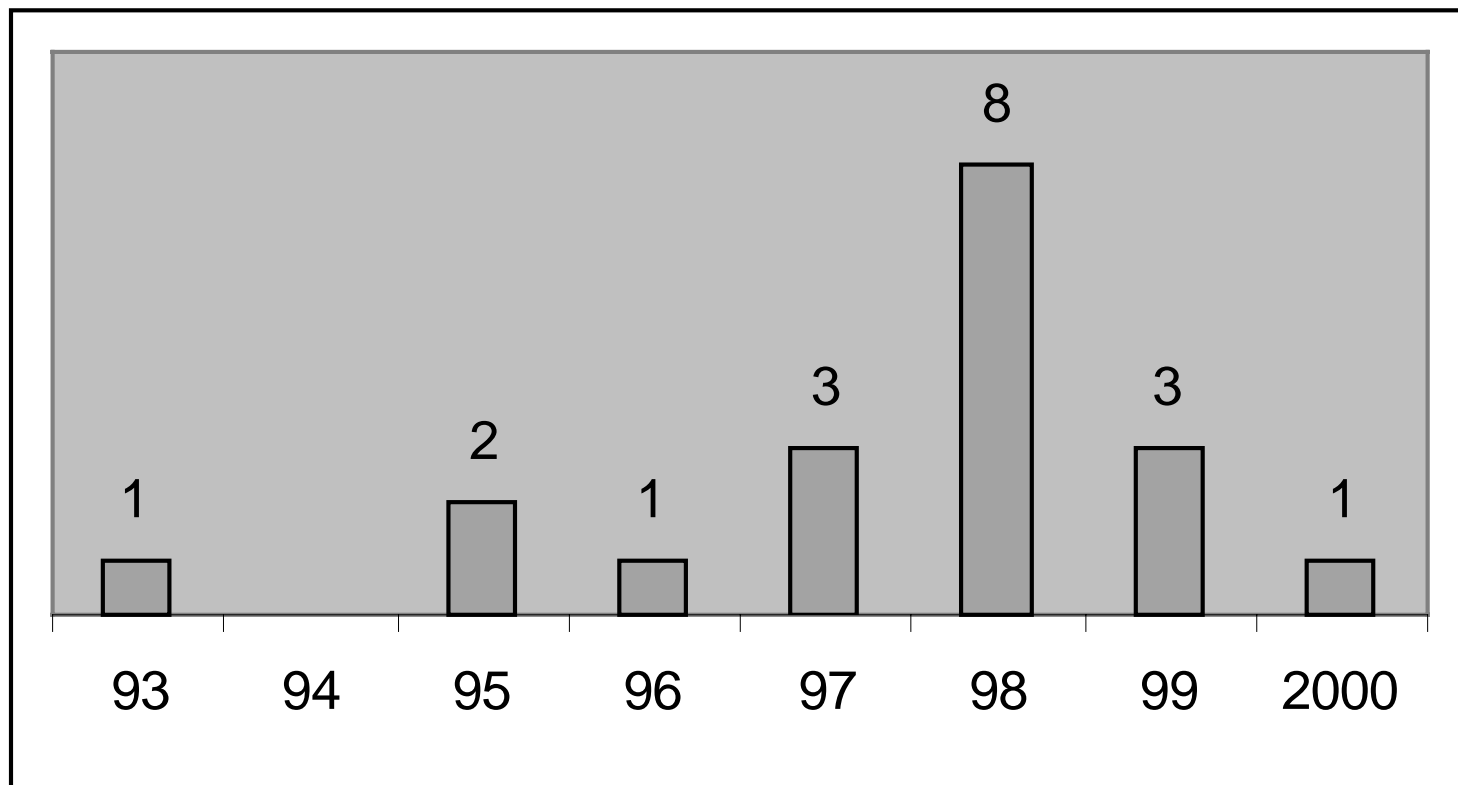
(most projects currently in maintenance)

Architectures & Architecting Processes

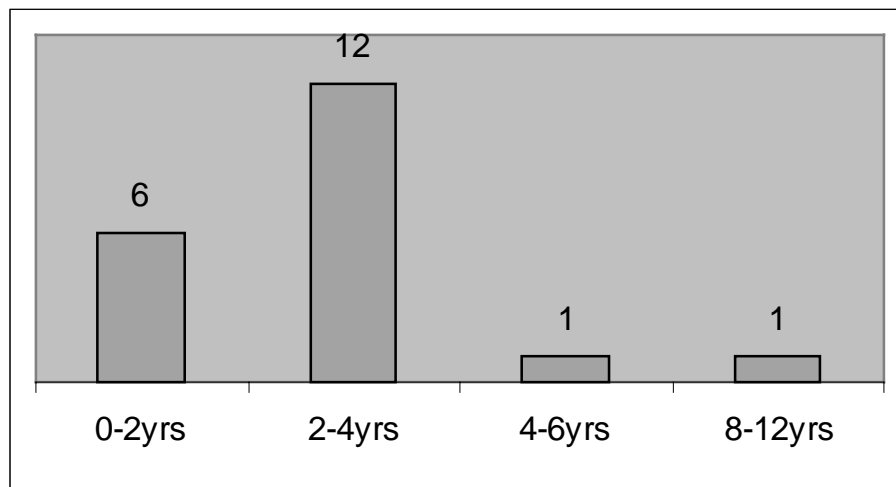
- Architectures
 - *all over the map but with two highly common elements:*
“distributed” and “multi-threaded”
- Process (*some projects used more than one*)
 - 1- demos/prototypes 11
 - 2- paper analysis 6
 - 3- (evolved) legacy 5
 - 4- chief architect 2
 - 5- adopt industry standard 1
 - 6- externally developed 1



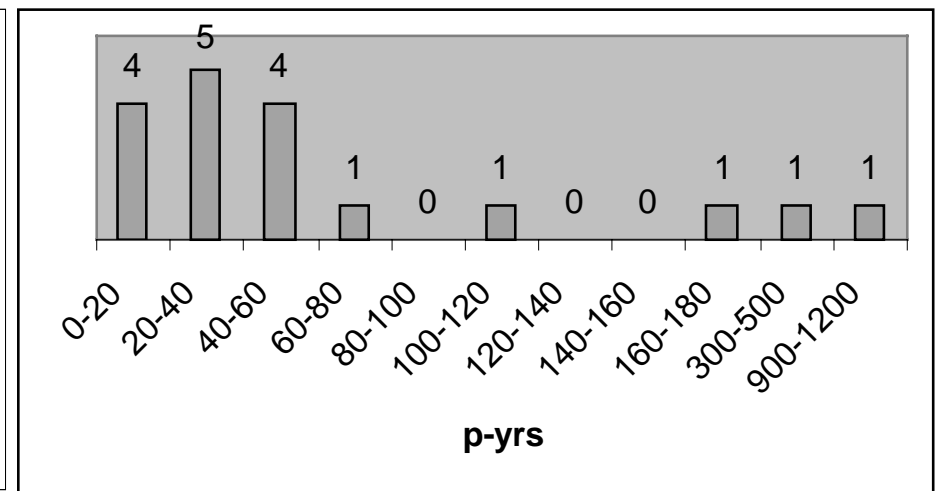
Delivery Dates



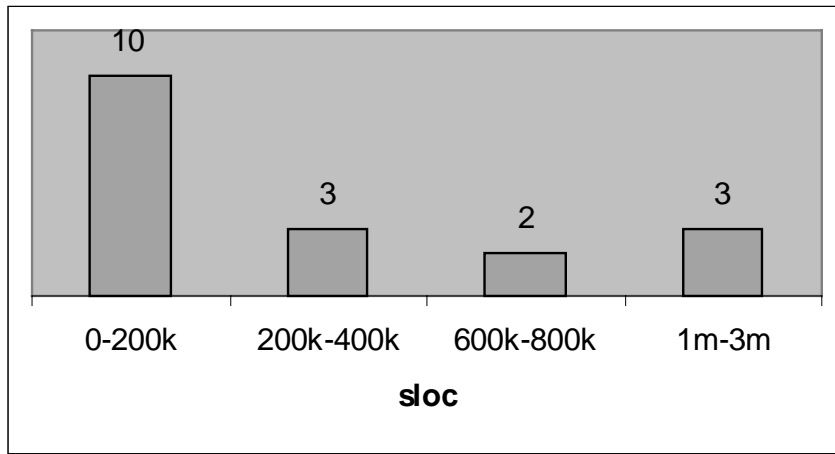
Total Duration



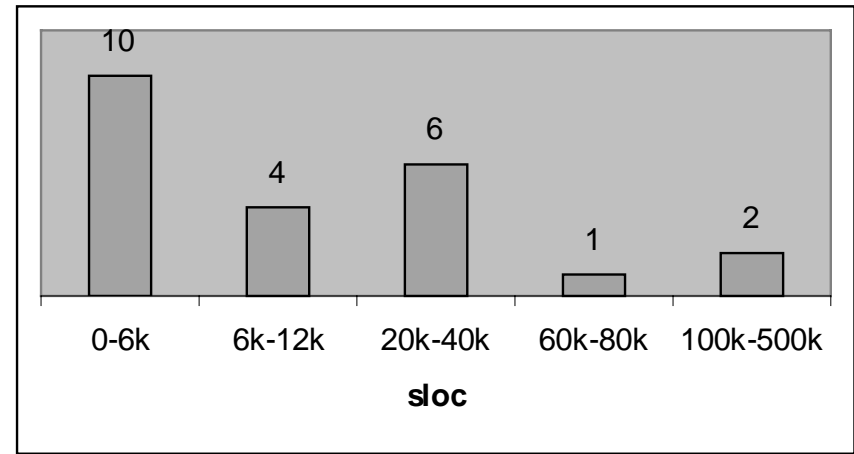
Total Effort



Total SLOC

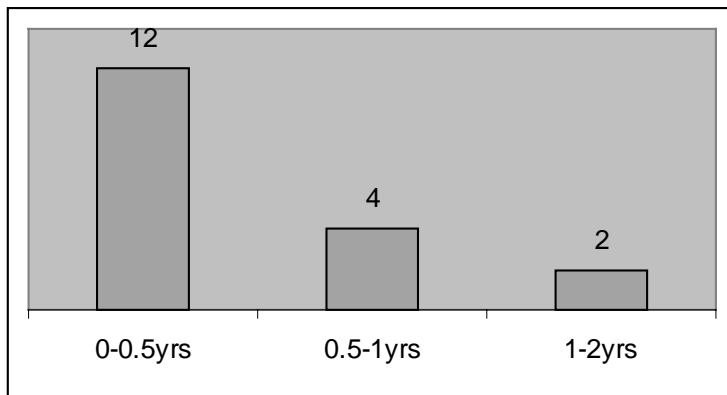


Glue SLOC

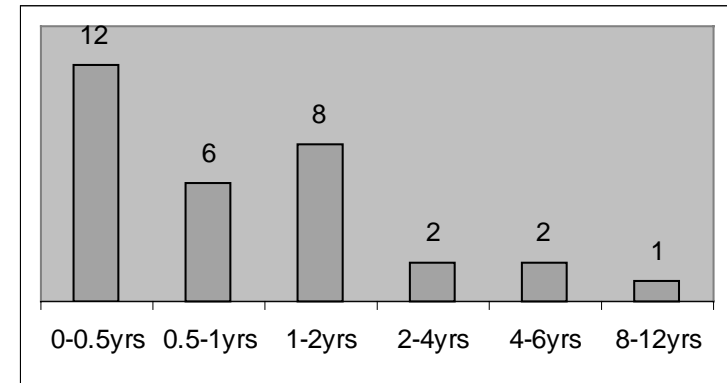


Schedule Duration by Activity

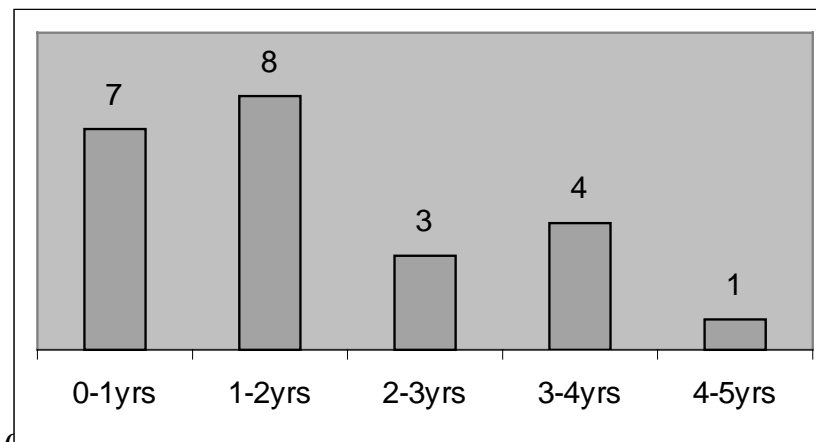
Assessment



Tailoring

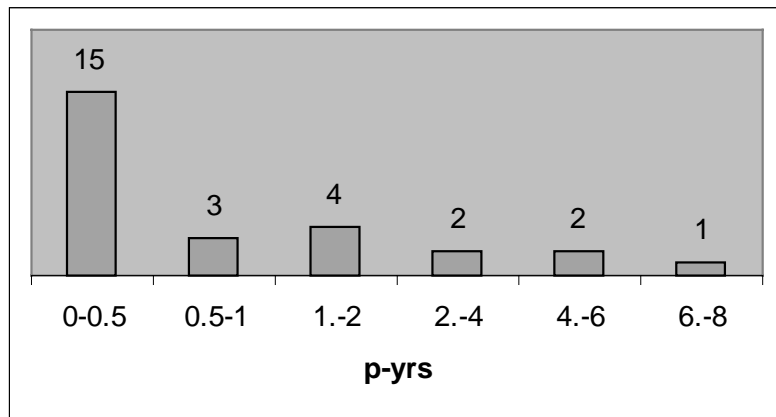


Glue Code

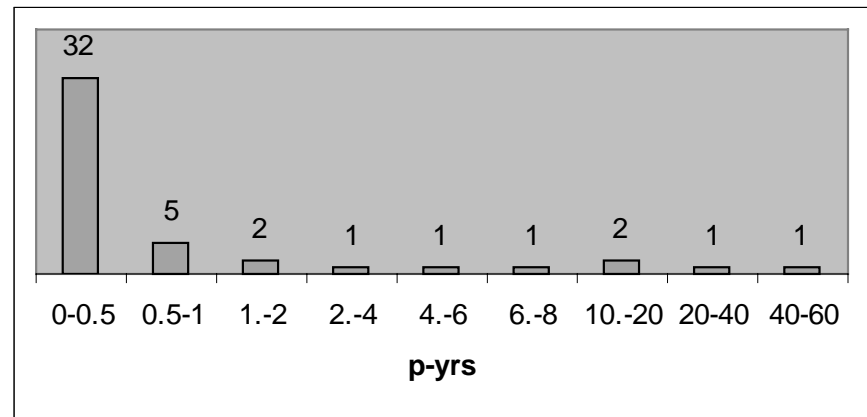


Effort by Activity

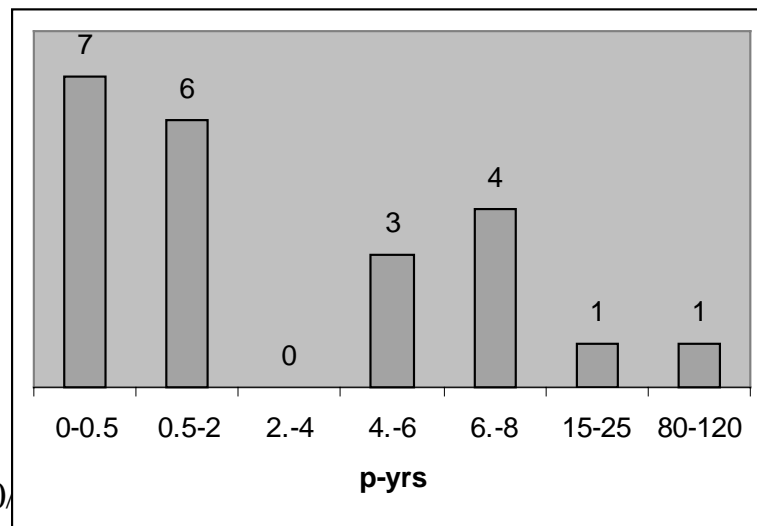
Assessment



Tailoring

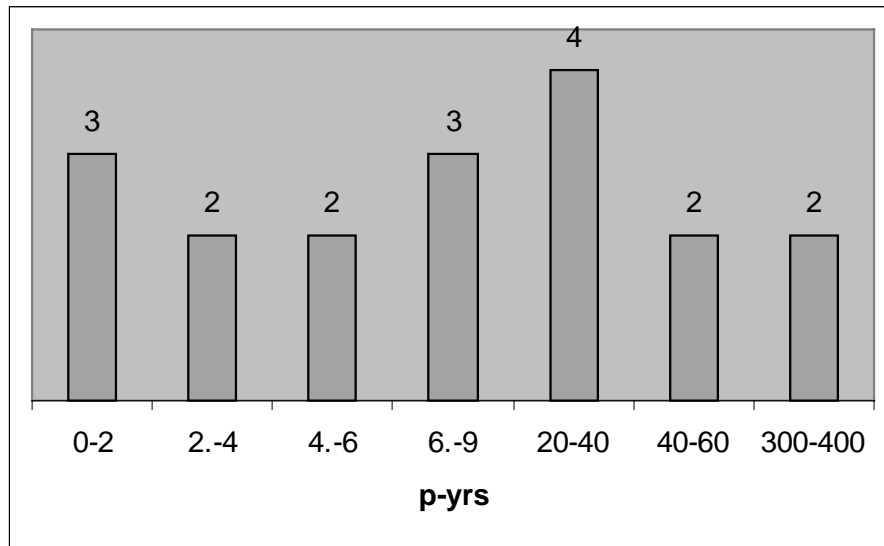


Glue Code

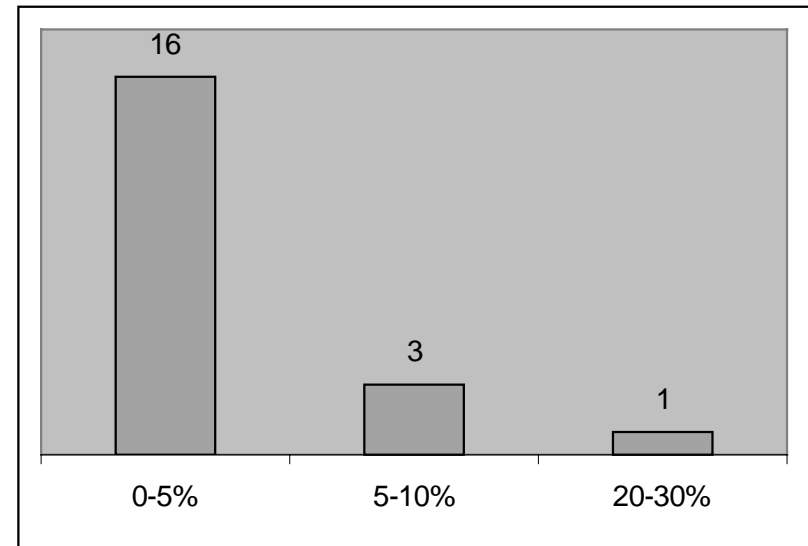


Effort by Activity (*cont'd*)

System effort due to COTS
volatility

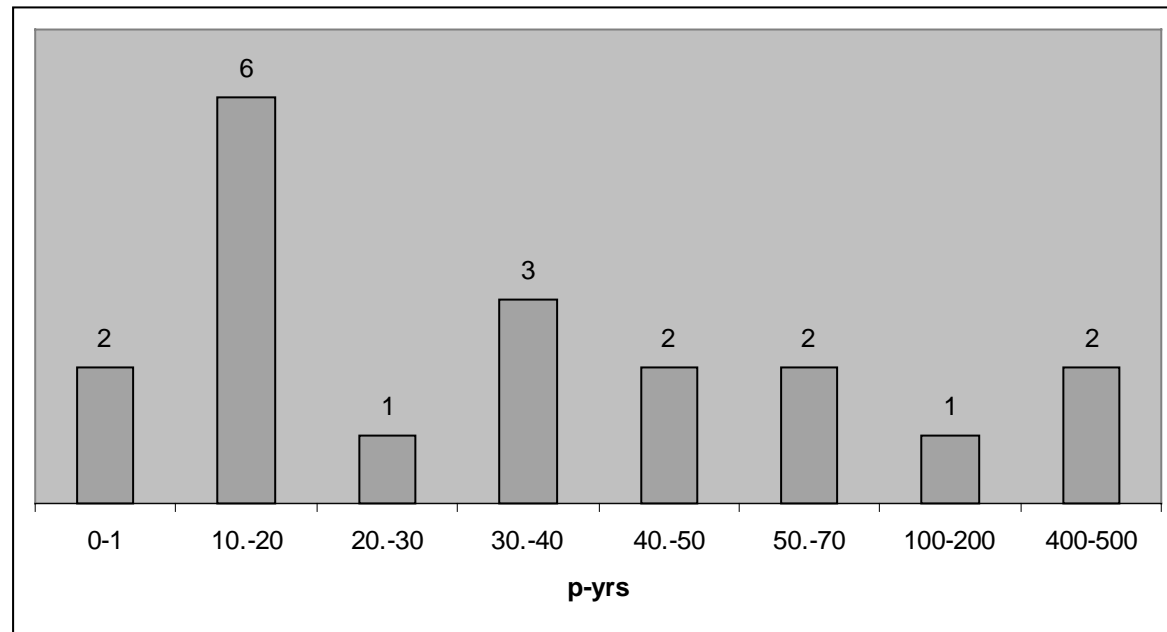


Percentage system rework
due to COTS volatility



Total COTS Effort in System

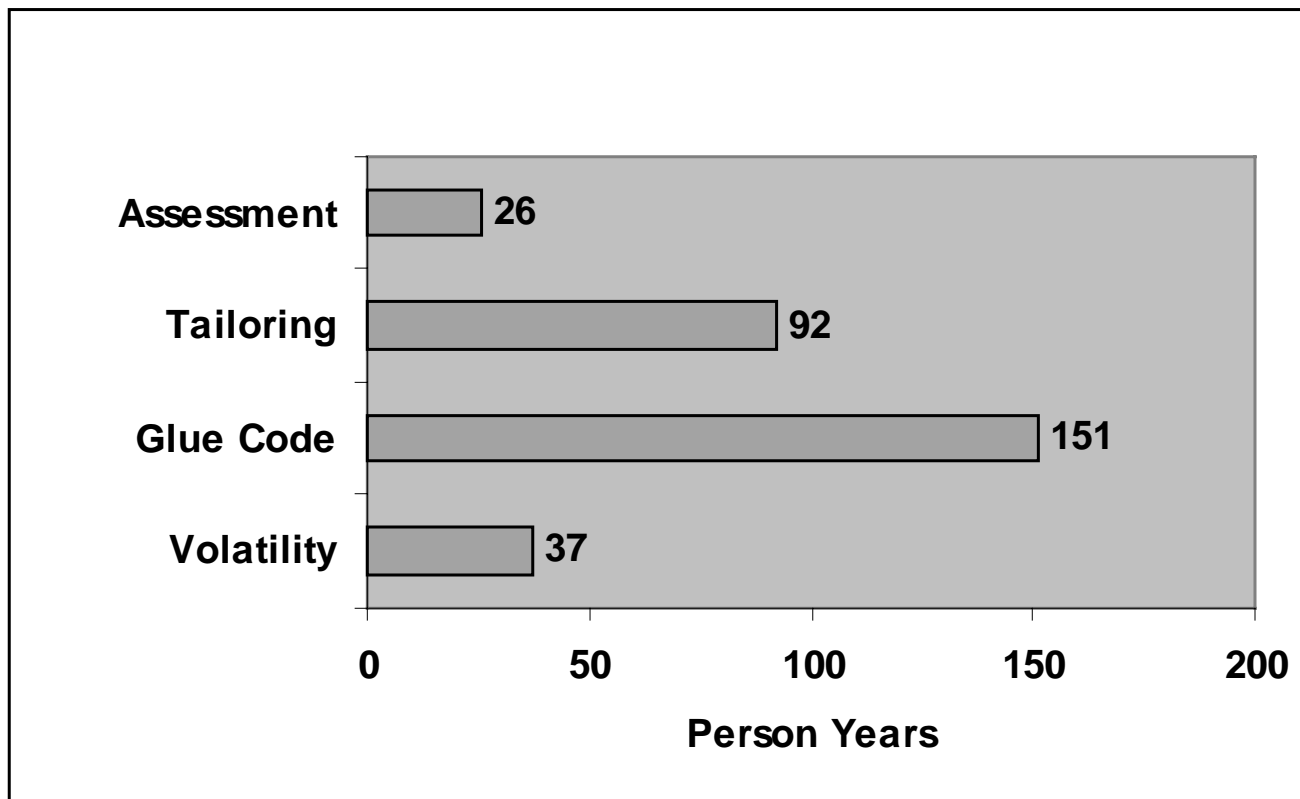
(assessment+tailoring+glue code+volatility)



More Data Insights

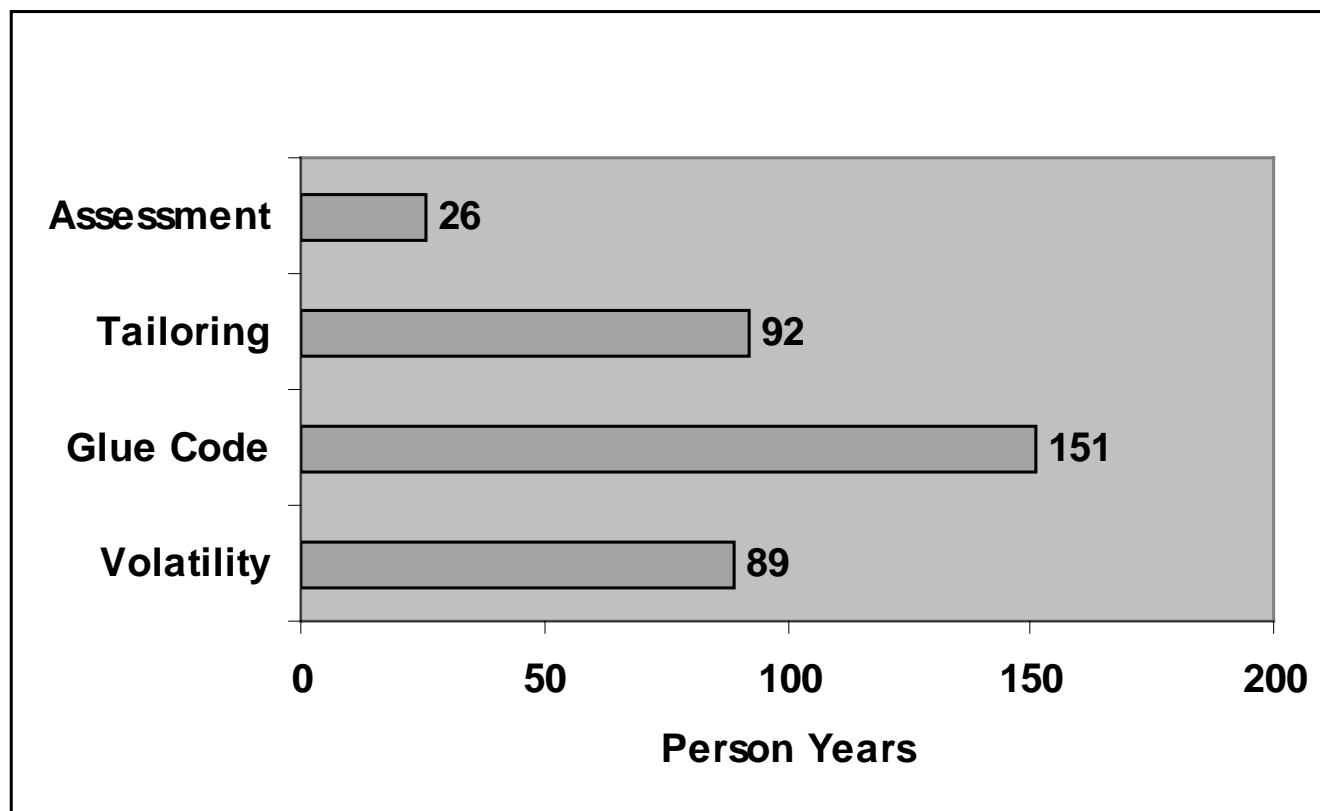
COTS Effort Distribution Over Existing 20 Calibration Data Projects

(volatility as reported)

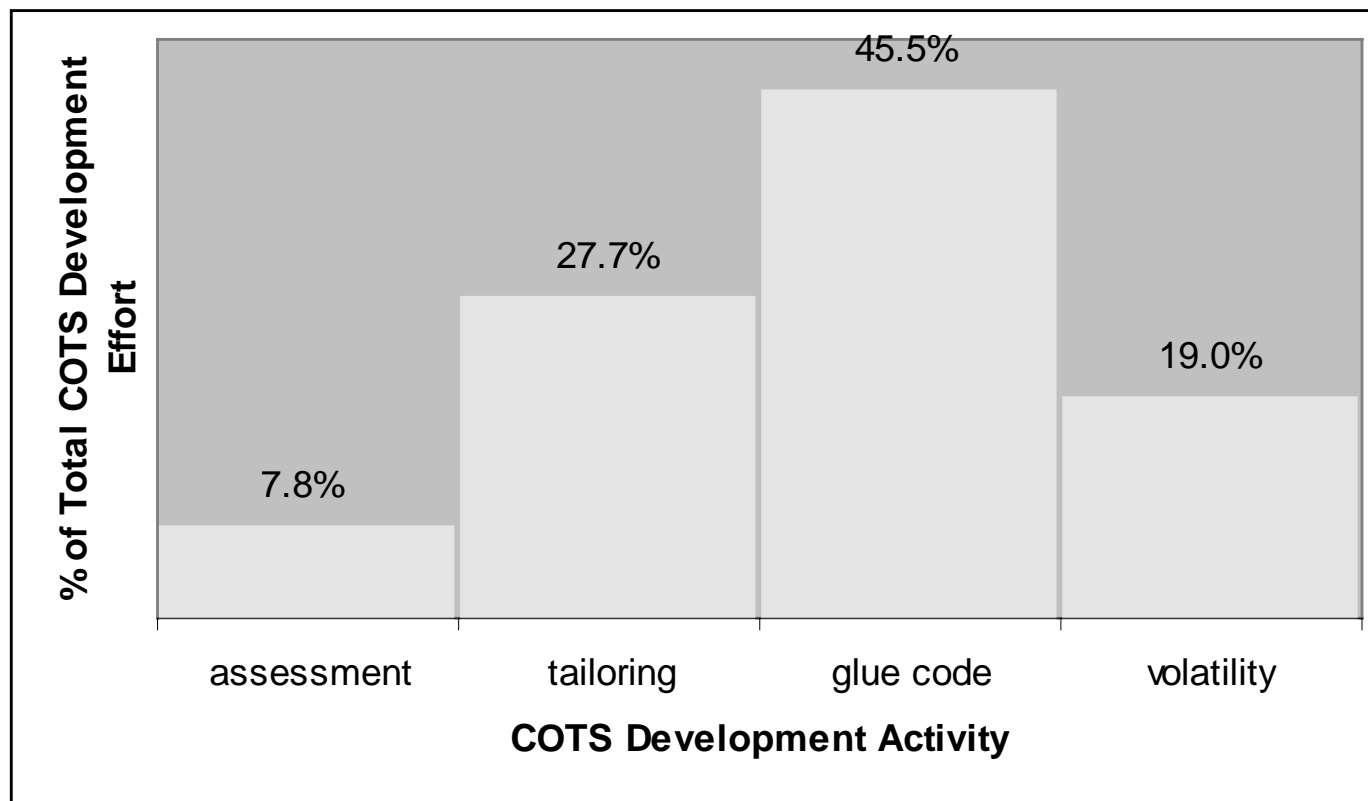


COTS Effort Distribution Over Existing 20 Calibration Data Projects

(volatility as derived)



Bottom Line: this data offers for the first time the beginnings of an *empirically*-based, initial COTS development effort distributed by activity



An Approach to Schedule Estimation

MBASE Effort & Schedule Distribution Percentages

Phase (endpoints)	MBASE	
	Effort%	Schedule%
Inception (IRR to LCO)	6 (2-15)	12.5 (2-30)
Elaboration (LCO to LCA)	24 (20-28)	37.5 (33-42)
Construction (LCA to IOC)	76 (72-80)	62.5 (58-67)
Transition (IOC to RRR)	12 (0-20)	12.5 (0-20)
Totals:	118	125

COCOMO II.2000 Schedule Equation

$$\text{TDEV} = [C \times (\text{PM}_{\text{NS}})^F] \times \frac{\text{SCED}\%}{100}$$

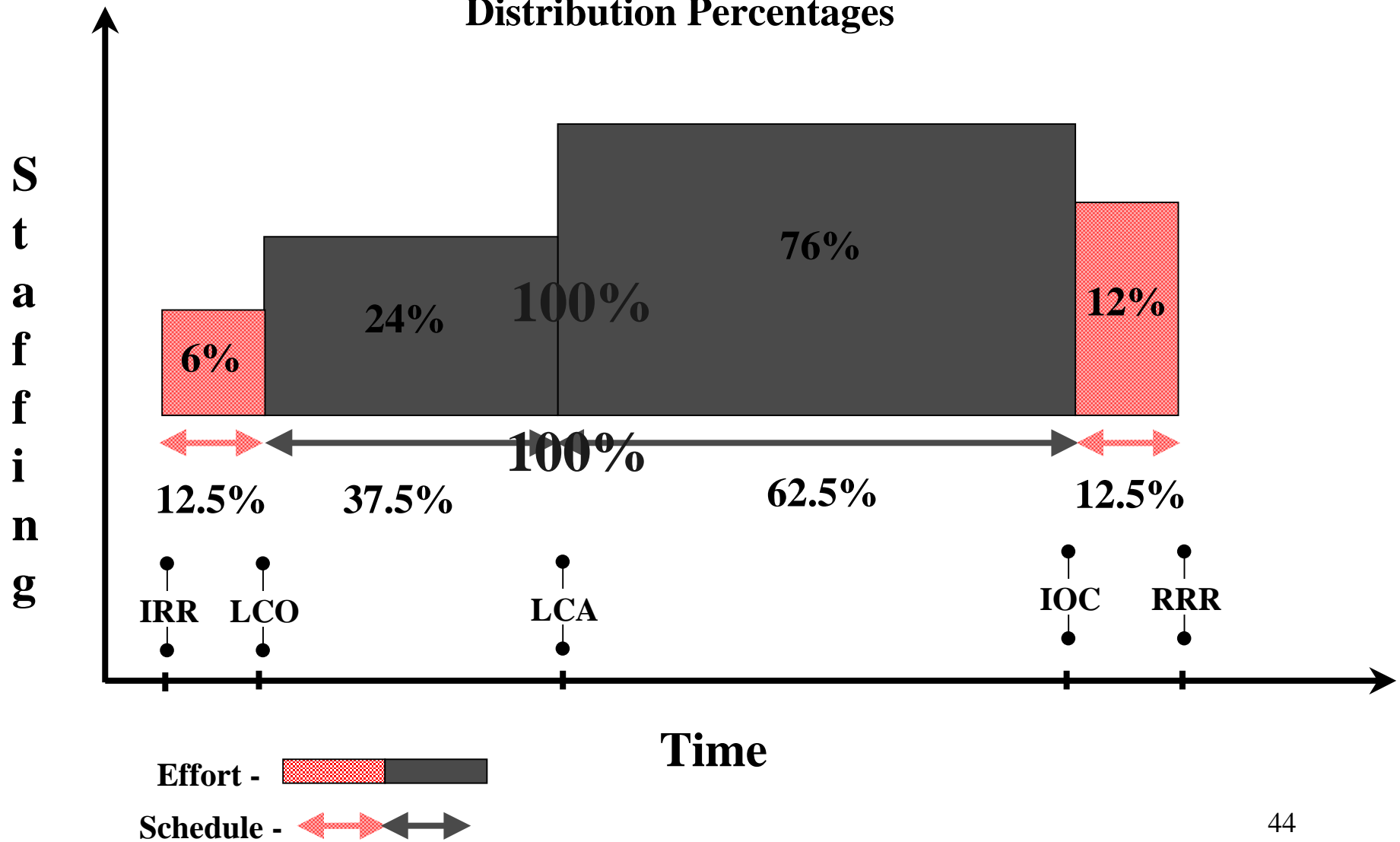
$$F = (D + 0.2 \times [E - B])$$

$$E = B + 0.01 \times \sum_{j=1}^5 \text{SF}_j$$

Where $C = 3.67$, $D = 0.28$, and $B = 0.91$

<u>Symbol</u>	<u>Description</u>
B	The scaling base-exponent for the effort equation
C	Coefficient that can be calibrated
D	Scaling base-exponent that can be calibrated
E	The scaling exponent for the effort equation
F	Scaling exponent for schedule
PM_{NS}	Person Months estimated without the SCED cost driver (Nominal Schedule)
SCED	Percentage of Required Schedule Compression
SF	COCOMO II Scale Drivers
TDEV	Time to Develop in calendar months

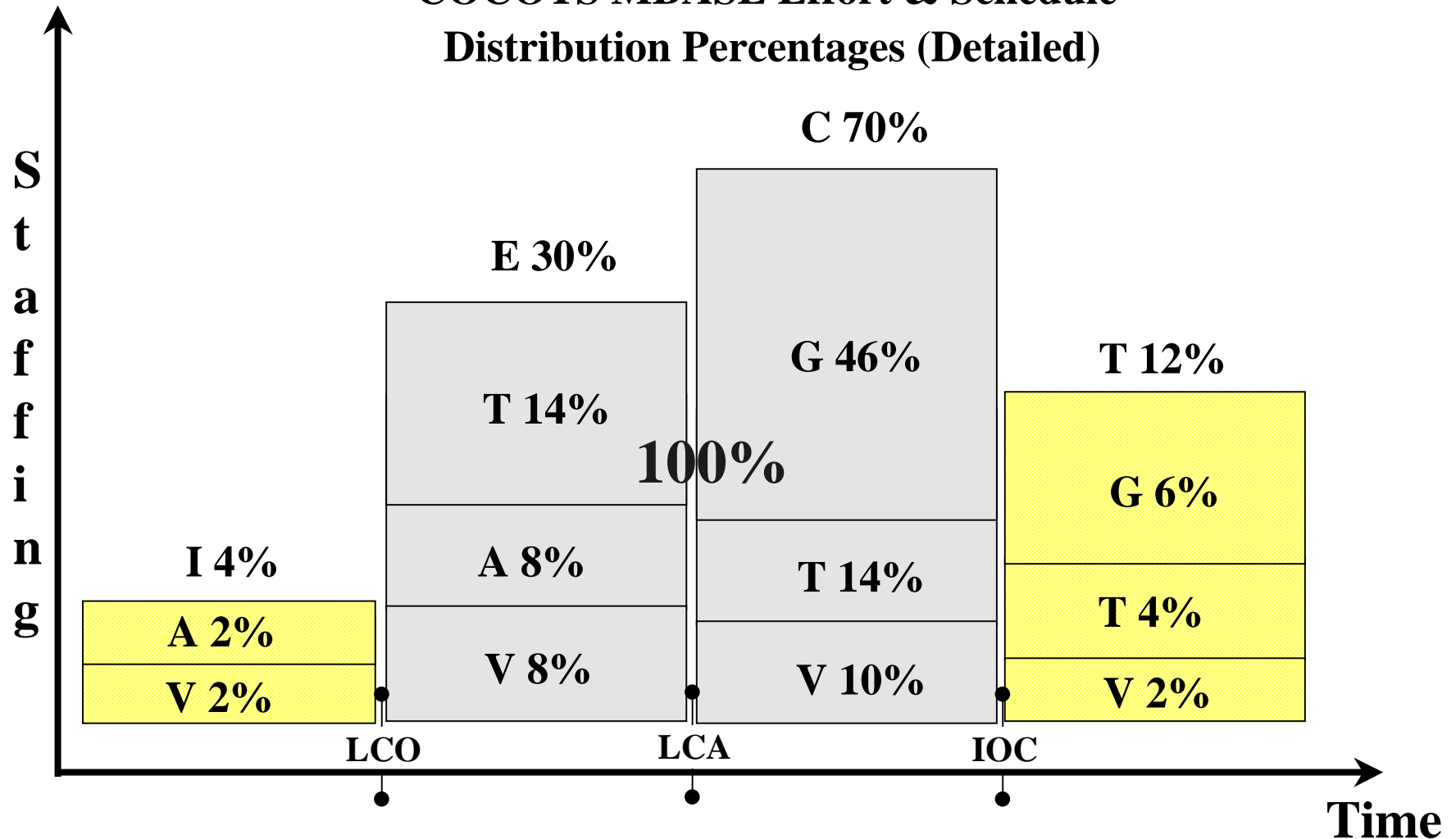
COCOMO II MBASE Effort & Schedule Distribution Percentages



Rules for Combining COCOMO II & COCOTS Efforts for Unified Schedule Equation

- Overall MBASE effort & schedule distributions remain fixed.
- Relative distribution of COTS effort across COTS activities
A, T, G & V remain fixed at percentages derived from
initial 20 data points (7.8%, 27.7%, 45.5%, 19%).
- V percentage allocated evenly across A, T & G.
- Parse A, T, G & V effort across I, E, C & T as follows:
 - A → 25% * 7.8% I & 75% * 7.8% E
 - T → 50% * 27.7% E, 50% * 27.7% C & 12% * 27.7% T
 - G → 100% * 45.5% C & 12% * 45.5% T
 - V → 25% * 33% * 19% I, (75% * 33% * 19% + 50% * 33% * 19%) E,
(50% * 33% * 19% + 100% * 33% * 19%) C, & 12% * 19% T
- Balance final percentages by rounding to ensure percentages
for E & C => 100% and for T => 12%

COCOTS MBASE Effort & Schedule Distribution Percentages (Detailed)



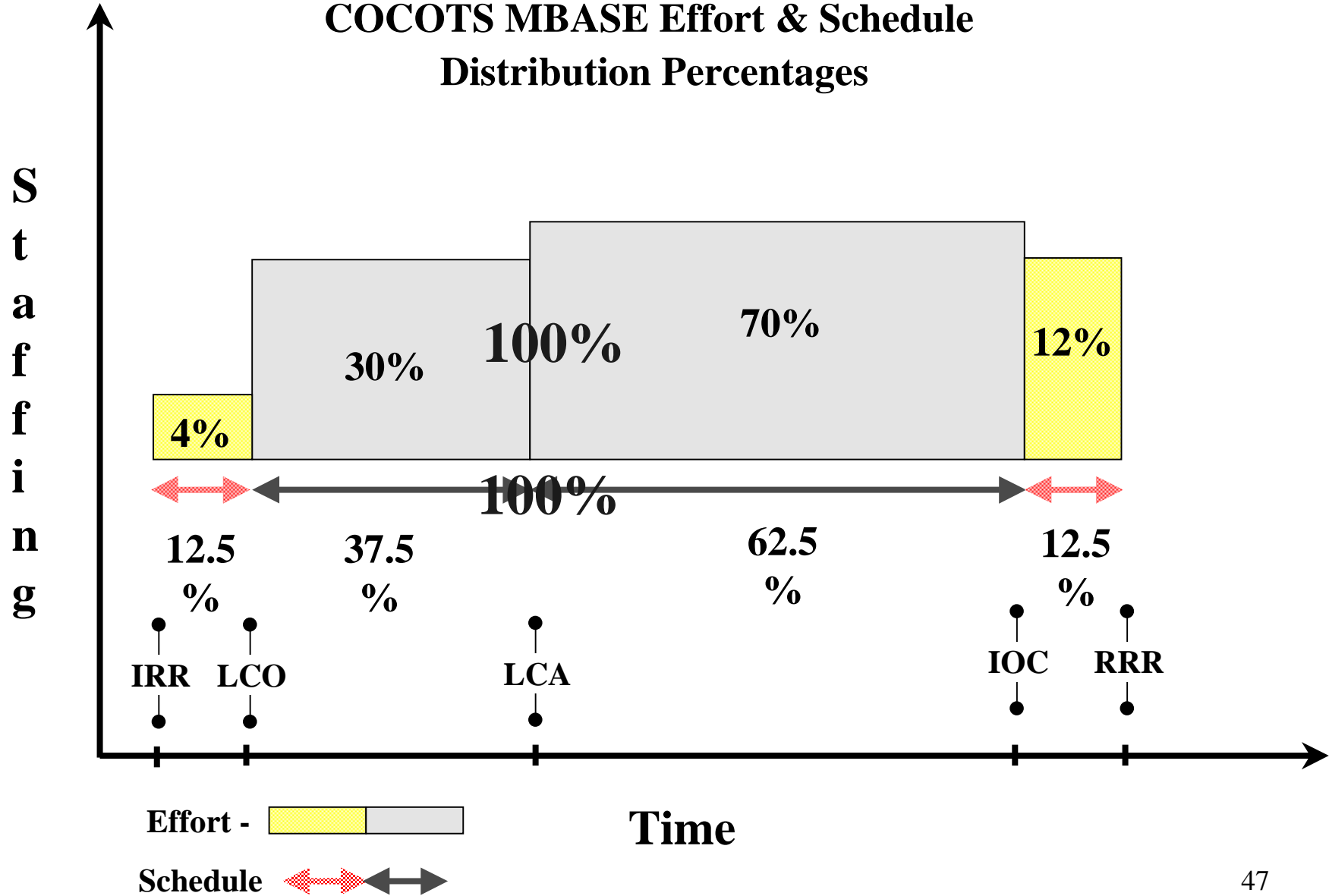
$$A_E = 8\% \sim 7.8\%$$

$$T_E + T_C = 14\% + 14\% = 28\% \sim 27.7\%$$

$$G_C = 46\% \sim 45.5\%$$

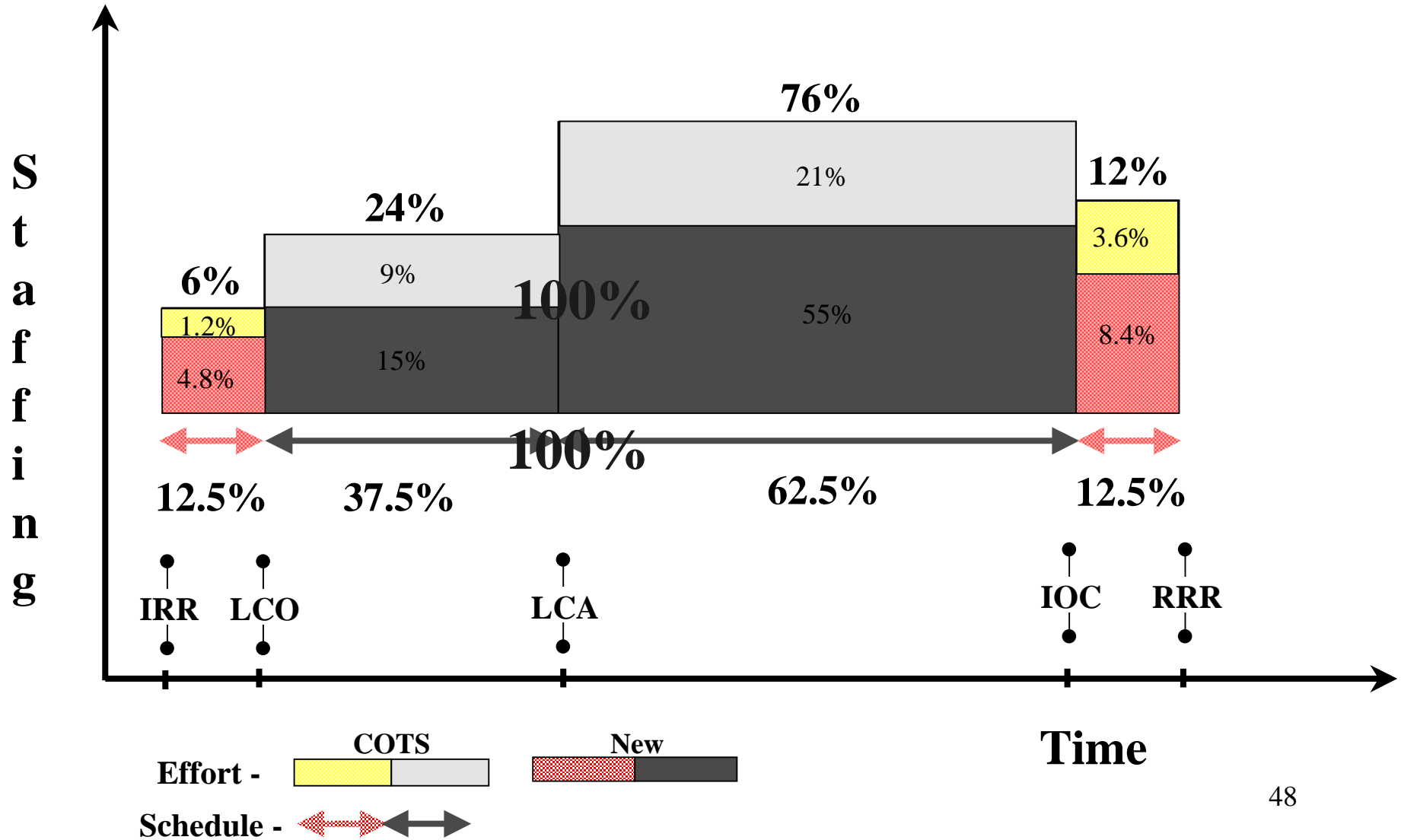
$$V_E + V_C = 8\% + 10\% = 18\% \sim 19\% \quad 46$$

COCOTS MBASE Effort & Schedule Distribution Percentages



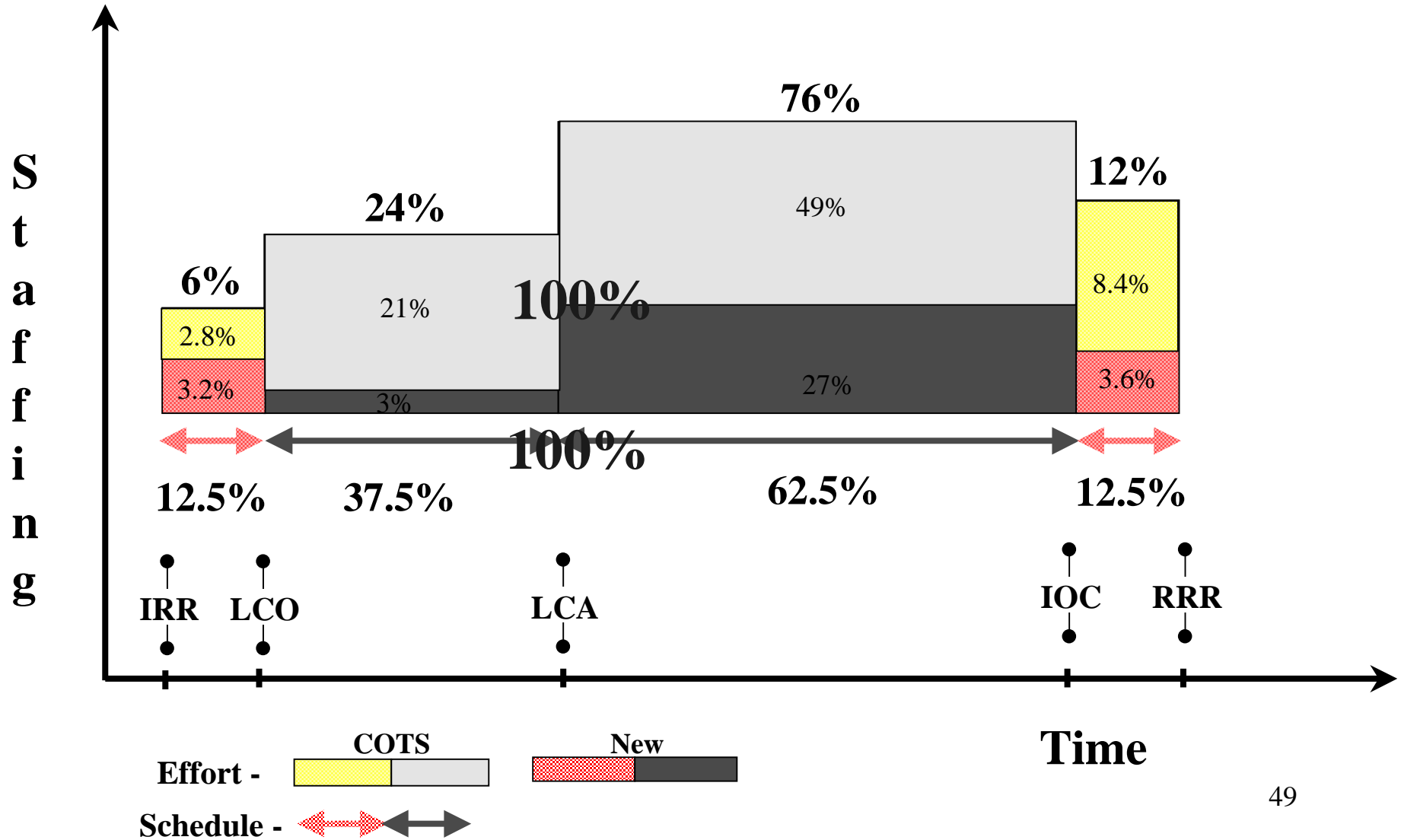
COCOMO II/COCOTS MBASE Effort & Schedule

Distribution Percentages for a 30% of Total Effort COTS-based System



COCOMO II/COCOTS MBASE Effort & Schedule

Distribution Percentages for a 70% of Total Effort COTS-based System



COCOMO II/COCOTS Schedule Equation

$$\text{TDEV} = [C \times (\text{PM}_{\text{NS}})^F] \times \frac{\text{SCED}\%}{100}$$

$$F = (D + 0.2 \times [E - B])$$

$$E = B + 0.01 \times \sum_{j=1}^5 \text{SF}_j$$

Where $C = 3.67$, $D = 0.28$, and $B = 0.91$

Symbol

B	The scaling base-exponent for
C	Coefficient that can be calibrated
D	Scaling base-exponent that can be calibrated
E	The scaling exponent for the effort equation
F	Scaling exponent for schedule
PM_{NS}	Person Months estimated without the SCED cost driver (Nominal Schedule)
SCED	Percentage of Required Schedule Compression
SF	COCOMO II Scale Drivers
TDEV	Time to Develop in calendar months

Now includes COCOMO + COCOTS
effort between LCO & IOC



**UNIVERSITY OF
SOUTHERN CALIFORNIA**

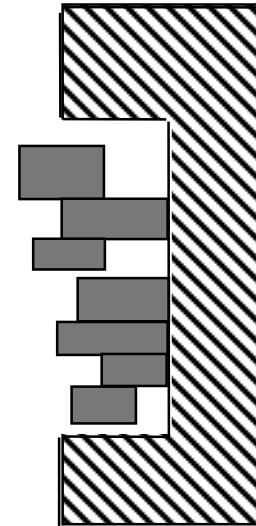
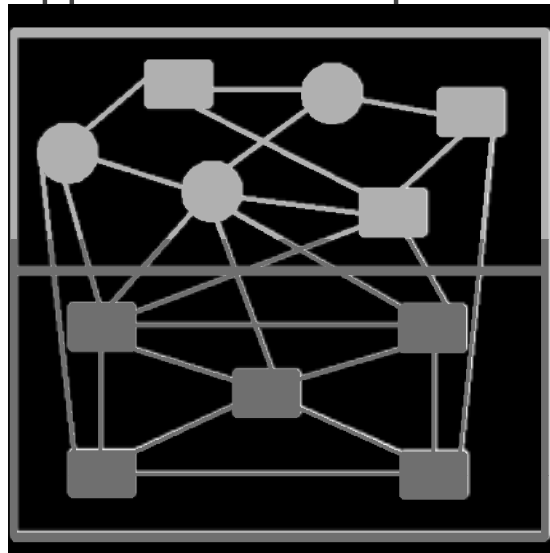
COCOTS Model Overview



Problem Context: Modeling

COTS  and Custom 
Applications Components

New **COCOTS**
Modeling
Problem



COTS Infrastructure
COCOMO II: PVOL, PEXP

COTS Tools
LTEX, TOOL

Cost Modeling Currently Addressed



COTS Software Integration Lifecycle

- 1) Qualify COTS product
- 2) Perform system requirements
- 3) Administer COTS software acquisition
- 4) Prototype the system including COTS software
- 5) Fully integrate COTS s/w and interface code
- 6) Test completed prototype

Objectory Management Checkpoints

Inception	Elaboration		Construction			Transition
Iteration 1	Iteration 2	Iteration 3	Iteration 4	Iteration 5	Iteration 6	Iteration 7

Major Milestones



LCO



LCA



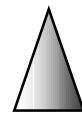
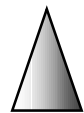
IOC



Full
Release

Strategic focus on global concerns of the entire software project

Minor Milestones



Tactical focus on local concerns of current iteration

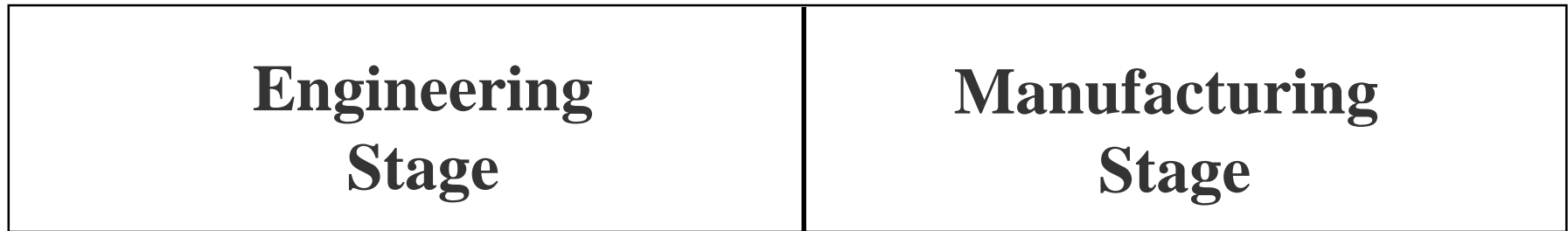
Status

Assessments

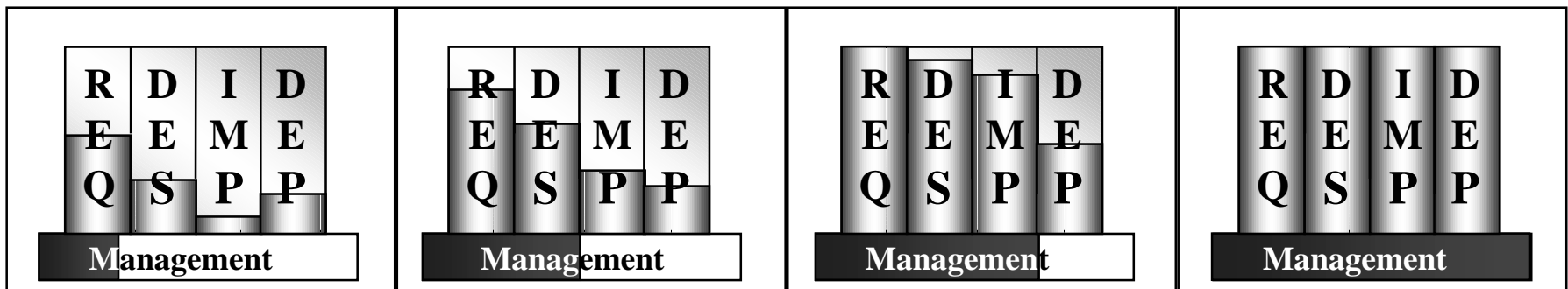


Periodic synchronization of stakeholder expectations

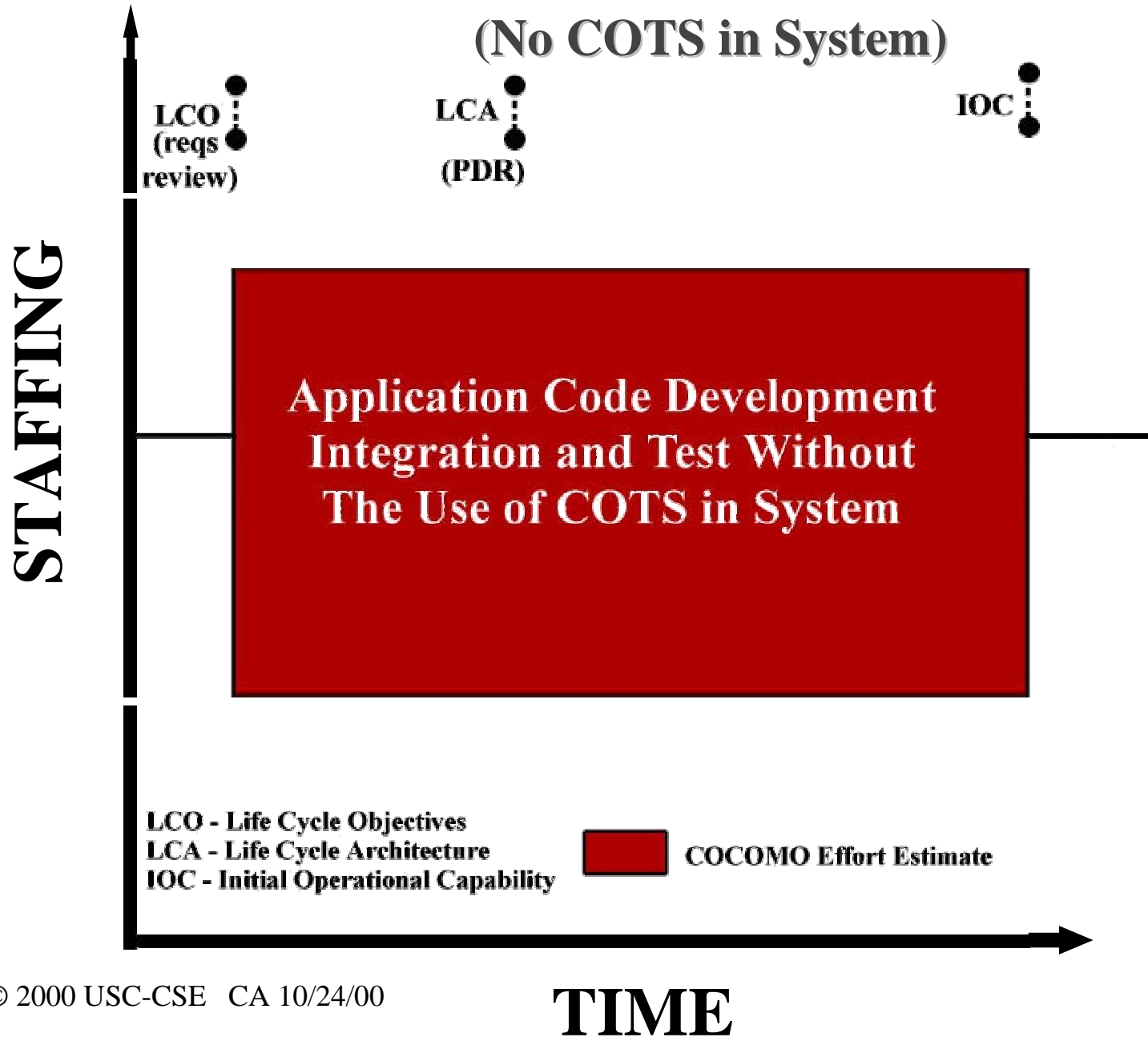
Objectory Information Set Evolution



Feasibility Iterations	<small>LCO</small>	Architecture Iterations	<small>LCA</small>	Usable Iterations	<small>IOC</small>	Product Releases
-----------------------------------	--------------------	------------------------------------	--------------------	------------------------------	--------------------	-----------------------------



COCOMO Cost Sources (No COTS in System)

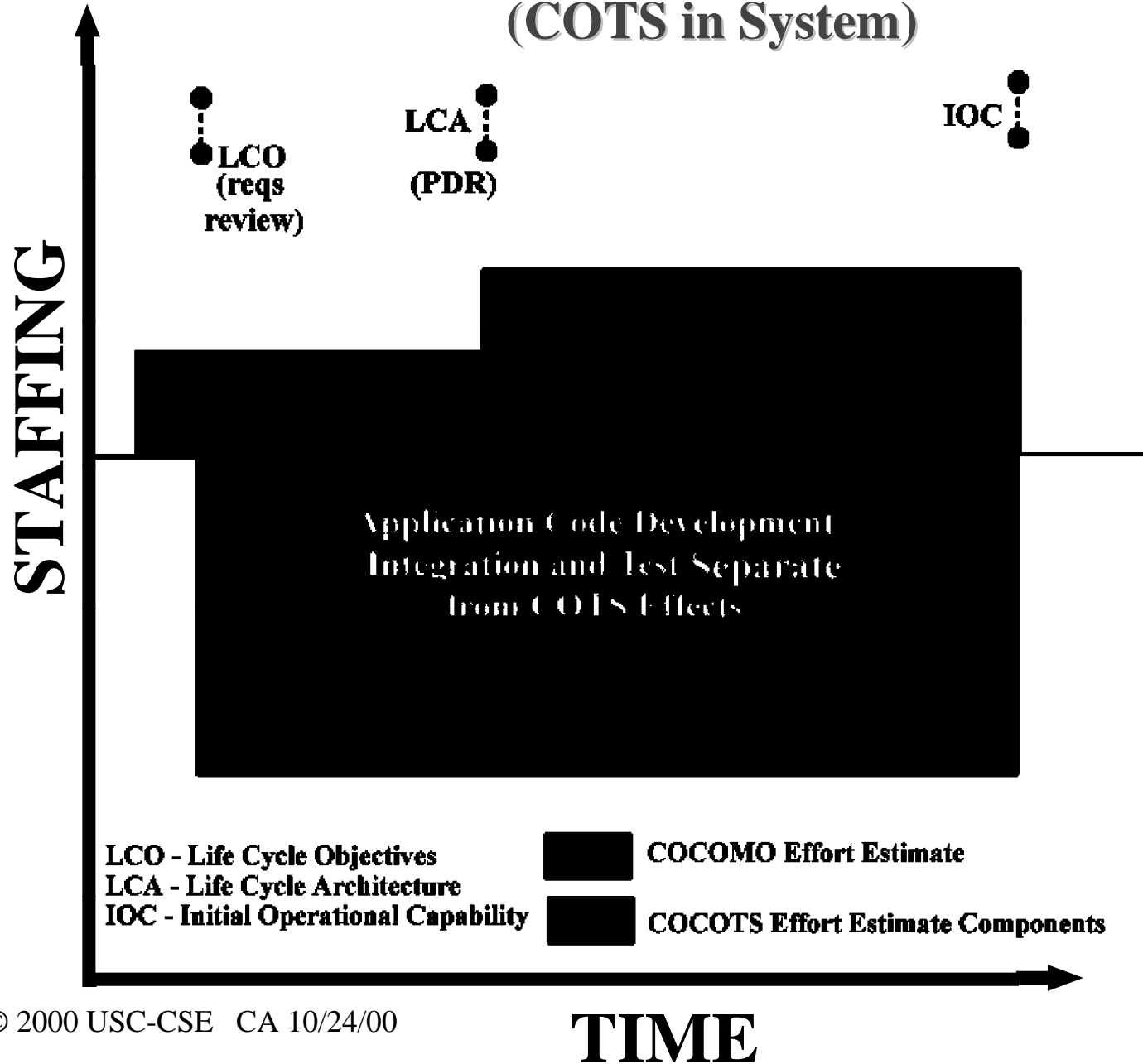




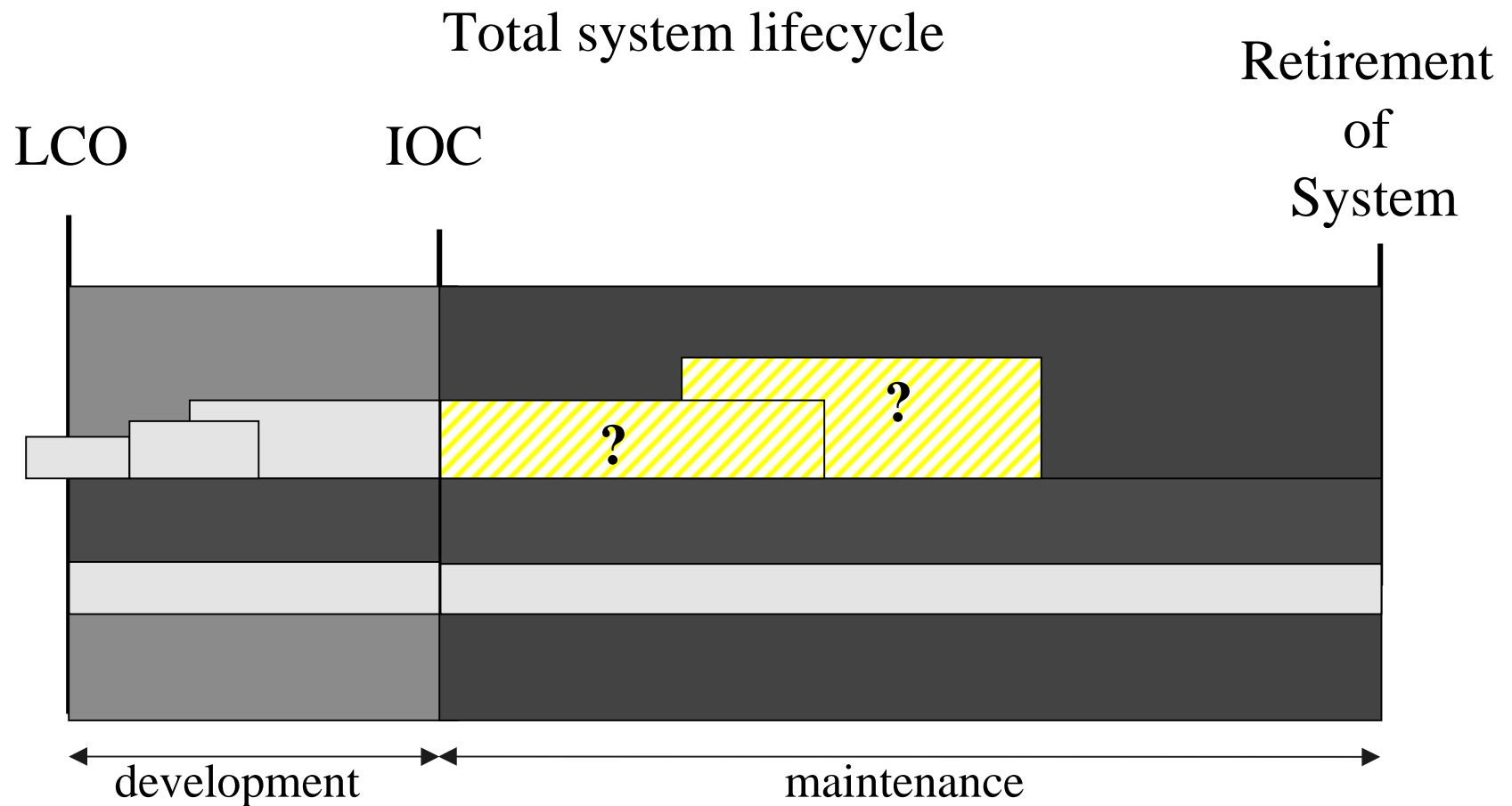
COTS Integration Sources of Effort

- COTS Assessment (pre- and post- commitment)
 - Of functionality, performance, interoperability, etc.
- COTS Tailoring and Tuning
 - Effects of platform, other COTS products
- Glue Code Development
 - Similar to other COCOMO II estimation
- Application Volatility Due to COTS
 - COTS volatility, shortfalls, learning curve
- Added Application V&V Effort
 - COTS option and stress testing
 - Debugging complications, incorrect fixes

COCOMO vs. COCOTS Cost Sources (COTS in System)



COCOMO vs. COCOTS Cost Sources (COTS in System)





COTS Integration Cost Sources:

1) Assessment

Initial Filtering Effort (IFE) =

$$\sum_{\text{Over all Classes}} \left(\# \text{ COTS Filtered} \right) \left(\frac{\text{Median Filtering Effort}}{\text{class}} \right)$$

Detailed Assessment Effort (DAE) =

$$\sum_{\text{Over all Classes}} \left(\# \text{ COTS Detailed} \right) \left(\frac{\text{Median Detailed Effort}}{\text{class}} \right)$$

Final Project Assessment Effort (FPAE) = IFE + DAE



COTS Integration Cost Sources:

1) Assessment - Assessment Attributes

Correctness		Understandability		Portability
Accuracy		Documentation quality		Portability
Correctness		Simplicity		
		Testability		Functionality
Availability/Robustness				Functionality
Availability		Ease of use		
Fail safe		Usability/Human Factors		Price
Fail soft				Initial purchase/lease
Fault tolerance		Version Compatibility		Recurring costs
Input error tolerance		Downward compatibility		
Redundancy		Upward compatibility		Maturity
Reliability				Product Maturity
Robustness		Inter-component Compatibility		Vendor Maturity
Safety		Compatibility with other components		
		Interoperability		Vendor Support
Security				Response time for critical problems
Security (Access related)		Flexibility		Support
Security (sabotage related)		Extendability		Warranty
		Flexibility		
Product Performance				User Training
Execution performance		Installation/Upgrade Ease		User training
Information/data capacity		Installation Ease		
Precision		Upgrade/Refresh ease		Vendor Concessions
Memory performance				Willingness to escrow source code
Response time				Willingness to make modifications
Throughput				



COTS Integration Cost Sources:

2) Tailoring

Project Tailoring Effort (PTE) =

$$\sum_{\text{Over all Classes}} \left[\left(\begin{array}{l} \# \text{ COTS Candidates} \\ \text{Tailored in Class} \end{array} \right) \left(\begin{array}{l} \text{Average Tailoring} \\ \text{Effort for Class} \end{array} \right) \cdot \text{TCQ} \right]$$

– *where TCQ = Tailoring Complexity Qualifier;*
with ratings Very Low, Low, Nominal, High, Very High



COTS Integration Cost Sources:

2) Tailoring - Dimensions of Tailoring Difficulty

Tailoring Activities & Aids	Individual Activity & Aid Complexity Ratings					Corresponding Points
	Very Low (point value = 1)	Low (point value = 2)	Nominal (point value = 3)	High (point value = 4)	Very High (point value = 5)	
Parameter Specification	Zero to 50 parms to be initialized.	51 to 100 parms to be initialized.	101 to 500 parms to be initialized.	501 to 1000 parms to be initialized.	1001 or more parms to be initialized.	-----
Script Writing	Menu driven; 1 to 5 line scripts; 1 to 5 scripts needed.	Menu driven; 6 to 10 line scripts; 6 to 15 scripts needed.	Hand written; 11 to 25 line scripts; 16 to 30 scripts needed.	Hand written; 26 to 50 line scripts; 31 to 50 scripts needed.	Hand written; 51 or more line scripts; 51 or more scripts needed.	-----
I/O Report & GUI Screen Specification & Layout	Automated or standard templates used; 1 to 5 reports/screens needed.	Automated or standard templates used; 6 to 15 reports/screens needed.	Automated or standard templates used; 16 to 25 reports/screens needed.	Hand written or custom designed; 26 to 50 reports/screens needed.	Hand written or custom designed; 51 or more reports/screens needed.	-----
Security/Access Protocol Initialization & Set-up	1 security level; 1 to 20 user profiles; 1 input screen/user.	2 security levels 21 to 50 user profiles; 2 input screens/user.	3 security levels 51 to 75 user profiles; 3 input screens/user.	4 security levels 76 to 100 user profiles; 4 input screens/user.	5 or more security levels 101 or more user profiles; 5 or more input screens/user.	-----
Availability of COTS Tailoring Tools	Tools available.	N/A	N/A	N/A	No tools available.	-----

Total Point Score = _____



COTS Integration Cost Sources:

3) Glue Code Development and Test

$$\text{Total Effort} = A \cdot [(\text{size})(1 + \text{breakage})]^B \cdot \prod (\text{effort multipliers})$$

- **A** - a linear scaling constant
- **Size** - of the glue code in SLOC or FP
- **Breakage** - of the glue code due to change in requirements and/or COTS volatility
- **Effort Multipliers** - 13 parameters, each with settings ranging VL to VH
- **B** - an architectural scale factor with settings VL to VH



COTS Integration Cost Sources:

3) Glue Code Development and Test - Glue Code Cost Drivers

Personnel Drivers

- 1) ACIEP - COTS Integrator Experience with Product
- 2) ACIPC - COTS Integrator Personnel Capability
- 3) AXCIP - Integrator Experience with COTS Integration Processes
- 4) APCON - Integrator Personnel Continuity

COTS Component Drivers

- 5) ACPMT - COTS Product Maturity
- 6) ACSEW - COTS Supplier Product Extension Willingness
- 7) APCPX - COTS Product Interface Complexity
- 8) ACPPS - COTS Supplier Product Support
- 9) ACPTD - COTS Supplier Provided Training and Documentation

Application/System Drivers

- 10) ACREL - Constraints on Application System/Subsystem Reliability
- 11) AACPX - Application Interface Complexity
- 12) ACPER - Constraints on COTS Technical Performance
- 13) ASPRT - Application System Portability

Nonlinear Scale Factor

- 1) AAREN - Application Architectural Engineering



COTS Integration Cost Sources:

4) Increased Application Effort Due to COTS Volatility

Approximate Model:

$$\text{Total Effort} = (\text{Application Effort}) \cdot \left[\frac{\text{BRAK COTS}}{100} \right] \cdot (\text{EAF})_{\text{COTS}}$$

Detailed Model with COCOMO II Parameters:

$$\text{Total Effort} = (\text{Application Effort}) \cdot \left[\left(1 + \frac{\text{BRAK COTS}}{1 + \text{BRAK}} \right)^{1.01 + \Sigma} - 1 \right] \cdot (\text{EAF})_{\text{COTS}}$$

BRAK COTS: % application code breakage due to COTS volatility

BRAK : % application code breakage otherwise

Σ : COCOMO II scale factor

EAF : Effort Adjustment Factor (product of effort multipliers)



COTS Integration Cost Sources:

*4) Increased Application Effort Due to COTS Volatility
- COCOMO II Scale Factors*

Scale Factor	Very Low	Low	Nominal	High	Very High	Extra High
Precedentedness	thoroughly unprecedented	Largely unprecedented	somewhat unprecedented	generally familiar	largely familiar	thoroughly familiar
Development Flexibility	rigorous	Occasional Relaxation	some relaxation	general conformity	some conformity	general goals
Architecture/Risk Resolution	little (20%)	some (40%)	often (60%)	generally (75%)	mostly (90%)	full (100%)
Team Cohesion	some difficult interactions	Basically cooperative interactions	largely cooperative	highly cooperative	seamless interactions	N/A
Process Maturity	Chaos	CMM Level 1	CMM Level 2	CMM Level 3	CMM Level 4	CMM Level 5

* percentage of module interfaces specified, percentage of significant risks eliminated.



Total COTS Integration Cost Estimate

**Total Integration Effort (in Person-Months) =
Assessment Effort + Tailoring Effort + Glue Code Effort + Volatility Effort**

where

Assessment Effort = Filtering Effort + Final Selection Effort

**Total integration Cost =
(Total Integration Effort) • (\$\$/Person-Month)**

Latest Calibration Results

10.23.2000 Assessment - Initial Filtering Effort										
			No.	Median						
Program	Project	COTS Class	Candidates in class filtered	IFE for class	IFE for Class (PM)			Total Program IFE (PM)		
					Est.	Rept.	%Re rr	Est.	Rept.	%Re rr
A	A	databases	22	1.909	41.998	18	133.32%	41.998	18	133.32%
B	B	databases	0	1.909	0	0	n/a	0	0	n/a
C	C1	GUIs/GUIbuilders	0	0.1	0	0	n/a	0	0	n/a
	C2	operating systems	0	0.5	0	0	n/a			
	C3	databases	0	1.909	0	0	n/a			
D	D	network managers	11	0.273	3.003	3	0.10%	3.003	3	0.10%
G	G5	operating systems	12	0.5	6	6	0.00%	6	6	0.00%
H	H1	GUIs/GUIbuilders	0	0.1	0	0	n/a	0	0	n/a
	H2	operating systems	0	0.5	0	0	n/a			
I	I	databases	3	1.909	5.727	9	-36.37%	5.727	9	-36.37%
J	J	operating systems	0	0.5	0	0	n/a	0	0	n/a
K	K1	GUIs/GUIbuilders	0	0.1	0	0	n/a	0	0	n/a
	K3	network managers	0	0.273	0	0	n/a			
	K4	operating systems	0	0.5	0	0	n/a			
L	L	GUIs/GUIbuilders	15	0.1	1.5	1.5	0.00%	1.5	1.5	0.00%
M	M2	emulators	16	0.078	1.248	1.25	-0.16%	1.248	1.25	-0.16%
N	N3	GUIs/GUIbuilders	0	0.1	0	0	n/a	5.5	2	175.00%
	N4	operating systems	0	0.5	0	0	n/a			
	N5	operating systems	11	0.5	5.5	2	175.00%			
Q	Q	databases	4	1.909	7.636	2	281.80%	7.636	2	281.80%
R	R1	data conversion packages	2	0.75	1.5	1.5	0.00%	23.681	57.5	-58.82%
	R2	operating systems	10	0.5	5	16	-68.75%			
	R3	databases	9	1.909	17.181	40	-57.05%			
S	S1	databases	0	1.909	0	0	n/a	12	12	0.00%
	S3	telemetry processing	6	2	12	12	0.00%			

10.23.2000 Assessment - Detailed Assessment Effort										
			No.	Median						
			Candidates in	DAE for	DAE for Class (PM)			Total Program DAE (PM)		
Program	Project	COTS Class	class detailed	class	Est.	Rept.	%Re rr	Est.	Rept.	%Re rr
A	A	databases	12	5.75	69	69	0.00%	69	69	0.00%
B	B	databases	3	5.75	17.25	40	-56.88%	17.25	40	-56.88%
C	C1	GUIs/GUIbuilders	2	0.667	1.334	1	33.40%	19.584	4	389.60%
	C2	operating systems	2	0.5	1	1	0.00%			
	C3	databases	3	5.75	17.25	2	762.50%			
D	D	network managers	6	7.165	42.99	66	-34.86%	42.99	66	-34.86%
G	G5	operating systems	12	0.5	6	1	500.00%	6	1	500.00%
H	H1	GUIs/GUIbuilders	3	0.667	2.001	5	-59.98%	5.001	6	-16.65%
	H2	operating systems	6	0.5	3	1	200.00%			
I	I	databases	2	5.75	11.5	12	-4.17%	11.5	12	-4.17%
J	J	operating systems	29	0.5	14.5	24	-39.58%	14.5	24	-39.58%
K	K1	GUIs/GUIbuilders	2	0.667	1.334	2	-33.30%	24.329	18	35.16%
	K3	network managers	3	7.165	21.495	10	114.95%			
	K4	operating systems	3	0.5	1.5	6	-75.00%			
L	L	GUIs/GUIbuilders	9	0.667	6.003	6	0.05%	6.003	6	0.05%
M	M2	emulators	0	0	0	0	n/a	0	0	n/a
N	N3	GUIs/GUIbuilders	4	0.667	2.668	2	33.40%	4.168	2.5	66.72%
	N4	operating systems	3	0.5	1.5	0.5	200.00%			
	N5	operating systems	0	0.5	0	0	n/a			
Q	Q	databases	1	5.75	5.75	4	43.75%	5.75	4	43.75%
R	R1	data conversion packages	0	0	0	0	n/a	6.25	90	-93.06%
	R2	operating systems	1	0.5	0.5	15	-96.67%			
	R3	databases	1	5.75	5.75	75	-92.33%			
S	S1	databases	2	5.75	11.5	1	1050.00%	56.53	46	22.89%
	S3	telemetry processing	19	2.37	45.03	45	0.07%			

10.23.2000 Assessment - Final Project Assessment Effort							
	IFE (PM)		DAE (PM)		FPAE (PM)		
<u>Program</u>	<u>Est.</u>	<u>Rept.</u>	<u>Est.</u>	<u>Rept.</u>	<u>Est.</u>	<u>Rept.</u>	<u>%Re rr</u>
R	23.681	57.5	6.25	90	29.931	147.5	-79.71%
B	0	0	17.25	40	17.25	40	-56.88%
J	0	0	14.5	24	14.5	24	-39.58%
D	3.003	3	42.99	66	45.993	69	-33.34%
I	5.727	9	11.5	12	17.227	21	-17.97%
H	0	0	5.001	6	5.001	6	-16.65%
M	1.248	1.25	0	0	1.248	1.25	-0.16%
L	1.5	1.5	6.003	6	7.503	7.5	0.04%
S	12	12	56.53	46	68.53	58	18.16%
A	41.998	18	69	69	110.998	87	27.58%
K	0	0	24.329	18	24.329	18	35.16%
G	6	6	6	1	12	7	71.43%
N	5.5	2	4.168	2.5	9.668	4.5	114.84%
Q	7.636	2	5.75	4	13.386	6	123.10%
C	0	0	19.584	4	19.584	4	389.60%
9 out of 15 (60%) of Assessment estimates within (+/-) 40% of actuals.							



Project	COTS class	Data points in class	Reported PTE			TCQ Rating	TCQ Multiplier	Duration Months
			No.	PM				
			Tailored	Avg	Total			
BB1	back office retail	1	1	3	3	VL	0.69	2
N1	communications protocol	1	1	1	1	L	0.83	0
P1	DBMS	3	3	1	3	N	1.00	3
A			7	38.29	268	H	1.21	6
R3			1	640	640	H	1.21	16
HH1	device drivers	3	4	2	2	N	1.00	12
P2			2	3	6	VL	0.69	12
K1 (K1,2,3)			5	18	90	H	1.21	24
N2	disk array	1	1	4	4	L	0.83	n/g
M2	emulator	1	1	3	3	N	1.00	10
L	GIU	4	3	62	186	H	1.21	24
T1			1	4	4	VL	0.69	6
N3			1	1	1	N	1.00	0
P3			1	24	24	VL	0.69	36
J	network management	3	3	12.67	38	N	1.00	17
D			1	12	12	N	1.00	6
S3			4	45.75	183	H	1.21	27
T2	OS	9	1	8	8	L	0.83	6
H2			6	4	24	L	0.83	72
N4			1	2	2	N	1.00	n/g
OO4			13	0.923	12	N	1.00	120
P4			3	1	3	L	0.83	12
G5			4	2.75	11	N	1.00	60
N5			4	2	8	N	1.00	20
Q			2	2	4	N	1.00	20
U			3	16.67	50	N	1.00	18
P5	Report generator	1	1	6	6	VL	0.69	12
Totals		27	78	20.46	1596			541
		VL	L	N	H	VH		
TCQ values		0.69	0.83	1.00	1.21	1.45		

10.23.2000 Tailoring - Project Tailoring Effort												
				<i>Median</i>								
				No.	<i>TE for</i>	<i>TCQ</i>	<i>TE for Class PM</i>			<i>Total Program TE PM</i>		
<u>Program</u>	<u>Project</u>	<u>COTS Class</u>	<u>Tailored</u>	<u>class</u>	<u>Multiplier</u>	<u>Est.</u>	<u>Rept.</u>	<u>%Re rr</u>	<u>Est.</u>	<u>Rept.</u>	<u>%Re rr</u>	
A	A	DBMS	7	38.29	1.21	324.32	268	21.01%	324.32	268	21.01%	
B	BB1	back office retail	1	3	0.69	2.07	3	-31.00%	2.07	3	-31.00%	
D	D	network management	1	12.67	1	12.67	12	5.58%	12.67	12	5.58%	
G	G5	OS	4	2	1	8	11	-27.27%	8	11	-27.27%	
H	H2	OS	6	2	0.83	9.96	24	-58.50%	21.96	26	-15.54%	
	HH1	device drivers	4	3	1	12	2	500.00%				
J	J	network management	3	12.67	1	38.01	38	0.03%	38.01	38	0.03%	
K	K1 (K1,2,3)	device drivers	5	3	1.21	18.15	90	-79.83%	18.15	90	-79.83%	
L	L	GIU	3	14	1.21	50.82	186	-72.68%	50.82	186	-72.68%	
M	M2	emulator	1	3	1	3	3	0.00%	3	3	0.00%	
N	N1	communications protocol	1	1	0.83	0.83	1	-17.00%	28.15	16	75.94%	
	N2	disk array	1	4	0.83	3.32	4	-17.00%				
	N3	GIU	1	14	1	14	1	1300.00%				
	N4	OS	1	2	1	2	2	0.00%				
	N5	OS	4	2	1	8	8	0.00%				
O	OO4	OS	13	2	1	26	12	116.67%	26	12	116.67%	
P	P1	DBMS	3	38.29	1	114.87	3	3729.00%	137.79	42	228.07%	
	P2	device drivers	2	3	0.69	4.14	6	-31.00%				
	P3	GIU	1	14	0.69	9.66	24	-59.75%				
	P4	OS	3	2	0.83	4.98	3	66.00%				
	P5	Report generator	1	6	0.69	4.14	6	-31.00%				
Q	Q	OS	2	2	1	4	4	0.00%	4	4	0.00%	
R	R3	DBMS	1	38.29	1.21	46.331	640	-92.76%	46.331	640	-92.76%	
S	S3	network management	4	12.67	1.21	61.323	183	-66.49%	61.323	183	-66.49%	
T	T1	GIU	1	14	0.69	9.66	4	141.50%	11.32	12	-5.67%	
	T2	OS	1	2	0.83	1.66	8	-79.25%				
U	U	OS	3	2	1	6	50	-88.00%	6	50	-88.00%	

10.23.2000 Tailoring - Project Tailoring Effort			
	Total Program PTE PM		
Program	Est.	Rept.	%Re rr
R	46.3309	640	-92.76%
U	6	50	-88.00%
K	18.15	90	-79.83%
L	50.82	186	-72.68%
S	61.3228	183	-66.49%
B	2.07	3	-31.00%
G	8	11	-27.27%
H	21.96	26	-15.54%
T	11.32	12	-5.67%
M	3	3	0.00%
Q	4	4	0.00%
J	38.01	38	0.03%
D	12.67	12	5.58%
A	324.3163	268	21.01%
N	28.15	16	75.94%
O	26	12	116.67%
P	137.79	42	228.07%
9 out of 17 (53%) of Tailoring estimates within (+/-) 40% of actuals.			

10.23.2000 Glue Code - Project Glue Code Effort												
PROJECT	A	size (ksloc)	% breakage	size *(1+break) eff size (ksloc)	AAREN RATING	AAREN VALUE	1+.04(aaren) B	xEAF	PM (est) (rpt'd)		%Re rr	
A	2.33	7	15.00%	8.05	N+0.50	1.5	1.06	0.20	4.35	12	-63.75%	
BB1	2.33	390	20.00%	468	H	1	1.04	0.51	711.82	60	1086.36%	
D	2.33	6.15	1.00%	6.2115	VH	0	1	0.34	4.94	74	-93.32%	
E6	2.33	60	10.00%	66	H	1	1.04	1.08	196.43	250	-21.43%	
G3	2.33	50	0.00%	50	H	1	1.04	0.53	72.08	84	-14.19%	
HH1	2.33	0.5	100.00%	1	N+0.25	1.75	1.07	0.34	0.79	6	-86.90%	
J	2.33	1.2	0.00%	1.2	H	1	1.04	0.29	0.83	1	-17.27%	
K1 (K1,2,3,4)	2.33	1	50.00%	1.5	H	1	1.04	0.44	1.58	4	-60.49%	
L	2.33	1	30.00%	1.3	VH	0	1	0.59	1.78	7	-74.64%	
M1	2.33	5	0.00%	5	N+0.75	1.25	1.05	3.07	38.76	81	-52.15%	
N3	2.33	20	0.00%	20	H	1	1.04	0.32	16.94	12	41.14%	
OO4	2.33	125	25.00%	156.25	VH	0	1	1.63	595.03	1411	-57.83%	
P1	2.33	10	0.00%	10	H	1	1.04	0.98	25.06	12	108.87%	
P2	2.33	3	0.00%	3	H	1	1.04	2.49	18.16	6	202.61%	
P3	2.33	30	0.00%	30	H	1	1.04	0.42	33.46	60	-44.24%	
P4	2.33	0.1	100.00%	0.2	H	1	1.04	0.46	0.20	6	-96.68%	
P5	2.33	5	0.00%	5	H	1	1.04	1.01	12.51	12	4.23%	
Q	2.33	50	0.00%	50	VH	0	1	0.67	78.50	72	9.03%	
S3	2.33	4.2	0.00%	4.2	H	1	1.04	2.88	29.84	18	65.79%	
U	2.33	25	0.00%	25	H	1	1.04	1.85	122.74	75	63.65%	

10.23.2000 Glue Code - Project Glue Code Effort			
	Total Glue Code PM		
<u>PROGRAM</u>	<u>(e s t)</u>	<u>(r e p o r t e d)</u>	<u>%R e r r</u>
D	4.94	74	-93.32%
H	0.79	6	-86.90%
L	1.78	7	-74.64%
A	4.35	12	-63.75%
K	1.58	4	-60.49%
O	595.03	1411	-57.83%
M	38.76	81	-52.15%
E	196.43	250	-21.43%
J	0.83	1	-17.27%
G	72.08	84	-14.19%
P	89.39	96	-6.89%
Q	78.50	72	9.03%
N	16.94	12	41.14%
U	122.74	75	63.65%
S	29.84	18	65.79%
B	711.82	60	1086.36%
6 out of 16 (38%) of Glue Coding estimates within (+/-) 42% of actuals.			

10.23.2000 Total Program COTS Integration Effort												
Total FPAE (PM)				Total PTE (PM)				Total Glue Code (PM)				
Program	Est.	Rept.	%Re rr	Program	Est.	Rept.	%Re rr	Program	Est.	Rept.	%Re rr	
A	110.998	87	27.58%	A	324.3163	268	21.01%	A	4.35	12	-63.75%	
B	17.25	40	-56.88%	B	2.07	3	-31.00%	B	711.82	60	1086.36%	
C	19.584	4	389.60%									
D	45.993	69	-33.34%	D	12.67	12	5.58%	D	4.94	74	-93.32%	
								E	196.43	250	-21.43%	
G	12	7	71.43%	G	8	11	-27.27%	G	72.08	84	-14.19%	
H	5.001	6	-16.65%	H	21.96	26	-15.54%	H	0.79	6	-86.90%	
I	17.227	21	-17.97%									
J	14.5	24	-39.58%	J	38.01	38	0.03%	J	0.83	1	-17.27%	
K	24.329	18	35.16%	K	18.15	90	-79.83%	K	1.58	4	-60.49%	
L	7.503	7.5	0.04%	L	50.82	186	-72.68%	L	1.78	7	-74.64%	
M	1.248	1.25	-0.16%	M	3	3	0.00%	M	38.76	81	-52.15%	
N	9.668	4.5	114.84%	N	28.15	16	75.94%	N	16.94	12	41.14%	
				O	26	12	116.67%	O	595.03	1411	-57.83%	
				P	137.79	42	228.07%	P	89.39	96	-6.89%	
Q	13.386	6	123.10%	Q	4	4	0.00%	Q	78.50	72	9.03%	
R	29.931	147.5	-79.71%	R	46.3309	640	-92.76%					
S	68.53	58	18.16%	S	61.3228	183	-66.49%	S	29.84	18	65.79%	
				T	11.32	12	-5.67%					
				U	6	50	-88.00%	U	122.74	75	63.65%	

10.23.2000 Total Program COTS Integration Effort				
	Total COTS IE (PM)			
Program	Est.	Rept.	%Rerr	
R	76.26	787.5	-90.32%	
L	60.10	200.5	-70.03%	
K	44.06	112	-60.66%	
D	63.61	155	-58.96%	
O	621.03	1423	-56.36%	
M	43.01	85.25	-49.55%	
S	159.69	259	-38.34%	
H	27.75	38	-26.98%	
E	196.43	250	-21.43%	
I	17.23	21	-17.97%	
J	53.34	63	-15.34%	
G	92.08	102	-9.73%	
T	11.32	12	-5.67%	
U	128.74	125	2.99%	
Q	95.89	82	16.93%	
A	439.66	367	19.80%	
P	227.18	138	64.62%	
N	54.75	32.5	68.48%	
C	19.58	4	389.60%	
B	731.14	103	609.84%	
10 out of 20 (50%) of COTS Integration estimates within (+/-) 40% of actuals.				

Tool Demo



In Conclusion: COCOTS' Most Important Aspect

- COCOTS is completely open. Regardless of whatever estimates it provides, the descriptions of the elements that have gone into the model help highlight the most important factors that should be of concern to managers and developers of software systems using COTS software components.
- It's the very essence of a "constructive" cost model:
 - one that helps an estimator better *understand the complexities* of a given software job to be done
 - by being open permits the estimator to know exactly *why* a model gives the estimate it does