Software Project Management

Lecture 20

Words of Wisdom

- Instead of worrying about things and events that you cannot control, start focusing on things and activities you can control.

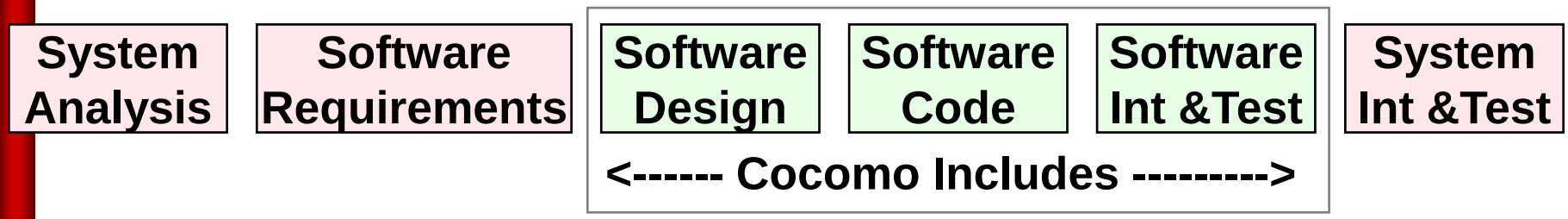
- *But verily thy Lord for those who do wrong in ignorance but afterward repent and make amends thy Lord after all this (to such) is Oft-Forgiving and Most Merciful. Al-Quran (16:119).*

Issues With Cocomo

- Amount of judgment required in determining values for cost adjustment factors and which mode applies to the software
- Your data may not match the data used to develop the model you wish to use
  - But your company may not want to collect the data needed to correlate the model
- Many models, including Cocomo, assume a basic waterfall process model
  - 30% design; 30% coding; 40% integration and test
  - Your process may not match up well

Each Model Includes Only Selected Tasks of Software Development

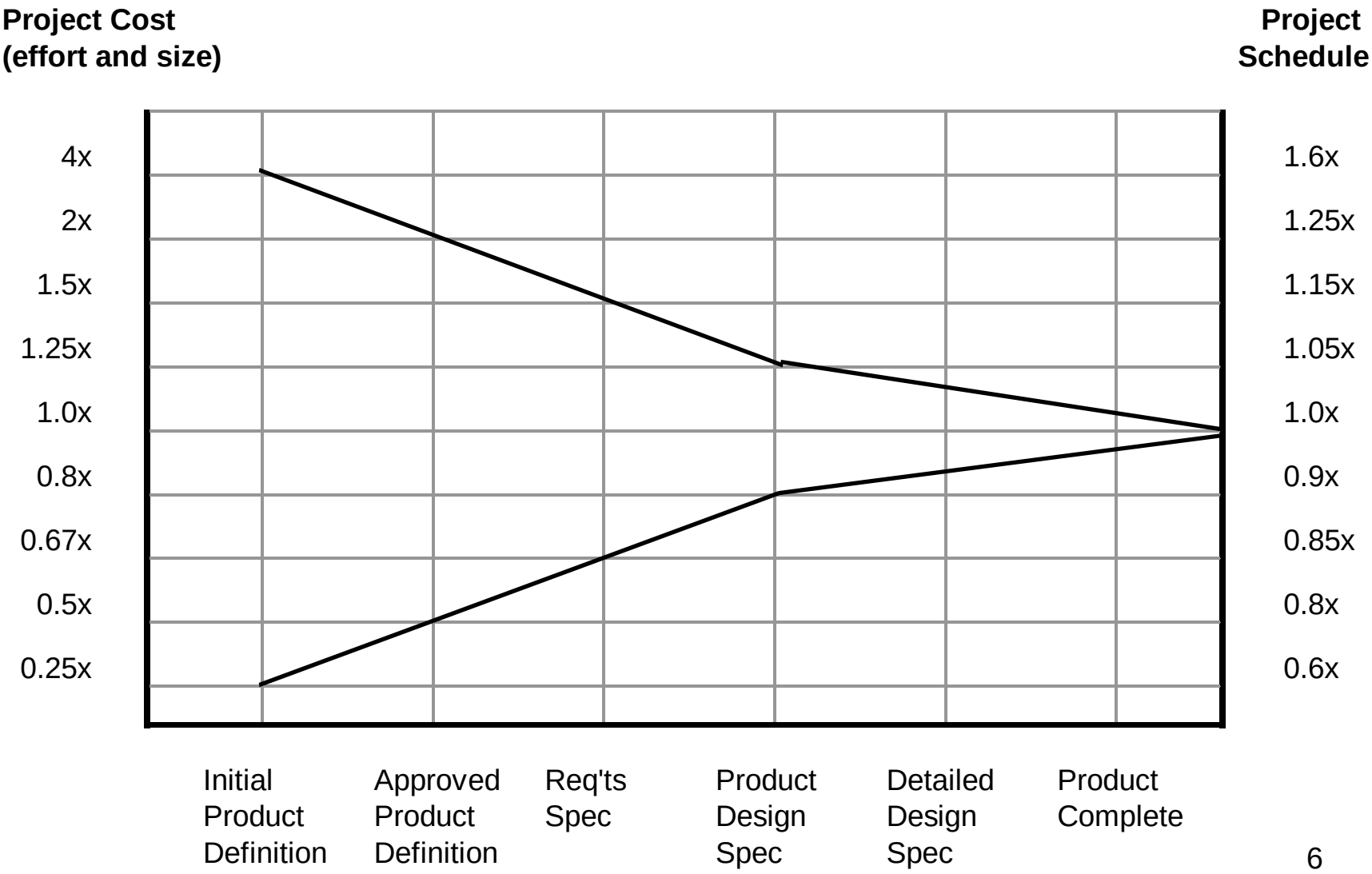
- For example, Cocomo does not include
  - System level analysis and design
  - Software requirements analysis or
  - Software support of system level integration and test



Software Practice

- End-User Programming
  - Develop Application themselves using application generators, spread sheets, query systems
  - 55 million only in USA by 2005
- Application Generators & Composition Aids
  - Packages e.g., Access, Active X
  - 0.6 million
- Application Composition
  - Diverse Applications and Domain Specific components
  - 0.7 million
- System Integration
  - Software Intensive products, highly embedded large scale systems e.g., manufacturing support system
  - 0.7 million

COCOMO & Cone of Uncertainty



COCOMO II Models

- Application Composition
  - Involves prototyping efforts to resolve potential high risk
- Early Design
  - Involves exploration – not enough is generally known
- Post-Architecture
  - Involves the actual development and maintenance

Application Composition & Reuse

- eKSLOC = aKSLOC * (1 – AT/100) * AAM
- eKSLOC = equivalent K numbers of lines of new code
- aKSLOC = the K number of lines of reusable code which must be modified
- AT = Percentage of the Adapted SLOC that is re-engineered by automatic translation
- AAM = Adaption Adjustment Modifier
  - = [AA+AAF+(SU x UNFM)]/100 for AAF>0.5
  - = [AA+AAF+(1+(0.02) x SU x UNFM)]/100 AAF<=0.5
  - AA = Assessment & Assimilation (0 – 8)
  - AAF = Adoption Adjustment Factor
  - SU = Software Understanding (50 – 10)
  - UNFM = Programmers Unfamiliarity (0-1)

Application Composition & Reuse

- AAF = Adoption Adjustment Factor
  - = 0.4DM +0.3CM +0.3IM
  - DM = Percentage of design modification
  - CM = Percentage of code modification
  - IM = Percentage of integration effort required
- COCOMO II Sizing Equation
  - Size = (1+ REVL/100) * (New KSLOC + eKSLOC)

Early Design

- Nominal Effort (PMn) = **A** x Size^{B}
- **A** = 2.94
- Size in KDSI (or KSLOC)
- Calculate uFPs and then convert it to KDSI
- **B** = 0.91 + 0.01 x Σ *SFs*
- Five *scaling factors*
  - Precedentness (Very low … Extra High)
  - Development flexibility
  - Architecture/Risk Resolution
  - Team cohesion
  - Process maturity

Scaling Factors

- Precedentness
  - Organizational understanding of product objectives
  - Experience of working with related software systems
  - Concurrent development of associated new hardware and operational procedures
  - Need for innovative processing algorithms
- Development flexibility
  - Need for software conformance with pre-established requirements
  - Need for software with external interface specifications
  - Early completion

Scaling Factors

- Architecture/Risk Resolution
  - Risk Management Plan
  - Number and criticality of risk items
  - Percentage of development schedule devoted to establish architecture
  - Percentage of required top software architects available to project
  - Tools support available
- Team cohesion
  - Consistency of stakeholder objectives
  - Ability and willingness of stakeholder to accommodate other stake holders' objectives
  - Experience of stakeholders as a team
  - Stakeholder teambuilding to achieve shared vision

Scaling Factors

- Process maturity
  - Capability Maturity Model's Level
  - CMM Level 5 = Extra High
  - CMM Level 1 (lower half) = Very Low

Scaling Factors

SF	VL	L	N	H	VH	XH
PREC	6.20	4.96	3.72	2.48	1.24	0
FLEX	5.07	4.05	3.04	2.03	1.01	0
RESL	7.07	5.65	4.24	2.83	1.41	0
TEAM	5.48	4.38	3.29	2.19	1.10	0
PMAT	7.80	6.24	4.68	3.12	1.56	0

Early Design

- Adjusted Effort (PMa) = PMn x Π EM
- The multiplier EM is based on a set of seven project and process drivers (very low … very high)
  - Product reliability and complexity (RCPX)
  - Reused required (RUSE)
  - Platform difficulty (PDIF)
  - Personnel capability (PERS)
  - Personnel experience (PREX)
  - Schedule (SCED)
  - Support facilities (FCIL)

Early Design Effort Multipliers

EM	XL	VL	L	N	H	VH	XH
RCPX	0.49	0.60	0.83	1.00	1.33	1.91	2.72
RUSE			0.95	1.00	1.07	1.15	1.24
PDIF			1.00	1.00	1.00		
PERS	2.12	1.62	1.26	1.00	0.83	0.63	0.50
PREX	1.59	1.33	1.12	1.00	0.87	0.74	0.62
FCIL	1.43	1.30	1.10	1.00	0.87	0.73	0.62
SCED		1.43	1.14	1.00	1.00	1.00	

COCOMO II
Post Architecture level

- Nominal Effort as in Early Design
- Adjusted Effort (PMa) = PMn x Π EM
- The multiplier EM is based on a set of 17 drivers (very low … very high)
  - Project
  - Platform
  - Personnel and
  - Process drivers
- Dropped
  - TURN
  - MODP

- Added
  - DOCU (Extent of Documentation)
  - RUSE (Required % of reusable components)
  - PCON (Personnel Continuity)
  - SITE (Multi-site working and quality of communication)
- Two dropped four added (= 17 Cost drivers)
- Changed
  - VIRT to PVOL (dev platform volatility)
  - VEXP to PEXP (programmer domain exp)

- Changed
  - CPLX
    - Control Operations
    - Computational Operations
    - Device dependent Operations
    - Data Management Operations
    - User Interface Management Operations
  - Very low to Extra High

Cost Drivers

- Required software reliability – RELY
- Data size (large test data) – DATA
- Product complexity - CPLX
- Developed for reusability - RUSE
- Documentation match to life-cycle need - DOCU
- Execution time constraints - TIME
- Main storage constraints - STOR
- Platform volatility - PVOL
- Analyst capability - ACAP
- Programmers' capability - PCAP
- Personnel continuity - PCON
- Application experience - APEX
- Platform experience - PLEX
- Language and tool experience - LTEX
- Use of software tools - TOOL
- Multi-site development - SITE
- Required development schedule - SCED

Effort Multipliers COCOMO II

Drivers	Very Low	Low	Normal	High	Very High	Extra High
RELY	0.82	0.92	1.00	1.10	1.26	
DATA		0.90	1.00	1.14	1.28	
CPLX	0.73	0.87	1.00	1.17	1.34	1.74
RUSE		0.95	1.00	1.07	1.15	1.24
DOCU	0.81	0.91	1.00	1.11	1.23	
TIME			1.00	1.11	1.29	1.63
STOR			1.00	1.05	1.17	1.46
PVOL		0.87	1.00	1.15	1.30	
ACAP	1.42	1.19	1.00	0.85	0.71	
PCAP		1.15	1.00	0.88	0.76	
PCON		1.12	1.00	0.90	0.81	
APEX		1.10	1.00	0.88	0.81	
PLEX		1.09	1.00	0.91	0.85	
LTEX		1.09	1.00	0.91	0.84	
TOOL		1.09	1.00	0.90	0.78	
SITE		1.09	1.00	0.93	0.86	0.80
SCED		1.14	1.00	1.00	1.00	

Schedule Estimates

- For both Early Design and Post Architecture
- TDEV = [3.67 * $(PM)^F$] *
  SCEDPercentage/100
  - PM is the estimated person months excluding SCED effort multiplier
  - F = 0.28 + 0.2 * (0.01 x Σ SF)

Project Phases

- Included
  - Product Design
  - Implementation
  - Integration & Test
  - Project Management
  - Project Configuration Management
  - Project Quality Assurance
- Not Included
  - Requirements Definition
  - Project Management activities done above the project level

Effort and Schedule Distribution

Phases	Effort	Schedule
Plans & Requirements	2%-15%	2% - 34%
Product Design	17%	24% - 28%
Programming	64% - 52%	56% - 40%
Integration and Testing	19% - 31%	20% - 32%
Transition	0% - 20%	0 – 20%

Spiral Based/RUP

Phases	Effort	Schedule
Inception	5%	10%
Elaboration	20%	30%
Construction	65%	50%
Transition	10%	10%

Q&A