

Aula 33

Professor:

Geraldo Xexéo
UFRJ

Conteúdo:

COCOMO II

Quanto Custa?

➡ Esforço

➡ Tempo de Desenvolvimento

Esforço ...

- ➡ O principal fator de custo no desenvolvimento de software é o gasto com pessoal
- ➡ Uma das principais preocupações da Engenharia de Software é determinar qual será a quantidade de pessoas e o tempo por elas dedicado a um projeto.
- ➡ Para isso usamos o conceito de **Esforço** que representa a quantidade de trabalho realizado, medido em pessoa-mês,
 - ➡ o trabalho feito por uma pessoa em um mês.

... Esforço

- ➡ Assim, podemos dizer que um sistema precisa de 4 pessoas-mês para ser realizado, ou seja, que uma pessoa trabalhando 4 meses ou 4 pessoas trabalhando um mês.
- ➡ Acontece que sistemas de informação são um pouco como bebês: não podemos ter a gestação de um bebê com nove mães em um mês.
- ➡ Na verdade, Boehm achou uma relação entre o esforço necessário e o tempo necessário para fazer um sistema, e conseqüentemente o tamanho médio da equipe.

COCOMO II

- ➡ Constructive Cost Model II
 - ▬ Segunda versão do método
- ➡ Busca calcular qual o esforço necessário e o tempo de desenvolvimento adequado de um produto de software em função do seu tamanho
 - ▬ Prevê uma gama de valores e não só um único valor

Desenvolvimento

- ➡ Barry Boehm
- ➡ Desenvolvido no Center for Software Engineering (CSE)
- ➡ University of Southern California (USC)

Porém, é algo mais

Objetivo do COCOMO II

- ➡ Ajudar as pessoas a pensar sobre o impacto que suas decisões sobre um projeto de software tem sobre o custo e o prazo de desenvolvimento

COCOMO II Apoia

- ➡ Decisões de investimento
- ➡ Calcular prazo e orçamento
- ➡ Negociar **trade-offs** de prazo, custo e desempenho
- ➡ Tomar decisões sobre o risco do projeto
- ➡ Tomar decisões sobre a melhoria de um produto de software

Tamanho?

➡ Linhas de Código...

➡ Ou, em Pontos de Função

— Convertidos em linhas de código

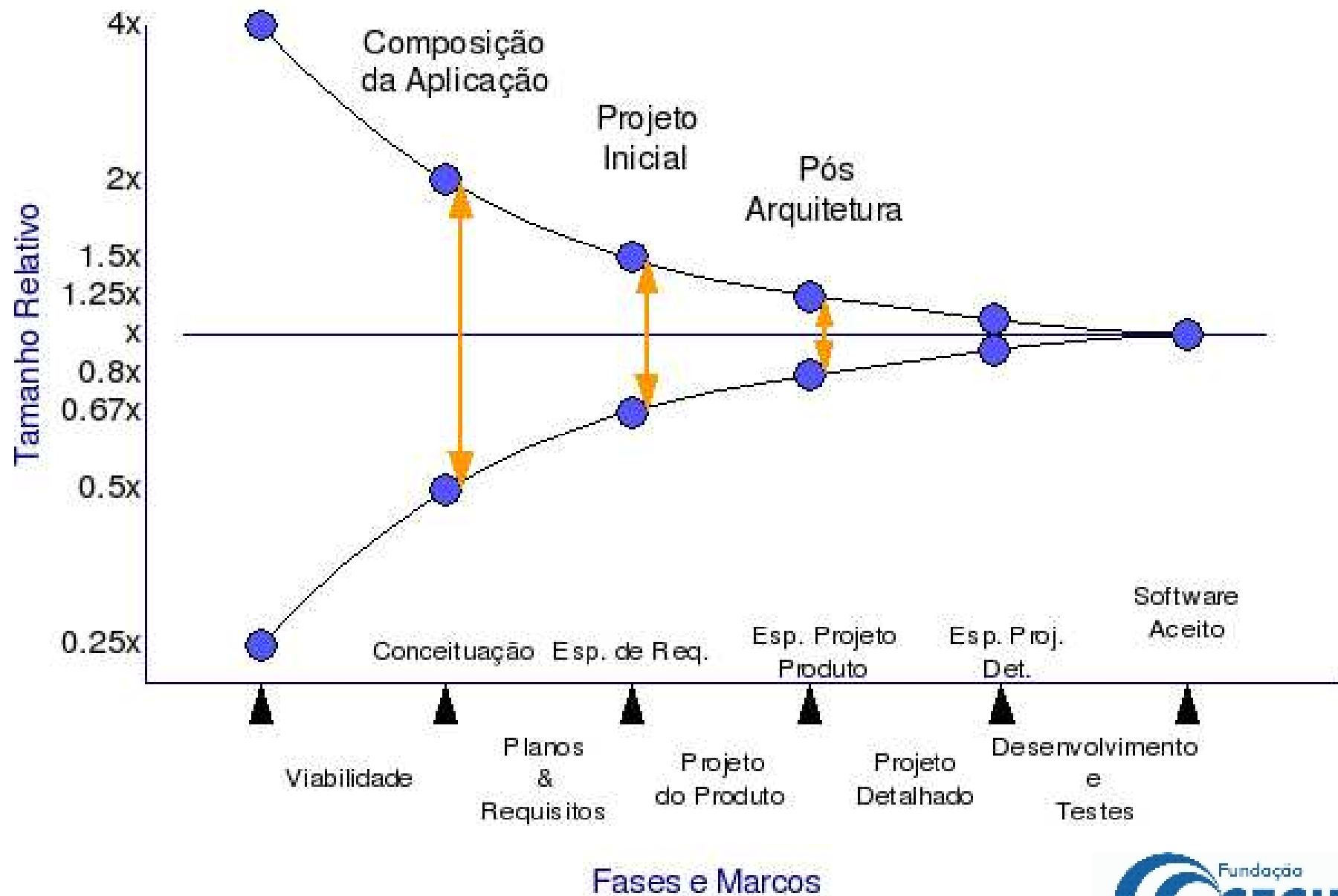
Estágios do COCOMO II

➡ Composição da Aplicação

➡ Início do Projeto

➡ Pós-Arquitetura

Estágios COCOMO II

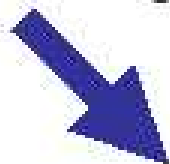


Visão Rápida

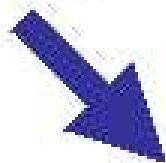


Visão Rápida

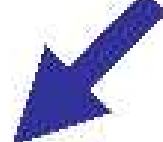
Pontos de Função



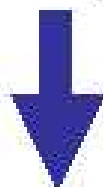
Linhas de Código



Condições do Projeto



Esforço



Tempo de Desenvolvimento

COCOMO II

- ➡ Fornece uma fórmula de previsão de Esforço e Tempo de Desenvolvimento a partir do Tamanho em Linhas de Código (previsto)
 - ▬ Tabelas de Conversão PF->LOC

Fórmula do Esforço

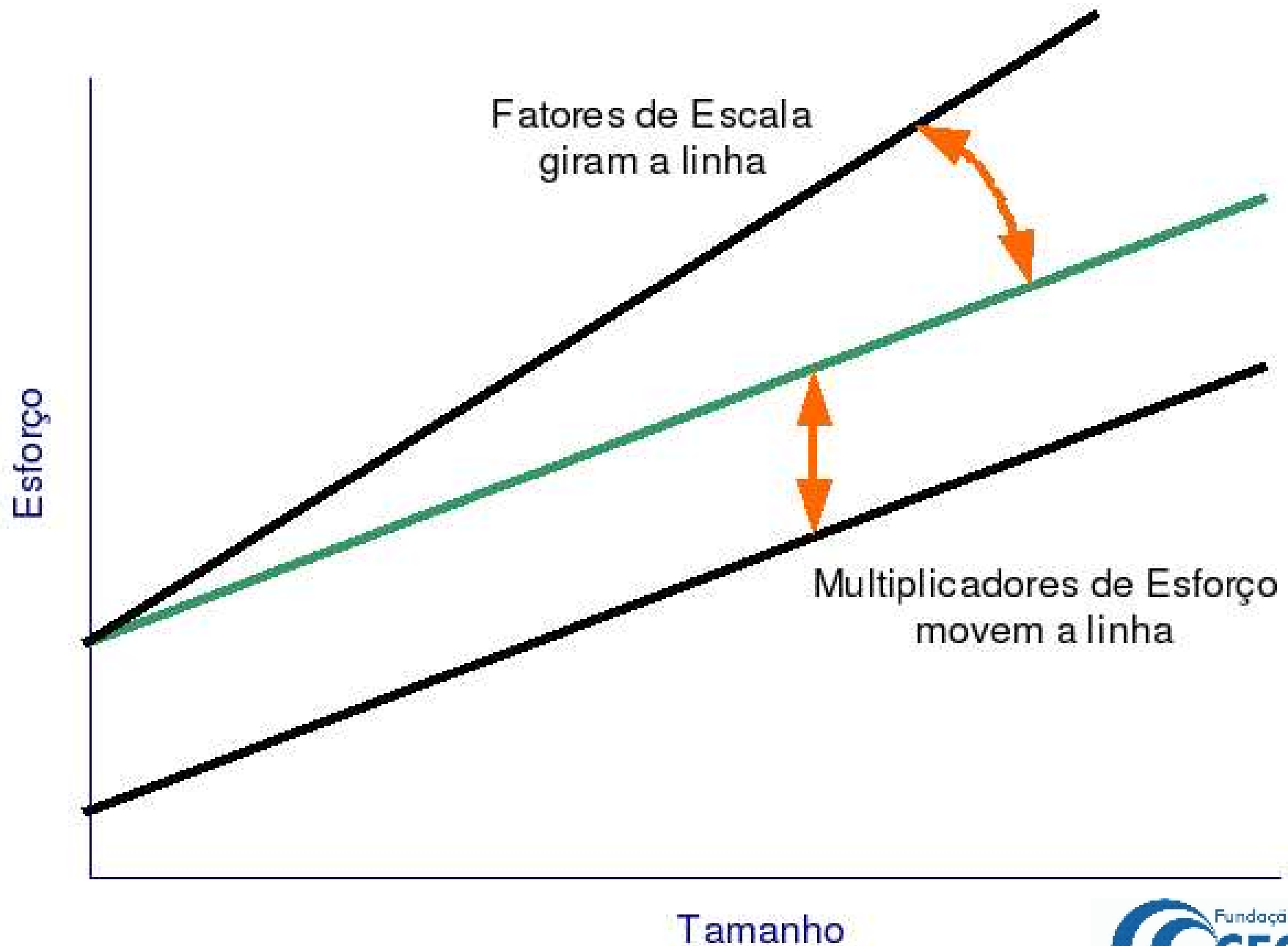
$$PM = Effort = A \times EAF \times (KSLOC)^E$$

EAF = multiplicadores de custo

E = derivado dos scale drivers

A = Constante = calibrada em 2,94

Entendendo a Curva



A Fórmula Real

$$PM = \prod_{i=1}^{17} (EM_i) \times A \times \left[\left(1 + \frac{BRAK}{100} \right) \times \text{Size} \right]^{(0.91 + 0.01 \sum_{j=1}^5 SF_j)} + \left(\frac{ASLOC \times \left(\frac{AT}{100} \right)}{ATPROD} \right)$$

where

$$\text{Size} = KNSLOCK + \left[KASLOC \times \left(\frac{100 - AT}{100} \right) \times \frac{AA + SU + 0.4 \times DM + 0.3 \times IM}{100} \right]$$

$$B = 0.91 + 0.01 \sum_{j=1}^5 SF$$

Symbol	Description
A	Constant, currently calibrated as 2.45
AA	Assessment and assimilation
ADAPT	Percentage of components adapted (represents the effort required in understanding software)
AT	Percentage of components that are automatically translated
ATPROD	Automatic translation productivity
BRAK	Breakage: Percentage of code thrown away due to requirements volatility
CM	Percentage of code modified
DM	Percentage of design modified
EM	Effort Multipliers: RELY, DATA, CPLX, RUSE, DOCU, TIME, STOR, PVOL, ACAP, PCAP, PCON, AEXP, PEXP, LTEX, TOOL, SITE
IM	Percentage of integration and test modified
KASLOC	Size of the adapted component expressed in thousands of adapted source lines of code
KNSLOC	Size of component expressed in thousands of new source lines of code
PM	Person Months of estimated effort
SF	Scale Factors: PREC, FLEX, RESL, TEAM, PMAT
SIZE	In thousands of lines of code, of new or adapted source lines of code
SU	Software understanding (zero if DM = 0 and CM = 0)

Para isso, precisamos de um software adequado

Source Line of Code (SLOC)

- ➡ Apenas linhas de código entregues como parte do produto
 - ▬ Testes e outros softwares de suporte não contam
- ➡ Apenas linhas criadas pela equipe
 - ▬ Linhas geradas automaticamente não contam
- ➡ Um SLOC é uma linha de código lógica
- ➡ Declarações contam como SLOC
- ➡ Comentários não contam como SLOC

O valor do expoente

$$E = 0.91 + 0.01 \times \sum_{i=1}^5 w_i$$

➡ w_i são os fatores de escala

➡ Economia de Escala

- ➡ Se $E < 1.0$, o software produz economia de escala
- ➡ Se $E = 1.0$, o software está balanceado quanto a economia
- ➡ Se $E > 1.0$, o software produz uma **deseconomia** de escalam

Deseconomia????

Principais fatores

- ⇒ Excesso de linhas de comunicação
- ⇒ Integração de grandes sistemas

Scale Drivers

- ➡ No modelo COCOMO II, os fatores mais importantes que contribuem para a duração e custo de projetos são os Fatores de Escala (Scale Drivers)
- ➡ Cinco Fatores de Escala determinam o expoente (E) da equação de Esforço
 - ➡ Precedência (PREC)
 - ➡ Flexibilidade do Desenvolvimento (FLEX)
 - ➡ Arquitetura e Resolução de Risco (RESL)
 - ➡ Coesão da Equipe (TEAM)
 - ➡ Maturidade do Processo (PMAT)

Fatores de Escala

- ⇒ Variam entre
- 5: Very Low
 - 4: Low
 - 3: Nominal
 - 2: High
 - 1: Very High
 - 0: Extra High

- ⇒ Quanto menor o número, menor o expoente
- ⇒ Quanto menor o expoente, maior a economia de escala
- Ou menor a deseconomia





$$E = 0.91 + 0.01 \times \sum_{i=1}^5 w_i$$

Tabela para os Scale Drivers

Fatores de Escala	VL	L	N	H	VH	EH
PREC	Totalmente novo	Muito novo	Alguma novidade	Em geral familiar	Muito familiar	Totalmente familiar
FLEX	Rigoroso	Relaxamento ocasional	Algum relaxamento	Conformidade geral	Alguma conformidade	Objetivos gerais
RESL	Pouca (20%)	Alguma (40%)	Normalmente (60%)	Geralmente (75)	Majoritariamente (90%)	Totalmente (100%)
TEAM	Interações difíceis	Algumas dificuldades de interação	Basicamente cooperativo	Muito cooperativo	Altamente Cooperativos	Interação Perfeita
PMAT	Uma média ponderada das respostas SIM a uma avaliação CMM					

Multiplicadores de Custo

Quatro Tipo de Fatores

-  Projeto
-  Equipe
-  Produto
-  Ambiente de Execução

Um Fator Extra

Fatores de Custo - Produto

- ➡ Confiabilidade Exigida do Software
- ➡ Tamanho do Banco de Dados
- ➡ Complexidade do Produto
- ➡ Exigência de Reusabilidade
- ➡ Casamento entre a documentação e as necessidades do Ciclo de Vida

Fatores de Custo - Equipe

- ➡ Capacidade dos Analistas
- ➡ Capacidade dos Programadores
- ➡ Experiência na Aplicação
- ➡ Experiência na Plataforma
- ➡ Experiência com Linguagem e Ferramenta
- ➡ Continuidade do Pessoal

Fatores de Custo - Projeto

➡ Uso de Ferramentas de Software

➡ Desenvolvimento Multi-site

Fatores de Custo - Ambiente de Execução

- ➡ Restrições de tempo de execução
- ➡ Restrições de tamanho de memória
- ➡ Volatilidade da Plataforma

Fator de Custo - Extra

➡ Prazos de desenvolvimento exigidos

Como calcular os valores

- ➡ Cada Fator de Escala ou Multiplicador de Esforço possui uma tabela que indica como calcular o seu valor
- ➡ Além disso, existem outras tabelas explicativas que ajudam posicionar o fator em uma das casas da tabela principal

E o Tempo?

$$TDEV = [C \times (PM_{NS})^{(D+0.2 \times (E-B))}] \times \frac{SCED}{100}$$

A	2.94
B	0.91
C	3.67
D	0.28



Mais Informações

- ➡ <http://sunset.usc.edu/research/COCOMOII/>
- ➡ ftp://ftp.usc.edu/pub/soft_engineering/COCOMOII/cocomo98.0/modelman.pdf
- ➡ ftp://ftp.usc.edu/pub/soft_engineering/COCOMOII/cocomo99.0/userman.pdf

O Software

Untitled - USC-COCOMO II.1999.0

File Edit View Parameters Calibrate Phase Maintenance Help

Project Name: <sample> Scale Factor Schedule

Development Model: Post Architecture

X	Module Name	Module Size	LABOR Rate (\$/month)	EFF	NOM Effort DEV	EST Effort DEV	PROD	COST	INST COST	Staff	RISK																																
<div> Total Lines of Code: <input type="text" value="0"/> <table> <tr> <td>Estimated</td> <td>Effort</td> <td>Sched</td> <td>PROD</td> <td>COST</td> <td>INST</td> <td>Staff</td> <td>RISK</td> </tr> <tr> <td>Optimistic</td> <td>0.0</td> <td>0.0</td> <td>0.0</td> <td>0.00</td> <td>0.0</td> <td>0.0</td> <td></td> </tr> <tr> <td>Most Likely</td> <td>0.0</td> <td>0.0</td> <td>0.0</td> <td>0.00</td> <td>0.0</td> <td>0.0</td> <td>0.0</td> </tr> <tr> <td>Pessimistic</td> <td>0.0</td> <td>0.0</td> <td>0.0</td> <td>0.00</td> <td>0.0</td> <td>0.0</td> <td></td> </tr> </table> </div>												Estimated	Effort	Sched	PROD	COST	INST	Staff	RISK	Optimistic	0.0	0.0	0.0	0.00	0.0	0.0		Most Likely	0.0	0.0	0.0	0.00	0.0	0.0	0.0	Pessimistic	0.0	0.0	0.0	0.00	0.0	0.0	
Estimated	Effort	Sched	PROD	COST	INST	Staff	RISK																																				
Optimistic	0.0	0.0	0.0	0.00	0.0	0.0																																					
Most Likely	0.0	0.0	0.0	0.00	0.0	0.0	0.0																																				
Pessimistic	0.0	0.0	0.0	0.00	0.0	0.0																																					

Ready

Definindo um Módulo

SLOC Input Dialog - Fluxo de Caixa

Sizing Method:

- ☐ SLOC
- ☒ Function Points
- ☐ Adaptation

Breakage: % of code thrown away due to requirements volatility
BRK: 10.00

Module Size in Function Points

Language: Object-oriented

Function Type	# of Function Points			SubTotal
	Low	Average	High	
Inputs	1	1	0	7
Outputs	0	1	0	5
Files	1	1	0	17
Interfaces	1	0	0	5
Queries	0	1	0	4
Total Unadjusted Function Points				38
Equivalent Total in SLOC				1216

OK Cancel Help

SLOC Input Dialog - Fluxo de Caixa

Sizing Method:

- ☒ SLOC
- ☐ Function Points
- ☐ Adaptation

Breakage: % of code thrown away due to requirements volatility
BRK: 10.00

Module Size in SLOC: 0

OK

SLOC Input Dialog - Fluxo de Caixa

Sizing Method:

- ☐ SLOC
- ☐ Function Points
- ☒ Adaptation

Breakage: % of code thrown away due to requirements volatility
BRK: 10.00

Adaptation:

Initial SLOC	0	
% Design Modified (DM)	100	%
% Code Modified (CM)	100	%
% Integration Modified (IM)	100	%
Software Understanding (SU)	90	100
Assessment & Animation (AA)	4	AA
Familiarity with Software	0.4	100%
% Components Automatically Translated (AT)	0	%
Automatic Translation Productivity (ATPSOD)	2400	
Computed Adaptation Adjustment Factor	100	
Computed SLOC	0	

OK Cancel Help

Definindo Fatores

EAF - Fluxo de Caixa

base + Incr % = rating

Product: RELY DATA DOCU CPLX RUSE
base NOM NOM NOM NOM NOM
Incr% 0% 0% 0% 0% 0%

Platform: TIME STOR PVOL
base NOM NOM NOM
Incr% 0% 0% 0%

Personnel: ACAP AEXP PCAP PEXP LTEX PCON
base NOM NOM NOM NOM NOM NOM
Incr% 0% 0% 0% 0% 0% 0%

Project: TOOL SITE
base NOM NOM
Incr% 0% 0%

User: USER1 USER2
base NOM NOM
Incr% 0% 0%

EAF is also affected by Schedule
EAF: 1.00

OK Cancel Help

Parâmetros Usados

Early Design Parameters

	VLO	LO	NOM	HI	VHI	NHI
PCDC	0.75	0.81	0.88	1.00	1.50	1.74
SCSD	0.00	0.00	0.94	1.00	1.07	1.18
SDIP	0.00	0.00	0.97	1.00	1.29	1.81
PRDC	2.11	1.50	1.38	1.00	0.88	0.68
PRDC	1.24	1.38	1.12	1.00	0.97	0.91
PCIL	1.43	1.30	1.10	1.00	0.87	0.78
SCSD	0.00	0.00	1.14	1.00	1.00	1.00
PRDC	0.00	1.00	1.00	1.00	1.00	1.00
PCIL	0.00	0.00	1.00	1.00	1.00	1.00

Platform Parameters

Product Parameters

Personnel Parameters

Project Parameters

	VLO	LO	NOM	HI	VHI	NHI
TOOL	1.17	1.09	1.00	0.90	0.78	0.00
SCSD	1.43	1.14	1.00	1.00	1.00	0.00
SITE	1.22	1.09	1.00	0.93	0.86	0.80

Scalefactor Parameters

	VLO	LO	NOM	HI	VHI	NHI
PREC	6.20	4.96	3.72	2.48	1.24	0.00
PLEM	5.07	4.05	3.04	2.03	1.01	0.00
RESL	7.07	5.65	4.24	2.83	1.41	0.00
TEAM	5.48	4.38	3.29	2.19	1.10	0.00
PMAT	7.80	6.24	4.68	3.12	1.56	0.00

USER DEFINED PARAMETERS

	VLO	LO	NOM	HI	VHI	NHI
USR1	1.00	1.00	1.00	1.00	1.00	1.00
USR2	1.00	1.00	1.00	1.00	1.00	1.00

Equação

Equation Parameters

Exponent Equation

$$B = 0.91 + 0.01 (SF1 + \dots + SF5)$$

Effort Equation

$$PM = EMI + \dots + EMI7 + 2.94 * (Size)^B * (ASLOC * (AT/100) / ATPROD)$$

Schedule Equation

$$TDEV = [3.67 * PM^{(0.28 + 0.2 * (B - 0.91))}] * (SCED * /100)$$

OK Cancel Help

Resultados

Untitled - USC-COCOMO II.1999.0

File Edit View Parameters Calibrate Phase Maintenance Help

Project Name:

Scale Factor:

Schedule:

Development Model:

K	Module Name	Module Size	LRNED Rate (\$/month)	EXP	EST Effort DEV	EST Effort TEST	PEER	COST	INST COST	Staff	RISK
X	Fluxo de Caixa	8:10000	1000.00	2.08	34.5	71.2	140.3	71187.27	7.1	3.7	3.0

Total Lines of Code:

Estimated	Effort	Sched	PEER	COST	INST	Staff	RISK
Optimistic	56.9	17.0	175.6	56948.01	3.7	3.2	
Most likely	71.2	19.1	140.3	71187.27	7.1	3.7	3.0
Pessimistic	89.0	20.5	112.4	88984.06	8.9	4.3	

Ready

Resultados

Phase Distribution - Project Overall

Overall Phase Distribution

PROJECT	Teste				
SLOC	10000				
TOTAL EFFORT	71.187 Person Months				
	PCNT	EFFORT (PM)	PCNT	SCHEDULE	Staff
Plans And Requirements	7.000	4.983	10.167	3.467	1.437
Product Design	17.000	12.102	25.083	4.786	2.528
Programming	60.750	43.246	51.667	9.859	4.386
- Detailed Design	25.917	18.449	----	----	----
- Code and Unit Test	34.833	24.797	----	----	----
Integration and Test	22.250	15.839	23.250	4.437	3.570

Phase Distribution - Project Plans & Requirements

Life Cycle Phase Plans And Requirements
 Life Cycle Effort 4.983 Person Months
 Life Cycle Schedule 3.467 Months

	PCNT	EFFORT (PM)	SCHEDULE	Staff
Requirements Analysis	46.917	2.338	3.467	0.674
Product Design	16.542	0.624	3.467	0.238
Programming	3.583	0.179	3.467	0.052
Test Planning	3.042	0.152	3.467	0.044
Verification and Validation	6.542	0.326	3.467	0.094
Project Office	14.417	0.718	3.467	0.207
CM/QA	3.000	0.149	3.467	0.043
Manuals	5.958	0.297	3.467	0.086

OK

Help

Simplificando

➡ Bem, podemos usar fórmulas simplificadas para ter uma idéia dos valores

$$PM = 2.94 \times MLDC^{1.1}$$

$$TDEV = [3.67 \times (PM_{NS})^{0.32}]$$

Unindo COCOMO e APF

- ➡ Use uma tabela de conversão de PF para COCOMO
- ➡ Use apenas os PF básicos (não responda as perguntas)
- ➡ Ou, use direto o software

Tabela Conversão PF/SLOC

Linguagem	SLOC/FP			
	Média	Mediana	Mais Baixo	Mais Alto
Access	35	38	15	47
ASP	69	62	32	127
Assembler	172	157	86	320
C	148	104	9	704
C++	60	53	29	178
C#	59	59	51	66
Clipper	38	39	27	70
COBOL	73	77	8	400
Excel	47	46	31	63
J2EE	61	50	40	60
Java	60	59	14	97
Lotus Notes	21	22	15	25
Oracle	38	29	4	122
Oracle Dev 2K/EOBMS	41/42	30	21/23	100
Powerbuilder	30	24	7	105
SQL	39	35	15	143
Visual Basic	50	42	14	276

Ajustando o COCOMO

- ➡ O método é aberto
- ➡ Ele permite que você ajuste as equações a partir dos seus dados
- ➡ O Software também permite isso

Aula 33

Professor:

Geraldo Xexéo
UFRJ

Conteúdo:

FIM: COCOMO II