

# O Modelo COCOMO II

Eduardo Figueiredo

<http://www.dcc.ufmg.br/~figueiredo>

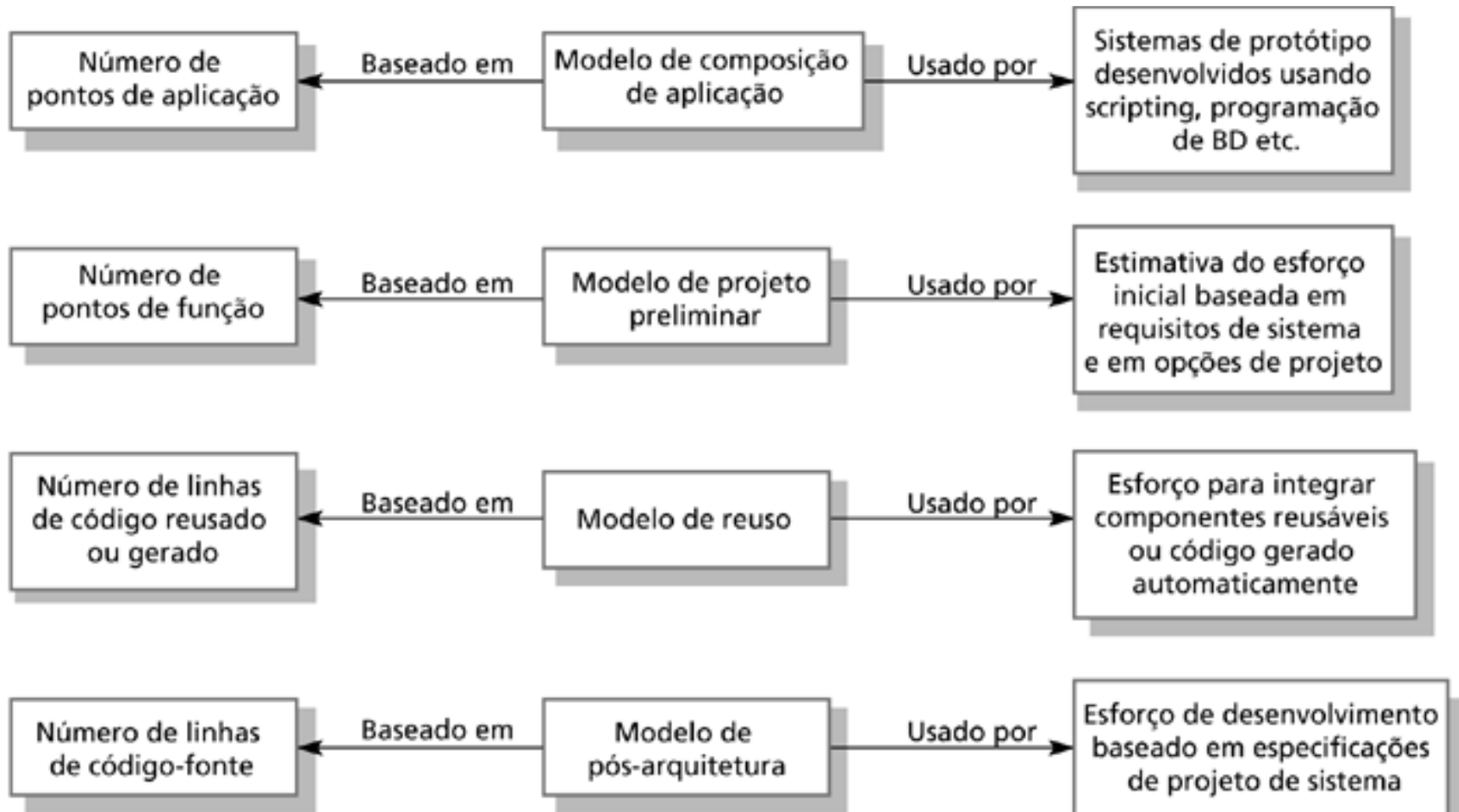
# [ COCOMO II ]

- Modelo de estimativa de custos, baseado no COCOMO - *Constructive Cost Modeling*
  - Considera abordagens mais modernas de desenvolvimento de software
- COCOMO II é um modelo empírico, obtido por meio de dados históricos
  - Dados foram coletados e analisados para descobrir fórmulas que os representem

# Sub-Modelos do COCOMO II

- COCOMO II incorpora quatro sub-modelos que produzem estimativas cada vez mais detalhadas
- Sub-modelos do COCOMO II
  - Modelo de composição de aplicação
  - Modelo de projeto preliminar
  - Modelo de reuso
  - Modelo de pós-arquitetura

# Modelo COCOMO II



# [ 1) Composição de Aplicação ]

- Usado quando o software é composto de partes existentes
  - Também usado na estimativa de protótipos
- A estimativa é feita em pontos de aplicação
  - Número de telas, relatórios e módulos
  - Código de *scripts* e consulta ao BD

# [ Fórmula ]

$$PM = ( NAP * (1 - \%reuso/100) ) / PROD$$

- Dado que
  - **PM** é o esforço em pessoa-mês
  - **NAP** é o número de pontos de aplicação
  - **%reuso** estima o código a ser reusado
  - **PROD** é a produtividade

# [ Produtividade ]

- Tabela de produtividade de ponto de aplicação por desenvolvedor (PROD)

Experiência e capacidade de desenvolvimento	Muito baixa	Baixa	Média	Alta	Muito Alta
Capacidade e maturidade em ferramentas (CASE)	Muito baixa	Baixa	Média	Alta	Muito Alta
PROD (NAP/mês)	4	7	13	25	50

## [ 2) Projeto Preliminar ]

- Usado quando os requisitos foram feitos, mas o projeto ainda não foi iniciado
  - O objetivo é uma estimativa aproximada de custo baixo (mesmo que imprecisa)
- A estimativa é feita em pontos de função e convertida para linhas de código
  - Número de entradas, saídas, interações com o usuário, interfaces externas e arquivos

# [ Fórmula Geral ]

$$\text{Esforço} = A * \text{Tamanho}^B * M$$

- O valor sugerido para A é 2,94
- O tamanho do sistema deve ser KLOC
  - A estimativa de KLOC pode ser feita a partir de um cálculo de pontos de função
- B pode variar de 1,1 a 1,24
- M é um multiplicador de sete atributos

# [ Multiplicadores (M) ]

$$M = \text{PERS} * \text{RCPX} * \text{RUSE} * \text{PDIF} * \text{PREX} * \text{FCIL} * \text{SCED}$$

## ■ Dado que

- PERS: capacidade pessoal
- RCPX: confiabilidade e complexidade
- RUSE: reuso necessário
- PDIF: dificuldade de plataforma
- PREX: experiência pessoal
- FCIL: recursos de apoio à equipe
- SCED: cronograma

## [ 3) Modelo de Reuso ]

- Usado para calcular o esforço de integração de componentes reusáveis
  - Geralmente usado em conjunto com o modelo pós arquitetura
- Impactado por três fatores principais
  - SU: custos de compreender o código
  - AA: custos de avaliação e tomada de decisão
  - AFF: adaptação do código para reuso

## [ 4) Modelo de Pós-Arquitetura ]

- Usado quando já foi projetada a arquitetura de sistema
  - Informações adicionais sobre o sistema estão disponíveis
- A estimativa é feita usando a fórmula geral com 17 multiplicadores
  - Os multiplicadores refletem capacidade de pessoal e características do produto e projeto

# [ Formula Geral ]

$$PM = A * Tamanho^B * M$$

- O expoente B depende de 5 fatores (*próximo slide*)
  - Para cada fator, é usada uma escala de 0 (muito alto) a 5 (muito baixo)
  - A soma dos fatores é dividida por 100 e acrescida de 1,01

# [ Fatores do Expoente B ]



## Fator de escala

## Explicação

### Precedência

Reflete a experiência prévia da organização com esse tipo de projeto. Muito baixo significa nenhuma experiência prévia; extra alto significa que a organização está completamente familiarizada com esse domínio de aplicação.

### Flexibilidade do desenvolvimento

Reflete o grau de flexibilidade no processo de desenvolvimento. Muito baixo significa que um processo prescrito é usado; extra alto significa que o cliente estabelece somente metas gerais.

### Arquitetura/resolução de riscos

Reflete a extensão da análise de riscos realizada. Muito baixo significa pouca análise; extra alto significa uma análise completa e abrangente.

### Coesão da equipe

Reflete o quanto os membros de equipe de desenvolvimento conhecem uns aos outros e trabalham juntos. Muito baixo significa interações muito difíceis; extra alto significa uma equipe integrada e eficaz, sem problemas de comunicação.

### Maturidade do processo

Reflete a maturidade do processo da organização. O cálculo desse valor depende do Questionário de Maturidade de CMM, mas uma estimativa pode ser obtida subtraindo-se de 5 o nível de maturidade de processo CMM.

# [ Bibliografia ]

- Ian Sommerville. **Engenharia de Software**, 9ª Edição. Pearson Education, 2011.
  - Cap. 23 Planejamento de Projeto (Seção 23.5)