



PLANEJAMENTO DO PROJETO

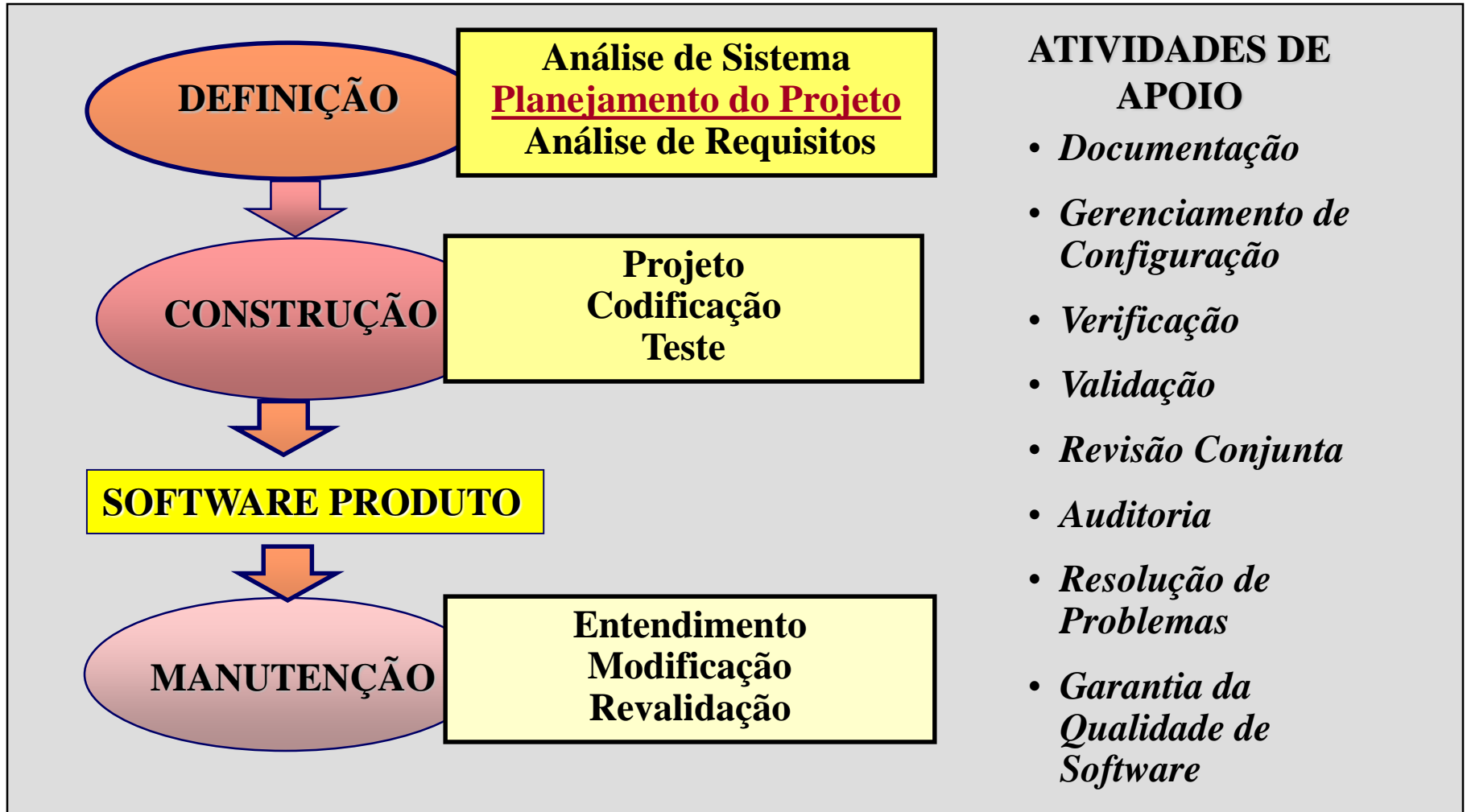
Ricardo Argenton Ramos  CECOMP

ricargentonramos@gmail.com



Engenharia de Software I – 2017.2

Atividades da Engenharia de Software



Atividades da Engenharia de Software

DEFINIÇÃO

Análise de Sistema
Planejamento do Projeto
Análise de Requisitos

No **Planejamento do Projeto de Software** devem ser derivados:
estimativa do esforço humano
exigido, duração cronológica
e custo

Revalidação

ATIVIDADES DE APOIO

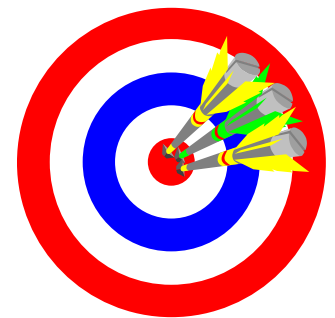
- *Documentação*
- *Gerenciamento de Configuração*
- *Verificação*
- *Validação*
- *Revisão Conjunta*
- *Auditoria*
- *Resolução de Problemas*
- *Garantia da Qualidade de Software*

Por que planejar?

- O desenvolvimento de software possui vários ciclos, que podem ser repetidos diversas vezes, até que se obtenha um produto que satisfaça aos requisitos do cliente
- O cliente precisa saber quanto custará e quando ficará pronto!!
- Há riscos envolvidos
- O planejamento é essencial para:
 - decidir se o projeto continuará ou não
 - servir de base para o gerenciamento de projeto

Objetivos do Planejamento

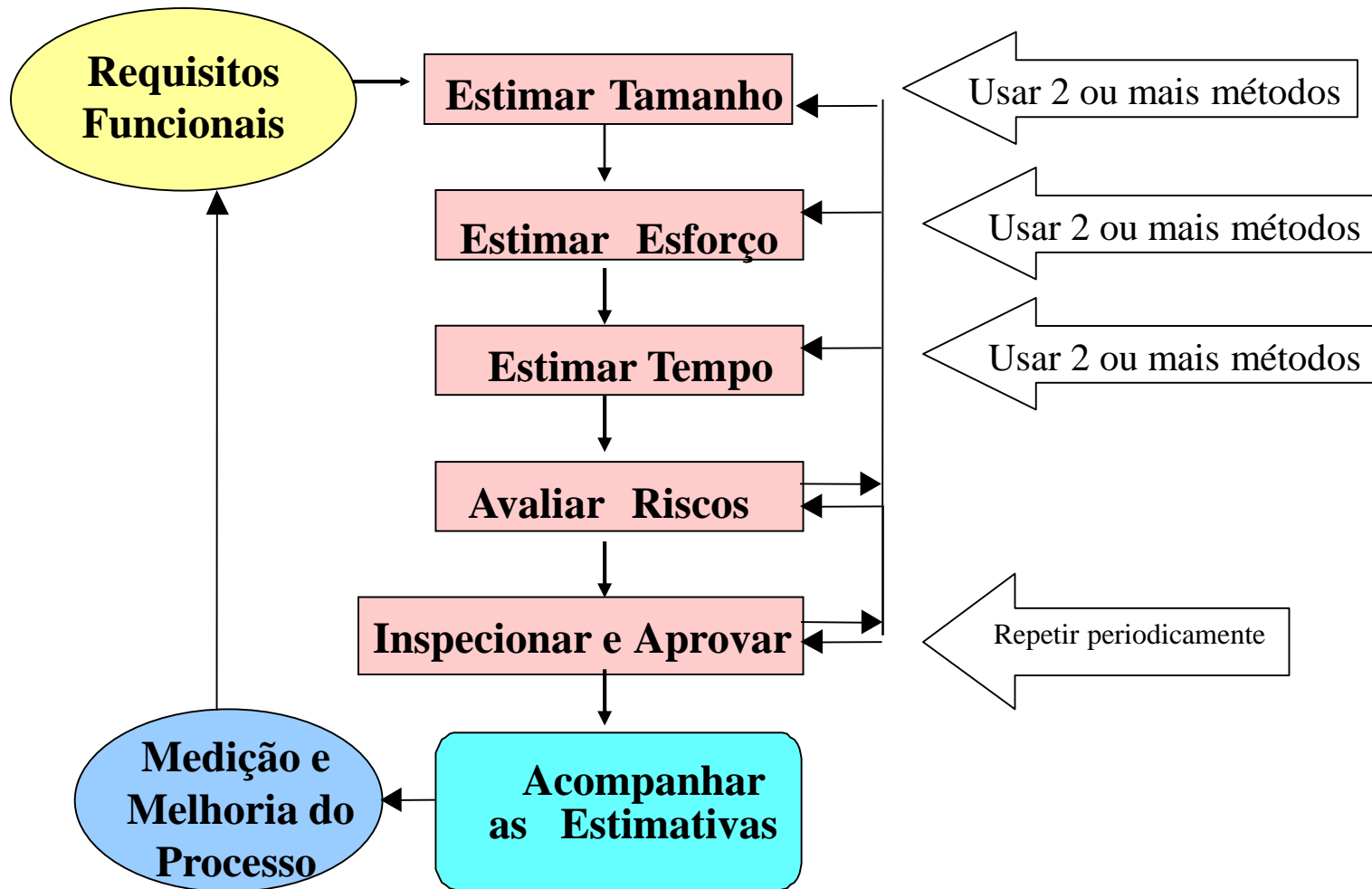
- Determinar o alcance do trabalho a ser realizado: função, desempenho, interface e segurança
- Estimar recursos necessários ao desenvolvimento do software: recursos humanos, de hardware e de software
- Identificar tarefas a serem efetuadas
- Elaborar cronogramas
- Estimar esforço (custo) despendido



Atividades Fundamentais de Planejamento de Projeto

- Elaboração de Estimativas
- Análise de Riscos
- Elaboração de Cronograma
- Elaboração do Plano e Aprovação

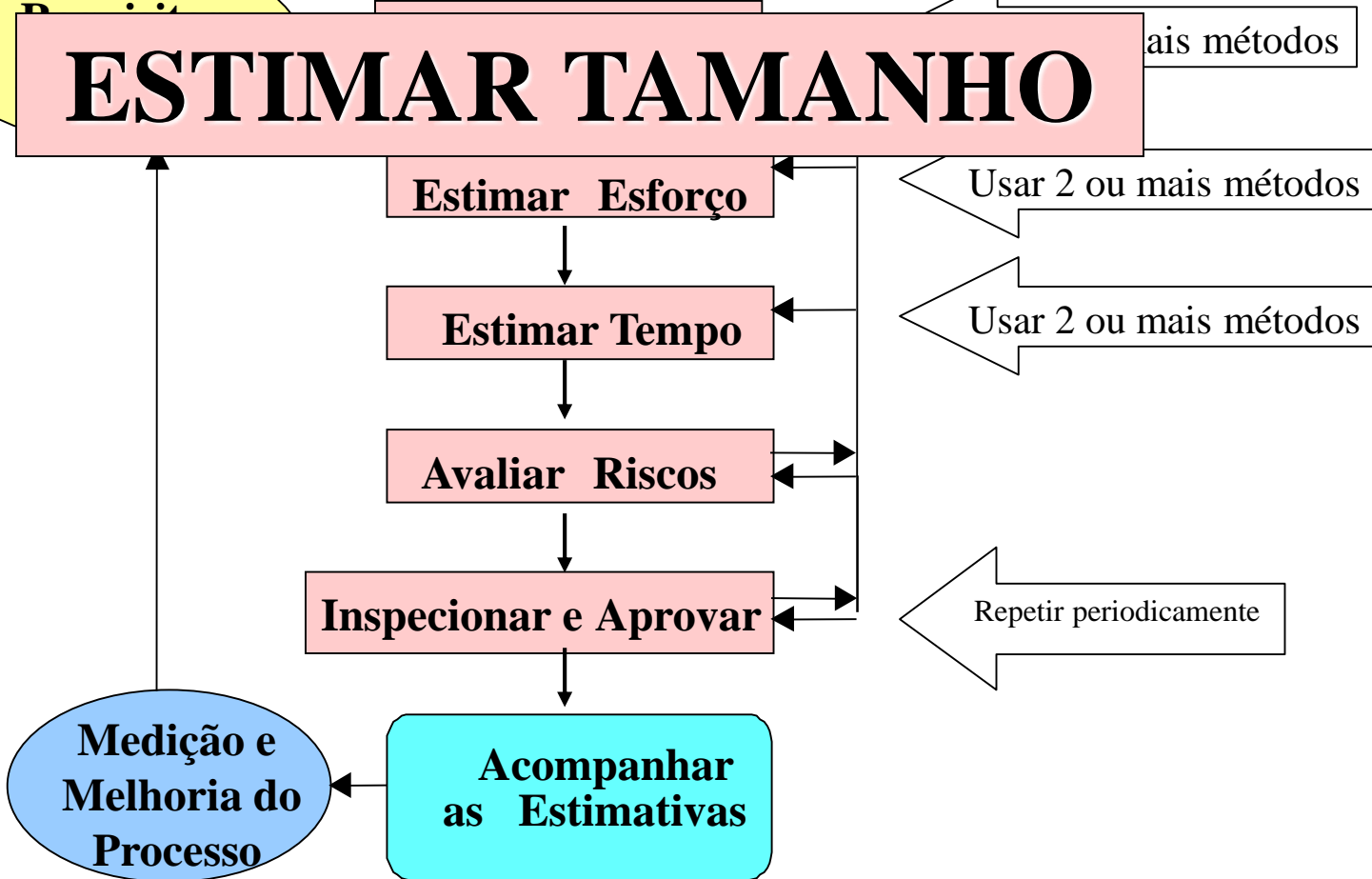
Estimativas de Projeto de Software



**LINHAS DE CÓDIGO
PONTOS POR
FUNÇÃO**

Projeto de Software

ESTIMAR TAMANHO



Como Medir o Tamanho do Software?

- O primeiro **problema** que se depara para elaborar **estimativas** é o dilema da **escolha** da **métrica** mais adequada para medir o **tamanho** de aplicações.

- Contagem de Linhas de Código (LOC)

- Contagem de Pontos por Função (PF)

Contagem de Linhas de Código

- A forma **familiar** de se medir tamanho de software é por meio da contagem de linhas de código.

- Contagem de Linhas de Código (LOC)

Contagem de Linhas de Código

VANTAGENS:

- Fáceis de serem obtidas
- Vários modelos de estimativa baseados em LOC ou KLOC

DESVANTAGENS:

- LOC depende da linguagem de programação
- Penalizam programas bem projetados, mas pequenos
- Não se adaptam às linguagens não procedimentais
- Difícil de obter em fase de planejamento

Contagem de Pontos por Função

- A contagem de Pontos por Função é uma técnica utilizada para medir o tamanho do software pela quantificação da funcionalidade do processamento da aplicação.

- Contagem de Pontos por Função (PF)

Contagem de Pontos por Função

- Uma das principais vantagens da contagem de pontos por função é a possibilidade de estimar a dimensão de projetos desde as primeiras fases de análise e projeto de sistemas, quando se dispõe de poucas informações sobre o sistema.

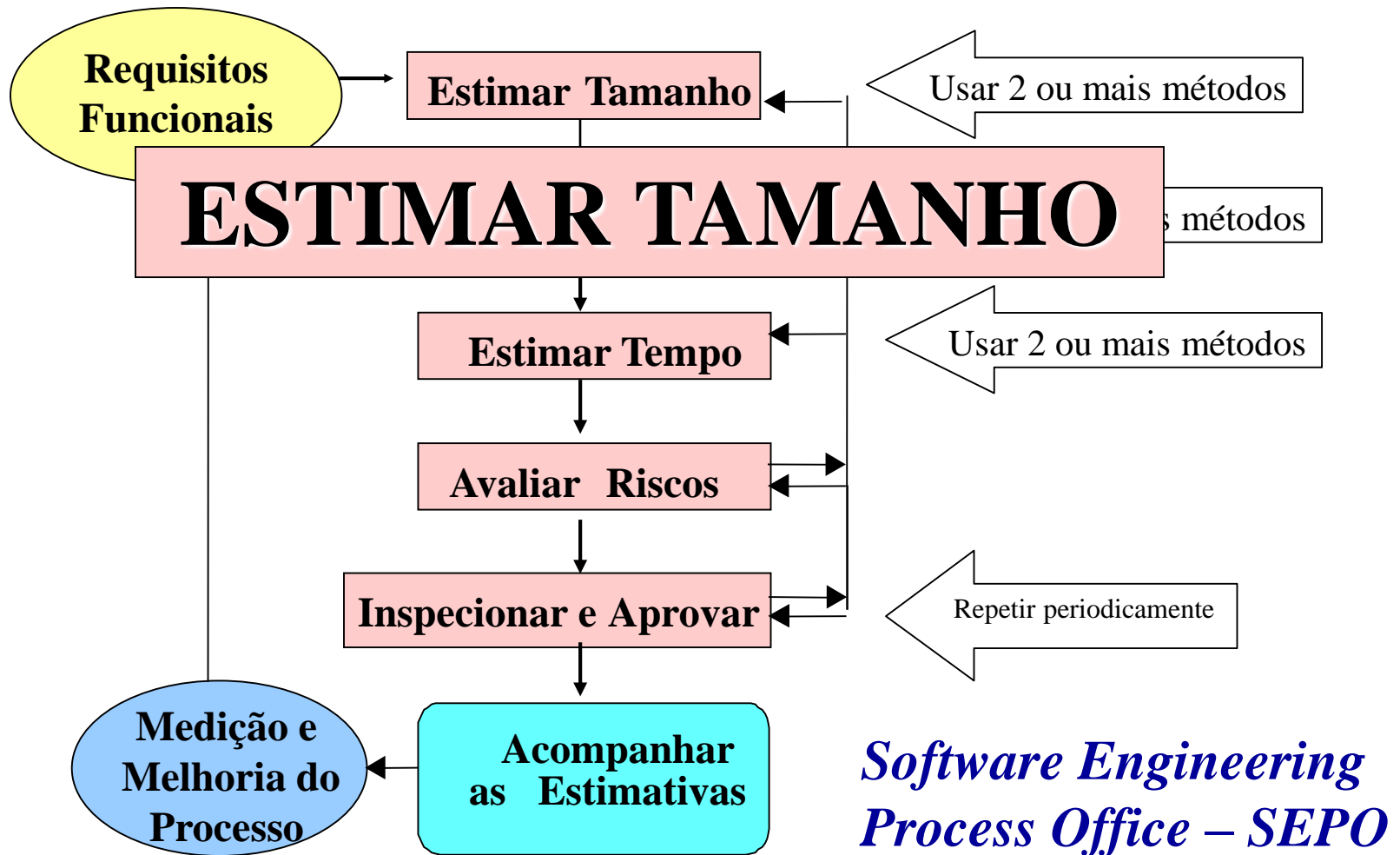
Como Medir o Tamanho do Software?

*Análise de Pontos por
Função
IFPUG (International
Function Points Users
Group)*

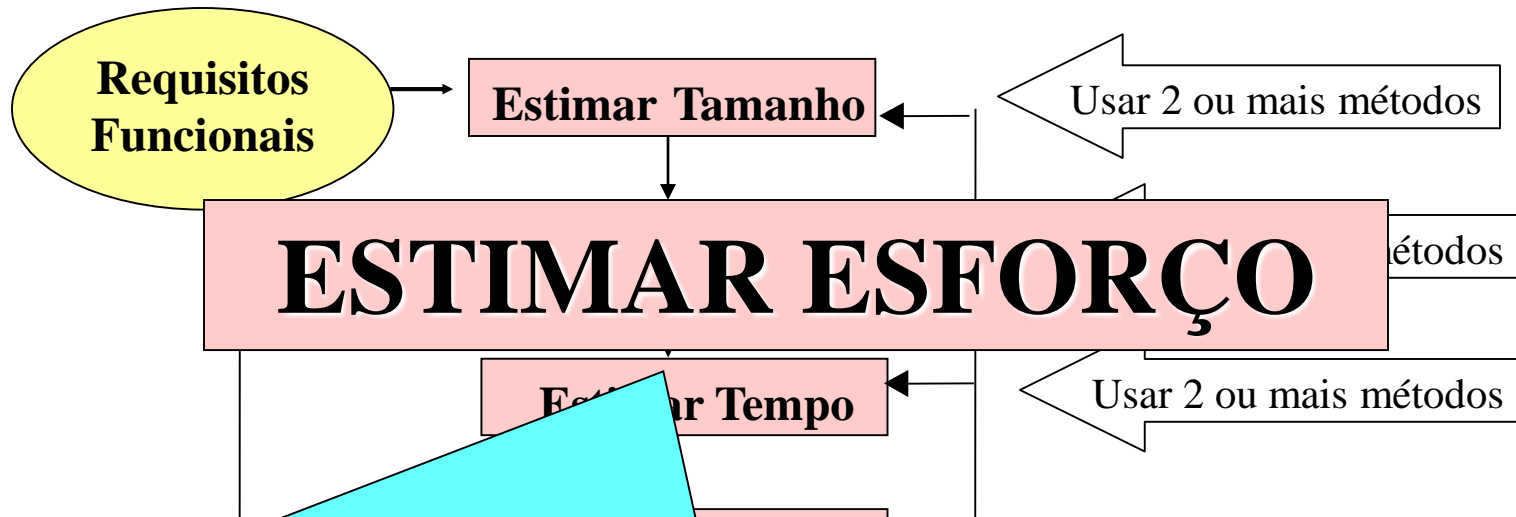
*Pontos por Função
NESMA (Netherlands
Function Points Users
Group)*

- **Contagem de Pontos por Função (PF)**

Estimativas de Projeto de Software



Estimativas de Projeto de Software



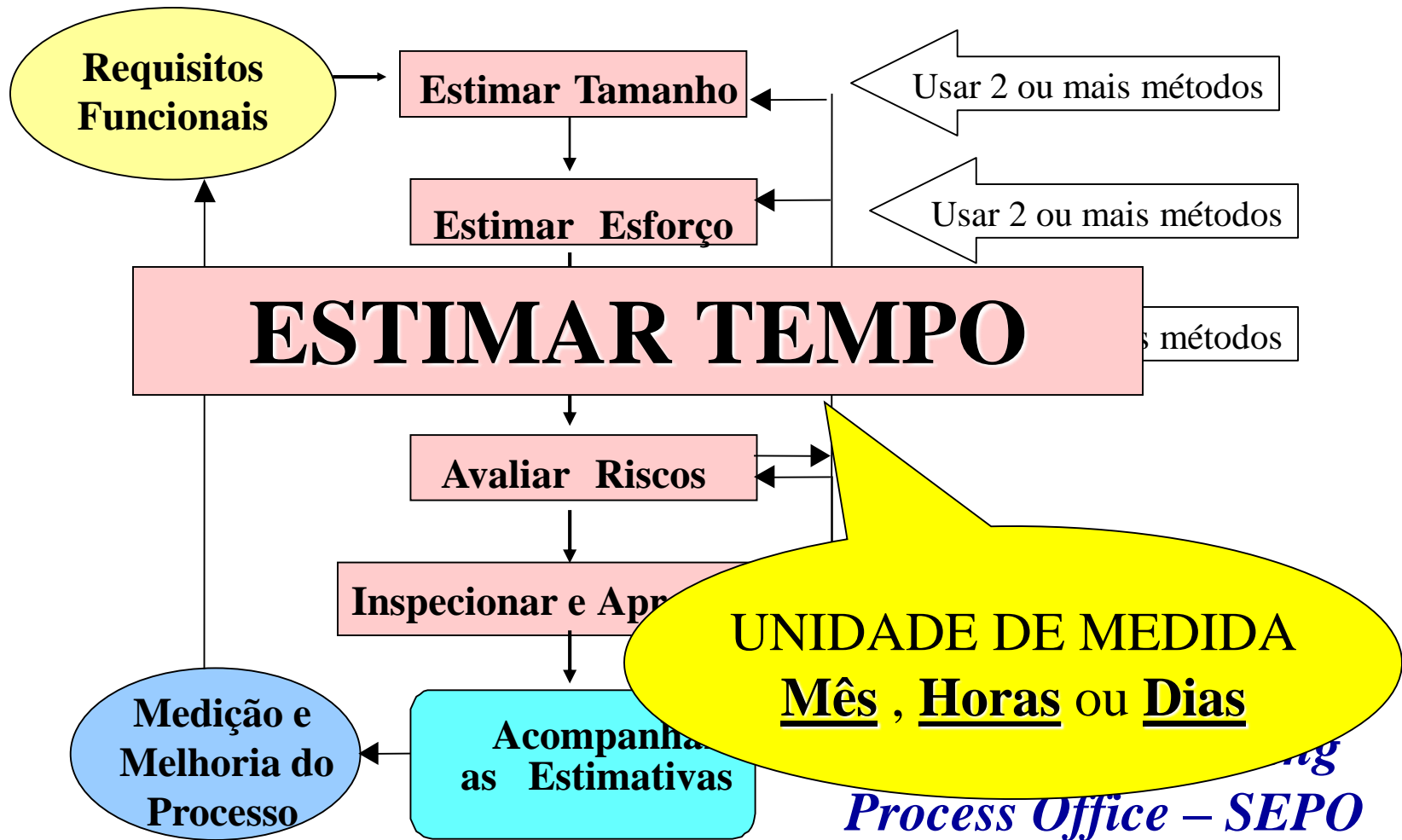
UNIDADE DE MEDIDA

Pessoas.Mês ou Pessoas.Hora

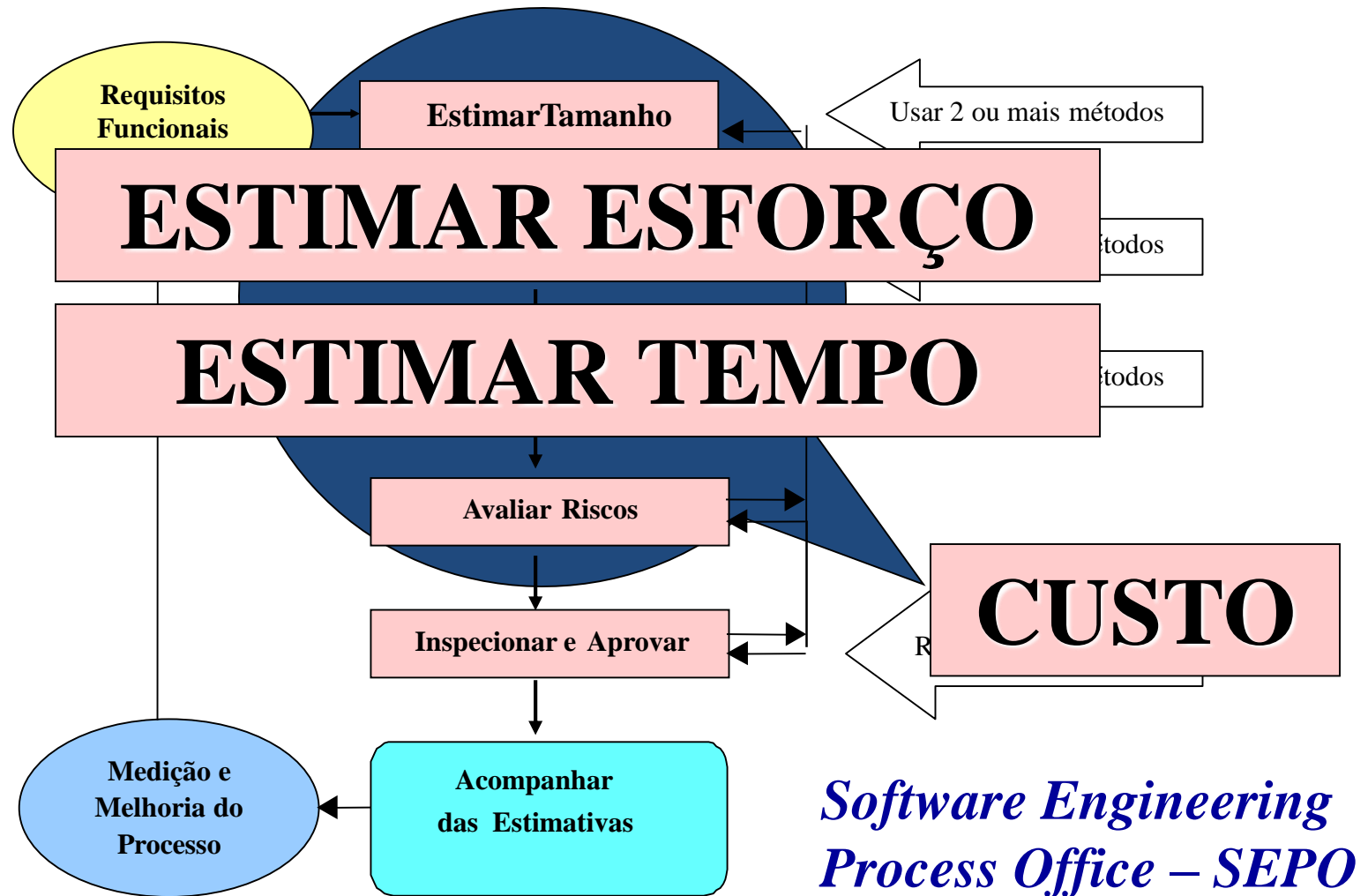
Exemplo:

esforço necessário para desenvolver o projeto = 12 Pessoas.Mês
(1 pessoas durante 12 meses)

Estimativas de Projeto de Software



Estimativas de Projeto de Software



Estimativas de Recursos e Software

Conhecendo o tamanho

Requisitos Funcionais

Estimar Tamanho

Usar 2 ou mais métodos

ESTIMAR ESFORÇO

todos

ESTIMAR TEMPO

todos

Avaliar Riscos

CUSTO

modelos empíricos
Modelo COCOMO 81

*Software Engineering
Process Office – SEPO*

MODELO *COCOMO* 81

COⁿstructive CO^st Model (Modelo de Custo Construtivo)

- Apresentado em 1981 por Boehm
- O **COCOMO** é um modelo desenvolvido para estimar esforço, prazo, custo e tamanho da equipe para um projeto de software
- Todas as referências ao **COCOMO** encontradas na literatura publicada até 1995 são citações desse modelo

MODELO *COCOMO* 81

- O **COCOMO** apresenta uma série de equações derivadas a partir do estudo de uma base de dados de 63 projetos, em sua maior parte na empresa *TRW Systems, Inc*

Aplicações de diferentes domínios

- **negócios**
- **aplicações científicas**
- **sistemas de controle**
- **sistemas operacionais**

MODELO *COCOMO* 81

- O **COCOMO** apresenta uma série de equações derivadas a partir do estudo de uma base de dados de 63 projetos, em sua maior parte na empresa *TRW Systems, Inc*

Aplicações implementadas em várias linguagens diferentes, cujas dimensões variavam de 2.000 até 1.000.000 de linhas de código (comentários excluídos)

MODELO *COCOMO* 81

- Para obter as equações do *COCOMO* foram combinados:
 - a experiência
 - resultados de outros modelos de estimativa de custo e
 - a opinião subjetiva de gerentes de software experientes

MODELO *COCOMO* 81

- O **COCOMO** é apresentado na forma de um conjunto de modelos divididos hierarquicamente em três níveis:

- **Modelo COCOMO Básico**
- **Modelo COCOMO Intermediário**
- **Modelo COCOMO Avançado**

MODELO *COCOMO* 81

MODELO 1

Modelo **COCOMO Básico**

- calcula o esforço do desenvolvimento de software em função do tamanho estimado do programa, expresso em linhas de código

MODELO *COCOMO* 81

MODELO 1

Modelo *COCOMO* Básico

- Esta versão é aplicável à grande maioria dos projetos de software, de pequeno ou médio porte.
- É limitada por não considerar fatores que interferem no desenvolvimento do projeto, do tipo:
 - restrições de hardware
 - qualificação e experiência do pessoal de desenvolvimento e
 - uso de ferramentas técnicas modernas, entre outros.

MODELO *COCOMO* 81

MODELO 2

Modelo **COCOMO Intermediário**

- calcula o esforço de desenvolvimento de software em função do tamanho do programa e de um conjunto de direcionadores de custo, alternativamente chamados atributos ou fatores de software, que incluem avaliações subjetivas do produto, do hardware, do peçoal e dos atributos do projeto

MODELO COCOMO 81

MODELO

Modelo COCOMO

- calcula o esforço de desenvolvimento de software em função do tempo e de um conjunto de diretrizes e alternativamente chamadas fatores de software, que são subjetivas do produto, do pessoal e dos atributos do

Característica de desenvolvimento de software que tem efeito aumentativo ou diminutivo na quantidade de esforço de desenvolvimento final do projeto

Exemplos:

- ↳ a experiência da equipe de projeto
- ↳ a confiabilidade requerida do software

MODELO *COCOMO* 81

MODELO 3

Modelo **COCOMO** **Avançado**

- incorpora todas as características da versão intermediária, porém em cada passo do processo de engenharia de software.

MODELO *COCOMO* 81

- Depois da análise dos **requisitos funcionais do software**, o **tamanho** da aplicação deve ser estimado em milhares de linhas de código (**KLOC**)
- Determinar o tamanho no início do projeto é uma das limitações do método
- Uma alternativa viável é a utilização da técnica de contagem de **Pontos de Função**, por ser facilmente efetuada logo no início do projeto

MODELO *COCOMO* 81

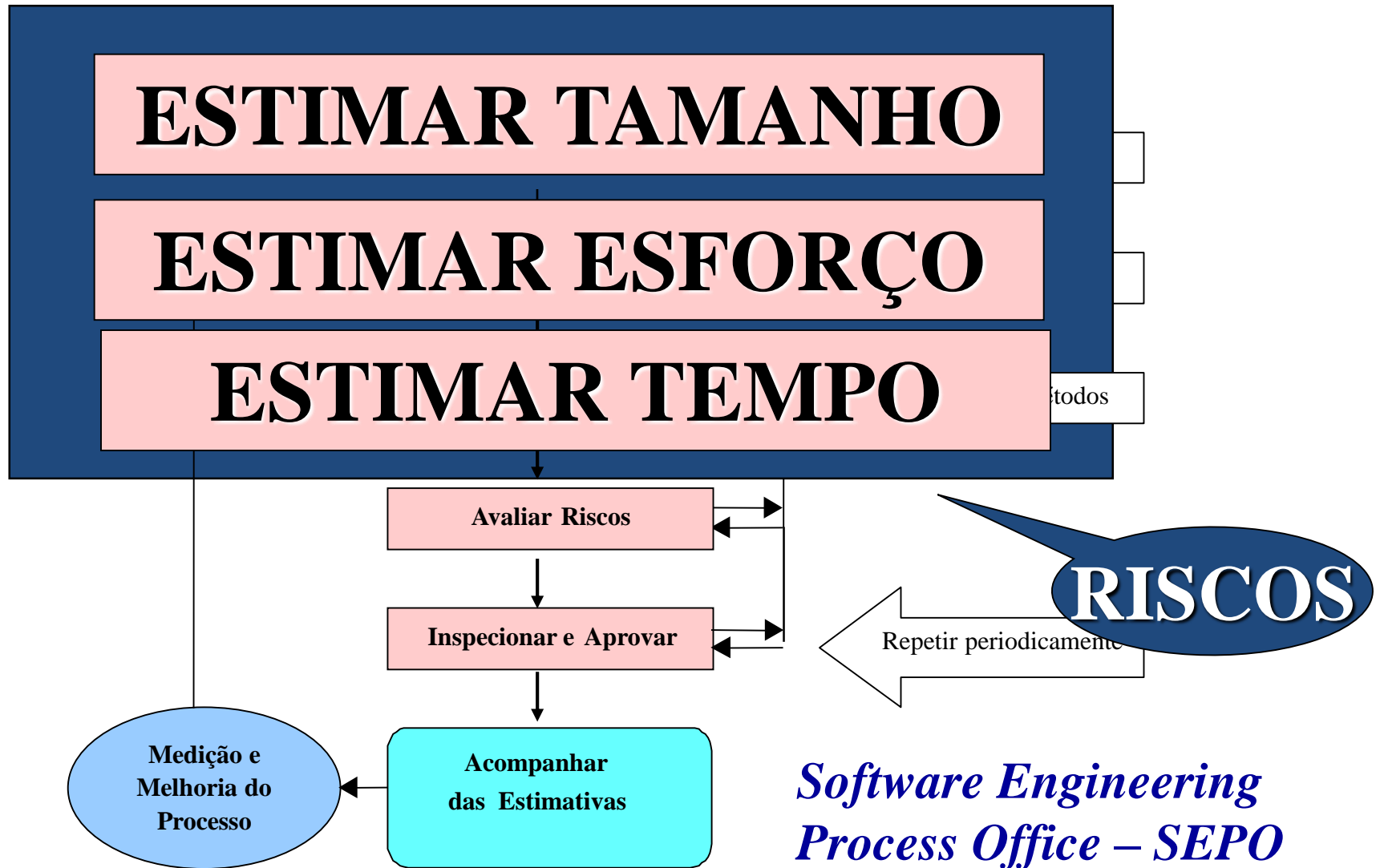
- Pontos de função podem ser convertidos em linhas de código

<i>Linguagem</i>	<i>LOC/PF</i>	<i>Linguagem</i>	<i>LOC/PF</i>
ACCESS	38	FoxPro 2.5	34
Ansi SQL	13	HTML 3.0	15
Ansi COBOL 85	91	JAVA	53
C	128	LISP	64
C++	53	Smalltalk	22
Clipper	19	Object Pascal	29
COBOL II	107	Oracle	40
dBase IV	36	Turbo C	128
Delphi	29	Turbo Pascal V.5	49
Fortran 95	71	Visual Basic 5	29

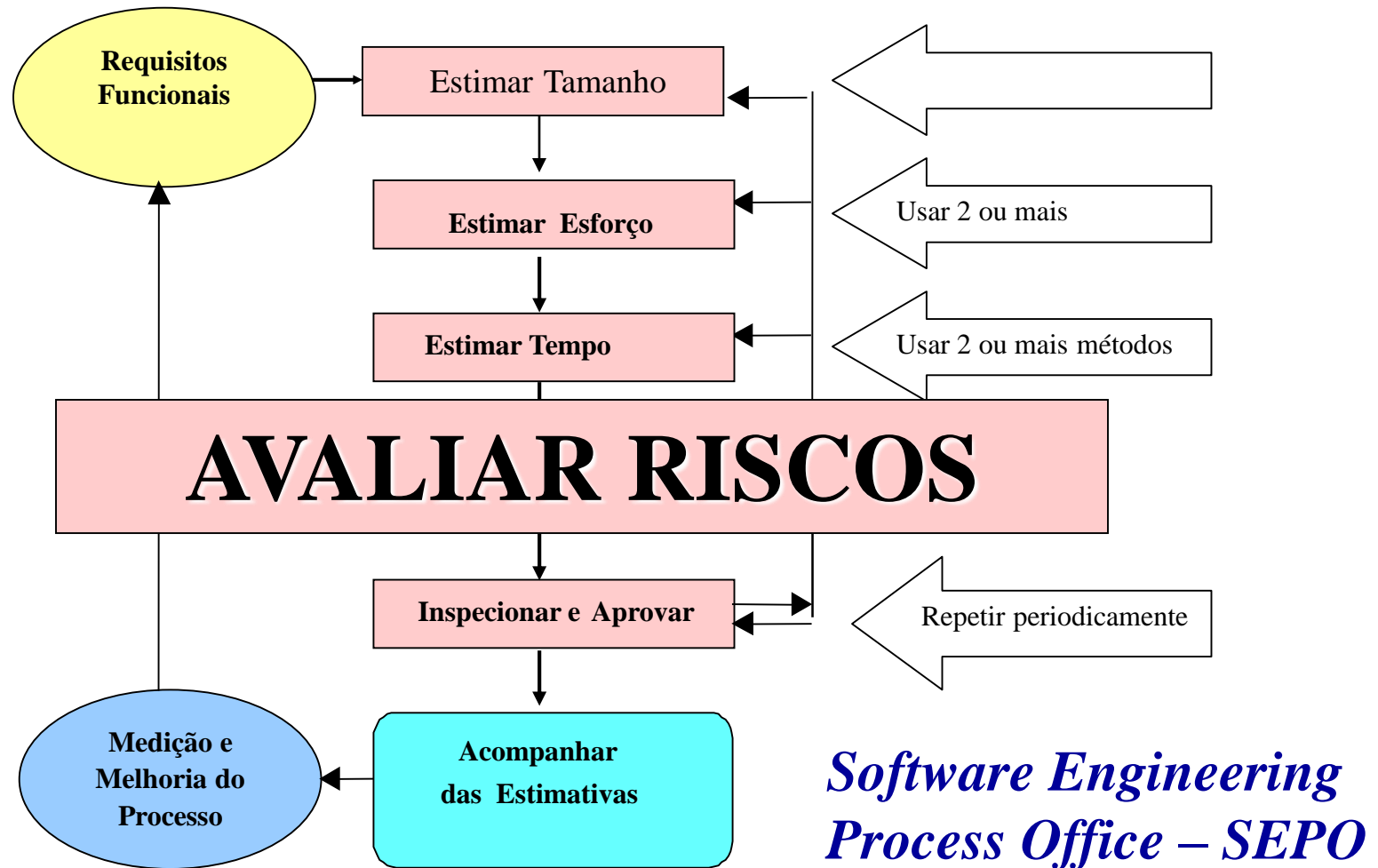
MODELO *COCOMO* 81

- A aplicação do método começa pela **classificação** do produto a ser mensurado, categorizando o software em um de três tipos fundamentais de desenvolvimento identificados por Boehm:
 - *Orgânico*
 - *Embutido*
 - *Semi-destacado*

Estimativas de Projeto de Software



Estimativas de Projeto de Software



Atividades Fundamentais de Planejamento de Projeto

- Elaboração de Estimativas
- Análise de Riscos
- Elaboração de Cronograma
- Elaboração do Plano e Aprovação

Gerenciamento de Riscos

- **Risco** é um problema em potencial – pode ou não acontecer
- É importante:
 - Identificá-lo
 - Avaliar sua probabilidade de ocorrência
 - Estimar seu impacto
 - Estabelecer um plano de contingência para o caso dele efetivamente ocorrer
- Não são tarefas fáceis!!!

Plano de Projeto-Riscos

III. RISCOS DO PROJETO

1. Análise dos riscos



Passos para analisar os riscos:

- identificação
- avaliação
- disposição por ordem de prioridade

2. Administração dos riscos



Passos para atacar os riscos:

- estratégias de administração
- resolução
- monitoração

“O fundamental é que os Riscos assumidos sejam os Riscos certos”

Plano de Projeto-Riscos

Identificação dos Riscos

de Projeto

identificam problemas orçamentários, de cronograma, de pessoal, de recursos, de clientes, de requisitos e o impacto no projeto do software

Técnicos

identificam potenciais problemas de projeto, implementação, interface, verificação e manutenção

do Negócio

podem destruir até os melhores projetos: construir um produto que ninguém quer; ou que não se encaixe mais na estratégia da empresa; perder o apoio da administração, ou o compromisso orçamentário

“Se você não atacar ativamente os riscos técnicos e de projeto, eles lhe atacarão ativamente.” Gilb

Atenuação, Monitoração e Administração do Risco

- **Atenuação** — como podemos evitar o risco?
- **Monitoração** — que fatores podem ser rastreados para ajudar-nos a prevenir a ocorrência do risco?
- **Administração** — que planos de contingência temos para o caso do risco se tornar efetivo?

Exemplos para pensar: Cliente não sabe o que quer, Alta Rotatividade de Pessoal

Cuidado com a administração de riscos!

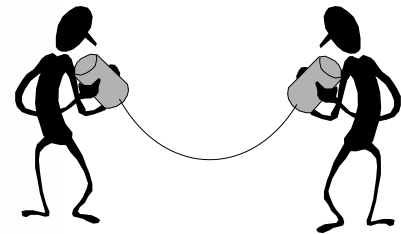
Mito: “Se sairmos fora do cronograma, adicionamos mais programadores e recuperamos o atraso”.



Isso faz o cronograma atrasar ainda mais!

Motivo: a comunicação é absolutamente essencial para o desenvolvimento do software.

Todo novo caminho de comunicação exige esforço adicional e portanto, tempo adicional.



Exemplo 1: riscos relacionados ao cliente

Questões a serem respondidas

- **Você já trabalhou com esse cliente no passado?**
- **O cliente tem uma idéia sólida dos requisitos?**
- **O cliente deseja participar das revisões?**
- **O cliente é tecnicamente sofisticado?**
- **O cliente entende o processo de engenharia de software?**

Exemplo 2: Riscos Tecnológicos

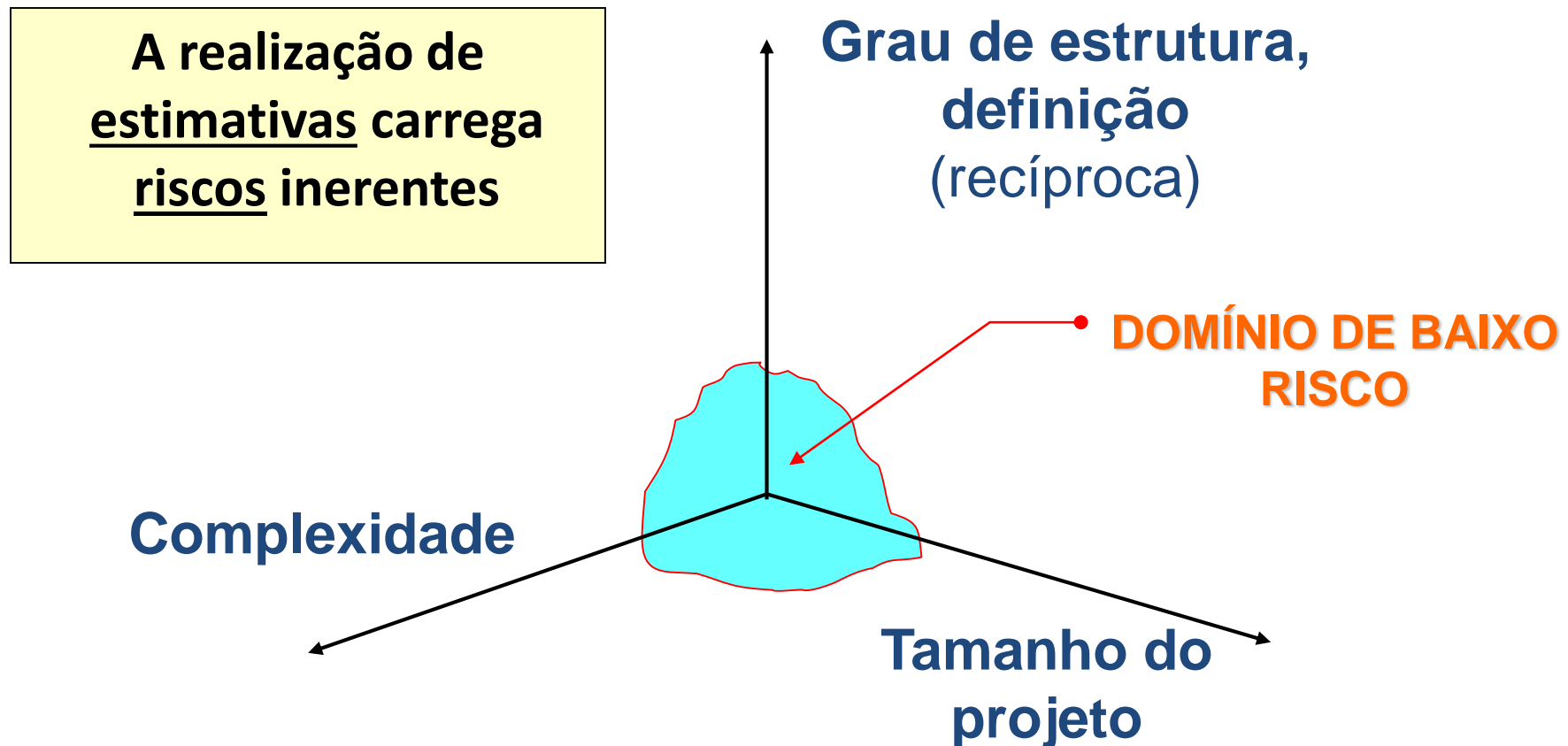
Questões a serem respondidas

- A tecnologia é nova para sua empresa?
- **Algum hardware novo ou não testado está envolvido?**
- **Será necessária uma interface com o usuário especializada?**
- **Você está usando novos métodos de engenharia de software?**
- **Você está usando métodos de desenvolvimento de software não convencionais, como métodos formais, abordagens de IA, redes neurais?**
- **Existem restrições significativas de desempenho?**

Riscos: os 10 mais (Bohem)

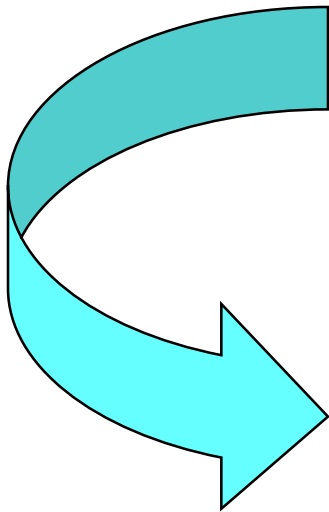
- Imprevistos de pessoal
- Cronogramas e orçamentos não realísticos
- Desenvolvimento das funções erradas
- Desenvolvimento da interface com o usuário errada
- Requisitos sofisticados, sem necessidade
- Fluxo contínuo de mudanças nos requisitos
- Imprevistos em serviços terceirizados
- Imprevistos em componentes terceirizados
- Imprevistos de desempenho em tempo real
- Capacidade de ciência de computação excedida

Fatores que aumentam o Risco das Estimativas



Fator que Reduz o Risco das Estimativas

DADOS HISTÓRICOS



- Estimativas podem ser feitas com maior segurança
- Prazos podem ser estabelecidos para se evitar dificuldades passadas
- Riscos globais podem ser reduzidos

Atividades Fundamentais de Planejamento de Projeto

- Elaboração de Estimativas
- Análise de Riscos
- Elaboração de Cronograma
- Elaboração do Plano e Aprovação

Elaboração do Cronograma

TAREFAS:

1. Identificar e selecionar os recursos para o projeto
2. Inter-relacionar as atividades e definir precedências
3. Calcular o caminho crítico
4. Alocar recursos nas atividades
5. Preparar cronograma do projeto

Elaboração do Cronograma

TAREFAS:

1. Identificar e selecionar os recursos para o projeto
2. Inter-relacionar as atividades e definir precedências
3. Calcular o caminho crítico
4. Alocar recursos nas atividades
5. Preparar cronograma do projeto

Identificar e Selecionar os Recursos para o Projeto

- A identificação e seleção de recursos para o projeto é usualmente conduzida em paralelo com a elaboração de estimativas de tempo, devido à dependência intrínseca entre duração e quantidade de recursos.
- Para se calcular a duração mais precisa do projeto, é necessário que se conheçam todos os recursos alocados nas atividades e a produtividade de cada um deles.

Identificar e Selecionar os Recursos para o Projeto

- Devem ser identificados e selecionados:
 - todos os recursos humanos (quantos e quais profissionais),
 - todos os materiais de consumo e equipamentos (quantos, quando e quais os tipos de equipamentos) e
 - todos os recursos financeiros (quanto e quando) necessários à execução do projeto.

Elaboração do Cronograma

TAREFAS:

1. Identificar e selecionar os recursos para o projeto
2. Inter-relacionar as atividades e definir precedências
3. Calcular o caminho crítico
4. Alocar recursos nas atividades
5. Preparar cronograma do projeto

Inter-relacionar as Atividades e Definir Precedências

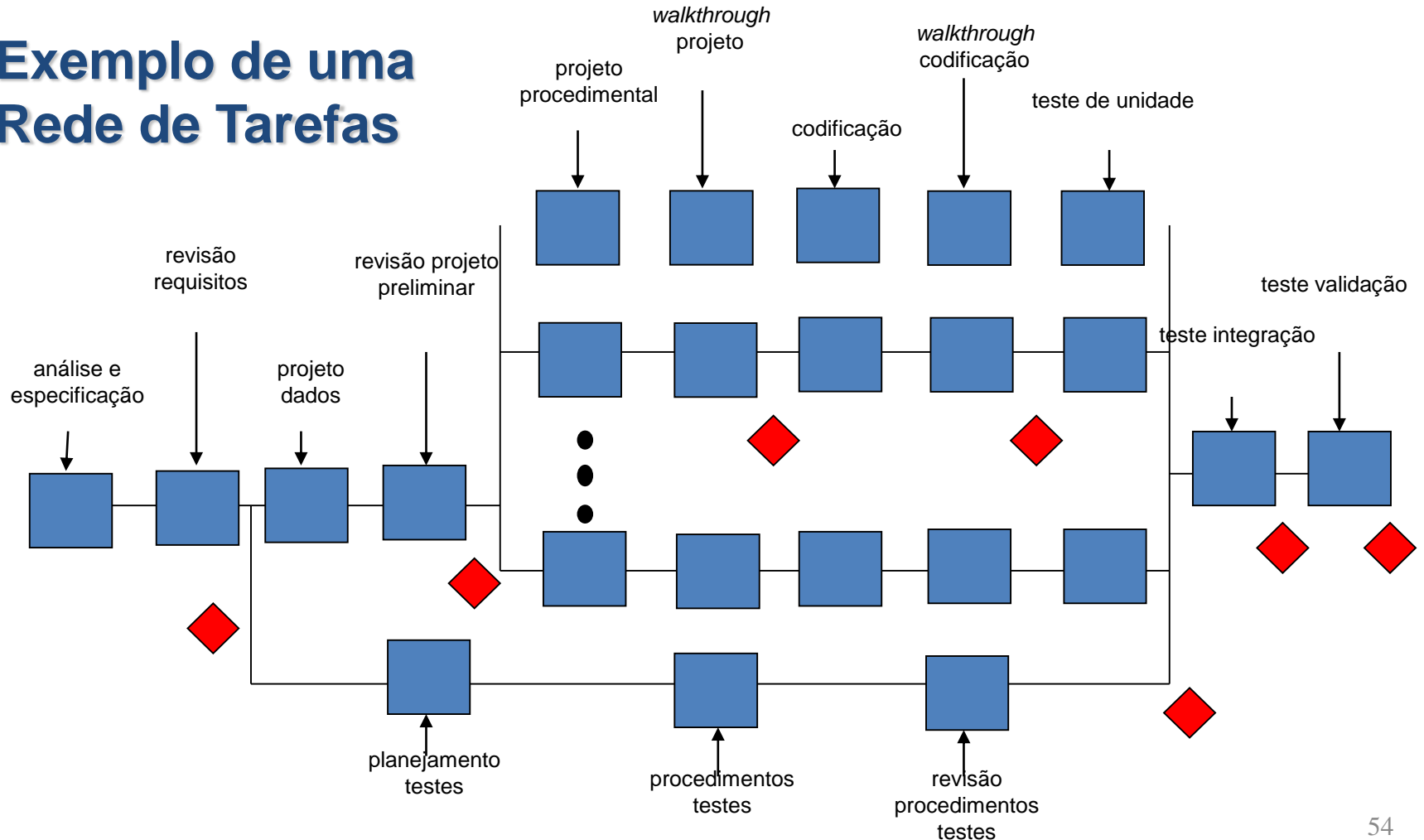
- O objetivo dessa tarefa é identificar atividades interdependentes para que o cronograma do projeto seja elaborado.

Inter-relacionar as Atividades e Definir Precedências

- Existem várias técnicas gráficas para representar os interrelacionamentos entre as atividades e definir as precedências
- A mais consagrada:
 - **a rede de PERT**

Rede PERT (*P*rogram *E*valuation and *R*evision *T*echinique)

Exemplo de uma Rede de Tarefas



Elaboração do Cronograma

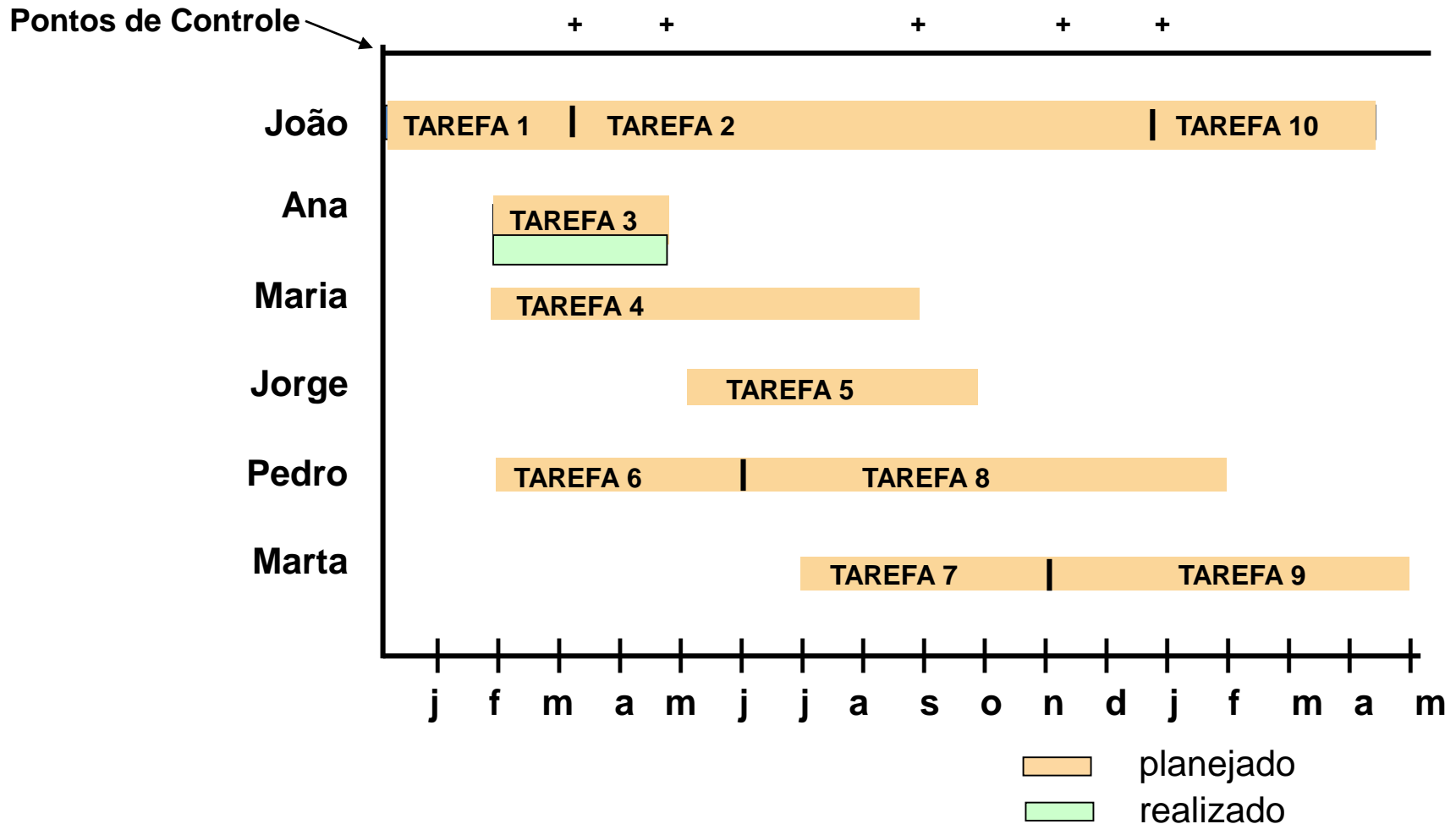
TAREFAS:

1. Identificar e selecionar os recursos para o projeto
2. Inter-relacionar as atividades e definir precedências
3. Calcular o caminho crítico
4. Alocar recursos nas atividades
5. Preparar cronograma do projeto

Preparar Cronograma do Projeto

- Essa tarefa tem como objetivo apresentar graficamente as datas de início e término de cada atividade, uma vez que os recursos, durações e as interdependências já estão estabelecidas.
- O cronograma do projeto pode ser apresentado de diferentes formas:
 - Tabelas com listas de atividades
 - Gráficos de *Gantt*,
 - Gráficos de marcas ou etapas, etc

Exemplo de Gráfico de Gantt



Atividades Fundamentais de Planejamento de Projeto

- Elaboração de Estimativas
- Análise de Riscos
- Elaboração de Cronograma
- Elaboração do Plano e Aprovação

Elaboração do Plano do Projeto

- Essa tarefa consiste no preenchimento de todas as seções do plano de projeto.

Esboço do Plano de Projeto de Software

I. Introdução.

1. Escopo e propósito do documento.
2. Objetivos do projeto.
 - a. Objetivos.
 - b. Funções principais.
 - c. Questões de desempenho.
 - d. Restrições técnicas e administrativas.

II. Estimativas de projeto.

1. Dados históricos usados nas estimativas.
2. Técnicas de estimativa.
3. Estimativas.

III. Riscos do projeto.

1. Análise dos riscos.
 - a. Identificação.
 - b. Estimativa dos riscos.
 - c. Avaliação.
2. Administração dos riscos.
 - a. Opções para evitar os riscos.
 - b. Procedimentos de monitoração dos riscos.

IV. Cronograma.

1. *Work breakdown* - divisão de trabalho no projeto.
2. Rede de tarefas.
3. Gráfico de timeline (gráfico de Gantt).
4. Tabela de recursos.

V. Recursos do projeto.

1. Pessoal.
2. Hardware e software.
3. Recursos especiais.

VI. Organização do pessoal.

1. Estrutura de equipe (se for o caso).
2. Relatórios administrativos.

VII. Mecanismos de tracking (rastreamento) e controle.

VIII. Apêndices.