

MANUTENÇÃO DE SOFTWARE

Prof. Andréa Sabedra Bordin



SWEBOK[®] V3.0

*Guide to the Software
Engineering Body of Knowledge*

INTERNATIONAL
STANDARD

ISO/IEC
14764

IEEE
Std 14764-2006

Second edition
2006-09-01

**Software Engineering — Software Life
Cycle Processes — Maintenance**

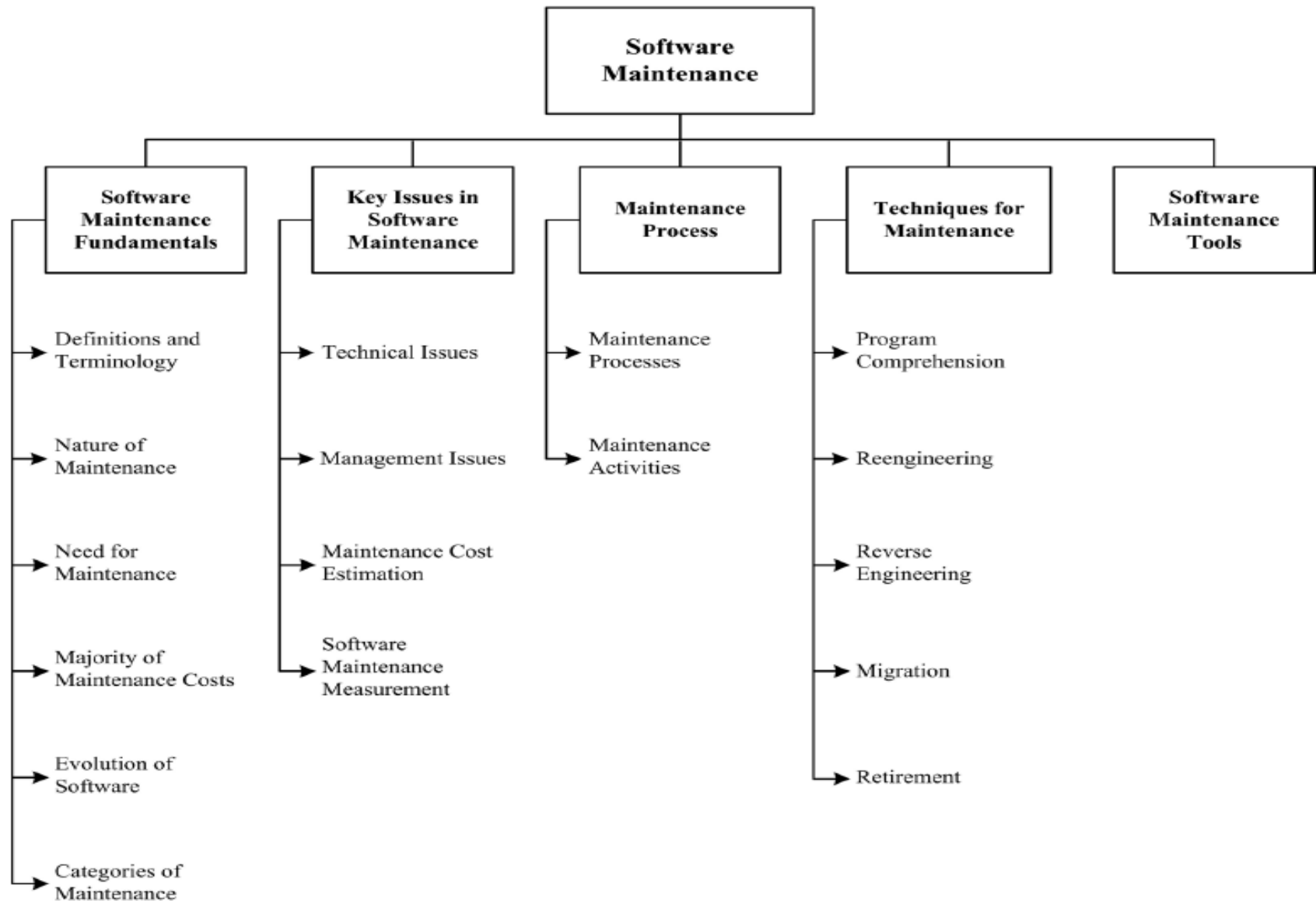
*Ingénierie du logiciel — Processus du cycle de vie du logiciel —
Maintenance*



SOFTWARE MAINTENANCE
Concepts and Practice
SECOND EDITION

Penny Grubb
Armstrong A Takang

Tópicos de Manutenção de Software



Definições e Terminologia

- ▶ Manutenção de Software é parte integral do ciclo de vida do software.
 - ▶ Entretanto, ela não tem recebido o mesmo grau de atenção que as outras fases.
- ▶ É definida como a totalidade das atividades requeridas para prover suporte ao software.
 - ▶ Atividades são realizadas durante o estágio de pré-entrega (*predelivery*), bem como durante o estágio de pós-entrega (*postdelivery*).
 - ▶ **Atividades de pré-entrega** incluem o planejamento para as operações de pós-entrega, manutenibilidade e determinação da logística para as atividades de transição.
 - ▶ **Atividades de pós-entrega** incluem modificação no software, treinamento e operação ou interface para *help desk*.



Natureza da Manutenção

- ▶ Pedidos de modificação são *logged and tracked*, o impacto das mudanças propostas é determinado, código e outros artefatos são modificados, teste é conduzido e uma nova versão do produto de software é liberada.
- ▶ Treinamento e suporte diário também são providos a usuários.
- ▶ Mantenedores podem aprender a partir do conhecimento dos desenvolvedores do código.
 - ▶ Em algumas situações, o desenvolvedor inicial não pode ser localizado ou está envolvido com outras tarefas, o que cria um **desafio** adicional para mantenedores.



Necessidade de Manutenção

- ▶ Manutenção é necessária para assegurar que o software continua a satisfazer os requisitos/necessidades do usuário.
- ▶ Manutenção é aplicável para software que é desenvolvido através de qualquer **modelo de ciclo de vida** (espiral ou linear, por exemplo).
- ▶ Produtos de software mudam devido a ações **corretivas** e **não-corretivas**.



Maioria dos Custos de Manutenção

- ▶ Manutenção consome a **maior fatia dos recursos financeiros** em um ciclo de vida.
- ▶ Uma percepção comum é que manutenção meramente **conserta falhas**.
- ▶ Entretanto, estudos e pesquisas feitas ao longo dos anos tem indicado que a maioria, **mais de 80%**, da manutenção de software é usada para ações **não-corretivas**.

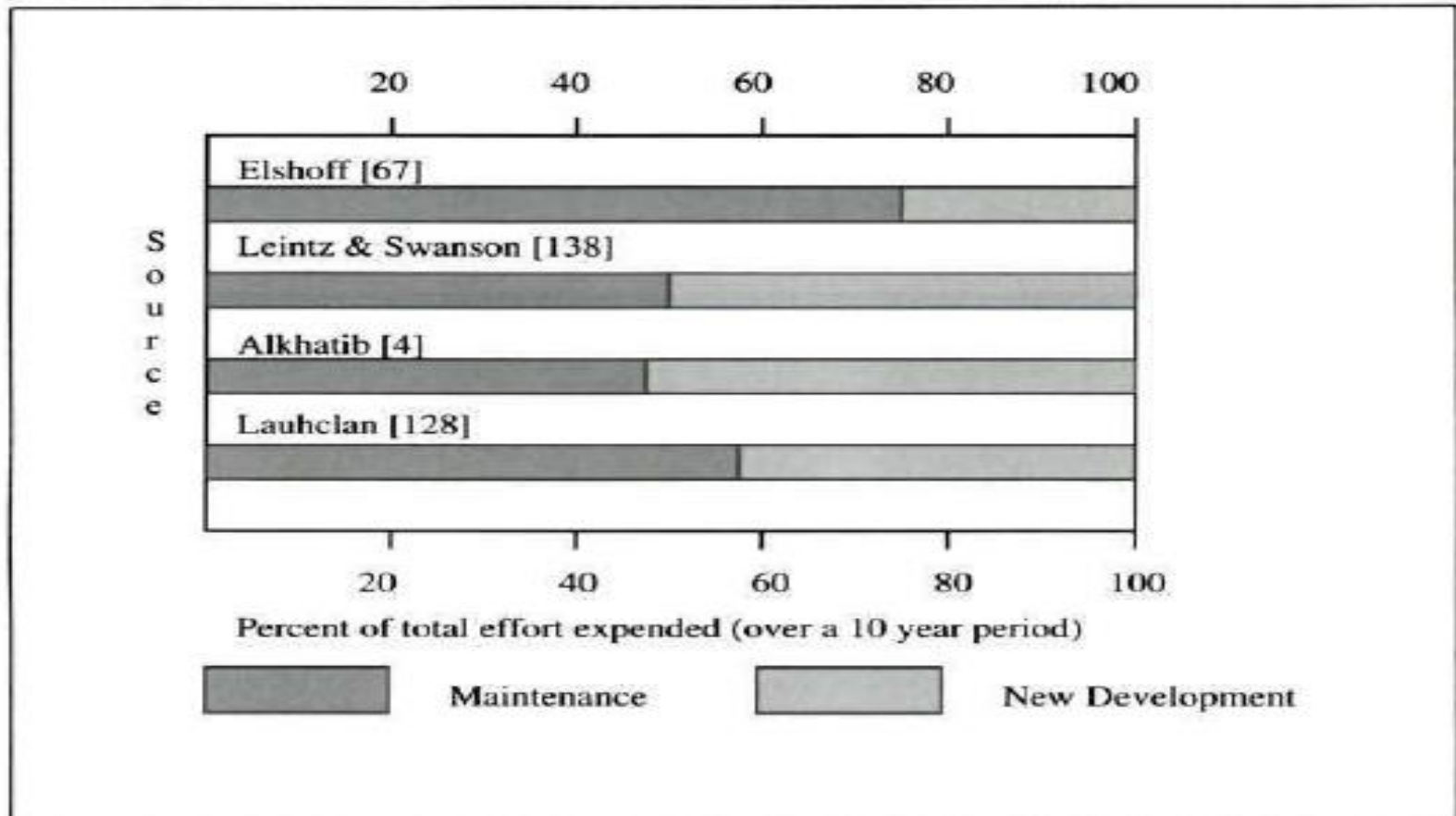


Maioria dos Custos de Manutenção

- ▶ Agrupar **melhorias** e **correções** conjuntamente em relatórios de gestão contribui para alguns **equivocos** em relação ao alto custo das correções.
- ▶ Entender as **categorias de manutenção de software** ajuda a entender a estrutura dos custos de manutenção de software.
- ▶ Entender os **fatores que influenciam a manutenibilidade** do software pode ajudar a conter custos.
 - ▶ Fatores de ambiente organizacional: políticas, competição, processos, produtos e pessoal.

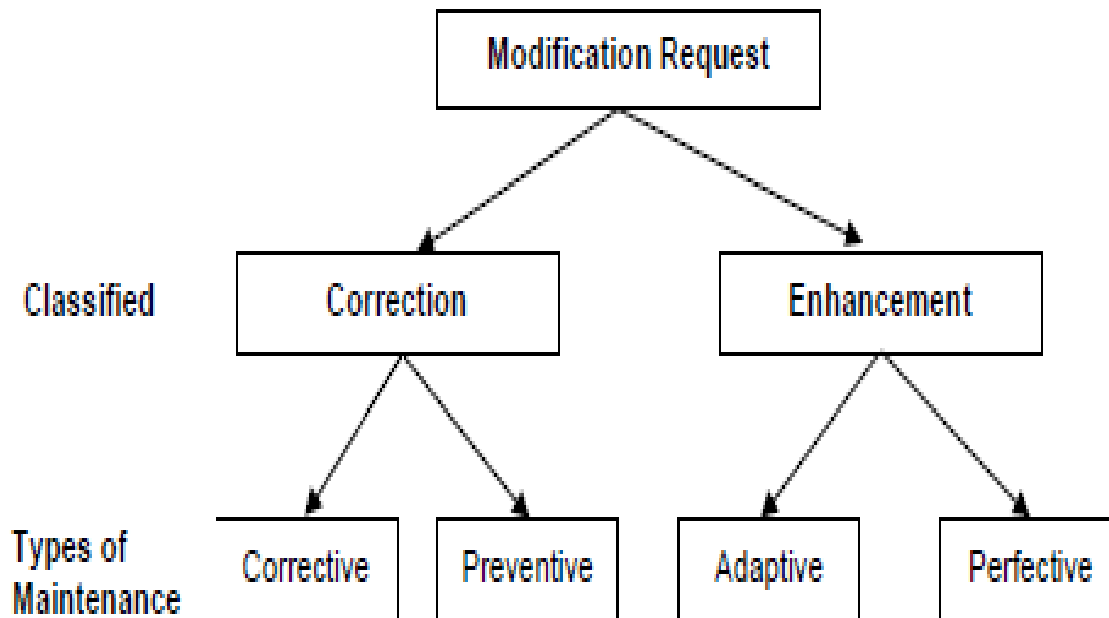


Maioria dos Custos de Manutenção



The maintenance costs range from 49% for a pharmaceutical company[4] to 75% for an automobile company [67].

Categorias de Manutenção



Enhancement is responsible for roughly 60 percent of software maintenance costs.

Error correction is roughly 17 percent.

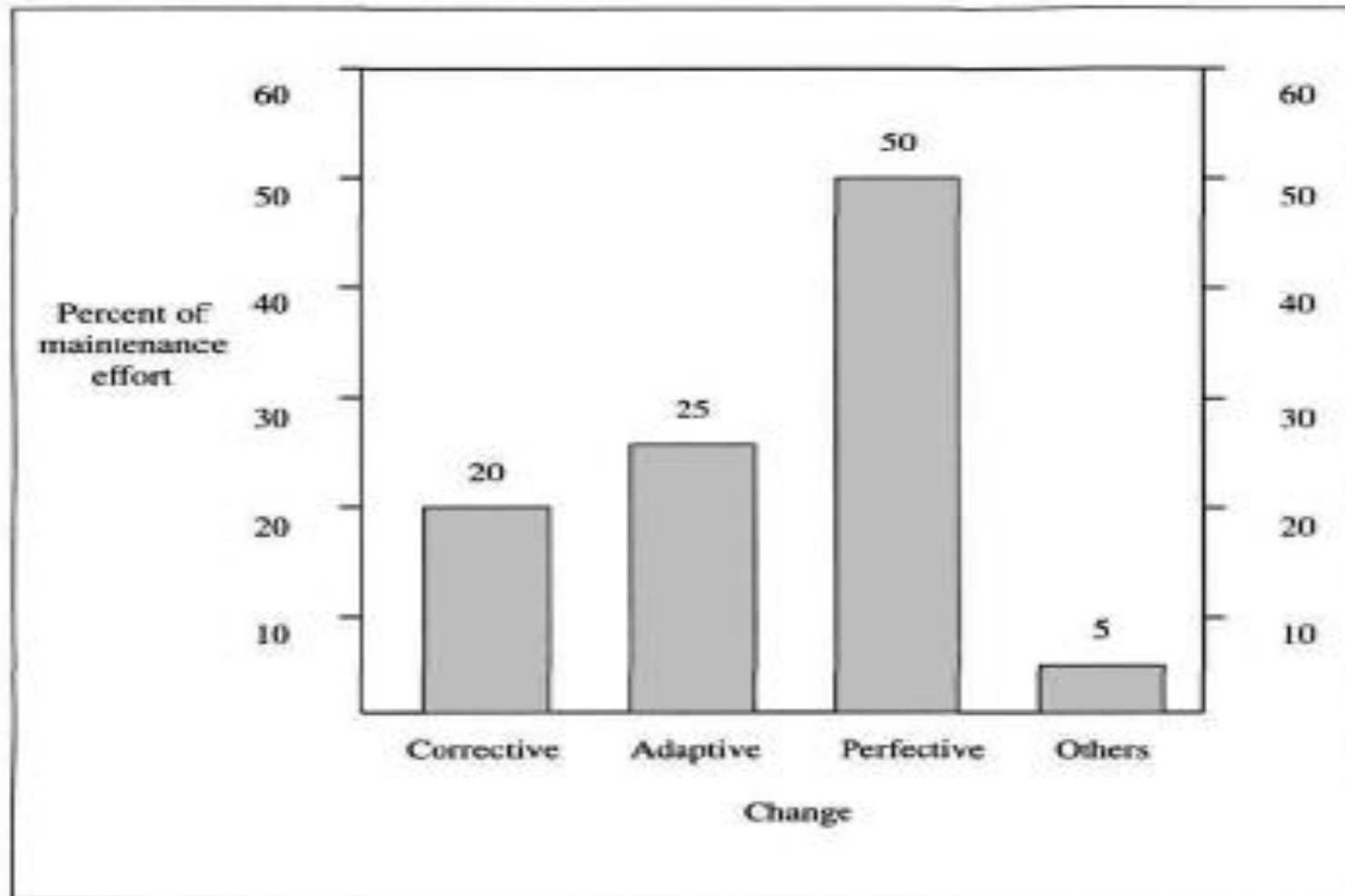
So, software maintenance is largely about **adding new capability to old software, not about fixing it.**

Categorias de Manutenção

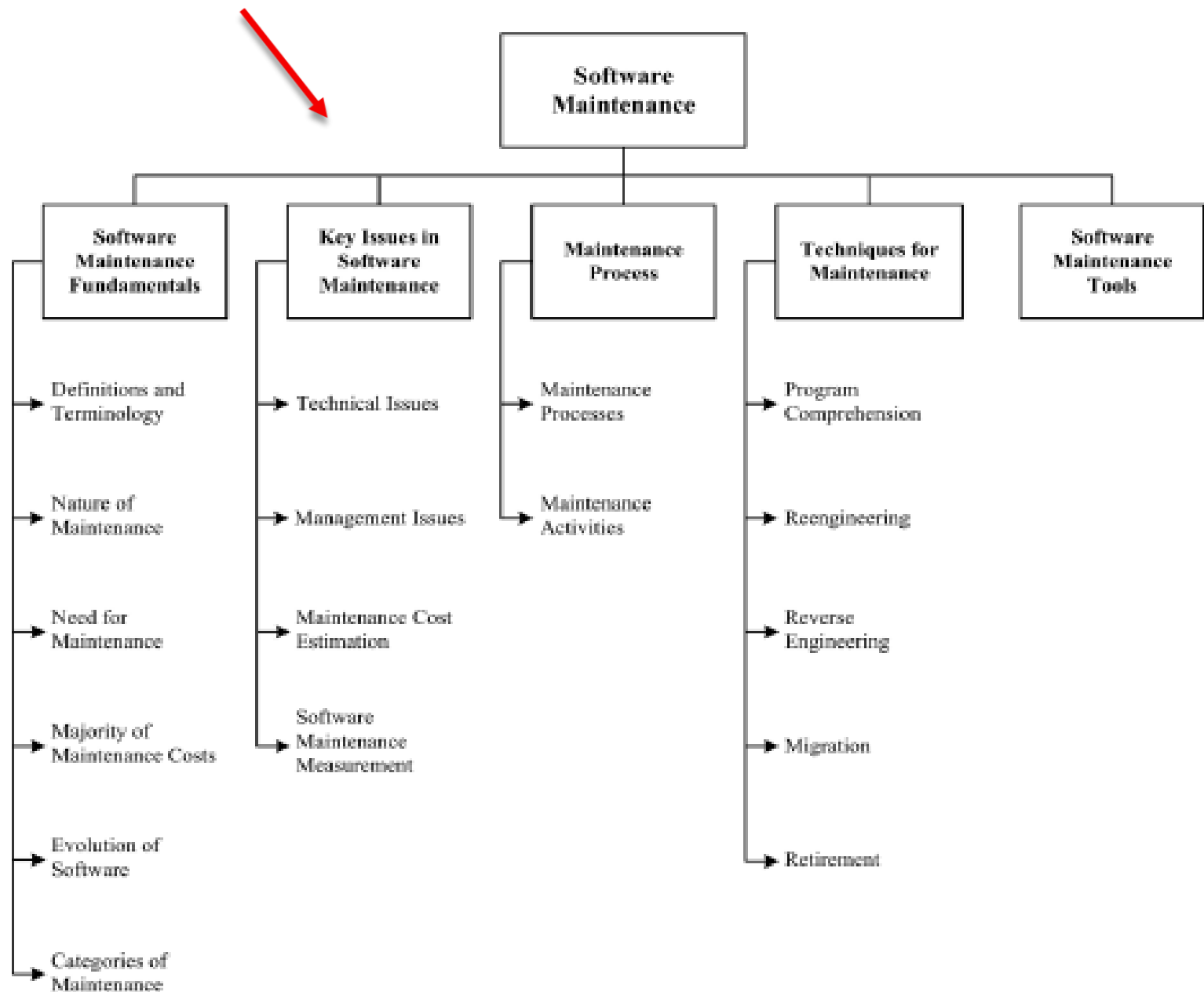
- ▶ **Manutenção Corretiva:** modificação reativa (ou reparação) de um produto de software para corrigir problemas descobertos.
- ▶ **Manutenção Preventiva:** modificação de um produto de software para detectar e corrigir falhas latentes no produto antes que elas se tornem falhas operacionais.(IEEE 14764)
- ▶ **Manutenção Adaptativa:** modificação de um produto de software para mantê-lo utilizável em um ambiente mudado ou em mudança.
- ▶ **Manutenção Perfectiva:** modificação de um produto de software para prover melhorias para usuários, melhorias da documentação do software e recodificação (refatoração) para melhorar performance, manutenibilidade e outros atributos do software.



Custos por categoria de manutenção



Fonte: Grubb e Takang (2003, p.50)



Questões Chave em Manutenção de Software

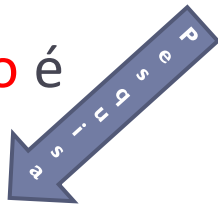
- ▶ Manutenção de Software provê desafios **técnicos** e de **gestão** para engenheiros de software.
 - ▶ Tentar encontrar uma falha em um software que contém um grande número de linhas de código que outro engenheiro de software desenvolveu.
 - ▶ Codificar a próxima versão, enquanto envia *patches* de emergência para a versão atual
- ▶ Questões Técnicas
- ▶ Questões de Gestão
- ▶ Questões de Estimativa de Custos
- ▶ Questões de Medição



Questões Técnicas

► *Entendimento Limitado*

- Refere-se a quão rapidamente um engenheiro de software pode entender onde fazer uma mudança ou correção no software que ele/ela não desenvolveram.
- Aproximadamente **metade do esforço total de manutenção** é dedicado a entender o software a ser modificado.
- **Compreensão de software (*software comprehension*)** é de grande interesse para engenheiros de software.



► **Teste**

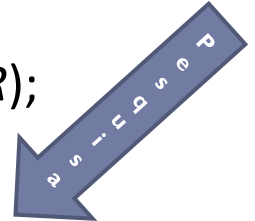
- O mantenedor deve replicar ou verificar problemas através da execução de testes apropriados.
 - **Teste de Regressão** (verifica se as modificações não causaram problemas não intencionais)



Questões Técnicas

▶ Análise de Impacto

- ▶ Identifica todos os sistemas e produtos de software afetados por um pedido de mudança e desenvolve uma estimativa dos recursos necessários para cumprir a mudança.
- ▶ Adicionalmente, o risco de fazer a mudança é determinado.
- ▶ **Tarefas** típicas da análise de impacto:
 1. Analisar pedido de modificação (*Modification Request - MR*);
 2. Replicar ou verificar o problema;
 3. Desenvolver opções para implementar a modificação;
 4. Documentar a MR/PR, as opções de execução e os **resultados**;
 5. Obter aprovação para a modificação selecionada.



Questões de Gestão

▶ *Staffing*

- ▶ Pessoal de manutenção de software são frequentemente visualizados como “*second-class citizens*,” portanto, “sofredores”.

▶ *Aspectos Organizacionais de Manutenção*

- ▶ Organizações de engenharia de software podem manter o **desenvolvedor original** ou **time de manutenção permanente**.

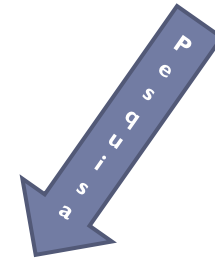
▶ *Outsourcing (terceirização)*

- ▶ Organizações estão terceirizando portfólios de software, incluindo **manutenção de software**.
- ▶ Terceirizados (*outsourcers*) necessitarão investir em uma estrutura de manutenção e um *help desk* deve ser operado por pessoas com conhecimento nativo da língua dos usuários.



Estimativa dos Custos de Manutenção

- ▶ Custos de um projeto de manutenção: **peçoal, máquinas, tempo e dinheiro** gastos na mudança.
- ▶ Análise de Impacto identifica todos os sistemas e produtos de software afetados pelo pedido de mudança e desenvolve uma **estimativa dos recursos** necessários para cumprir a mudança.
- ▶ *“the two most popular approaches to estimating resources for software maintenance are the use of parametric models and the use of experience”* (IEEE 14764)
 - ▶ Function Point Models
 - ▶ Use Case Points (UCPs)
 - ▶ Story Points (agile)
 - ▶ COCOMO II (Wazlawick, 2013)
- ▶ Uma maneira de estimar custos de uma tarefa de manutenção é a partir de **dados históricos coletados** de uma tarefa similar.
 - ▶ Dados históricos para estimação de manutenção devem ser providos a partir de uma programa de mensuração.



Frequently Forgotten Fundamental Facts about Software Engineering

Robert L. Glass

Maintenance

M1. Quality and maintenance have an interesting relationship (see Q3 and Q4).

M2. Maintenance typically consumes about 40 to 80 percent (60 percent average) of software costs. Therefore, it is probably the most important life cycle phase.

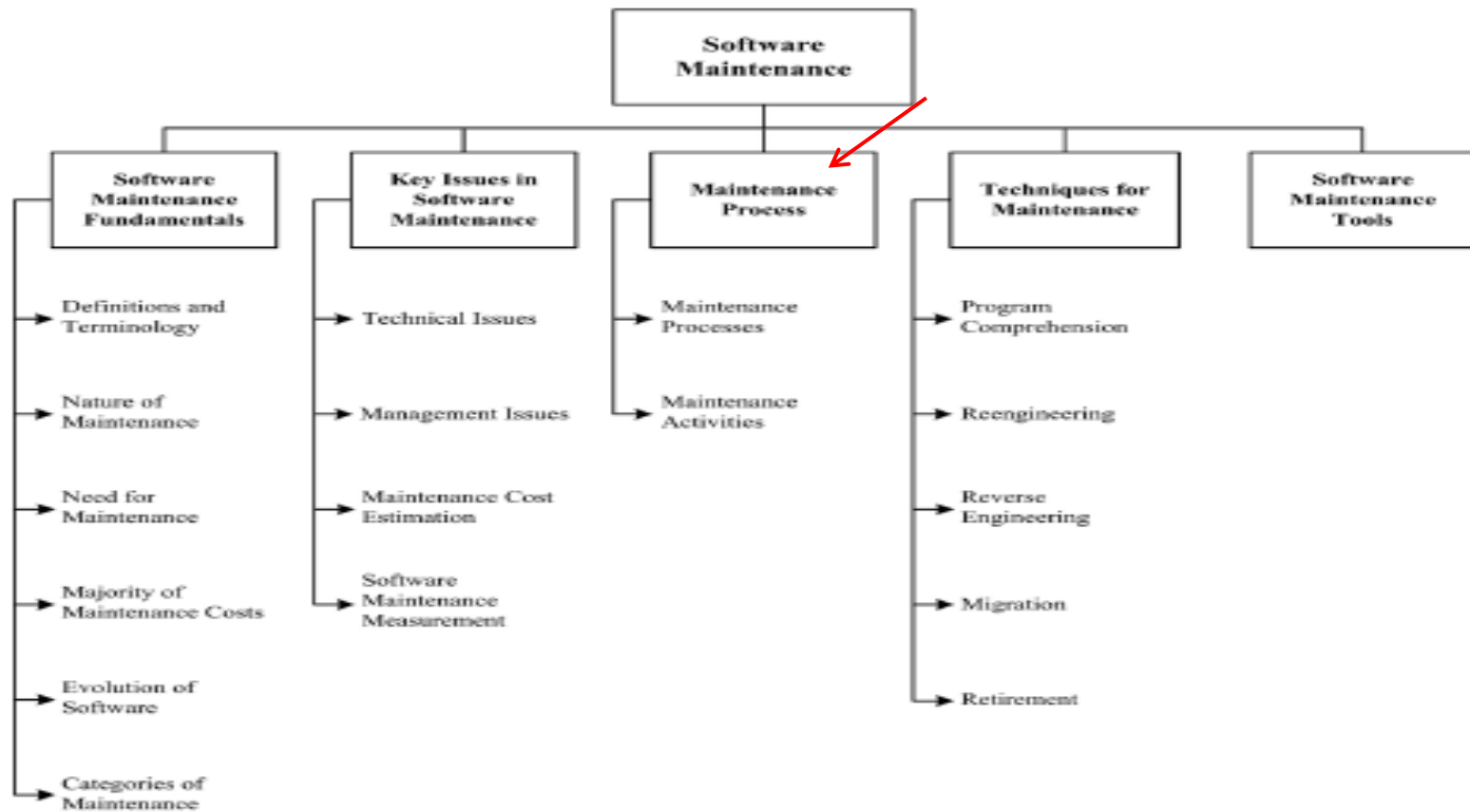
M3. Enhancement is responsible

for roughly 60 percent of software maintenance costs. Error correction is roughly 17 percent. So, software maintenance is largely about adding new capability to old software, not about fixing it.

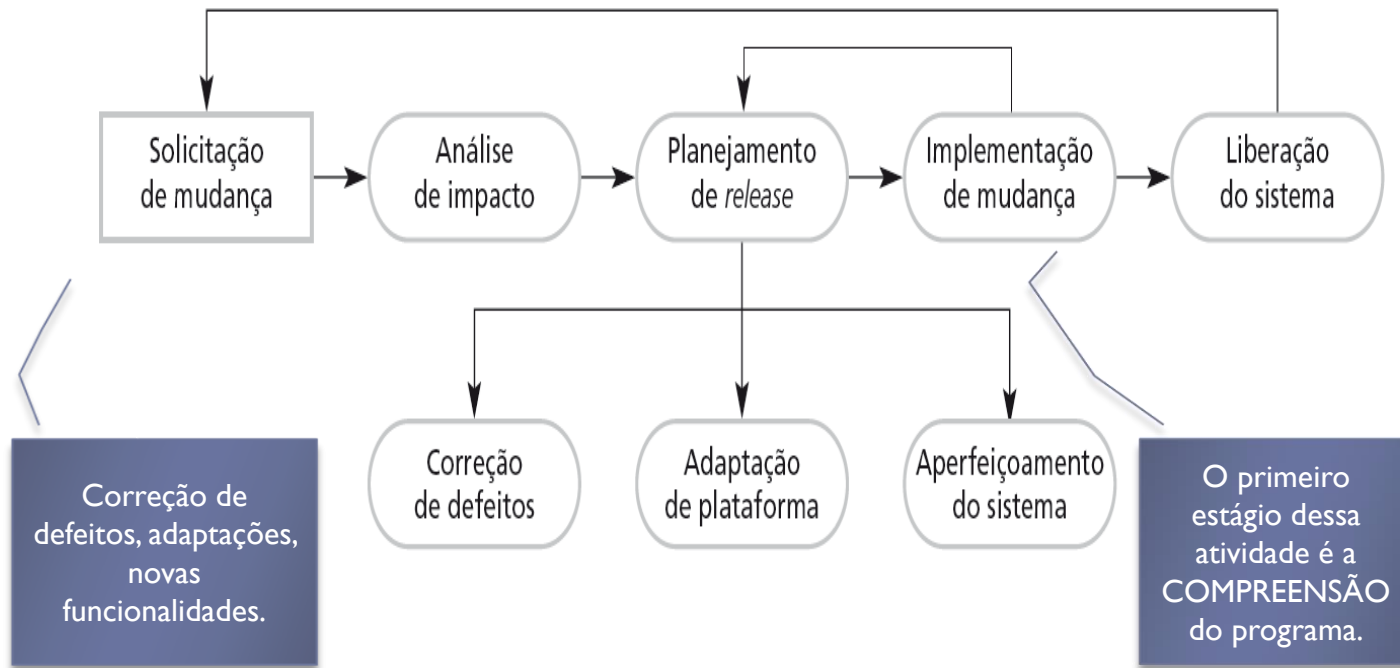
M4. The previous two facts constitute what you could call the “60/60” rule of software.

M5. Most software development tasks and software maintenance tasks are the same—except for the additional maintenance task of “understanding the existing product.” This task is the dominant maintenance activity, consuming roughly 30 percent of maintenance time. So, you could claim that maintenance is more difficult than development.



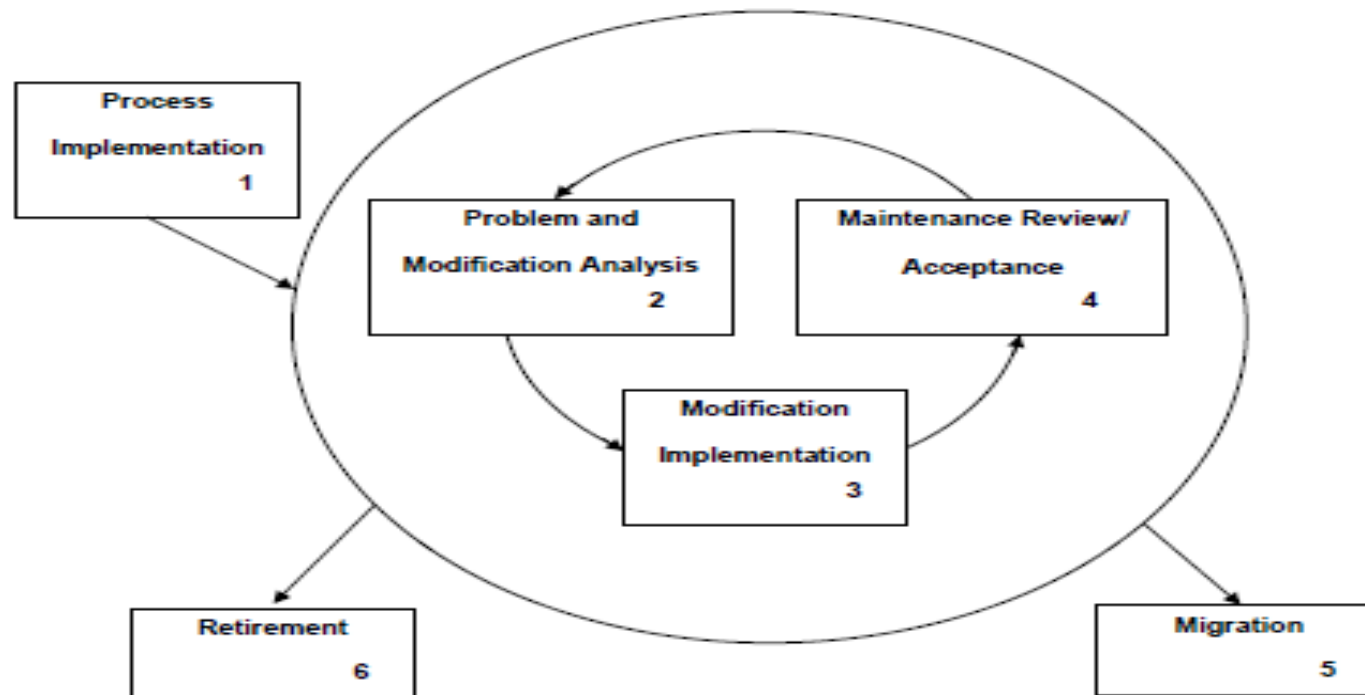


Processo de Manutenção



Arthur (1988)

Processo de Manutenção - **ISO/IEC/EEE 14764**



1- Implementação do Processo

- ▶ 1. Desenvolver **plano** e **procedimentos** que serão executados durante o processo de manutenção
 - ▶ O **plano de manutenção** deve documentar a **estratégia** a ser usada para manter o sistema.
 - ▶ Os **procedimentos de manutenção** devem prover uma abordagem detalhada de como cumprir a manutenção.



1- Implementação do Processo

- ▶ 1. Desenvolver **plano** e **procedimentos**
 - ▶ a) Auxiliar o adquirente no desenvolvimento do **conceito de manutenção**.
 - ▶ b) Auxiliar o adquirente na determinação do **escopo** da manutenção.
 - ▶ c) Auxiliar o adquirente na análise de **alternativas de organização** de manutenção.
 - ▶ d) Assegurar designação por escrito como **mantenedor** do produto de software.
 - ▶ e) Realizar análises de recursos.
 - ▶ f) Estimar custos de manutenção.
 - ▶ g) Realizar uma avaliação de manutenibilidade do sistema.
 - ▶ h) Determinar os requisitos de transição.
 - Não são requisitos que o aplicativo precisa para funcionar, mas aqueles essenciais para colocar o aplicativo em produção. ex. Dados, treinamento.
 - ▶ i) Determinar os marcos da transição.
 - ▶ j) Identificar o processo de manutenção que será usado.
 - ▶ k) Documentar o processo de manutenção na forma de procedimentos operacionais.



1. Implementação do Processo

2. Estabelecer procedimentos de **pedido de modificação**

1. Deve-se estabelecer procedimentos para **receber, registrar e rastrear** relatórios de problemas e pedidos de modificação dos usuários e para fornecer ***feedback*** para os usuários.

3. Implementar Gestão de Configuração

1. *Configuration Management Process* (ISO/IEC 12207 sub-clause 6.2)



2 – Análise do Problema e Modificação

- ▶ Antes de modificar o sistema, o mantenedor deve analisar o pedido para determinar o seu **impacto** sobre a organização, o sistema existente e os sistemas de interface; desenvolver e documentar possíveis soluções recomendadas; e obter a aprovação para implementar a solução desejada.
1. Analisar pedido
 - a) **Tipo**: corretiva, preventiva, etc.
 - b) **Escopo**: por exemplo, tamanho da modificação, tempo para modificar, custo envolvido;
 - c) **Criticidade**: por exemplo, impacto na performance, segurança, etc.
 2. Replicar ou verificar o problema;
 3. Desenvolver opções para implementar a modificação;
 4. Documentar o pedido e as opções de execução;
 5. Obter aprovação para a opção de modificação selecionada.



2.1 Analisar pedido - ISO/IEC 14764 Pag.10

- a) Determine se o mantenedor está adequadamente preparado para implementar a modificação proposta.
 - b) Determine se o sistema está adequadamente orçado para implementar a modificação proposta.
 - c) Determine se recursos suficientes estão disponíveis e se essa modificação afetará os projetos em andamento ou projetados (pode não ser necessário para os PRs).
 - d) Determine os problemas operacionais a serem considerados. Por exemplo, quais são as modificações previstas nos requisitos de interface do sistema, a vida útil esperada do sistema, as prioridades operacionais, os impactos de segurança, se não forem implementados? (pode não ser necessário para PRs).
 - e) Determinar prioridade de tratamento.
 - f) Classificar o tipo de manutenção.
 - g) Determine o impacto para usuários atuais e futuros.
 - h) Determine as implicações de segurança e proteção (pode não ser necessário para PRs).
 - i) Identificar os efeitos cascata.
-



2.1 Analisar pedido

- ▶ j) Avalie quaisquer restrições de software ou hardware que possam resultar das modificações.
- ▶ k) Determinar os custos a curto e longo prazo.
- ▶ l) Determinar o valor do benefício de fazer a modificação.
- ▶ m) Determinar o impacto nas agendas existentes.
- ▶ n) Documentar qualquer risco de software and projeto resultante da análise de impacto.
- ▶ o) Determinar o nível de teste e avaliação necessários.
- ▶ p) Determinar o custo de gerenciamento estimado para implementar a modificação.



4. Revisão/Aceitação da Manutenção

- ▶ Essa atividade garante que as modificações no sistema estejam corretas e que foram realizadas de acordo com os padrões aprovados, usando a metodologia correta.
- ▶ Tarefas: As revisões são conduzidas para garantir que as modificações estejam corretas e que a aprovação seja obtida para conclusão satisfatória das modificações.
 - ▶ Revisão
 - ☐ a) Rastrear o MR / PR dos requisitos, projeto ao código.
 - ☐ b) Verificar a testabilidade do código.
 - ☐ ...
 - ▶ Aprovação



Processo de Manutenção - IEEE 14764

▶ **4 – Retirement (Aposentadoria)**

- ▶ Uma vez que um produto de software chegou ao fim da sua vida útil, deve ser aposentado.
- ▶ Uma análise deve ser realizada para auxiliar na **tomada de decisão** para se aposentar um produto de software.
- ▶ A análise deve levar em conta vários fatores.
- ▶ O produto de software pode ser substituído por um novo produto de software, mas em alguns casos, não será substituído.
- ▶ A fim de se aposentar um produto de software, o mantenedor deve determinar as ações necessárias para realizar a reforma e, em seguida, desenvolver e documentar os passos necessários para efetuar a **aposentadoria**.
- ▶ Deve ser dada atenção ao acesso aos dados armazenados pelo produto de software aposentado.



Processo de Manutenção - IEEE 14764

▶ VI – Aposentadoria (Retirement)

▶ Tarefas:

- ▶ Desenvolver um Plano de Aposentadoria,
- ▶ Notificar os usuários da aposentadoria,
- ▶ Execução de operações e treinamento paralelas,
- ▶ Fornecer uma notificação de conclusão e
- ▶ Arquivar os dados.



Processo de Manutenção - IEEE 14764

▶ V – Migration (Migração)

- ▶ Durante a vida de um sistema ele pode ter que ser modificado para funcionar em ambientes diferentes.
- ▶ A fim de migrar um sistema para um novo ambiente, o mantenedor precisa determinar as ações necessárias para realizar a migração e, em seguida, desenvolver e documentar os passos necessários para efetuar a migração.
- ▶ Tarefas:
 - ▶ desenvolver um Plano de Migração,
 - ▶ notificar os usuários sobre a migração,
 - ▶ fornecer treinamento,
 - ▶ fornecer uma notificação de conclusão,
 - ▶ avaliar o impacto do novo ambiente e
 - ▶ arquivar os dados .

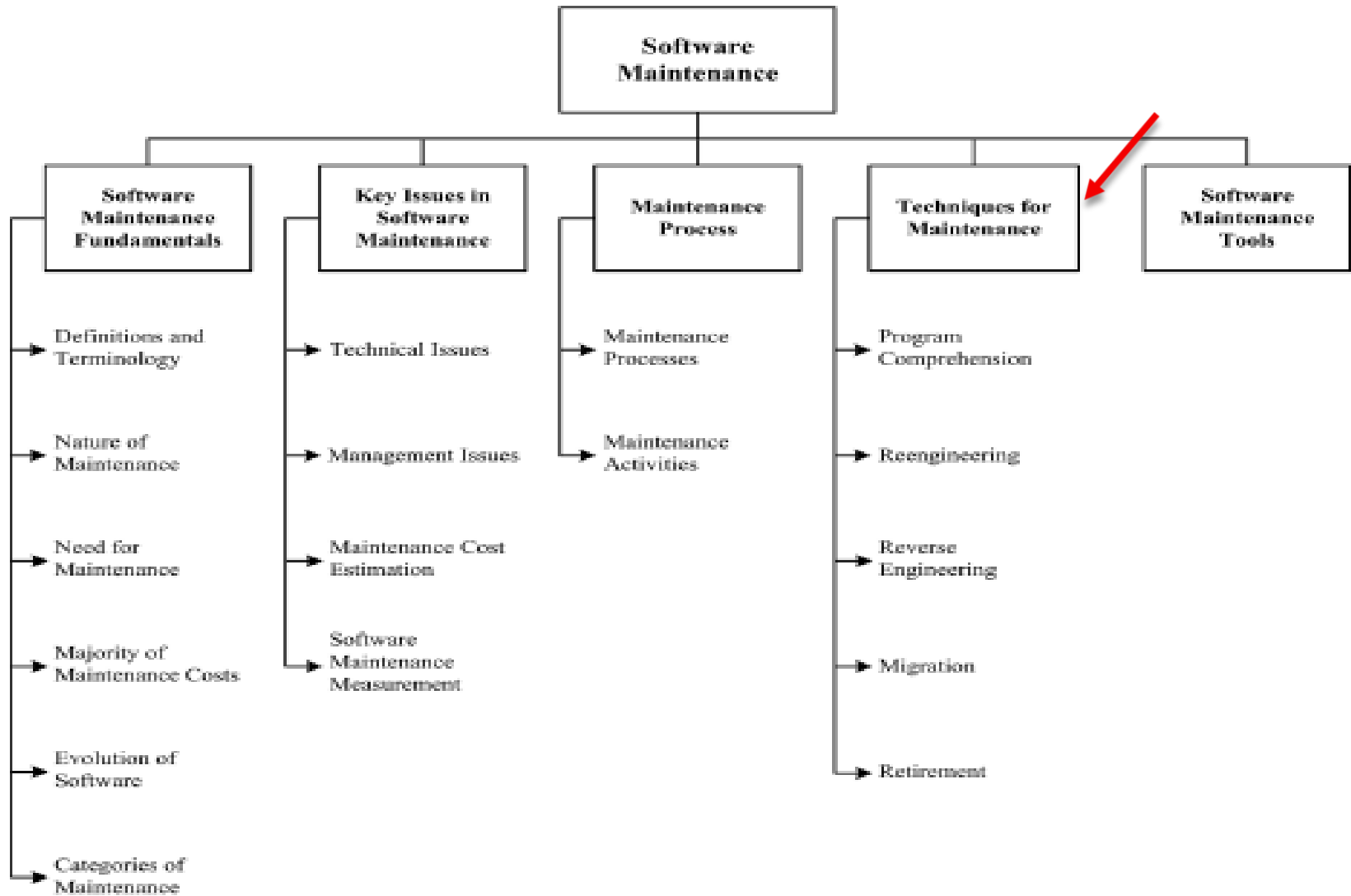




Técnicas para Manutenção

- ▶ *Program Comprehension or Program Understanding*
 - ▶ Ter um conhecimento geral do que o software faz e como ele se relaciona com o ambiente;
 - ▶ Identificar no sistema **onde** as mudanças causam efeitos;
 - ▶ Ter um conhecimento profundo de **como** corrigir as partes ou executar o trabalho;
 - ▶ Dados da indústria e outras fontes indicam que **aproximadamente metade do esforço total** gasto em efetuar mudanças é usado no entendimento de programas;
 - ▶ At Hewlett Packard it was estimated that reading code (a fundamental element in comprehension) costs \$200 million a year.
- ▶ *Reengineering*
 - ▶ *Reverse Engineering*
- ▶ *Migration*
- ▶ *Retirement*





Ferramentas para manutenção

- ▶ Um meio potencial de conter os custos de manutenção de software é usar ferramentas CASE .
- ▶ Estas ferramentas auxiliam as atividades de manutenção de software.



Referências

- ▶ SWEBOK. Guide to the Software Engineering Body of Knowledge. Version 3.0.
- ▶ Grubb, P. ; Takang, A.A. Software Maintenance: Concepts and Practice, 2nd ed., World Scientific Publishing, 2003.
- ▶ IEEE/ISO/IEC 14764:2006 Software Engineering — Software Life Cycle Processes — Maintenance
- ▶ Wazlawick, R. [Engenharia de Software - Conceitos e Práticas](#). Ed. Elsevier. 2013.

