

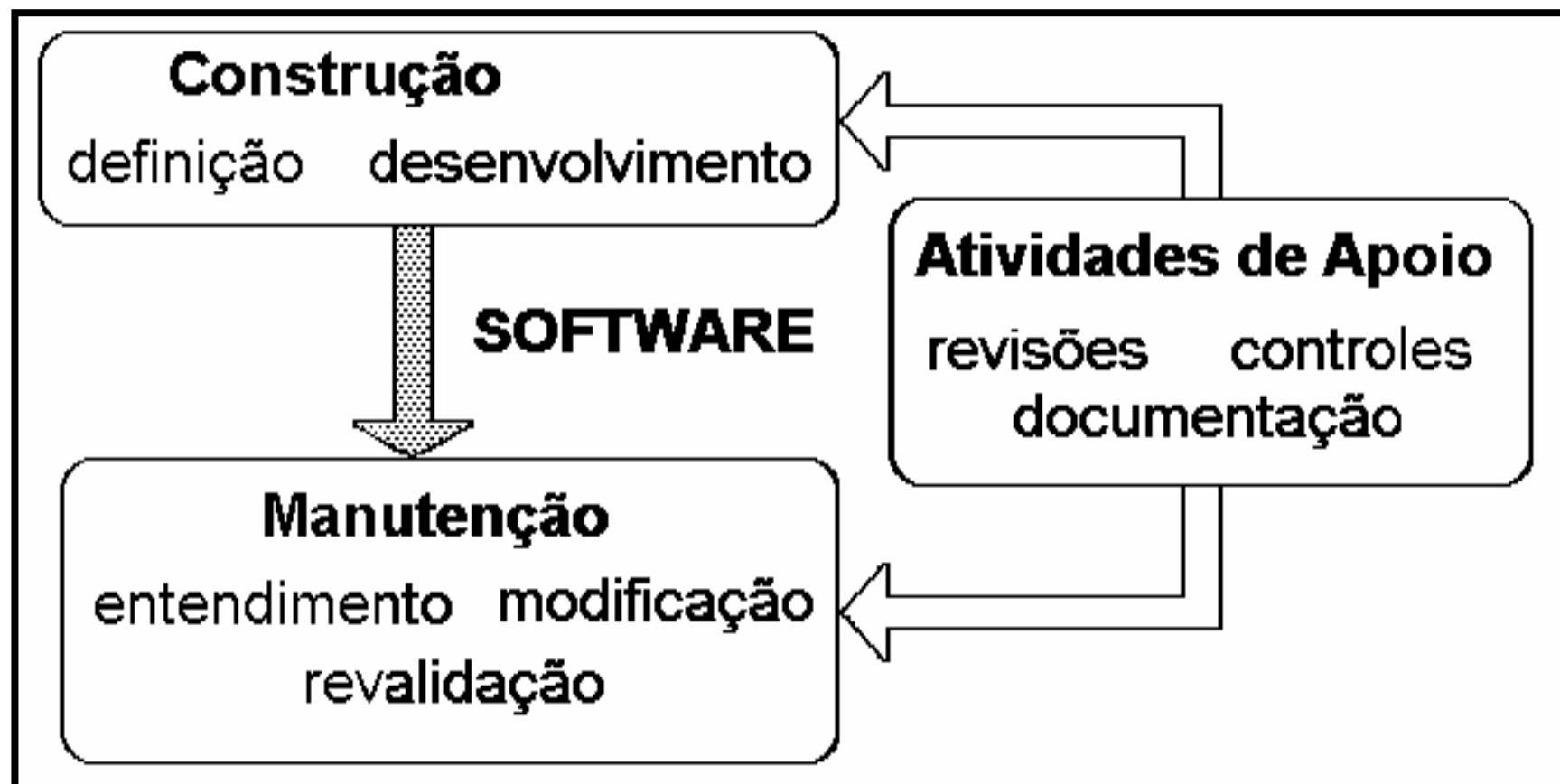


# **Manutenção de Software**

Prof. Ricardo A. Ramos



# Ciclo de Vida de Software



# Manutenção de Software

- Alterações efetuadas no software depois de sua liberação.
- As alterações ocorrem por diversas razões.
- As razões para as alterações determinam a categoria de manutenção.

# Categorias de Manutenção

1) Identificar e Corrigir Erros

⇒ *Manutenção Corretiva*

# Categorias de Manutenção

2) Adaptar o Software ao Ambiente

⇒ *Manutenção Adaptativa*

# Categorias de Manutenção

- 3) Atender Pedidos do Usuário para  
Modificar Funções Existentes, Incluir  
Novas Funções e Efetuar Melhoramentos  
Gerais  
⇒ *Manutenção Perfectiva (ou  
aperfeiçoativa)*

# Categorias de Manutenção

- 4) Melhorar a manutenibilidade ou confiabilidade futuras e fornecer uma base melhor para futuros melhoramentos  
⇒ *Manutenção Preventiva*

# A Fase de Manutenção de Software

- Fase mais problemática do Ciclo de Vida de Software
- Pode despende mais de 70% de todo esforço de uma Organização
- Esses sistemas devem continuar rodando e as alterações são inevitáveis

# A Fase de Manutenção de Software

- Por que é exigida tanta Manutenção e por que é despendido tanto Esforço nessa atividade?
  - Idade Média de 10 a 15 anos
  - Quando foram implementados , o tamanho do programa e espaço de armazenamento eram o principal interesse
  - Migração Para Novas Plataformas
  - Sistemas mal estruturados

# A Fase de Manutenção de Software

- Por que é exigida tanta Manutenção e por que é despendido tanto Esforço nessa atividade? (cont)
  - Melhoramentos Para Atender Novas Necessidades
  - Nenhuma preocupação com a Arquitetura Global
  - Codificação, Lógica e Documentação ruins

# Custo de Manutenção

- Custo da manutenção como uma porcentagem do orçamento do software

	Lientz e Swanson ( 1980)	Arfa et alii (1988-1989)
Manutenção	48,8 %	44,63%
Desenvolvimento	43,3 %	46,59 %
Outros	7,9 %	8,78 %

# Custo de Manutenção

- Outros Custos não Monetários
  - Adiamento de oportunidades de desenvolvimento
  - Redução da qualidade global do software
  - Insatisfação do cliente
  - Insatisfação do pessoal de manutenção

# Custo de Manutenção

- Custo Final da Manutenção
  - Diminuição dramática na produtividade
- (Boehm )
  - Custo do desenvolvimento \$250.00 por linha de código
  - Custo da manutenção \$1000.00 por linha de código

# Custo de Manutenção

- O Custo de manutenção pode ser dividido em:
  - tentar entender o que o software faz
  - interpretar as estruturas de dados, as características de interface e limites de desempenho
  - analisar, avaliar, projetar, codificar e testar as modificações

*(Atividades Produtivas)*

# Problemas da Manutenção

- A maioria dos problemas com a manutenção do software é causada por deficiências na maneira como o software foi planejado e desenvolvido

# Problemas da Manutenção

## PROBLEMAS CLÁSSICOS

- É difícil ou impossível traçar a evolução do software através das várias versões. As alterações não são adequadamente documentadas
- É difícil ou impossível traçar o processo através do qual o software foi criado.

# Problemas da Manutenção

## **PROBLEMAS CLÁSSICOS (cont.)**

- É muito difícil entender programas "de outras pessoas". A dificuldade aumenta conforme o número de elementos na configuração de software diminui.
- "As outras pessoas" freqüente-mente não estão presentes para explicar.

# Problemas da Manutenção

## PROBLEMAS CLÁSSICOS (cont.)

- A documentação não existe, é incompreensível ou está desatualizada.
- A maioria dos softwares não foi projetado para suportar alterações.
- A manutenção não é vista como um trabalho “glamouroso”.

# Manutenibilidade

- A Manutenibilidade pode ser definida qualitativamente como a facilidade com que o software pode ser entendido, corrigido, adaptado e ou melhorado

# Manutenibilidade

- A manutenibilidade é afetada por muitos fatores:
  - *cuidado inadequado com o projeto, codificação e teste*
  - *configuração de software ruim*
  - *disponibilidade de pessoal qualificado de software*

# Manutenibilidade

- A manutenibilidade é afetada por muitos fatores: (cont)
  - *facilidade de manusear o sistema*
  - *uso de linguagens de programação padronizadas*
  - *uso de sistemas operacionais padronizados*
  - *estruturas padronizadas de documentação*

# Manutenibilidade

- A manutenibilidade é afetada por muitos fatores: (cont)
  - *disponibilidade de um computador próprio para a manutenção*
  - *disponibilidade da pessoa ou grupo que desenvolveu o software*
  - *o planejamento para manutenibilidade (fator mais importante que afeta a manutenibilidade)*

# Medidas Quantitativas de Manutenibilidade

- Difícil de quantificar
- Pode-se determinar a manutenibilidade indiretamente considerando-se atributos das atividades de manutenção que podem ser medidos

# Medidas Quantitativas de Manutenibilidade

## Métricas de Manutenibilidade (Gilb)

- *tempo de reconhecimento do problema*
- *tempo de demora administrativa*
- *tempo de coleta de ferramentas de manutenção*
- *tempo de análise do problema*
- *tempo de especificação da alteração*
- *tempo de correção ou modificação*
- *tempo de teste local e global*
- *tempo de revisão da manutenção*

# Medidas Quantitativas de Manutenibilidade

- A manutenibilidade pode ser medida indiretamente considerando medidas da estrutura do projeto e medidas da complexidade do software

# Revisões de Manutenibilidade

- A manutenibilidade deve ser considerada em cada nível do processo de revisão da engenharia de software.

# Revisões de Manutenibilidade

## **ETAPA DE REVISÃO DE REQUISITOS**

observar:

- áreas de melhoramentos futuros
- aspectos de portabilidade do software
- interfaces que poderiam causar impacto à manutenção

# Revisões de Manutenibilidade

## **ETAPA DE REVISÃO DE PROJETO**

avaliar:

- projeto arquitetural
- projeto procedural
- projeto de interfaces e
- projeto de dados

quanto à facilidade de manutenção  
e à qualidade global.

# Revisões de Manutenibilidade

## **ETAPA DE REVISÃO DE CÓDIGO**

dar ênfase:

- ao estilo
- à documentação interna

# Revisões de Manutenibilidade

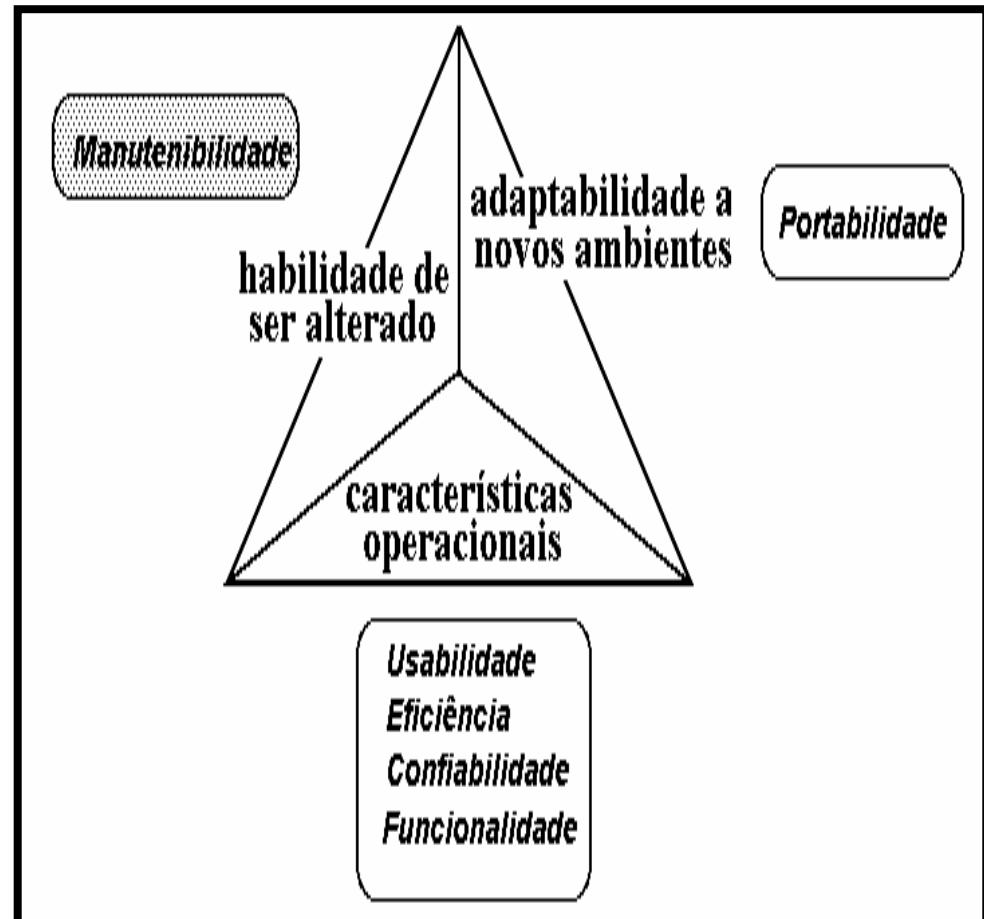
## **ETAPA DE REVISÃO DE TESTE**

- Cada passo do teste pode fornecer indícios sobre partes do software que poderiam exigir manutenção preventiva

# Fatores de Qualidade de Software

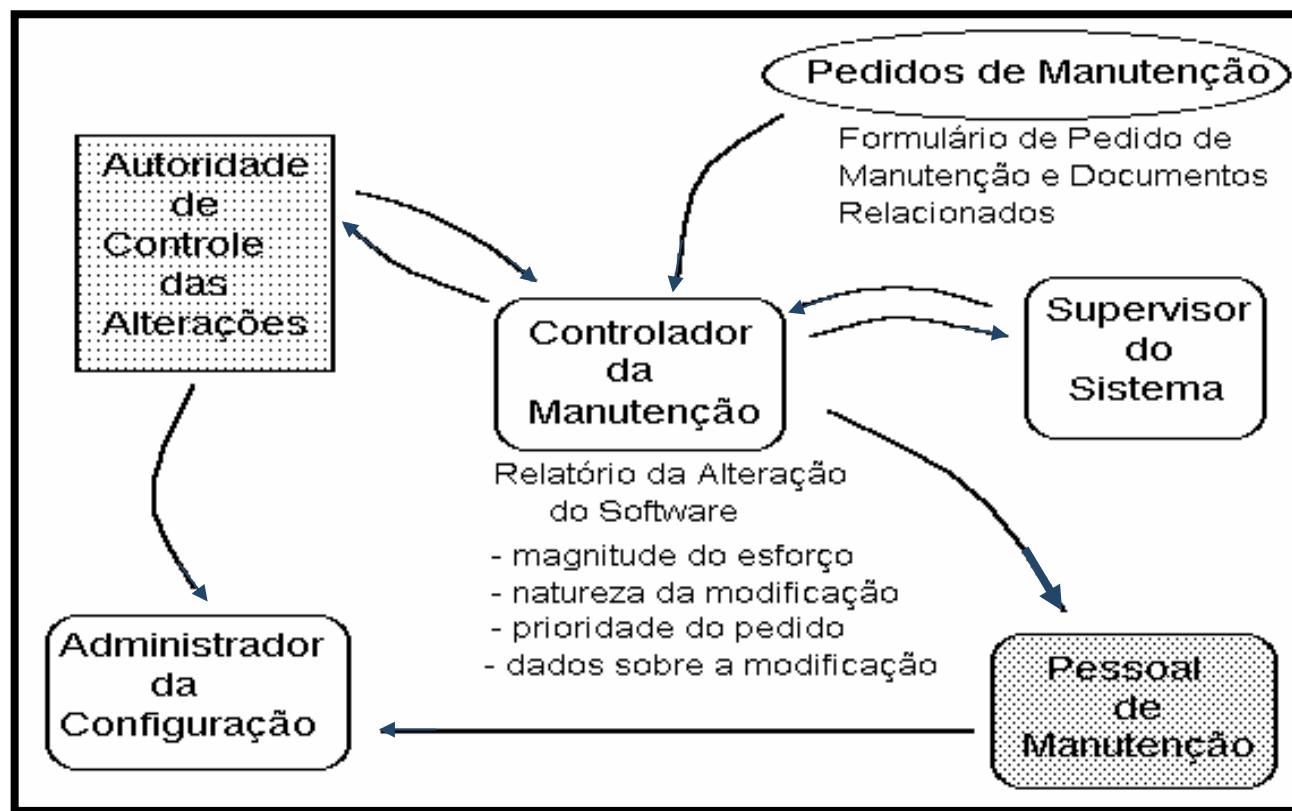
(Norma ISO 9126)

- Os fatores de qualidade de software focalizam três aspectos importantes do software produto



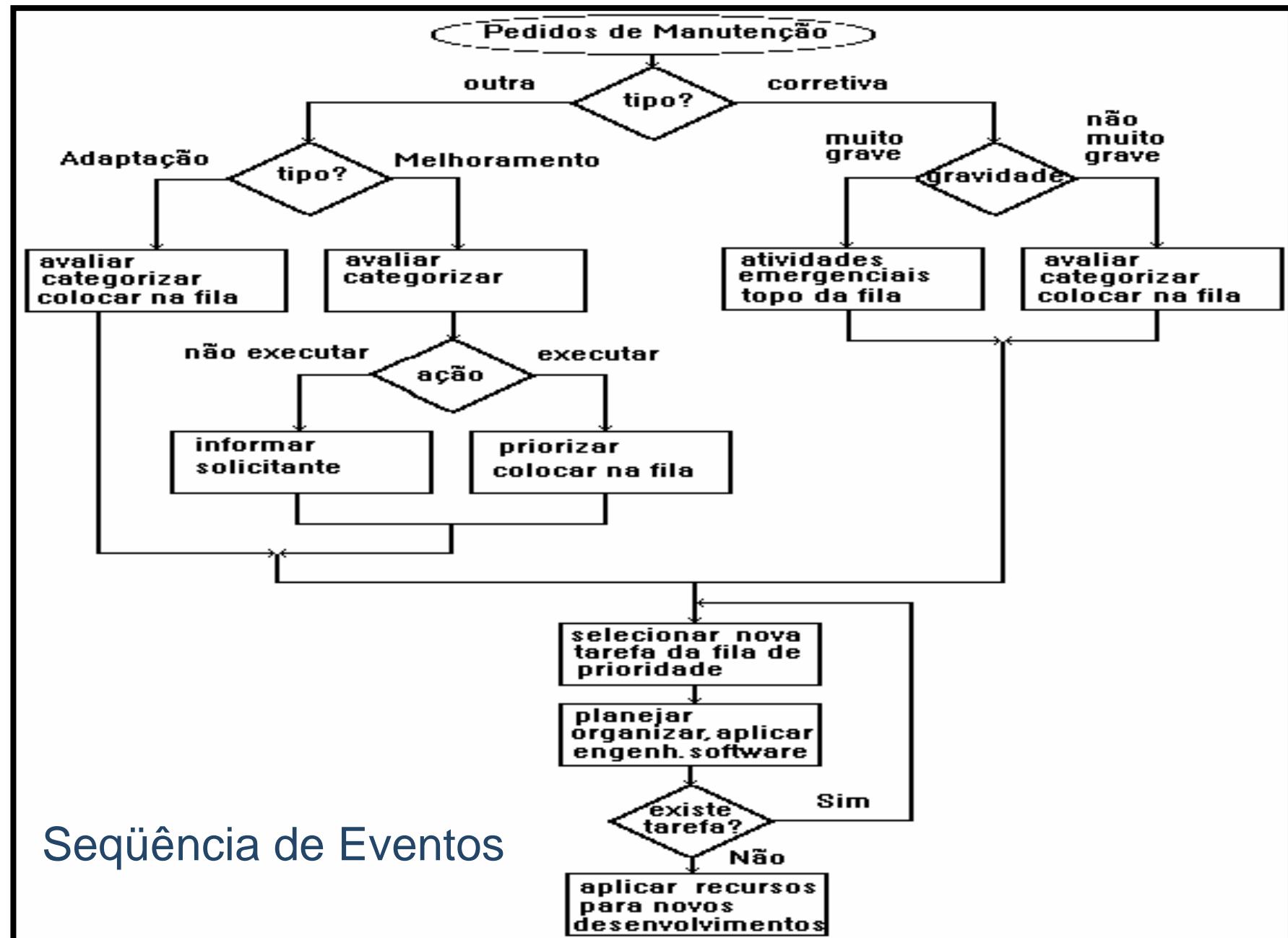
# Tarefas de Manutenção

1) Estabelecer uma organização para a manutenção (“de fato” ou formal)



# Tarefas de Manutenção

- 2) Descrever procedimentos de avaliação e de comunicação
- 3) Definir seqüências padronizadas de eventos (para os pedidos de manutenção)



# Seqüência de Eventos

# Tarefas de Manutenção

- 4) Estabelecer procedimentos para registrar a história das atividades de manutenção

## **Exemplo de Dados que podem ser Armazenados (Swanson)**

- identificação do programa
- número de comandos fonte
- linguagem de programação usada
- data da instalação do programa
- número de execuções do programa desde a instalação
- número de falhas de processamento associadas ao item anterior
- nível e identificação da alteração no programa
- número de comandos fonte adicionados por alteração no programa
- número de pessoas-horas despendidos na manutenção
- identificação do pedido de manutenção
- tipo de manutenção
- datas de início e fim da manutenção

# Tarefas de Manutenção

5) Definir critérios de revisão e avaliação

# **MEDIDAS de DESEMPENHO e MANUTENÇÃO**

(Swanson)

- número médio de falhas de processamento por execução do programa
- pessoas-horas despendido em cada categoria de manutenção
- número médio de pessoas-horas despendido por comando fonte adicionado ou retirado devido a manutenção
- tempo médio de processamento para um pedido de manutenção
- porcentagem de pedidos de manutenção por tipo

# Manutenção de Código Alienígena

- Os programas “alienígenas” são assim chamados porque:
  - Programas com fluxo de controle equivalente a um prato de espaguete
  - Módulos muito grandes
  - Poucas linhas de comentários significativos

# Manutenção de Código Alienígena

- Os programas “alienígenas” são assim chamados porque: (cont.)
  - Não existe nenhum outro elemento da configuração de software, além do código.
  - Nenhum membro do pessoal atual de manutenção trabalhou no desenvolvimento do programa.

# Manutenção de Código Alienígena

- Os programas “alienígenas” são assim chamados porque: (cont.)
  - Nenhuma metodologia de desenvolvimento foi aplicada:
    - Projeto de dados e projeto arquitetural ruins
    - Documentação e registro histórico das alterações incompletos

# Manutenção de Código Alienígena

**O QUE PODE SER FEITO COM CÓDIGO  
"ALIENÍGENA" ?**



Engenharia Reversa e  
Reengenharia

# Engenharia Reversa

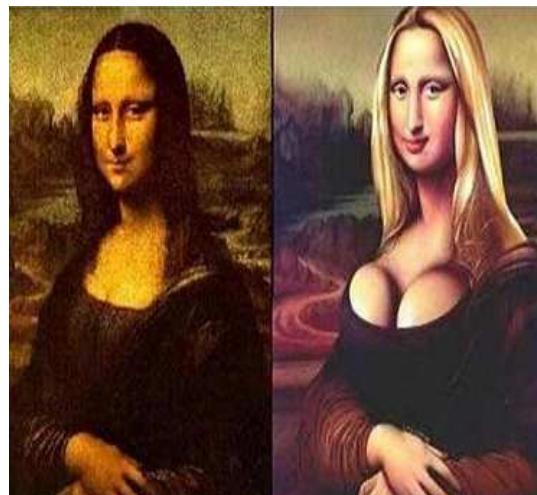
## **ENGENHARIA REVERSA**

- processo de análise de um software, partindo-se inicialmente da implementação para um nível mais alto de abstração

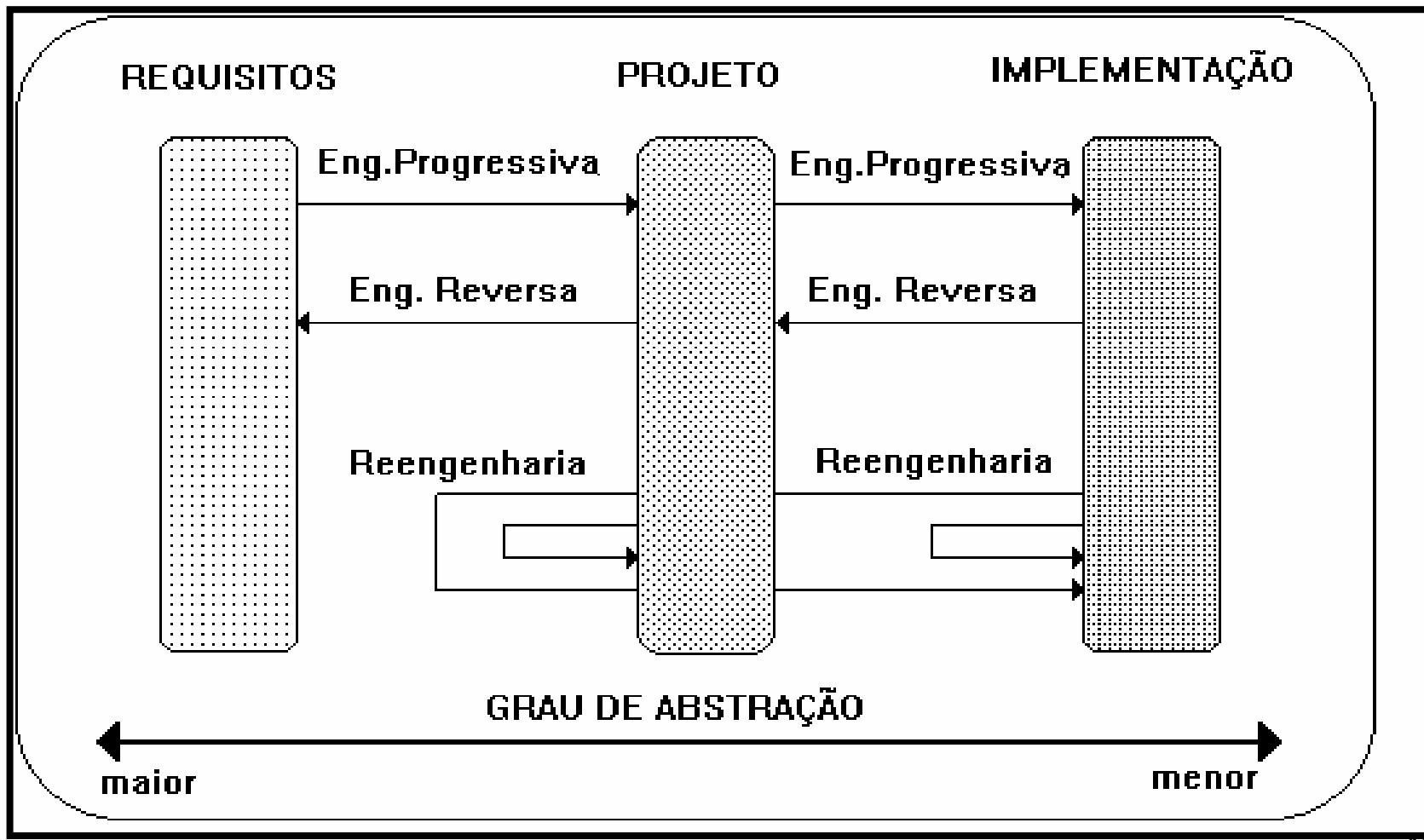
# Reengenharia

## REENGENHARIA

- implica no exame e na alteração do software para reconstruí-lo em uma nova forma.



# Engenharia Reversa e Reengenharia



# Elementos da Engenharia Reversa

## **1) NÍVEL DE ABSTRAÇÃO**

- conforme o nível de abstração aumenta, mais compreensíveis se tornam as informações.

# Elementos da Engenharia Reversa

## **2) COMPLETITUDE DO PROCESSO**

- refere-se ao nível de detalhes que é fornecido em cada nível de abstração.

# Elementos da Engenharia Reversa

## **3) INTERATIVIDADE**

- refere-se ao grau de participação do ser humano no processo de engenharia reversa.
- conforme o nível de abstração aumenta, a interatividade deve aumentar ou a completitude será prejudicada

# Elementos da Engenharia Reversa

## 3) DIRECIONALIDADE

- se a direcionalidade tem sentido único, toda informação extraída a partir do código fonte é usada durante as atividades de manutenção
- se a direcionalidade tem sentido duplo, a informação é usada para "alimentar" uma ferramenta de reengenharia.