

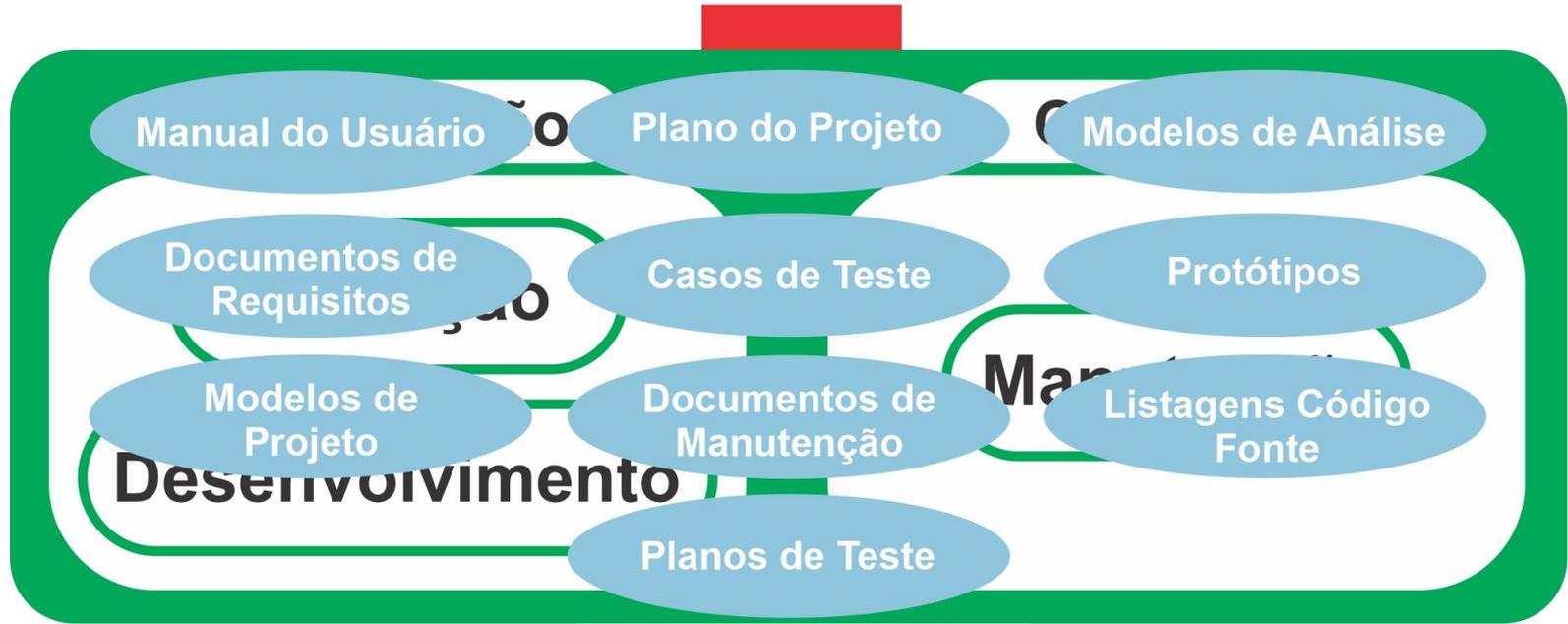
# AULA 3: Gerenciamento de Configuração de Software

---

Professora: **Rosana T. Vaccare Braga**



# Informações do Processo Desenvolvimento de Software



**Informações**



# Problema:

- **Os elementos de informação variam com o tempo:**
  - novos arquivos são incluídos
  - arquivos existentes são alterados ou removidos
- **Como organizar todos estes elementos para controlar em qual estado o sistema se encontrava nos momentos chave do desenvolvimento?**
  - exemplos: no momento da entrega ao cliente, na mudança de uma versão para outra, no momento de auditoria do sistema, etc.

# Gerenciamento de Configuração

- **O que é uma configuração a ser gerenciada?**
  - **Configuração** é o estado do conjunto de itens que formam o sistema em um determinado momento
  - **Item de configuração** é qualquer um dos itens desse conjunto

# Gerenciamento de Configuração

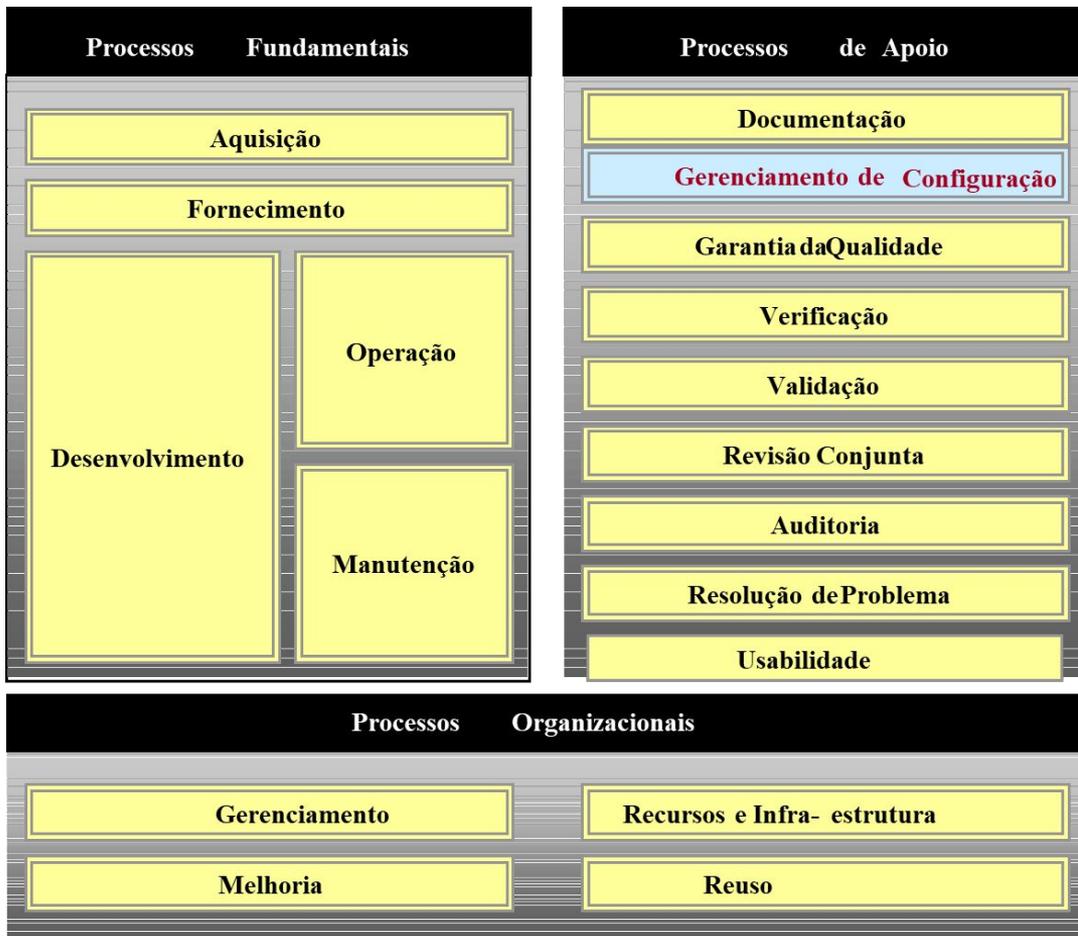
É uma disciplina da Engenharia de Software que busca **identificar** e **controlar o acesso**, as **versões** e as **mudanças** nos **itens de configuração** com o objetivo de **garantir sua integridade**

# Gerenciamento de Configuração de Software

- **Visa controlar e gerenciar as diferentes versões dos componentes de um produto**
  - **Mudanças (evolução) durante o desenvolvimento e após a entrega**
- **Arte de **coordenar** o desenvolvimento de software para minimizar a confusão.**

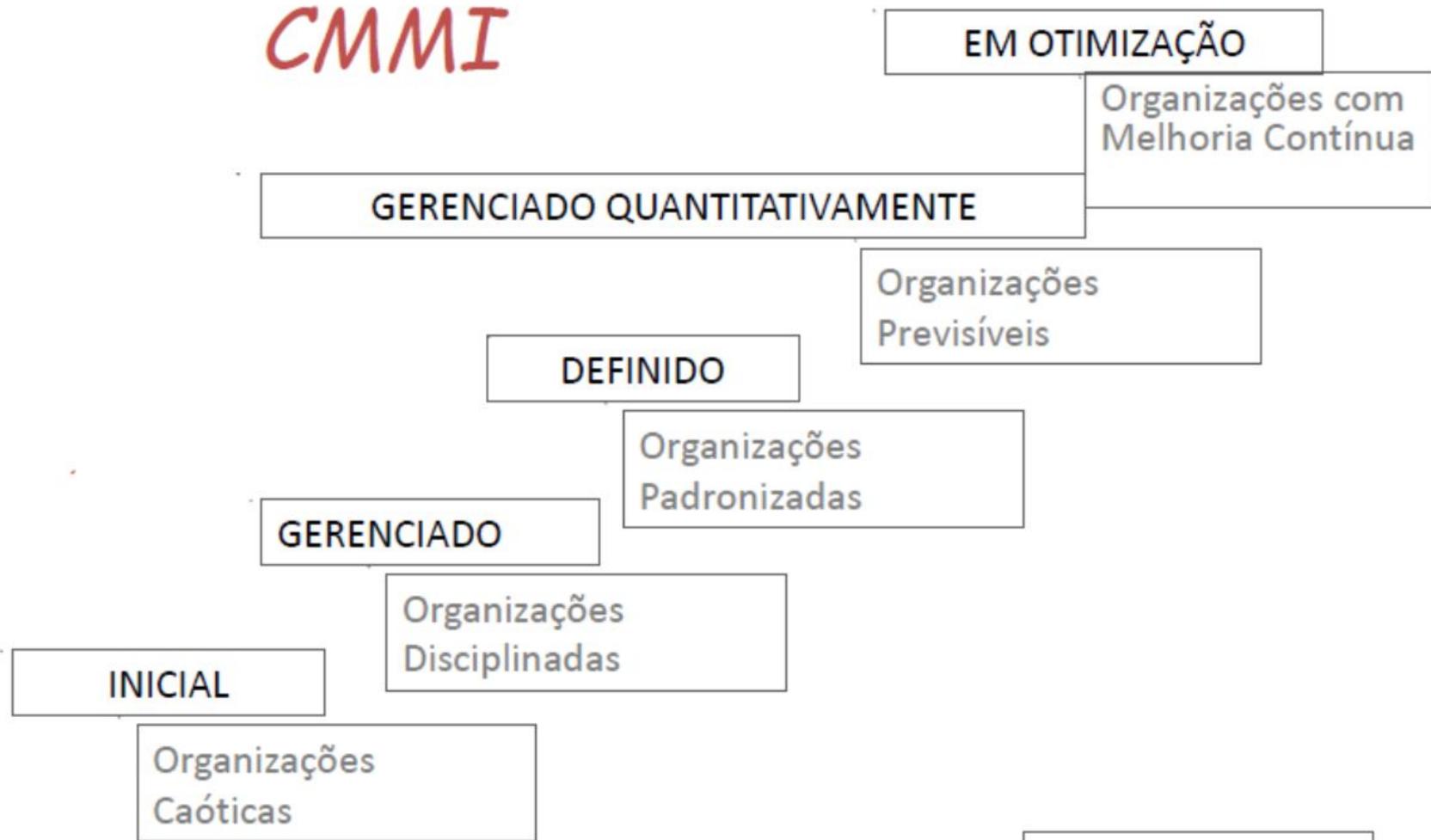
# Gerenciamento de Configuração de Software

- **É um dos processos fundamentais para que uma empresa desenvolvedora de software consiga certificações como ISO/IEC 12207**
- **Primeiro nível de capacidade em modelos como CMMI e MPS-Br**



**Norma ISO/IEC 12207**

# CMMI



GCS no CMMI

GERENCIADO

Organizações  
Disciplinadas

- 7- Gerenciamento de Configuração
- 6- Garantia de Qualidade de Processo e de Produto
- 5- Medição e Análise
- 4- Gestão de subcontratação
- 3- Monitoramento e Controle de Projeto
- 2- Planejamento de Projeto
- 1- Gerenciamento de Requisitos

INICIAL

Organizações  
Caóticas

Atividades necessárias  
para sair do nível inicial  
e passar para o nível  
gerenciado.

Modelo de Referência (MR mps)  
Níveis de Maturidade

Nível A – EM OTIMIZAÇÃO

Nível B – GERENCIADO  
QUANTITATIVAMENTE

Nível C – DEFINIDO

Nível D – LARGAMENTE  
DEFINIDO

Nível E – PARCIALMENTE  
DEFINIDO

Nível F - GERENCIADO

Nível G – PARCIALMENTE  
GERENCIADO

1. Inovação e Implantação na Organização
2. Análise e Resolução de Causas
3. Desempenho do Processo Organizacional
4. Gerência Quantitativa do Projeto
5. Análise de Decisão e Resolução
6. Gerência de Riscos
7. Desenvolvimento de Requisitos
8. Solução Técnica
9. Integração do Produto
10. Instalação do Produto
11. Liberação do Produto
12. Verificação
13. Validação
14. Treinamento
15. Avaliação e Melhoria Organizacional
16. Definição do Projeto
17. Adaptação do Processo de Gerência de Projeto
18. Medição
19. Gerência de Configuração
20. Aquisição
21. Garantia da Qualidade
22. Gerencia de requisitos
23. Gerencia de Projeto

PROCESSOS

GCS no MPS-Br

# Principais Tarefas do Gerenciamento de Configuração

**O gerenciamento de configuração de um produto de sistema de software envolve quatro atividades estreitamente relacionadas:**

**1. Controle de versão:** envolve o acompanhamento das várias versões dos componentes do sistema e a garantia de que as alterações feitas nos componentes por diferentes desenvolvedores não interfiram umas nas outras.

**2. Construção do sistema:** Este é o processo de montagem de componentes do programa, dados e bibliotecas, compilando-os e vinculando-os para criar um sistema executável.

# Principais Tarefas do Gerenciamento de Configuração

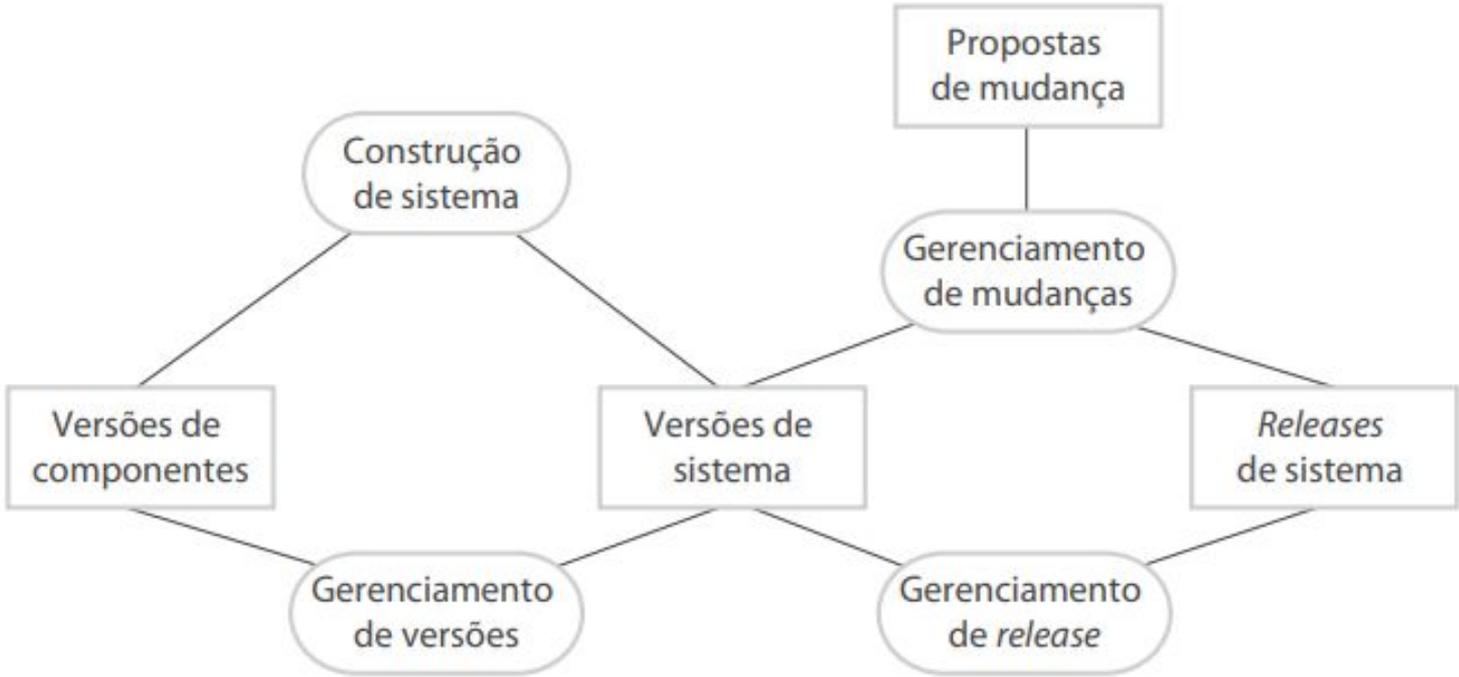
**3. Gerenciamento de alterações:** envolve acompanhar as solicitações de alterações de clientes e desenvolvedores no software entregue, calcular os custos e o impacto de fazer essas alterações e decidir se e quando as alterações devem ser implementadas.

**4. Gerenciamento de liberação:** envolve preparar o software para liberação externa e acompanhar as versões do sistema que foram lançadas para uso do cliente.

# Principais Tarefas do Gerenciamento de Configuração

**Figura 25.1**

Atividades de gerenciamento de configuração



# Tarefas Preliminares

- **Selecionar os itens a serem gerenciados.**
  - a. **É importante que seja efetuada uma seleção dos itens relevantes.**
  - b. **Uma **super documentação** torna o gerenciamento de configuração muito caro.**
    - i. **Itens mais **usados** no ciclo de vida.**
    - ii. **Itens mais **genéricos**.**
    - iii. **Itens mais **importantes** para a segurança.**
    - iv. **Itens projetados para **reuso**.**
    - v. **Itens que podem ser **modificados** por vários desenvolvedores ao mesmo tempo.**

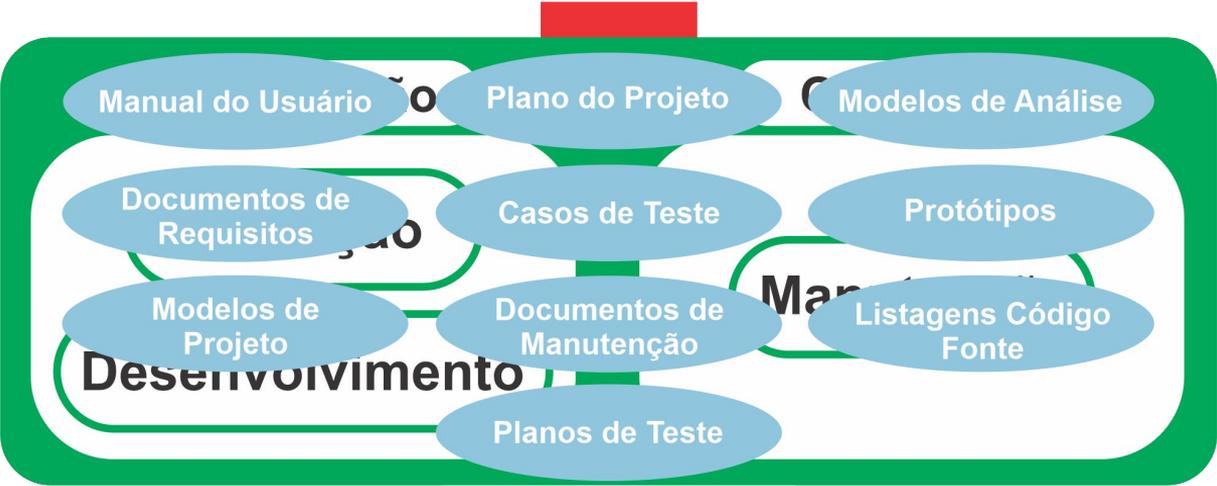
# Item de Configuração

- **Elemento unitário ou um grupo de elementos para efeito de controle de versão. Exemplos:**
  - Código
  - Documentação
  - Diagramas, planos, ferramentas, casos de teste

# Item de Configuração

- **Todos os documentos que podem ser úteis para a evolução futura do sistema**
- **Necessário determinar a granularidade**
- **O processo de desenvolvimento adotado influencia em como os itens de configuração serão definidos.**

# Item de Configuração



Informações

**Alguns itens de informação são selecionados**

# Conceitos Básicos

## I. Relacionamento entre os itens de configuração

- **Dependência**
- **Agregação**
- **Realização**
- **Especialização**

# Conceitos Básicos

## II . Rastreabilidade

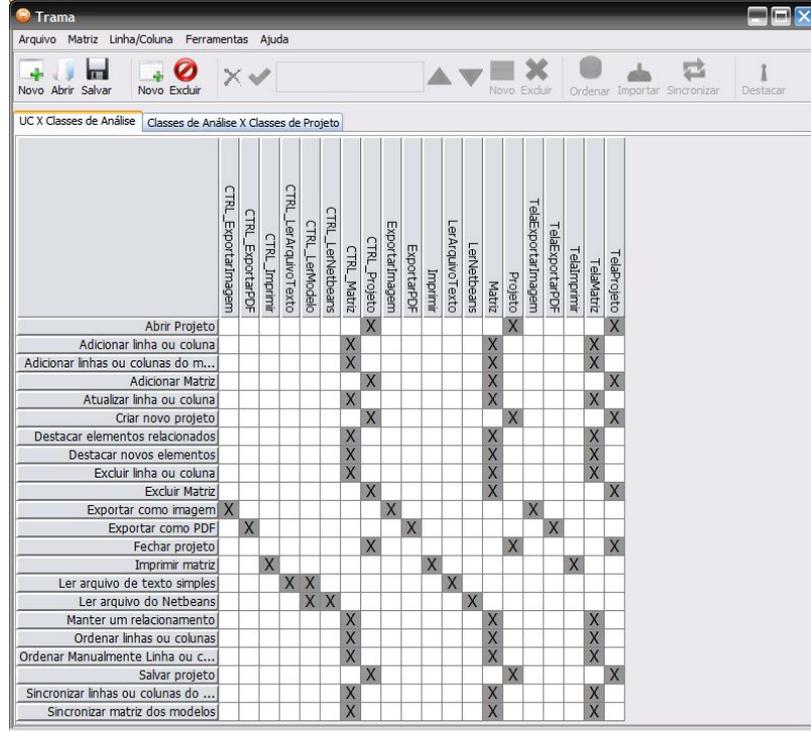
- Manter a consistência nas mudanças feitas

www . Fabio Cruz . com **Matriz de rastreabilidade de requisitos - versão 1.1**

	Funcionalidade 1	Funcionalidade 2	Funcionalidade 3	Funcionalidade 4		
Requisito 1	x					
Requisito 2		x	x			
Requisito 3				x		
Requisito 4				x		

**Projeto A**

Requisitos Funcionais	Casos de Uso			
	UC-1	UC-2	UC-3	UC-4
FR-1	↙			
FR-2	↙			
FR-3			↙	
FR-4			↙	
FR-5		↙		↙
FR-6			↙	



# Conceitos Básicos

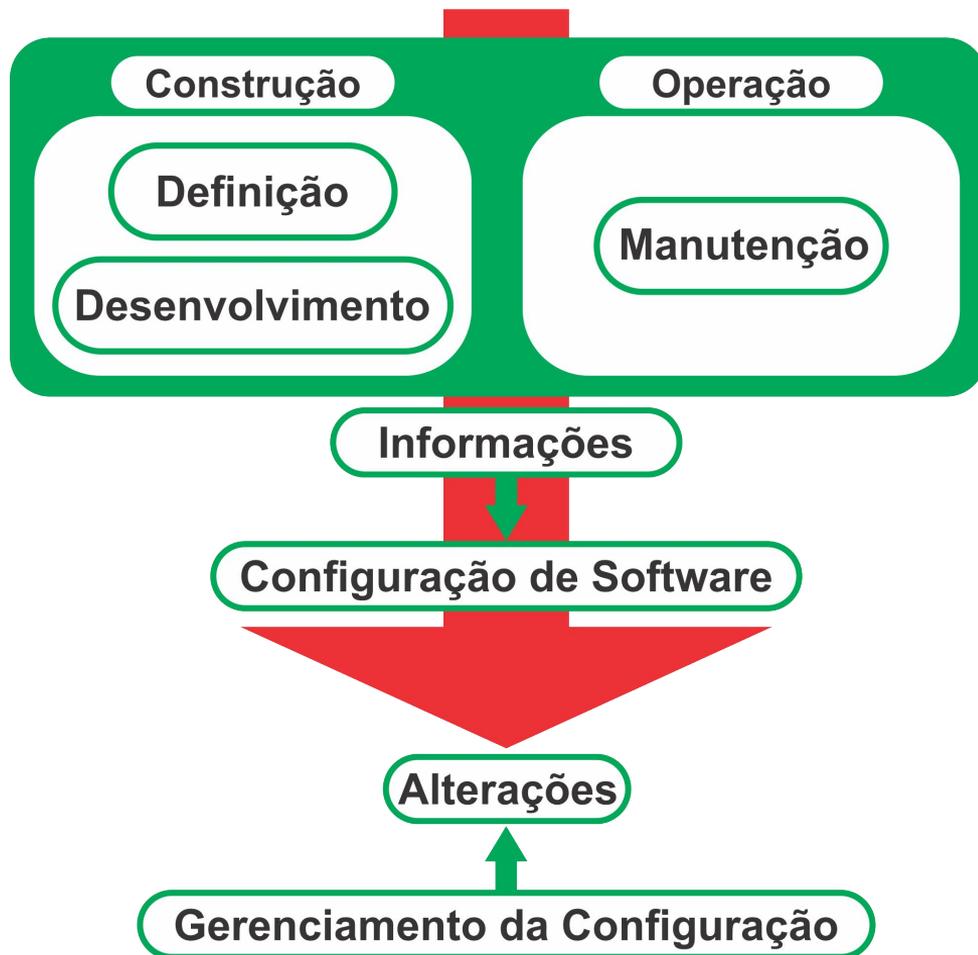
## III. Versões de Itens de Configuração

- a) revisão (substitui um item anterior)
- b) variante (cria um novo)

# Conceitos Básicos

## IV. Configuração de Software

Um conjunto de itens de configuração de software inter-relacionados compõem uma **configuração de software**.



# Controle de Mudanças

- **Garantem que as mudanças ocorram de maneira controlada e otimizada.**
- **Assim, deve ser instituído na organização um processo que combine...**
  - a. **procedimentos humanos e**
  - b. **ferramentas automatizadas...**
- **... para proporcionar um mecanismo de controle das mudanças.**

# Controle de Mudanças

- **Manter histórico de mudança**
- **Justificar mudança**

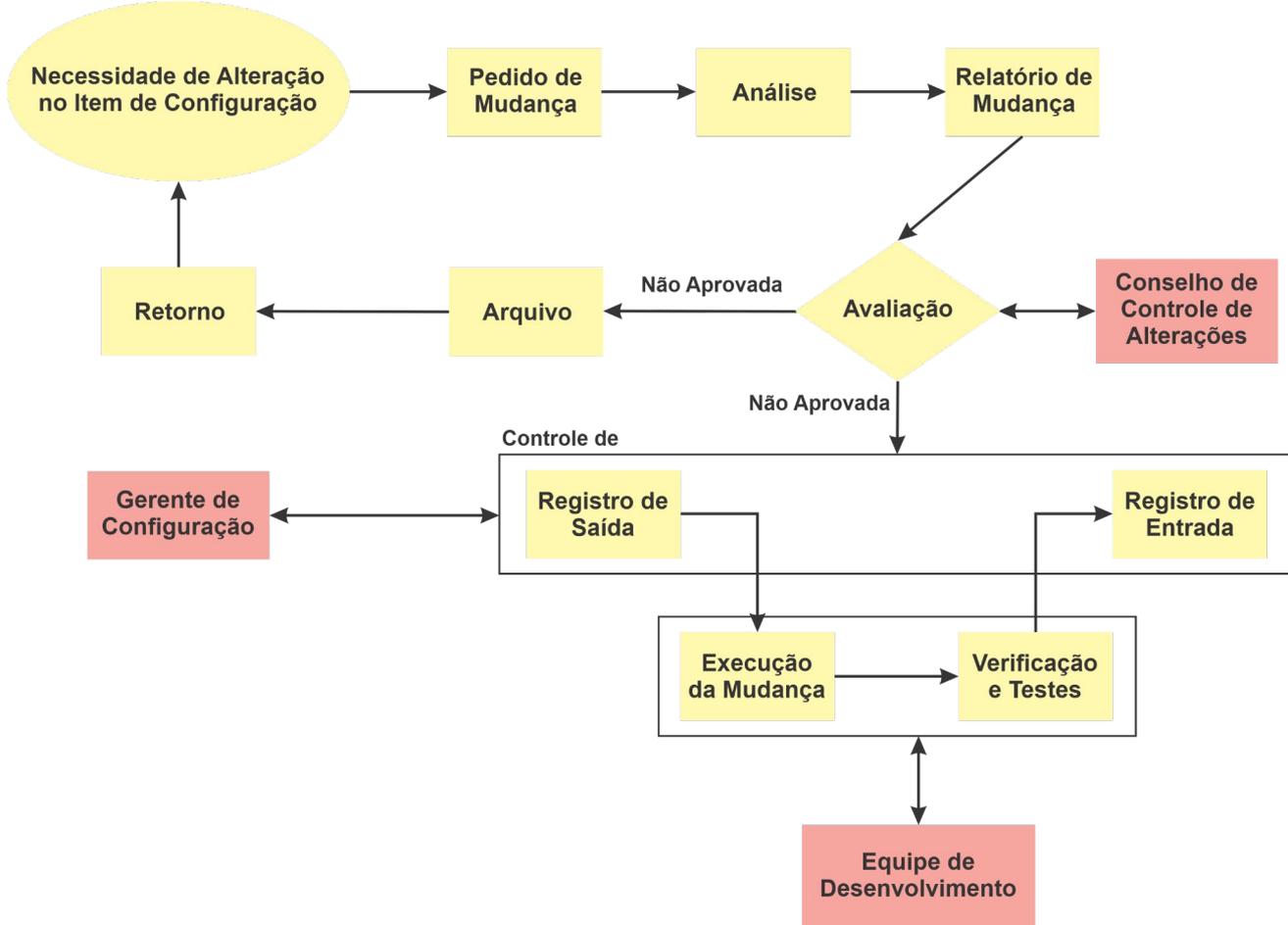
**Exemplo:**

- a) Mudanças da versão 2.2 para a versão 2.3:**
- **correção do defeito D345**
  - **correção do defeito D346**
  - **adicionada a funcionalidade do RF44**
- b) Pendências para uma versão posterior:**
- **melhorar usabilidade da interface I43**

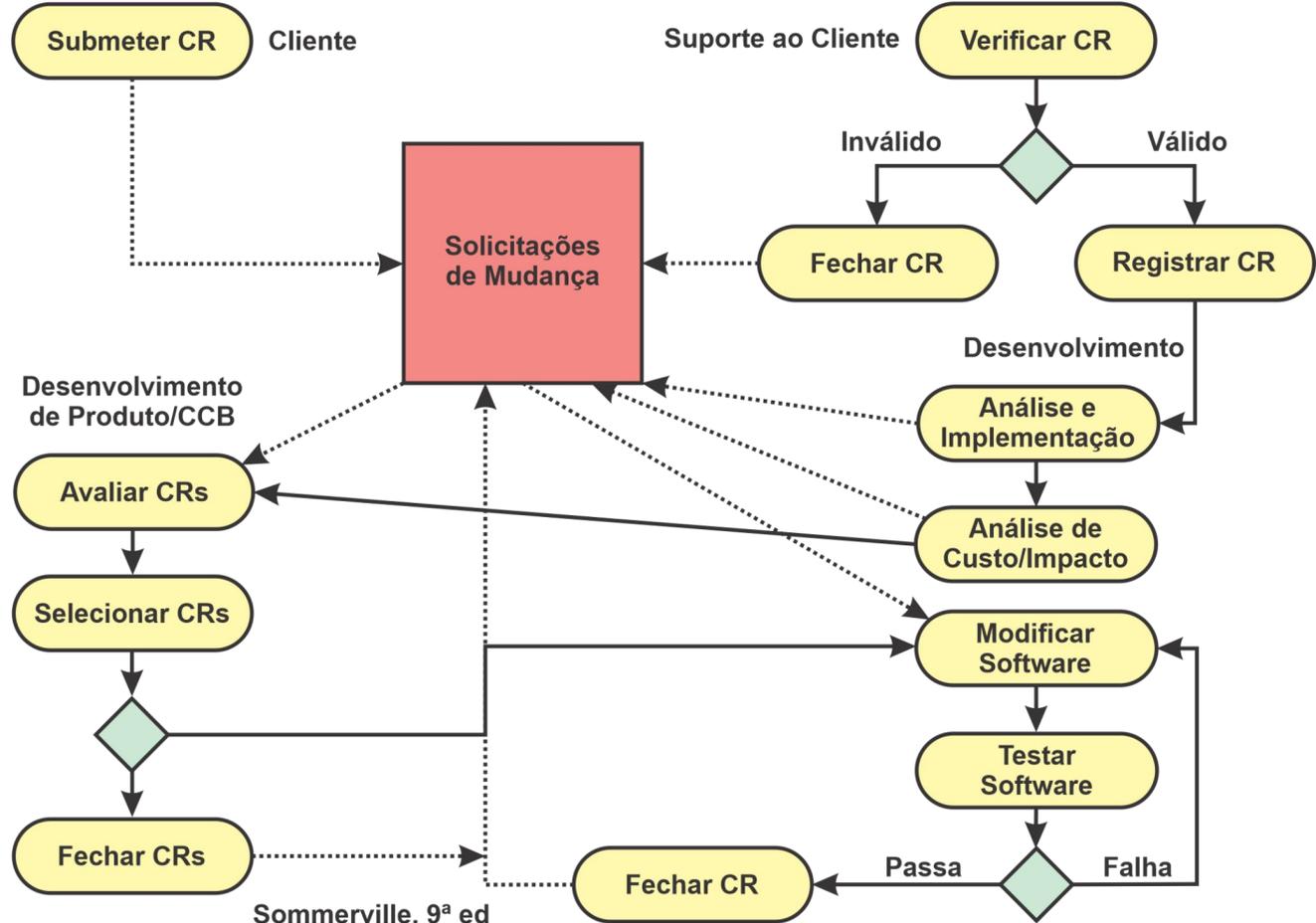
# Controle de Mudanças

- **Como realizar a mudança ?**
  - **Solicitação de Mudanças (Change Request)**
  - **Análise de Impacto**
- **Importante que as mudanças sejam gerenciadas, evitando o caos no sistema**

# Controle de Mudanças - Exemplo de um processo



# Outro exemplo de Processo para Controle de Mudanças



## Formulário de solicitação de mudança

---

**Projeto:** SICSA/AppProcessing

**Solicitante de mudança:** I. Sommerville

**Mudança solicitada:** O *status* dos requerentes (rejeitados, aceitos etc.) deve ser mostrado visualmente na lista de candidatos exibida.

**Número:** 23/02

**Data:** 20/jan./2009

**Analista de mudança:** R. Loeek

**Componentes afetados:** ApplicantListDisplay, StatusUpdater

**Data da análise:** 25/jan./2009

**Componentes associados:** StudentDatabase

**Avaliação de mudança:** Relativamente simples de implementar, alterando a cor de exibição de acordo com *status*. Uma tabela deve ser adicionada para relacionar *status* a cores. Não é requerida alteração nos componentes associados.

**Prioridade de mudança:** Média

**Implementação de mudança:**

**Esforço estimado:** 2 horas

**Data para equipe de aplicação de SGA:** 28/jan./2009

**Decisão:** Aceitar alterar. Mudança deve ser implementada no *Release* 1.2

**Implementador de mudança:**

**Data de submissão ao QA:**

**Data de submissão ao CM:**

**Comentários:**

**Data de decisão do CCB:** 30/jan./2009

**Data de mudança:**

**Decisão de QA:**

# Auditoria de Configuração

- **Verifica a Solicitação de Mudança x Liberação de IC**
- **Verifica se a Matriz de Rastreabilidade foi criada**
- **Verifica a localização e acessos aos IC**
- **Verifica a presença dos itens de configuração na Baseline**

## O Gerente de Configuração



- **Manter banco de dados de configuração atualizado**
- **Garantir o acesso aos itens de configuração**
- **Auditoria de Configuração**
- **Garantir a rastreabilidade dos IC**
- **Plano de Gerência de Configuração**

# Plano de Gerência de Configuração

- **Papéis e responsabilidades**
- **Identificação dos itens de configuração**
- **Banco de dados de configuração**
- **Planejamento de Baselines**
  - **Criação**
  - **Conteúdo**
- **Auditorias**
- **Padronização das nomenclaturas**
- **Releases e Integração Contínua**

# Identificação dos Itens de Configuração

- Criar um **esquema de identificação**.
  - a. Atribuir nomes únicos a cada um dos componentes.
  - b. Pelo nome deve ser possível reconhecer:
    - i. A **evolução** de cada uma das versões dos componentes.
    - ii. A **hierarquia** existente entre componentes.

# Controle de Versões

Este tópico é tema da próxima aula!

- Um item, ao ser desenvolvido, evolui até que atinja um estado em que atenda aos propósitos para o qual foi criado.
  - Isso implica em diversas alterações, gerando uma **versão** do item a cada estado.
- Processo de **identificar** e **acompanhar** o desenvolvimento de **diferentes versões** de um sistema.

# Preparação de Relatórios de Estado

- **Relatar a todas as pessoas envolvidas no desenvolvimento e na manutenção do software informações sobre as alterações realizadas.**
  - **O que aconteceu?**
  - **Quem fez?**
  - **Quando aconteceu?**
  - **O que mais será afetado?**
- **Melhora a comunicação.**

# Ferramentas

- **Controle de Versão**
  - **SVN, CVS, GIT, ClearCase**
- **Controle de Mudança**
  - **Redmine, Trac, ClearQuest**

# Ferramentas

- **Controle de Mudança**

- **Redmine:** é uma ferramenta de gerenciamento de projetos de software de código aberto com recursos de controle de mudança. Ele permite que os desenvolvedores gerenciem o fluxo de trabalho, incluindo o rastreamento de problemas, atribuição de tarefas e controle de versão. O Redmine também fornece recursos de gerenciamento de mudanças, como fluxos de trabalho personalizáveis e acompanhamento de alterações em tarefas e problemas.

# Ferramentas

- **Controle de Mudança**

- **Trac:** é uma ferramenta de gerenciamento de projetos de software de código aberto que integra com sistemas de controle de versão como o Subversion e o Git. Ele permite que os desenvolvedores gerenciem o fluxo de trabalho de desenvolvimento de software, incluindo o rastreamento de problemas, tarefas e controle de versão, além de fornecer recursos de gerenciamento de mudanças, relatórios e monitoramento de progresso do projeto.

# Ferramentas

- **Controle de Mudança**

- **ClearQuest:** é uma ferramenta comercial da IBM que permite que equipes gerenciem processos de mudança em projetos de software, rastreiem problemas, atribuam tarefas e registrem todas as mudanças no código-fonte. É personalizável, oferecendo fluxos de trabalho e campos personalizados, e fornece recursos de geração de relatórios e gráficos para monitorar o progresso das mudanças ao longo do tempo.

# Ferramentas

## Outras ferramentas para controle de mudança

- **Assembla:** plataforma de gerenciamento de projetos que inclui recursos de rastreamento de bugs, colaboração em equipe e controle de mudanças;
- **Basecamp:** ferramenta de gerenciamento de projetos com recursos de colaboração em equipe, rastreamento de tarefas e gerenciamento de projetos;
- **VersionOne:** plataforma de gerenciamento de Agile que inclui recursos de rastreamento de histórias de usuários, gerenciamento de projetos e controle de mudanças.

# Próxima aula: Controle de Versões

Se possível, leiam com antecedência:

**Seção 10.2 do livro Engenharia de Software Moderna**  
**<https://engsoftmoderna.info/cap10.html>**