

Gerência da Configuração

Itens de Configuração

Prof. Márcia Pantoja.

marcia.pantoja@unama.br

Gerenciamento da Configuração de Software

O Gerenciamento de Configuração do Software é um importante elemento da **garantia da qualidade de software.**

Tarefas de Gerenciamento de Configuração

- 1- Tarefas Preliminares
- 2- Identificação
- 3- Controle de Mudanças
- 4- Controle de Versão
- 5- Auditoria de Configuração
- 6- Relato de Situação
- 7- Controle de Interface
- 8- Controle de Subcontratados e Fornecedores

TAREFAS PRELIMINARES

Tarefas Preliminares

1.1: Selecionar os itens a serem gerenciados.

- É importante que seja efetuada uma seleção dos itens relevantes, porque uma superdocumentação torna o gerenciamento de configuração muito oneroso.
- Geralmente, devem sofrer gerenciamento de configuração:
 - os itens mais usados no ciclo de vida,
 - os itens mais genéricos
 - os itens mais importantes para a segurança
 - os itens projetados para reúso
 - os itens que podem ser modificados por vários desenvolvedores ao mesmo tempo.

Tarefas Preliminares

1.1: Selecionar os itens a serem gerenciados (cont).

- Somente os itens selecionados serão controlados, sendo que os outros itens poderão ser alterados livremente.
- Exemplo de itens sugeridos por Pressman

Tarefas Preliminares

EXEMPLO DE ITENS DE CONFIGURAÇÃO

1. Especificação do Sistema
2. Plano de Projeto de Software
3. Especificação de Requisitos do Software
4. Manual Preliminar do Usuário
5. Especificação do Projeto
 - a) Descrição do Projeto de Dados
 - b) Descrição do Projeto Arquitetural
 - c) Descrições do Projeto Modular
 - d) Descrições do Projeto de Interface
 - e) Descrições de Objetos (se forem usadas técnicas orientadas a objetos)
6. Listagem do código-fonte
7. Planos, Procedimentos, Casos de Testes e Resultados Registrados
8. Manuais Operacionais e de Instalação
9. Programa Executável e Módulos Interligados
10. Descrição do Banco de Dados
 - a) Esquema e estrutura de arquivo
 - b) Conteúdo inicial
11. Manual do Usuário
12. Documentos de Manutenção
 - a) Relatórios de problemas de software
 - b) Solicitações de manutenção
 - c) Pedidos de mudança
13. Padrões e procedimentos para engenharia de software
14. Ferramentas de produção de software (editores, compiladores, CASE, etc.)

Tarefas Preliminares

- **1.2:** Descrever como os itens selecionados relacionam-se
 - A identificação desses relacionamentos é muito importante para a manutenção, pois permite que se localize rapidamente os itens afetados por cada alteração.
 - Consideram-se cinco classes de relacionamento:

CLASSES DE RELACIONAMENTO

- Equivalência
 - Ex: BD em disco e em fita magnética
- Dependência
 - Ex: a descrição do projeto modular é dependente da especificação do projeto
- Derivação
 - Ex: código objeto é derivado do código fonte
- Sucessão
 - Ex: a versão 1.2 é sucessora da versão 1.1
- Variante
 - Ex: versão para DOS ou para UNIX

- **1.3** Planejar as linhas de referência dentro do ciclo de vida do projeto.
 - Geralmente, cria-se uma linha de referência ao final de cada fase do ciclo de vida do projeto e, periodicamente, depois de cada manutenção.
 - Deve-se especificar quais itens serão revisados e armazenados em cada uma das linhas de referência planejadas.

Tarefas Preliminares

- **1.4-** Descrever a maneira como os itens serão arquivados e recuperados do repositório.

IDENTIFICAÇÃO

Identificação

Criar um esquema de identificação

- atribuir nomes únicos a cada um dos componentes
- através do nome deve ser possível reconhecer
 - a evolução de cada uma das versões dos componentes
 - a hierarquia existente entre componentes

Identificação

- Exemplo simples de um esquema de identificação para um projeto cuja sigla é “AA”
- O esquema de identificação utiliza a combinação de:
 - nome do projeto
 - tipo de item
 - nome do item
 - versão do item

Identificação

Exemplo de um esquema de identificação dos itens de configuração

Item	Projeto	Tipo	Nome	Versão	Nome completo	
Especificação do Sistema	AA	ES		1.1	AAES	vl.1
Plano de Projeto	AA	PP		1.1	AAPP	vl.1
Especificação de Requisitos do Software	AA	ER		1.1	AAER	vl.1
Especificação de Projeto	AA	EP		1.1	AAEP	vl.1
Programa Fonte	AA	PF	Prin	1.1	AAPFPrin	vl.1
Programa Fonte (sub-rotinas)	AA	PF	Rot	1.1	AAPFRot	vl.1
Plano e Casos de Testes	AA	TT		1.1	AATT	vl.1
Nova versão das sub-rotinas	AA	PF	Rot	1.2	AAPFRot	vl.2

CONTROLE DE MUDANÇA

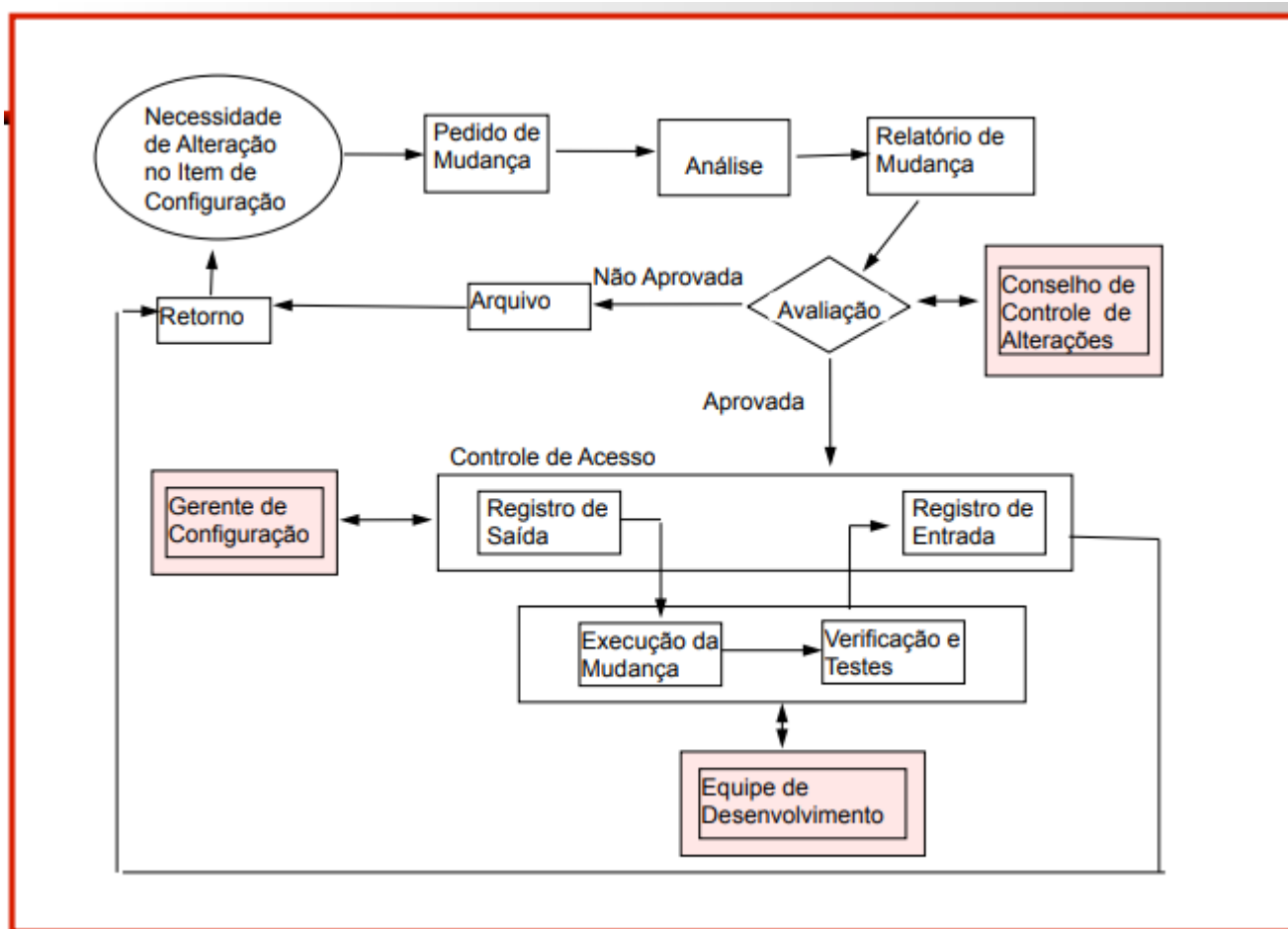
Controle de Mudança

- Durante o processo de desenvolvimento de software, mudanças descontroladas podem levar rapidamente ao caos.
- Assim, deve ser instituído na organização um processo que combine procedimentos humanos e ferramentas automatizadas para proporcionar um mecanismo de controle das mudanças.

Controle de Mudança

- O processo de controle de mudanças deve ser implementado depois que uma linha de referência for fixada - antes disso, somente um controle de mudanças informal precisa ser aplicado
- A seguir, um exemplo para ilustrar um processo de controle de mudanças que pode ser implementado para os itens que já passaram por uma linha de referência.

Processo de Controle de Mudança



Controle de Mudança

- Os procedimentos de controle das mudanças:
 - asseguram que as mudanças em um software sejam feitas de modo controlado, permitindo-se prever o efeito das mesmas em todo o sistema.

Controle de Mudança

- Procedimentos formais de organização e de controle das mudanças no sistema permitem que:
 1. Os pedidos de alteração possam ser considerados em conjunto com outros pedidos.
 2. Os pedidos similares possam ser agrupados.
 3. Os pedidos incompatíveis entre si ou com os objetivos do sistema possam ser identificados.
 4. Possam ser atribuídas prioridades aos pedidos e, de acordo com essas prioridades, possam ser gerados cronogramas.

CONTROLE DE VERSÃO

Controle de Versão

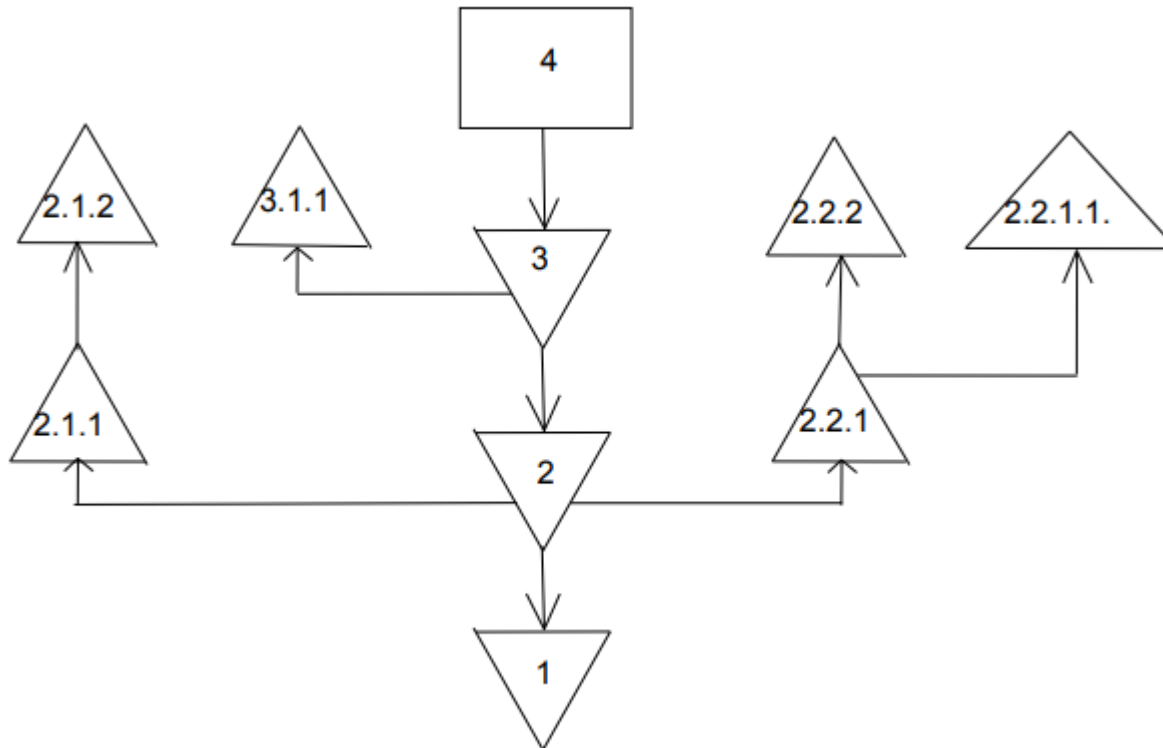
- Um item, ao ser desenvolvido, evolui até que atinja um estado em que atenda aos propósitos para o qual foi criado.
- Isso implica em diversas alterações, gerando uma versão do item a cada estado.
- Para estabelecer o controle sobre as diversas versões, todas as versões devem ser armazenadas e identificadas.
- Isso, geralmente, é feito com o auxílio de uma ferramenta.
- A versão do item pode ser incluída no esquema de identificação ou ser acessível a partir de uma tabela à parte.

Controle de Versão

- É conveniente que o esquema de identificação das versões dos itens seja feito em forma de árvore, pois ao mesmo tempo em que mantém um histórico das versões dos itens, permite identificação única e ramificações a partir de qualquer versão

Controle de Versão

- Árvore de versões de um item de configuração usando delta negativo.



AUDITORIA DE CONFIGURAÇÃO

Auditoria de Configuração

- A auditoria da configuração compreende atividades para garantir que as alterações na configuração de software foram efetuadas apropriadamente.
- A identificação e controle das alterações ajudam a manter ordem mas, para assegurar que a alteração foi implementada apropriadamente, há necessidade de auditorias.

Auditoria de Configuração

- Existem dois tipos de auditoria de configuração de software que são obrigatórias nas linhas de referência do ciclo de desenvolvimento e manutenção de software:
 - Auditoria Funcional
 - Auditoria Física

AUDITORIA FUNCIONAL

- A **auditoria funcional** preocupa-se com aspectos internos dos arquivos, compreendendo uma verificação técnica formal nos itens de configuração.
- Essa verificação é uma atividade de controle de qualidade que tenta descobrir omissões ou erros na configuração, que degradam os padrões de construção do software.

Auditoria de Configuração

AUDITORIA FÍSICA

- A **auditoria física** complementa a auditoria funcional, determinando características não consideradas durante a revisão
- Algumas questões podem ser colocadas:

Questões da Auditoria Física

- *As alterações especificadas na Ordem de Alteração foram efetuadas? Alguma modificação adicional foi incorporada?*
- *Foi efetuada a revisão técnica formal ?*
- *Os padrões de engenharia de software foram seguidos ?*
- *A data e o autor da alteração foram especificados? Os atributos do item da configuração refletem a alteração?*
- *Foram seguidos os procedimentos de administração de configuração?*
- *Todos os itens de configuração relacionados foram atualizados apropriadamente?*

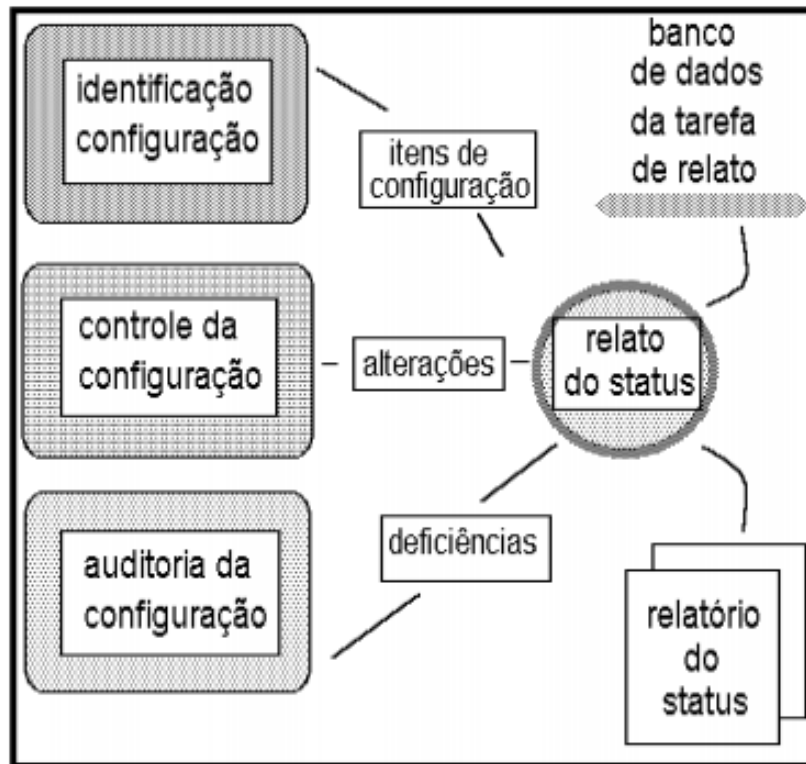
RELATO DA SITUAÇÃO

Relato da Situação

- O objetivo do Relato da Situação é relatar a todas as pessoas envolvidas no desenvolvimento e na manutenção do software. As seguintes informações sobre as alterações na configuração de software:
 - O que aconteceu?
 - Quem o fez?
 - Quando aconteceu?
 - O que mais será afetado?

Relato da Situação

FLUXO DE INFORMAÇÃO PARA O RELATO DO STATUS



CONTROLE DE INTERFACE

Controle de Interface

- As atividades de controle de interface (ou fronteira) coordenam as mudanças nos itens de configuração que são afetados por itens que não estejam sendo controlados.
- Equipamentos, sistemas de software, software de suporte, assim como outros projetos devem ser examinados na busca por possíveis interfaces com o projeto sob controle.

Controle de Interface

- Para cada interface deve-se descrever:
 - a) O tipo de interface
 - b) As unidades organizacionais afetadas
 - c) Como será feito o controle sobre a interface
 - d) Como os documentos de controle da interface serão aprovados.

Ferramentas de GCS

- Ferramentas de software podem auxiliar as atividades de gerenciamento de configuração de software.
- Exemplos de ferramentas:
 - CVS (Concurrent Versions System)
<http://www.cvshome.org/>
 - Subversion <http://subversion.apache.org>

Dúvidas??

