



UNIDADE 11

Gerenciamento de Configuração (Cap. 25)

Tópicos abordados



Gerenciamento de mudanças



Gerenciamento de versões



Gerenciamento de *releases*

Gerenciamento de configuração

- Porque os softwares mudam frequentemente, os sistemas podem ser pensados como um conjunto de versões, e cada qual precisa ser mantida e gerenciada.
- Versões implementam propostas de mudanças, correções de defeitos, e adaptações de hardware e sistemas operacionais diferentes.
- O gerenciamento de configuração (CM – Configuration Management) se interessa pelas políticas, processos e ferramentas para o gerenciamento de sistemas de software que sofrem mudanças. Você precisa de CM porque é fácil perder a noção de quais mudanças e versões de componentes foram incorporadas em cada versão do sistema.

Atividades de gerenciamento de configuração

- **Gerenciamento de mudanças**

- ✓ Manter o acompanhamento das solicitações de mudanças no software dos clientes e desenvolvedores, definir os custos e o impacto das mudanças, e decidir quais mudanças devem ser implementadas.

- **Gerenciamento de versões**

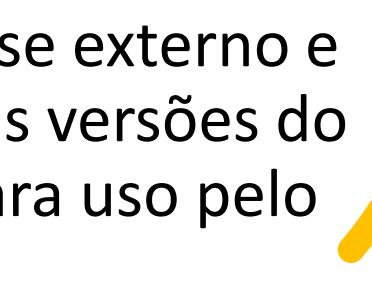
- ✓ Manter o controle das múltiplas versões de componentes do sistema e assegurar que as alterações feitas aos componentes por diferentes desenvolvedores não interfiram umas com as outras.

Atividades de gerenciamento de configuração

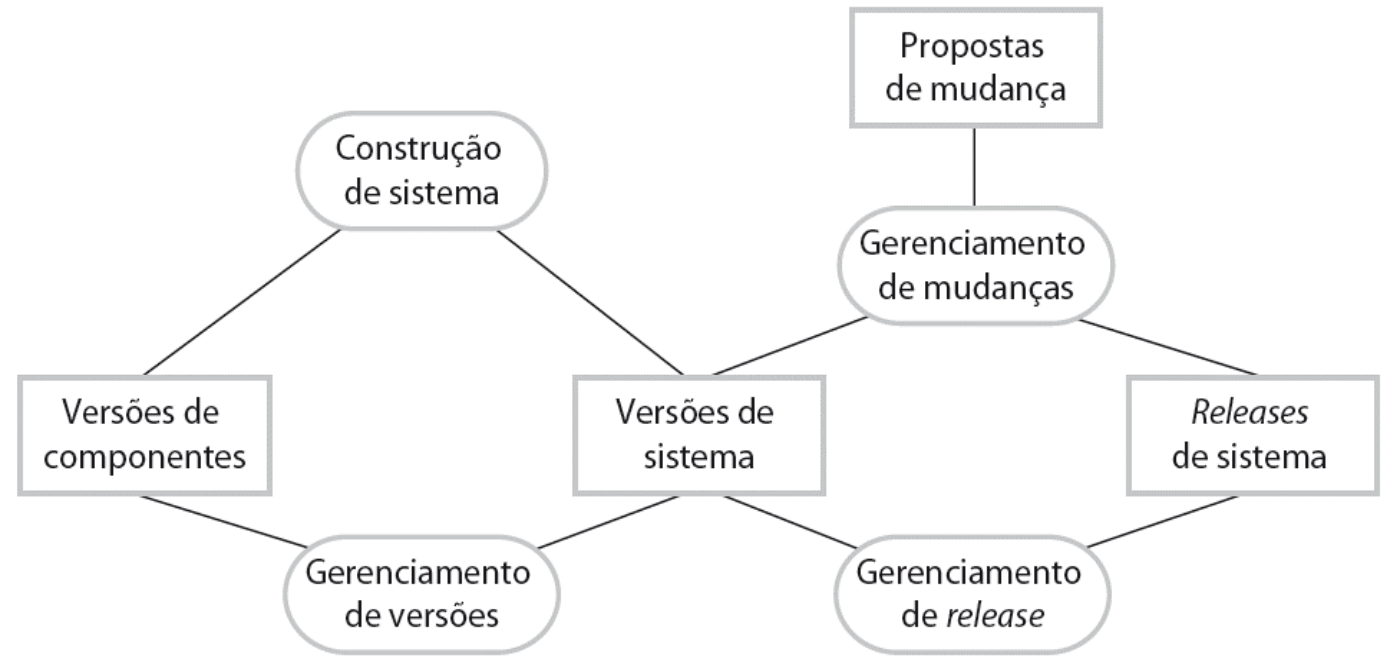
- **Construção do sistema**

- ✓ O processo de montagem dos componentes de programa, dados e bibliotecas, e em seguida, a compilação desses para criar um sistema executável.

- **Gerenciamento de releases**

- ✓ Preparar o software para release externo e manter o acompanhamento das versões do sistema que foram liberadas para uso pelo cliente.
- 

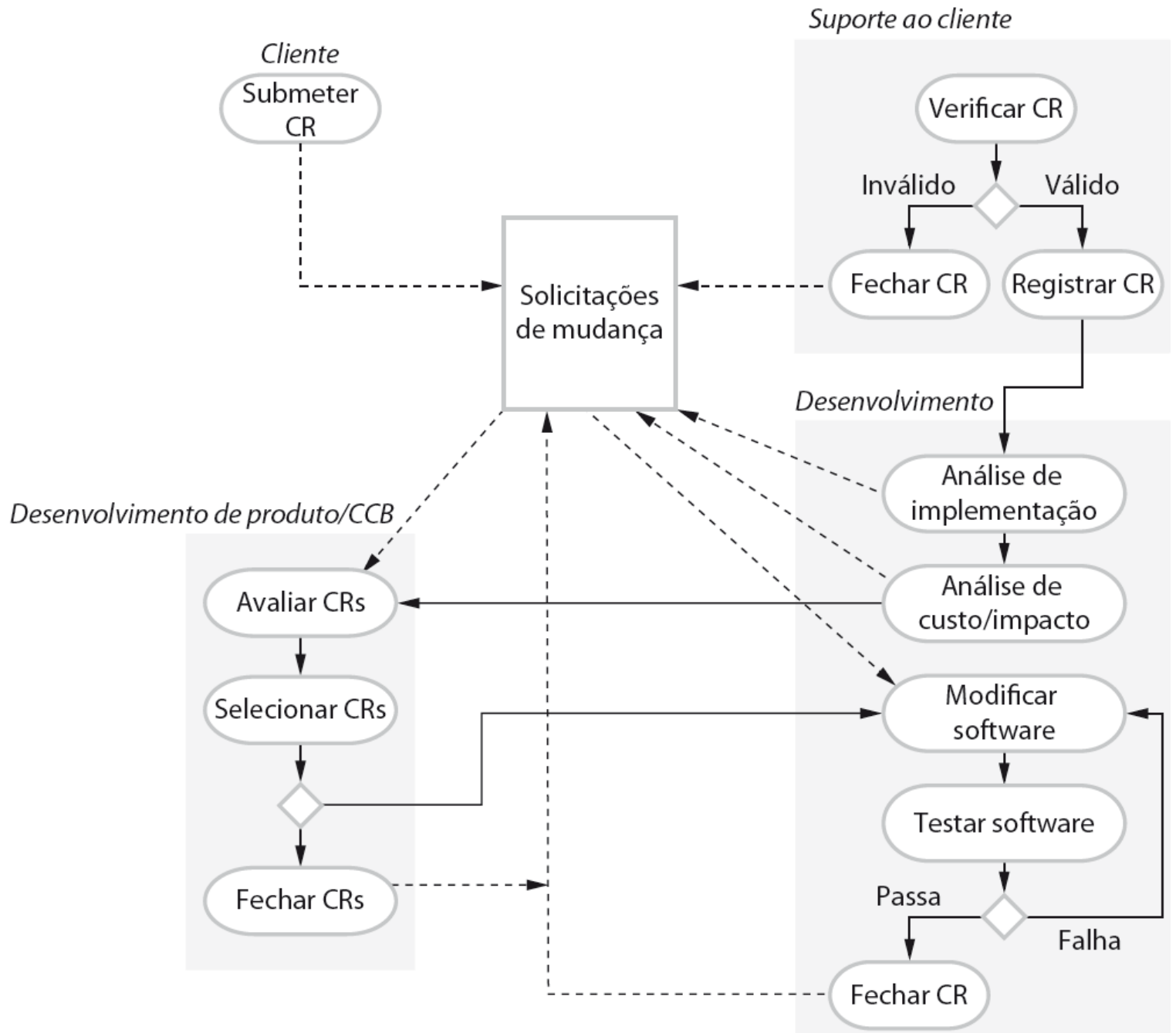
Atividades de gerenciamento de configuração



Gerenciamento de mudanças

- Durante a vida útil de um sistema as necessidades organizacionais e os requisitos desse mudam, bugs precisam ser reparados e os sistemas têm de se adaptar às mudanças em seu ambiente.
- O gerenciamento de mudanças visa garantir que a evolução do sistema seja um processo gerenciado e que seja dada prioridade às mudanças mais urgentes e de custo-benefício.
- O processo de gerenciamento de mudanças está relacionado com a análise dos custos e benefícios das mudanças propostas, a aprovação das mudanças que valem a pena e o acompanhamento das alterações nos componentes do sistema.

O processo de gerenciamento de mudanças



Um formulário de solicitação de mudança

Projeto: SICSA/AppProcessing

Número: 23/02

Solicitante de mudança: I. Sommerville

Data: 20/jan./2009

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

Analista de mudança: R. Looek

Data da análise: 25/jan./2009

Componentes afetados: ApplicantListDisplay, StatusUpdater

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.

Fatores na análise de mudança

- As consequências de não fazer a mudança
- Os benefícios da mudança
- O número de usuários afetados pela mudança
- Os custos de se fazer a mudança
- O ciclo de release de produto

Gerenciamento de mudanças e métodos ágeis

- Em alguns métodos ágeis, os clientes estão diretamente envolvidos no gerenciamento de mudanças.
- Propor uma mudança nos requisitos e trabalhar com a equipe para avaliar o seu impacto e decidir se a mudança deve ter prioridade sobre os recursos planejados para o próximo incremento do sistema.
- As mudanças para melhorar o software são decididas pelos programadores que trabalham no sistema.
- A refatoração, em que o software é melhorado continuamente, não é vista como uma sobrecarga, mas como uma parte necessária do processo de desenvolvimento.

/ SICSA project (XEP 6087)

/ APP-SYSTEM/AUTH/RBAC/USER_ROLE

/ Objeto: currentRole

/ Autor: R. Looek

/ Data de criação: 13/11/2009

/ © Universidade ST. Andrews 2009

/ Histórico de modificações

/ Versão	Modificador	Data	Mudança	Razão
/ 1.0	J. Jones	11/11/2009	Adicionar cabeçalho	Submetido ao CM
/ 1.1	R. Looek	13/11/2009	Novo campo	Solicitação de mudança R07/01

Histórico de derivação

Gerenciamento de versões

- O gerenciamento de versões (VM – *Version Management*) é o processo de manter o controle das diferentes versões dos componentes do software ou itens de configuração e os sistemas em que esses componentes são usados.
- Também envolve assegurar que as mudanças sejam feitas por desenvolvedores diferentes para que essas versões não interfiram umas com as outras.

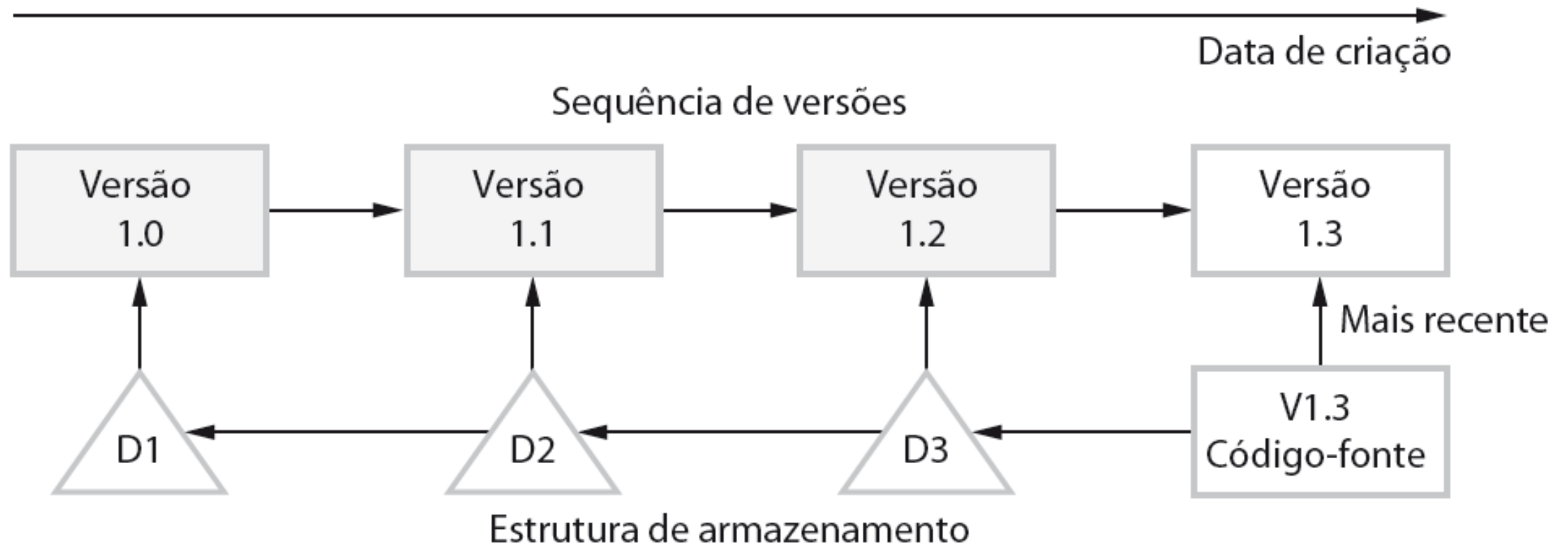
Sistemas de gerenciamento de versões

- **Identificação de versão e release**
 - ✓ Versões gerenciadas recebem identificadores quando são submetidos ao sistema.
- **Gerenciamento de armazenamento**
 - ✓ Para reduzir o espaço de armazenamento exigido por múltiplas versões de componentes que diferem apenas ligeiramente, sistemas de gerenciamento de versões geralmente oferecem recursos de gerenciamento de armazenamento.
- **Registro de histórico de mudanças**
 - ✓ Todas as mudanças feitas no código de um sistema ou componente são registradas e listadas.

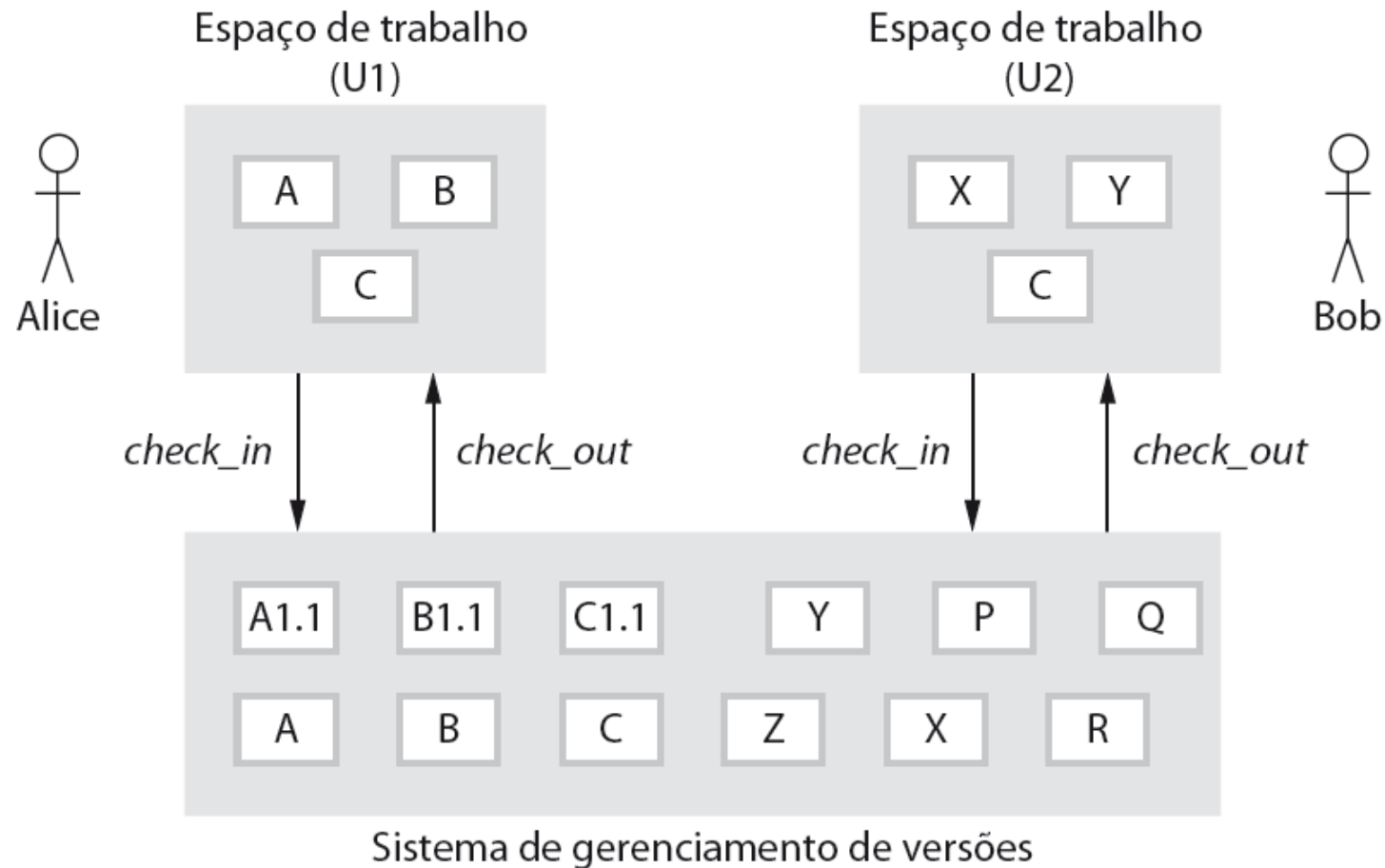
Sistemas de gerenciamento de versões

- **Desenvolvimento independente**
 - ✓ O sistema de gerenciamento de versões mantém o acompanhamento de componentes que foram retirados para edição e garante que as mudanças feitas em um componente por diferentes desenvolvedores não interfiram.
- **Suporte a projetos**
 - ✓ Um sistema de gerenciamento de versões pode apoiar o desenvolvimento de vários projetos que compartilham componentes.

Gerenciamento de armazenamento usando deltas



Check-in e check-out a partir de um repositório versões



Pontos importantes

- Gerenciamento de configuração é o gerenciamento de um sistema de software em evolução. Durante a manutenção de um sistema, uma equipe CM é responsável para garantir que as mudanças são incorporadas ao sistema de uma forma controlada e que os registros são mantidos com os detalhes das mudanças que foram implementadas.
- Os principais processos de gerenciamento de configuração são gerenciamento de mudanças, gerenciamento de versões, a construção de sistemas e o gerenciamento de releases.
- Gerenciamento de mudanças envolve avaliar propostas de mudanças do sistema de clientes e outros *stakeholders* e decidir se é efetivo implementá-las em um novo release de um sistema.
- Gerenciamento de versões envolve manter o acompanhamento das diferentes versões de componentes de software como as mudanças são feitas.

Gerenciamento de *releases*

- Um release de um sistema é uma versão de um sistema de software distribuído aos clientes.
- Para um software de mercado, normalmente é possível identificar dois tipos de release: releases grandes que proporcionam nova funcionalidade importante, e releases menores, que reparam bugs e solucionam os problemas dos clientes.
- Para softwares customizados ou linhas de produto de software, os releases do sistema podem ter que ser produzidos para cada cliente e clientes individuais podem estar executando várias versões diferentes do sistema, ao mesmo tempo.

Acompanhamento de *releases*

- No caso de um problema, pode ser necessário reproduzir exatamente o software que foi entregue para um cliente particular.
- Quando é produzida uma versão do sistema, essa deve ser documentada para assegurar que possa ser recriada no futuro.
- Isso é particularmente importante para sistemas embutidos e customizados, de longa vida útil, tais como os que controlam máquinas complexas.
 - ✓ Os clientes podem usar um único release desses sistemas por muitos anos e podem exigir mudanças específicas para um sistema de software especial muito tempo depois da data do release original.

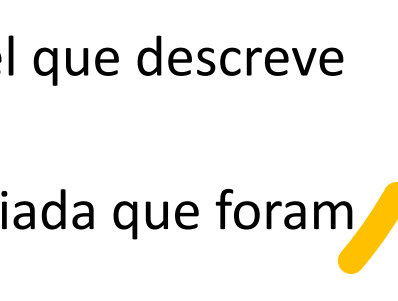
Documentação de *releases*

- Para um documento de release, você precisa gravar as versões específicas dos componentes do código-fonte que foram usados para criar o código executável.
- Você deve manter cópias dos arquivos de código-fonte, executáveis correspondentes e todos os dados e arquivos de configuração.
- Você também deve gravar as versões do sistema operacional, bibliotecas, compiladores e outras ferramentas usadas para construir o software.

Planejamento de *releases*

- Bem como o trabalho técnico envolvido na preparação e distribuição de um release, deve-se preparar material de publicidade e divulgação além de estratégias de marketing para convencer os clientes a comprarem o novo release do sistema.
- Calendário de release
 - ✓ Se os releases são muito frequentes ou exigem atualizações do hardware, os clientes podem não mudar para o novo release, especialmente se tiverem que pagar por isso.
 - ✓ Se os releases do sistema são muito pouco frequentes, pode se perder parte do mercado pois os clientes mudam para sistemas alternativos.

Componentes de *releases*

- Bem como o código executável do sistema, um release também pode incluir:
 - ✓ Os arquivos de configuração definem como o release deve ser configurado para instalações particulares;
 - ✓ Os arquivos de dados, tais como arquivos de mensagens de erro, são necessários para a operação do sistema ser bem sucedida;
 - ✓ Um programa de instalação que é usado para ajudar a instalar o sistema no hardware alvo;
 - ✓ Documentação eletrônica e em papel que descreve o sistema;
 - ✓ Empacotamento e publicidade associada que foram projetadas para esse release.
- 

Fatores que influenciam o planejamento de *releases* do sistema

Fator	Descrição
Qualidade técnica do sistema	Caso sejam relatados defeitos graves de sistema, que afetem a maneira como muitos clientes o usam, pode ser necessário emitir um <i>release</i> de reparação de defeitos. Pequenos defeitos de sistema podem ser reparados mediante a emissão de <i>patches</i> (normalmente distribuídos pela Internet) que podem ser aplicados no <i>release</i> atual do sistema.
Mudanças de plataforma	Talvez você precise criar um novo <i>release</i> de uma aplicação de software quando uma nova versão da plataforma do sistema operacional for lançada.
Quinta lei de Lehman (ver Capítulo 9)	Essa 'lei' sugere que se você adicionar nova funcionalidade a um sistema, você também introduzirá <i>bugs</i> que limitarão a quantidade de funcionalidade que pode ser incluída no próximo <i>release</i> . Portanto, um <i>release</i> de sistema com funcionalidade nova e significativa pode ser seguido por um <i>release</i> que se concentra em reparar os problemas e melhorar o desempenho.

Fatores que influenciam o planejamento de *releases* do sistema

Concorrência	Para software de mercado de massa, um novo <i>release</i> de sistema pode ser necessário porque um produto concorrente introduziu novos recursos e a fatia de mercado pode ser perdida caso estes não sejam fornecidos aos clientes existentes.
Requisitos de marketing	O departamento de marketing de uma organização pode ter feito um compromisso para <i>releases</i> estarem disponíveis em uma determinada data.
Propostas de mudança de cliente	Para sistemas customizados, os clientes podem ter feito e pago por um conjunto específico de propostas de mudanças de sistema e eles esperam um <i>release</i> de sistema assim que estas sejam implementadas.

FERRAMENTAS

- Concurrent Versions System (CVS)
- Subversion (SVN)
- Git
- TortoiseGit



Concurrent Versions System (CVS)


- Site oficial: <http://cvs.nongnu.org/>
- Software livre
- Destaque:
 - Cliente-servidor
 - Unreserved checkouts: um arquivo pode ser trabalhado por mais de um desenvolvedor ao mesmo tempo
 - Vendor branches: compartilhamento de versões próprias de arquivos selecionados

Subversion (SVN)

- Site oficial: <https://subversion.apache.org/>
- Software livre/aberto
- Destaques:
 - Cliente-servidor
 - Atomic operations: “aplica todas as mudanças ou nenhuma, de modo que nenhuma mudança parcial poderá quebrar o fonte original.”

Git

- Site oficial: <https://git-scm.com/>
- Software aberto
- Destaque:
 - Desempenho
 - Distribuído

	git
Desenvolvedor	Linus Torvalds, Junio Hamano
Plataforma	Multiplataforma
Lançamento	7 de abril de 2005 (15 anos)
Versão estável	2.22.0 (7 de junho de 2019; há 15 meses)
Mercado-alvo	Versionamento de Software
Linguagem	C, Shell, Perl
Sistema operacional	POSIX
Gênero(s)	Sistema de controle de versões
Licença	GNU GPLv2
Estado do desenvolvimento	Corrente
Tamanho	~44MB
Página oficial	git-scm.com

GitHub

- **GitHub** é uma plataforma de hospedagem de código-fonte com [controle de versão](#) usando o [Git](#).
- Ele permite que programadores, utilitários ou qualquer usuário cadastrado na plataforma contribuam em projetos privados e/ou [Open Source](#) de qualquer lugar do mundo.
- Utilizado por programadores para divulgação de seus trabalhos ou para que outros programadores contribuam com o projeto, além de promover fácil comunicação através de recursos que relatam problemas ou mesclam [repositórios](#) remotos.

GitHub	
GitHub	
<i>Slogan</i>	<i>Social Code Hosting</i>
Proprietário(s)	Microsoft
País de origem	 Estados Unidos
Idioma(s)	Inglês
Lançamento	10 de abril de 2008
Posição no Alexa	64 (junho de 2018) ^[1]
Desenvolvedor	GitHub, Inc. - 2008 até 2018 Microsoft Corporation - 2018 até atualmente
Endereço eletrônico	github.com 