

Gerenciamento de configuração



Elton Moraes

Objetivos



Compreender os processos e procedimentos envolvidos no gerenciamento de mudanças de software;

Conhecer as funcionalidades essenciais fornecidas por um sistema de gerenciamento de versões e os relacionamentos entre o gerenciamento das versões e a construção de sistemas;

Entender as diferenças entre uma versão de sistema e uma release de sistema, assim como os estágios no processo de gerenciamento de releases.

Introdução

Os sistemas de software sempre mudam durante seu desenvolvimento e uso.

- Bugs
- Mudança de requisitos
- Novas versões de hardware e novas plataformas de sistemas
- Concorrência

O que é gerenciamento de Configuração?

CM – Configuration Management

O que é gerenciamento de configuração?

O gerenciamento de configuração (*Configuration Management – CM*) está relacionado com as políticas, processos e ferramentas para gerenciamento de mudanças dos sistemas de software. (SOMMERVILLE, 2011)

O gerenciamento de configuração de software (*Software Configuration Management – SCM*), também é chamado de gestão de alterações, é um conjunto de atividades destinadas a gerenciar as alterações. (PRESSMAN; MAXIM, 2016)

O que é gerenciamento de configuração?

O gerenciamento de configuração é útil para projetos individuais, que vários desenvolvedores e/ou equipes trabalham ao mesmo tempo. O gerenciamento de configuração garante que as equipes tenham acesso a informações sobre um sistema que está em desenvolvimento e não interfiram no trabalho umas das outras.

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

- Gerenciamento de mudanças;
- Gerenciamento de versões;
- Construção do sistema;
- Gerenciamento de releases.

O que é gerenciamento de configuração?

O gerenciamento de configuração, por vezes, é considerado parte do gerenciamento de qualidade de software. Geralmente possuem o mesmo gerente para qualidade e configuração.

A definição e a utilização de padrões de gerenciamento de configuração são essenciais para certificação de qualidade de padrões ISO 9000, CMM e CMMI.

- ISO 9000 – Sistema de Gestão de Qualidade
- CMM – Modelo de Maturidade da Capacitação
- CMMI – Modelo de Maturidade em Capacitação – Integração

CMM – *Capability Maturity Model* é um modelo de gestão que aponta as melhores práticas para que a empresa consiga diagnosticar e avaliar a maturidade de seus processos, sejam eles relacionados ao desenvolvimento de softwares ou não.



As características do CMM envolvem um conjunto de elementos que vão descrever os estágio de maturidade de um processo organizacional, tais como:


- Ponto de partida do processo;
- Práticas e subpráticas para momentos-chave do processo;
- Vocabulário comum;
- Visão compartilhada do processo;
- Framework com foco na priorização de ações;
- Melhorias a serem implementadas.



Níveis de maturidade:

- Inicial
- Repetível
- Definido
- Gerenciado com métricas
- Otimizado





CMMI – *Capability Maturity Model Integration* (**Modelo de Capacidade e Maturidade Integrado**) é considerado uma evolução do CMM, por acrescentar o fator integração, buscando estabelecer um modelo único e integrado para promoção da melhoria constante dos processos corporativos.

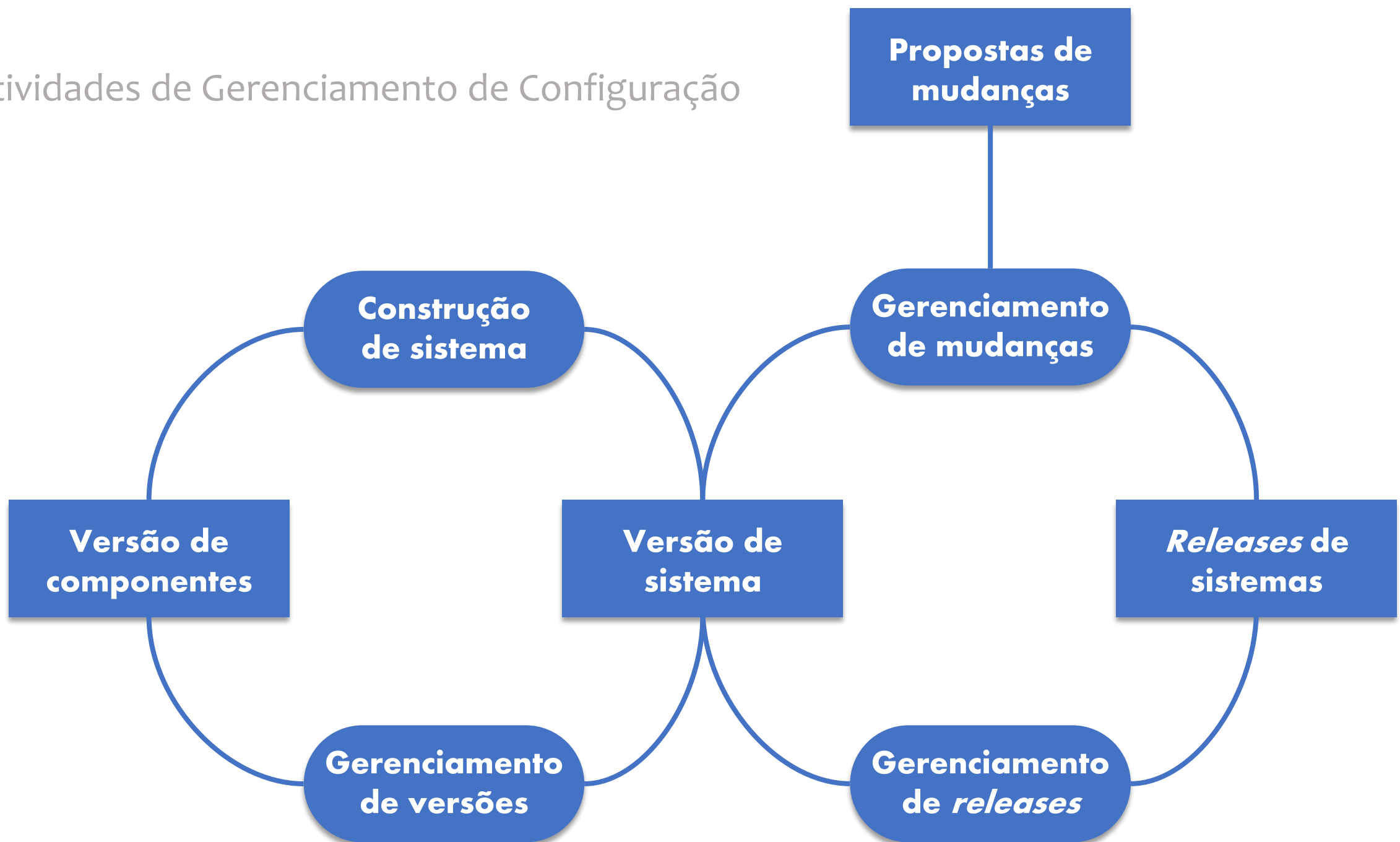


Métricas:

- Inicial
- Gerenciado
- Definido
- Gerenciado Quantitativamente
- Otimizado

Atividades de Gerenciamento de Configuração

Atividades de Gerenciamento de Configuração



Gerenciamento de mudanças

Envolve manter o acompanhamento das solicitações dos clientes e desenvolvedores por mudanças no software, definir os custos e o impacto de fazer tais mudanças, bem como decidir se e quando as mudanças deve ser implementadas.



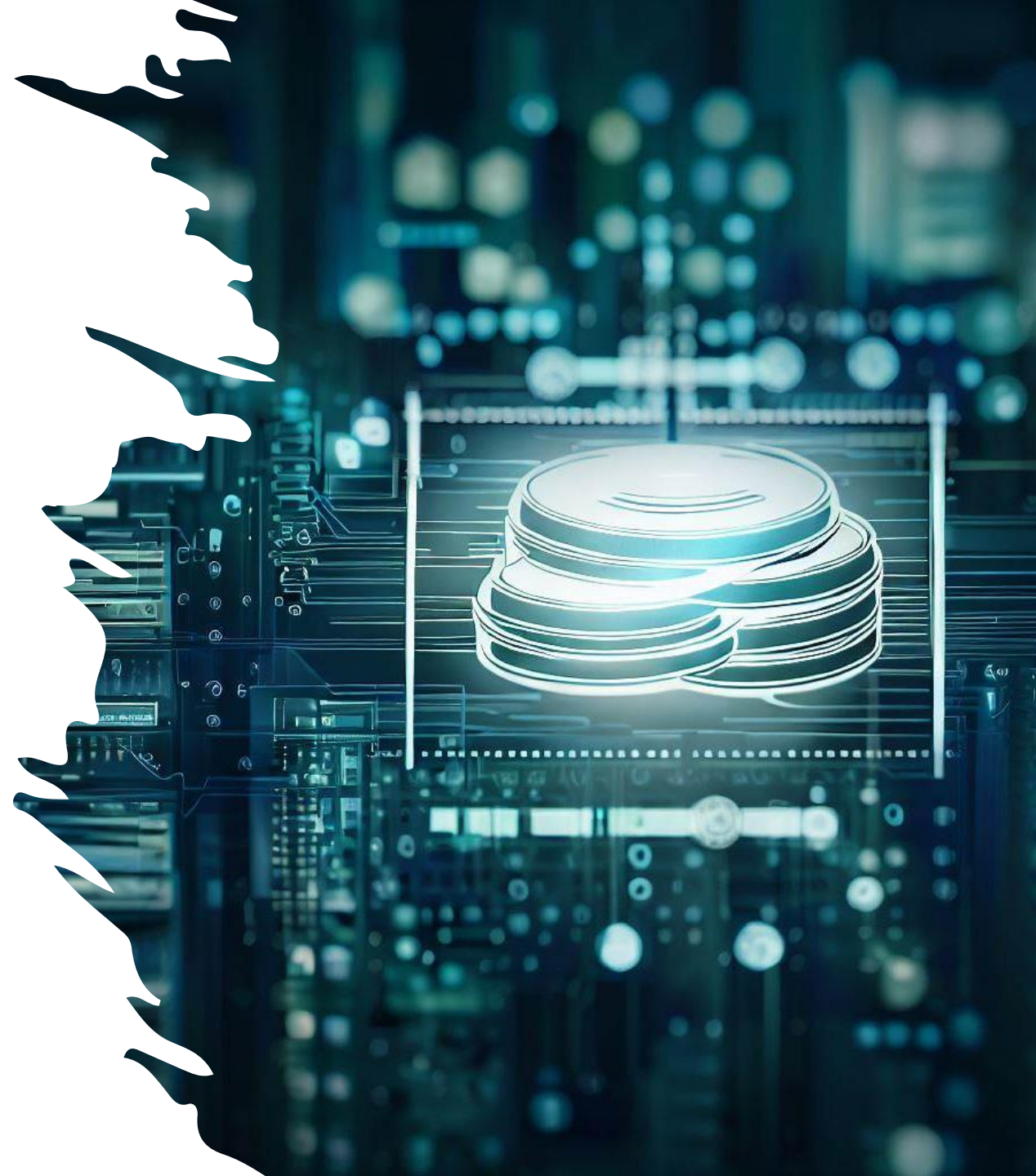
Gerenciamento de versões

Envolve manter o acompanhamento de várias versões de componentes do sistema e assegurar que as mudanças nos componentes, realizadas por diferentes desenvolvedores, não interfiram umas nas outras.



Construção do sistema

É o processo de montagem de componentes de programa, dados e bibliotecas e, em seguida, compilação e ligação destes, para criar um sistema executável.



Gerenciamento de releases

Envolve a preparação de software para o *release* externo e manter o acompanhamento das versões de sistema que foram liberadas para uso do cliente.



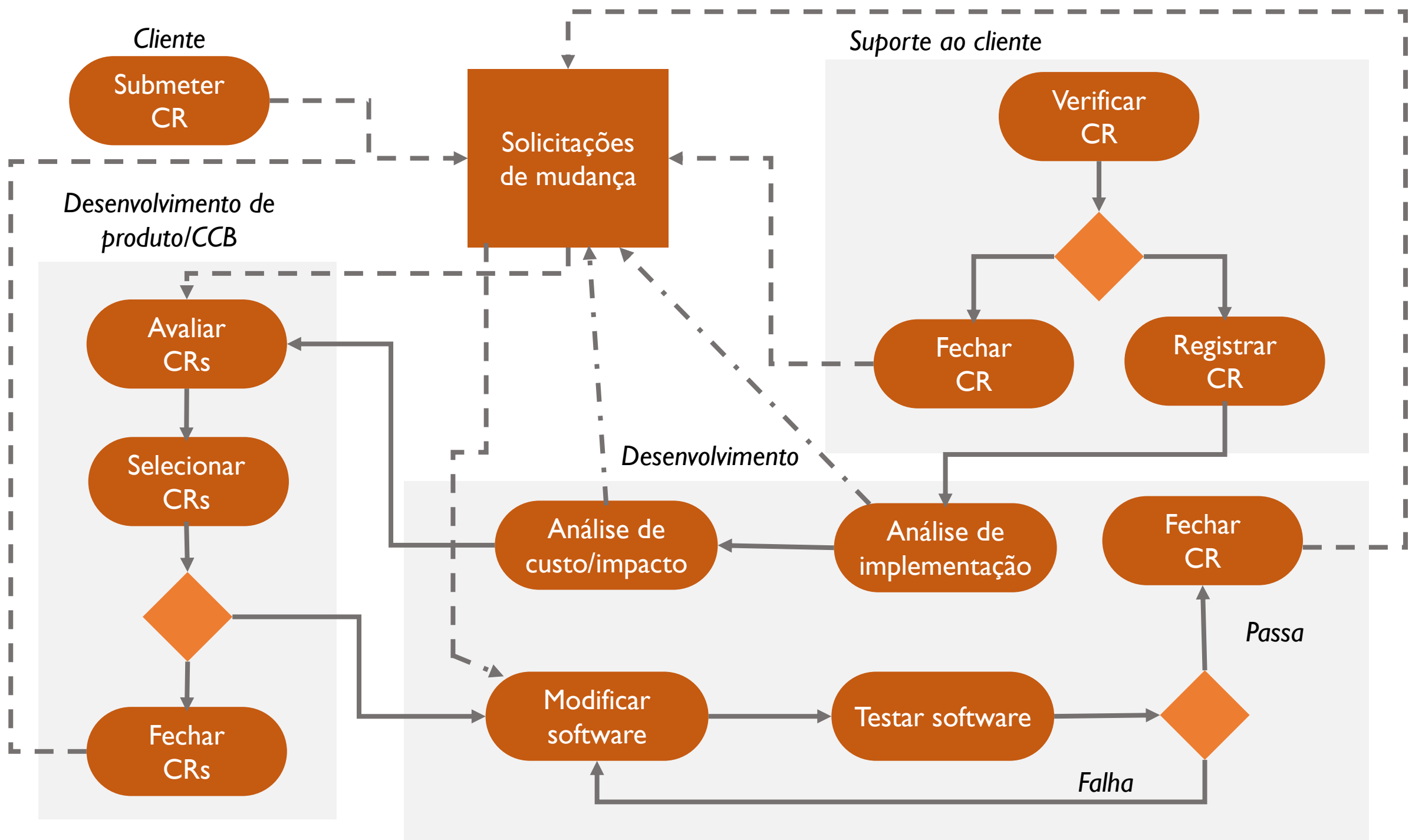
Terminologias de Gerenciamento de Configuração

Termo	Explicação
Item de configuração ou item de configuração de software (SCI, do inglês <i>software configuration item</i>)	Qualquer coisa associada a um projeto de software (projeto, código, dados de teste, documentos etc.) que tenha sido colocado sob controle de configuração. Muitas vezes, existem diferentes versões de um item de configuração. Itens de configuração têm um nome único.
Controle de configuração	O processo de garantia de que versões de sistemas e componentes sejam registradas e mantidas para que as mudanças sejam gerenciadas e todas as versões de componentes sejam identificadas e armazenadas por todo o tempo de vida do sistema.
Versão	Uma instância de um item de configuração que difere, de alguma forma, de outras instâncias desse item. As versões sempre têm um identificador único, o qual é geralmente composto pelo nome do item de configuração mais um número de versão.
<i>Baseline</i>	Uma <i>baseline</i> é uma coleção de versões de componentes que compõem um sistema. As <i>baselines</i> são controladas, o que significa que as versões dos componentes que constituem o sistema não podem ser alteradas. Isso significa que deveria sempre ser possível recriar uma <i>baseline</i> a partir de seus componentes.
<i>Codeline</i>	Uma <i>codeline</i> é um conjunto de versões de um componente de software e outros itens de configuração dos quais esse componente depende.

Termo	Explicação
<i>Mainline</i>	Trata-se de uma sequência de <i>baselines</i> que representam diferentes versões de um sistema.
<i>Release</i>	Uma versão de um sistema que foi liberada para os clientes (ou outros usuários em uma organização) para uso.
Espaço de trabalho	É uma área de trabalho privada em que o software pode ser modificado sem afetar outros desenvolvedores que possam estar usando ou modificando o software.
<i>Branching</i>	Trata-se da criação de uma nova <i>codeline</i> de uma versão em uma <i>codeline</i> existente. A nova <i>codeline</i> e uma <i>codeline</i> existente podem, então, ser desenvolvidas independentemente.
Merging	Trata-se da criação de uma nova versão de um componente de software, fundindo versões separadas em diferentes <i>codelines</i> . Essas <i>codelines</i> podem ter sido criadas por um <i>branch</i> anterior de uma das <i>codelines</i> envolvidas.
Construção de sistema	É a criação de uma versão de sistema executável pela compilação e ligação de versões adequadas dos componentes e bibliotecas que compõem o sistema.

Gerenciamento de mudanças

A mudança é uma realidade para grandes sistemas. As necessidades e requisitos organizacionais se alteram, bugs precisam ser reparados e os sistemas necessitam se adaptar às mudanças.



Gerenciamento de mudanças

O gerenciamento de mudanças destina-se a garantir que a evolução do sistema seja um processo gerenciado e que seja dada prioridade às mudanças mais urgentes e efetivas.

O processo de gerenciamento de mudanças está relacionado com a análise de custos e benefícios das mudanças propostas, a aprovação dessas mudanças que valem a pena e o acompanhamento dos componentes do sistema que foram alterados.

Quadro 25.1

Um formulário de solicitação de mudança parcialmente concluído

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. 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.

Prioridade de mudança: Média

Implementação de mudança:

Esforço estimado: 2 horas

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

Data de decisão do CCB: 30/jan./2009

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

Implementador de mudança:

Data de mudança:

Data de submissão ao QA:

Decisão de QA:

Data de submissão ao CM:

Comentários:

CCB – Change Control Board

O CCB – Comitê de Controle de Mudanças, revisa e aprova todas as solicitações de mudança, a menos que as mudanças envolvam a pequenas correções;

O CCB considera o impacto da mudança a partir de um ponto de vista estratégico e organizacional, em vez de um ponto de vista técnico.

CCB – *Change Control Board*

Fatores importantes, considerados pelo CCB, ao decidir pela aprovação ou não de uma 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;

A 3D diagram of a network graph. It features several nodes, which are white, hollow cylindrical rings of varying sizes, connected by thin, light blue lines representing edges. The nodes are arranged in a non-uniform pattern across the frame. The connections include a small cluster of three nodes in the upper left, a central hub-and-spoke structure, and a long, straight edge connecting a node at the bottom left to a node in the middle right. The entire scene is set against a dark blue background with a subtle gradient and soft shadows on the nodes.

Gerenciamento de versões

Gerenciamento de versões

O gerenciamento de versões é o processo de acompanhamento de diferentes versões de componentes de software ou itens de configuração e os sistemas em que esses componentes são usados.

O gerenciamento de versões pode ser considerado como gerenciamento de *codelines* e *baselines*.

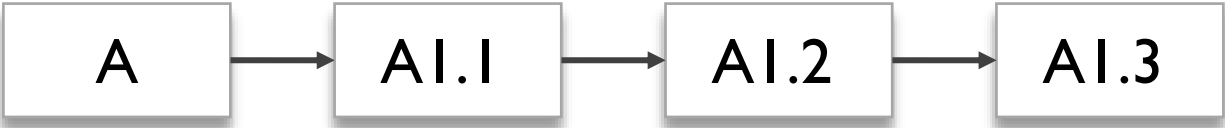
Gerenciamento de versões

Uma *codeline* é uma sequência de versões de código-fonte com versões posteriores na sequência derivadas de versões anteriores.

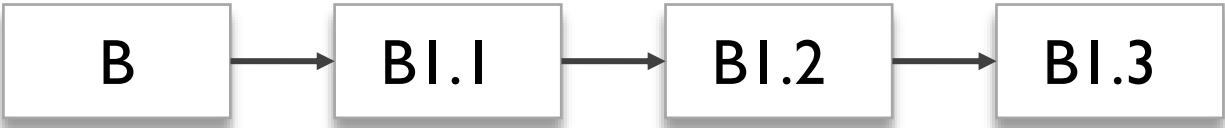
Uma *baseline* é uma definição de um sistema específico, portanto, especifica a versão de cada componente incluída no sistema, bibliotecas, arquivos de configuração etc.

A *mainline* é uma sequência de versões de sistema desenvolvidas a partir de uma *baseline* original.

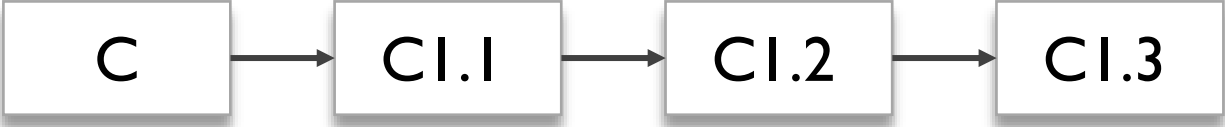
Codeline (A)



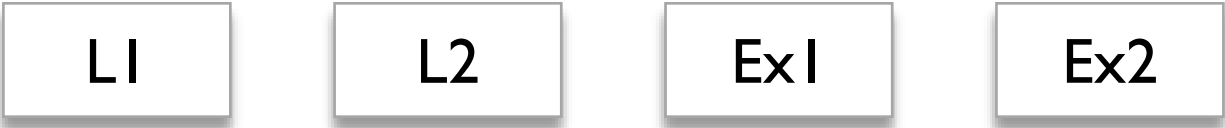
Codeline (B)



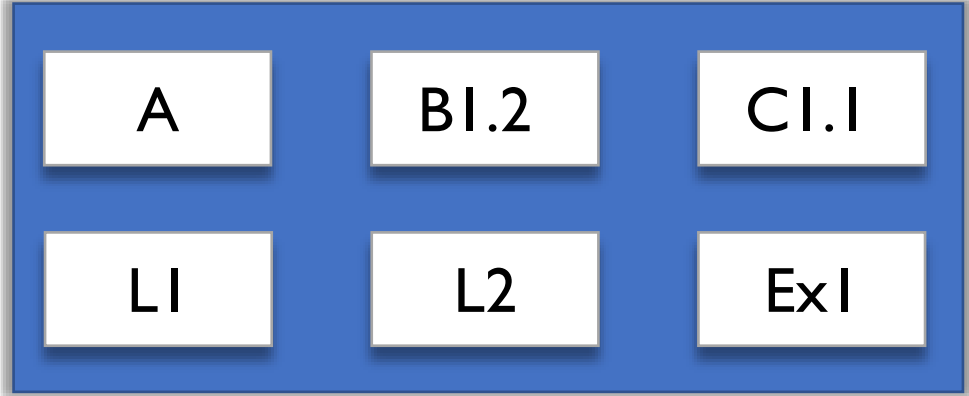
Codeline (C)



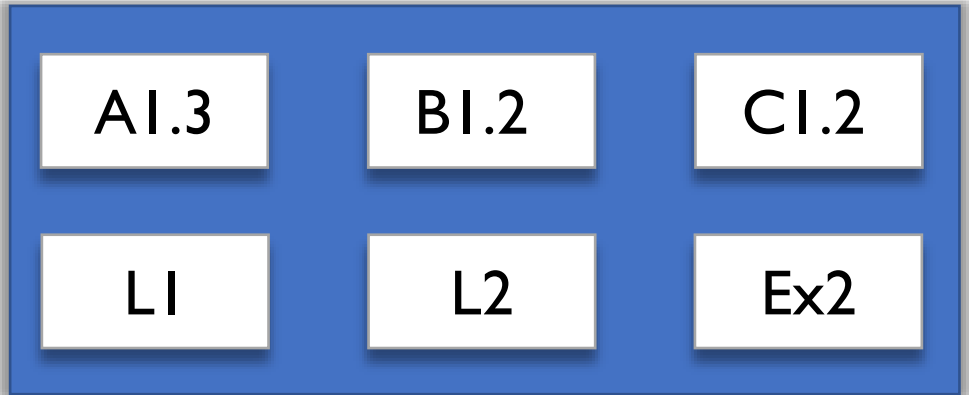
Bibliotecas e componentes externos



Baseline - V1



Baseline - V2



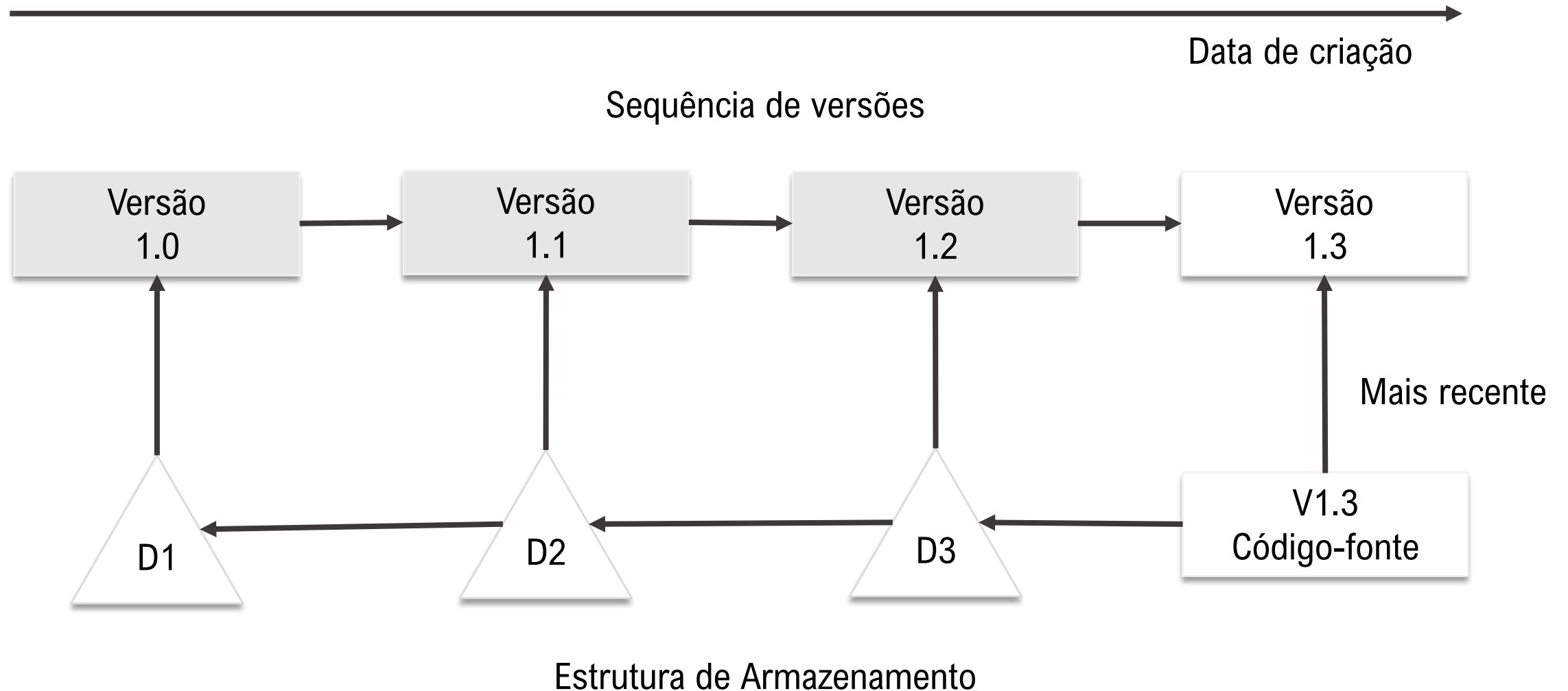
Mainline

Gerenciamento de versões

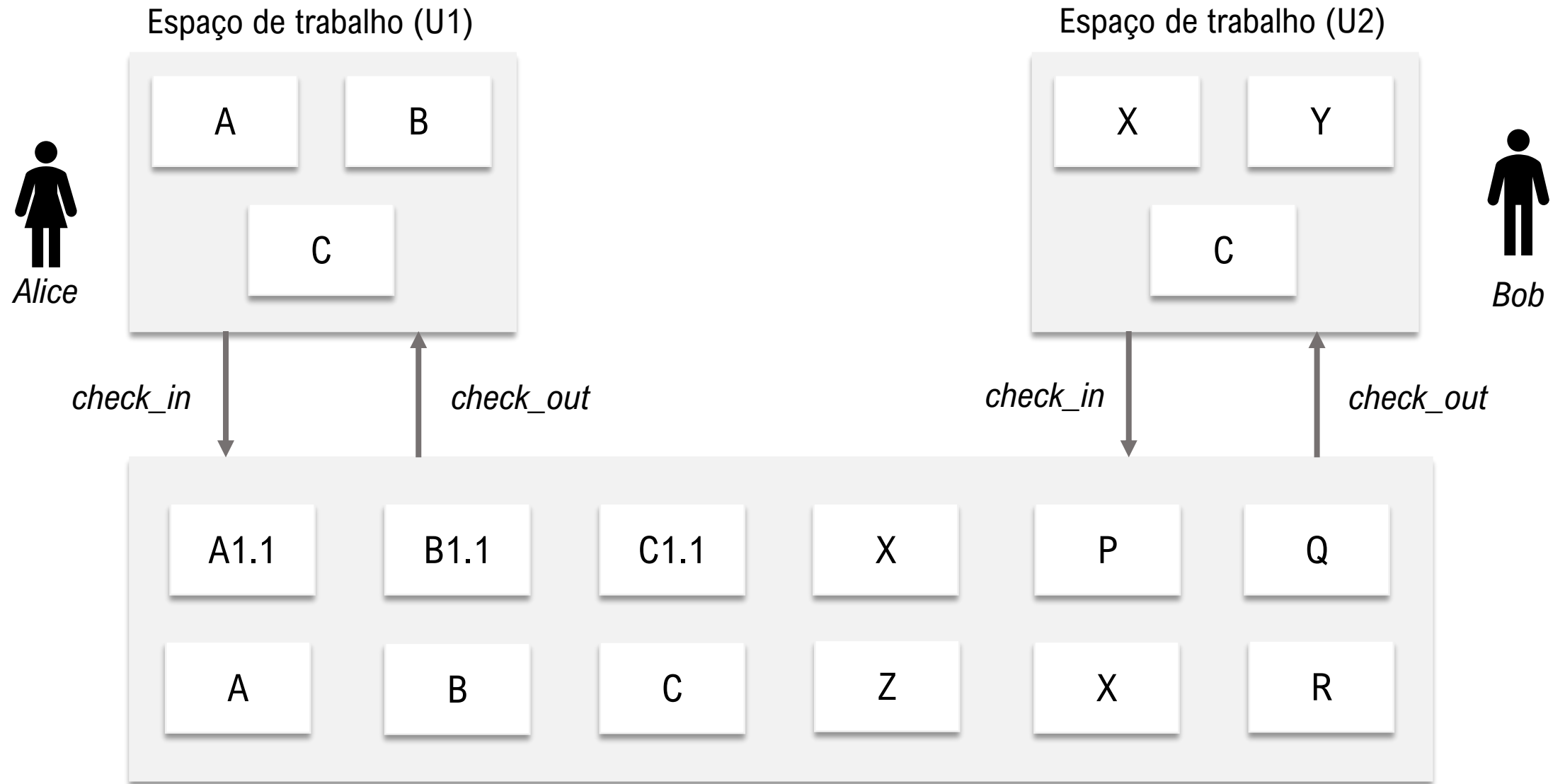
O uso de ferramentas de gerenciamento de versões identificam, armazenam e controlam o acesso a diferentes versões de componentes. Geralmente oferecem recursos de:

- Identificação de versão e release;
- Gerenciamento de armazenamento;
- Registro de histórico de mudanças;
- Desenvolvimento independente;
- Suporte a projetos.

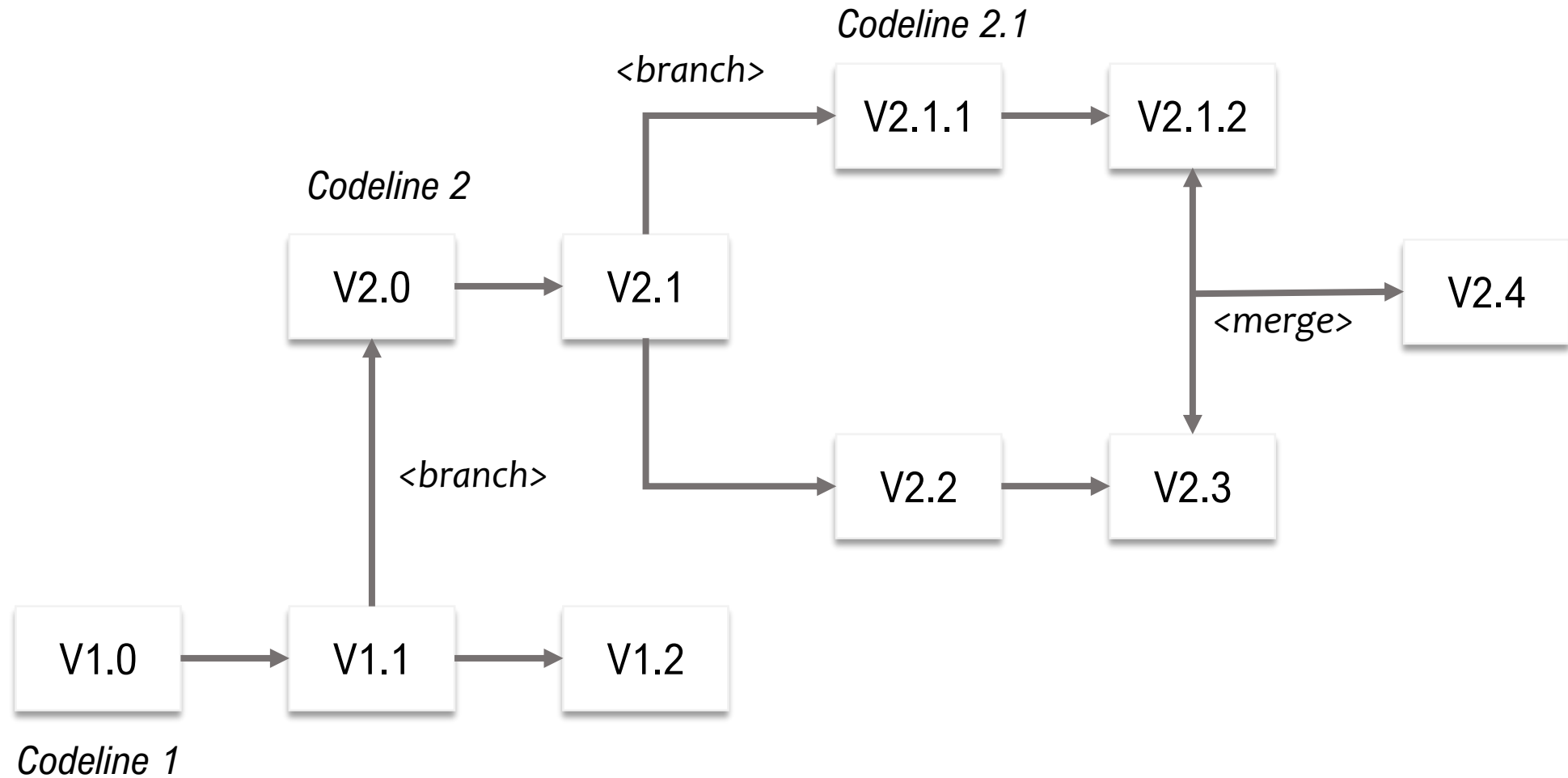
Gerenciamento de versões usando deltas



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



Branching e merging



Construção de sistemas

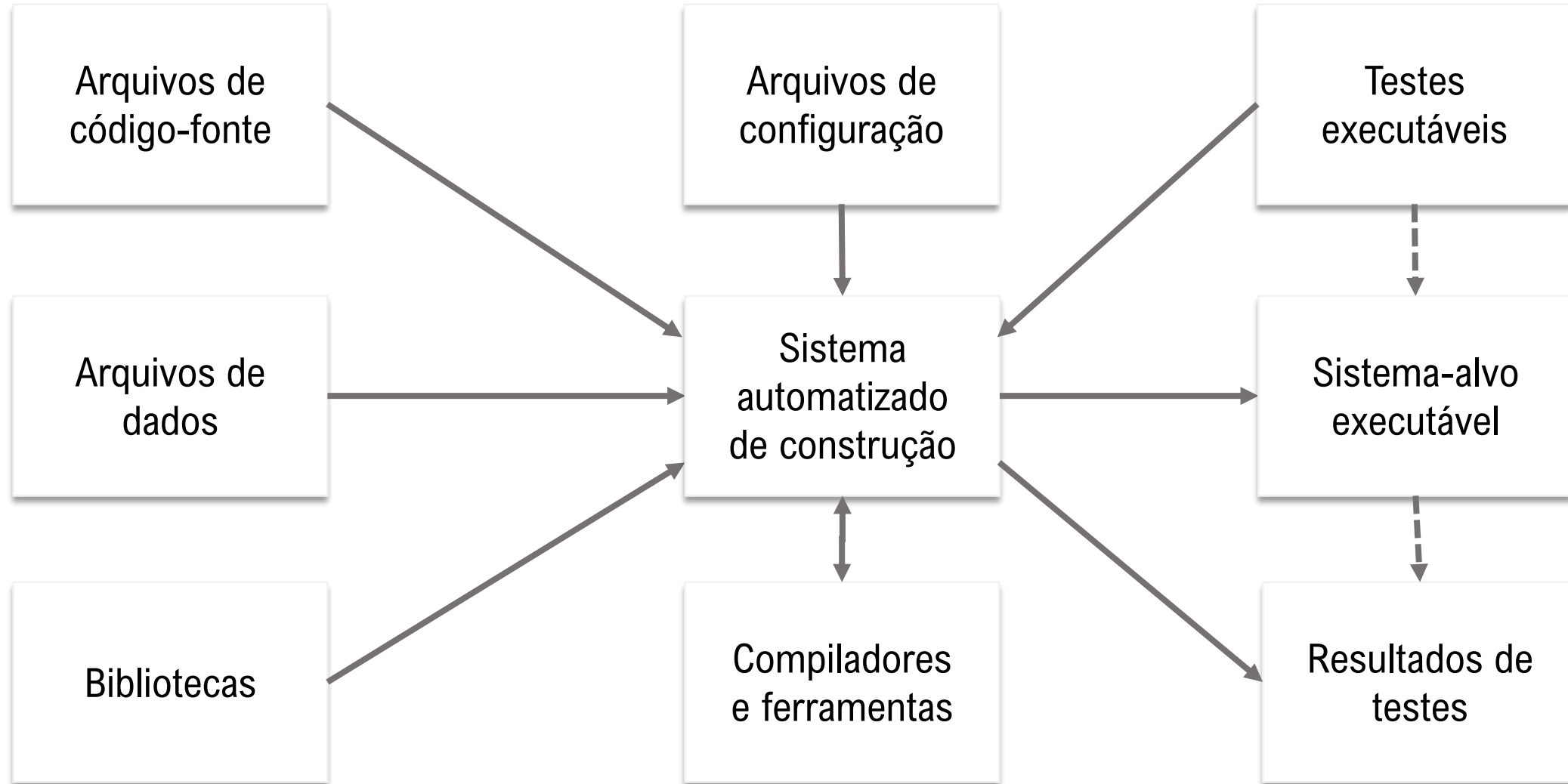


Construção de sistemas

A construção de sistemas é o processo da criação de um sistema completo, executável por meio da construção e ligação dos componentes de sistema, bibliotecas externas, arquivos de configuração etc.

A construção de sistemas envolve a montagem de uma grande quantidade de informações sobre o software e seu ambiente operacional.

A construção de sistemas



Construção de sistemas

Ferramentas de construção pode fornecer:

- Geração de script de construção;
- Integração de sistema de gerenciamento de versões;
- Recompilação mínima;
- Criação de sistemas executáveis;
- Automação de testes;
- Emissão de relatórios;
- Geração de documentação;

Gerenciamento de *releases*



Gerenciamento de releases

Um *release* de sistema é uma versão de um sistema de software distribuída aos clientes.

Para softwares de mercado de massa, há dois tipos de *releases*: os principais, que fornecem nova e significativa funcionalidade; e *releases* menores, que reparam *bugs* e corrigem problemas de clientes.

Para softwares customizados ou linhas de produtos de software, o gerenciamento de *releases* de sistema é um processo complexo.

Fatores que influenciam o planejamento de release de sistema

Fatores que influenciam o planejamento de release de sistema

- Qualidade técnica do sistema:
 - Relatos de defeitos graves de sistema;
- Mudanças de plataforma:
 - Mudança de sistema operacional;
- Quinta lei de Lehman:
 - Nova funcionalidade introduz novos bugs.
- Concorrência:
 - Concorrente lança novos recursos.

Fatores que influenciam o planejamento de release de sistema

- Requisitos de marketing:
 - Marketing agenda um lançamento de release;
- Propostas de mudança de clientes:
 - Cliente solicita novas mudanças.

Modelo de Documento



SUA LOGO
AQUI

DATA: [DATA]
VERSÃO: [VERSÃO]

Plano de Gerenciamento de Configuração de Software

Introdução

1.1 Objetivo

1.2 Escopo

1.3 Definições, Acrônimos e Abreviações

1.4 Referências

1.5 Visão Geral do Plano de Gerenciamento de Configuração

SUA LOGO
AQUI

DATA: [DATA]
VERSÃO: [VERSÃO]

Configuração do Ambiente

2.1 Hardware

2.2 Software

2.3 Ferramentas de Controle de Versão

2.4 Ambientes de Desenvolvimento, Teste e Produção

2.5 Controles de Acesso

Identificação e Controle de Configurações

3.1 Estrutura de Nomenclatura

3.2 Critérios de Identificação

3.3 Controle de Versões

3.4 Controle de Mudanças

3.5 Revisão e Aprovação de Mudanças

SUA LOGO
AQUI

DATA: [DATA]
VERSÃO: [VERSÃO]

Baselines e Controle de Mudanças

- 4.1 Estabelecimento de Baselines
- 4.2 Revisão e Aprovação de Baselines
- 4.3 Controle e Rastreamento de Mudanças
- 4.4 Análise de Impacto de Mudanças
- 4.5 Aprovação e Implementação de Mudanças

Rastreamento de Configurações

- 5.1 Sistema de Controle de Versão
- 5.2 Documentação de Configurações
- 5.3 Auditorias de Configuração

SUA LOGO
AQUI

DATA: [DATA]
VERSÃO: [VERSÃO]

Responsabilidades

- 6.1 Responsabilidades do Gerente de Projeto
- 6.2 Responsabilidades dos Desenvolvedores
- 6.3 Responsabilidades dos Testadores
- 6.4 Responsabilidades dos Administradores de Configuração

Comunicação e Colaboração

- 7.1 Canais de Comunicação
- 7.2 Reuniões de Revisão de Mudanças
- 7.3 Ferramentas Colaborativas

SUA LOGO
AQUI

DATA: [DATA]
VERSÃO: [VERSÃO]

Treinamento

8.1 Treinamento Inicial

8.2 Treinamento Contínuo

Auditoria e Revisão

9.1 Auditorias Regulares

9.2 Revisões Periódicas

SUA LOGO
AQUI

DATA: [DATA]
VERSÃO: [VERSÃO]

Controle de Documentação

10.1 Controle de Versões

10.2 Armazenamento de Documentos

10.3 Controle de Acesso

Glossário

Aprovação

A top-down view of a wooden workbench covered with an assortment of hand tools. The tools are organized into several rows. In the top row, there's a red level, a yellow handle, a blue triangle square, another yellow handle, a pair of red-handled pliers, a silver adjustable wrench, a red-handled screwdriver, a silver caliper, and a red-handled screwdriver. The second row features a large black and red plier, a red-handled screwdriver, a red-handled screwdriver, a red-handled screwdriver, a red-handled screwdriver, a red-handled screwdriver, a red-handled screwdriver, a red-handled screwdriver, a red-handled screwdriver, and a red-handled screwdriver. The third row shows a silver adjustable wrench, several hex keys of different sizes, a row of six silver screws, and a yellow level with a circular bubble level. The bottom row includes a red-handled screwdriver, a yellow level, a silver adjustable wrench, a black brush, a hammer with a wooden handle, a black and yellow handle, a silver flashlight, a yellow and red handle, a red handle, and a yellow handle. The word "Ferramentas" is written in a white, sans-serif font across the center of the image.

Ferramentas

- Controle de versões:
 - Git;
 - Subversion;
 - Mercurial;
- Controle de mudanças:
 - Jira;
 - Trello;
 - Asana;

Ferramentas

- Integração contínua e entrega contínua:
 - Jenkins;
 - Travis CI
 - CircleCI
- Gerenciamento de dependências:
 - Maven;
 - Gradle;
 - NPM

The background features a complex network of thin, curved lines in shades of purple and gold. These lines connect various small, circular nodes, some of which are highlighted with a double-circle effect. The overall composition is dynamic and abstract, with lines flowing across the dark space.

Resumo

Resumo

- O gerenciamento de configuração é o gerenciamento de um sistema de software em constante evolução. Durante a manutenção de um sistema, uma equipa de CM é responsável por garantir que as mudanças sejam incorporadas no sistema de uma forma controlada e que os registros sejam mantidos com os detalhes das mudanças que foram implementadas.
- Os principais processos de gerenciamento de configuração estão interessados no gerenciamento de mudanças, no gerenciamento de versões, na construção de sistemas e no gerenciamento de releases. Ferramentas de software estão disponíveis para apoiar todos esses processos.

Resumo

- O gerenciamento de mudanças envolve avaliar propostas de mudanças dos clientes de sistema e outros *stakeholders* e decidir se é efetivo implementá-las em um novo *release* de um sistema.
- O gerenciamento de versões envolve o acompanhamento das diferentes versões dos componentes de software criadas à medida que as mudanças são feitas.
- A construção de sistemas é o processo de montagem de componentes de sistema em um programa executável para executar em um sistema de computador-alvo.

Resumo

- O software deve ser reconstruído e testado frequentemente, imediatamente após a construção de uma nova versão. Isso facilita a detecção de *bugs* e problemas introduzidos desde a última construção.
- Os *releases* de sistema incluem códigos executáveis, arquivos de dados, arquivos de configuração e documentação. O gerenciamento de versões envolve tomar decisões sobre as datas de *release* de sistema, preparar todas as informações para distribuição e documentar cada *release* de sistema.



Atividade

Atividade

Criar um plano de gerenciamento de configuração para um projeto fictício. A atividade pode ser desenvolvida em grupo, com base em um projeto já em andamento. Discuta com seus colegas a experiência e desafios do gerenciamento de configuração.

