

Estudo Autodidata Git e GitHub



Aula: 3

Tema: A Evolução do Git e GitHub

Um pouco de história

Linha do tempo do vcs (Softwares de controle de versão)

1. 1985 CVS

O **Concurrent Versions System** (CVS) é um dos primeiros sistemas de controle de versão amplamente utilizados, **desenvolvido em 1985 por Dick Grune**. Ele foi uma evolução do RCS (Revision Control System), permitindo colaboração simultânea entre vários desenvolvedores em um mesmo projeto.

✧ Principais Características do CVS:

- ✓ **Controle de Versão Centralizado**: CVS usa um repositório central para armazenar os arquivos e suas versões, permitindo que os desenvolvedores façam checkout, commit e atualização dos arquivos.
- ✓ **Suporte para Múltiplos Usuários**: Diferente do RCS, o CVS permite que várias pessoas trabalhem no mesmo projeto simultaneamente.
- ✓ **Registro de Histórico**: Mantém um histórico completo de mudanças nos arquivos.
- ✓ **Branches e Tags**: Possui suporte para ramificações e marcação de versões específicas do projeto.
- ✓ **Modelo de Trabalho Offline**: Os desenvolvedores podem trabalhar localmente e depois enviar suas alterações para o repositório.

Muitas das características das quais tem o GitHub hoje em dia

✧ Por que o CVS caiu em desuso?

Apesar de ter sido revolucionário para sua época, o CVS apresentava diversas limitações:

- ✗ **Falta de suporte** nativo para operações atômicas, podendo levar a inconsistências no repositório.
- ✗ Problemas de desempenho com repositórios grandes.
- ✗ Gerenciamento de branches complicado em comparação com ferramentas mais modernas.
- ✗ Falta de rastreamento eficiente de mudanças em diretórios.

Por conta disso, foi substituído por sistemas mais avançados, como Subversion (SVN) nos anos 2000 e, posteriormente, pelo Git, que hoje domina o mercado de versionamento.

2. 2000 Apache Subversion (SVN)

O Subversion (SVN) foi lançado em 2000 pela empresa CollabNet, com o objetivo de corrigir as deficiências do CVS e oferecer um sistema de controle de versão mais eficiente e confiável. Ele rapidamente se tornou popular e foi amplamente adotado por empresas e projetos de código aberto.

Características Principais do SVN

- ✓ **Controle de Versão Centralizado:** Assim como o CVS, o SVN mantém um repositório central, onde todas as versões dos arquivos são armazenadas.
- ✓ **Commits Atômicos:** Diferente do CVS, o SVN garante que um commit é aplicado completo ou não é aplicado, evitando inconsistências. **Porem ainda sofriam de uma demora de 1 a 2 minutos**
- ✓ **Controle de Diretórios:** Suporta versionamento de diretórios, algo que o CVS não fazia bem.
- ✓ **Branches e Tags Melhorados:** Criar branches e tags no SVN é mais eficiente e não cria cópias físicas dos arquivos.
- ✓ **Registro de Metadados:** Permite armazenar informações adicionais nos arquivos usando propriedades personalizadas.
- ✓ **Suporte para Binários:** Gerencia arquivos binários de maneira mais eficiente que o CVS.

Declínio do SVN

Apesar de ter sido amplamente utilizado nos anos 2000, o SVN começou a perder popularidade para sistemas descentralizado, como o Git, devido a algumas limitações:

- ✗ **Dependência do Servidor Central:** O repositório centralizado significa que se o servidor cair, ninguém pode fazer commits.
- ✗ **Branches e Merges Menos Eficientes:** Apesar de ser melhor que o CVS, o merge no SVN ainda era complicado comparado ao Git.
- ✗ **Performance Inferior em Projetos Grandes:** Para repositórios enormes, o desempenho do SVN não era tão bom quanto o Git.
- ✗ **Trabalho Offline Limitado:** O SVN permite apenas leitura offline, mas os commits precisam de conexão com o servidor.

Embora o Git tenha se tornado o padrão de mercado, o SVN ainda é usado em algumas empresas e projetos, especialmente onde um modelo centralizado é preferido por razões de segurança e conformidade.

3. BitKeeper 2000

◆ O que é o BitKeeper?

O BitKeeper foi criado por Larry McVoy e lançado em 2000 pela BitMover Inc. Ele foi um dos primeiros sistemas de controle de versão distribuído e serviu de "inspiração para o Git".

◆ Principais Características do BitKeeper

- ✓ **Modelo Distribuído:** Diferente do CVS e SVN, ele permitia que os desenvolvedores tivessem cópias locais do repositório, possibilitando commits offline.
- ✓ **Branches e Merges Eficientes:** O BitKeeper facilitava a criação e fusão de branches, algo que era mais difícil em sistemas centralizados como o SVN.
- ✓ **Controle de Acesso:** Tinha recursos avançados para gerenciar permissões de usuários.
- ✓ **Desempenho Melhorado:** Era mais rápido que SVN e CVS para operações em grandes projetos.

☒ O Caso BitKeeper x Linux (2005)

O BitKeeper foi adotado pelo projeto Linux em 2002, mas com restrições de uso (versão gratuita limitada). Em 2005, houve um conflito entre a BitMover e a comunidade Linux, levando à revogação da versão gratuita. Isso forçou Linus Torvalds a criar o Git, que rapidamente substituiu o BitKeeper e se tornou o padrão de mercado.

✂ BitKeeper Hoje

Em 2016, o BitKeeper foi disponibilizado como open-source, mas com a popularidade do Git, seu uso tornou-se muito nichado.



4. 2005 Git

O Git foi criado por Linus Torvalds (criador do linux) em 2005, o mesmo criador do Linux.

◆ Por que o Git foi criado?

Até 2005, o desenvolvimento do kernel Linux usava o BitKeeper, um sistema de controle de versão distribuído. No entanto, a empresa BitMover, dona do BitKeeper, decidiu revogar a licença gratuita usada pela comunidade Linux, por causa de um cliente que usava ing reversa e dava os direitos de um cliente pago a pessoas que não pagavam. Isso forçou Linus Torvalds a buscar uma alternativa.

Como não encontrou nenhuma solução que atendesse às suas necessidades, ele decidiu criar seu próprio sistema de controle de versão. Em apenas duas semanas, ele desenvolveu a primeira versão do Git.

◆ O que torna o Git diferente?

- ✓ **Distribuído**: Cada desenvolvedor tem uma cópia completa do repositório.
- ✓ **Rápido**: Usa algoritmos eficientes para rastrear mudanças.
- ✓ **Seguro**: Usa SHA-1 (e depois SHA-256) para garantir a integridade dos dados.
- ✓ **Branches e Merges eficientes**: Melhor gerenciamento de ramificações que SVN ou CVS.
- ✓ **Código Aberto**: Gratuito e amplamente adotado.

◆ Git Hoje

O Git rapidamente se tornou o “padrão global” para controle de versão e é usado por gigantes como Microsoft, Google, Facebook e Amazon. Plataformas como GitHub, GitLab e Bitbucket ajudaram a popularizá-lo ainda mais.

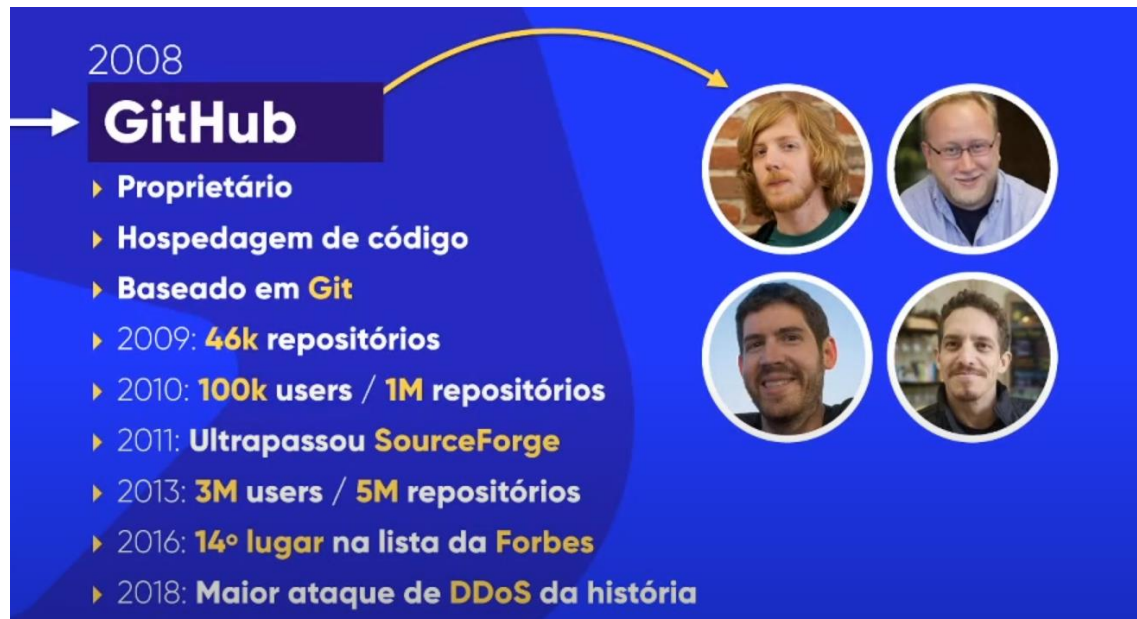


E o GitHub?

◆ GitHub – A Maior Plataforma de Hospedagem de Código do Mundo

O GitHub é uma plataforma de hospedagem de repositórios Git baseada na nuvem, lançada em 2008. Ele permite que desenvolvedores colaborem, compartilhem código e gerenciem projetos de software de forma eficiente.

Criado por Tom Preston-Werner, Chris Wanstrath, PJ Hyett e Scott Chacon.



2018 – O GitHub é comprado pela Microsoft



Em 2018 o GitHub é comprado pela Microsoft e em 2020 o Github compra a NPM, nada mais nada menos que o maior gestor de pacotes de JS do mundo.