

Universidade Federal de Pernambuco  
Centro de Informática  
Graduação em Ciência da Computação

Wendell Jânio de Oliveira Júnior  
wjoj@cin.ufpe.br

## **Armazenando Artefatos Associados às Alterações com Git**

Proposta de Trabalho de Graduação

Orientador: Paulo Henrique Monteiro Borba  
Área: Engenharia de Software

Recife  
2023

# 1 Resumo

Git é um Sistema de Controle de Versão (*Version Control System* – VCS) amplamente utilizado no desenvolvimento de *software*. Arquivos de texto são os mais comuns em repositórios Git, pois este sistema consegue automaticamente examinar o conteúdo desses arquivos e mostrar o que mudou entre as versões.

Apesar de possuir a interface de um VCS, Git é internamente um Sistema de Arquivos Endereçável por Conteúdo (*Content-Addressable Filesystem*) e permite que qualquer tipo de dado seja adicionado ao repositório e identificado diretamente pelo seu conteúdo. Desta forma, este sistema pode ser utilizado para armazenar diversas informações além daquelas normalmente associadas às versões do repositório.

Cada versão de um repositório Git é registrada por meio de *commits*. São objetos que armazenam uma mensagem explicativa para as alterações, o autor do *commit*, o autor das alterações, o conteúdo do repositório no momento em que o *commit* foi registrado e quando ocorreu. Embora diferentes fluxos de trabalho possam gerar diversos tipos de dados a partir de uma versão do repositório, por padrão, apenas essas informações são registradas.

Por exemplo, um sistema pode ser compilado e gerar arquivos binários que normalmente não devem ser armazenados com o código-fonte, pois estão associados a apenas uma versão do sistema. Outras opções de armazenamento poderão ser utilizadas, caso seja necessário que estes e outros dados sejam armazenados para consultas futuras, mas não estarão vinculados diretamente a uma versão do sistema no repositório.

Links e referências adicionadas às mensagens dos *commits* podem apontar para a localização dos artefatos gerados, mas estes são gerenciados por outros sistemas e podem estar indisponíveis, mudar para referenciar outros recursos ou deixarem de existir.

Podemos tomar como exemplo um *commit* criado por Linus Torvalds em 2006 introduzindo uma correção na ferramenta Git. A mensagem descreve a alteração e fornece um link para o que parece ser a *thread* de discussão que contribuiu para essa correção. O problema encontrado por quem analisa esse *commit* e precisa de um contexto para entender a alteração, é que o link se encontra indisponível.

## 2 Objetivos

Será implementada uma ferramenta para estender a funcionalidade do Git e permitir que artefatos gerados por diferentes fluxos de trabalho possam ser arquivados em um repositório Git, separado dos arquivos que compõem os projetos. Além disso, esses artefatos estarão diretamente associados às alterações que os geraram e diretamente acessíveis no repositório.

Com essa nova funcionalidade, um *commit*, como o que foi tomado como exemplo, pode ter a *thread* de discussão e qualquer artefato que possa ajudar no entendimento da alteração como anexo. Assim, esses artefatos poderão ser acessados com a mesma facilidade que é realizada a leitura da mensagem de *commit*.

O repositório de desenvolvimento do próprio Git será usado para demonstrar a funcionalidade e verificar a capacidade de associar diferentes tipos de artefatos aos *commits*. Testes serão implementados para garantir que a solução proposta possa se recuperar de conflitos gerados pelo uso simultâneo por vários usuários.

### 3 Cronograma

	Junho	Julho	Agosto	Setembro
Estudo da Literatura				
Implementação				
Testes e Avaliação				
Escrita da Monografia				
Entrega da Monografia				
Preparação da Defesa				
Defesa				
Correção da Monografia				
Entrega da Monografia Corrigida				

## **4 Possíveis Avaliadores**

- Breno Miranda
- Henrique Rebelo

## 5 Referências

GIT. **About Version Control**. Disponível em: <https://git-scm.com/book/en/v2/Getting-Started-About-Version-Control>. Acesso em: 20 dez. 2022.

GIT. **Git Objects**. Disponível em: <https://git-scm.com/book/en/v2/Git-Internals-Git-Objects>. Acesso em: 20 dez. 2022.

GIT. **Recording Changes to the Repository**. Disponível em: <https://git-scm.com/book/en/v2/Git-Basics-Recording-Changes-to-the-Repository>. Acesso em: 20 dez. 2022.

TORVALDS, Linus. **Fix hash function in xdiff library**. Disponível em: <https://git.kernel.org/pub/scm/git/git.git/commit/?id=9de0834663f28bee9e6b2c4647ed6128241ed26f>. Acesso em: 26 dez. 2022.

GIT. **The core git plumbing**. Disponível em: <https://git.kernel.org/pub/scm/git/git.git>. Acesso em: 26 dez. 2022.