

DOMINANDO O CONTROLE DE VERSÃO DE CÓDIGO COM GIT



Thayana Corrêa Mamoré

SOBRE MIM



Apaixonada por Open Source

Desenvolvedora Web

Entusiasta de tecnologias web imersivas - WebXR
(Realidade Virtual e Realidade Aumentada na Web)

Organizadora da comunidade Devs Norte

Coordenadora de Projetos na Compass UOL

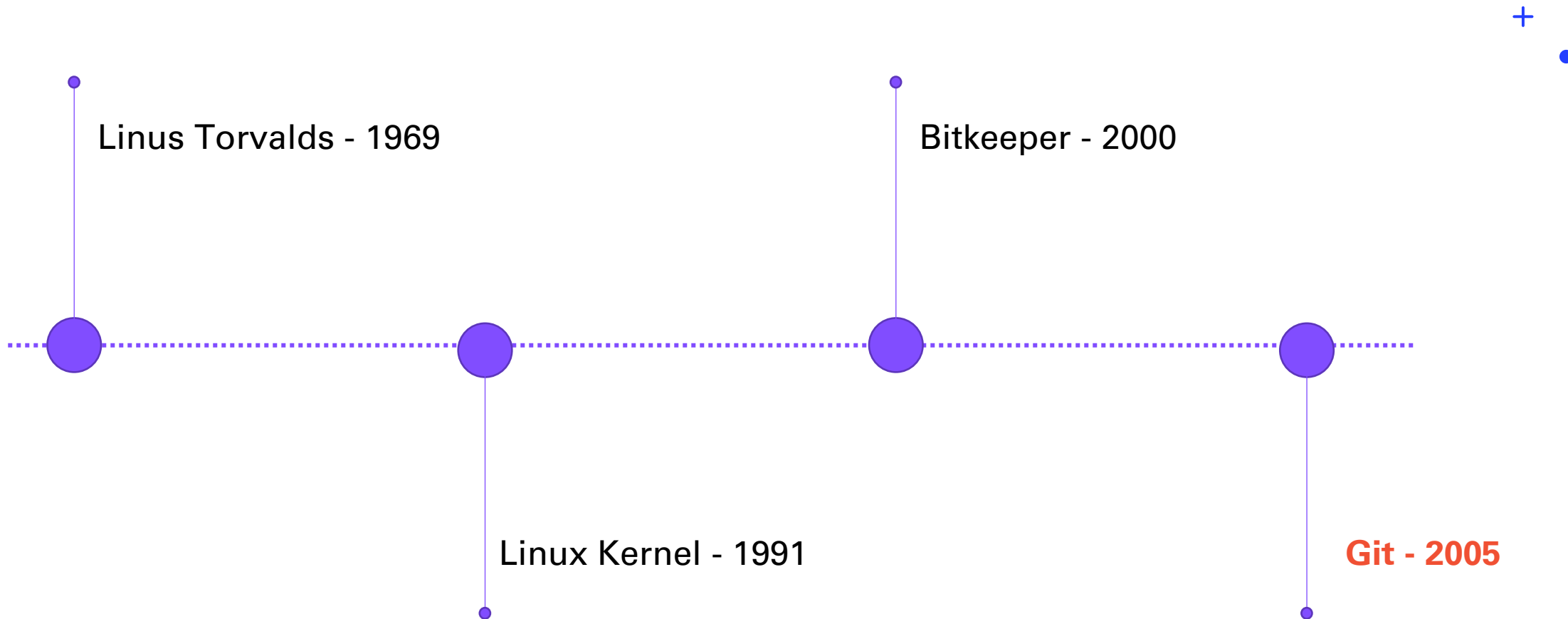
Sistema de controle de versão

Os sistemas de controle de versão são ferramentas de software que ajudam as equipes de software a gerenciar as alterações ao código-fonte ao longo do tempo.





git





O que é Git?

+

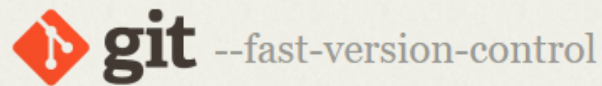
•

Sistema de controle de versão distribuído, de código aberto, projetada para lidar com tudo, desde pequenos a grandes projetos com velocidade e eficiência.



Criado em 2005 por **Linus Torvalds**, criador do **Linux**

<https://git-scm.com/downloads>



Search entire site...

About

Documentation

Downloads

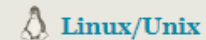
GUI Clients

Logos

Community

The entire [Pro Git book](#) written by Scott Chacon and Ben Straub is available to [read online for free](#). Dead tree versions are available on [Amazon.com](#).

Downloads



Latest source Release

2.37.0

[Release Notes](#) (2022-06-27)

[Download for Windows](#)

Older releases are available and the [Git source repository](#) is on GitHub.

GUI Clients

Git comes with built-in GUI tools ([git-gui](#), [gitk](#)), but there are several third-party tools for users looking for a platform-specific experience.

[View GUI Clients →](#)

Logos

Various Git logos in PNG (bitmap) and EPS (vector) formats are available for use in online and print projects.

[View Logos →](#)

Git via Git

If you already have Git installed, you can get the latest development version via Git itself:

```
git clone https://github.com/git/git
```

You can also always browse the current contents of the git repository using the [web interface](#).

A Linha de Comando

A linha de comando é o único lugar onde você pode rodar todos os comandos do Git - a maioria das GUI (interface gráfica de usuário) implementa somente um subconjunto das funcionalidades do Git. Além disso, enquanto a sua escolha da interface gráfica é uma questão de gosto pessoal, todos os usuário terão as ferramentas de linha de comando instaladas e disponíveis.

GUI



TortoiseGit
Windows Shell Interface to Git



Desktop



GitKraken

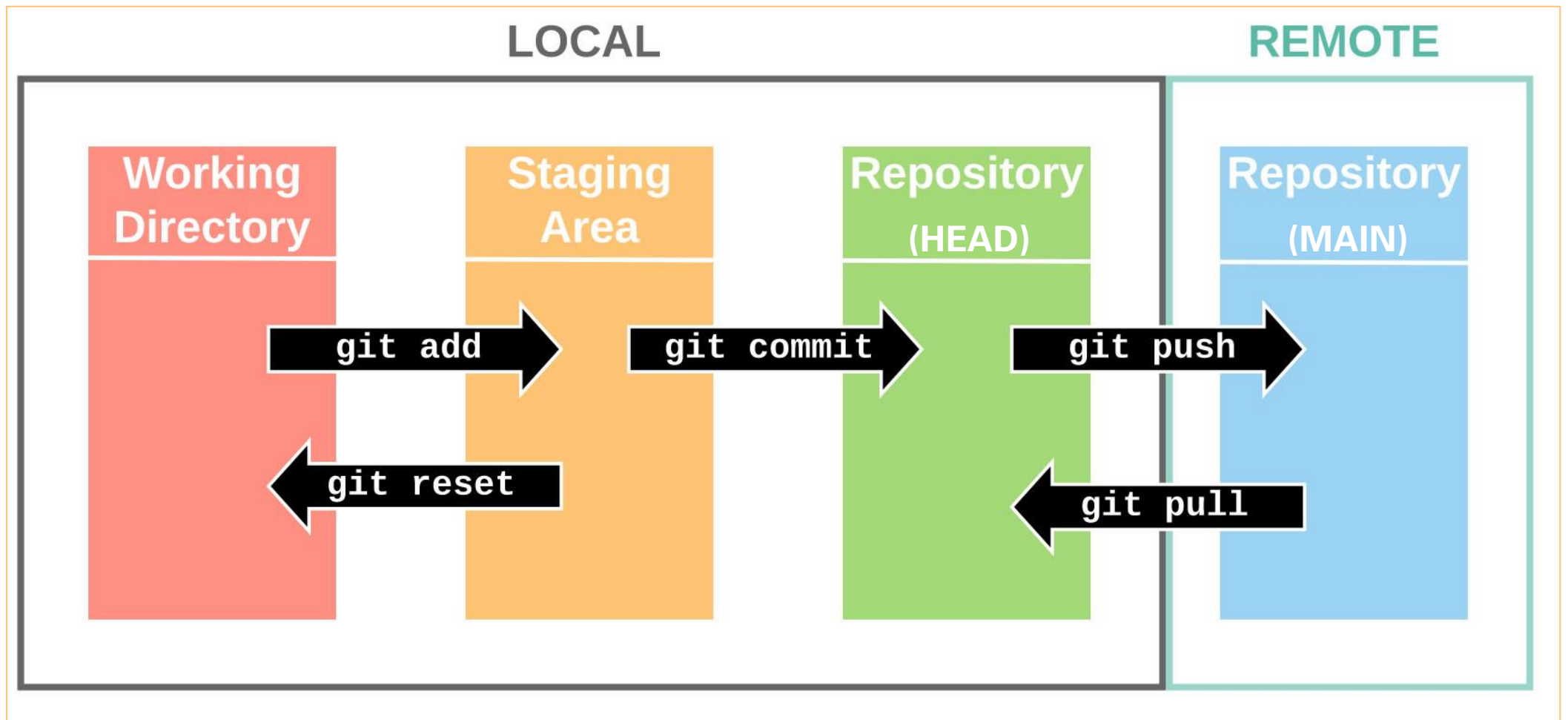
Definição e Conceitos Importantes

Repositório ou
“repo”



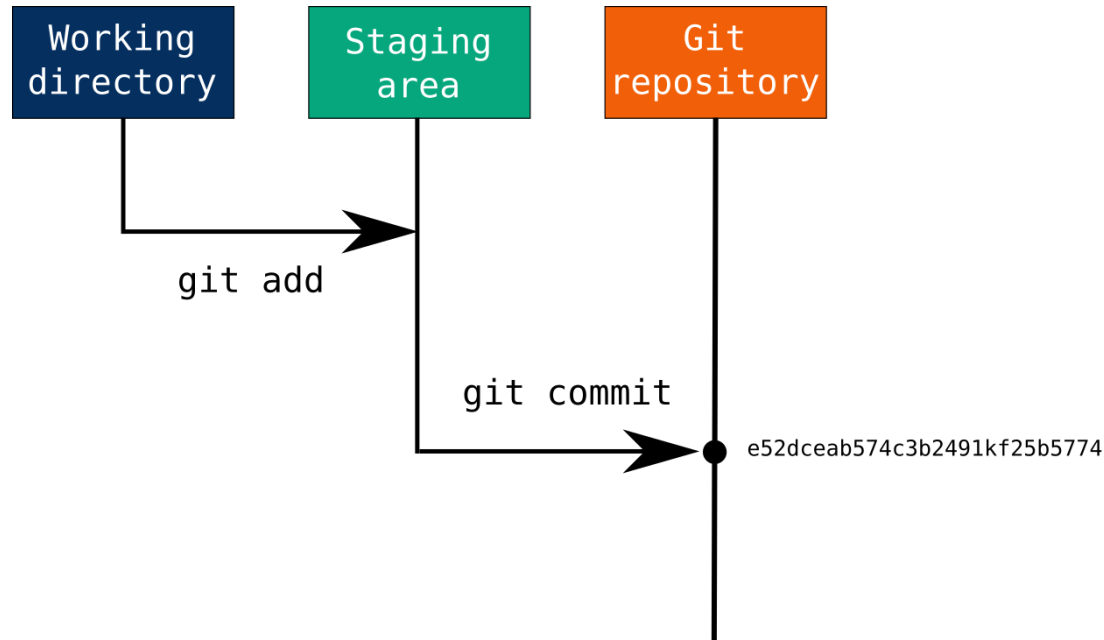
Fluxo de trabalho do Git

Diretório de trabalho | Área de "staging" | repositório local | repositório remoto



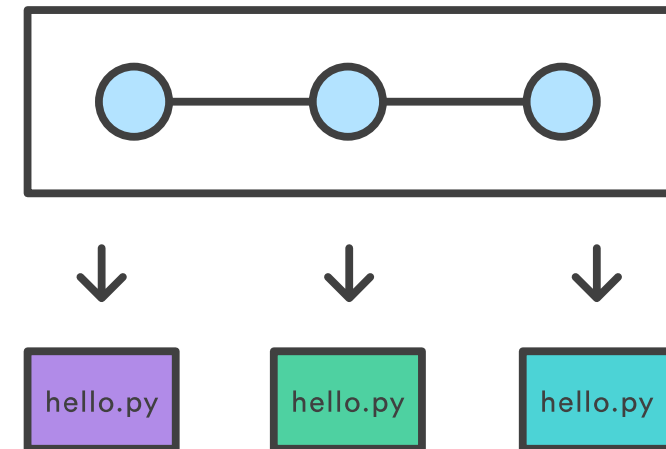
Definição e Conceitos Importantes

Add

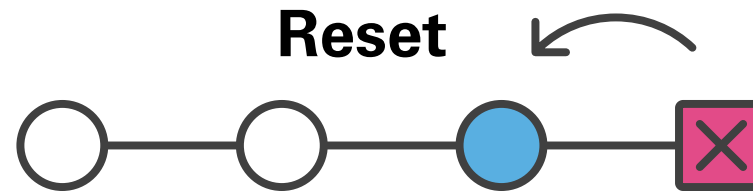


Commit

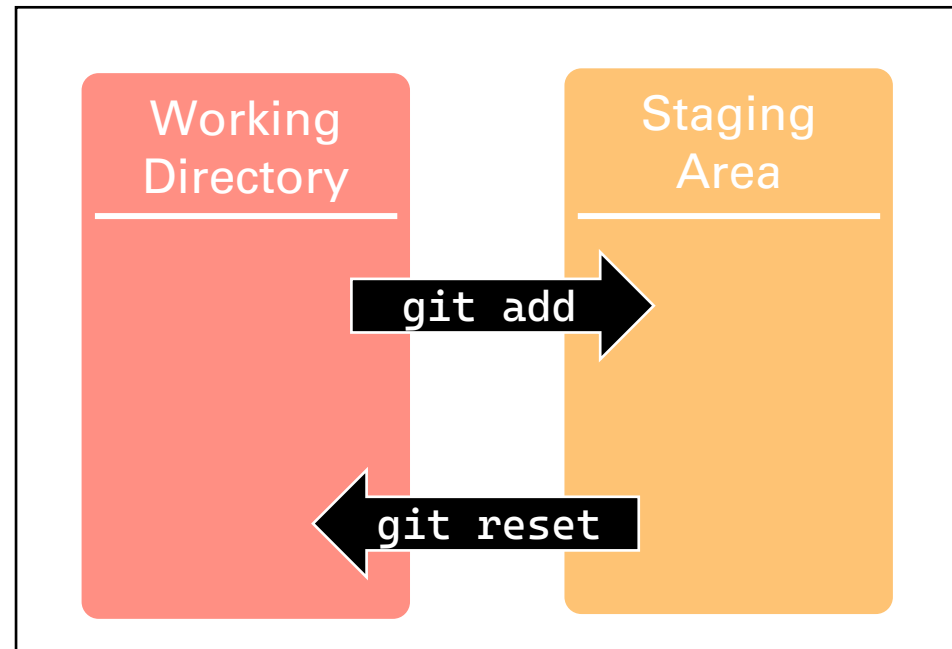
Recording Snapshots (Git)



Definição e Conceitos Importantes



LOCAL



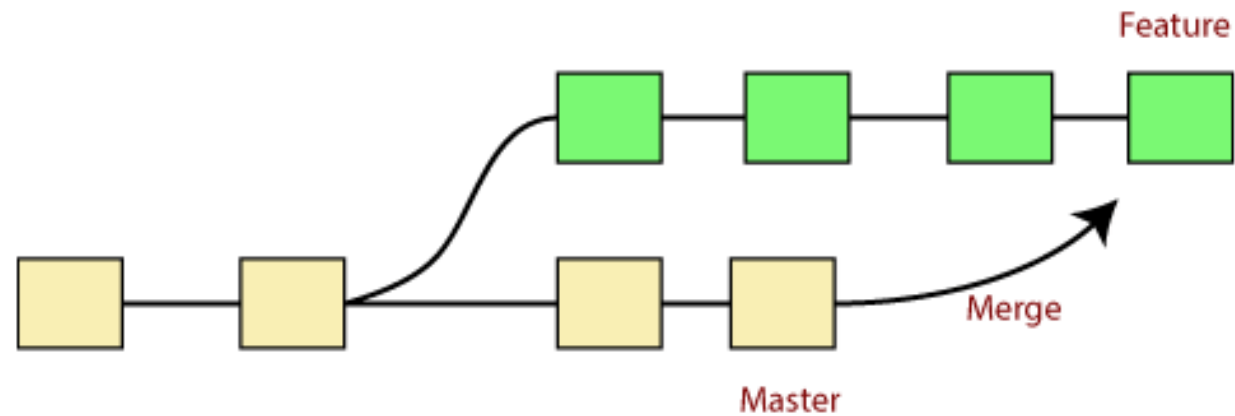
Definição e Conceitos Importantes

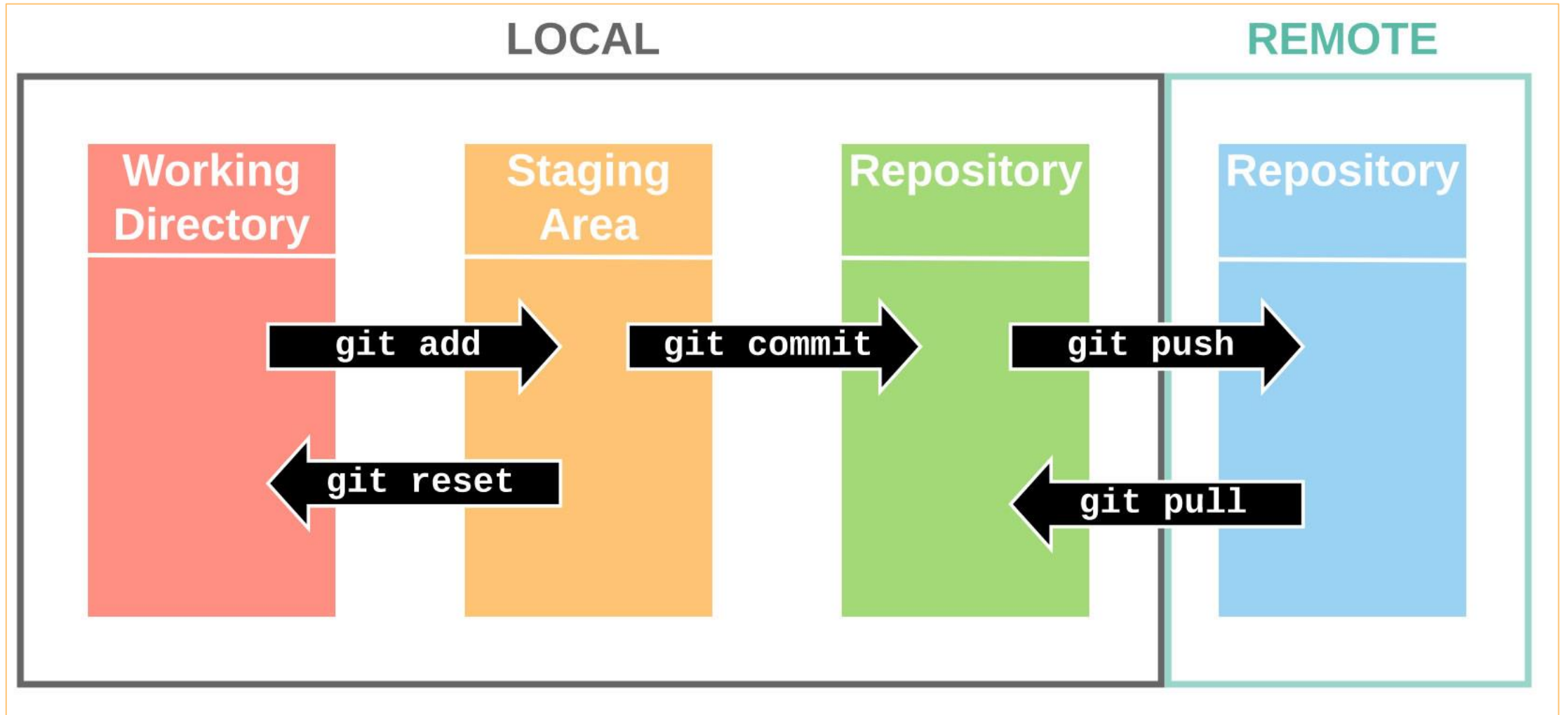
Branch



Definição e Conceitos Importantes

Merge





Definição e Conceitos Importantes

Push

Pull



Plataformas para armazenamento de código-fonte





1. It is a software

2. It is installed locally on the system

3. It is a command line tool

4. It is a tool to manage different versions of edits, made to files in a git repository

5. It provides functionalities like Version Control System Source Code Management

1. It is a service

2. It is hosted on Web

3. It provides a graphical interface

4. It is a space to upload a copy of the **Git** repository

5. It provides functionalities of Git like VCS, Source Code Management as well as adding few of its own features



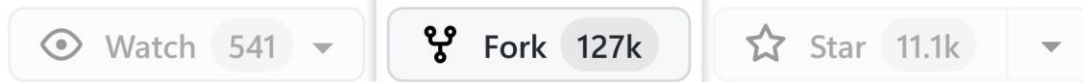
Definição e Conceitos Importantes

Fork

- Uma bifurcação é uma cópia de um repositório. Bifurcar um repositório permite que você faça experiências à vontade sem comprometer o projeto original.

Pull Request

- Propor alterações no projeto ou branch principal



MÃOS A OBRA

Comandos Bash

- `cd`
- `ls`
- `mkdir`
- `touch`
- `rm -rf`

Para lembrar

Git: configurações

- `git config --global user.name "Nome Sobrenome"`
- `git config --global user.email "seu-email@provedor.com"`
- `git config --global color.ui true`
- `git config --list`



Git: iniciando o repositório

- `git init`
- `git status`



Git: área de "staging"

- `git add <arquivo>`
- `git add -A`
- `git reset`



Git: commitando

- `git commit -m "comentario"`



Git: pull e push

- `git remote add origin <link>`
- `git push -u origin main`
- `git clone <clone>`
- `git pull`



Git: branching

- `git branch`
- `git branch <nome-branch>`
- `git switch -c <nome-branch>`
- `git switch <nome-branch>`
- `git merge <nome-branch>`
- `git branch -d <nome-branch>`

Referências

- <https://docs.github.com/pt/get-started>
- <http://djalma.blog.br/material-texto/git-book.pdf>
- <https://www.alura.com.br/artigos/o-que-e-git-github>
- <https://training.github.com/>
- <https://git-school.github.io/visualizing-git/>
- <https://www.conventionalcommits.org/pt-br/v1.0.0/>
- https://rogerdudler.github.io/git-guide/index.pt_BR.html
- <https://medium.com/@andgomes/git-github-evitando-informar-usu%C3%A1rio-e-senha-a-cada-push-para-o-github-d8edbb5c6de4>



OBRIGADA!

@thauska



@thayanacmamore



poly.me/thayanacmamore

