

GIT - Alguns Comandos

Profº. MSc. Adalto Selau Sparremerger



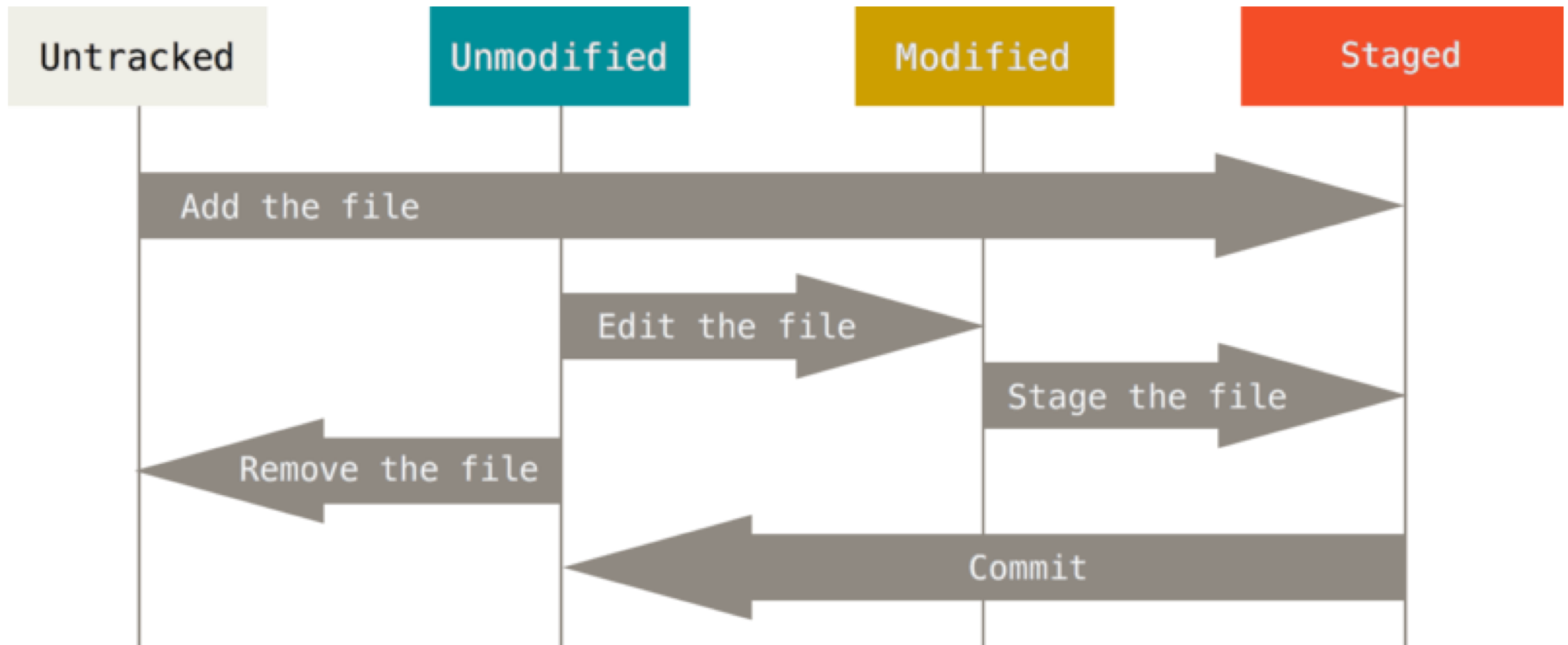
@adaltoss



/adaltoss1

/adaltofadergs

File Status



Fonte: <https://git-scm.com/book/en/v2/Git-Basics-Recording-Changes-to-the-Repository>

Comandos

git init

Inicializa o repositório

git add <nome_do_arquivo>

Adiciona um arquivo na visão do git para que possa ser comitado

git status

Serve para reportar como está meu repositório no momento

git commit [-m "Comentário"]

Pega todos os arquivos do repositório e cria um snapshot deles

Comandos

git log [--decorate] [--author="João"]

Mostra informações sobre os commits

git shortlog [-sn]

Mostra em ordem alfabética os autores e seus commits

git shortlog --graph

Mostra de forma gráfica o que está acontecendo com os branches e versões

git show <hash>

retorna o que foi realizado em um determinado commit

Comandos

git diff

Mostra as alterações antes de serem enviadas pelo commit

git diff --name-only

Mostra os nomes dos arquivos que foram alterados

git checkout <nome_do_arquivo>

desfaz as alterações no arquivo antes do commit

Comandos

git reset HEAD <nome_do_arquivo>

Retira da fila do staged

git reset --soft <hash>

Desfaz o commit e deixa o arquivo na fila do staged

git reset --mixed <hash>

Desfaz o commit e mantém o arquivo na fila de modified

git reset --hard <hash>

Desfaz o commit e todas alterações

Comandos

git branch

Lista os Branch

git checkout -b <nome_do_branch>

Cria um branch e entra nele

git branch -D <nome_do_branch>

Apaga um branch

Gitignore

```
vi .gitignore
```

```
*.xls
```

```
*.json
```

```
ola.txt
```

Arquivo utilizado para listar arquivos ou extensões a serem ignoradas no commit, no exemplo, todos arquivos json e xls e também o arquivo ola.txt serão ignorados

git stash

Guarda modificações que não sofreram commit, para que em um momento posterior possam ser "comitadas"

git stash apply

Aplica as mudanças que foram reservadas no stash

git stash list

Lista todos os stash

git stash clear

Exclui todos stash criados

Alias

```
git config --global alias.s status
```

A partir de então, para chamar o comando **git status** bastaria chamar:

```
git s
```

Tags

git tag -a 1.0.0 -m "Primeira versão finalizada"

Utilizada para "fechar versões" ou delimitar um grupo de commits

git push origin master --tags

Envia as tags para o remoto

git tag -d 1.0.0

Apaga uma tag do local

git push origin :1.0.0

Apaga uma tag do remoto

REVERT

git revert <hash>

Reverte as modificações de um commit, como se ele nunca estivesse existido, mas guarda para que depois possa ser reutilizado

Receita de Bolo...

Quando um novo projeto for criado:

- 1) **git init**
- 2) **git config --global http.proxy 0.0.0.0:0000** [ip:porta] (opcional se houver proxy)
- 3) **git remote add origin <https://exemplo.com/usuario/repositorio.git>**
- 4) **git add --all**
- 5) **git commit -m "Comentário com o resumo do que foi alterado"**
- 6) **git push -u origin master**

Ps: em destaque comandos que devem ser executados apenas na primeira vez

Receita de Bolo...

Nos demais commits:

- 1) `git add --all`
- 2) `git commit -m "Comentário com o resumo do que foi alterado"`
- 3) `git pull`
- 4) `git push`