



Introdução ao

git

Diogo Gomes <dgomes@ua.pt>

O que é o git ?

- Sistema de controlo de versões
- Criado em 2005 por Linus Torvalds
- Trata-se de um Sistema Distribuído de Controlo de Versões (DVCS)
 - Não existe um servidor central, todos programadores têm uma cópia de todas as alterações
 - Possibilita desenvolvimento completamente offline
- Seguro através do uso de SHA1 para criptograficamente fazer a rastreabilidade de todas as alterações.
- Flexível pois suporta mais do que um padrão de utilização (workflow)
- É o standard na industria, suportado por todas grandes empresas e pela comunidade OSS (github.com, gitlab.com, bitbucket.com)

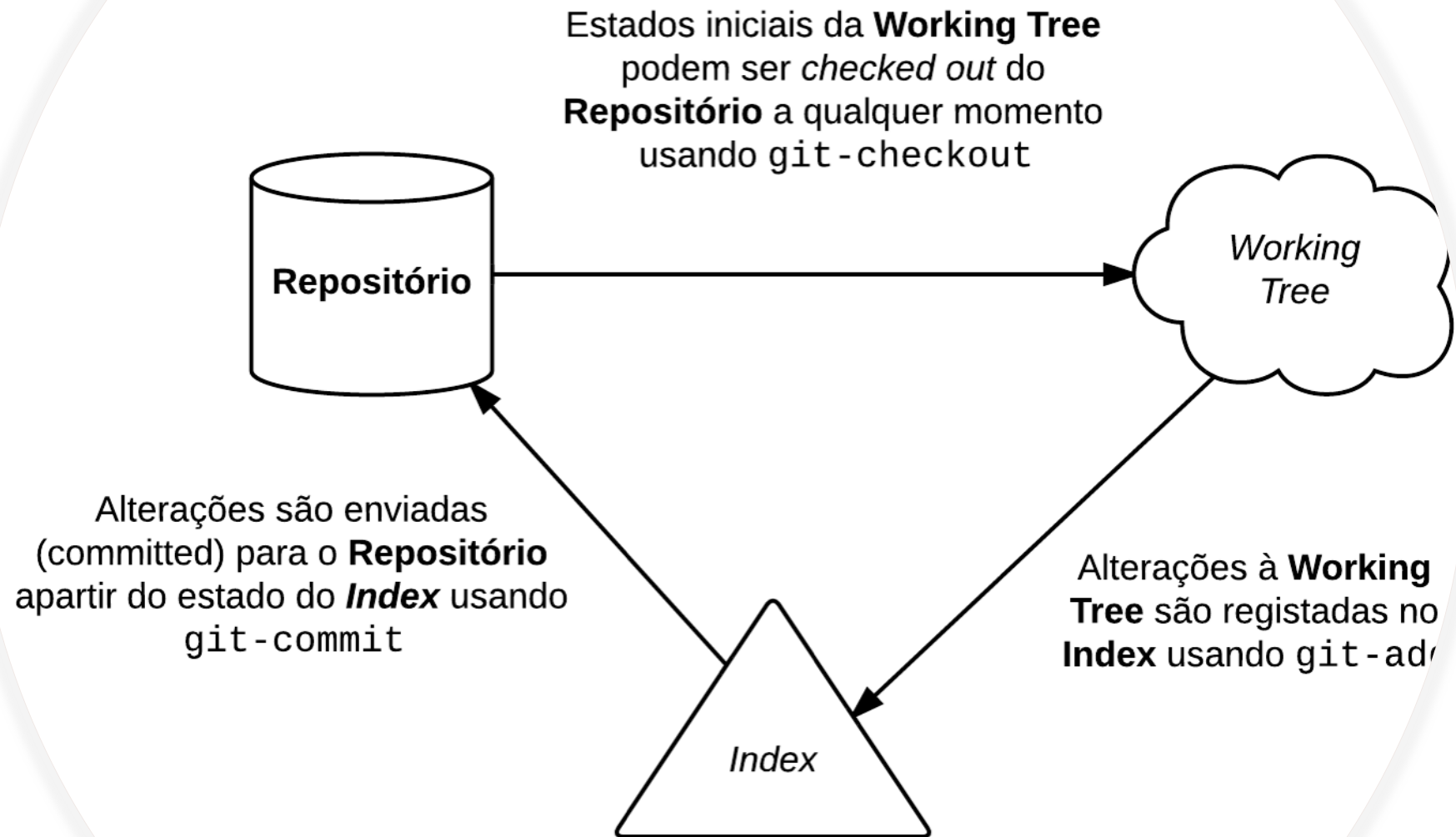
Criar repositório g i t

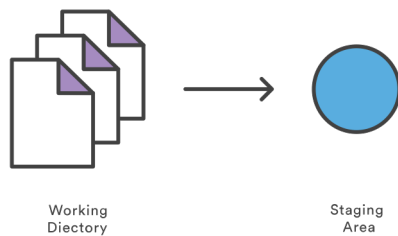
- Não precisamos de nenhum servidor! Podemos criar um repositório directamente a partir da linha de comandos:

```
% git init
```

- Mas também podemos criar o repositório online (ex. github.com) e clonar o repositório para o nosso computador.

```
% git clone git@github.com:dgomes/introducao_ao_git.git
```





Acrescentar um ficheiro

```
% git add nome_do_ficheiro
```

```
% git commit -m "mensagem"
```

- Ficheiro reside agora no nosso **repositório local**
- Podemos igualmente apagar ou mover um ficheiro:

```
% git rm nome_do_ficheiro
```

```
% git mv nome_do_ficheiro
```

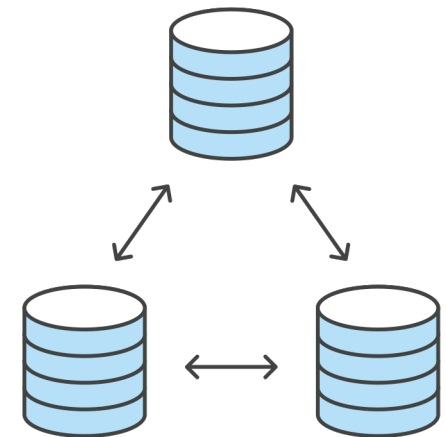
Sincronizar com servidor remoto

```
% git remote add origin git@github.com:dgomes/introducao_ao_git.git
```

```
% git push -u origin master
```

- O repositório “central” é normalmente apelidado de *origin*
- Podemos acrescentar mais do que um repositório *remote*
- Antes de um push num repositório partilhado é importante sincronizar o repositório local e resolver qualquer conflito:

```
% git pull
```



Conhecer o estado da *working tree*

`% git status`

- Informa que ficheiros precisam ser adicionados (porque foram alterados)
- Informa que ficheiros existem no computador mas que não pertencem ao repositório

`% git log`

- Informa de todas as alterações feitas até ao momento (HEAD)
- Informa de commits anteriores e qual o seu identificador (SHA1)

Conhecer o que foi feito

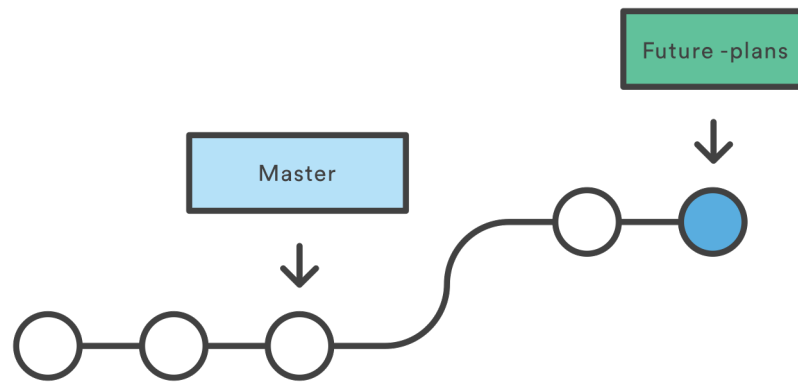
```
% git diff commit_id
```

- Mostra-nos as diferenças entre a *HEAD* e o *commit_id*

```
% git show commit_id
```

- Mostra-nos o que foi submetido em *commit_id*

Branching



- `% git checkout -b future-plans`
- Inicia um novo branch e coloca-nos no mesmo
- `% git branch`
- Lista os branches e informa do actual

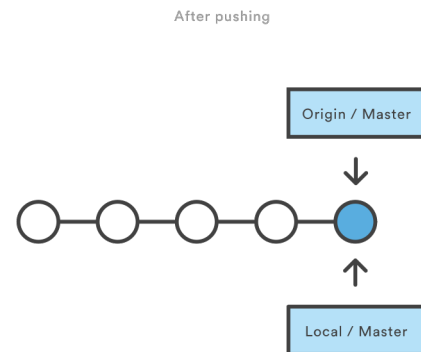
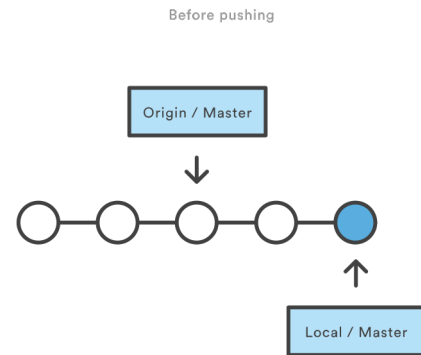
Merging

```
% git merge future-plans
```

- Como não existem histórias paralelas o merge é apenas a actualização do HEAD de master

```
% git branch -d future-plans
```

- Para apagar um branch que já foi merged / não interessa



Push

```
% git push origin master
```

- Envia as nossas alterações feitas a *master* para o remote *origin*

Undo! Undo!

```
% git revert commit_id
```

- Cria um novo commit que desfaz as alterações de `commit_id` e aplica ao branch actual

```
% git reset --hard commit_id
```

- Desfaz todas alterações até ao `commit_id`, apaga todas alterações (sem `--hard`, as alterações mantêm-se em disco)

Ups! Merge falhou

- Por vezes durante um merge podemos ter conflitos (alterações concorrentes no mesmo pedaço de código)

- % git status

- Saber que ficheiros precisam de ser resolvidos

- Editar os ficheiros e procurar por "<<< === >>>"

- Escolher qual a versão correcta (do branch actual ou do branch que está a ser merged)

- % git add ficheiro

- % git commit -m "corrigir conflitos"

- Por favor ajudem! Não sei resolver os conflitos! Deixem-me voltar!

- % git merge -abort

- Oh não! Fiz merge e agora isto está tudo mal!

- % git reset --hard

- Dicas:
 - Configurar sempre uma chave SSH
 - <https://help.github.com/en/github/using-git/which-remote-url-should-i-use#cloning-with-ssh-urls>
 - Configurar endereço email correcto (não usar o do computador)
 - <https://help.github.com/en/github/using-git/setting-your-username-in-git>
 - Evitar editar ficheiros online (maior propensão para criar conflitos)



Obrigado pela vossa atenção!

Algumas das imagens são retiradas de
<https://www.atlassian.com/git/tutorials/learn-git-with-bitbucket-cloud>