



Git Advanced

Leandro Acosta, Mateo Ostorero, Gonzalo Gigena y Joaquin de Francesca



Estructura de la presentación

1. Introducción
 - a. ¿Qué es?
 - b. Historia
 - c. Diseño
 - d. Nombre
2. Workflow de Git
 - a. Áreas de trabajo
 - b. Ciclos de vida de los archivos
3. Git Internals
 - a. Comandos de Git (Plumbing y Porcelain)
 - b. Git Objects
 - c. Git Reference
4. Demo práctica



Introducción

¿Qué es?

Es el sistema de control de versiones más utilizado en el mundo. Git es un proyecto de código abierto con un mantenimiento activo, originalmente desarrollado por Linus Torvalds y actualmente mantenido por Junio Hamano.





Historia

- 2002 linux kernel empieza a usar Bitkeeper
- 2005 empieza el desarrollar Git
- Primera release 2005-07-11



Junio Hamano



Linus torvalds



Diseño

- Inspirado por BitKeeper y Monotone
- Cogito or StGIT



¿Por qué se llama GIT?

"Global information tracker"

Linus torvals confirmó que está nombrado por la palabra "git". En la jerga inglesa "git" es un adjetivo para describir a una persona tonta o desagradable.

"I name all my projects after myself. First Linux, now git."

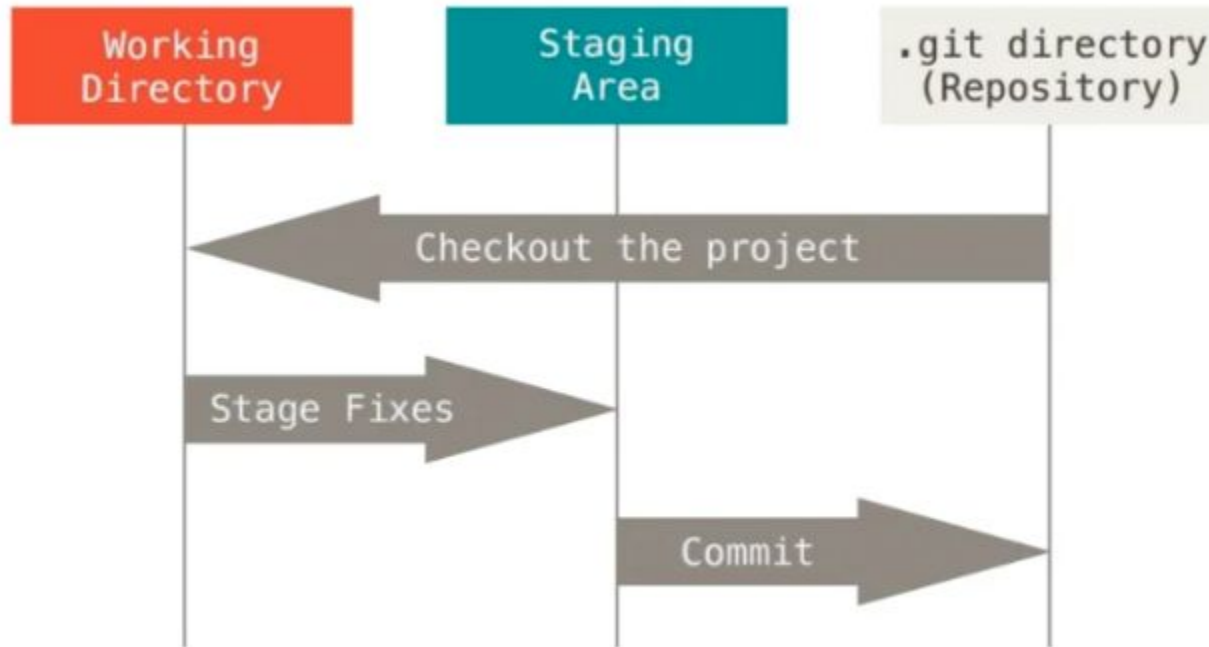
"The information manager from hell"



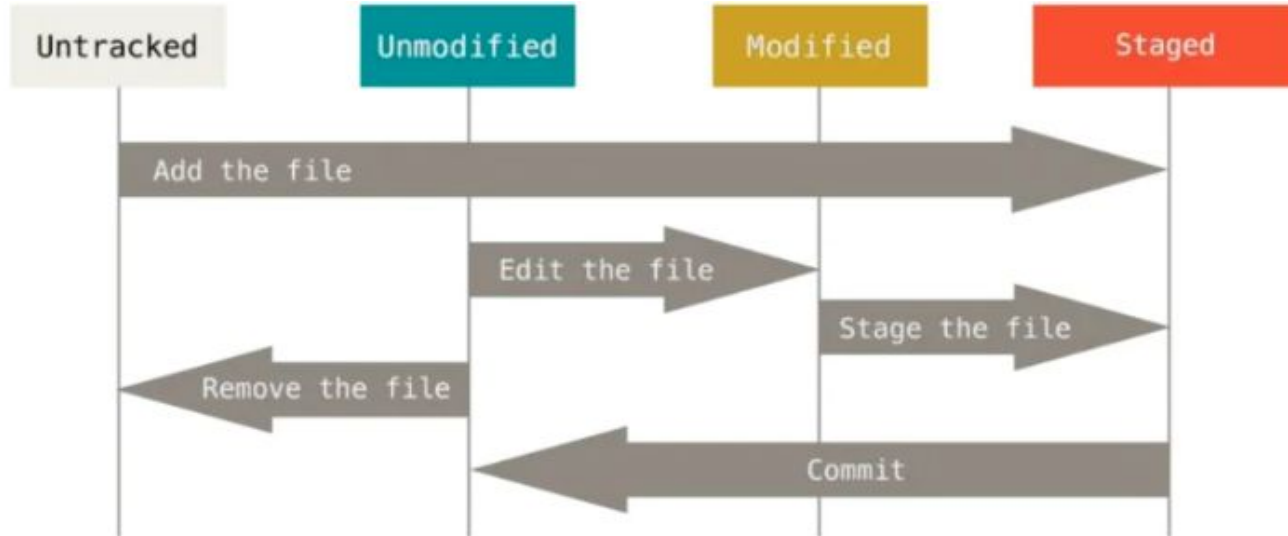


Workflow de Git

Áreas de trabajo



Ciclo de vida de los archivos





Git Internals

Comandos de Git

- Porcelain (orientados a usuario)
 - 50 comandos
 - Con autocompletado en la Cli de Git
- Plumbing (núcleo de Git)
 - > 60 comandos
 - Sin autocompletado



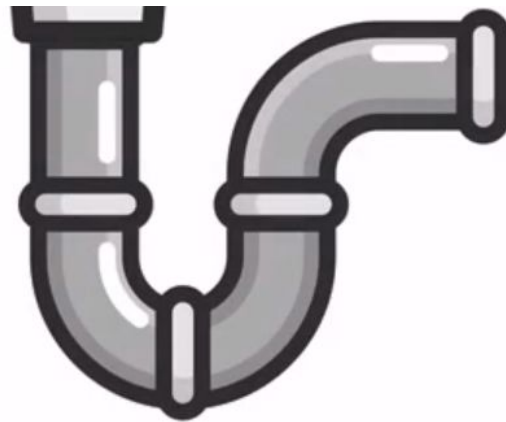
Porcelain

- `git init`
- `git add`
- `git commit`
- `git checkout`
- `git status`
- `git merge`
- ... y muchos más



Plumbing

- `git hash-object`
- `git cat-file`
- `git update-index`
- `git ls-tree`
- `git read-tree`
- ... y muchos más



Git Objects

1. Blob (Binary large object)
2. Tree
3. Commit

5b1d3..

blob	size
<pre>#ifndef REVISION_H #define REVISION_H #include "parse-options.h" #define SEEN (1u<<0) #define UNINTERESTING (1u #define TREESAME (1u<<2)</pre>	

c36d4..

tree	size	
blob	5b1d3	README
tree	03e78	lib
tree	cdc8b	test
blob	cba0a	test.rb
blob	911e7	xdiff

ae668..

commit	size
tree	c4ec5
parent	a149e
author	Scott
committer	Scott
my commit message goes here and it is really, really cool	



Estructura de carpetas

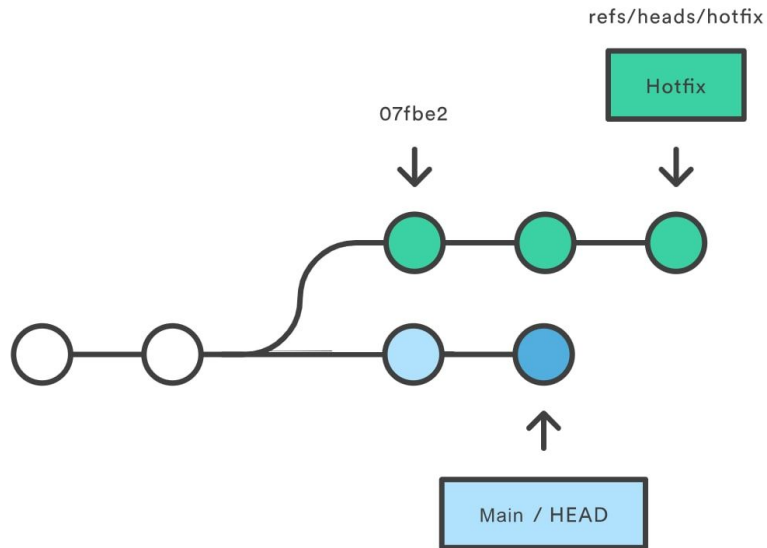
```
├── README.md
├── requirements.txt
└── src
    └── main.py

1 directory, 3 files
```


Git Reference

Una referencia es una forma indirecta de referirse a un **Commit**.

Puede considerarse como un alias fácil de usar para un hash de un commit. Este es el mecanismo interno de Git para representar **branches** y **tags**





Demo