



Entendamos Git

En el engranaje del control de código fuente



Michael Kaufmann

DevOps Consultant & Trainer

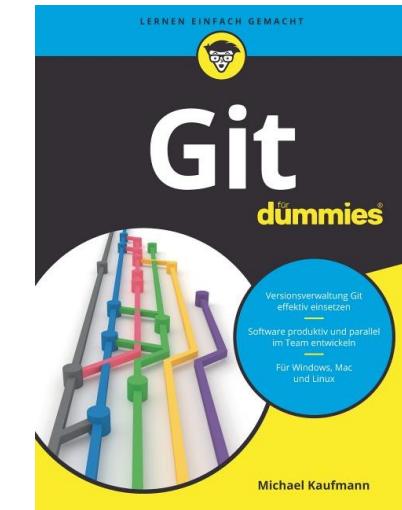
20 años Desarrollador de software
15 años ALM & DevOps
8 años Git
Microsoft Regional Director & MVP

@mike_kaufmann

@wulfland

<https://writeabout.net>

Git für Dummies, Michael Kaufmann, Wiley-VCH, 2021



THIS IS GIT. IT TRACKS COLLABORATIVE WORK
ON PROJECTS THROUGH A BEAUTIFUL
DISTRIBUTED GRAPH THEORY TREE MODEL.

COOL. HOW DO WE USE IT?

NO IDEA. JUST MEMORIZIZE THESE SHELL
COMMANDS AND TYPE THEM TO SYNC UP.
IF YOU GET ERRORS, SAVE YOUR WORK
ELSEWHERE, DELETE THE PROJECT,
AND DOWNLOAD A FRESH COPY.

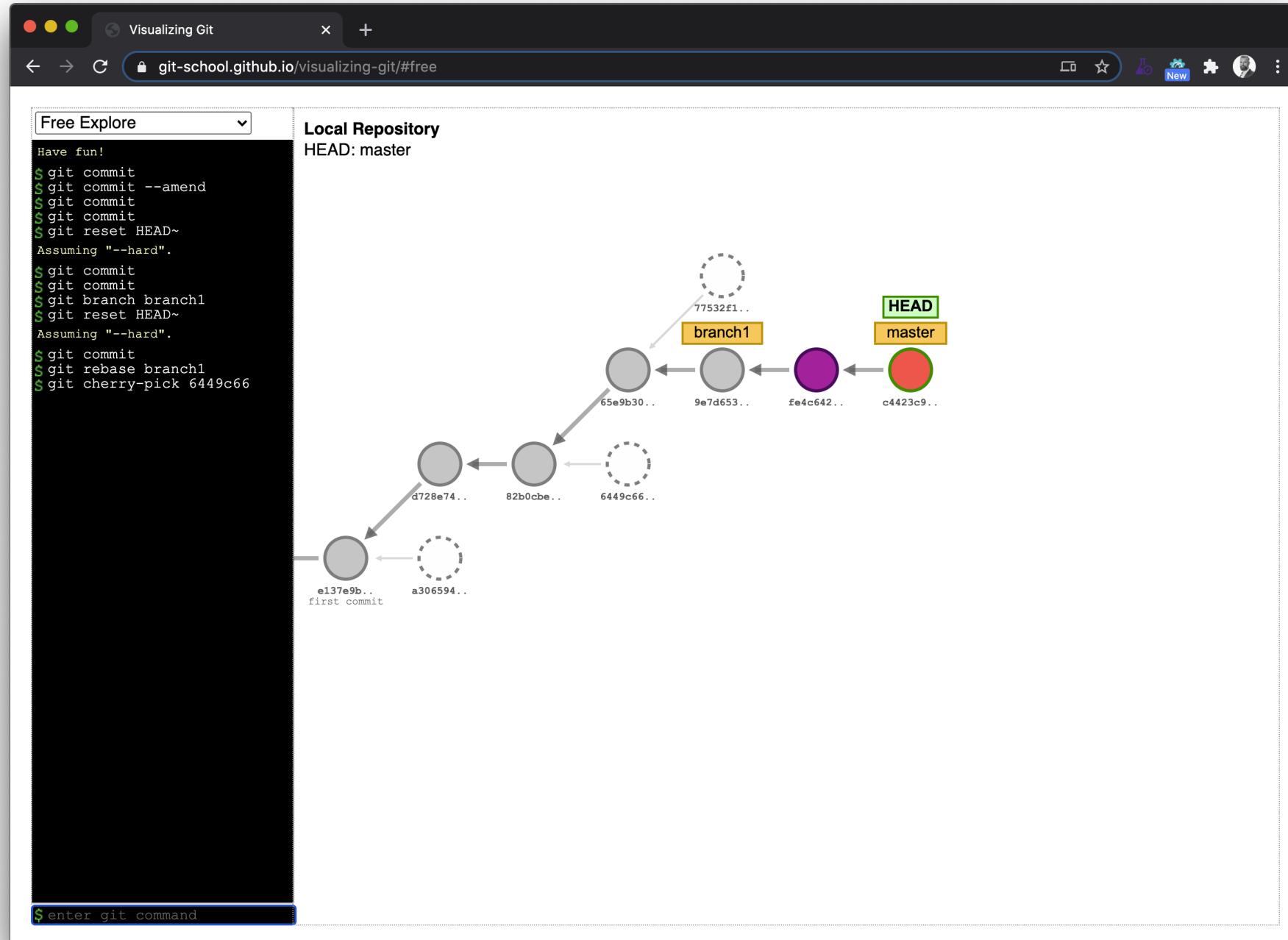


¿Qué es Git?

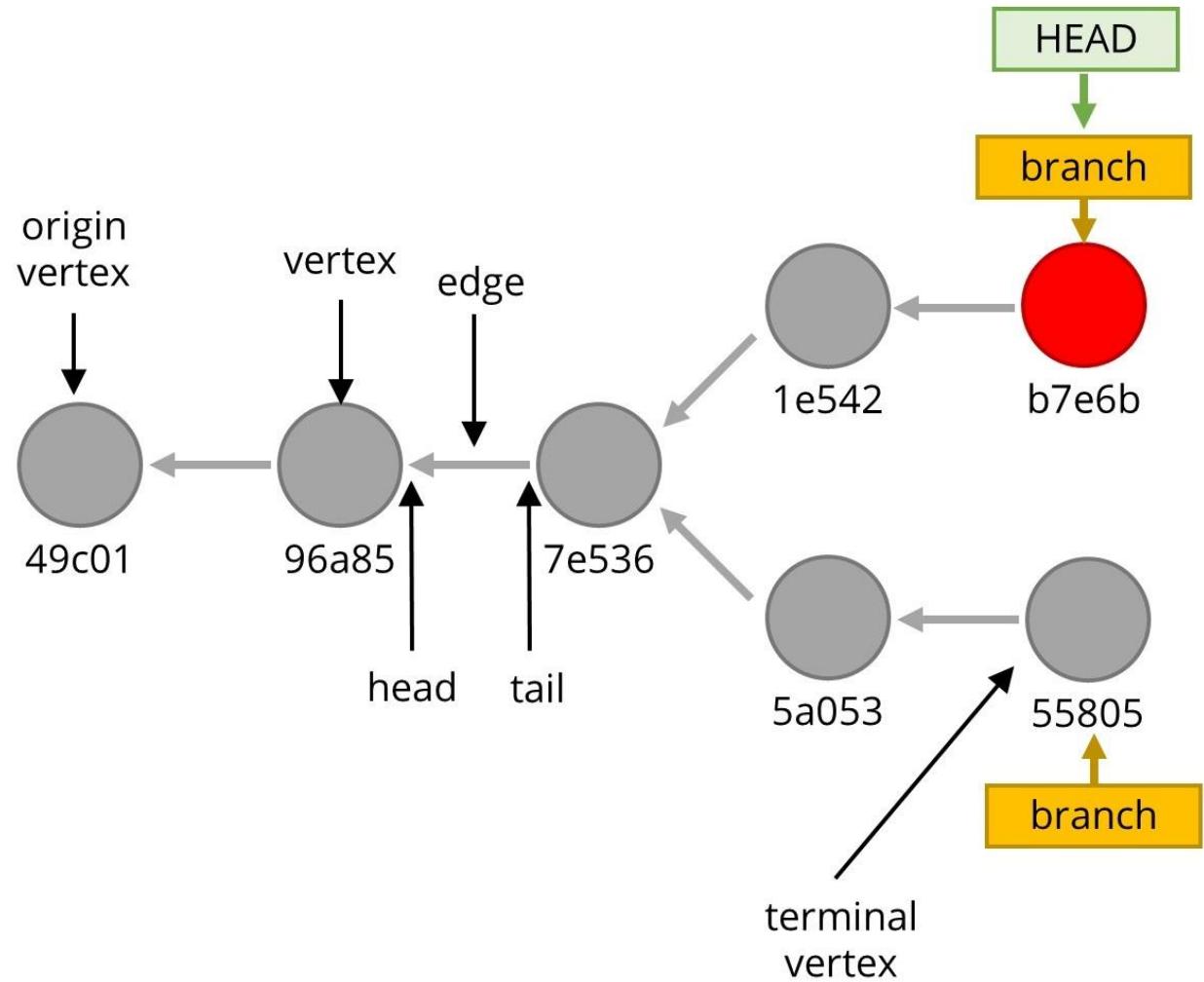
«Esto es Git. Ayuda al seguimiento del trabajo colaborativo en proyectos a través del uso de un bello modelo de árbol distribuido de la teoría de grafos.»

«¡Qué interesante! ¿Cómo se utiliza?»

«Ni idea. Apréndete simplemente los siguientes comandos e introdúcelos en la línea de comandos para sincronizar tu trabajo. Si te da error, guarda tu trabajo en otro sitio, borra el proyecto y descárgate una nueva copia.»



El Grafo Acíclico Dirigido (DAG)



El HASH

Un *HASH* es un valor de comprobación criptográfico para la verificación de la integridad de datos, por ejemplo, de archivos. Con la comparación del valor *HASH* puede establecerse si se ha modificado un archivo. Un archivo siempre da como resultado el mismo *HASH* si no ha sido modificado.

El SHA-1

El *SHA-1* es un ***Secure Hash Algorithm***, un grupo de funciones Hash estándares y criptológicas. Sirven para calcular los valores de comprobación para datos digitales. El SHA original fue ya corregido en 1995 debido a un “fallo de construcción” y, por ello, ya no desempeña ningún papel. Hoy en día es conocido como SHA-0 y la variante corregida como SHA-1.

El SHA-256

Con la actual versión de Git (2.29) uno puede decidir en primer lugar al crear un nuevo Repository si quiere iniciar el Repository con SHA-1 o con SHA 256:

```
$ git init --object-format=sha256
```

A diferencia de un valor de comprobación SHA-1, el SHA-256 se compone de 64 caracteres en vez de 40.

¿Qué es un
Commit?





Advertencia

¡Muchos de los commandos aquí
no están concebidos para el uso
diario!



Demo: Objects, Trees and Commits

..UnterDerHaube (-zsh)

```

apple ~ % source/repos/UnterDerHaube
> git init
Initialized empty Git repository in /Users/m.kaufmann/source/repos/UnterDerHaube/.git/
apple ~ % source/repos/UnterDerHaube git <main ?1>
> git hash-object txt/Doc.txt
f20acb769ad757b2058c8c3c96e7eac2e1258503
apple ~ % source/repos/UnterDerHaube git <main ?1>
> vim txt/Doc.txt
apple ~ % source/repos/UnterDerHaube git <main ?1>
14s ✘ 10:58:21 ⓘ
> git hash-object txt/Doc.txt
f85de0e273f9ed77d2a37b1dd7397e667b0ad335
apple ~ % source/repos/UnterDerHaube git <main ?1>
> git add txt/Doc.txt
apple ~ % source/repos/UnterDerHaube git <main +1>
10:59:03 ⓘ
> cat .git/HEAD
ref: refs/heads/main
apple ~ % source/repos/UnterDerHaube git <main +1>
11:00:27 ⓘ
> cat .git/Index
DIRC_0*00
0_*0g000000>0]00s00wH{09~f{
05
txt/Doc.txt0({0700030>0pC0k0J0%
apple ~ % source/repos/UnterDerHaube git <main +1>
11:00:43 ⓘ
> git ls-files --stage
100644 f85de0e273f9ed77d2a37b1dd7397e667b0ad335 0      txt/Doc.txt
apple ~ % source/repos/UnterDerHaube git <main +1>
11:01:05 ⓘ
> cat .git/objects/f8/5de0e273f9ed77d2a37b1dd7397e667b0ad335
xK00R03b0K0M0Rp0Sp00M-QPr00VRp*0/000/0L00K0Q(0)2000000L0â000000%
apple ~ % source/repos/UnterDerHaube git <main +1>
11:03:05 ⓘ
> git cat-file -t f85de
blob
apple ~ % source/repos/UnterDerHaube git <main +1>
11:05:52 ⓘ
> git cat-file -p f85de
Name: Dr. Emmet "Doc" Brown
Original time: 1985
Nickname: Doc
apple ~ % source/repos/UnterDerHaube git <main +1>
11:05:57 ⓘ

```

10:56:10 ⓘ

10:56:36 ⓘ

10:57:58 ⓘ

10:58:21 ⓘ

10:58:24 ⓘ

10:59:03 ⓘ

11:00:27 ⓘ

11:00:43 ⓘ

11:01:05 ⓘ

11:03:05 ⓘ

11:05:52 ⓘ

11:05:57 ⓘ

Andoria.local: Wed Nov 25 11:06:00 2020

```

.
+-- .git/
    +-- HEAD
    +-- config
    +-- description
    +-- index
    +-- info/
    +-- exclude
    +-- objects/
        +-- f8/
            +-- 5de0e273f9ed77d2a37b1dd7397e667b
    +-- info/
    +-- pack/
    +-- refs/
        +-- heads/
        +-- tags/
    +-- txt/
        +-- Doc.txt

```

0ad335

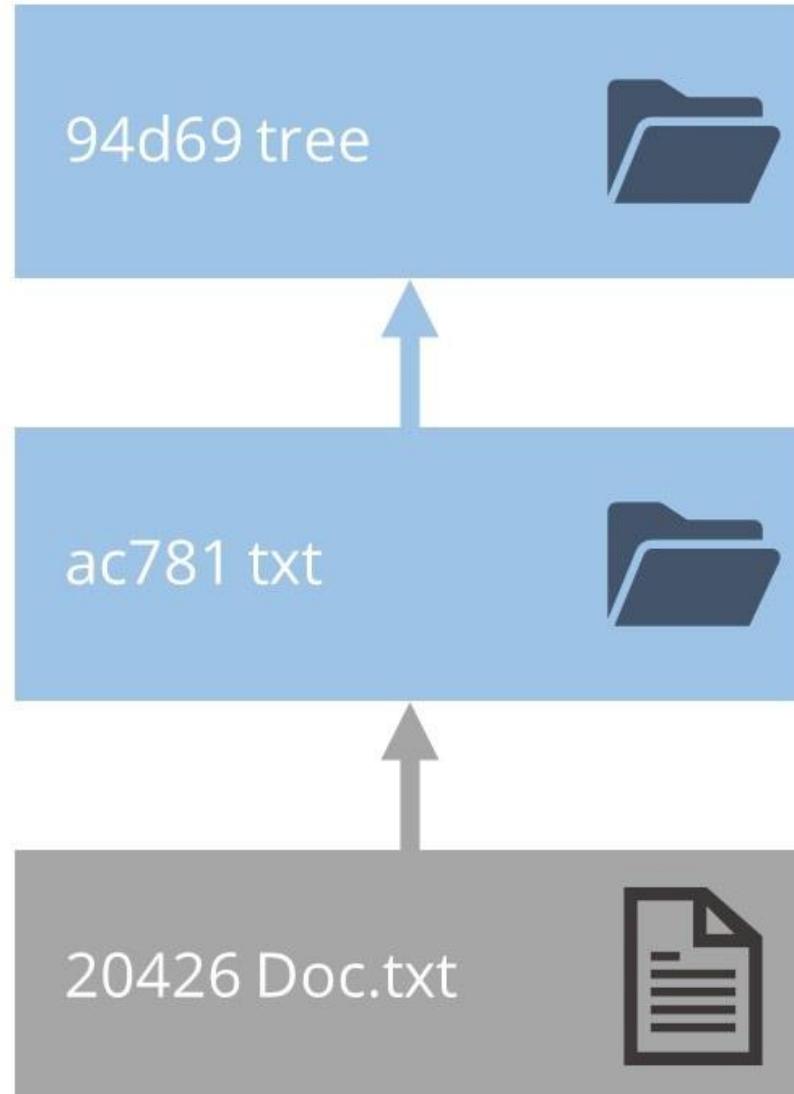
10 directories, 7 files

```
..UnterDerHaube (-zsh) 11:09:37 ⓘ watch (watch) Every 2.0s... Andoria.local: Wed Nov 25 11:13:11 2020
❯ git cat-file -t fa79
commit
Apple ~ /source/repos/UnterDerHaube git ⚡ main 11:09:42 ⓘ
❯ git cat-file -p fa79
tree 70f8d78e3776ac011b60926fc52e30d76789de59
author Michael Kaufmann <wulfland@hotmail.com> 1606298912 +0100
committer Michael Kaufmann <wulfland@hotmail.com> 1606298912 +0100

ErsterCommit 11:09:46 ⓘ
Apple ~ /source/repos/UnterDerHaube git ⚡ main
❯ git cat-file -t 70f8
tree
Apple ~ /source/repos/UnterDerHaube git ⚡ main 11:10:57 ⓘ
❯ git cat-file -p 70f8
040000 tree ba105a49862bcf36b8ffff1dae703fc5678f7358a txt
Apple ~ /source/repos/UnterDerHaube git ⚡ main 11:11:01 ⓘ
❯ git ls-tree ba105
100644 blob f85de0e273f9ed77d2a37b1dd7397e667b0ad335 Doc.txt
Apple ~ /source/repos/UnterDerHaube git ⚡ main 11:11:19 ⓘ
❯ cat .git/refs/heads/main
fa794f016773b66bed29596a0cec389295a5d758
Apple ~ /source/repos/UnterDerHaube git ⚡ main 11:12:48 ⓘ
❯

.git/
  COMMIT_EDITMSG
  HEAD
  MERGE_RR
  config
  description
  index
  info/
    exclude
  logs/
    HEAD
    refs/
      heads/
        main
  objects/
    70/
      f8d78e3776ac011b60926fc52e30d76789de59
    ba/
      105a49862bcf36b8ffff1dae703fc5678f7358a
    f8/
      5de0e273f9ed77d2a37b1dd7397e667b0ad335
    fa/
      794f016773b66bed29596a0cec389295a5d758
    info/
    pack/
  refs/
    heads/
      main
    tags/
    rr-cache/
  txt/
    Doc.txt

17 directories, 15 files
```



La estructura del árbol (tree)

```
..UnterDerHaube (-zsh)
apple ~ % source/repos/UnterDerHaube git ⌂ main
> git cat-file -p c565
tree 7cf2180461d7d876ece3c340606339a4f4137e77
parent fa794f016773b66bed29596a0cec389295a5d758
author Michael Kaufmann <wulfland@hotmail.com> 1606299443 +0100
committer Michael Kaufmann <wulfland@hotmail.com> 1606299443 +0100

ZweiterCommit
apple ~ % source/repos/UnterDerHaube git ⌂ main
> git ls-tree 7cf2
040000 tree 832b0d1f161855de45a37b3d21ae71179c8a1011    txt
apple ~ % source/repos/UnterDerHaube git ⌂ main
> git ls-tree 832b
100644 blob c30cfb9381a9cf06cf3455aee11d8bcff65c1478   Doc.txt
apple ~ % source/repos/UnterDerHaube git ⌂ main
> git cat-file -p c30cf
Name: Dr. Emmet "Doc" Brown
Original time: 1985
Nickname: Doc
Lala
apple ~ % source/repos/UnterDerHaube git ⌂ main
>

11:20:24 ⓘ
11:20:31 ⓘ
11:20:46 ⓘ
11:20:51 ⓘ
11:21:04 ⓘ
```

Every 2.0s... Andoria.local: Wed Nov 25 11:21:08 2020

```
.git/
  COMMIT_EDITMSG
  HEAD
  MERGE_RR
  config
  description
  index
  info/
    exclude
  logs/
    HEAD
    refs/
      heads/
        main
  objects/
    70/
      f8d78e3776ac011b60926fc52e30d76789de59
    7c/
      f2180461d7d876ece3c340606339a4f4137e77
    83/
      2b0d1f161855de45a37b3d21ae71179c8a1011
    ba/
      105a49862bcf36b8fff1dae703fc5678f7358a
    c3/
      0cfb9381a9cf06cf3455aee11d8bcff65c1478
    c5/
      6593411572c162114cffcf244169ac62e8b3d70
    f8/
      5de0e273f9ed77d2a37b1dd7397e667b0ad335
    fa/
      794f016773b66bed29596a0cec389295a5d758
    info/
    pack/
  refs/
    heads/
      main
    tags/
```

```
..UnterDerHaube (-zsh)
apple ~ % source/repos/UnterDerHaube git my-branch
> git log -1
commit cbe2ee3d98c4b283c97f749684d433375e325b6a (HEAD -> my-branch)
Author: Michael Kaufmann <wulfland@hotmail.com>
Date:   Wed Nov 25 11:37:40 2020 +0100

Test
apple ~ % source/repos/UnterDerHaube git my-branch
> git tag tag1
apple ~ % source/repos/UnterDerHaube git my-branch
> cat .git/refs/tags/tag1
cbe2ee3d98c4b283c97f749684d433375e325b6a
apple ~ % source/repos/UnterDerHaube git my-branch
> git checkout tag1
Note: switching to 'tag1'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by switching back to a branch.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -c with the switch command. Example:

git switch -c <new-branch-name>

Or undo this operation with:

git switch -

Turn off this advice by setting config variable advice.detachedHead to false

HEAD is now at cbe2ee3 Test
apple ~ % source/repos/UnterDerHaube git #tag1
> cat .git/HEAD
cbe2ee3d98c4b283c97f749684d433375e325b6a
apple ~ % source/repos/UnterDerHaube git #tag1
> 11:38:26 ⓘ
11:38:35 ⓘ
11:38:39 ⓘ
11:38:54 ⓘ
11:39:35 ⓘ
11:39:53 ⓘ
Every 2.0s... Andoria.local: Wed Nov 25 11:39:56 2020
.
  .git/
    -- COMMIT_EDITMSG
    -- HEAD
    -- ORIG_HEAD
    -- config
    -- description
    -- index
    -- info/
      -- exclude
      -- refs
    -- logs/
      -- HEAD
      -- refs/
        -- heads/
          -- main
          -- my-branch
      -- objects/
        -- info/
          -- packs
        -- pack/
          -- pack-53bac75e3de72bbca7cce5c5b6fede87
          -- pack-53bac75e3de72bbca7cce5c5b6fede87
        -- 47c40de.idx
        -- 47c40de.pack
      -- refs/
        -- heads/
          -- main
          -- my-branch
        -- tags/
          -- tag1
      -- rr-cache/
    -- txt/
      -- Doc.txt
47c40de.idx
47c40de.pack
13 directories, 18 files
```

```
..UnterDerHaube (-zsh)
git > main 11:41:01 ⚡
> git tag -a tag2 -m "Tag Message"
git > main 11:41:19 ⚡
> cat .git/refs/tags/tag2
3e8f3401b1313366371961b2d4d4be4785e43174
git > main 11:41:37 ⚡
> git cat-file -t 3e8f
tag
git > main 11:41:49 ⚡
> git cat-file -p 3e8f
object 2b568f94292134b0aefc5977f22e61163125330d
type commit
tag tag2
tagger Michael Kaufmann <wulfland@hotmail.com> 1606300879 +0100
Tag Message
git > main 11:41:52 ⚡
>
```

Every 2.0s... Andoria.local: Wed Nov 25 11:41:59 2020

```
.git/
  -- COMMIT_EDITMSG
  -- HEAD
  -- ORIG_HEAD
  -- config
  -- description
  -- index
  -- info/
    -- exclude
    -- refs
  -- logs/
    -- HEAD
    -- refs/
      -- heads/
        -- main
        -- my-branch
  -- objects/
    -- 3e/
      -- 8f3401b1313366371961b2d4d4be4785e43174
    -- info/
      -- packs
    -- pack-
      -- pack-53bac75e3de72bbca7ccee5c5b6fede87
      -- pack-53bac75e3de72bbca7ccee5c5b6fede87
  -- 47c40de.idx
  -- 47c40de.pack
    -- refs/
      -- heads/
        -- main
        -- my-branch
      -- tags/
        -- tag1
        -- tag2
    -- rr-cache/
  -- txt/
    -- Doc.txt
```

```
..UnterDerHaube (-zsh)
apple ~ % source/repos/UnterDerHaube git ⌂ main
apple ~ % vim txt/Doc.txt
apple ~ % source/repos/UnterDerHaube git ⌂ main !1
> git commit -am DritterCommit
[main 0e34b63] DritterCommit
 1 file changed, 2 insertions(+), 1 deletion(-)
apple ~ % source/repos/UnterDerHaube git ⌂ main
> git diff HEAD~2..HEAD
modified: txt/Doc.txt

@ txt/Doc.txt:2 @
Name: Dr. Emmet "Doc" Brown
Full-Name: Dr. Emmet "Doc" Brown
Original time: 1985
Nickname: Doc
Lala
Lala2

apple ~ % source/repos/UnterDerHaube git ⌂ main
> git format-patch HEAD~2..HEAD
0001-ZweiterCommit.patch
0002-DritterCommit.patch
apple ~ % source/repos/UnterDerHaube git ⌂ main ?2
> git reset HEAD~2 --hard
HEAD is now at fa794f0 ErsterCommit
apple ~ % source/repos/UnterDerHaube git ⌂ main ?2
> git apply 0001-ZweiterCommit.patch
apple ~ % source/repos/UnterDerHaube git ⌂ main !1 ?2
> git commit -am ZweiterCommit
[main 53babd2] ZweiterCommit
 1 file changed, 1 insertion(+)
apple ~ % source/repos/UnterDerHaube git ⌂ main ?2
> git am 0002-DritterCommit.patch
Applying: DritterCommit
apple ~ % source/repos/UnterDerHaube git ⌂ main ?2
> rm 000*
apple ~ % source/repos/UnterDerHaube git ⌂ main
>

watch (watch)
Every 2.0s... Andoria.local: Wed Nov 25 11:32:11 2020
.
└ .git/
    COMMIT_EDITMSG
    HEAD
    MERGE_RR
    ORIG_HEAD
    config
    description
    index
    info/
        exclude
    logs/
        HEAD
        refs/
            heads/
                main
    objects/
        0e/
            34b635bfae9ba4c5b182f04ce1d6818a685a8c
        2b/
            568f94292134b0aefc5977f22e61163125330d
        31/
            2ad420545b6166465de642930d81ab3a6e8618
        53/
            babd2d4f955bdbec8cefa12432a29264c6e927
        70/
            f8d78e3776ac011b60926fc52e30d76789de59
        7c/
            f2180461d7d876ece3c340606339a4f4137e77
        83/
            2b0d1f161855de45a37b3d21ae71179c8a1011
        aa/
            7831ac117e45ea660bd845f2042c998bb7aeed
        ba/
            105a49862bcf36b8ffff1dae703fc5678f7358a
        c3/
            0cfb9381a9cf06cf3455aee11d8bcff65c1478
```

```
..UnterDerHaube (-zsh)
apple ~ % source/repos/UnterDerHaube git ⌂ main
> git count-objects -Hv
count: 14
size: 56.00 KiB
in-pack: 0
packs: 0
size-pack: 0 bytes
prune-packable: 0
garbage: 0
size-garbage: 0 bytes
apple ~ % source/repos/UnterDerHaube git ⌂ main
> git repack -ad
Enumerating objects: 14, done.
Counting objects: 100% (14/14), done.
Delta compression using up to 12 threads
Compressing objects: 100% (8/8), done.
Writing objects: 100% (14/14), done.
Total 14 (delta 3), reused 0 (delta 0), pack-reused 0
apple ~ % source/repos/UnterDerHaube git ⌂ main
> git count-objects -Hv
count: 0
size: 0 bytes
in-pack: 14
packs: 1
size-pack: 2.45 KiB
prune-packable: 0
garbage: 0
size-garbage: 0 bytes
apple ~ % source/repos/UnterDerHaube git ⌂ main
>
```

11:33:06 ⓘ

11:33:15 ⓘ

11:33:22 ⓘ

11:33:28 ⓘ

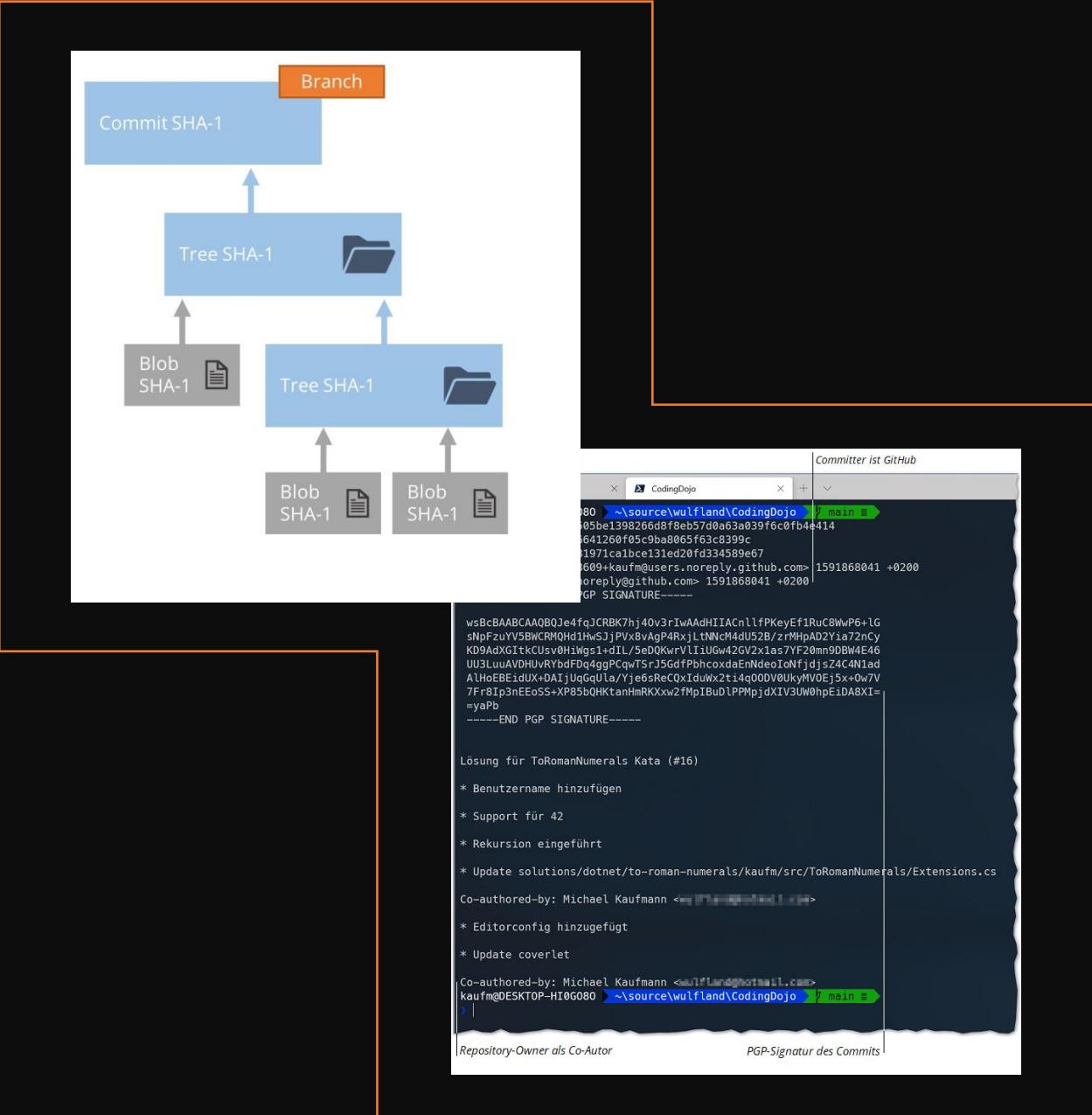
x watch (watch)

Every 2.0s... Andoria.local: Wed Nov 25 11:33:30 2020

```
.git/
  COMMIT_EDITMSG
  HEAD
  MERGE_RR
  ORIG_HEAD
  config
  description
  index
  info/
    exclude
    refs
  logs/
    HEAD
    refs/
      heads/
        main
  objects/
    info/
      packs
    pack/
      pack-f35c1409ecd10c777d3a738c201ec5a98
      a835913.idx
      a835913.pack
    refs/
      heads/
        main
      tags/
      rr-cache/
  txt/
    Doc.txt
```

13 directories, 16 files

Estructura del Commit





Demo: Parent Commit



```

❯ git commit-tree 409d1da9e721d43e6e42c5fee167e8ba36d5133a -m "Hallo Basta"
d7b9f5205e34558ede629149e27772c2101a6f4f
❯ git status
On branch main

No commits yet

Changes to be committed:
(use "git rm --cached <file>..." to unstage)
  new file:  txt/Dox.txt

❯ git cat-file -p d7b9f
tree 409d1da9e721d43e6e42c5fee167e8ba36d5133a
author Michael Kaufmann <5276337+wulfland@users.noreply.github.com> 1600341694 +0200
committer Michael Kaufmann <5276337+wulfland@users.noreply.github.com> 1600341694 +0200

Hallo Basta
❯ git update-ref refs/heads/main d7b9f5205e34558ede629149e27772c2101a6f4f
❯ git status
On branch main
nothing to commit, working tree clean
❯

```

```

Every 2.0s: tree -a
Michaels-MacBook-Pro.local: Thu Sep 17 13:23:20 2020

.
+-- .git
    |-- HEAD
    |-- config
    |-- description
    |-- hooks
        |-- applypatch-msg.sample
        |-- commit-msg.sample
        |-- fsmonitor-watchman.sample
        |-- post-update.sample
        |-- pre-applypatch.sample
        |-- pre-commit.sample
        |-- pre-merge-commit.sample
        |-- pre-push.sample
        |-- pre-rebase.sample
        |-- pre-receive.sample
        |-- prepare-commit-msg.sample
        |-- update.sample
    |-- index
    |-- info
        |-- exclude
    |-- logs
        |-- HEAD
    |-- refs
        |-- heads
            |-- main
    |-- objects
        |-- 40
            |-- 9d1da9e721d43e6e42c5fee167e8ba36d5133a
        |-- 45
            |-- 90a439fb3308d87d6d8f8d9b34f55e9fa6a11a
        |-- 7d
            |-- dc5a21ba0210931c432b5589102a04c2fd17cf
        |-- d7
            |-- b9f5205e34558ede629149e27772c2101a6f4f
        |-- f2
            |-- 0acb769ad757b2058c8c3c96e7eac2e1258503
        |-- info
        |-- pack
    |-- refs
        |-- heads
            |-- main
        |-- tags

```

```

❯ ~source/repos/basta/UnterDerHaube git ⌂ main ..... 13:31:31 ⓘ
❯ echo "Change" >> txt/Dox.txt
❯ ~source/repos/basta/UnterDerHaube git ⌂ main !1 ..... 13:31:45 ⓘ
❯ git add .
❯ ~source/repos/basta/UnterDerHaube git ⌂ main +1 ..... 13:31:51 ⓘ
❯ git commit -m "Normal change"
[main 124b36b] Normal change
 1 file changed, 1 insertion(+)
❯ ~source/repos/basta/UnterDerHaube git ⌂ main ..... 13:32:05 ⓘ
❯ git log -1
commit 124b36b0d18dbe356e30442105bf7529ebc3e971 (HEAD -> main)
Author: Michael Kaufmann <5276337+wulfland@users.noreply.github.com>
Date:   Thu Sep 17 13:32:05 2020 +0200

```

Normal change

```

❯ ~source/repos/basta/UnterDerHaube git ⌂ main ..... 13:32:23 ⓘ
❯ git cat-file -p 124b36
tree e52701bc3e95de2279598fb8a41e5f5bae9bf1bf
parent d7b9f5205e34558ede629149e27772c2101a6f4f
author Michael Kaufmann <5276337+wulfland@users.noreply.github.com> 1600342325 +0200
committer Michael Kaufmann <5276337+wulfland@users.noreply.github.com> 1600342325 +0200

```

Normal change

```

❯ ~source/repos/basta/UnterDerHaube git ⌂ main ..... 13:32:40 ⓘ
❯

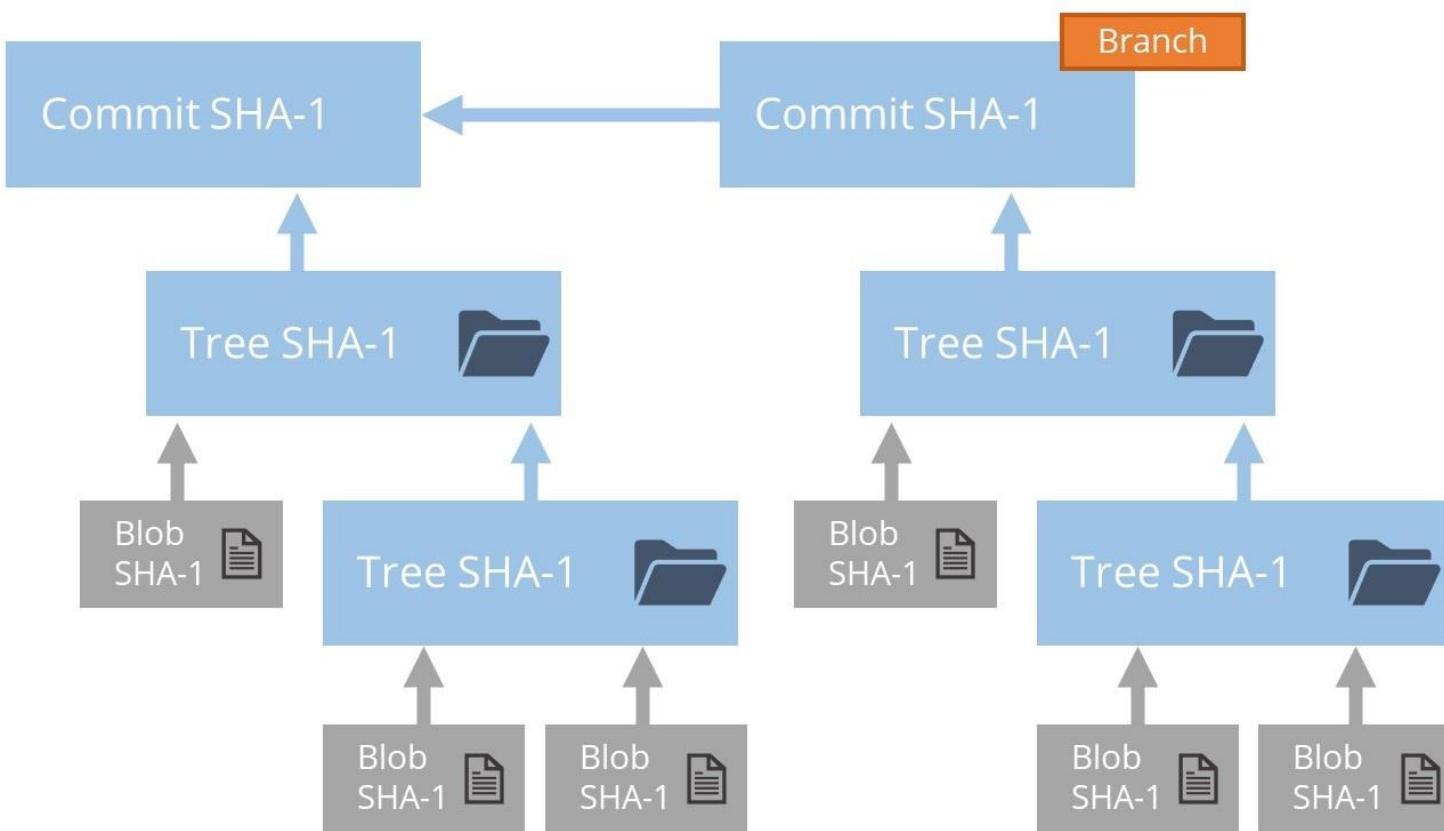
```

```

Every 2.0s: tree -a
.
  .git
    COMMIT_EDITMSG
    HEAD
    MERGE_RR
    config
    description
    hooks
      applypatch-msg.sample
      commit-msg.sample
      fsmonitor-watchman.sample
      post-update.sample
      pre-applypatch.sample
      pre-commit.sample
      pre-merge-commit.sample
      pre-push.sample
      pre-rebase.sample
      pre-receive.sample
      prepare-commit-msg.sample
      update.sample
    index
    info
      exclude
    logs
      HEAD
      refs
        heads
          main
    objects
      12
        4b36b0d18dbe356e30442105bf7529ebc3e971
      2b
        0df5dc9f2e17d4fd0945a2c6bf851d64aa8030
      40
        9d1da9e721d43e6e42c5fee167e8ba36d5133a
      45
        90a439fb3308d87d6d8f8d9b34f55e9fa6a11a
      7d
        dc5a21ba0210931c432b5589102a04c2fd17cf
      d7
        b9f5205e34558ede629149e27772c2101a6f4f
      d9
        587c473053800b9b074a952db01942800760

```

¿Qué ocurre en un Commit?



- `$ git add`
- `$ git hash-object`
- `$ git update-index`
- `$ git commit`
- `$ git write-tree`
- `$ git commit-tree`
- `$ git update-ref`

iGit
siempre
guarda el
archivo
completo!

```
✗ ..UnterDerHaube (zsh)
❯ git cat-file -p 2b0df5dc9f2e17d4fd0945a2c6bf851d64aa8030
Name: Dr. Emmet "Doc" Brown
Original time: 1985
Change
❯ git diff HEAD^!
modified: txt/Dox.txt

@ txt/Dox.txt:1 @
Name: Dr. Emmet "Doc" Brown
Original time: 1985
Change

❯
```



Demo: Pack



```
..UnterDerHaube (zsh)
❯ git format-patch HEAD~2..HEAD
0001-Hallo-Basta2.patch
0002-Hallo-Patch.patch
❯ git reset HEAD~2 --hard
HEAD is now at 139fc85 Hallo Basta
❯ git apply 0001-Hallo-Basta2.patch
❯ git commit -am "patched change"
[main f1a7a86] patched change
 1 file changed, 1 insertion(+)
❯ git am 0002-Hallo-Patch.patch
Applying: Hallo Patch
❯
```

iGit
siempre
transfiere
los parches
(patch)!



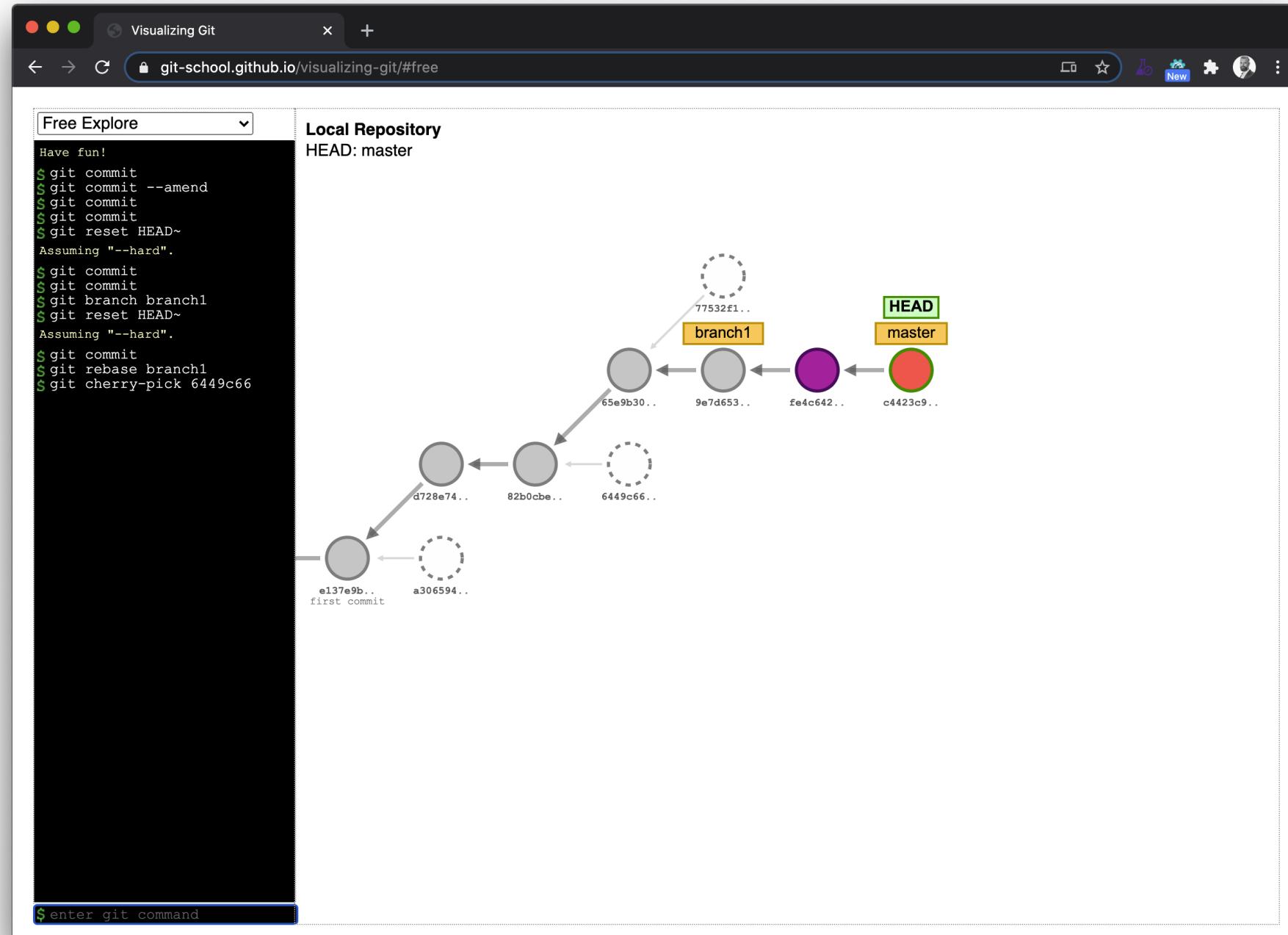
Demo: Patches





A close-up photograph showing a person's hands working on a red mechanical device, possibly a car engine. One hand holds a wrench, and the other hand is partially visible. The background is blurred, focusing on the hands and the tool.

Demo:
Branches y
Tags



Viajes en el tiempo en Git



Demo: Rebase Demo



A Clean History

»I want clean history, but that really means (a) clean and (b) history.

People can (and probably should) rebase their *private* trees (their own work). That's a *cleanup*. But never other peoples code. That's a *destroy history*.« – Linus Torvalds





Keep your own history readable

» Some people do this by just working things out in their head first, and not making mistakes. But that's very rare, and for the rest of us, we use git rebase etc while we work on our problems. « - Linus Torvalds.

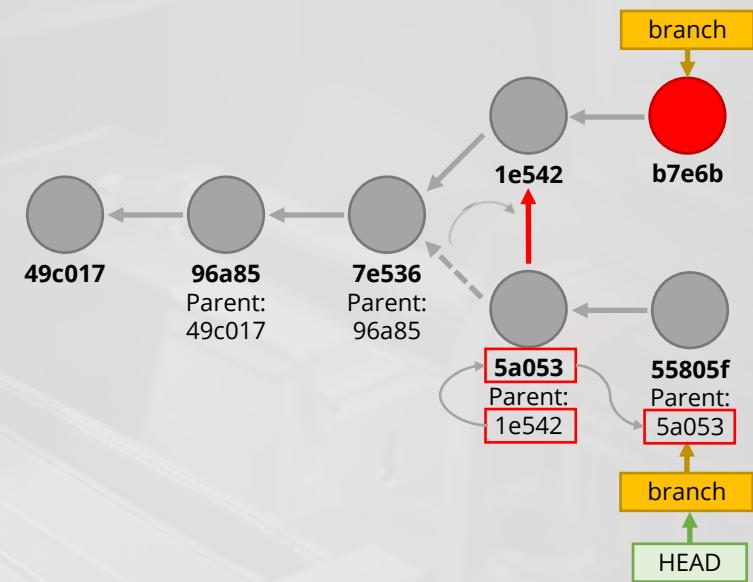
Directriz principal:

¡No cambies jamás la línea del tiempo de otras personas! No cambies jamás tu propia línea del tiempo si ya es parte de la historia.



Viajes en el tiempo

```
$ git commit --amend  
$ git rebase  
$ git merge -squash  
$ git cherry-pick
```

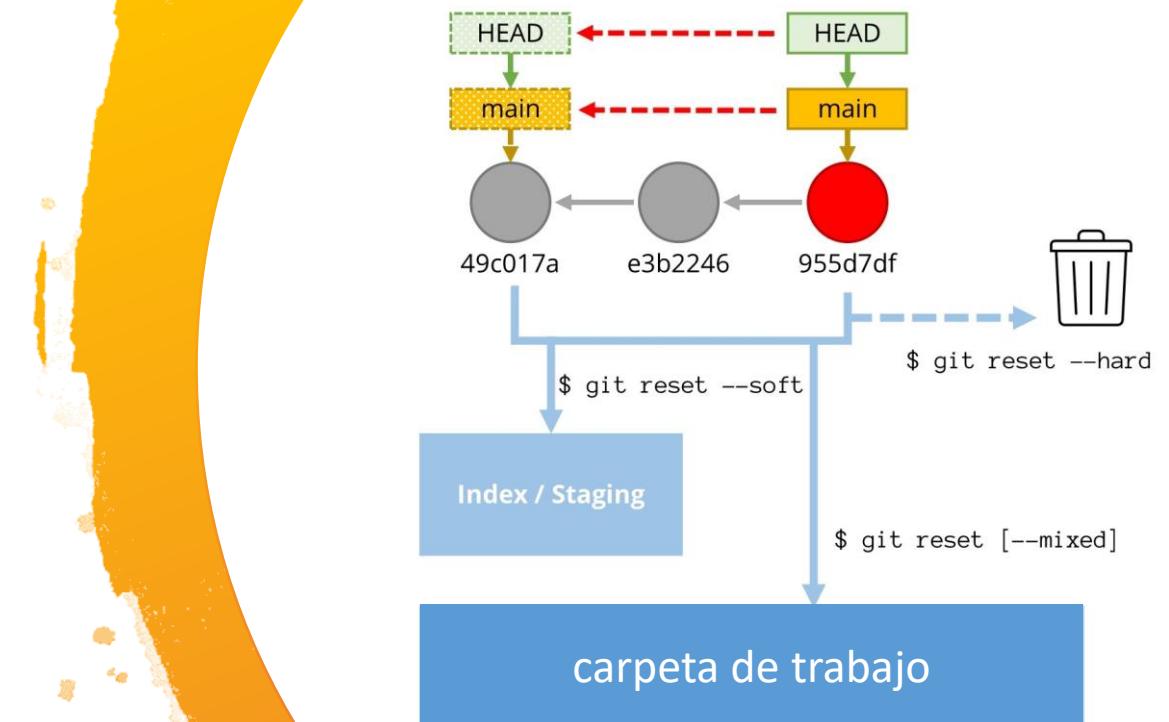




Demo:
Navegación
para viajes en
el tiempo

Viajes en el tiempo con reset

```
$ git reset REF  
  [--hard |  
   --soft |  
   --mixed]
```



```
..UnterDerHaube (zsh)
❯ ~ /s/repos/basta/UnterDerHaube git ⚡ main ?2 18:27:37 ⓘ
> git reset --hard 'HEAD@{3}'
HEAD is now at 6126639 Hallo Patch
❯ ~ /s/repos/basta/UnterDerHaube git ⚡ main ?2 18:27:51 ⓘ
> git reflog
6126639 (HEAD -> main) HEAD@{0}: reset: moving to HEAD@{3}
7738027 HEAD@{1}: am: Hallo Patch
f1a7a86 HEAD@{2}: commit: patched change
139fc85 HEAD@{3}: reset: moving to HEAD~2
6126639 (HEAD -> main) HEAD@{4}: commit: Hallo Patch
c9bdc02 (branch1) HEAD@{5}: reset: moving to HEAD@{13}
ddfbfe8 HEAD@{6}: am: Add -p demo
560b534 HEAD@{7}: commit: Patch Demo
32f9299 HEAD@{8}: reset: moving to HEAD~2
```

Reflog

- ¡Default son 30 días!
- ¡ Local!

```
reflogExpireUnreachable = 30
reflogExpire = 90
pruneExpire = 2.weeks.ago
```

Garbage Collection

Git activa internamente con regularidad gc con los siguientes comandos:

- \$ git pull
- \$ git merge
- \$ git rebase
- \$ git commit

gc activa internamente muchos otros comandos, como por ejemplo:

- \$ git prune,
- \$ git repack oder
- \$ git rerere.



DON'T PANIC

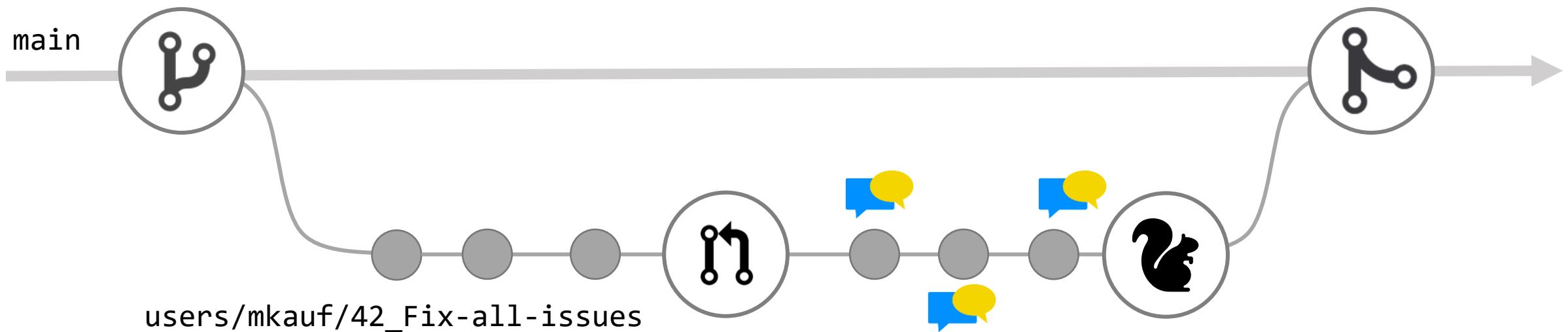
The Hitchhiker's Guide to the Galaxy



Hitchhikers Guide to Time Travel (Guía del autoestopista por el tiempo)

- **Viaja con poco equipaje**
(Modificaciones en el stash)
- **Manipula siempre únicamente tu línea del tiempo**
- **Force-Push siempre solo en Branches “privados”**
- **Usa el Reflog como Backup**

Welcome to MyFlow



Private Branches (users/* private/* etc.)

Colaboración con Pull-Requests

Solo el autor **modifica** su Branch (**suggestions**)

El Branch principal **main** está **protegido** con Branch-Protection for Force-Push

\$ git push origin +users/mkauf/42_Fix-all-issues

THIS IS GIT. IT TRACKS COLLABORATIVE WORK
ON PROJECTS THROUGH A BEAUTIFUL
DISTRIBUTED GRAPH THEORY TREE MODEL.

COOL. HOW DO WE USE IT?

NO IDEA. JUST MEMORIZIZE THESE SHELL
COMMANDS AND TYPE THEM TO SYNC UP.
IF YOU GET ERRORS, SAVE YOUR WORK
ELSEWHERE, DELETE THE PROJECT,
AND DOWNLOAD A FRESH COPY.



¿Qué es Git?

«Esto es Git. Ayuda al seguimiento del trabajo colaborativo en proyectos a través del uso de un bello modelo de árbol distribuido de la teoría de grafos.»

«¡Qué interesante! ¿Cómo se utiliza?»

«Ni idea. Apréndete simplemente los siguientes comandos e introdúcelos en la línea de comandos para sincronizar tu trabajo. Si te da error, guarda tu trabajo en otro sitio, borra el proyecto y descárgate una nueva copia.»



GIT(1)

man git

Git Manual

⌘⌘3

GIT(1)

NAME

git - the stupid content tracker

SYNOPSIS

```
git [--version] [--help] [-C <path>] [-c <name>=<value>]
     [--exec-path[=<path>]] [--html-path] [--man-path] [--info-path]
     [-p|--paginate|-P|--no-pager] [--no-replace-objects] [--bare]
     [--git-dir=<path>] [--work-tree=<path>] [--namespace=<name>]
     [--super-prefix=<path>]
     <command> [<args>]
```

DESCRIPTION

Git is a fast, scalable, distributed revision control system with an unusually rich command set that provides both high-level operations and full access to internals.

See **gittutorial(7)** to get started, then see **giteveryday(7)** for a useful minimum set of commands. The **Git User's Manual[1]** has a more in-depth introduction.

After you mastered the basic concepts, you can come back to this page to learn what commands Git offers. You can learn more about individual Git commands with "git help command". **gitcli(7)** manual page gives you an overview of the command-line command syntax.

A formatted and hyperlinked copy of the latest Git documentation can be viewed at <https://git.github.io/htmldocs/git.html> or <https://git-scm.com/docs>.





Muchas gracias

Entendamos Git - En el engranaje del control de código fuente