git (altran)

devops, cursos, git

Ponente: **Daniel Sanchez**

conceptos

- HEAD: puntero al commit de trabajo

config

- `--system` : /etc/gitconfig
  - `git config --system color.status auto`
- `--global` : ~/.gitconfig
- `--local`: .git/config → por cada repositorio

valores

- `core.symlinks` : a **true** respeta los symlinks, a **false** (default) los trata como ficheros

commit

- `git commit –amend` : sobrescribe el último commit, editando mensaje → crea un nuevo commit con los cambios
  - si nos hemos dejado alguna cosa por añadir/modificar o hemos puesto un mensaje poco significativo o erróneo

checkout

- `git checkout -` : moverte a la rama anterior (de la que venías)

remote

- *upstreams* : remoto por defecto, configurado/ble en cada rama, donde sincronizará
  - `git push -u <remoto> <rama>` : define el *upstream* de la rama (al tiempo que está enviando). Al hacer un `git pull` no será necesario especificar

branch (ramas)

listar

- `git branch` : muestra ramas locales
  - `-v` : verboso

    - `-v –merged` : ramas mergeadas
    - `-v –no-merged` : ramas no mergeadas
    - `-a` : locales + remotas
    - `-avv` : locales + remotas + upstreams
    - `git branch <rama> -u <origin>/<rama>` : establece el upstream

crear

- `git branch <rama>` : crea rama desde la actual
    - `git branch <rama> <rama_madre>` crea rama desde rama_madre
- `git checkout <rama>` : nos movemos a rama
    - `git checkout -b <rama>` : creamos rama a partir de la actual y nos movemos a ella
    - `git checkout -b <rama> <rama_madre>`

sincronizar

- `git push <origin> <rama>` : envía rama al remoto <origin>
    - `-u` o `--set-upstream` : y establece el upstream
- `git pull` : descarga los cambios de la rama actual
    - `--all` : de todas las ramas
    - = git fetch + git merge @{u}
    - `--rebase` : reescribe la historia si esta ha sido reescritura en el remote. Puede evitar problemas de historias no coincidentes (DPS[1] lo usa por defecto)

borrar

borrar la 'rama' en local si está merged

git branch -d 'rama'

fuerza el borrado de la 'rama' local

git branch -D 'rama'

borra 'rama' en 'origin'

git push –delete [-d] 'origin' 'rama'

borra 'rama' en 'origin' (OJO 2 puntos)

git push 'origin' :'rama'

borra ramas borradas en 'origin'

git remote prune 'origin'

stash (limbo)

git STASH

tags

git TAG

merge

git MERGE

cherry-pick

git CHERRY-PICK

revert

git REVERT

reset

git RESET

rebase

git REBASE

+ info

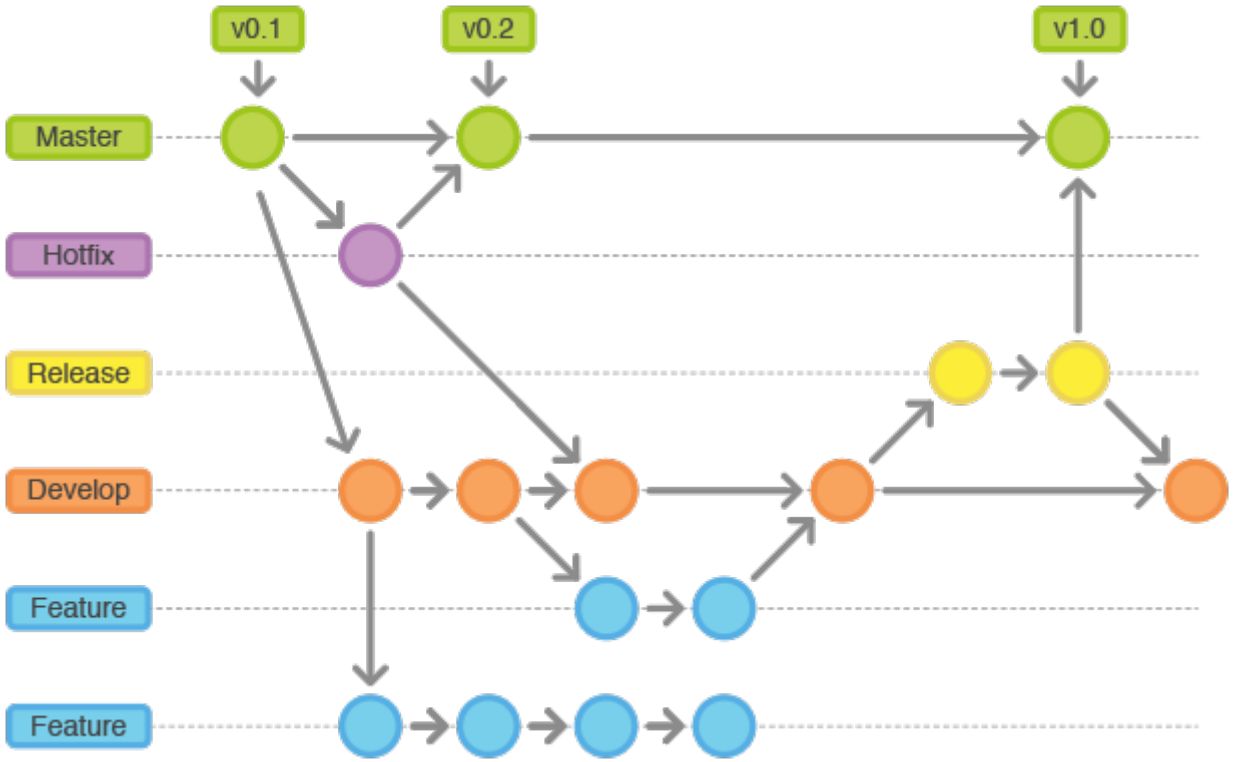
- http://cambrico.net/git-control-de-versiones/rebase-en-git
- https://git-scm.com/book/es/v1/Ramificaciones-en-Git-Reorganizando-el-trabajo-realizado
- https://www.solucionex.com/blog/git-merge-o-git-rebase

reflog

- `git reflog` : muestra las acciones realizadas en el repositorio
- `git merge <id_reflog>` : merge commits perdidos
- `git checkout -b <rama> <id_reflog>` crea rama con la información del pasado

FLOW

- patrón para el nombrado de ramas: **categoria/tipo-ID**
  - catetorias: feature, hotfix



[1)]

Daniel Sanchez Puig

From:
https://matebcn.eu/ - **miguel angel torres egea**

Permanent link:
https://matebcn.eu/info:cursos:altran:git

Last update: **31/10/2025 19:09**