

Fusionar ramas

La fusión es la forma que existe en Git de volver a unir un historial bifurcado.

En Git tenemos dos maneras de hacer: por medio de un **merge** o por medio de un **rebase**. Lo primero que tienes que saber es que un **git rebase** resuelve el mismo problema que un **git merge** pero ambos comandos lo hacen de maneras muy diferentes.

Empiezo por explicarte el funcionamiento del **merge** y más adelante te explico cómo usar el **rebase** y cómo funciona.

Merge: funcionamiento

El comando **git merge** permite tomar las líneas independientes creadas por un **git branch** e integrarlas en una sola rama.

Importante

Todos los comandos presentados en este apartado harán que se fusionen en la rama actual. La rama actual se actualizará para reflejar la fusión, pero la rama de destino no se verá afectada en absoluto.

Un **merge** se utiliza para combinar dos ramas. Imagina que tenemos dos ramas: **main** y **cambios**. Queremos fusionar el historial de la rama **cambios** en la rama **main**. Lo que haremos es:

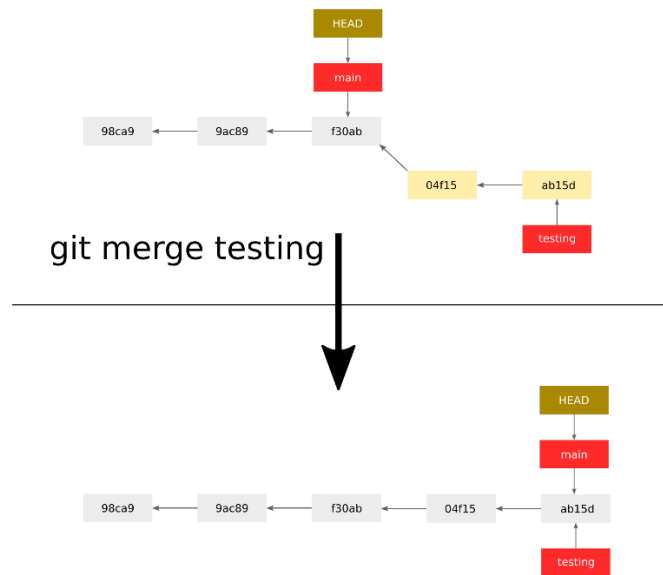
- Movernos a la rama **main**.
- Ejecutar el comando **git merge cambios**

Git tiene varias formas de realizar el *merge* o la fusión, dependiendo del estado del historial de ambas ramas a fusionar. Veámoslas en los siguientes apartados.

Fusión con avance rápido

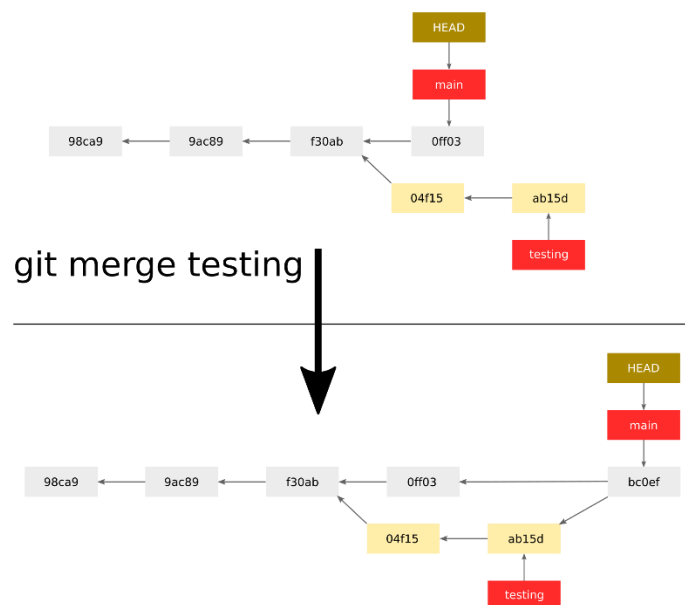
Se produce una fusión con avance rápido cuando hay un proceso lineal desde el extremo de la rama actual hasta la rama de destino. En lugar de fusionar «realmente» las ramas, lo único que tiene que hacer Git es mover el extremo de la rama actual al extremo de la rama de destino.

Ejemplo gráfico:



Fusión de tres vías

Git no puede fusionar mediante avance rápido cuando las ramas han divergido, es decir, la rama en la que vamos a fusionar ha seguido avanzando. En estos casos se usa una **fusión de tres vías** en la que se usa un *commit* para unir ambas ramas. Este *commit* une las ramas por los dos extremos, como se ve en el siguiente gráfico:



Fusión con conflictos

Si las dos ramas que tratas de fusionar han cambiado la misma parte del mismo archivo, Git no podrá averiguar qué versión utilizar. Cuando esto ocurre, Git se detiene justo antes de la confirmación de fusión para que puedas resolver los conflictos manualmente.

Cuando se observa un conflicto de fusión, la ejecución del comando **git status** muestra qué archivos se deben resolver.

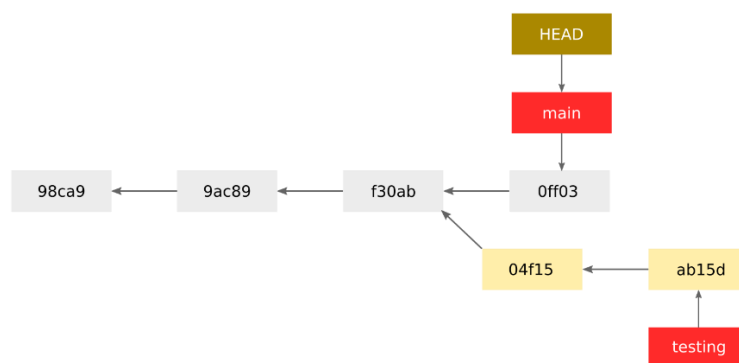
Más adelante veremos cómo resolver conflictos y entraremos en más detalles.

Rebase: funcionamiento

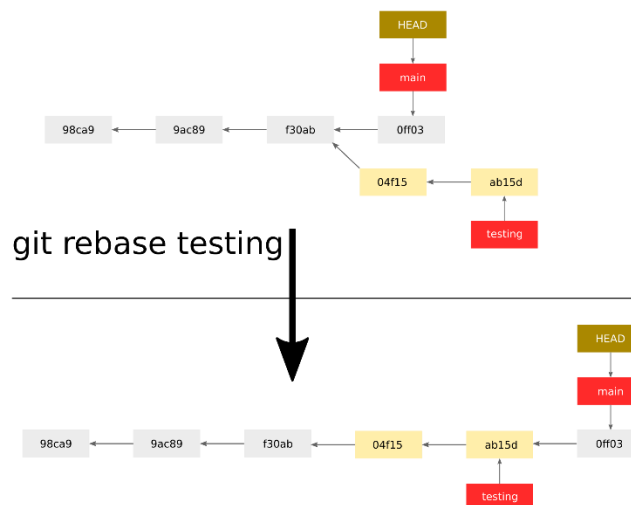
La ventaja del **merge** es que es una operación *no destructiva*, ningún commit se destruye o desaparece, están todos ahí, con las bifurcaciones que se hicieron incluidas. Lo primero que tienes que saber del **rebase** es que esta opción puede llevarnos a escollos de los que te hablaré más tarde.

Cuando haces un rebase toda la rama de características se incorporan al final de la rama sobre la que haces el *rebase*. Así que, el *rebase* reescribe la historia mediante la creación de nuevos *commits*.

Imagina la situación siguiente:



Si, desde la rama **main**, ejecutaras un **git rebase testing** este sería el resultado final:



El principal beneficio de **rebasing** frente al **merging** es que obtienes un historial mucho más limpio. Primero, este elimina los *merge* innecesarios requeridos por **git merge**. Y segundo, como ves en el diagrama de arriba el *rebasing* resulta en un historial lineal.

Pero hay dos contrapartidas a este historial de *commits* lineales y limpios: la seguridad y la trazabilidad.

Si no sigues la «Regla de Oro del Rebasing», reescribir el historial del proyecto puede ser potencialmente catastrófico y, menos importante, el *rebasing* pierde el contexto que te ofrece el *merge*.

La regla de oro del rebasing

A estas alturas, esta regla queda pendiente, ya que hasta que no veamos cómo trabajar de forma colaborativa en repositorios remotos no puedes entender esta regla.

Información de ramas: fusionadas y no fusionadas

Ya te he mostrado como ver las ramas que hay en tu repositorio, pero llegados aquí, y una vez sabes cómo mezclar o fusionar ramas, es importante conocer algunas opciones útiles a la hora de comprobar qué ramas hay en tu repositorio:

- **git branch -v**: para ver la última confirmación de cambios en cada rama.
- **git branch --merged**: para ver las ramas fusionadas.
- **git branch --no-merged**: para ver qué ramas no han sido fusionadas todavía.