

Git



¿Qué es Git?

Git es un sistema de control de versiones que permite registrar los cambios realizados a archivos de computadora con el fin de coordinar los avances de un equipo de trabajo.

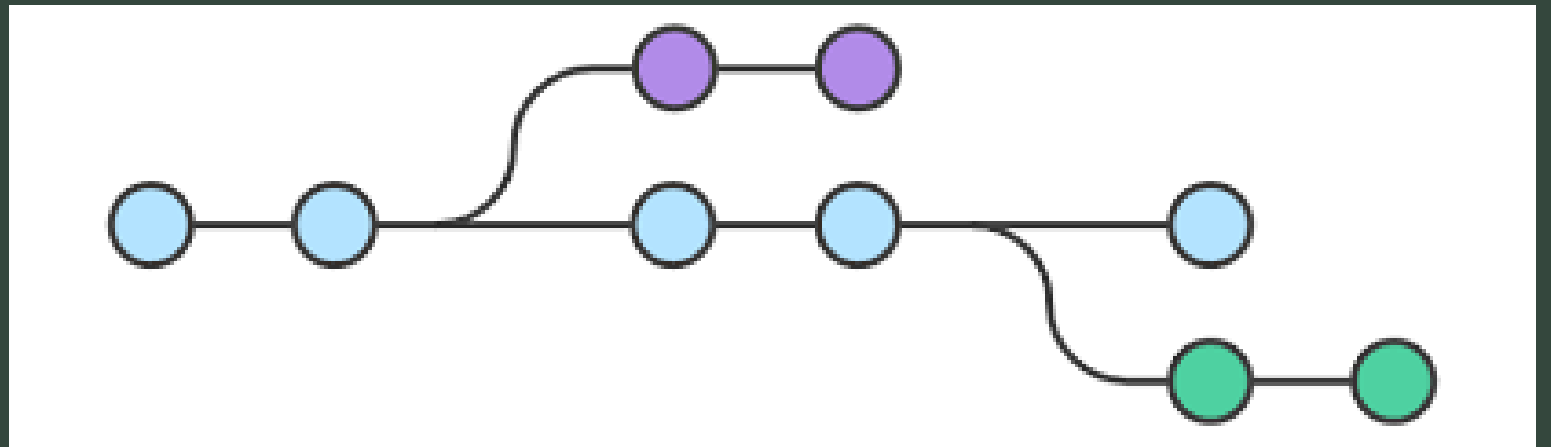


Terminología

Repositorio: Proyecto seguido por Git

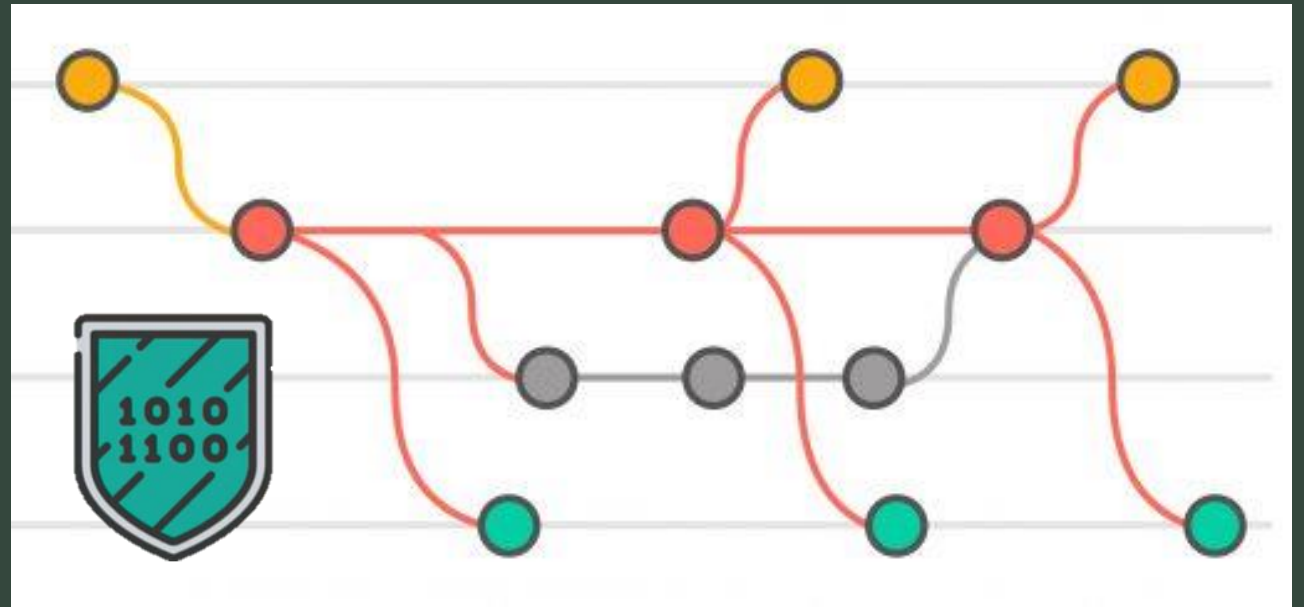
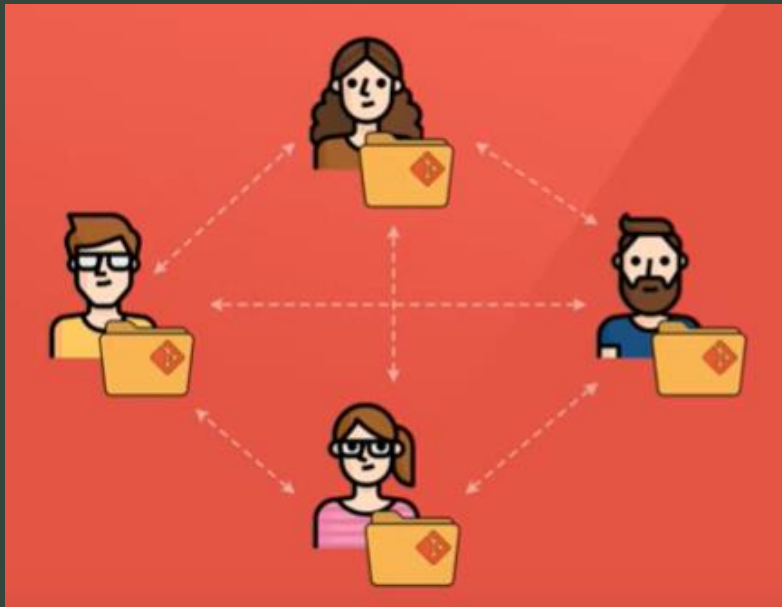
Commit: Cambio realizado a un proyecto

Rama: Copia aislada del proyecto en que una persona puede realizar su respectivo trabajo



Características

- 1) Distribuido: No hay una conexión central a Internet o un servidor
- 2) Ramas: Las diversificaciones del proyecto se pueden unir después
- 3) Integridad de datos: Asegura que cada persona tenga los mismos datos

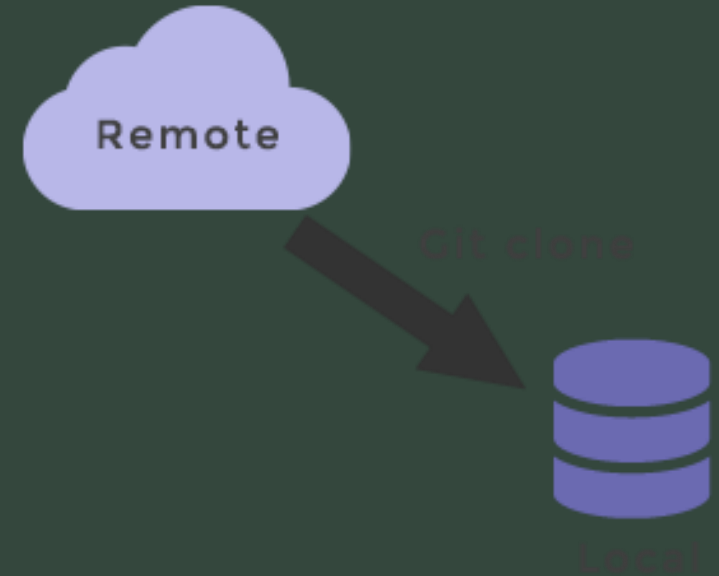


Clonar un repositorio

git clone (Dirección URL del repositorio):

Clona el repositorio en tu computadora

*NOTA: Primero debes crear una carpeta en tu computadora donde se guardará el repositorio

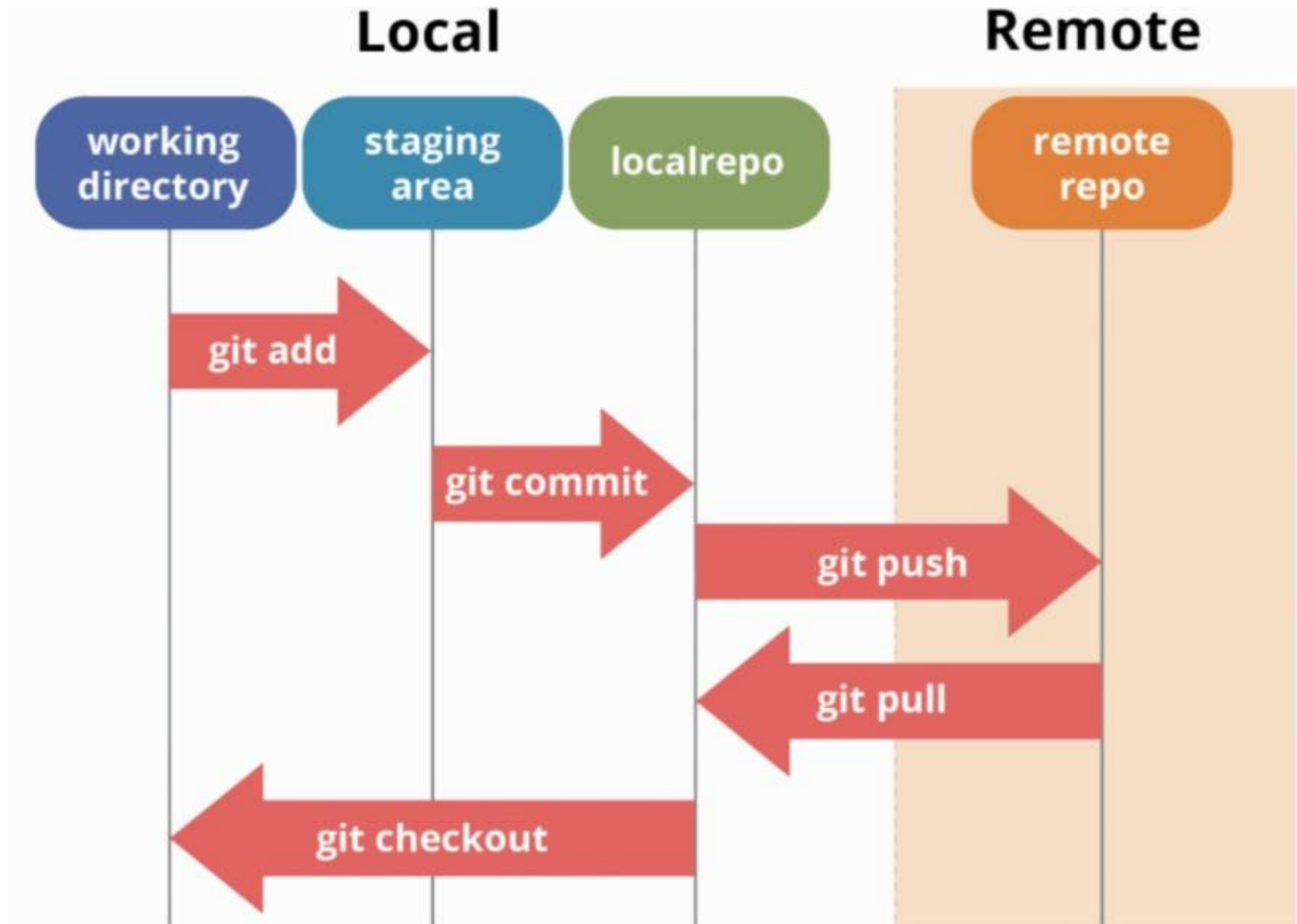


Subir archivos al repositorio

- 1) git add -A = Sube los cambios realizados a los archivos a una zona de pruebas para que Git empiece a rastrearlos.
- 2) git commit -m "Mensaje" = Sube el contenido al repositorio local
- 3) git push origin [Rama]* = Sube el contenido al repositorio remoto

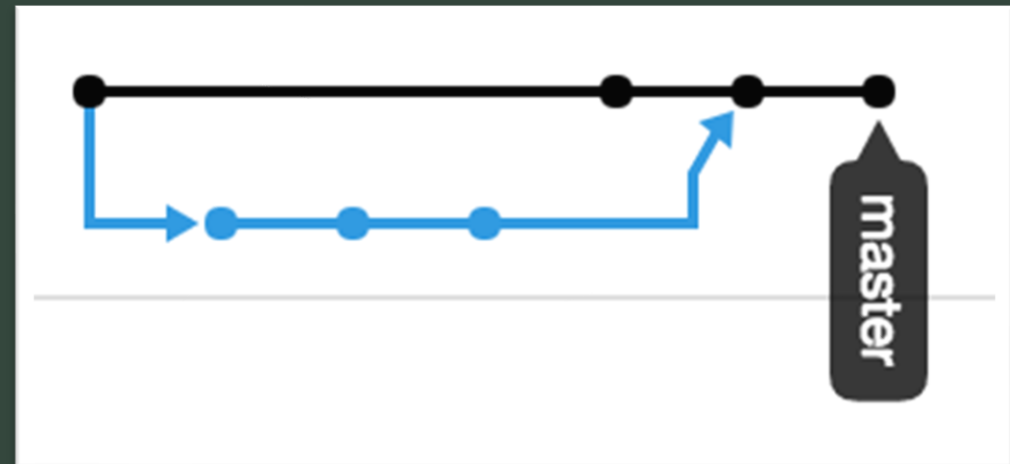
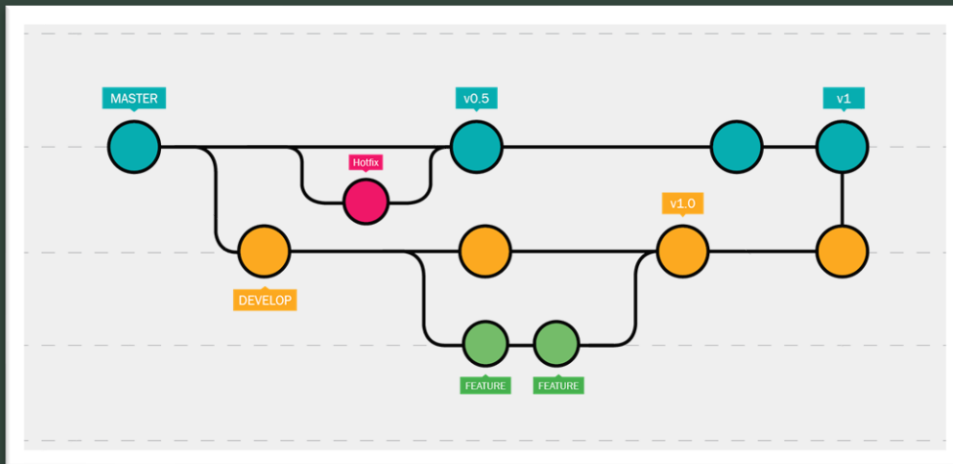
*Nota: El mensaje debe explicar a detalle qué cambio se ha realizado

*Nota: Debes elegir una rama del repositorio local en la cual se subirá el nuevo contenido



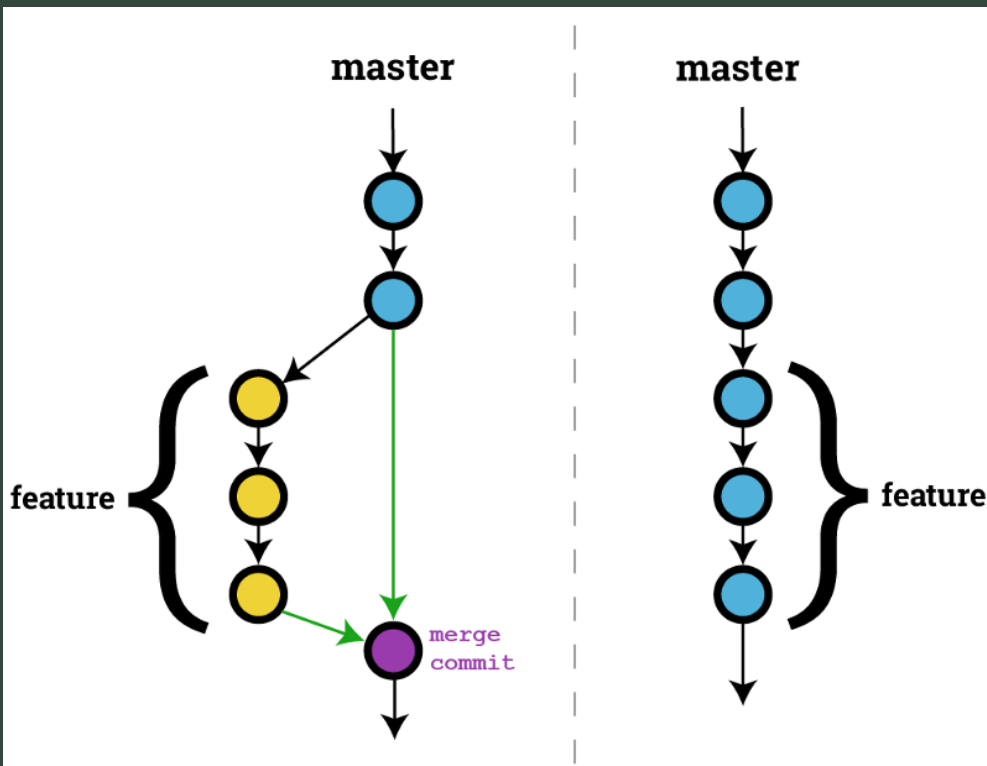
Administrar ramas

- 1) git checkout: Cambia de rama
- 2) git checkout -b: Crea una nueva rama y cambia a ella
- 3) git branch: Enlista todas las ramas del repositorio
- 4) git branch -d: Elimina una rama
- 5) git status: Identifica en cuál rama estás actualmente



Fusionar ramas

- 1) git pull = Carga el contenido de una rama del repositorio
- 2) git merge = Fusiona el contenido de dos ramas



after each git merge



Git merge

Fusionar ramas

- 1) git checkout [Rama destino]
- 2) git pull origin [Rama destino]
- 3) git merge [Rama origen]
- 4) git push origin [Rama destino]

*Rama destino = Rama donde se guardará la fusión

*Rama origen = Rama que se quiere fusionar

Nota: Hay que revisar de que el merge no genere conflictos

¿Qué es un conflicto?

Es una inconsistencia entre los archivos de dos ramas de un repositorio
Se da cuando dos personas cambiaron la misma porción de un código.

15

16

17

18

19

20

21

<<<<<< add-emoji

We're all authors :star: :zap: :sparkles:

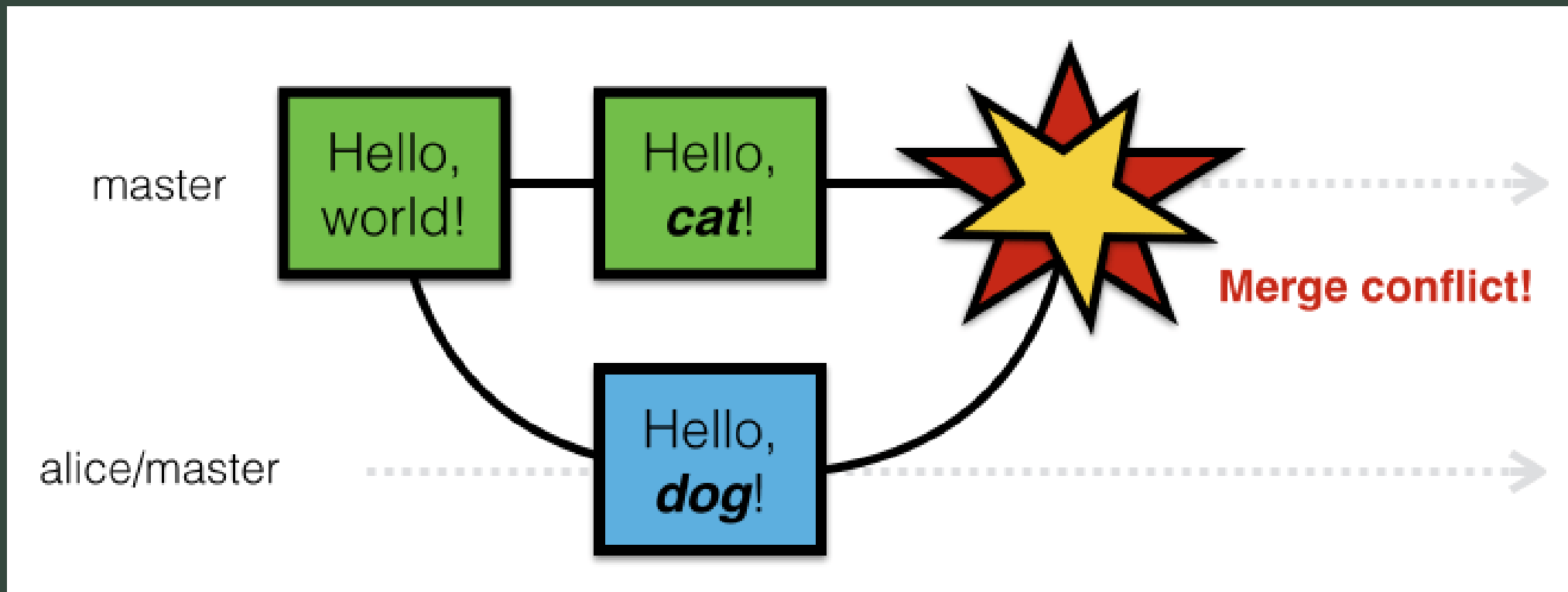
=====

We're all authors :star: :zap: :heart:

>>>>>> master

¿Qué hacer ante un conflicto?

Dos ramas no se fusionarán correctamente hasta que los conflictos se hayan arreglado. Esto se hace decidiendo cuál de los cambios se queda.



¿Qué hacer ante un conflicto?

Hay varios entornos de desarrollo integrado (IDE) y editores de texto que marcan las partes de código que están en conflicto. Estos son algunos de ellos:



VS Code



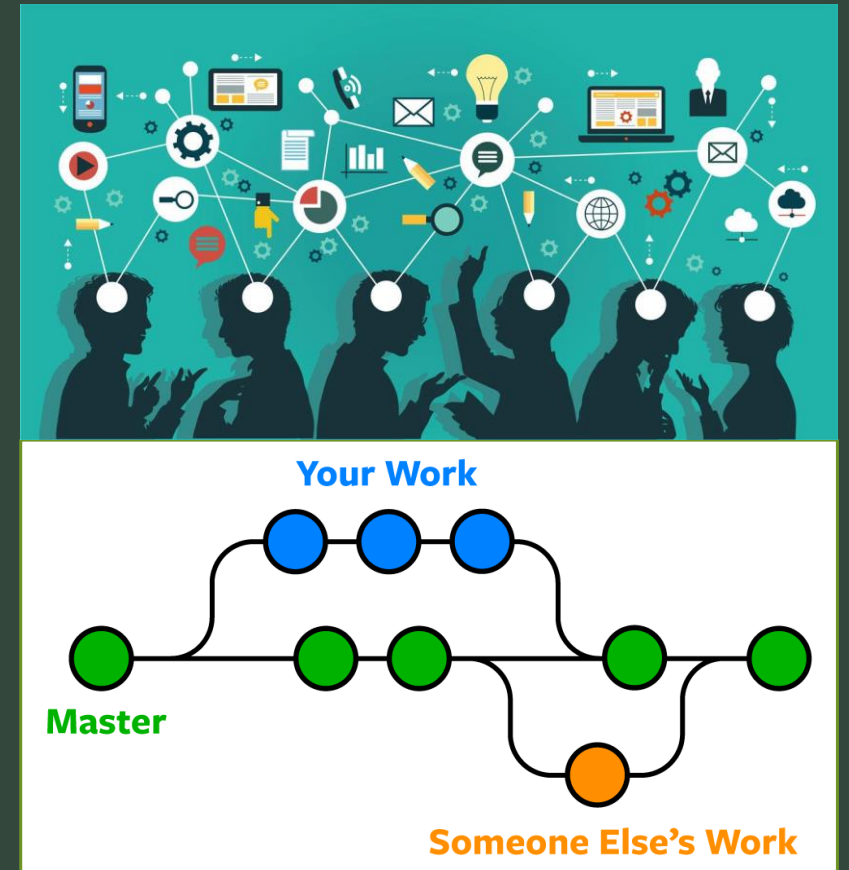
IntelliJ IDEA



Sublime Text 3

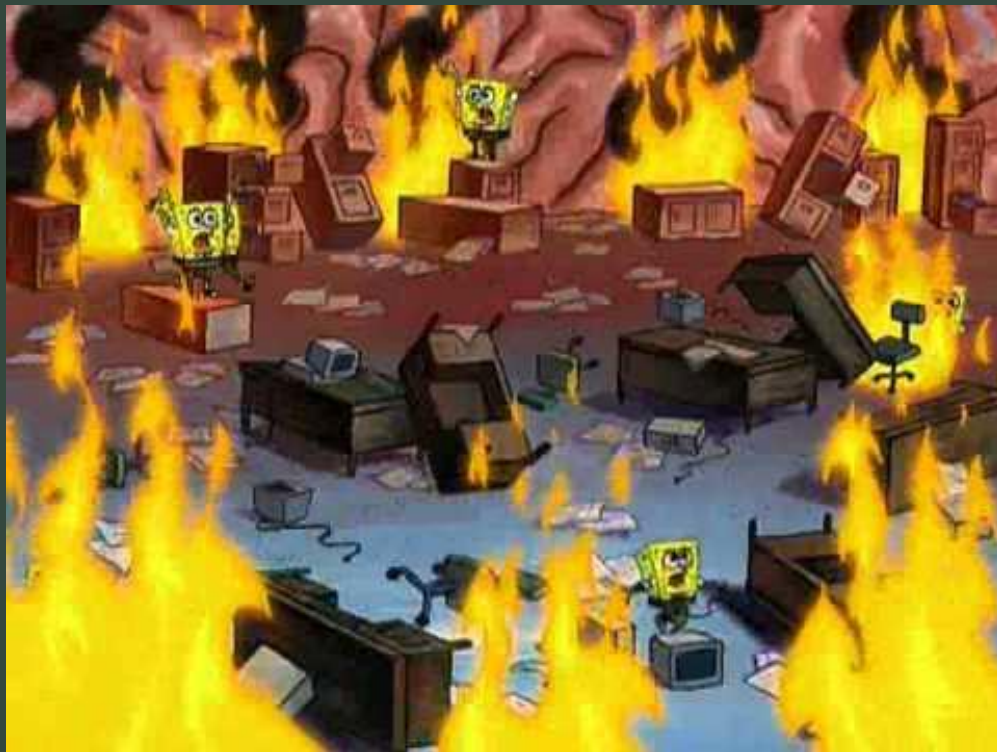
¿Cómo evitar un conflicto?

La mejor forma de evitar conflictos al fusionar ramas es mediante la integración continua de avances. Esto significa subir de forma constante tus avances al repositorio para que las diferencias de código entre tu rama y las de los demás no sea tan grande. Esto requiere que haya una comunicación constante en tu equipo para informar los cambios que se hacen al código.



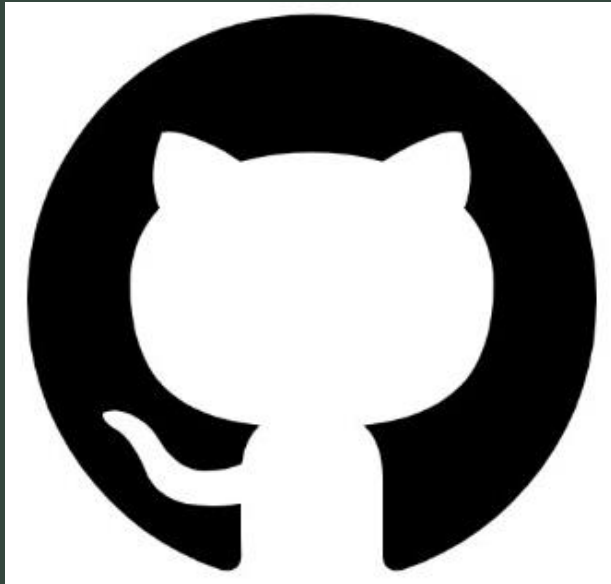
¿Cómo evitar un conflicto?

Si no subes tus avances de forma constante, entonces habrá más conflictos al hacer el merge, lo cual tomará más tiempo arreglarlos.



Repositorios en la nube

Hay varias empresas que ofrecen repositorios en la nube como remotos.
Algunas de las más importantes son:



Github



Bitbucket



Gitlab

¡Empecemos a
trabajar en Git!