



GitHub es una herramienta para el desarrollo colaborativo de software principalmente. utiliza el sistema de control de versiones "Git".

# Descripción del problema

## EL PROCESO DE TU TESIS



Tesis



Tesis final



Tesis final este  
sí



Tesis final este  
sí sí sí



Tesis final 2



Tesis final final



Tesis final listo



Tesis final por fin



Tesis final por fin  
eso espero



Tesis final por fin  
the end



Tesis finalisimo



Tesis ultimo



Tesis ultimo  
ahora si



Tesis ultimo de  
los ultimos



Tesis ultimo  
final ok

**My Code works**



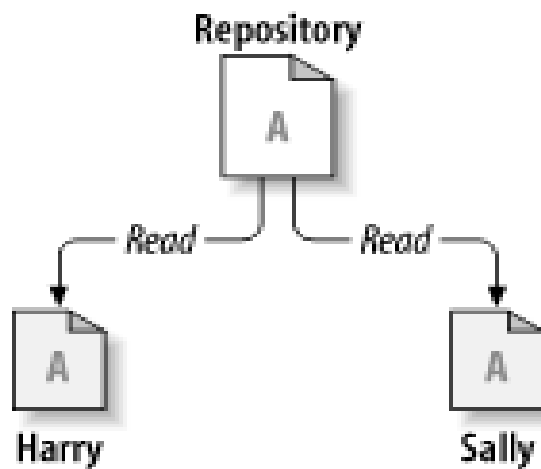
**And  
I don't even know why.**

quickmeme.com

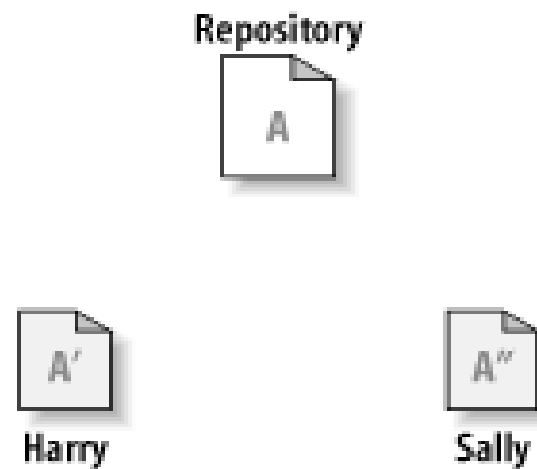
# ¿Colaborar?



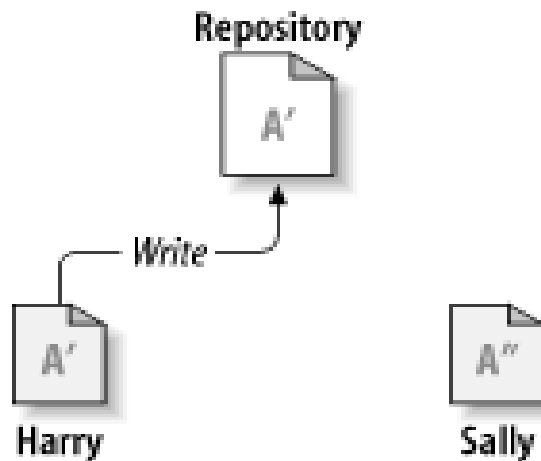
*Two users read the same file*



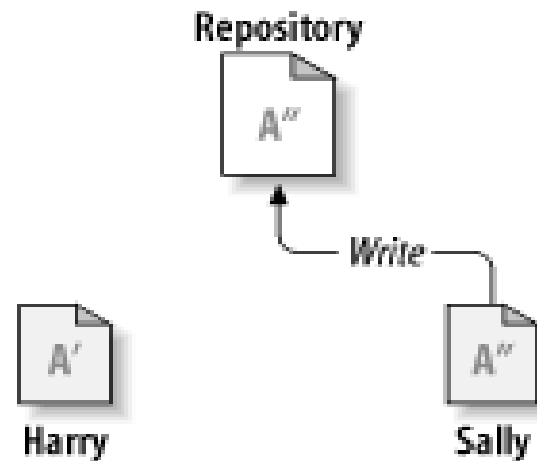
*They both begin to edit their copies*



*Harry publishes his version first*

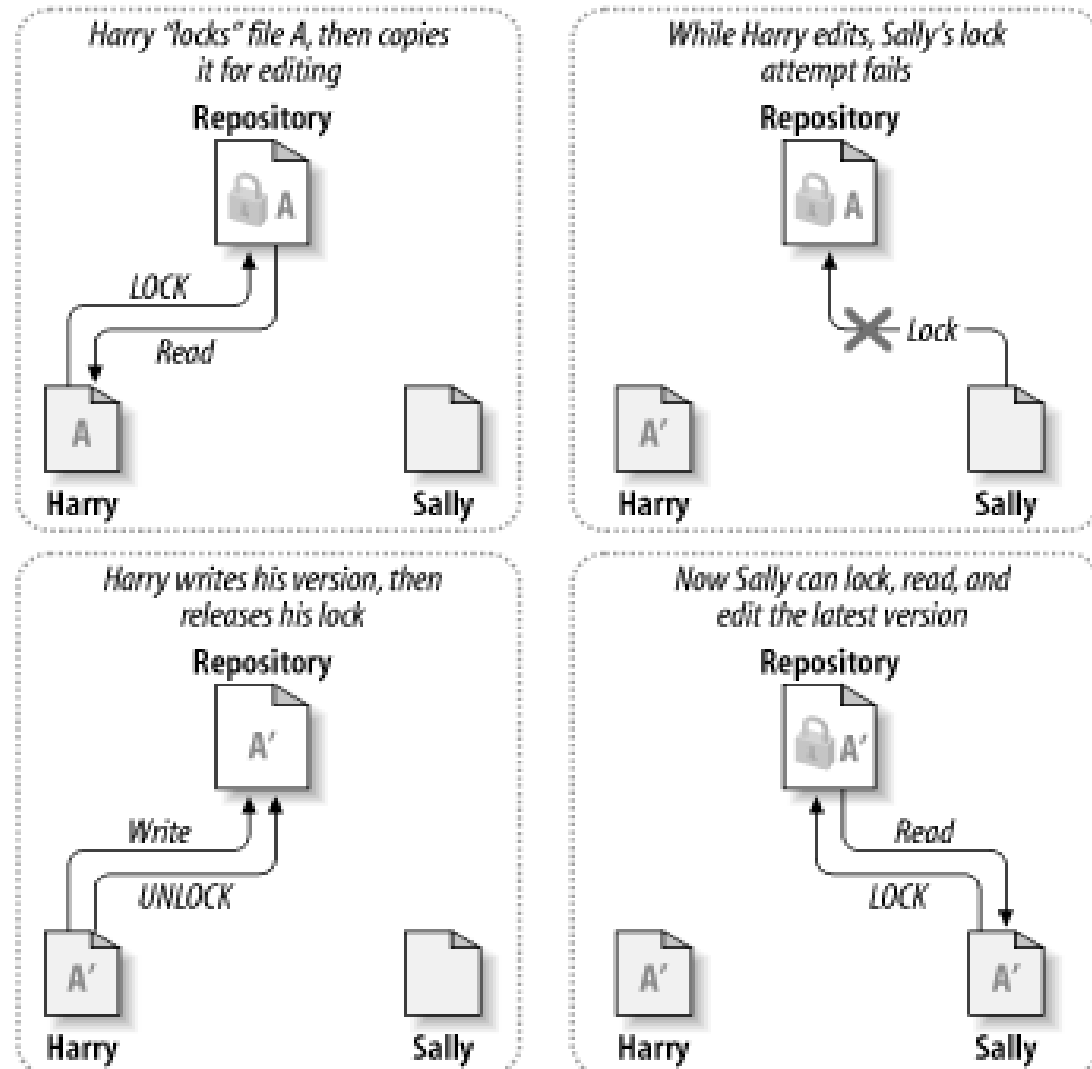


*Sally accidentally overwrites Harry's version*



# The lock-modify-lock solution

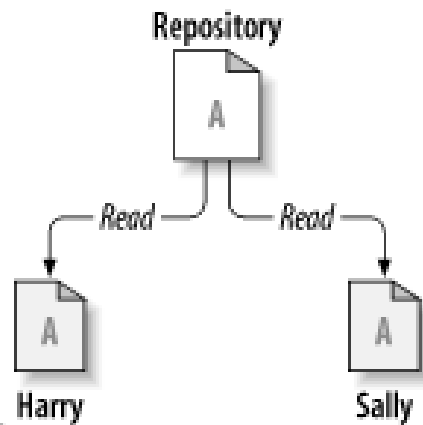
Problemas:  
¿Y si Harry lo olvidó?  
Harry esta de vacaciones  
Serialización innecesaria



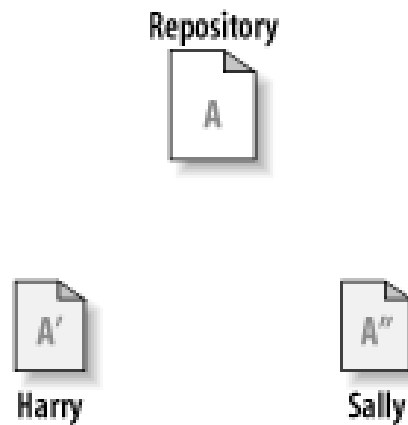


# copy-modify-merge solution

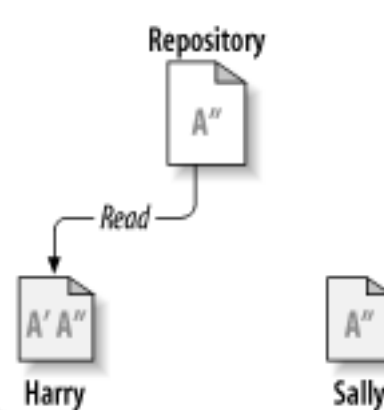
*Two users copy the same file*



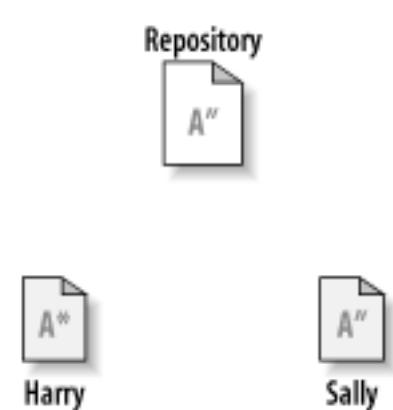
*They both begin to edit their copies*



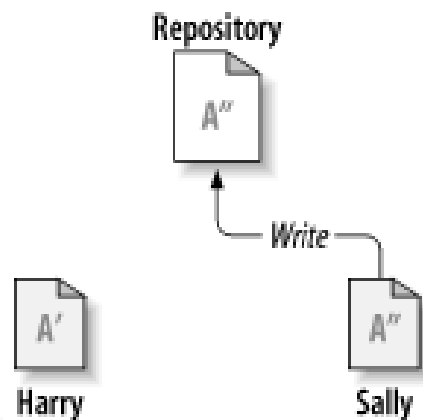
*Harry compares the latest version to his own*



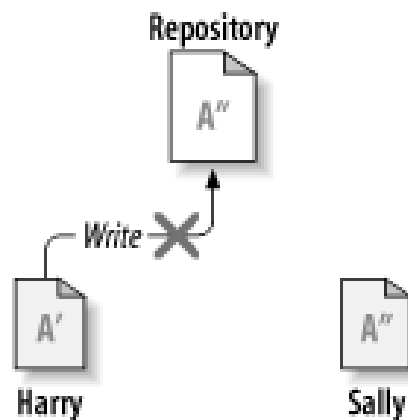
*A new merged version is created*



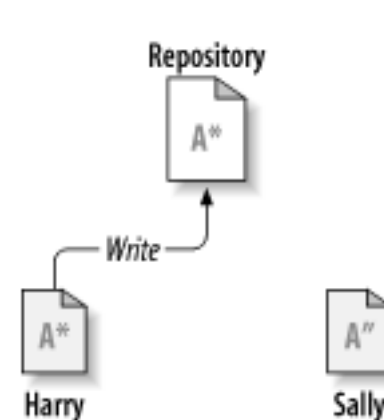
*Sally publishes her version first*



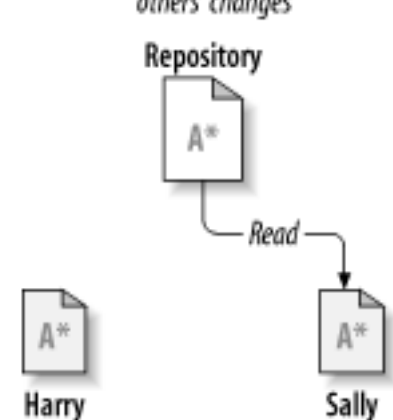
*Harry gets an "out-of-date" error*



*The merged version is published*



*Now both users have each others' changes*



- Que pasa si los cambios se sobrelapan/no son compatibles?:
- A esta situación se le llama **conflicto**:
- Harry será capaz de ver los cambios conflictivos y manualmente elegir entre ellos.
- El software no es capaz de automáticamente resolver los conflictos, los humanos son capaces de entender y hacer elecciones inteligentes. Posiblemente será necesaria una discusión académica con Sally.
- Suave en la práctica.
- Permite el trabajo en Paralelo.
- Conflictos infrecuentes en caso, el costo de tiempo para resolver los conflictos es menor que el tiempo muerto en la espera de “unlock”.

# Comunicación

- Poca comunicación ... conflictos semánticos/sintácticos.
- Ningún sistema puede obligar a los usuarios a comunicarse perfectamente.
- No es nada fácil detectar errores semánticos para el sistema.
- En la práctica “lockear” aparentemente inhibe la productividad más que alguna otra cosa.

# Metáfora del escenario

