.git* - .gitattributes

git

indica la naturaleza de cierto tipos de archivos para optimizar el control de cambios.

```
*.tgz binary
code/*.zip binary
```

- `diff`:Le indica a Git cómo comparar dos versiones del archivo.
  - Sin esto, Git intentaría comparar los archivos binarios (lo cual es inútil y consume mucha CPU).
  - Con diff=lfs, Git simplemente compara los hashes (huellas digitales) de los punteros para saber si el archivo ha cambiado o no.
- `merge`:Define qué hacer si dos personas modifican el mismo archivo al mismo tiempo.
  - Los archivos binarios no se pueden «mezclar» (no puedes fusionar media foto con otra media foto).
  - Este parámetro le dice a Git que, en caso de conflicto, no intente un auto-merge. Normalmente, Git LFS simplemente te pedirá que elijas una de las dos versiones.
- `-text`:El signo menos (-) delante de text significa desactivar el atributo de texto.
  - Evita que Git intente normalizar los finales de línea (LF vs CRLF).
  - Si Git intentara cambiar un «salto de línea» dentro de un video o un ejecutable para adaptarlo a Windows o Linux, corrompería el archivo. Con -text le dices: «No toques el contenido, trátalo como datos binarios puros».
- `binary`: La palabra binary es en realidad un macro (un atajo). Cuando escribes binary, Git lo traduce internamente como si hubieras escrito: `-text -diff`. Es decir: «No intentes arreglar los saltos de línea (`-text`) y no intentes mostrar las diferencias de contenido (`-diff`)».

renormalize

```
# 1. Definir el atributo
echo "*.bin binary" >> .gitattributes

# 2. Forzar a Git a tratar los existentes como binarios
git add --renormalize .

# 3. Tu script habitual
git commit -m "Update attributes and files"
git push origin master
```

```
# 1. Elimina todo del índice de Git de forma segura (sin borrar tus archivos del
disco)
git rm --cached -r .

# 2. Vuelve a añadir todos los archivos (Git los procesará aplicando el nuevo
.gitattributes)
git add .

git commit -m "Actualizar atributos de archivos binarios existentes"
```

LFS (filter)

- indicar que ficheros son tratados como LFS:

```
*.png filter=lfs diff=lfs merge=lfs -text
*.wav filter=lfs diff=lfs merge=lfs binary
```

- `filter`: Es el parámetro más importante. Le dice a Git que debe usar el «filtro» de LFS al mover archivos entre el disco y el área de preparación (staging).
    - Al hacer commit: El filtro intercepta el archivo, lo mueve a la carpeta LFS y crea un pequeño archivo de texto (puntero) en su lugar.
    - Al hacer checkout: El filtro busca el archivo real en la carpeta LFS (o lo descarga de Gitea) y lo pone en tu carpeta de trabajo.

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

Permanent link:
https://matebcn.eu/tech:git:dotfiles:gitattributes

Last update: **13/07/2026 10:57**