Git LFS Large files in Git

CONCEPTS

> Track a pattern before you add the files: git lfs track only affects files added afterwards, and the rule lives in .gitattributes, which you must commit.

Large file in working tree
↓ `git add (clean)`
Pointer file tiny text in Git
↓ `git push (pre-push)`
LFS server real contents
↓ `checkout (smudge)`
Working tree real file again

> Git itself stores only a small pointer; the contents are uploaded on push and downloaded on checkout.

SETUP

`git lfs install`
Set up LFS filters for your user

`git lfs env`
Show endpoint, paths and config

TRACKING

`git lfs track "<pattern>"`
Track files matching a pattern

`git lfs track`
List tracked patterns

`git lfs untrack "<pattern>"`
Stop tracking a pattern

.gitattributes entry

```
*.psd filter=lfs diff=lfs merge=lfs -
text
```

INSPECT

`git lfs ls-files`
List LFS files in this checkout

`git lfs ls-files -s`
Include object sizes

INSPECT (cont.)

`git lfs status`
LFS files changed or not yet pushed

`git lfs migrate info`
Size summary by file type

TRANSFER

`git lfs pull`
Fetch objects and update files

`git lfs fetch`
Download objects, keep files as is

`git lfs pull -I "<paths>"`
Only fetch matching paths

`git lfs push origin <branch>`
Upload objects for a branch

MIGRATE

`git lfs migrate import --include="<patterns>"`
Convert history to LFS

`git lfs migrate import --include="<patterns>" --everything`
Convert all refs

`git lfs migrate import --no-rewrite <files>`
Convert without rewriting history

> Migrate import/export rewrite history and change commit hashes. Commit or stash first, check the result, then force-push and tell collaborators to re-clone.

CLEAN UP

`git lfs prune`
Delete old local LFS objects

LOCKING

`git lfs lock <path>`
Lock a file against other users

`git lfs unlock <path>`
Release your lock