git-lfs(1)

NAME

git-lfs - Work with large files in Git repositories

SYNOPSIS

`git lfs` <command> [<args>]

DESCRIPTION

Git LFS is a system for managing and versioning large files in association with a Git repository. Instead of storing the large files within the Git repository as blobs, Git LFS stores special "pointer files" in the repository, while storing the actual file contents on a Git LFS server. The contents of the large file are downloaded automatically when needed, for example when a Git branch containing the large file is checked out.

Git LFS works by using a "smudge" filter to look up the large file contents based on the pointer file, and a "clean" filter to create a new version of the pointer file when the large file's contents change. It also uses a `pre-push` hook to upload the large file contents to the Git LFS server whenever a commit containing a new large file version is about to be pushed to the corresponding Git server.

COMMANDS

Like Git, Git LFS commands are separated into high level ("porcelain") commands and low level ("plumbing") commands.

High level porcelain commands

git-lfs-checkout(1)

Populate working copy with real content from Git LFS files.

git-lfs-completion(1)

Generate shell scripts for command-line tab-completion of Git LFS commands.

git-lfs-dedup(1)

De-duplicate Git LFS files.

git-lfs-env(1)

Display the Git LFS environment.

git-lfs-ext(1)

Display Git LFS extension details.

git-lfs-fetch(1)

Download Git LFS files from a remote.

git-lfs-fsck(1)

Check Git LFS files for consistency.

git-lfs-install(1)

Install Git LFS configuration.

git-lfs-lock(1)

Set a file as "locked" on the Git LFS server.

git-lfs-locks(1)

List currently "locked" files from the Git LFS server.

git-lfs-logs(1)

Show errors from the Git LFS command.

git-lfs-ls-files(1)

Show information about Git LFS files in the index and working tree.

git-lfs-migrate(1)

Migrate history to or from Git LFS

git-lfs-prune(1)

Delete old Git LFS files from local storage

git-lfs-pull(1)

Fetch Git LFS changes from the remote & checkout any required working tree files.

git-lfs-push(1)

Push queued large files to the Git LFS endpoint.

git-lfs-status(1)

Show the status of Git LFS files in the working tree.

git-lfs-track(1)

View or add Git LFS paths to Git attributes.

git-lfs-uninstall(1)

Uninstall Git LFS by removing hooks and smudge/clean filter configuration.

git-lfs-unlock(1)

Remove "locked" setting for a file on the Git LFS server.

git-lfs-untrack(1)

Remove Git LFS paths from Git Attributes.

git-lfs-update(1)

Update Git hooks for the current Git repository.

git-lfs-version(1)

Report the version number.

Low level plumbing commands

git-lfs-clean(1)

Git clean filter that converts large files to pointers.

git-lfs-filter-process(1)

Git process filter that converts between large files and pointers.

git-lfs-merge-driver(1)

Merge text-based LFS files

git-lfs-pointer(1)

Build and compare pointers.

git-lfs-post-checkout(1)

Git post-checkout hook implementation.

git-lfs-post-commit(1)

Git post-commit hook implementation.

git-lfs-post-merge(1)

Git post-merge hook implementation.

git-lfs-pre-push(1)

Git pre-push hook implementation.

git-lfs-smudge(1)

Git smudge filter that converts pointer in blobs to the actual content.

git-lfs-standalone-file(1)

Git LFS standalone transfer adapter for file URLs (local paths).

EXAMPLES

To get started with Git LFS, the following commands can be used.

1. Setup Git LFS on your system. You only have to do this once per user account:

```
git lfs install
```

2. Choose the type of files you want to track, for examples all `ISO` images, with git-lfs-track(1):

```
git lfs track "*.iso"
```

3. The above stores this information in gitattributes(5) files, so that file needs to be added to the repository:

```
git add .gitattributes
```

4. Commit, push and work with the files normally:

```
git add file.iso
git commit -m "Add disk image"
git push
```