NAME

git−lfs−fetch − Download all Git LFS files for a given ref

SYNOPSIS

git lfs fetch [options] [*remote* [*ref*...]]

DESCRIPTION

Download Git LFS objects at the given refs from the specified remote. See *DEFAULT REMOTE* and *DEFAULT REFS* for what happens if you don’t specify.

This does not update the working copy.

OPTIONS

−I *paths* **−−include=***paths*

Specify lfs.fetchinclude just for this invocation; see *INCLUDE AND EXCLUDE*

−X *paths* **−−exclude=***paths*

Specify lfs.fetchexclude just for this invocation; see *INCLUDE AND EXCLUDE*

−−recent

Download objects referenced by recent branches & commits in addition to those that would otherwise be downloaded. See *RECENT CHANGES*

−−all

Download all objects that are referenced by any commit reachable from the refs provided as arguments. If no refs are provided, then all refs are fetched. This is primarily for backup and migration purposes. Cannot be combined with −−recent or −−include/−−exclude. Ignores any globally configured include and exclude paths to ensure that all objects are downloaded.

−−prune −p

Prune old and unreferenced objects after fetching, equivalent to running **git lfs prune** afterwards. See git−lfs−prune(1) for more details.

INCLUDE AND EXCLUDE

You can configure Git LFS to only fetch objects to satisfy references in certain paths of the repo, and/or to exclude certain paths of the repo, to reduce the time you spend downloading things you do not use.

In gitconfig, set **lfs.fetchinclude** and **lfs.fetchexclude** to comma−separated lists of paths to include/exclude in the fetch. Only paths which are matched by **fetchinclude** and not matched by **fetchexclude** will have objects fetched for them.

Note that using the command−line options **−I** and **−X** override the respective configuration settings. Setting either option to an empty string clears the value.

Examples:

○ **git config lfs.fetchinclude "textures,images/foo*"**

This will only fetch objects referenced in paths in the textures folder, and files called foo* in the images folder

○ **git config lfs.fetchinclude "*.jpg,*.png,*.tga"**

Only fetch JPG/PNG/TGA files, wherever they are in the repository

○ **git config lfs.fetchexclude "media/reallybigfiles"**

Don’t fetch any LFS objects referenced in the folder media/reallybigfiles, but fetch everything else

○ **git config lfs.fetchinclude "media"**
git config lfs.fetchexclude "media/excessive"

Only fetch LFS objects in the ’media’ folder, but exclude those in one of its subfolders.

DEFAULT REMOTE

Without arguments, fetch downloads from the default remote. The default remote is the same as for **git fetch**, i.e. based on the remote branch you’re tracking first, or origin otherwise.

DEFAULT REFS

If no refs are given as arguments, the currently checked out ref is used. In addition, if enabled, recently changed refs and commits are also included. See *RECENT CHANGES* for details.

RECENT CHANGES

If the **−−recent** option is specified, or if the gitconfig option **lfs.fetchrecentalways** is true, then after the current ref (or those in the arguments) is fetched, we also search for 'recent' changes to fetch objects for, so that it's more convenient to checkout or diff those commits without incurring further downloads.

What changes are considered 'recent' is based on a number of gitconfig options:

- **lfs.fetchrecentrefsdays** If non−zero, includes branches which have commits within N days of the current date. Only local refs are included unless lfs.fetchrecentremoterefs is true. The default is 7 days.
- **lfs.fetchrecentremoterefs** If true, fetches remote refs (for the remote you're fetching) as well as local refs in the recent window. This is useful to fetch objects for remote branches you might want to check out later. The default is true; if you set this to false, fetching for those branches will only occur when you either check them out (losing the advantage of fetch −−recent), or create a tracking local branch separately then fetch again.
- **lfs.fetchrecentcommitsdays** In addition to fetching at branches, also fetches changes made within N days of the latest commit on the branch. This is useful if you're often reviewing recent changes. The default is 0 (no previous changes).
- **lfs.fetchrecentalways** Always operate as if −−recent was provided on the command line.

EXAMPLES

- Fetch the LFS objects for the current ref from default remote

  git lfs fetch

- Fetch the LFS objects for the current ref AND recent changes from default remote

  git lfs fetch −−recent

- Fetch the LFS objects for the current ref from a secondary remote 'upstream'

  git lfs fetch upstream

- Fetch all the LFS objects from the default remote that are referenced by any commit in the **main** and **develop** branches

  git lfs fetch −−all origin main develop

- Fetch the LFS objects for a branch from origin

  git lfs fetch origin mybranch

- Fetch the LFS objects for 2 branches and a commit from origin

  git lfs fetch origin main mybranch e445b45c1c9c6282614f201b62778e4c0688b5c8

SEE ALSO

git−lfs−checkout(1), git−lfs−pull(1), git−lfs−prune(1).

Part of the git−lfs(1) suite.