

How to I disable git-lfs?

Asked 1 year, 5 months ago Active 1 year, 5 months ago Viewed 4k times



4

I have a repository on bitbucket that is using LFS. Since using it for some time, I've decided to move the repository back to a space under my control. The only reason I used LFS in the first place was to effectively double my repository size limit (as files in LFS go in a separate bucket) but now I'm moving it, I no longer need to do this.



I need a way to trawl through the entire git history, removing all traces of the work git lfs does (so all files are committed 'normally'). Once this is done, I intend to force push to the new repository.



3

I've done quite a bit of searching, and come across [suggested solutions](#) but I don't understand how to implement/run them because they are high-level.

How do I wave goodbye to git lfs?

git

git-lfs

asked Feb 9 '18 at 5:31



Shadow

5,103 2 28 43

That would be something like github.com/git-lfs/git-lfs/issues/326#issuecomment-104881826, I suppose? – VonC Feb 9 '18 at 5:33

Let us [continue this discussion in chat](#). – Shadow Feb 9 '18 at 5:42

1 Answer



Update current commit only

6

If you want to move off LFS, but are not so worried about fixing the entire git history, you can do the following;





```
git lfs uninstall
touch **/*
git commit -a
```

This will uninstall LFS support, touch every single file (so that git recognises that it has changed) then commit them all. If you like you could be more specific (ie, `**/*.png` for example). Note that using `**` requires extended glob support enabled (`shopt -s globstar` on bash)

Update entire history

This worked for me - but it throws lots of errors (I *think* I'm getting an error for every commit that a file hasn't been added to LFS in) and takes a long time (roughly 2-3 seconds per commit).

```
git lfs uninstall
git filter-branch -f --prune-empty --tree-filter '
    git lfs checkout
    git lfs ls-files | cut -d " " -f 3 | xargs touch
    git rm -f .gitattributes
    git lfs ls-files | cut -d " " -f 3 | git add
' --tag-name-filter cat -- --all
```

It uninstalls git LFS support (theoretically preventing LFS from messing with the index) then for each commit it makes sure the LFS files are checked out properly, then touches them all (so git realises they have changed), removes the settings for LFS found in `.gitattributes` so that when cloning it doesn't keep trying to use LFS, then adds the real file to the index.

After you do the above, you will need to do a force push. Naturally, that'll throw anyone else working on your repo into a detached head state - so doing this during a code freeze is wise. Afterwards, it's probably easiest to get everyone to do a fresh clone.

edited Feb 12 '18 at 4:34

answered Feb 9 '18 at 6:09



Shadow

5,103 2 28 43

Got a question that you can't ask on public Stack Overflow? [Learn more](#) about sharing private information with Stack Overflow for Teams.

