

Githubでサイズ の大きなファイルを扱う話

2017 10 14 mishima.syk #11
@iwatobipen

自己紹介



A Twitter profile card for the user @iwatobipen. The background is a photograph of a colony of penguins on a snowy, icy landscape under a clear blue sky. At the top center is a circular profile picture showing two penguin chicks. Below the profile picture, the name "pen" is displayed in a bold, white, sans-serif font. Underneath the name is the Twitter handle "@iwatobipen" in a smaller, white, sans-serif font. Below the handle is a bio in English: "Medicinal Chemist. Interested in Chemoinformatics, Running." followed by a line of Japanese text: "Opinions and tweets are my own. 流れるつぶやきはあくまで個人の見解です。". At the bottom of the card is the location "静岡県東部のほう。" followed by a link to the user's website: "・ iwatobipen.wordpress.com".

pen
@iwatobipen
Medicinal Chemist. Interested in Chemoinformatics, Running.
Opinions and tweets are my own. 流れるつぶやきはあくまで個人の見解です。
静岡県東部のほう。 ・ iwatobipen.wordpress.com

Gitの問題

- バイナリファイルの差分は記録できない



- 大きなファイルを使うとローカルリポジトリが肥大化する

<https://help.github.com/articles/what-is-my-disk-quota/>

Gitの問題

File and repository size limitations

We recommend repositories be kept under 1GB each. This limit is easy to stay within if large files are kept out of the repository. If your repository exceeds 1GB, you might receive a polite email from GitHub Support requesting that you reduce the size of the repository to bring it back down.

In addition, we place a strict limit of files exceeding 100 MB in size. For more information, see "[Working with large files](#)."

Note: If you add a file to a repository via a browser, the file can be no larger than 25 MB. For more information, see "[Adding a file to a repository](#)".

<https://help.github.com/articles/what-is-my-disk-quota/>

**100MB以上のファイルプッシュすんなよ。
リポジトリ1GB以上にすんなよ。**

<https://help.github.com/articles/what-is-my-disk-quota/>



だって、、、

- 深層学習とか色々ファイルサイズ大きくないっすか。
- データサイズってだんだん増えますよね、、、

そこでGit Lfs!

Git Lfsって？

Lfsの名前のまんま。

Git Large File Storage

[Docs](#) [Downloads](#) [Source](#)

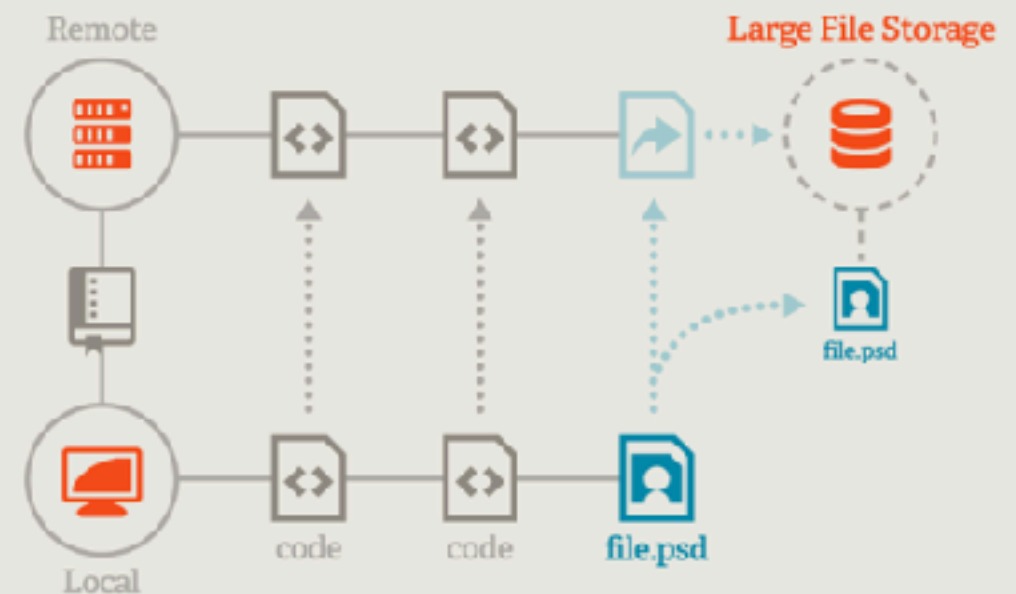
An open source Git extension for versioning large files

Git Large File Storage (LFS) replaces large files such as audio samples, videos, datasets, and graphics with text pointers inside Git, while storing the file contents on a remote server like GitHub.com or GitHub Enterprise.

 **Download** v2.3.3 (Mac)

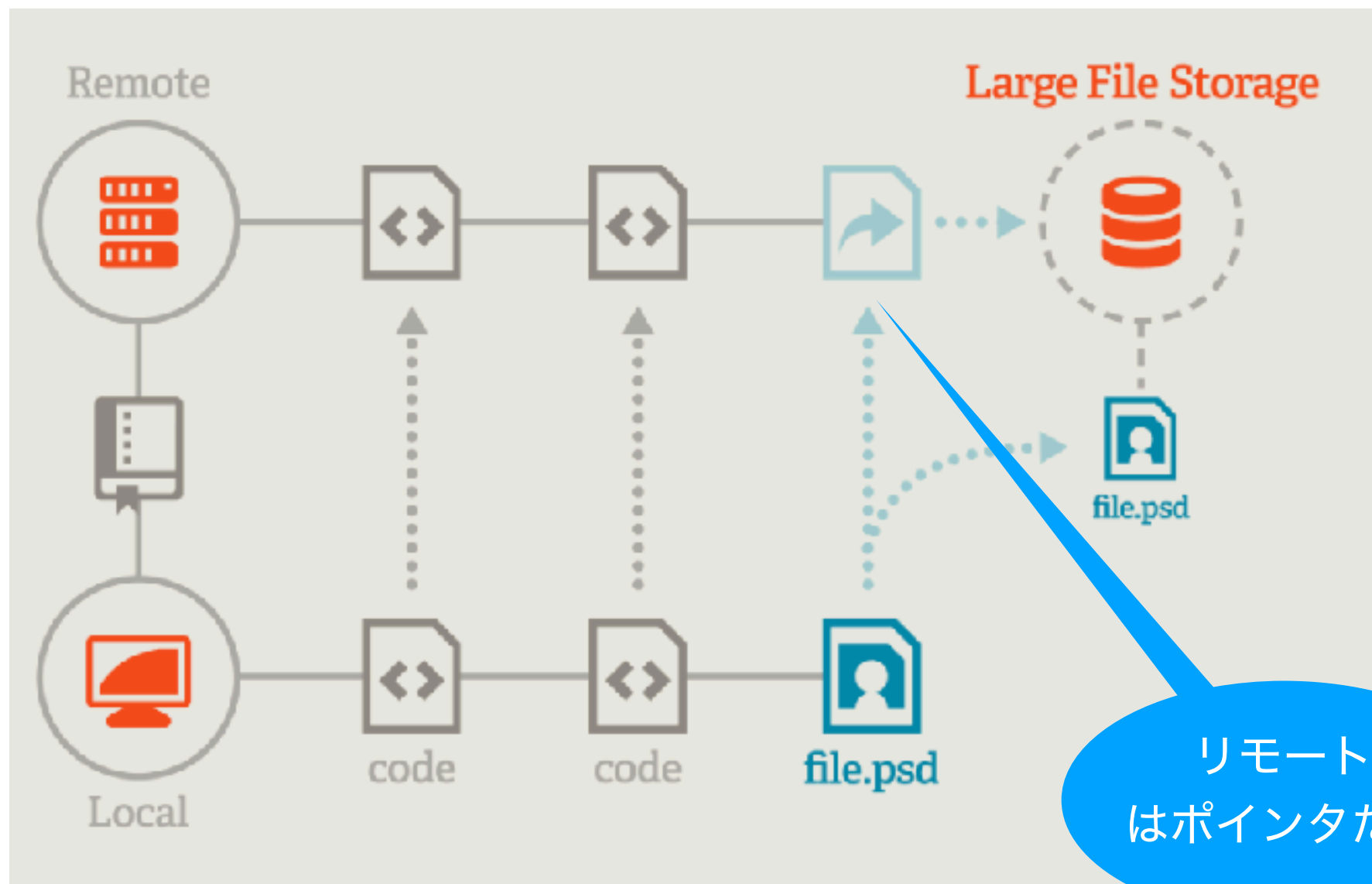
Homebrew: `brew install git-lfs`

MacPorts: `port install git-lfs`



Git Lfsって？

Gitはコード（文字）を管理する目的だったのでバイナリファイルは不得手。
そこを解消するソリューションがGit Lfs



2GBまでは無料プラン内



About Git Large File Storage

Git LFS lets you store files up to 2 GB in size.

Git LFS handles large files by storing references to the file in the repository, but not the actual file itself. To work around Git's architecture, Git LFS creates a *pointer file* which acts as a reference to the actual file (which is stored somewhere else). GitHub manages this pointer file in your repository. When you clone the repository down, GitHub uses the pointer file as a map to go and find the large file for you.

As an analogy, say you walk into a restaurant wearing an enormous coat. You hand your coat over to the attendant, and it's exchanged for a ticket that identifies where the coat is stored. After you finish your meal, you hand your ticket to the attendant, who retrieves your coat from storage and returns it to you. Git LFS works the same way.

You can also use Git LFS with GitHub Desktop. For more information about cloning Git LFS repositories in GitHub Desktop, see "[Cloning a repository from GitHub to GitHub Desktop](#)."

Article versions

[GitHub.com](#)

[GitHub Enterprise 2.11](#)

[GitHub Enterprise 2.10](#)

[GitHub Enterprise 2.9](#)

[GitHub Enterprise 2.8](#)

<https://help.github.com/articles/about-git-large-file-storage/>

セッティング手順

OSx

- brew install git-lfs

Debian

- sudo apt-get install git-lfs

windows

- <https://github.com/git-lfs/git-lfs/releases> (σ・∀・)σゲ ッツ!!

<https://github.com/git-lfs/git-lfs/wiki/Installation>

使い方

```
iwatobipen$ git lfs track '*.sdf'
```

Tracking “*.sdf”

上のコマンドでトラッキングするファイルを指定するだけ

```
iwatobipen$ cat .gitattributes
```

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

```
iwatobipen$ git lfs track
```

Listing tracked patterns

about/*.sdf (about/.gitattributes)

push push push


```
iwatobipen$ git push origin temp
Git LFS: (1 of 1 files) 367.39 KB / 367.39 KB
Counting objects: 5, done.
Delta compression using up to 8 threads.
Compressing objects: 100% (4/4), done.
Writing objects: 100% (5/5), 577 bytes | 577.00 KiB/s, done.
Total 5 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/Mishima-syk/11.git
4e5b2ed..024c70e temp -> temp
```

LFSに保存されている

Mishima-syk / 11 Unwatch 3

[Code](#) [Issues 0](#) [Pull requests 0](#) [Projects 0](#) [Wiki](#) [Insights](#)

Branch: temp 11 / [about](#) / CHEMBL1827733_5HT2A.sdf

 iwatobipen add lfs sample

1 contributor

367 KB [?](#) Stored with Git LFS

[View Raw](#)

まとめ



注意点

Git LFS を利用しているリポジトリは、一見すると普通の Git リポジトリと同じに見えます。

誤って Git LFS 非インストール環境下でリポジトリが clone され、ラージファイルの追加が push された場合、ポインタ管理が機能せずに実体そのままりポジトリ管理されてしまいます。

さらに、Git LFS インストール環境からは、ポインタとして見えるはずのラージファイルが実体として見えることで不整合が発生してしまい、ニッチもサッチもいかない状況になります。

