

sparse checkout でリポジトリの一部だけを持ってくる

created

2020-07-17

[Tweet](#)

last change

2020-07-18 13:59:24 +0000

偉大なリポジトリの一部だけを利用したい

Gitで特定のファイルやディレクトリだけcloneしたい

おそらくあなたがやりたいのは [Sparse checkout](#) ではないでしょうか

— [Gitで特定のファイルやディレクトリだけcloneしたい - Qiita](#)

先人の情報によれば「とりあえず普通に clone する」という例が多いのですが、ファイル数が多い場合など、すべてを clone したくない場合もあります。

そのような場合は下記のようにして sparse checkout を有効にします。

— [git sparse checkout で clone せずに一部のサブディレクトリだけを pull/checkout する](#)

リポジトリの内、必要となるのはコンパイルされたヘッダファイル 1 つだけ ということがそれなりにあるので sparse checkout ができるとそういう時に便利そうだ。

ただ、調査した感じでは大規模なリポジトリの一部だけを利用するという用途の場合、sparse checkout と submodule を組み合わせる必要があるらしい。

わからないことは 1 つずつチェックしていく。今回は sparse checkout だけを使ってみる

参考:

- [Gitで特定のファイルやディレクトリだけcloneしたい - Qiita](#)
- [git sparse checkout で clone せずに一部のサブディレクトリだけを pull/checkout する](#)

sparse checkout 単体の使いみち

調べて考えついた sparse checkout 単体の利用方法を挙げてみる

- リポジトリの一部をローカルにコピーしたい
 - コピーしたところだけローカルで動作確認する
 - コピーしたところだけ修正して Pull Request を送る
 - コピーしたところと、自分のところのモジュールの結合テストをする

こういう状況の時に sparse checkout 単体で利用することができそうだった。

sparse checkout を利用してみる

1. まずは空っぽのローカルリポジトリを作成

```
git init .
```

2. ローカルリポジトリ内で以下のコマンドを実行。 gitconfigファイルに `sparsecheckout=true` の1行が追加されます。

```
git config core.sparsecheckout true
```

3. `.git/info`以下に`sparse-checkout`という名前のファイルを作成。

```
vi .git/info/sparse-checkout
```

4. 必要なファイルやディレクトリを記述していきます。 記述方法は`.gitignore`と同じで、ファイル名やディレクトリ名の先頭に「!」を付けると除外指定もできます。

```
hoge.txt
/fuga
!/fuga/piy0
```

5. リモートリポジトリを指定します。

```
git remote add origin XXXX
```

6. 最後にpull

```
git pull origin master
```

これで指定したファイルやディレクトリだけ引っ張ってこれます。 Subversionのcheckoutみたいなやつですね。 重たくてcloneに時間がかかるリポジトリの一部分だけ欲しい時に便利です。

— [Gitで特定のファイルやディレクトリだけcloneしたい - Qiita](#)

やってみる。まずはリポジトリの用意をする。内容はこんな感じにした。

```
$ ls
A.md          B.md          C.md          D.md          README.md
```

これを別のディレクトリに sparse checkout する。 まずはディレクトリを作って `git init` する

```
$ mkdir sparse-checkout
$ cd sparse-checkout
$ ls
$ git init
Initialized empty Git repository in /Users/test_user/Project/public_git/sparse-checkout/.git/
```

sparse checkout を有効にする

```
$ git config core.sparsecheckout true
$ git config -l
credential.helper=osxkeychain
filter.lfs.required=true
filter.lfs.clean=git-lfs clean %f
filter.lfs.smudge=git-lfs smudge %f
user.name=****
user.email=****
core.excludesfile=~/.gitignore
core.repositoryformatversion=0
core.filemode=true
core.bare=false
core.logallrefupdates=true
core.ignorecase=true
core.precomposeunicode=true
core.sparsecheckout=true
```

remote に origin としてさっきのリポジトリを追加する。

```
$ git remote add origin https://github.com/leico/sparse_checkout_master
lev@leico_studio 20-07-18 13:14
~/Project/public_git/sparse-checkout
$ git config -l
credential.helper=osxkeychain
filter.lfs.required=true
filter.lfs.clean=git-lfs clean %f
filter.lfs.smudge=git-lfs smudge %f
user.name=*****
user.email=*****
core.excludesfile=~/.gitignore
core.repositoryformatversion=0
core.filemode=true
core.bare=false
core.logallrefupdates=true
core.ignorecase=true
core.precomposeunicode=true
core.sparsecheckout=true
remote.origin.url=https://github.com/leico/sparse_checkout_master
remote.origin.fetch=+refs/heads/*:refs/remotes/origin/*
```

.git/info/sparse-checkout を編集する。中にあるファイルの内、 C.md だけを取得するようにする。

```
$ vimr .git/info/sparse-checkout
```

```
C.md
```

origin からファイルを取得する。

```
$ git pull origin master
remote: Enumerating objects: 7, done.
remote: Counting objects: 100% (7/7), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 7 (delta 0), reused 7 (delta 0), pack-reused 0
Unpacking objects: 100% (7/7), done.
From https://github.com/leico/sparse_checkout_master
 * branch          master      -> FETCH_HEAD
 * [new branch]    master      -> origin/master
$ ls
C.md
```

.git/info/sparse-checkout に記述した通り、 C.md だけが取得できた。

C.md を編集して push してみる。

```
$ vimr C.md

# This is C changed
```

最後の **changed** を追加した。

```
$ git add .
$ git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   C.md

$ git commit
[master 70173bf] change C
 1 file changed, 1 insertion(+), 1 deletion(-)
$ git push --set-upstream origin master
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 12 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 275 bytes | 275.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/leico/sparse_checkout_master
    d6f14d9..70173bf  master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.
```

push できた。 *C.md* だけを取得して、変更して、その内容を反映させることができた。元のディレクトリで変更を pull して確認してみる。

```
$ cd sparse_checkout_master
$ git pull
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (1/1), done.
remote: Total 3 (delta 1), reused 3 (delta 1), pack-reused 0
Unpacking objects: 100% (3/3), done.
From https://github.com/leico/sparse_checkout_master
    d6f14d9..70173bf  master      -> origin/master
Updating d6f14d9..70173bf
Fast-forward
  C.md | 2 +-
  1 file changed, 1 insertion(+), 1 deletion(-)
```

```
$ cat C.md
# This is C changed
```

反映されている。

[Tweet](#)

[Atom](#)



contact

[@leico](#)

[twitter](#)

[web](#)

oishi[at]signalcompose.com

Technical Note Vol.1 2 3

Max for Live の項目が本になりました。

[Booth](#)

[Stores](#)

何かありましたらお気軽に。調査、開発
などのご依頼も承ります。

Merchandise/Schwags

支援してくれるとうれしい

