

Git

Git: Revision control

1. Backup your code on a distant server.
2. Share your code easily with everyone.
3. Collaborate easily with anyone.
4. Remember every change and save the whole history since start of time.

The old fashion way

1. Type the name of your collaborator in your email client.
2. Find the thread that pertains to the script you want.
3. See the history in the conversation.

Git: A short history



GUI or CLI

```
3. Local computer: IPython: sinclair/test (bash)
x #1 sinclair@blastocyst python_ebc_2016 (master) $ git pull
remote: Counting objects: 23, done.
remote: Compressing objects: 100% (11/11), done.
remote: Total 23 (delta 14), reused 20 (delta 11), pack-reused 0
Unpacking objects: 100% (23/23), done.
From github.com:xapple/python_ebc_2016
 570c3d4..1d8e1c0 master    -> origin/master
Updating 570c3d4..1d8e1c0
Fast-forward
 day_01/exercise_results.py
| 173 ++++++
 day_03.zip
es
 day_03/slides_and_tutorials/classes_glossary.p
3 files changed, 173 insertions(+)
 create mode 100644 day_01/exercise_results.py
 create mode 100644 day_03/slides_and_tutorials
> sinclair@blastocyst python_ebc_2016 (master) $
> sinclair@blastocyst python_ebc_2016 (master) $
> sinclair@blastocyst python_ebc_2016 (master) $
```

The screenshot shows the GitHub web interface for a pull request. The repository is 'xapple/python_ebc_2016'. The pull request is titled 'added exercise day 1 solutions' and was created by 'alper1976' 2 days ago. The pull request is currently open and shows a diff for the file 'day_01/exercise_results.py'. The diff shows 173 lines added, indicated by green plus signs and a green bar on the right. The code content is visible, showing a Python script for counting vowels in a string.

Filter Repositories

GitHub

- envonautics.gi...
- gefes
- ilumitag
- ipyreplay
- osx-env-sync
- pacfm
- pacfm_eval
- plumbing
- python_ebc_2...
- pythoncourse
- seqenv

Other

- envonautics
- Manuscript
- tutorial

Compare

master

added exercise day 1 solutions

2 days ago by alper1976

glossary

3 days ago by Idunolt

glossary

3 days ago by Idunolt

day03_lecturesback

3 days ago by Idunolt

day03_lecturesback

3 days ago by Idunolt

11

day03

3 days ago by Idunolt

2

day03

3 days ago by Idunolt

2

finalday03

3 days ago by Idunolt

2

added exercise day 1 solutions

alper1976 · 1d8e1c0 · 2 days ago

day_01/exercise_results.py

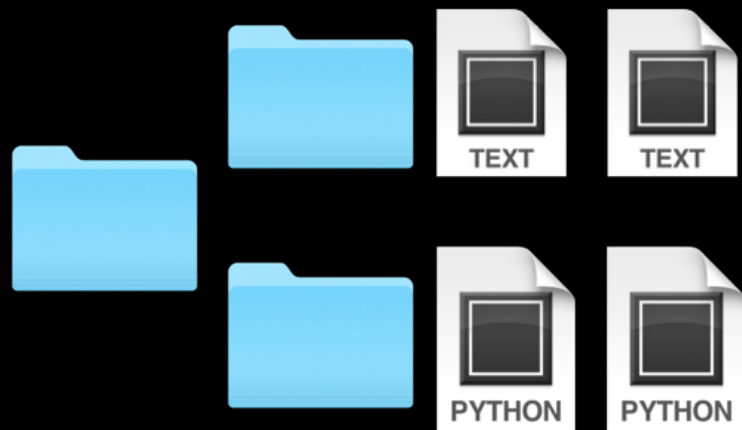
@@ -0,0 +1,173 @@

```
1 + ##### Exercise 1
2 + #####
3 +
4 + s = "ajskndna"
5 + num_vowels=0
6 + for char in s:
7 +     if char in "aeiou":
8 +         num_vowels = num_vowels+1
9 + print num_vowels,"vowels"
10 +
11 + #If one needs to account for upper string
12 + #character you can
13 + #look in a lower case version of the string
```

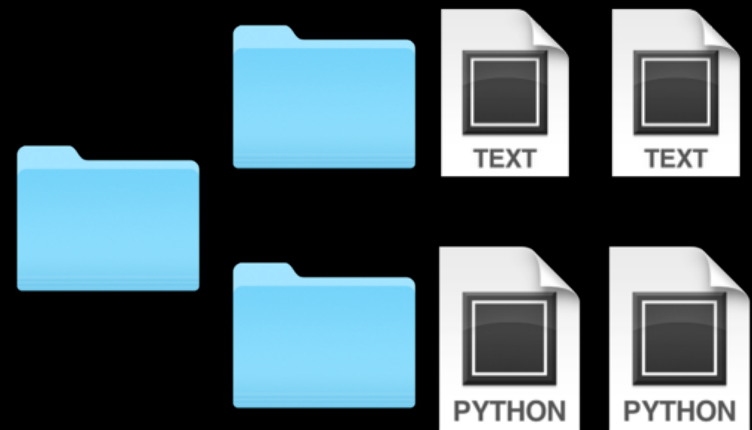
The file system



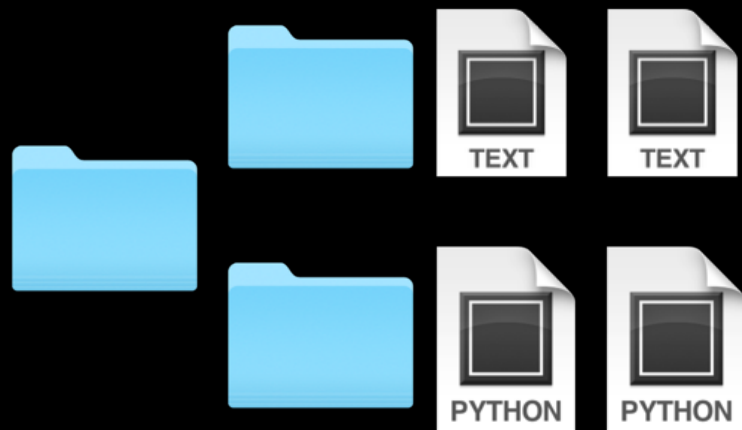
Git's memory



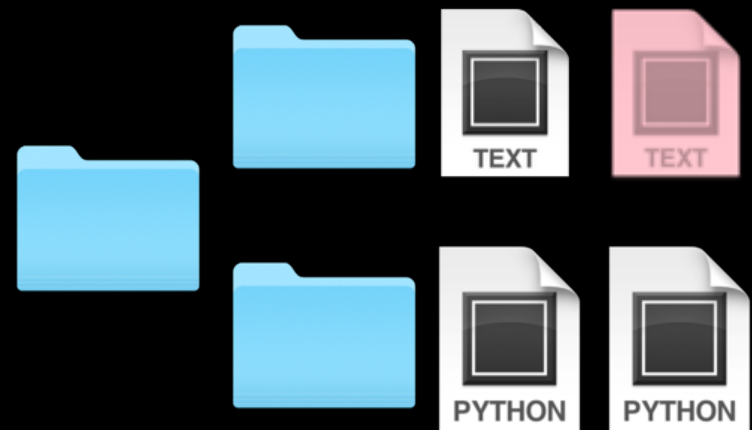
The file system



Git's memory

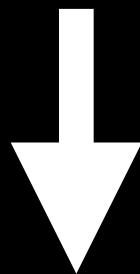


The file system



Git's memory

2016-06-04



xapple

2016-06-05

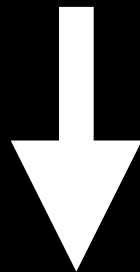


The file system



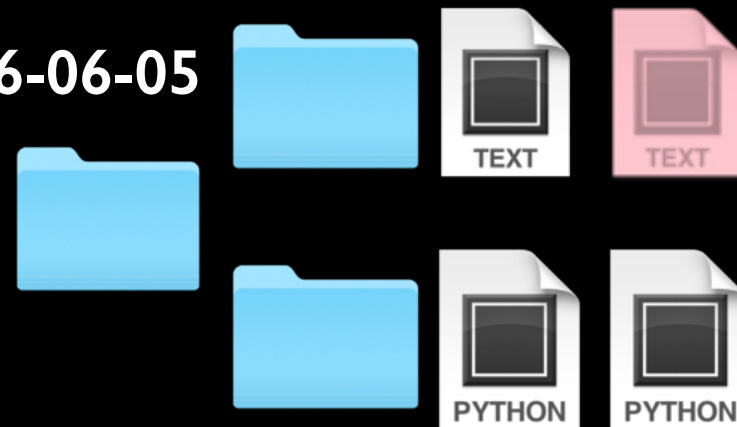
Git's memory

2016-06-04

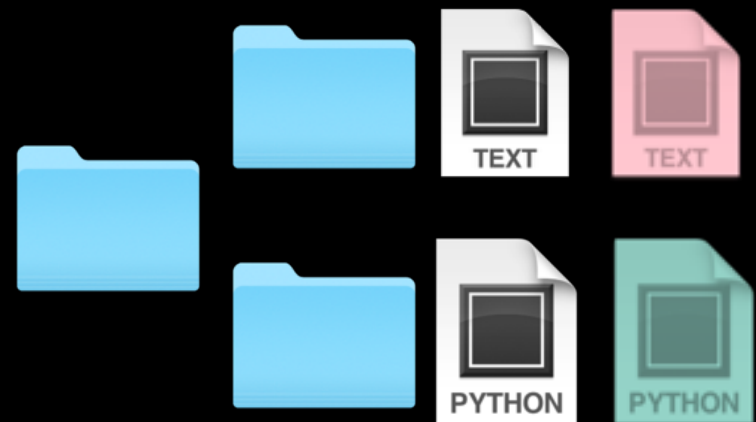


xapple

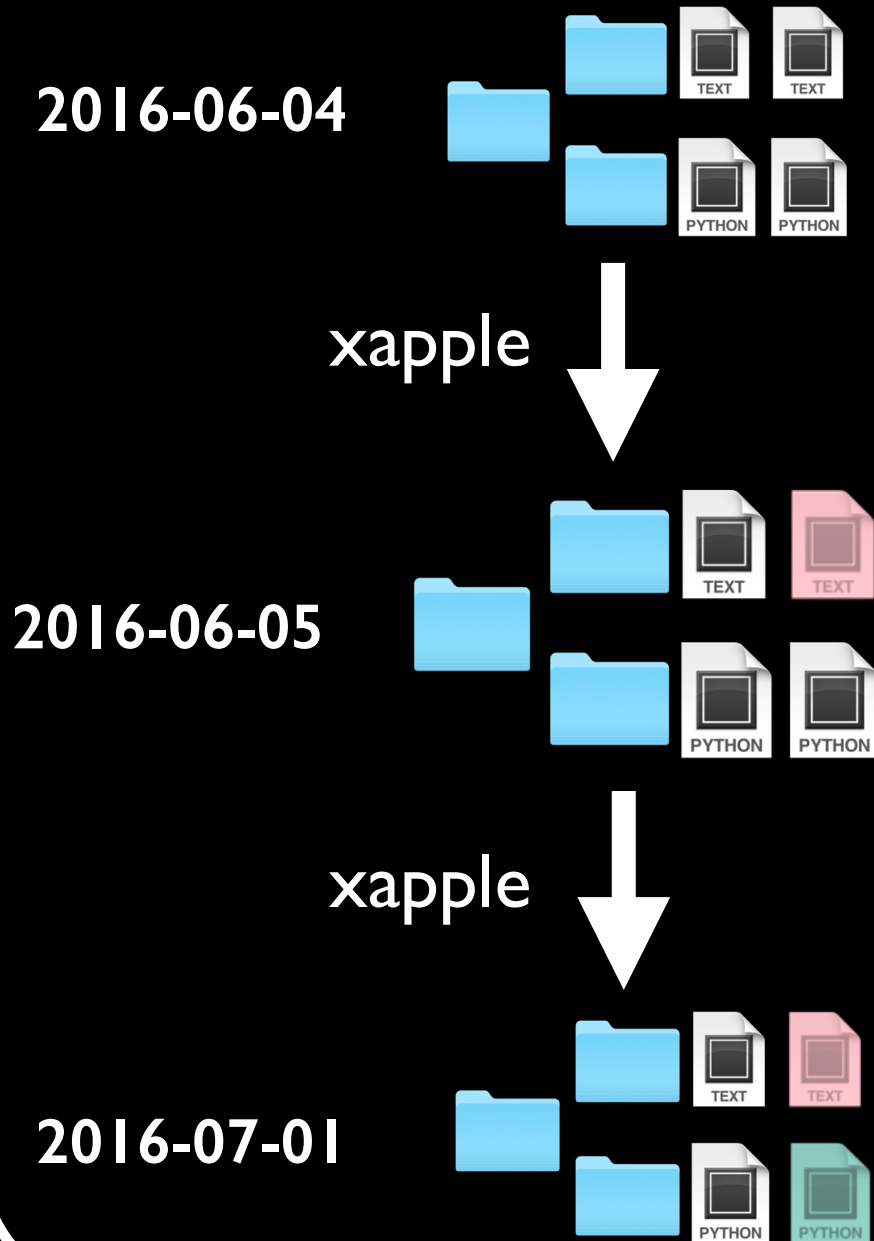
2016-06-05



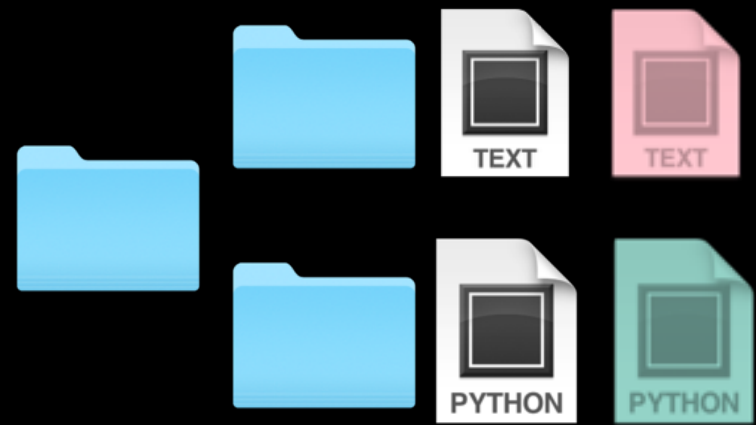
The file system



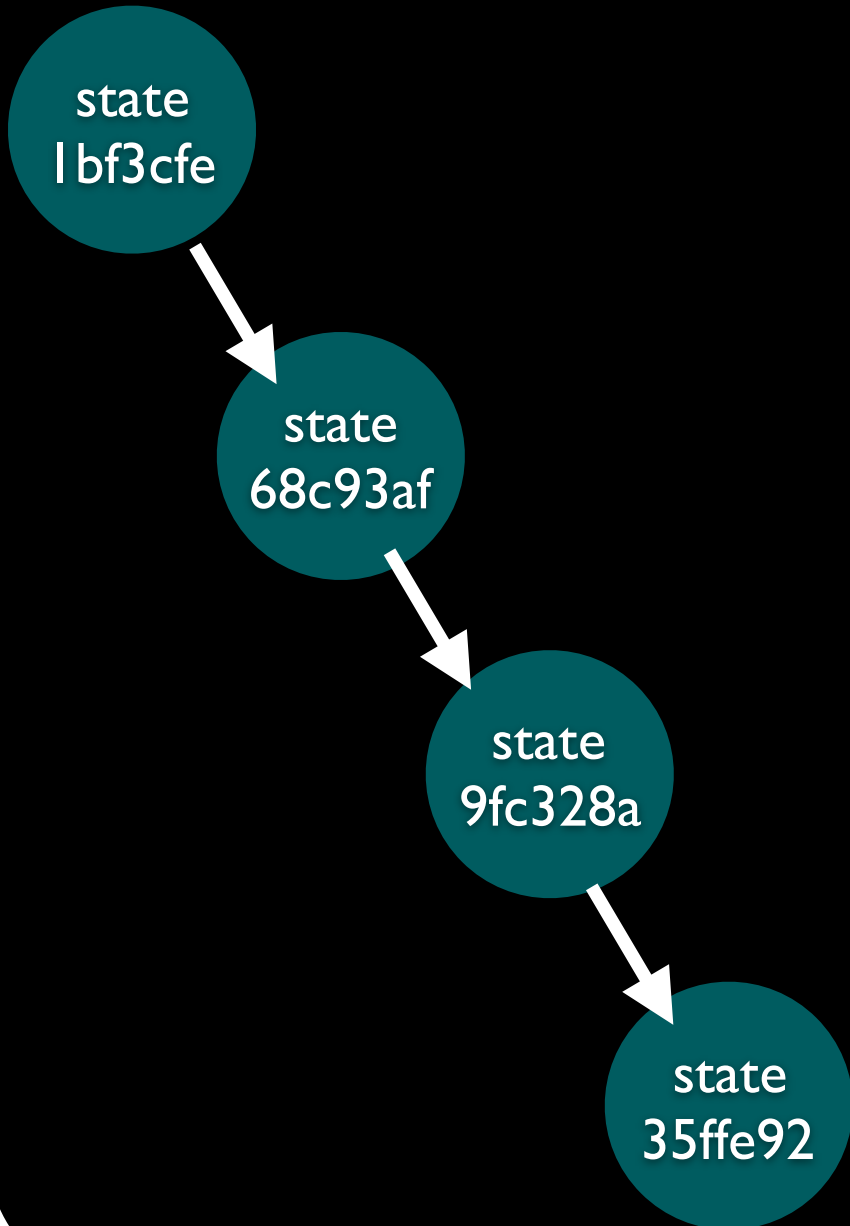
Git's memory



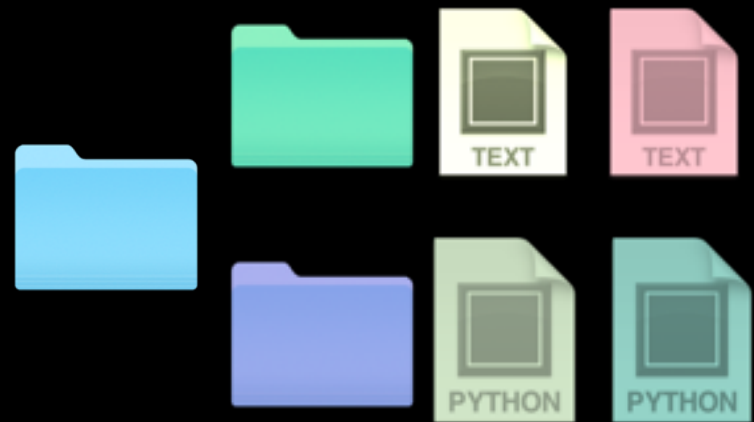
The file system



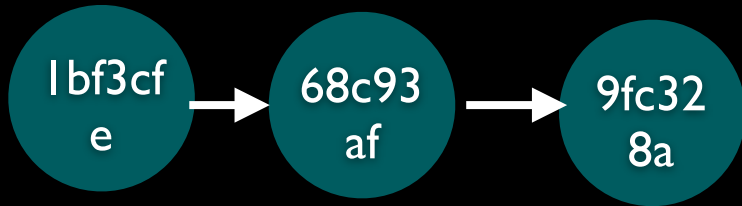
Git's memory



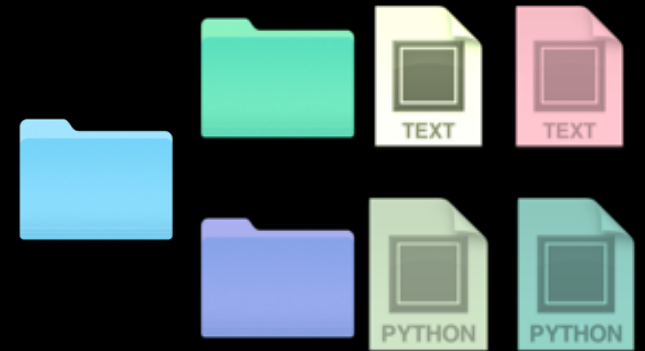
The file system



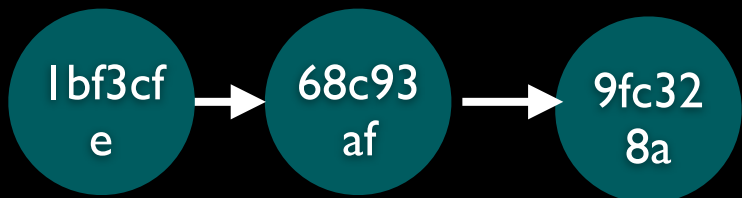
Bob's git repository



Bob's file system



Alice's git repository



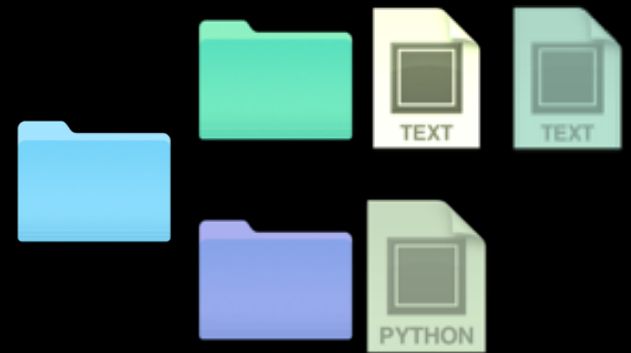
Alice's file system



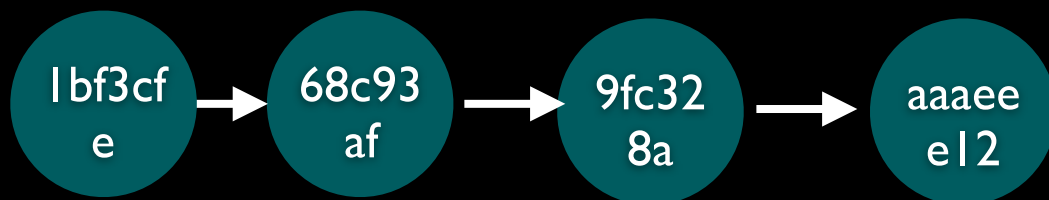
Bob's git repository



Bob's file system

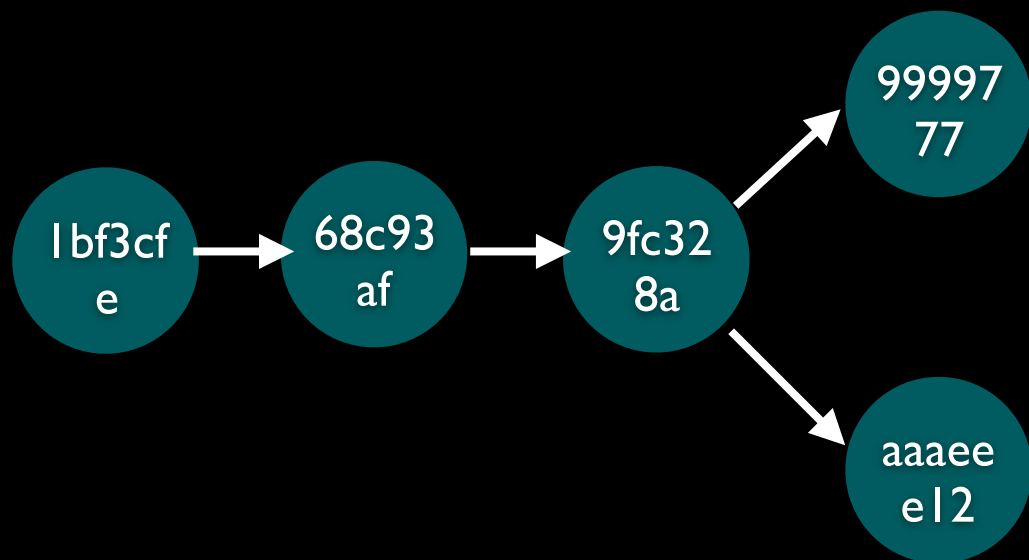


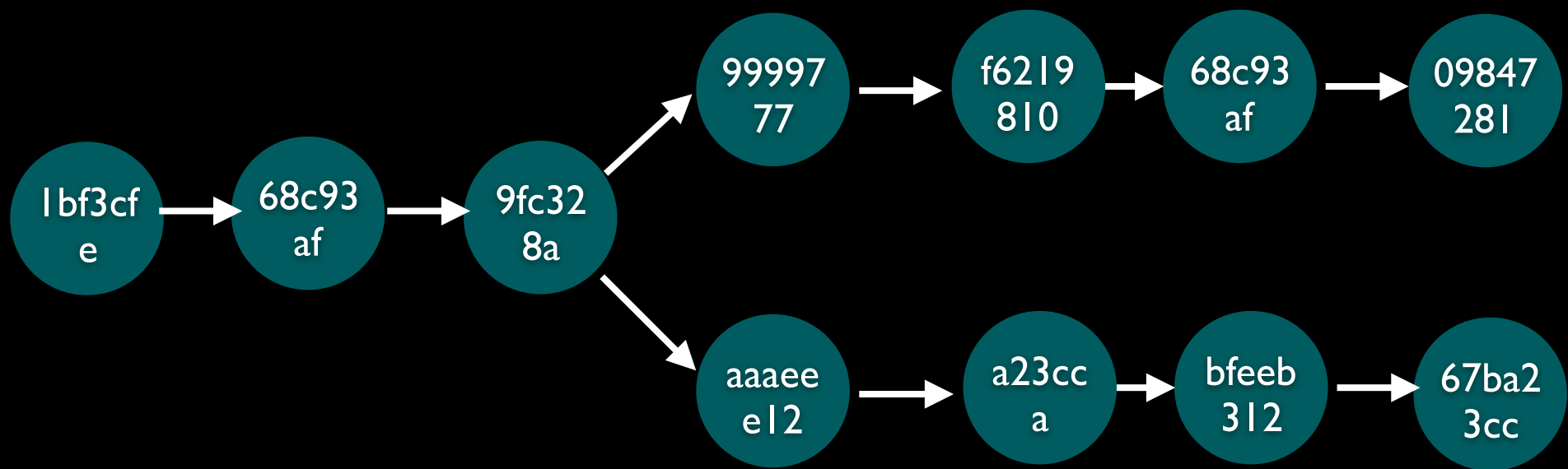
Alice's git repository



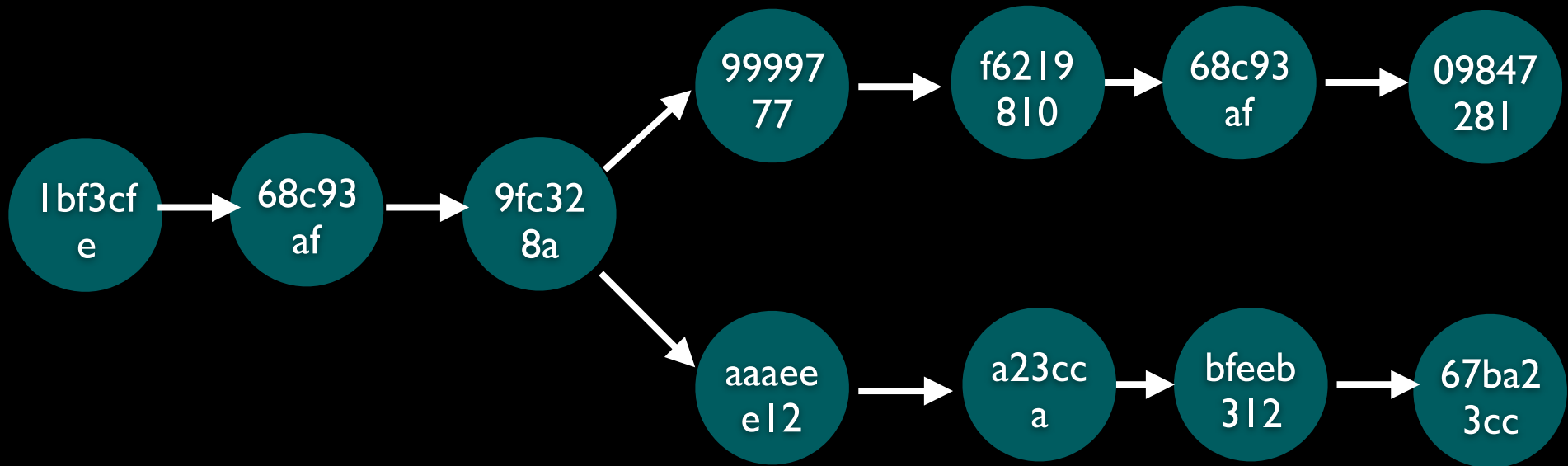
Alice's file system



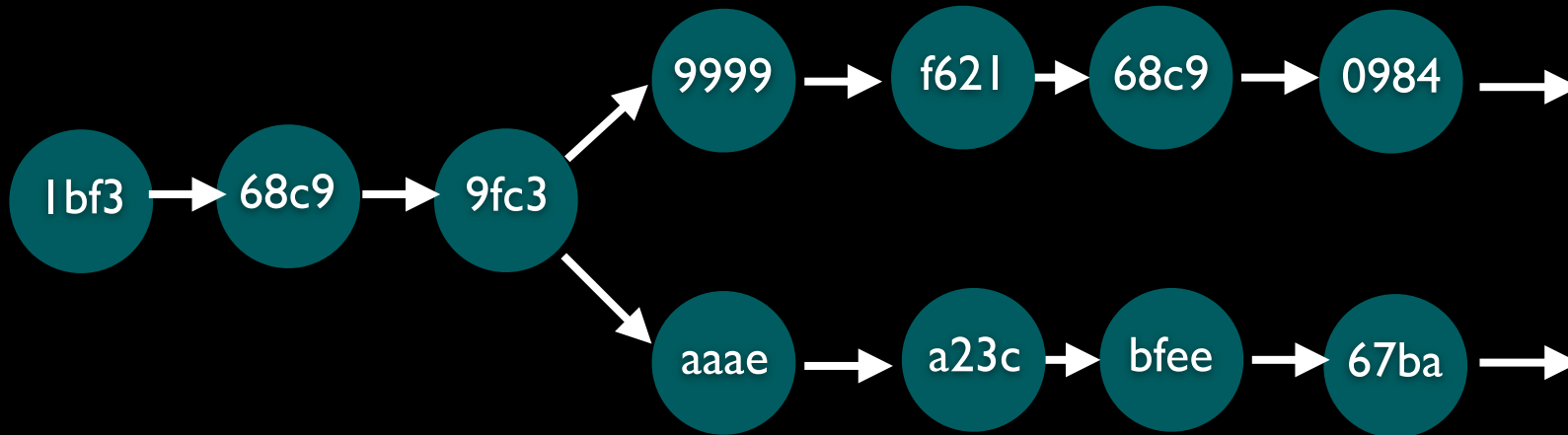




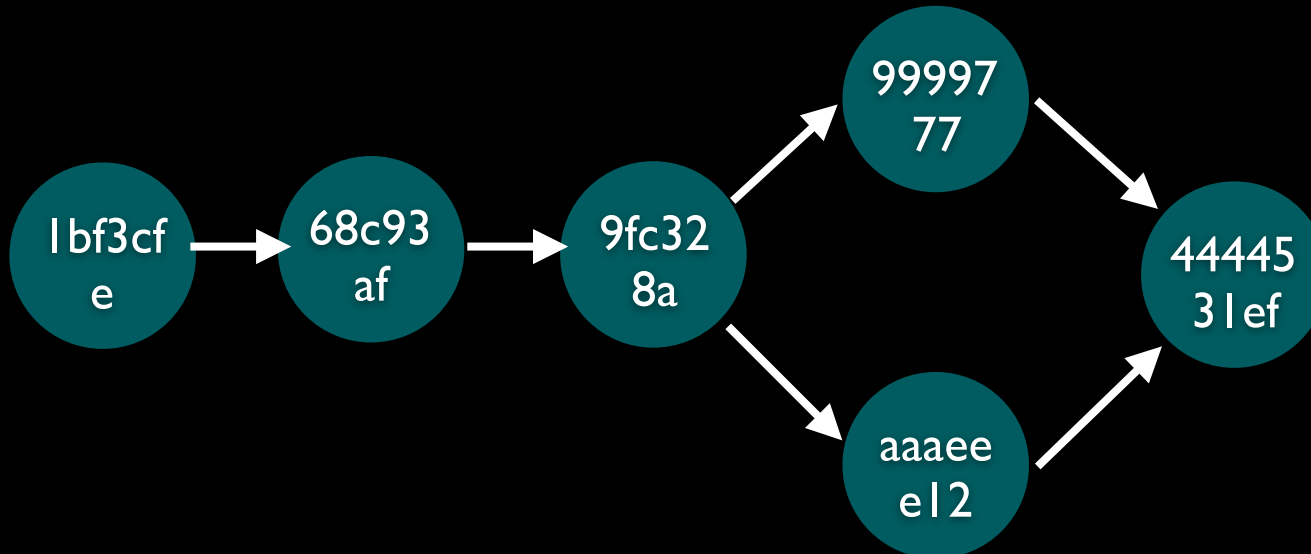
Permanent divergence (never join again)



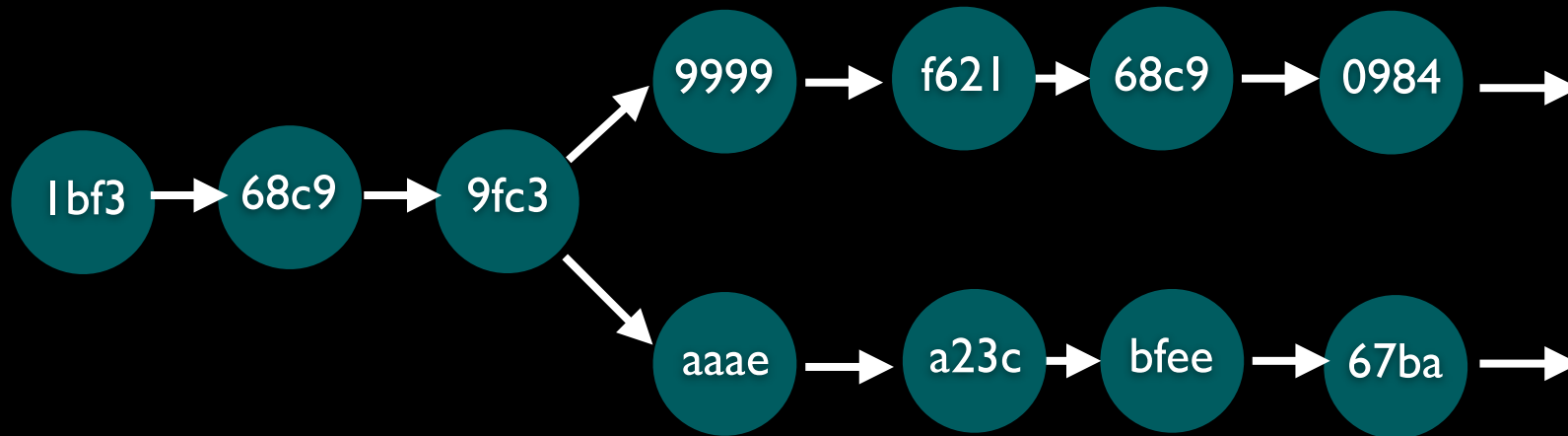
Permanent divergence (never join again)



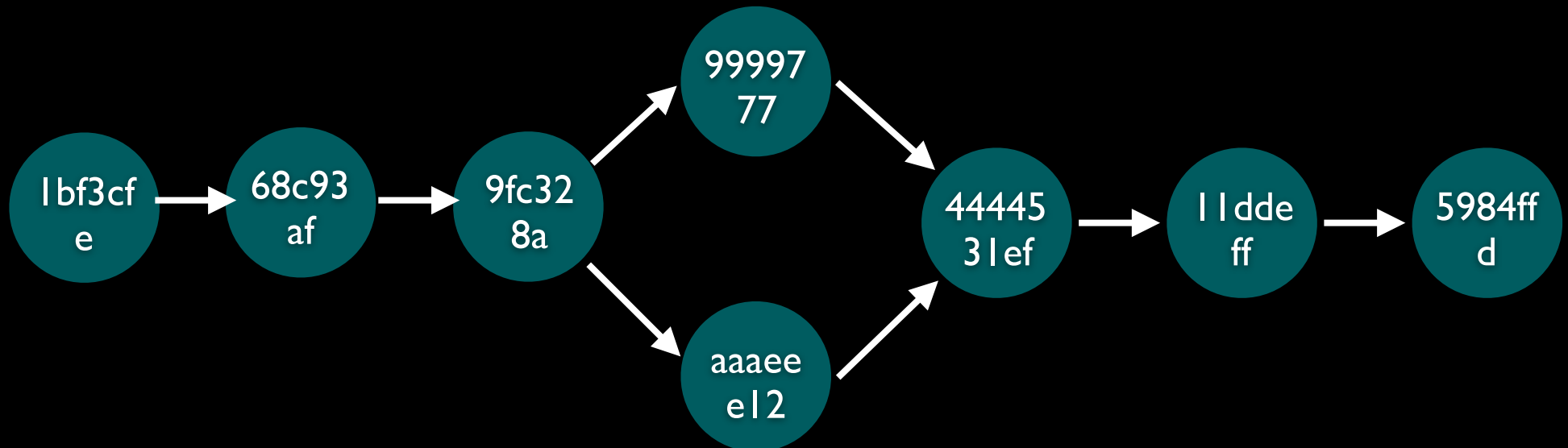
Merge (reconcile different histories)



Permanent divergence (never join again)



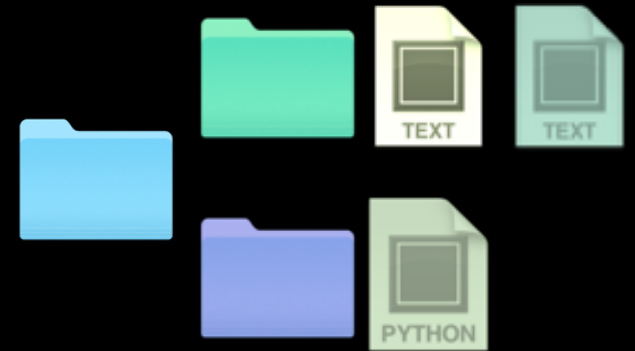
Merge (reconcile different histories)



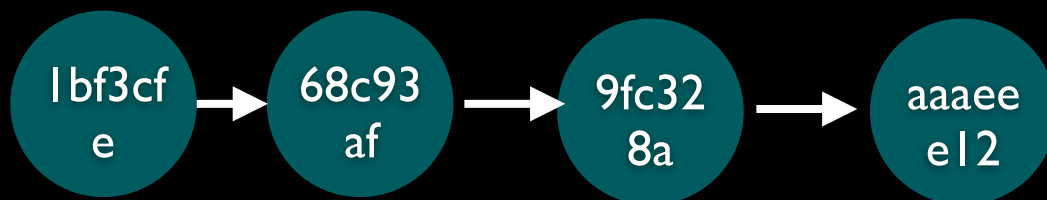
Bob's git repository



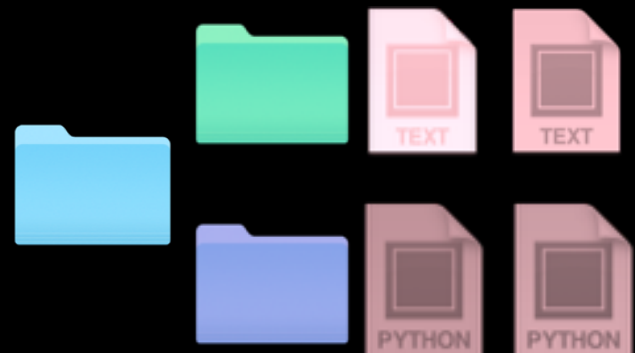
Bob's file system



Alice's git repository



Alice's file system



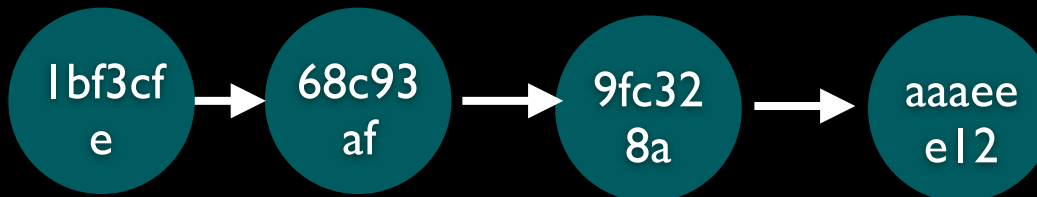
Bob's git repository



Bob's file system



Alice's git repository



Alice's file system



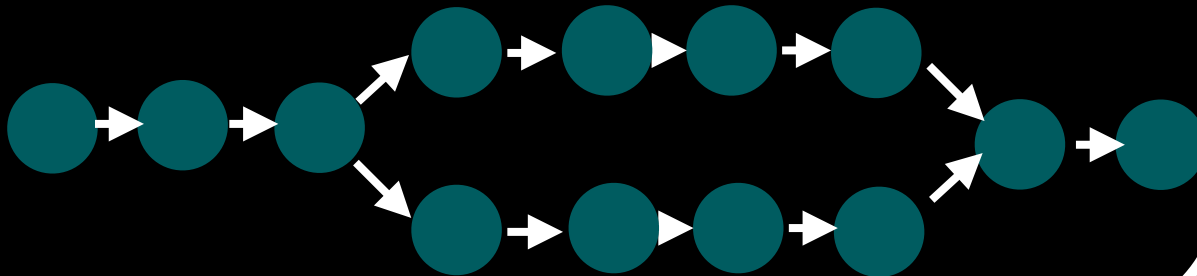
Bob's git repository



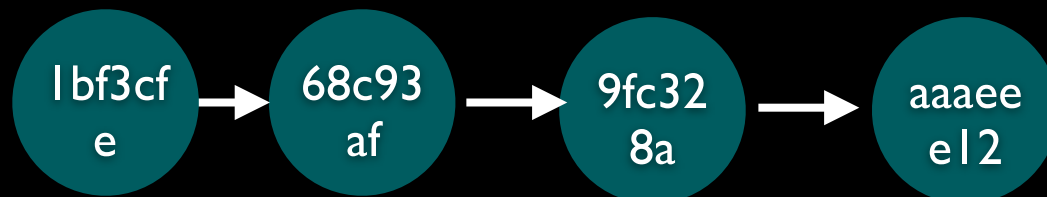
Bob's file system



Github or bitbucket



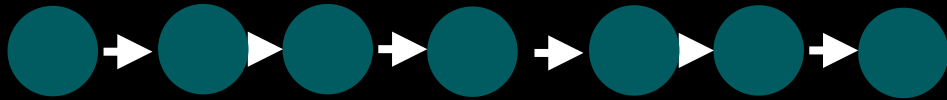
Alice's git repository



Alice's file system



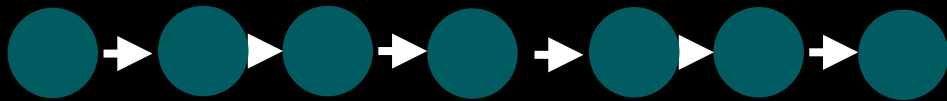
Bob's git repository



Bob's file system



Always online service

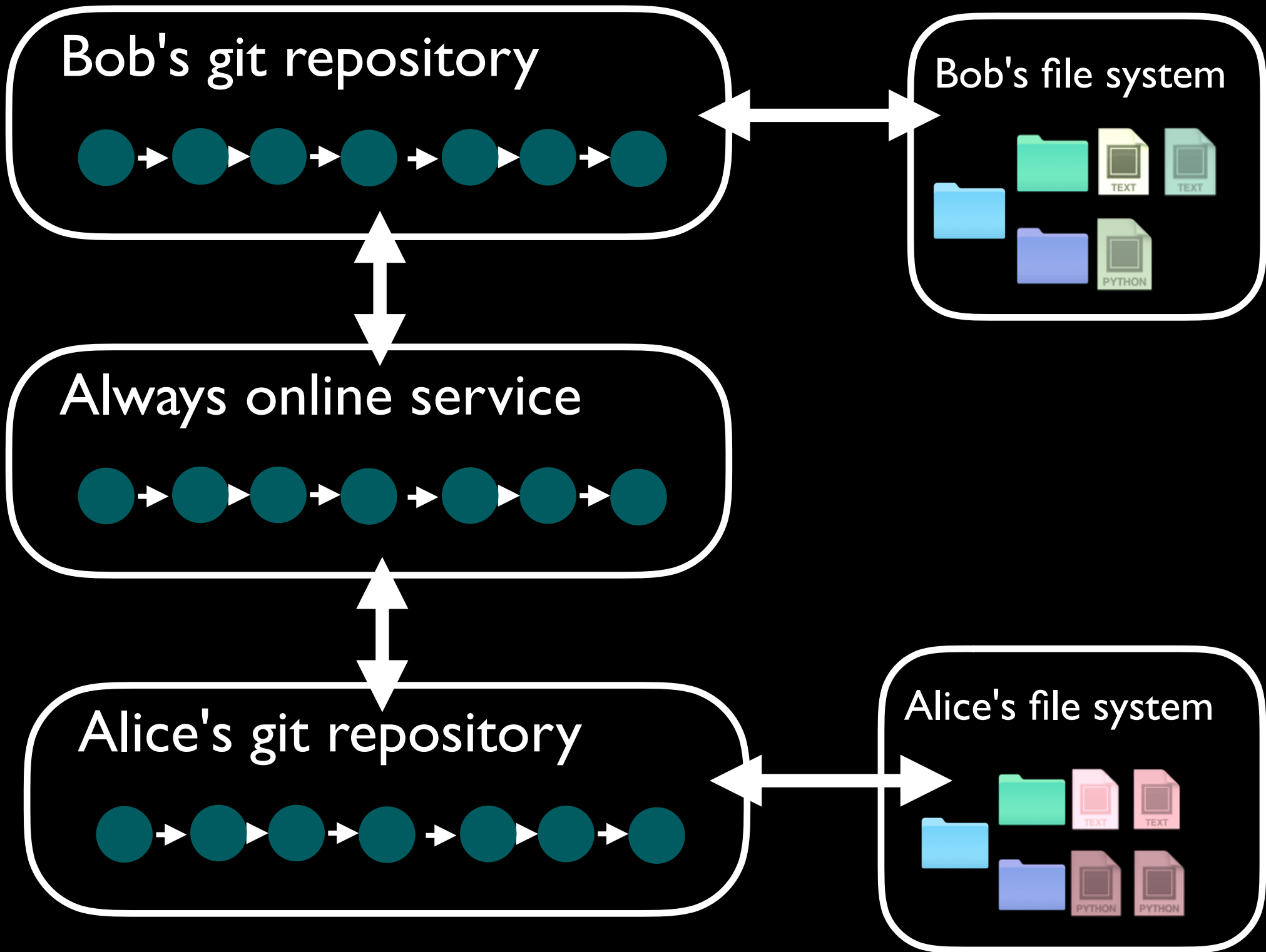


Alice's git repository

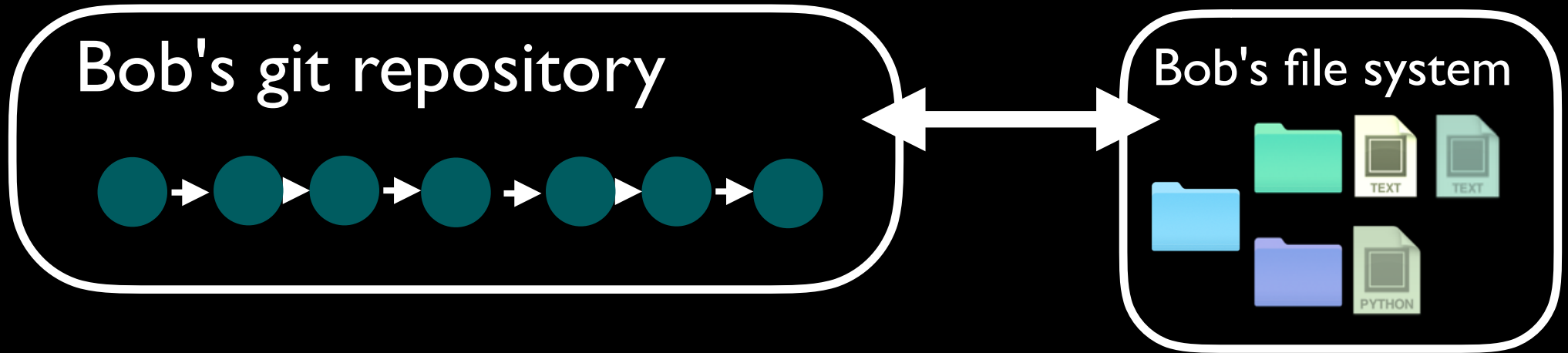


Alice's file system

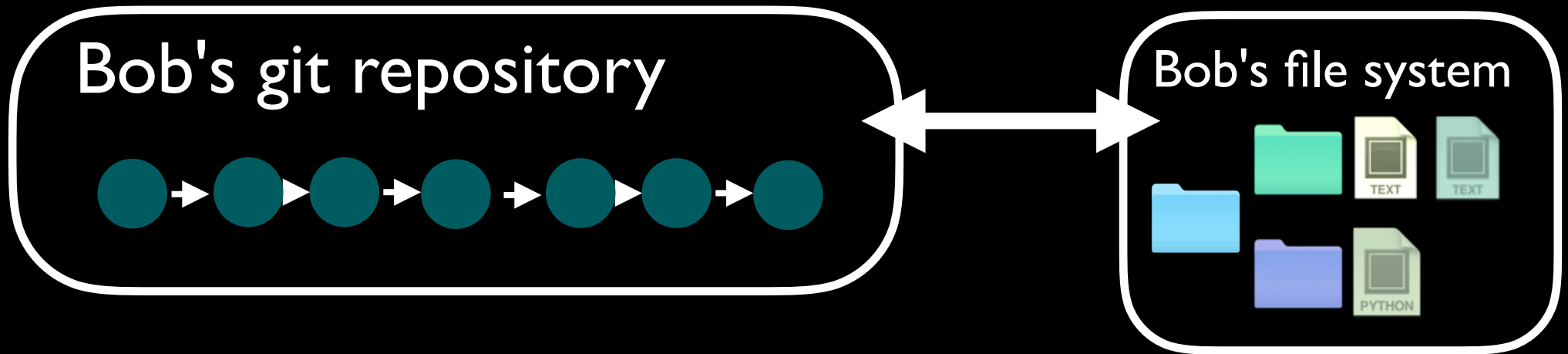




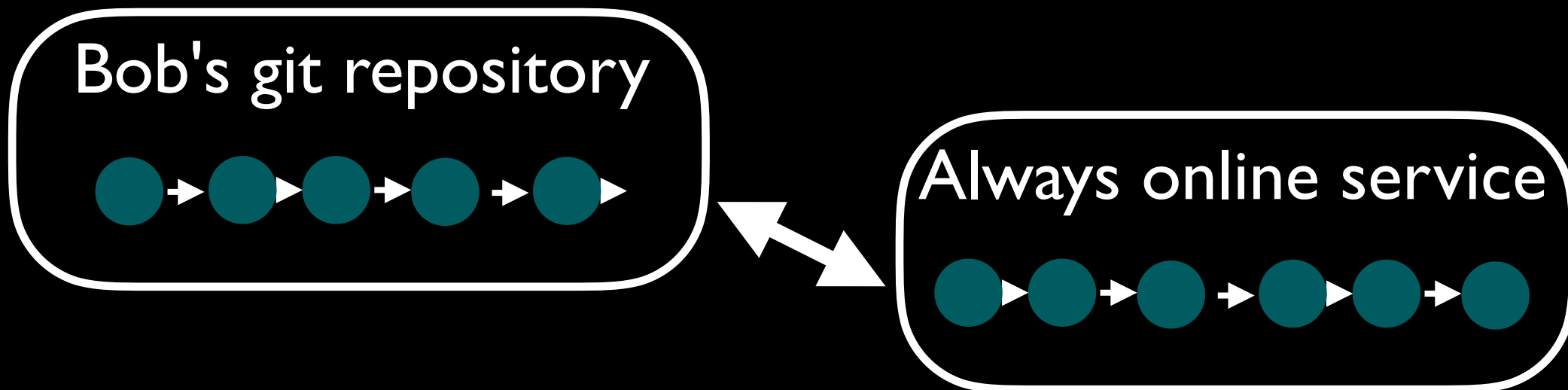
Interactions between file system and repository



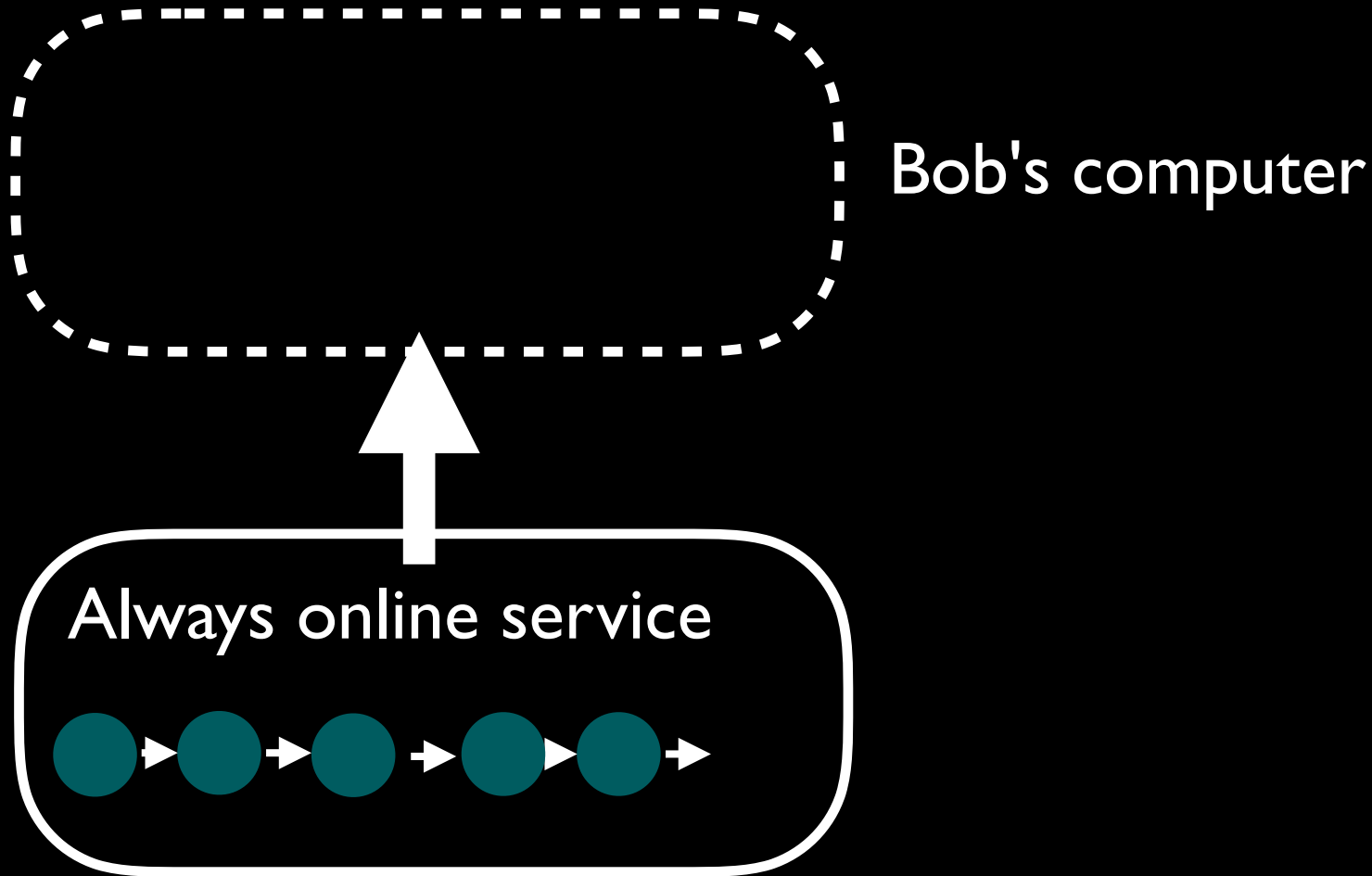
Interactions between file system and repository



Interactions between repository and repository

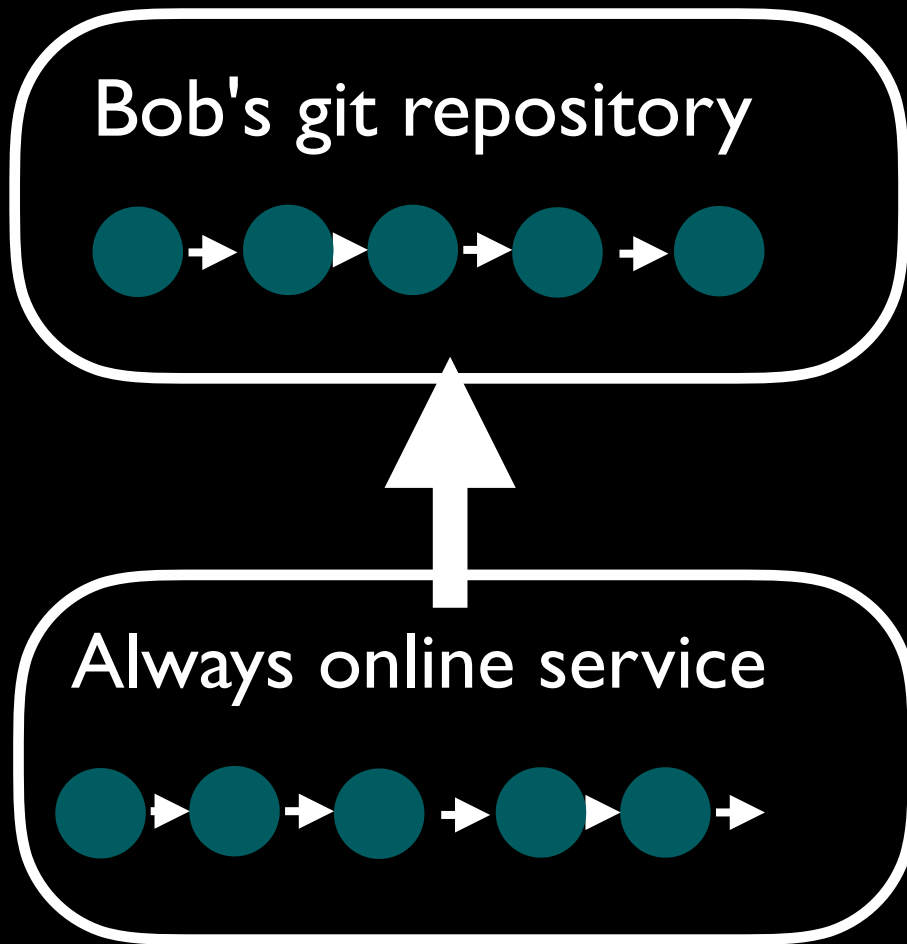


Repo to repo interaction



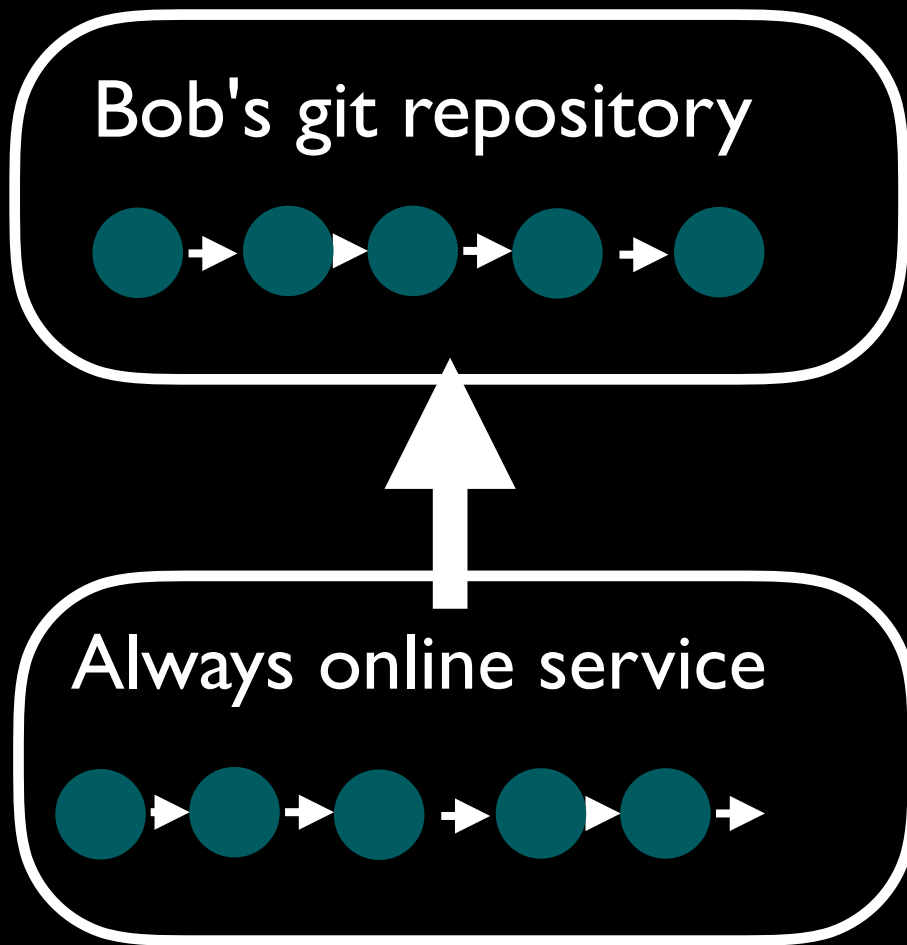
```
$ git clone
```

Repo to repo interaction

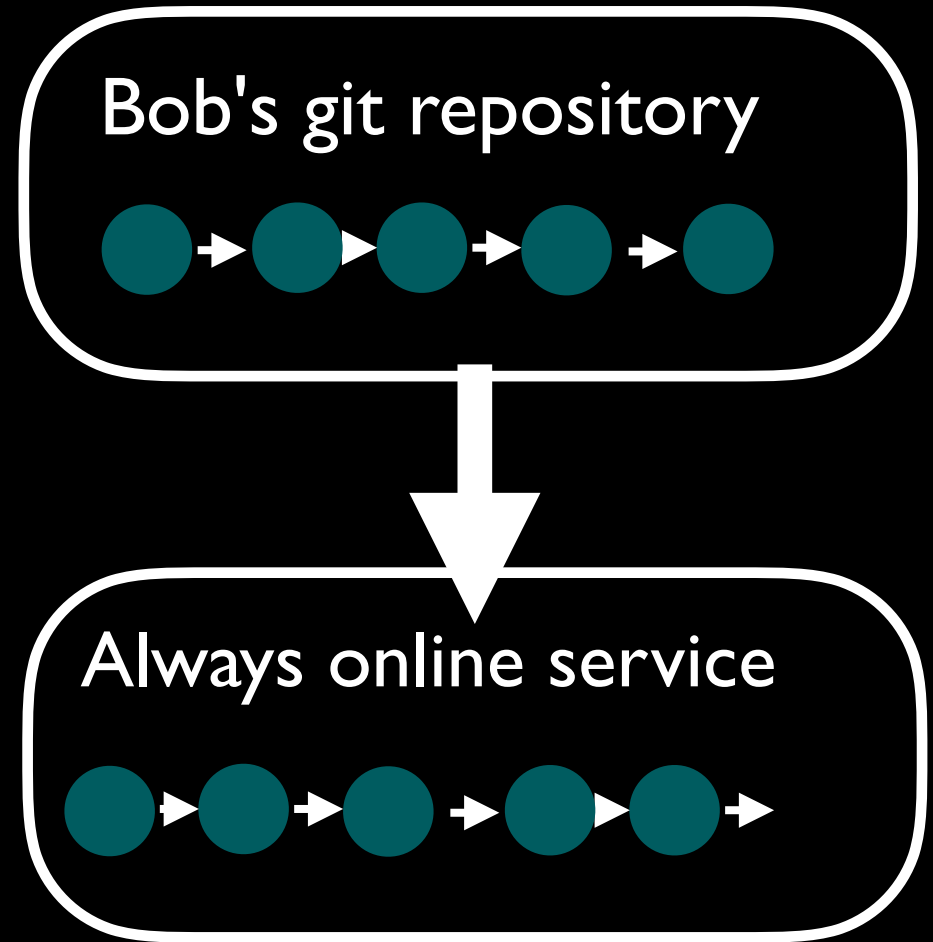


```
$ git pull
```

Repo to repo interaction



`$ git pull`



`$ git push`

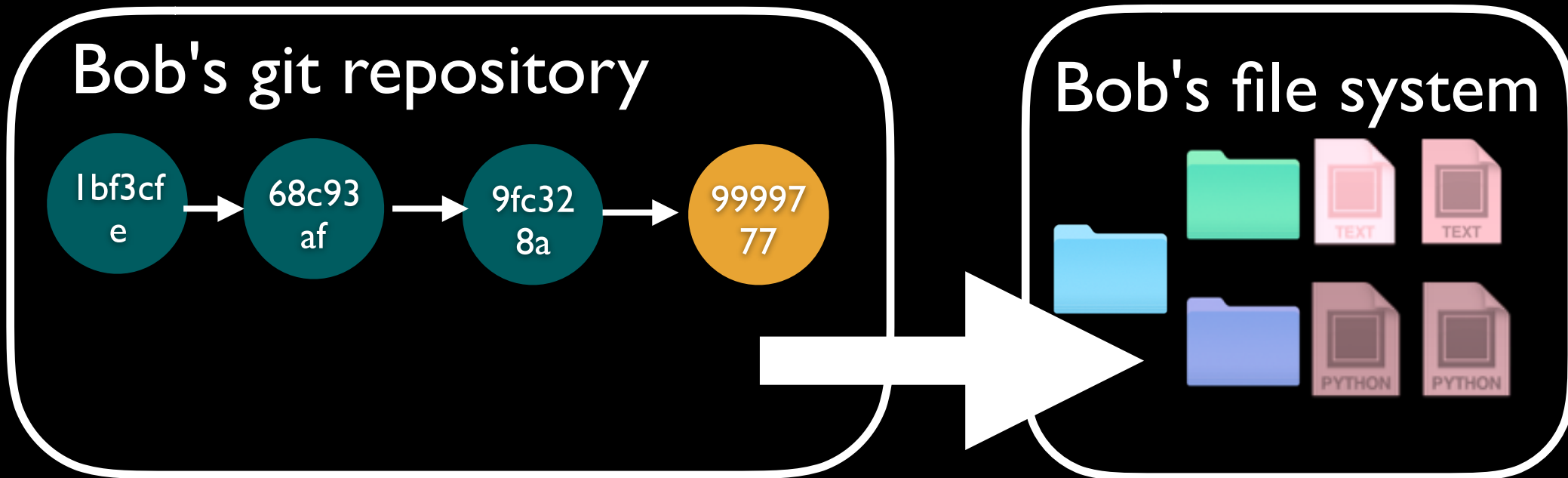
Repo to repo interaction

```
$ git clone
```

```
$ git pull
```

```
$ git push
```

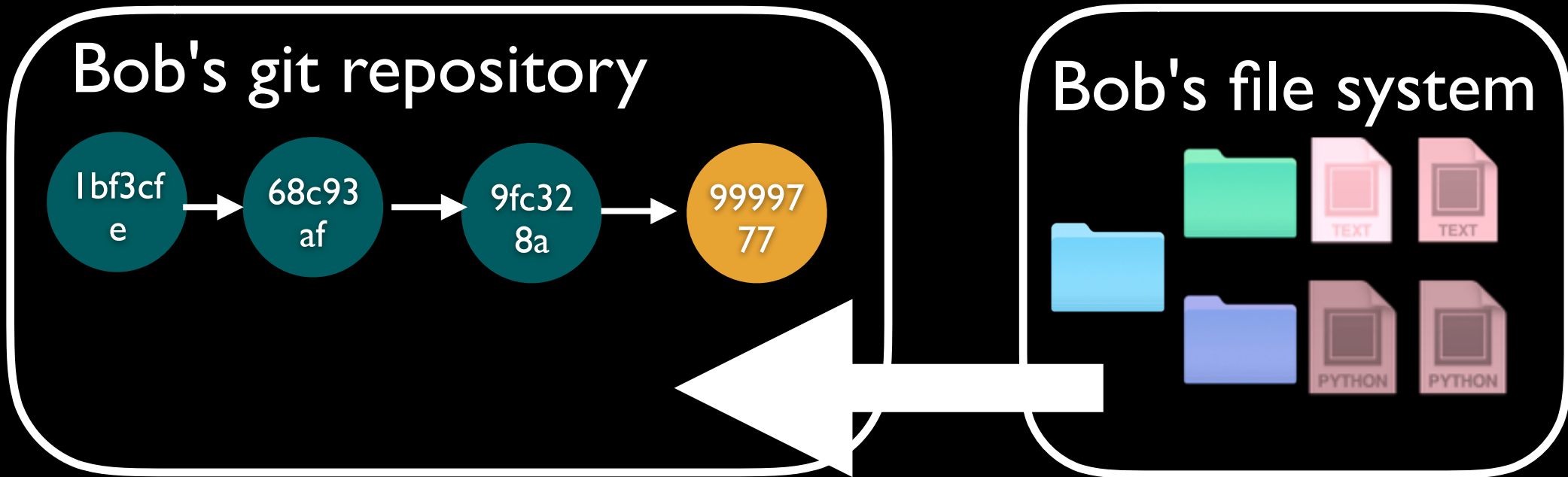
Repo to file interaction



```
$ git checkout af2367512
```

```
$ git checkout master
```


Files to repo interaction



`$ git commit`

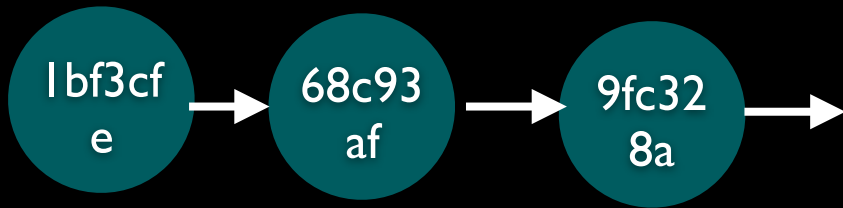
`$ git status`

`$ git diff`

`$ git add`

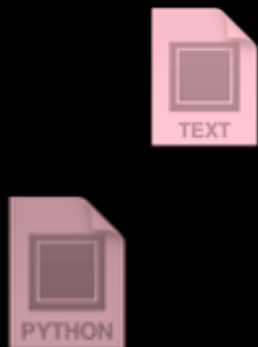
`$ git remove`

Bob's git repository



Git commit

The staging area

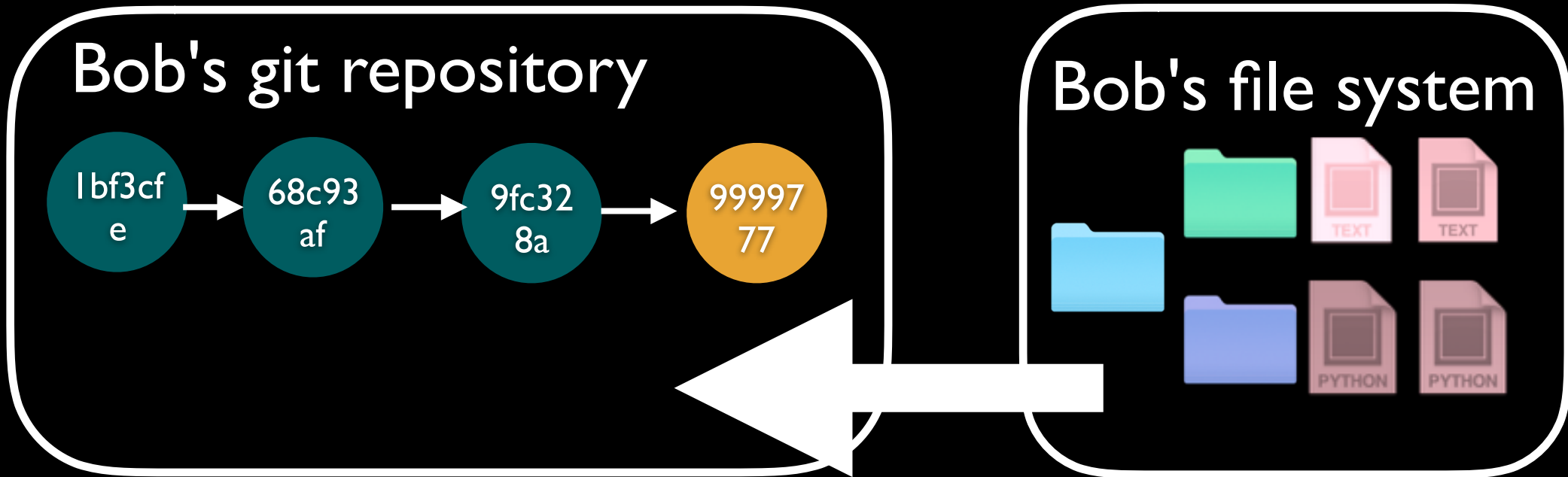


Git add

Bob's file system



Files to repo interaction



`$ git commit`

`$ git status`

`$ git diff`

`$ git add`

`$ git remove`

Git status

Shows the status of your file system when compared to the latest node in the history

```
# On branch master
# Changes to be committed:
# (use "git reset HEAD <file>..." to unstage)
#
# modified:   hello.py
# new file:   README
#
# Changes not staged for commit:
# (use "git add <file>..." to update what will be committed)
# (use "git checkout -- <file>..." to discard changes in working
directory)
#
# modified:   main.py
#
# Untracked files:
# (use "git add <file>..." to include in what will be committed)
#
# hello.pyc
```

Git diff

Shows actual line-by-line difference of files not staged for commit

```
diff --git a/hello.py b/hello.py
index d62ac43..8d15d50 100644
--- a/hello.py
+++ b/hello.py
@@ -1,7 +1,7 @@
class HelloWorld(object):

    def hello(self):
-        print "hello world"
+        print "hola mundo"

    def bye(self):
        x = 1
```

Git add

Puts a file from the "Changes not staged for commit" area to the "Changes to be committed" area.

```
$ git add hello.py
```

Git remove

In the next commit, the chosen file will not be tracked anymore and also removed from your file system.

```
$ git remove hello.py
```

```
$ git remove --cached hello.py
```

Git commit

Take the "changes to be committed" and add a node in the git history, assigning it a new hash such as:
1d8e1c08f7df2aa329c315f780595228279ad551

```
$ git commit -m "Fix bug that would corrupt  
user data when the address field was  
updated"
```


Optional nice extra tool

\$ tig

```
2016-10-11 15:31 Lucas Sinclair | o Correct a small typo
2016-10-11 14:10 Lucas Sinclair | o Added latest stuff
2016-10-11 13:33 Lucas Sinclair | o Added zipped version
2016-10-11 11:46 Alexander Eiler o | added files required for exercises in class
2016-10-11 11:45 Alexander Eiler M | Merge branch 'master' of github.com:xapple/python_ebc
2016-10-11 10:31 Alexander Eiler o | fixed typos in lecture 2 day 2
2016-10-11 10:02 ldutoit | o strcture
2016-10-11 10:01 ldutoit | o structureday03
2016-10-11 09:58 ldutoit | o strucctureday03
2016-10-10 21:48 Alexander Eiler M | Merge branch 'master' of github.com:xapple/python_ebc
2016-10-10 21:47 Alexander Eiler o | added second day exercises and day three exercises
2016-10-10 14:28 Lucas Sinclair | o Added the very first slides of day 1
[main] a53163c0556bba06f27bebf06c94fe9bedead721 - commit 24 of 30 93%
commit a53163c0556bba06f27bebf06c94fe9bedead721
Author: ldutoit <dutoit.ludovic@gmail.com>
AuthorDate: Tue Oct 11 10:01:12 2016 +0200
Commit: ldutoit <dutoit.ludovic@gmail.com>
CommitDate: Tue Oct 11 10:01:12 2016 +0200

    structureday03
----
day_03/home_exercises/exercise_instructions.pdf | Bin 0 -> 33090 bytes
day_03/home_exercises/seq.fas | 121 ++++++
2 files changed, 121 insertions(+)
```

Summary

\$ git clone

\$ git status

\$ git pull

\$ git diff

\$ git push

\$ git add

\$ git remove

\$ git checkout

\$ git commit

\$ tig

Any Questions?



Exercise

1. Go to github.com
2. Create a free account.
3. Go to github.com/xapple/python_homework
4. Fork that repository to your own github account.
5. Clone that repository to your own computer.

Exercise

1. Change the "README" with some content.
2. Add the file to the "to be committed" area.
3. Commit the file (with a good message).
4. Push the changes.
5. Confirm the change online in your browser.

Exercise by pairs

1. Give your neighbor write access to your repository.
2. Make him change your README file and push the changes.
3. Pull his changes back to your copy of the repository.

From now, on all
exercises and homework
are to be uploaded to
your "python_homework"
repository

Online materials

For further training, there are about a hundred good git tutorials online.

Warning

Don't combine revision control systems

Things to explore

- git branch
- the ".gitignore" file
- git stash
- Github personal pages

15 minutes break

