



GIT

Alex Barton @ CETiK 2014

Was tut es?

- „the stupid content tracker“
 - git(1) manual page
- „distributed version control system“
 - <http://git-scm.com>

Warum GIT?

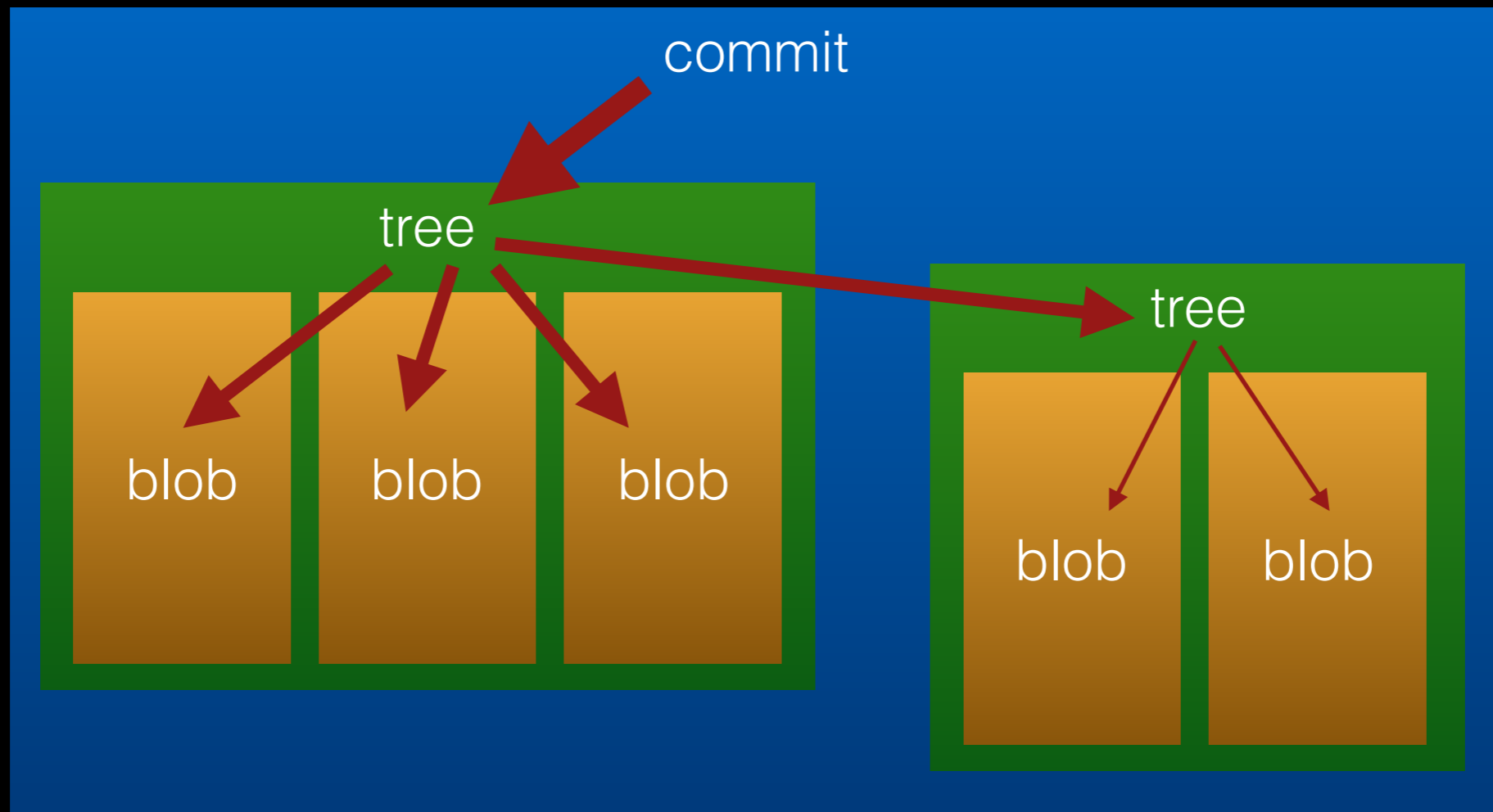
- Commit-basiert, Historie als Serie von Patches
- Unterstützt Umbenennen von Dateien
- Verteilte Architektur
 - jede Repository ist in sich „komplett“
- Effizient & schnell
 - Objekte werden komprimiert
 - Dateisystem-basiert
 - „pack files“
- Kryptographische Sicherheit der Projektgeschichte

Wie tut es das?

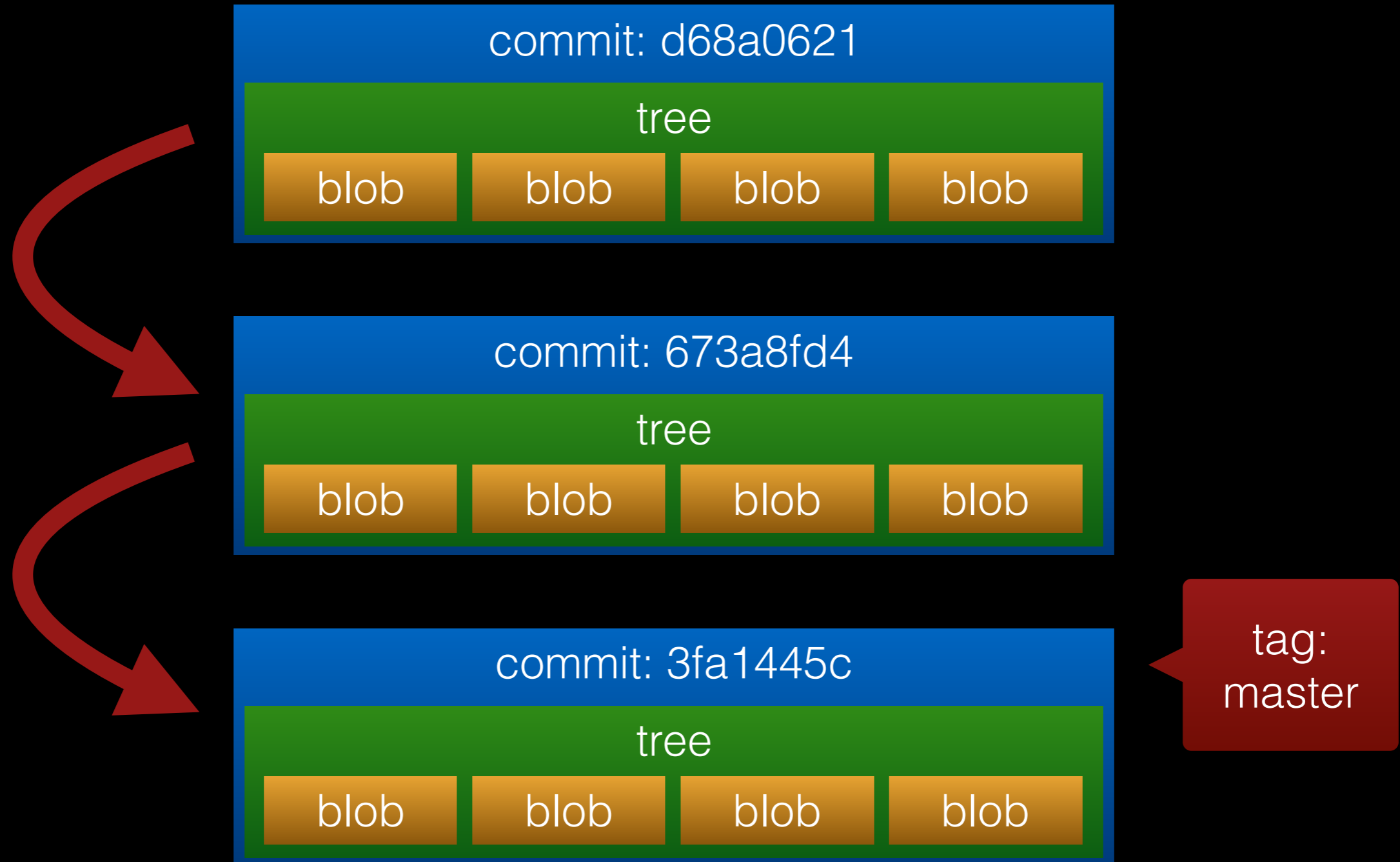
- repositories
- commits
- branches

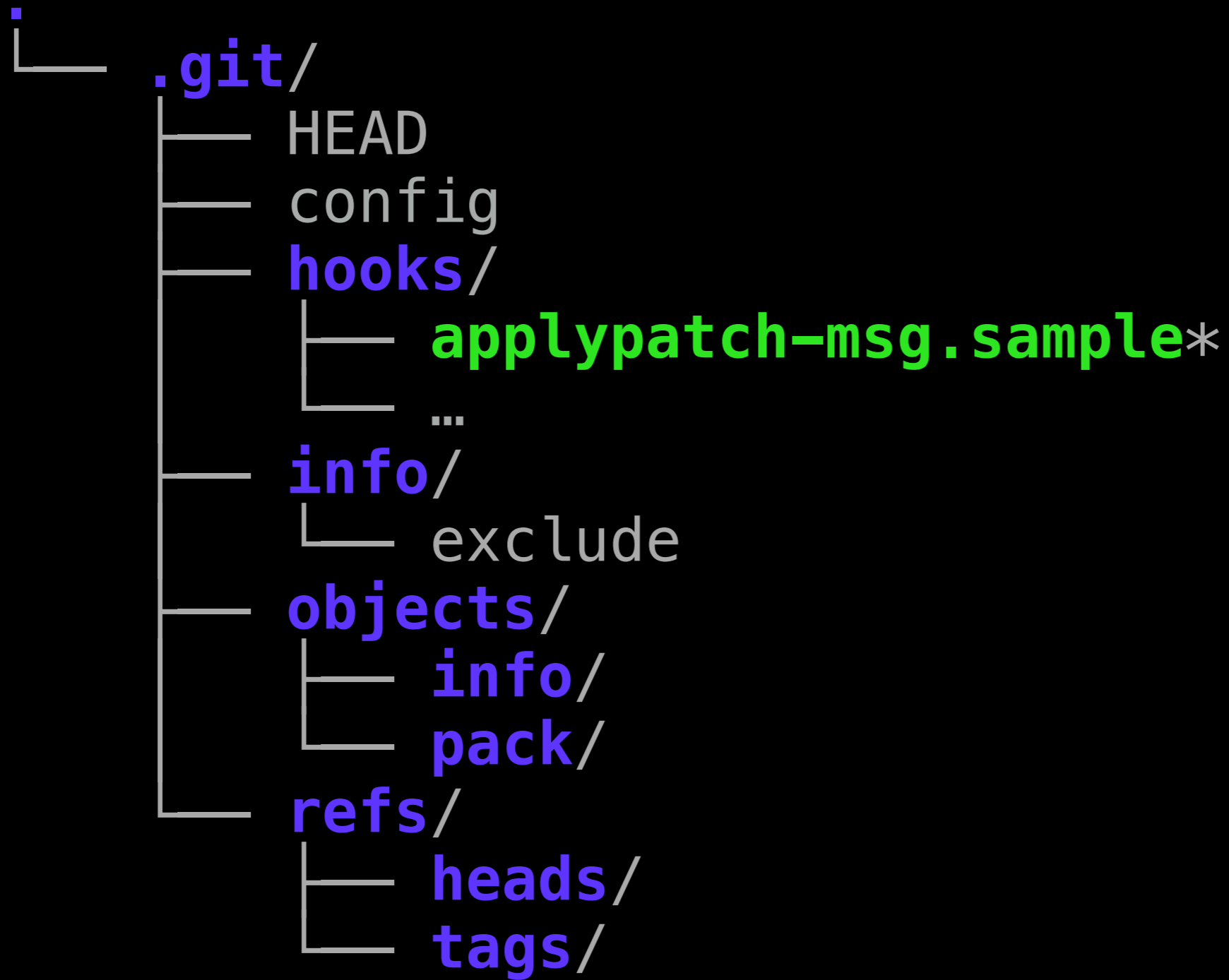
- tree, blob, commit & tag objects

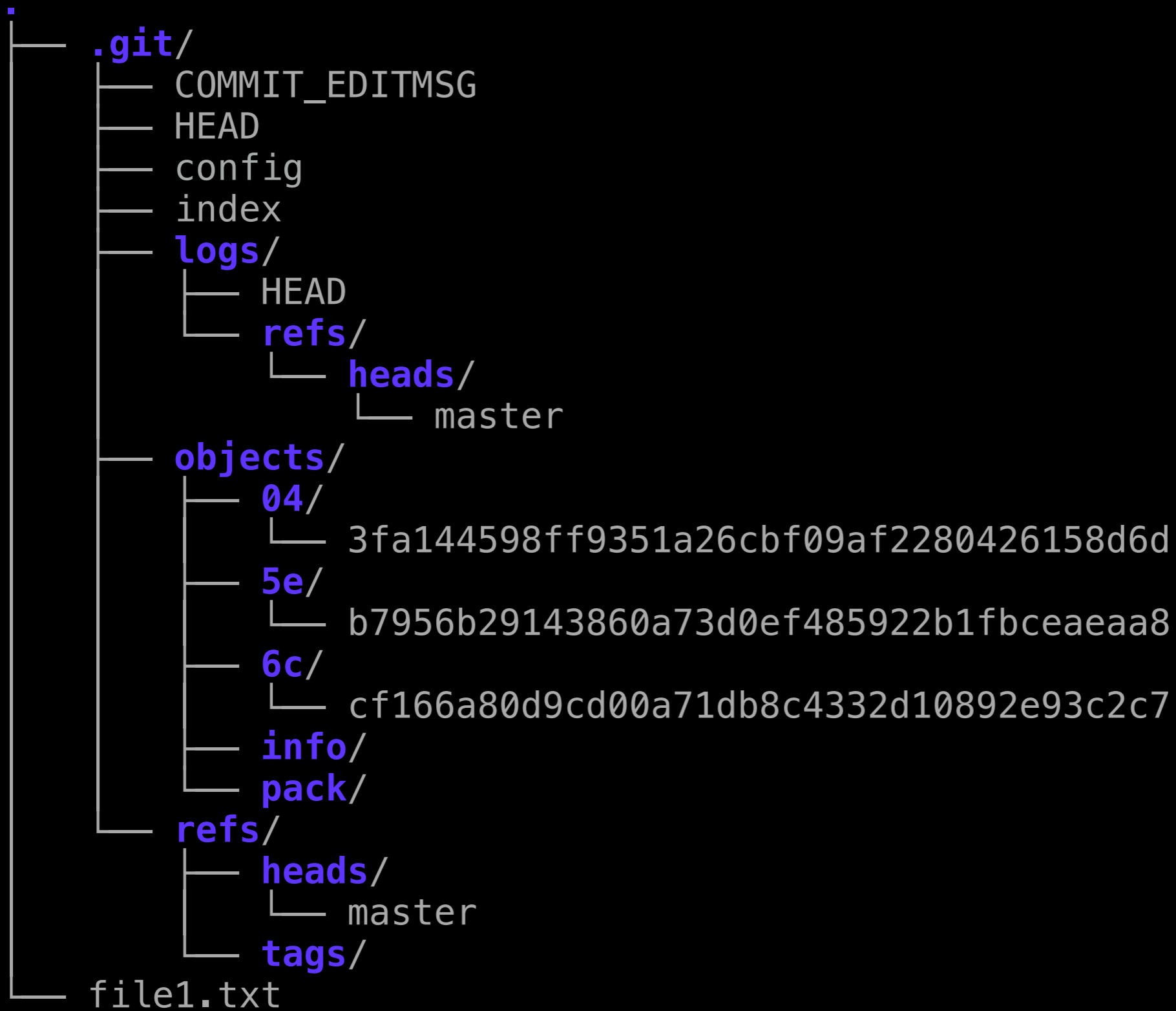
tree, blob, commit & tag



tree, blob, commit & tag







```
.
├── .git/
│   ├── COMMIT_EDITMSG
│   ├── HEAD
│   ├── ORIG_HEAD
│   ├── config
│   ├── index
│   ├── logs/
│   │   ├── HEAD
│   │   └── refs/
│   │       ├── heads/
│   │       │   └── master
│   ├── objects/
│   │   ├── 0e/
│   │   │   └── 6b9ad691ff9ce5c7f59c32687338fe60a23bfe
│   │   ├── 28/
│   │   │   └── caa6d07a324eba7309ad19be517823f1876b8e
│   │   ├── ad/
│   │   │   └── 604ec35e94470b2d6c2cc73dffe3e5aec48a8
│   │   ├── info/
│   │   │   └── packs
│   │   └── pack/
│   │       ├── pack-9c4a6910f7b164aff418148d0290d2b6b76672aa.idx
│   │       └── pack-9c4a6910f7b164aff418148d0290d2b6b76672aa.pack
│   ├── packed-refs
│   └── refs/
│       ├── heads/
│       │   └── master
│       └── tags/
├── file1.txt
├── file2.txt
└── file3.txt
```

Was muss ich tun?

- git config
- git init
- git add
- git commit
- git checkout
- git reset
- git branch

... und dann?

- git fetch
- git pull
- git push
- git merge
- git rebase

... noch mehr?

- git tag
- git cherry-pick
- git blame

GUI für GIT

- GitK
- GIT GUI
- SourceTree
- Tower

Tools, die GIT nutzen

- etckeeper
- bup