git

A practical overview

THIS IS GIT. IT TRACKS COLLABORATIVE WORK ON PROJECTS THROUGH A BEAUTIFUL DISTRIBUTED GRAPH THEORY TREE MODEL.
COOL. HOW DO WE USE IT?
NO IDEA. JUST MEMORIZE THESE SHELL COMMANDS AND TYPE THEM TO SYNC UP. IF YOU GET ERRORS, SAVE YOUR WORK ELSEWHERE, DELETE THE PROJECT, AND DOWNLOAD A FRESH COPY.

Topics

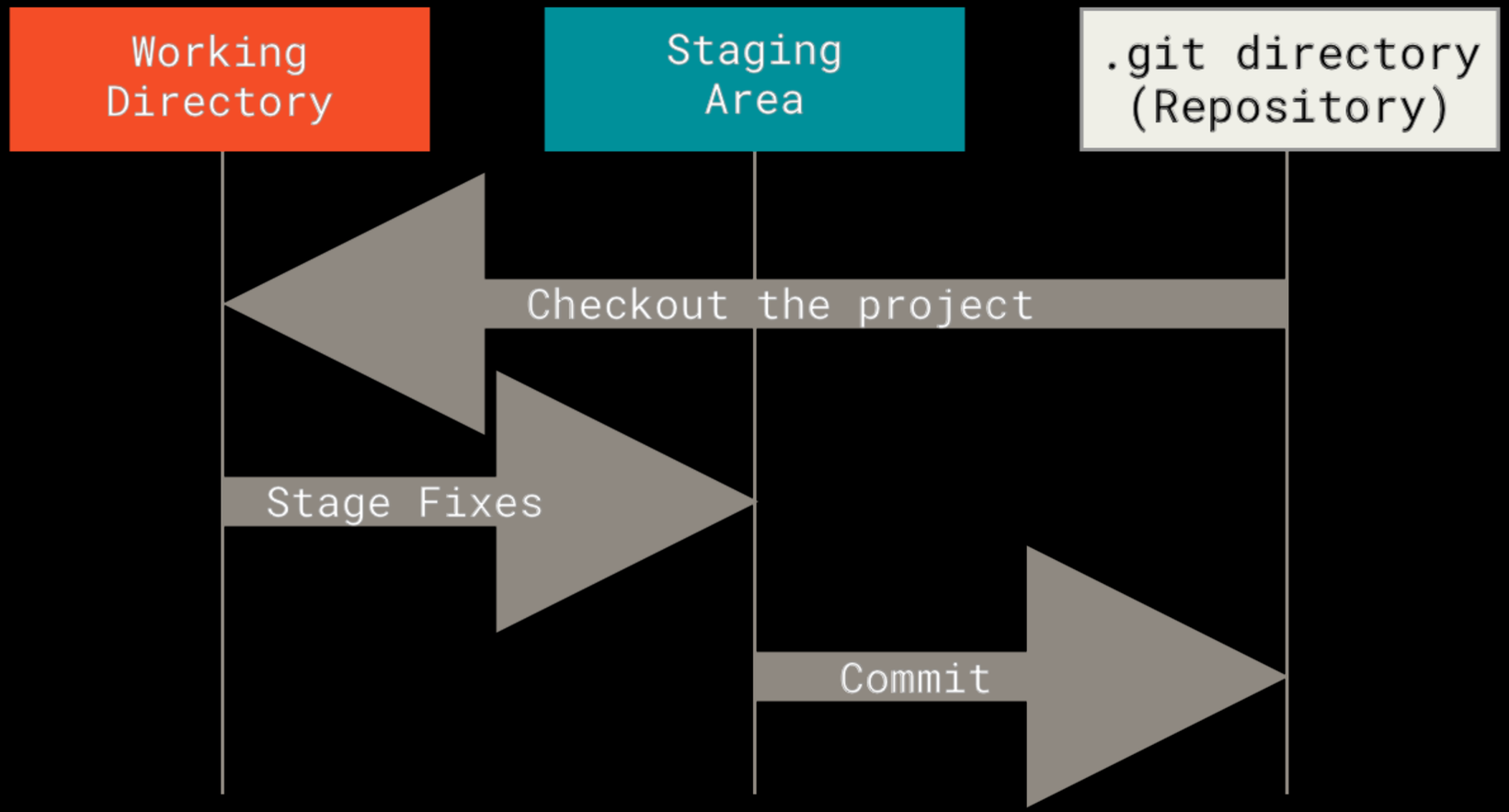
- Introduction
- Setup
  - Configuration
  - Aliases
- Common commands
- Best practice
- Advanced topics

General

- interactive, no listen only
- focus on the terminal/CLI
- git help/man pages (man git-<command>)
- Official documentation
- Test repo: https://github.com/nodeg/test-git

Git

3 main states that your files can reside in: modified, staged, and committed



Configuration

```
[init]
    defaultBranch = main
[pull]
    rebase = true
[user]
    name = Dominik Gedon
    useConfigOnly = true
[core]
    editor = nvim
    excludesfile = ~/.config/git/ignore
    pager = delta
[color "status"]
    added = yellow
    changed = green
    untracked = cyan

[include]
    path = ~/.config/git/alias
```

~/.config/git/

.gitignore

- specifies intentionally untracked files that git should ignore
- via a local *.gitignore* file or *~/config/git/ignore*

Aliases

```
amend        = commit --amend
branches     = branch -a
contributors= shortlog -s -n
graph        = log --graph -10 --branches --remotes --tags\
               --format=format:'%Cgreen%h %Creset %<(75,trunc)%s (%cN, %ar) %Cred%d' --date-order
history      = log --pretty=format:\"%C(yellow)%h%Creset %ad | %Cgreen%s%Creset %Cred%d%Creset %Cblue[%an]\" --date=short
last         = log -1 HEAD --format=format:"%Cred%H"
logs         = log --stat
what         = show -w
nevermind    = !git reset --hard HEAD && git clean -d -f
```

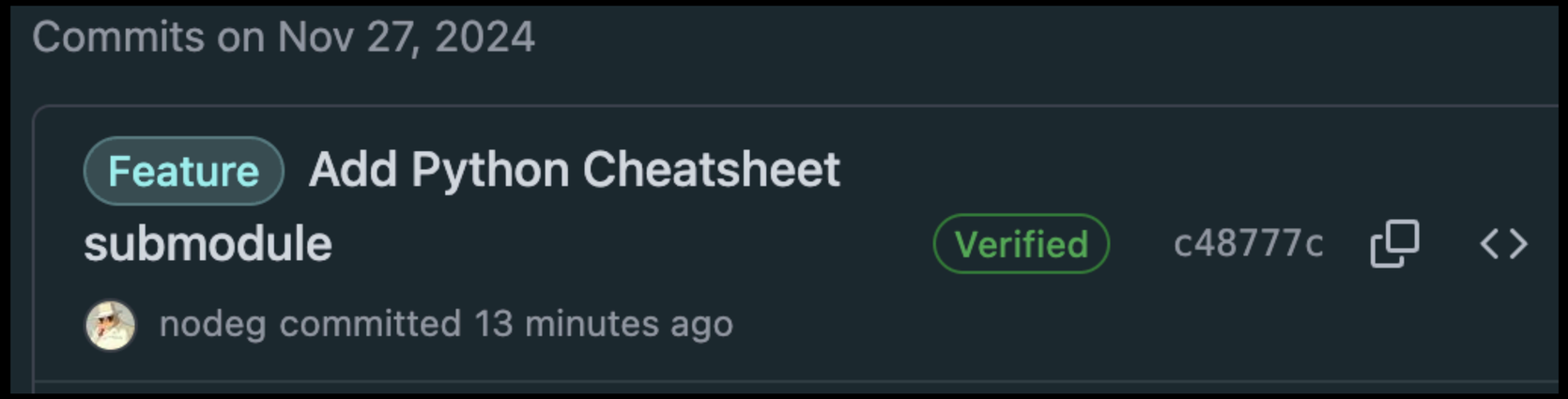
Commit Template

```
[commit]
    template = ~/.config/git/template
```

```
# <type>: If applied, this commit will
# |<———  Using a Maximum Of 50 Characters  ———>|
# Type can be
#    feat     (new feature)
#    fix      (bug fix, semver: PATCH)
#    refactor (refactoring production code, semver: MINOR)
#    style    (formatting, missing semi colons, etc; no code change)
#    docs     (changes to documentation)
#    test     (adding or refactoring tests; no production code change)
#    chore    (updating grunt tasks etc; no production code change)
#    BREAKING CHANGE (introduces a breaking API change, semver: MAJOR)

# Explain why this change is being made
# |<———   Try To Limit Each Line to a Maximum Of 72 Characters   ———>|
```

Signed commits

1. Specify the GPG key in the git configuration
2. Tell git to (always) GPG-sign commits




```
❯ git log --show-signature
commit c48777ccebe307b9cffb30d87b1343b0c5044b9d (HEAD → main, origin/main, origin/HEAD)
gpg: Signature made Wed Nov 27 15:09:46 2024 CET
gpg:                using RSA key 4F01C768F1A8DEEF49B895E3263BF422CC90C832
gpg: Good signature from "Dominik Gedon <dominik@gedon.org>" [ultimate]
Primary key fingerprint: 4F01 C768 F1A8 DEEF 49B8  95E3 263B F422 CC90 C832
Author: Dominik Gedon <dominik@gedon.org>
Date:   Wed Nov 27 15:09:03 2024 +0100

    feat: Add Python Cheatsheet submodule
```

Configuration - Tasks

1. Create a configuration file and set up commit signing by default
2. Create some aliases that fit your needs
3. Create and include a commit template file
4. Create a .gitignore file (locally or globally)

Common commands

Commands

config	fetch	log	cherry-pick
add, commit, push	pull	remote	tag
diff	merge	revert	
show	rebase	reset	
blame	stash	apply	

git config

Get and set repository or global options

```
git config list
git config commit.gpgsign true
git config --global commit.gpgsign true
```

git init

Create an empty Git repository

```
~/git/dom
❯ la
drwxr-xr-x dgedon staff  64 B Wed Nov 27 22:40:15 2024 🗁 .
drwxr-xr-x dgedon staff 320 B Wed Nov 27 22:40:15 2024 🗁 ..

~/git/dom
❯ git init
Initialized empty Git repository in /Users/dgedon/git/dom/.git/

dom on  main
❯ la
drwxr-xr-x dgedon staff  96 B Wed Nov 27 22:40:23 2024 🗁 .
drwxr-xr-x dgedon staff 320 B Wed Nov 27 22:40:15 2024 🗁 ..
drwxr-xr-x dgedon staff 288 B Wed Nov 27 22:40:23 2024  .git
```

```
dom on  main
❯ tree .git/
.git/
├── HEAD
├── config
├── description
├── hooks
│   ├── applypatch-msg.sample
│   ├── commit-msg.sample
│   ├── fsmonitor-watchman.sample
│   ├── post-update.sample
│   ├── pre-applypatch.sample
│   ├── pre-commit.sample
│   ├── pre-merge-commit.sample
│   ├── pre-push.sample
│   ├── pre-rebase.sample
│   ├── pre-receive.sample
│   ├── prepare-commit-msg.sample
│   ├── push-to-checkout.sample
│   ├── sendemail-validate.sample
│   └── update.sample
├── info
│   └── exclude
├── objects
│   ├── info
│   └── pack
└── refs
    ├── heads
    └── tags

9 directories, 18 files
```

git diff

Shows changes between different things

tetra.gemspec

```
30: Gem::Specification.new do |s|
```

```
30 : 30   s.add_development_dependency "simplecov", "~> 0.22.0"
31 : 31   s.add_development_dependency "rake", "~> 13.2.0"
32 : 32   s.add_development_dependency "rspec", "~> 3.13.0"
33 :      s.add_development_dependency "rubocop", "~> 1.64.0"
   : 33   s.add_development_dependency "rubocop", "~> 1.64.1"
34 : 34
35 : 35   s.add_runtime_dependency "clamp", "~> 1.3.2"
36 : 36   s.add_runtime_dependency "erb", "~> 4.0.3"
37 : 37   s.add_runtime_dependency "json_pure", "≥ 2.6.3", "< 2.8.0"
38 : 38   s.add_runtime_dependency "open4", "~> 1.3.4"
39 :      s.add_runtime_dependency "rexml", "~> 3.2.8"
   : 39   s.add_runtime_dependency "rexml", "~> 3.2.9"
40 : 40   s.add_runtime_dependency "rubyzip", "~> 2.3.2"
41 : 41   s.add_runtime_dependency "text", "~> 1.3.1"
42 : 42 end
```

git show

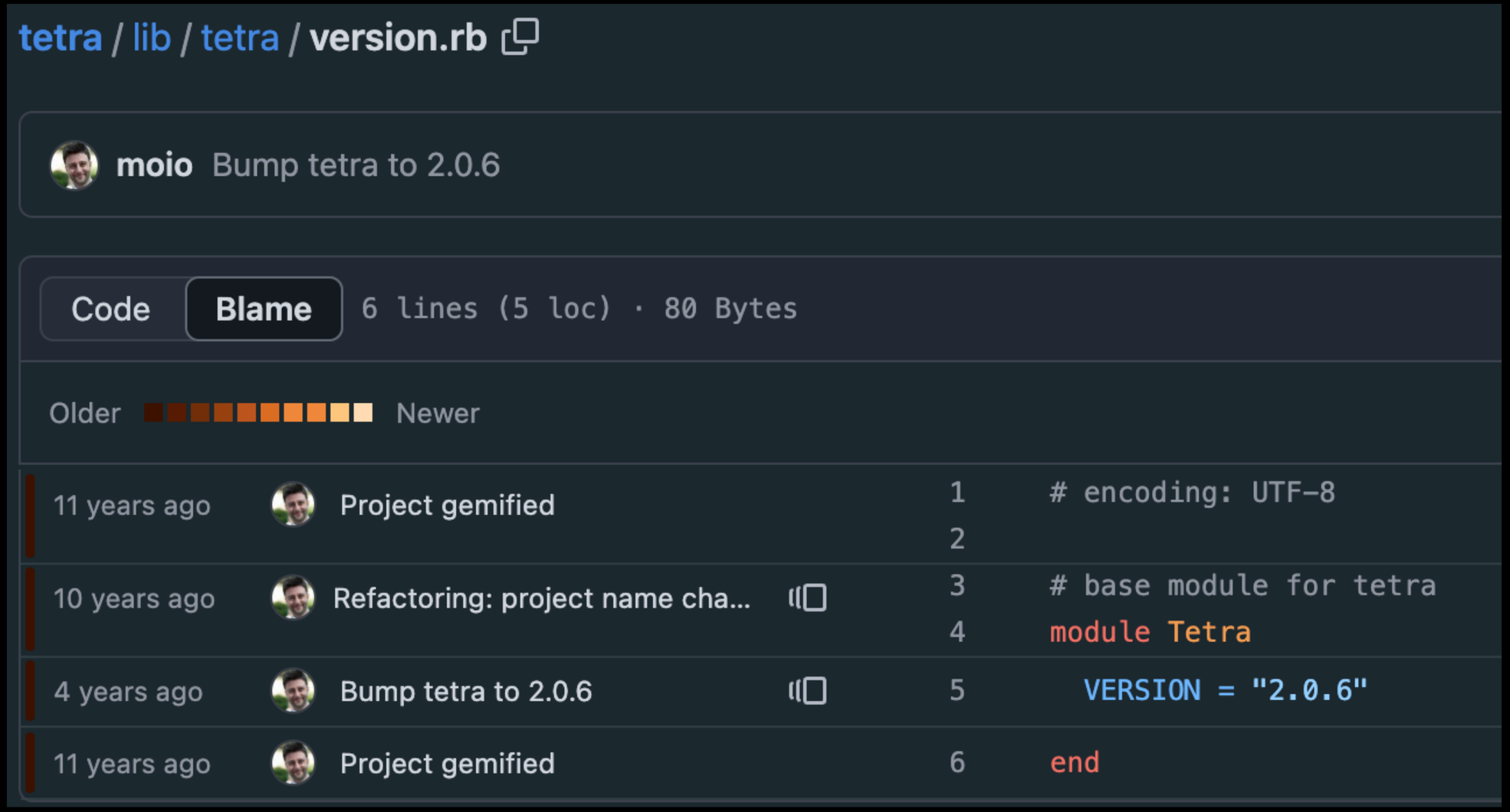
- Shows one or more objects
- For commits it shows the log message and textual diff
- For tags, it shows the tag message and the referenced objects.
- For trees, it shows the names

git blame

Show what revision and author last modified each line of a file

```
❯ git blame lib/tetra/version.rb
11 years ago    Silvio Moioli   cfca822a   1  # encoding: UTF-8
                                           2
10 years ago    Silvio Moioli   920fd8a0   3  # base module for tetra
                                           4  module Tetra
4 years ago     Silvio Moioli   02dcf4f0   5    VERSION = "2.0.6"
11 years ago    Silvio Moioli   cfca822a   6  end
```

git blame


tetra / lib / tetra / version.rb
moio Bump tetra to 2.0.6
Code
Blame
6 lines (5 loc) · 80 Bytes
Older
Newer
11 years ago
Project gemified
10 years ago
Refactoring: project name cha...
4 years ago
Bump tetra to 2.0.6
11 years ago
Project gemified
1 # encoding: UTF-8
2
3 # base module for tetra
4 module Tetra
5 VERSION = "2.0.6"
6 end

git fetch / git pull

git fetch

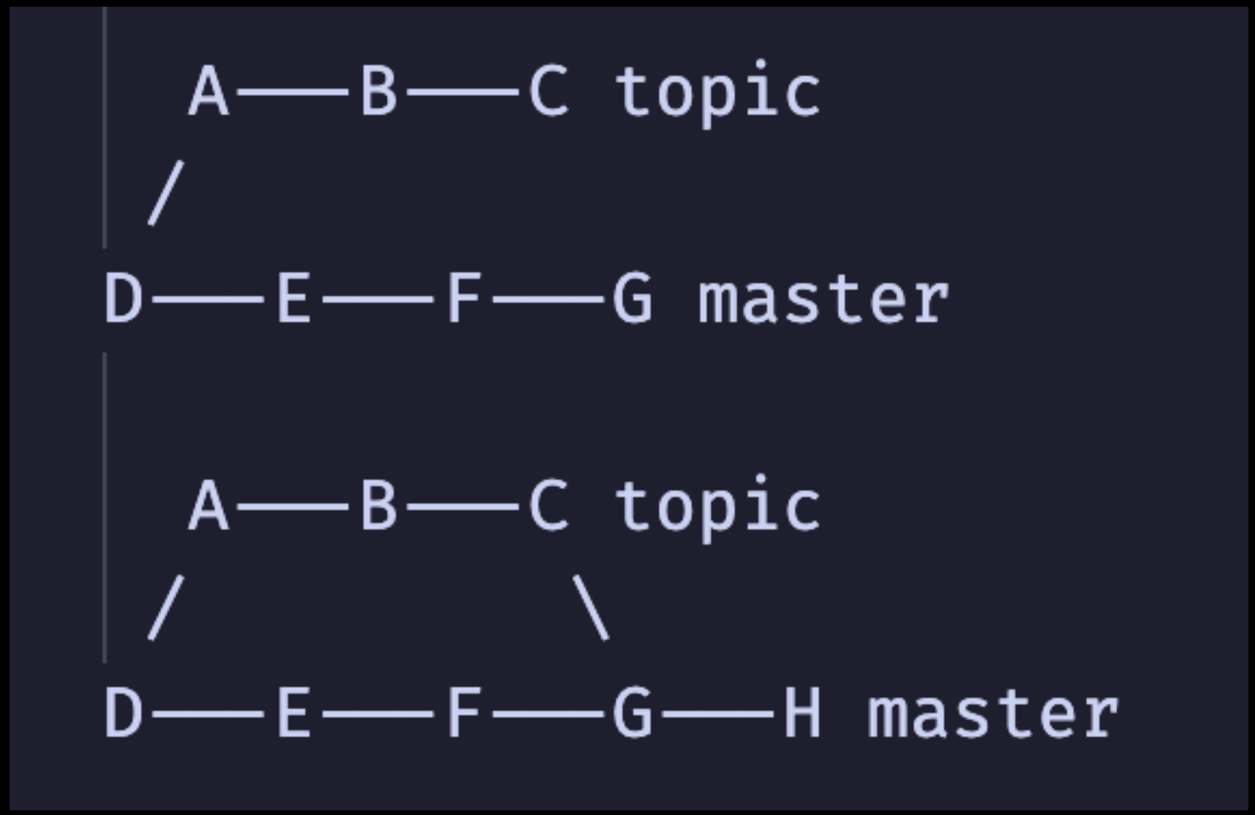
- Download objects and refs from another repository

git pull

- Incorporates changes from a remote repository into the current branch
- basically
  - git fetch
  - git merge oder git rebase

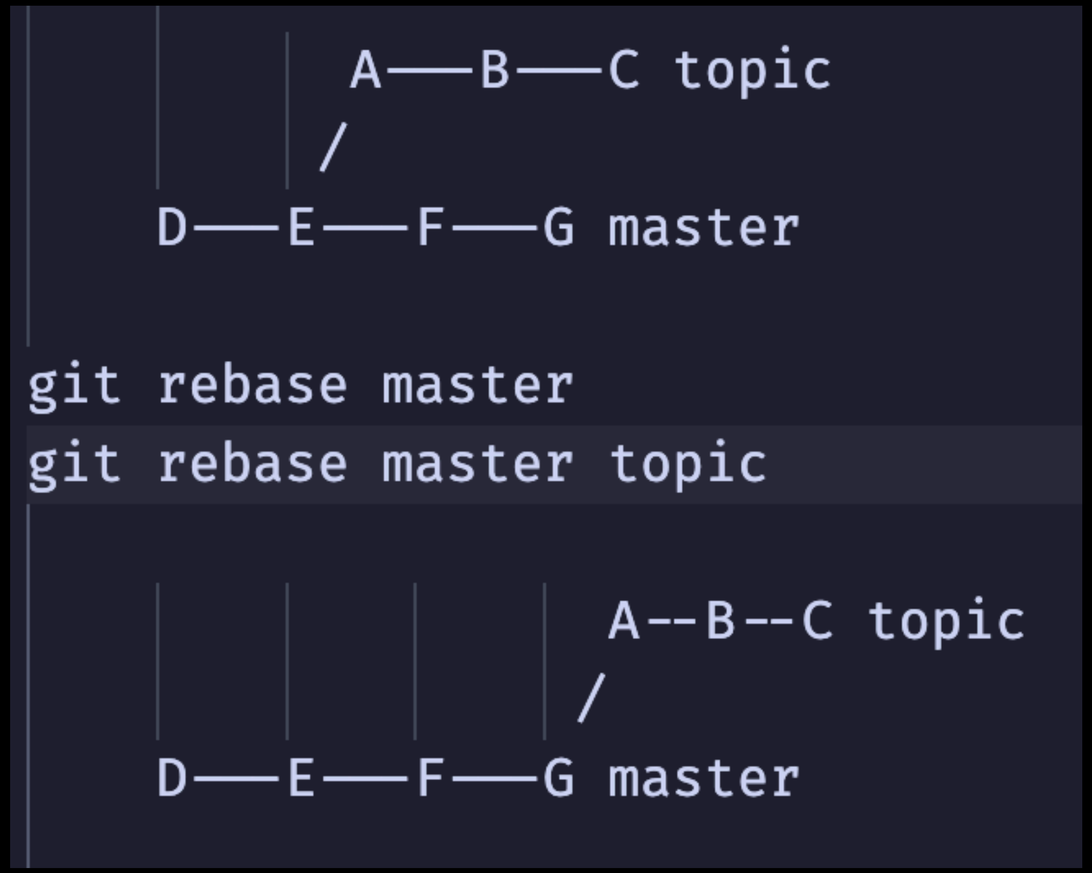
git merge

Incorporates changes from the named commits into the current branch



git rebase

Reapply commits on top of another base tip



git stash

Stash the changes in a dirty working directory away

```
❯ git stash
Saved working directory and index state WIP on main: 3065a2a Add a .gitignore file

test-git on  main [$?] via 💎 v3.3.6
❯ git stash list
stash@{0}: WIP on main: 3065a2a Add a .gitignore file

test-git on  main [$?] via 💎 v3.3.6
❯ git stash show
 test.gemspec | 8 ++++----
 version.rb   | 2 +-
 2 files changed, 5 insertions(+), 5 deletions(-)
```

git log

Shows the commit logs

```
❯ git log
commit d3c1891ea15c0fcd322654fb7de0d5d1b16f829c (HEAD → main)
Author: Dominik Gedon <dominik@gedon.org>
Date:   Wed Nov 27 17:42:49 2024 +0100

    Add .gitignore file

commit c48777ccebe307b9cffb30d87b1343b0c5044b9d (origin/main, origin/HEAD)
Author: Dominik Gedon <dominik@gedon.org>
Date:   Wed Nov 27 15:09:03 2024 +0100

    feat: Add Python Cheatsheet submodule
```

git remote

Manage the set of repositories ("remotes") whose branches you track

```
❱ git remote -v
origin  git@github.com:nodeg/tetra.git (fetch)
origin  git@github.com:nodeg/tetra.git (push)
upstream        git@github.com:uyuni-project/tetra.git (fetch)
upstream        git@github.com:uyuni-project/tetra.git (push)
```

git reset

Resets current HEAD to the specified state

```
git reset [<mode>] [<commit>]

--soft
Does not touch the index file or the working tree at all.

--mixed
Resets the index but not the working tree, the changed files
are preserved but not marked for commit.

--hard
Resets the index and working tree.
```

git revert

Revert some existing commits

```
❯ git revert 3065a2a
[main 14a2f40] Revert "Add a .gitignore file"
 1 file changed, 1 deletion(-)
 delete mode 100644 .gitignore

❯ glo
14a2f40 (HEAD → main) Revert "Add a .gitignore file"
3065a2a (origin/main, origin/HEAD) Add a .gitignore file
c48777c feat: Add Python Cheatsheet submodule
3b596e6 Add Ruby test files
cb9f20c Add README.md
```

```
test-git on ⚡ main [?⇡] via 💎 v3.3.6
❯ git show 3065a2a
commit 3065a2a11217acb60e7bdc85cc8fef9c152b96d9 (origin/main, origin/HEAD)
Author: Dominik Gedon <dominik@gedon.org>
Date:   Wed Nov 27 21:24:48 2024 +0100

    Add a .gitignore file


added: .gitignore
─────────────────────────────────────────────────────────────────────────────

1:

    ⋮  1  *.bin

test-git on ⚡ main [?⇡] via 💎 v3.3.6
❯ git show 14a2f40
commit 14a2f4083ec26cef968c2819d1ee4ffaff072861 (HEAD → main)
Author: Dominik Gedon <dominik@gedon.org>
Date:   Wed Nov 27 21:43:53 2024 +0100

    Revert "Add a .gitignore file"

    This reverts commit 3065a2a11217acb60e7bdc85cc8fef9c152b96d9.


removed: .gitignore
─────────────────────────────────────────────────────────────────────────────

0:

  1 ⋮     *.bin
```

git apply

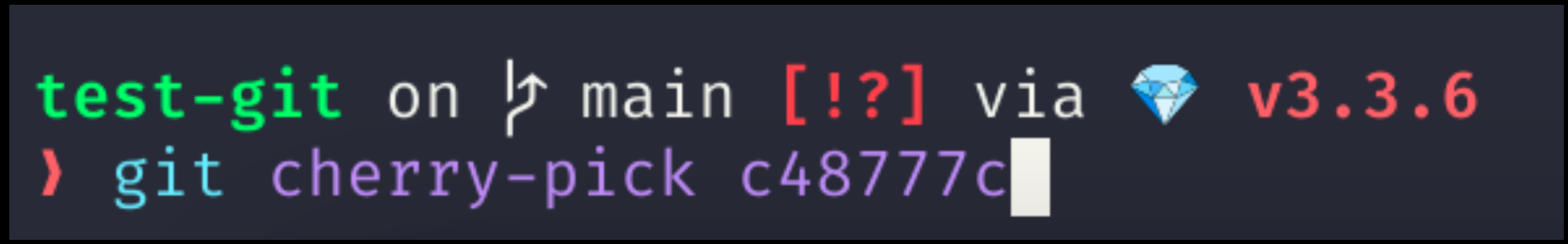
Applies the patch but does not create a commit

```
 Gem::Specification.new do |s|
-  s.name        = "tetra"
+  s.name        = "ttra"
   s.version     = Tetra::VERSION
   s.authors     = ["Dominik Gedon", "Silvio Moioli"]
   s.email       = "dgedon@suse.de"
@@ -27,12 +27,12 @@ Gem::Specification.new do |s|
   s.required_ruby_version = '>= 2.7.0'

   s.add_development_dependency "aruba", "~> 0.6.2"
-  s.add_development_dependency "simplecov", "~> 0.22.0"
-  s.add_development_dependency "rake", "~> 13.2.0"
-  s.add_development_dependency "rspec", "~> 3.13.0"
   s.add_development_dependency "rubocop", "~> 1.64.1"

   s.add_runtime_dependency "clamp", "~> 1.3.2"
+  s.add_development_dependency "simplecov", "~> 0.22.0"
+  s.add_development_dependency "rake", "~> 13.2.0"
+  s.add_development_dependency "rspec", "~> 3.13.0"
```

```
❯ git apply conflict.diff

test-git on  main [!?] via 💎 v3.3.6
❯ git diff

test.gemspec

2: $LOAD_PATH.push File.expand_path("../lib", __FILE__)

  2 :  2 | require "tetra/version"
  3 :  3 |
  4 :  4 | Gem::Specification.new do |s|
  5 :    |   s.name        = "tetra"
    :  5 |   s.name        = "ttra"
  6 :  6 |   s.version     = Tetra::VERSION
  7 :  7 |   s.authors     = ["Dominik Gedon", "Silvio Moioli"]
  8 :  8 |   s.email       = "dgedon@suse.de"

27: Gem::Specification.new do |s|

 27 : 27 |   s.required_ruby_version = '>= 2.7.0'
 28 : 28 |
 29 : 29 |   s.add_development_dependency "aruba", "~> 0.6.2"
 30 :    |   s.add_development_dependency "simplecov", "~> 0.22.0"
 31 :    |   s.add_development_dependency "rake", "~> 13.2.0"
 32 :    |   s.add_development_dependency "rspec", "~> 3.13.0"
 33 : 30 |   s.add_development_dependency "rubocop", "~> 1.64.1"
 34 : 31 |
 35 : 32 |   s.add_runtime_dependency "clamp", "~> 1.3.2"
    : 33 |   s.add_development_dependency "simplecov", "~> 0.22.0"
    : 34 |   s.add_development_dependency "rake", "~> 13.2.0"
    : 35 |   s.add_development_dependency "rspec", "~> 3.13.0"
 36 : 36 |   s.add_runtime_dependency "erb", "~> 4.0.3"
 37 : 37 |   s.add_runtime_dependency "json_pure", ">= 2.6.3", "< 2.8.0"
 38 : 38 |   s.add_runtime_dependency "open4", "~> 1.3.4"
```

git cherry-pick

Apply the changes introduced by some existing commits



Commands - Tasks

1. Create an empty local git repository, create a new GitHub repository and link the local one to the remote one
2. Take a look at the git history and use different parameters
3. Stash changes and apply them again. Then delete them the stash
4. Rest your local repository to 2 commits before the current one
5. Create a patch with some of your changes
6. Cherry-pick a commit
7. Create and push a tag
8. Checkout a GitHub pull request

Best practice

Proper commit messages

Proper commit description

- Why is the change being made?
- Provide references to PRs/issues or resources

Miscellaneous

- commit early and often
- squash related commits
- one change per commit
- do not push straight to *main/master*, but use a branch
- do not change published history

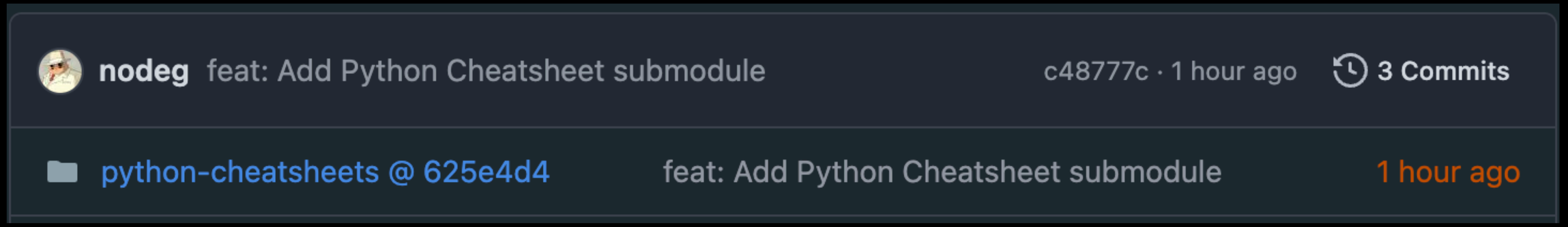
Advanced topics

Maintenance

- *git fsck* - Verifies the connectivity and validity of the objects in the database
- *git gc* - Cleanup unnecessary files and optimize the local repository
- *git clean* - Remove untracked files from the working tree
- *git remote prune* - deletes stale remote-tracking branches
- *git stash clear* - remove all the stash entries
- *git maintenance* - Run tasks to optimize Git repository data

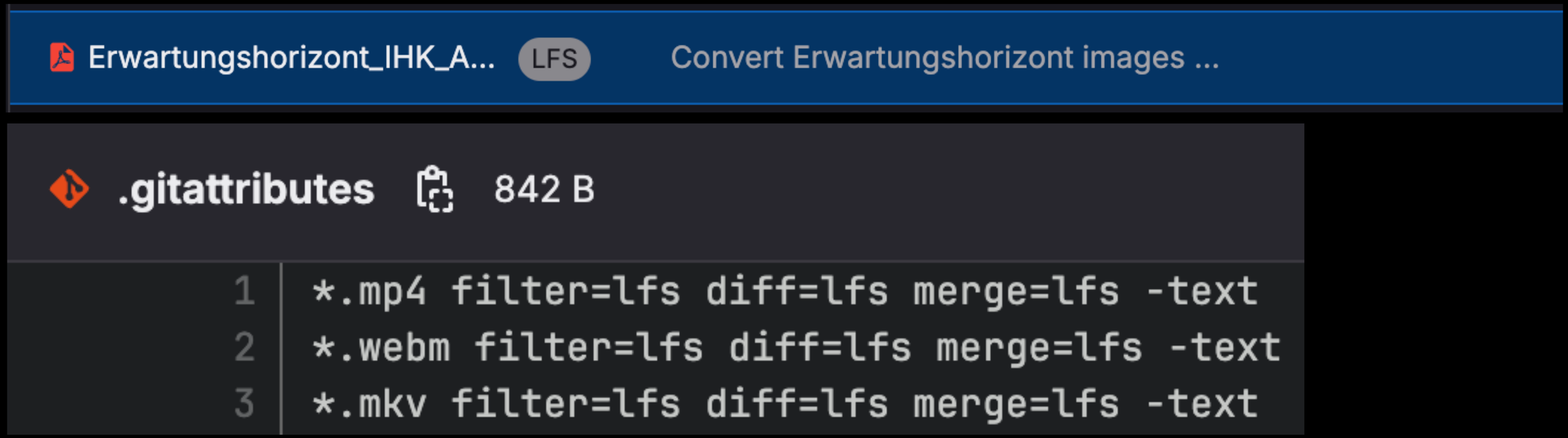
Submodules

- one repository embedded inside another repository
- submodule has its own history



Working with large files

- extension for versioning large files
- large files are replaced by pointers
- storing the file contents on a remote server e.g. gitlab.suse.de



Hooks

- Scripts that run automatically every time a particular event occurs

```
❯ ll .git/hooks/
.rwxr-xr-x dgedon staff 478 B  Wed Nov 27 13:55:48 2024 applypatch-msg.sample
.rwxr-xr-x dgedon staff 896 B  Wed Nov 27 13:55:48 2024 commit-msg.sample
.rwxr-xr-x dgedon staff 4.6 KB Wed Nov 27 13:55:48 2024 fsmonitor-watchman.sample
.rwxr-xr-x dgedon staff 189 B  Wed Nov 27 13:55:48 2024 post-update.sample
.rwxr-xr-x dgedon staff 424 B  Wed Nov 27 13:55:48 2024 pre-applypatch.sample
.rwxr-xr-x dgedon staff 1.6 KB Wed Nov 27 13:55:48 2024 pre-commit.sample
.rwxr-xr-x dgedon staff 416 B  Wed Nov 27 13:55:48 2024 pre-merge-commit.sample
.rwxr-xr-x dgedon staff 1.3 KB Wed Nov 27 13:55:48 2024 pre-push.sample
.rwxr-xr-x dgedon staff 4.8 KB Wed Nov 27 13:55:48 2024 pre-rebase.sample
.rwxr-xr-x dgedon staff 544 B  Wed Nov 27 13:55:48 2024 pre-receive.sample
.rwxr-xr-x dgedon staff  66 B  Wed Nov 27 22:18:28 2024 prepare-commit-msg
.rwxr-xr-x dgedon staff 1.5 KB Wed Nov 27 13:55:48 2024 prepare-commit-msg.sample
.rwxr-xr-x dgedon staff 2.7 KB Wed Nov 27 13:55:48 2024 push-to-checkout.sample
.rwxr-xr-x dgedon staff 2.3 KB Wed Nov 27 13:55:48 2024 sendemail-validate.sample
.rwxr-xr-x dgedon staff 3.6 KB Wed Nov 27 13:55:48 2024 update.sample
```

git reflog

Manage reference log information

```
❯ git reflog
d3c1891 (HEAD → main) HEAD@{0}: commit: Add .gitignore file
c48777c (origin/main, origin/HEAD) HEAD@{1}: reset: moving to HEAD
c48777c (origin/main, origin/HEAD) HEAD@{2}: reset: moving to HEAD
c48777c (origin/main, origin/HEAD) HEAD@{3}: reset: moving to c48777c
6f1aa04 HEAD@{4}: commit: test: Introduce some merge conflicts
c48777c (origin/main, origin/HEAD) HEAD@{5}: reset: moving to c48777c
a8bf561 HEAD@{6}: merge ruby: Merge made by the 'ort' strategy.
c48777c (origin/main, origin/HEAD) HEAD@{7}: checkout: moving from ruby to main
68fc021 (ruby) HEAD@{8}: checkout: moving from main to ruby
c48777c (origin/main, origin/HEAD) HEAD@{9}: checkout: moving from ruby to main
68fc021 (ruby) HEAD@{10}: checkout: moving from main to ruby
c48777c (origin/main, origin/HEAD) HEAD@{11}: cherry-pick: feat: Add Python Cheatsheet submodule
3b596e6 HEAD@{12}: checkout: moving from ruby to main
68fc021 (ruby) HEAD@{13}: commit: feat: Add Python Cheatsheet submodule
81e3ac8 (origin/ruby) HEAD@{14}: checkout: moving from main to ruby
3b596e6 HEAD@{15}: checkout: moving from ruby to main
81e3ac8 (origin/ruby) HEAD@{16}: checkout: moving from main to ruby
3b596e6 HEAD@{17}: checkout: moving from ruby to main
81e3ac8 (origin/ruby) HEAD@{18}: commit (amend): This is a very bad commit message since we do not stick to the rules and
cd1277c HEAD@{19}: commit: This is a very bad commit message since we do not stick to the rules
b54cf75 HEAD@{20}: commit: fix: reorder Ruby dependencies
3b596e6 HEAD@{21}: checkout: moving from main to ruby
3b596e6 HEAD@{22}: reset: moving to 3b596e6
3b596e6 HEAD@{23}: reset: moving to 3b596e6
3b596e6 HEAD@{24}: checkout: moving from ruby_debug to main
3b596e6 HEAD@{25}: checkout: moving from main to ruby_debug
3b596e6 HEAD@{26}: checkout: moving from ruby_debug to main
3b596e6 HEAD@{27}: checkout: moving from main to ruby_debug
3b596e6 HEAD@{28}: commit (amend): Add Ruby test files
4ec3ece HEAD@{29}: commit: Add Ruby test files
cb9f20c HEAD@{30}: clone: from github.com:nodeg/test-git.git
```

Advanced topics - Tasks

1. Use some of the maintenance commands
2. Add a new submodule to a git repository
3. Install and enable git LFS and add a (larger)PDF file to it
4. Create a pre-commit hook

Links

Links

- https://git-scm.com/book/en/v2
- https://rogerdudler.github.io/git-guide
- https://cbea.ms/git-commit
- https://www.conventionalcommits.org
- https://drewdevault.com/2019/02/25/Using-git-with-discipline.html
- https://nvie.com/posts/a-successful-git-branching-model
- https://ndpsoftware.com/git-cheatsheet.html
- https://learngitbranching.js.org