

# GIT

git, github.com Travis CI



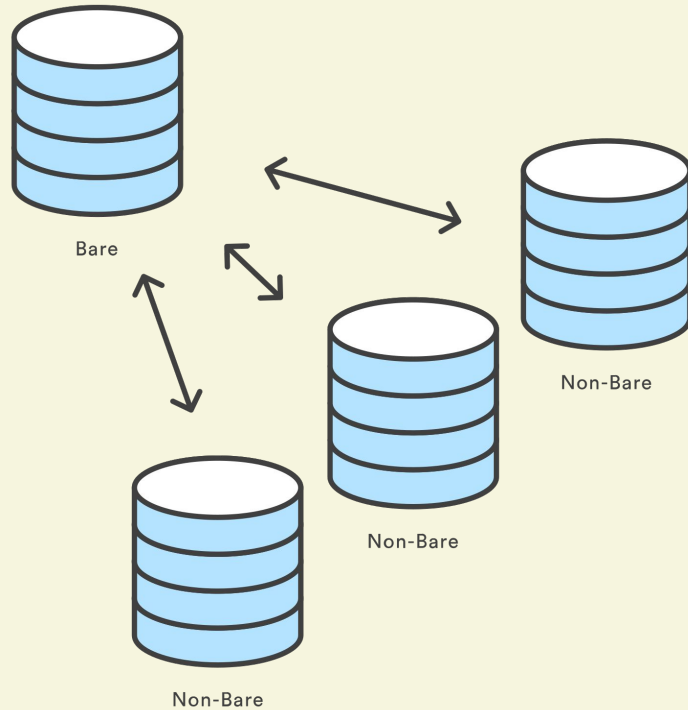
# git

= Schwachkopf, Danke Torvalds

# Was ist git?

Git ist eine freie Software zur **verteilten Versionsverwaltung** von Dateien [...]  
- Wikipedia

# Verteilte Repositories

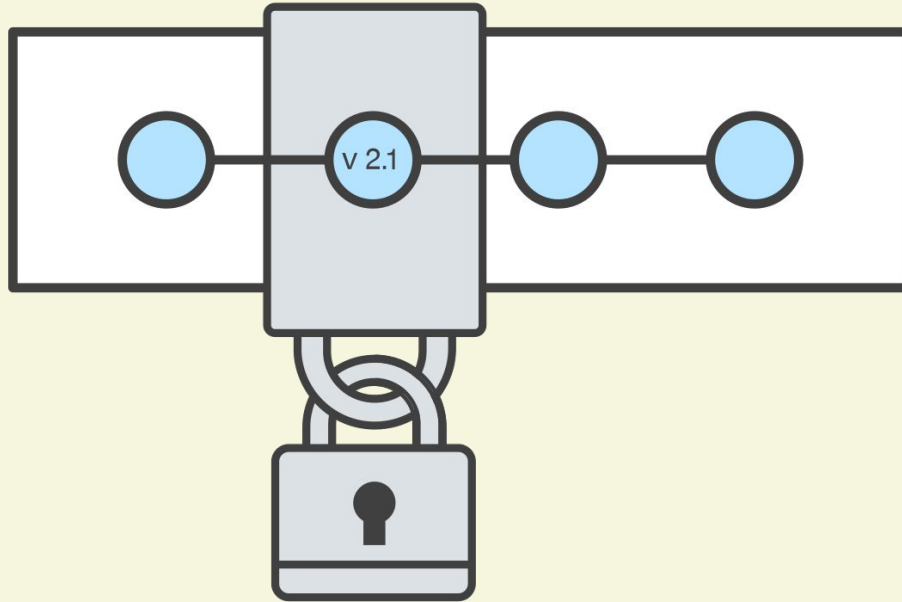


# Version

# ANDROID



# Versionsverwaltung



## git clone

- Lokale Kopie eines Repository
- Remote Repository ist als 'origin' erreichbar

```
$ git clone https://github.com/torvalds/linux.git
```

# git status

```
$ git status
```

Auf Branch master

Ihr Branch ist 1 Commit vor 'origin/master'.

(benutzen Sie "git push", um lokale Commits zu publizieren)

Zum Commit vorgemerkte Änderungen:

geändert: src/lib.rs

Änderungen, die nicht zum Commit vorgemerkt sind:

(benutzen Sie "git add <Datei>...", um die Änderungen zum Commit vorzumerken)

(benutzen Sie "git checkout -- <Datei>...", um die Änderungen im Arbeitsverzeichnis zu verwerfen)

geändert: Cargo.toml

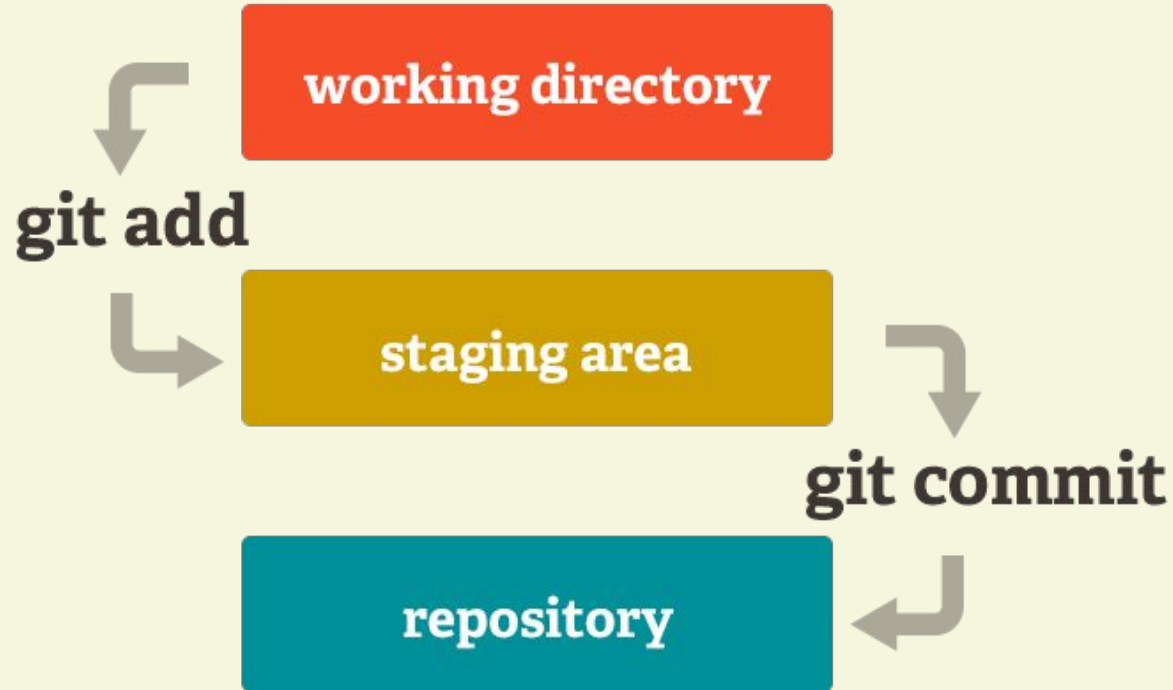
Unversionierte Dateien:

(benutzen Sie "git add <Datei>...", um die Änderungen zum Commit vorzumerken)

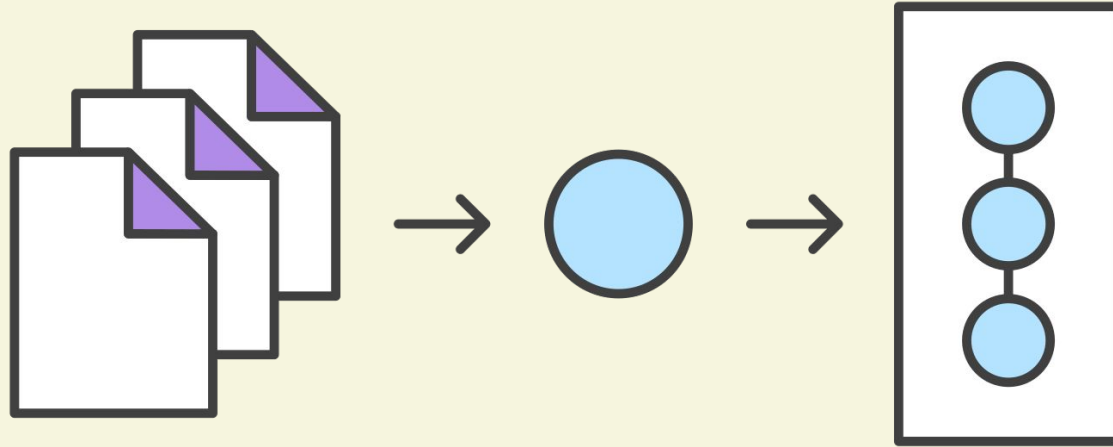
src/lib2.rs



# Dateien im Repository aktualisieren



# Staging Area

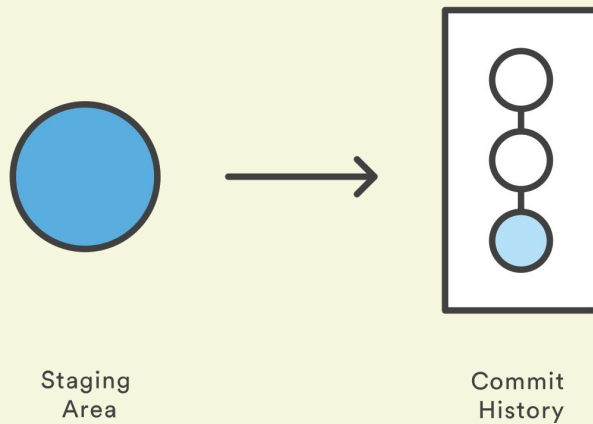


# git add

- Dateien zur Staging Area hinzufügen
- Vorbereitung für einen Commit
- Nur Dateien in der Staging Area werden committed

```
$ git add Cargo.toml
```

# git commit



```
$ git commit -m 'Initial commit'
```

# Commit Messages

Summarize changes in around 50 characters or less

More detailed explanatory text, if necessary. Wrap it to about 72 characters or so. The blank line separating the summary from the body is critical (unless you omit the body entirely).

Further paragraphs come after blank lines.

- Bullet points are okay, too
- Typically a hyphen or asterisk is used for the bullet..

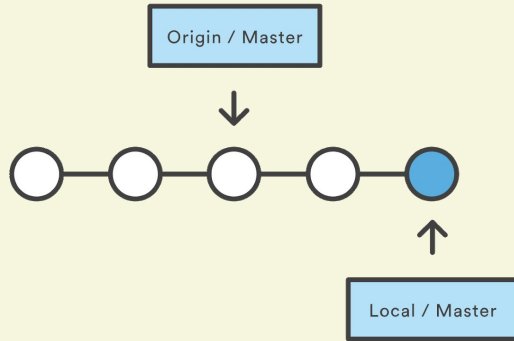
If you use an issue tracker, put references to them at the bottom, like this:

Resolves: #123

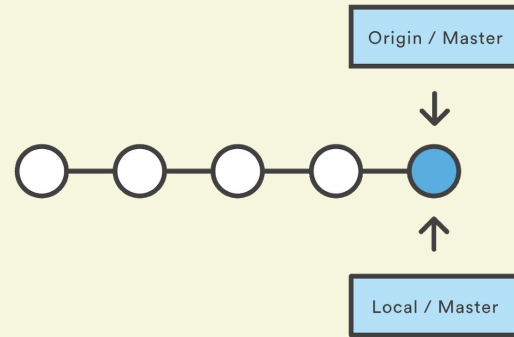
See also: #456, #789

# git push

Before pushing



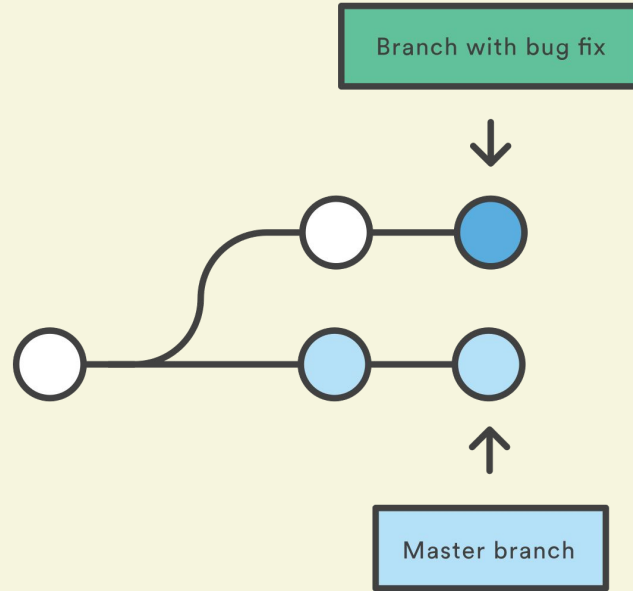
After pushing



# git pull

- aktualisiert lokales Repo auf den Stand von origin
- lädt Commits von anderen Entwicklern herunter und wendet sie lokal an

# Branch





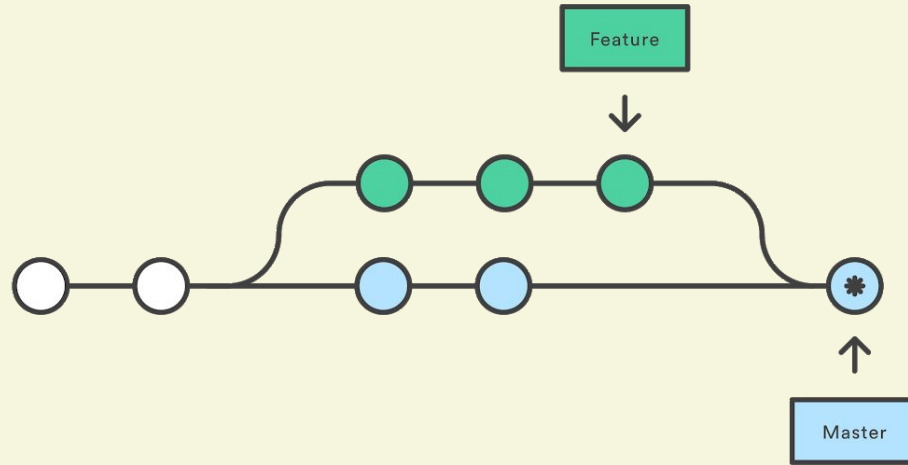
# Branching

erlaubt

- parallele Entwicklung von verschiedenen Features
- Enkapsulierung von Änderungen auf einen Zweig (Branch)
- experimentelle Features und unsicheren Code getrennt von sicherem Code zu verwalten

# git merge

Merge without rebasing



✳ Merge Commit

# Merge

- führt zwei Zweige wieder zusammen
- integriert neue Features oder Bugfixes in den Hauptzweig

# git checkout

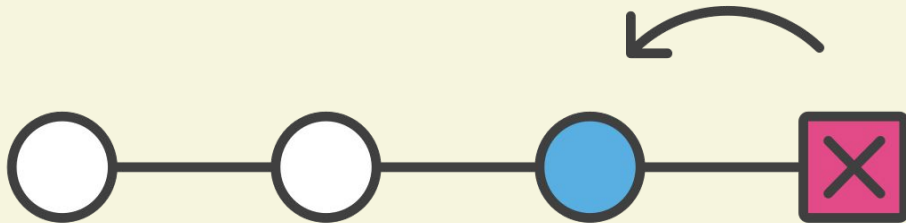
- Wechseln von Branch

```
$ git checkout master
```

# git checkout

- Datei auf ältere Version zurücksetzen

```
$ git checkout [file]
```



# .gitignore

```
# Generated by Cargo  
# will have compiled files and executables  
/target/
```

```
# Remove Cargo.lock from gitignore if creating an executable, leave it for libraries  
# More information here  
https://doc.rust-lang.org/cargo/guide/cargo-toml-vs-cargo-lock.html  
Cargo.lock
```

```
# These are backup files generated by rustfmt  
**/*.rs.bk
```

# git config

(*Wer* veröffentlicht?)

```
$ git config user.name fes0j  
$ git config user.email fes0j@users.noreply.github.com
```

# Weitere git Magie

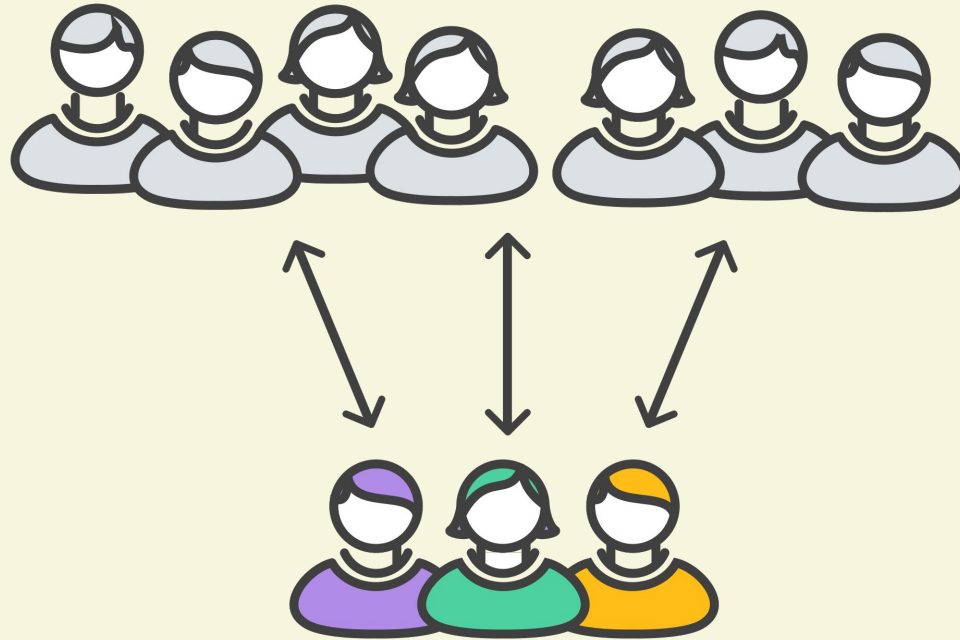
```
$ git init  
$ git reflog  
$ git log  
$ git blame  
$ git revert  
$ git diff  
$ git fetch  
$ git rebase  
$ git rm  
$ git mv  
$ git show  
$ git commit --amend
```



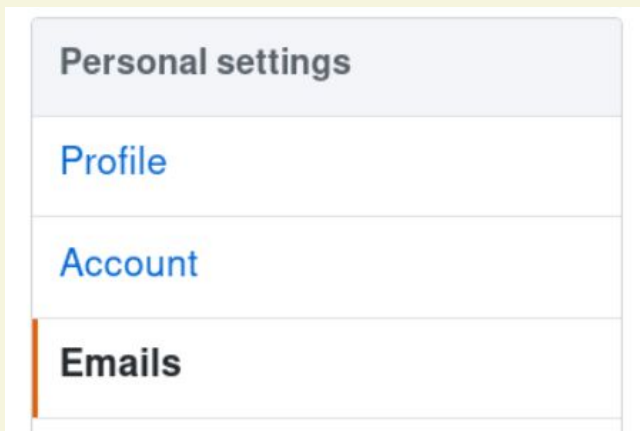


- soziales Netzwerk
- Repos + Issues
- Forks + Pull Requests

# Sozial



# E-Mail schützen



## ☒ Keep my email address private

We'll remove your public profile email and use **fes0j@users.noreply.github.com** when performing web-based Git operations and sending email on your behalf. If you want command line Git operations to use your private email you must [set your email in Git](#).




Travis CI


# Anmeldung: <https://travis-ci.com/>


**Travis CI** [About Us](#) [Plans & Pricing](#) [Enterprise](#) [Help](#)

## The simplest way to test and deploy your projects.

Easily sync your GitHub projects with Travis CI  
and you'll be testing your code in minutes.

 Sign up with GitHub





Sign in to **GitHub**  
to continue to **Travis CI**

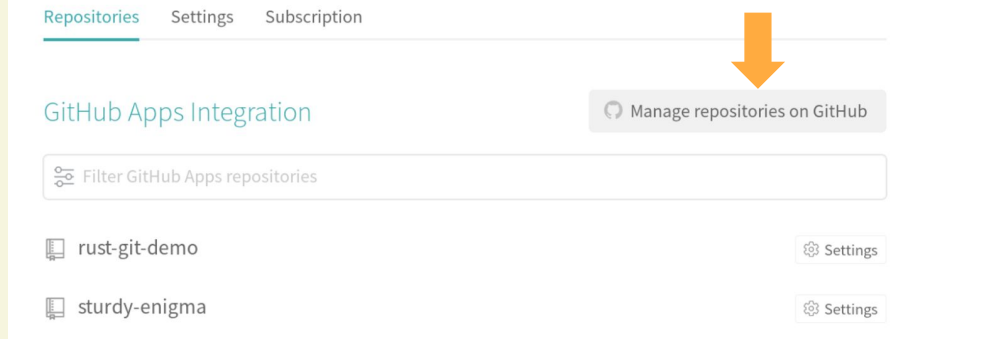
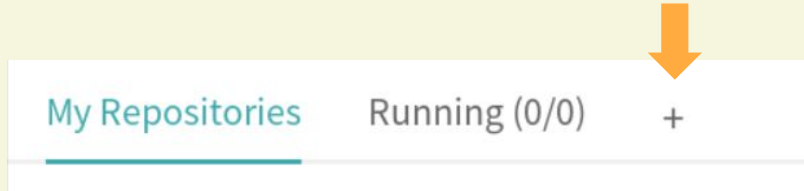
**Username or email address**

**Password** [Forgot password?](#)

Sign in

New to GitHub? [Create an account.](#)

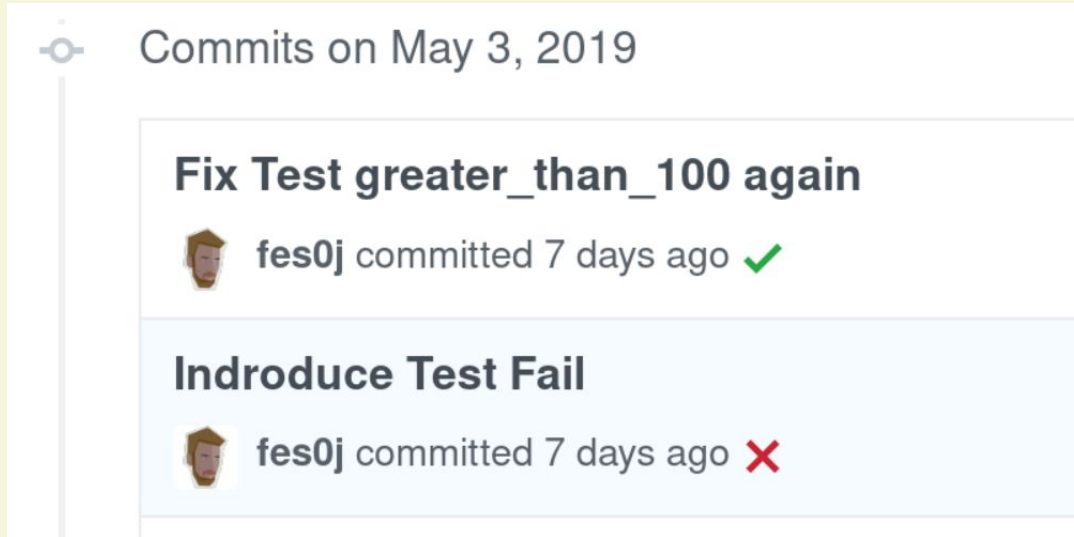
# Verknüpfung




# Konfiguration: .travis.yml


```
language: rust
cache: cargo
script:
  - cargo build --verbose --all
  - cargo test --verbose --all
```


# github + travis



A screenshot of a GitHub commit history snippet. It features a vertical sidebar on the left with a commit icon and a timeline. The main content area lists two commits by user 'fes0j' from 7 days ago. The first commit, 'Fix Test greater\_than\_100 again', is highlighted in white and marked with a green checkmark. The second commit, 'Indroduce Test Fail', is highlighted in light blue and marked with a red X.

 Commits on May 3, 2019


**Fix Test greater\_than\_100 again**  
 fes0j committed 7 days ago ✓

**Indroduce Test Fail**  
 fes0j committed 7 days ago ✗



# Travis Übersicht

Travis CI







[Dashboard](#)

[Changelog](#)

[Documentation](#)

[Help](#)



 fes0j / sturdy-enigma 

build passing


Current

Branches












Build History

Pull Requests

More options



Default Branch

<div> master</div> <div> 5 builds</div>	<div># 5 passed</div> <div> 7 days ago</div>	<div> e36a027 </div> <div> Josef</div>	<div></div>	<div></div>	<div></div>	<div></div>	<div></div>
---	---	---	--	--	--	--	--

# Live Demo

>  
\_