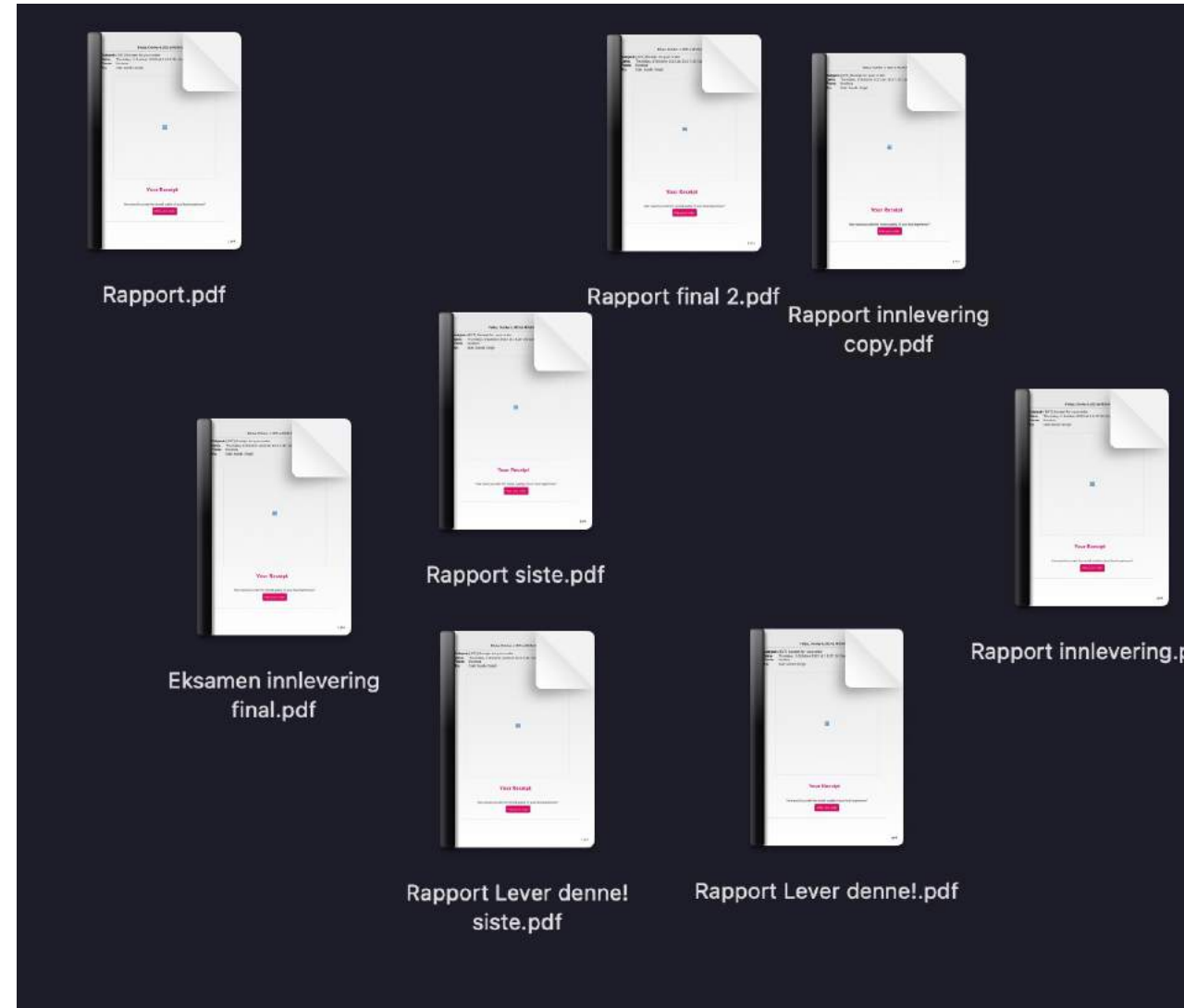


# Kurs i versjonskontroll



# Hva er versjonskontroll?

- En historikk over hva som er gjort og av hvem
- Dette gjør det mulig å:
  - «Rulle tilbake» endringer
  - Finne ut hvor og når noe har gått galt
  - Trygt kunne gjøre endringer uten å miste informasjon om hvordan ting var før



# Versjonskontroll for kode

- Ctrl-Z/undo fungerer bare en kort stund
- Kode endres stadig og kommer med forbedringer og feilrettinger
- Viktig å ha kontroll på de minste endringene for å finne feilkilder
- Enklere å samarbeide
- Enklere å dokumentere
- Finnes en rekke verktøy, men Git er bransjestandard:
  - Mercurial
  - Subversion (SVN)
  - Git
  - Pijul
  - osv



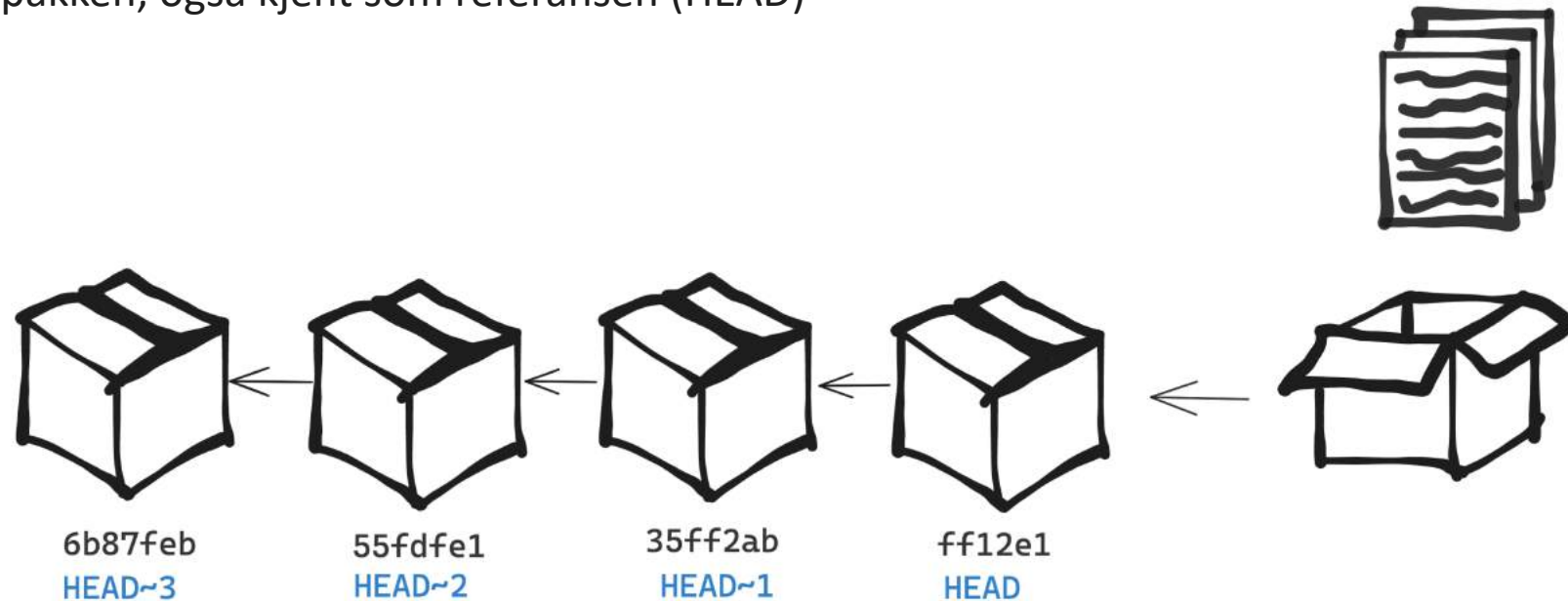
# Litt om Git

- En protokoll for versjonering av filer
- Ble lagd i 2005 for å forenkle arbeid på Linux
- Det holder styr på:
  - Versjon av filene
  - Hvem og hvilke endringer som er gjort når
  - Gjør det enkelt å dele prosjekter og kode
  - Gjør det enkelt å beskrive hva som er gjort og hvor
- De aller, aller fleste bruker git
- Veldig bra verktøy, men litt å skjønne før man kommer i gang med det



# Hvordan fungerer Git?

- Lagrer endringer i små pakker (commit)
- Pakkene settes opp i rekkefølge etter hverandre og har en unik id (commit hash)
- Man kan hente ut og dele pakker ved å bruke referanser
- En referanse peker mot én pakke, og alle tidligere pakker i den rekkefølgen
- Man jobber som regel mot den nyeste pakken, også kjent som referansen (HEAD)
- Nye endringer må man manuelt legge inn i Git for at systemet skal holde følge med hvilke endringer du vil ha med deg (tracke)



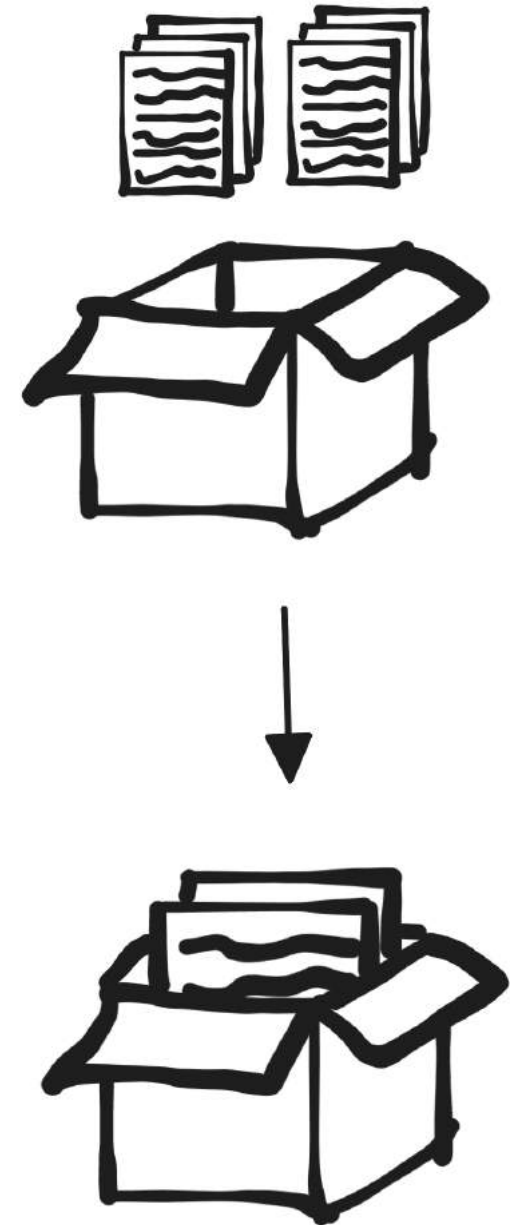
# Hvordan vet du statusen på endringene dine?

- git status
- Viser deg hvilken gren (branch) vi er på
  - Vanligvis kalt main, tidligere kalt master, eller andre grener du lager for å ha forskjellige versjoner på koden din
- Hvilke filer som er endret
- Hvilke filer Git holder styr på (tracker)
- Kjør git status ofte for å holde styr på hva som skjer



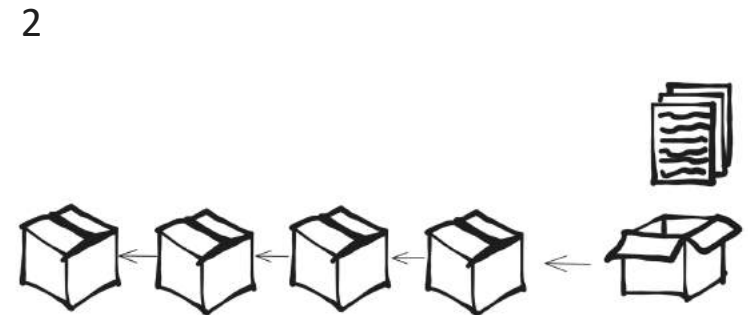
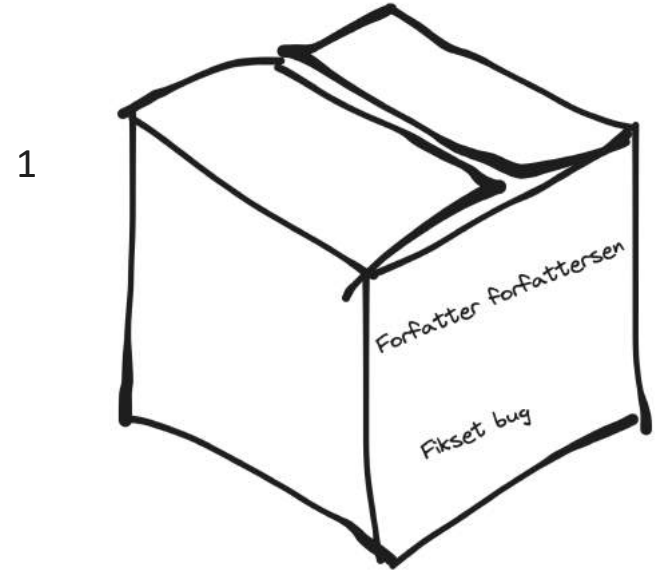
# Hvordan gjør vi endringer?

- Gjør endringen i filene dine
- `git add filnavn.txt`
  - Dette gjør tracker filen/endringene og git status vil vise at filen er tracked
  - Gjør du nye endringer i filnavn.txt må du adde de også
  - Du kan også legge til hele mapper med `git add mappenavn`



# Ferdigstill bolken med endringer

- git commit
1. «Lukk» pakken, gi den en signatur/id, melding som beskriver endringene, tidspunkt med mere.
  2. Alt som var staged/git add vil legges inn i bolken
- Committen blir nå den nyeste i rekken og HEAD referer til denne
  - Når du kjører git commit vil det komme opp et vindu hvor du kan skrive en melding som beskriver committen, eller kjøre `git commit -m "meldingen din"` direkte





# Hvordan ser man historikk?

- git log
- Viser en historikk over tidligere commits, forfatterne og meldingene
- Veldig vanlig å lage en finere versjon for bedre lesbarhet

Fancy:

```
* ea9c7d58 - (HEAD -> main, origin/staging, origin/main, origin/HEAD) Fix label macro for types with generics (#1498) (2 years, 8 months ago) <MinerSebas>
* 8f363544 - Reorder Imports in Examples (#1598) (2 years, 8 months ago) <MinerSebas>
* 9d60563a - Query::get_unique (#1263) (2 years, 8 months ago) <TheRawMeatball>
* 2c203f7b - Bump github/super-linter from v3.15.1 to v3.15.2 (#1596) (2 years, 8 months ago) <dependabot[bot]>
* 13aef050 - impl SystemParam for Option<Res<T>> / Option<ResMut<T>> (#1494) (2 years, 8 months ago) <Jasen Borisov>
* d9b8b3e6 - Add EventWriter (#1575) (2 years, 8 months ago) <TheRawMeatball>
* 2b0a48d9 - feat: clone indices (#1574) (2 years, 8 months ago) <Joshua J. Bouw>
* 58d687b8 - fix flip of contributor bird (#1573) (2 years, 8 months ago) <François>
* a7308155 - Make TypeRegistration::get_short_name() pub (#1571) (2 years, 8 months ago) <Jonas Matser>
* 891f6a1f - Silence annoying rustfmt config warnings (#1508) (2 years, 8 months ago) <Nathan Stocks>
* dabf4190 - update archetypes if needed before running system in SingleThreadedExecutor (#1586) (2 years, 8 months ago) <Carter Anderson>
* 2a3a32b6 - Add `mesa-vulkan-drivers` to Debian install (#1489) (2 years, 8 months ago) <Psychoticpotato>
* f61e44db - Update glam to 0.13.0. (#1550) (2 years, 8 months ago) <Cameron Hart>
* 0eba5f38 - update hexasphere to 3.2 (#1577) (2 years, 8 months ago) <Carter Anderson>
* 03e0a9f2 - Docs for Bundle showing how to nest bundles (#1570) (2 years, 8 months ago) <Alice Cecile>
* 00684831 - Documented some of the Mesh properties (#1566) (2 years, 8 months ago) <sdfgeoff>
* 64b29617 - Added documentation on the query filters (#1553) (2 years, 8 months ago) <sdfgeoff>
* 87399c35 - Fix staging buffer required size calculation (fixes #1056) (#1509) (2 years, 8 months ago) <Renato Caldas>
* b2d654cb - Use rand 0.8 again (#1567) (2 years, 8 months ago) <MinerSebas>
* ab407aa6 - o `Timer` refactor to duration. + Add `Stopwatch` struct. (#1151) (2 years, 8 months ago) <Chris Janaqi>
* 4a9e7aa9 - o `Timer` refactor to duration. + Add `Stopwatch` struct. (#1151) (2 years, 8 months ago) <Chris Janaqi>
```

Vanlig:

```
commit ea9c7d58ffd5bd196406d72a9331ca30188b1137 (HEAD -> main, origin/staging, origin/main, origin/HEAD)
Author: TheRawMeatball <therawmeatball@gmail.com>
Date: Tue Mar 9 03:49:48 2021 +0000

    Fix label macro for types with generics (#1498)

    Fixes #1497

    Co-authored-by: Carter Anderson <mcanders1@gmail.com>

commit 8f363544ad2812e08384dbbc3e8ea6fea0fb138c
Author: MinerSebas <scherthan_sebastian@web.de>
Date: Tue Mar 9 01:07:01 2021 +0000

    Reorder Imports in Examples (#1598)

    This only affected 2 Examples:
    * `generic_reflection`: For some reason, a `pub use` statement was used. This was removed, and all
    * `wireframe`: This example used the `bevy_internal` crate directly. Changed to use `bevy` instead.

    All other Example Imports are correct.

    One potential subjective change is the `removel_detection` example.
    Unlike all other Examples, it has its (first) explanatory comment before the Imports.

commit 9d60563adf4167059fcd1a0addc11487a4394c15
Author: TheRawMeatball <therawmeatball@gmail.com>
Date: Mon Mar 8 21:21:47 2021 +0000

    Query::get_unique (#1263)

    Adds `get_unique` and `get_unique_mut` to extend the query api and cover a common use case. Also

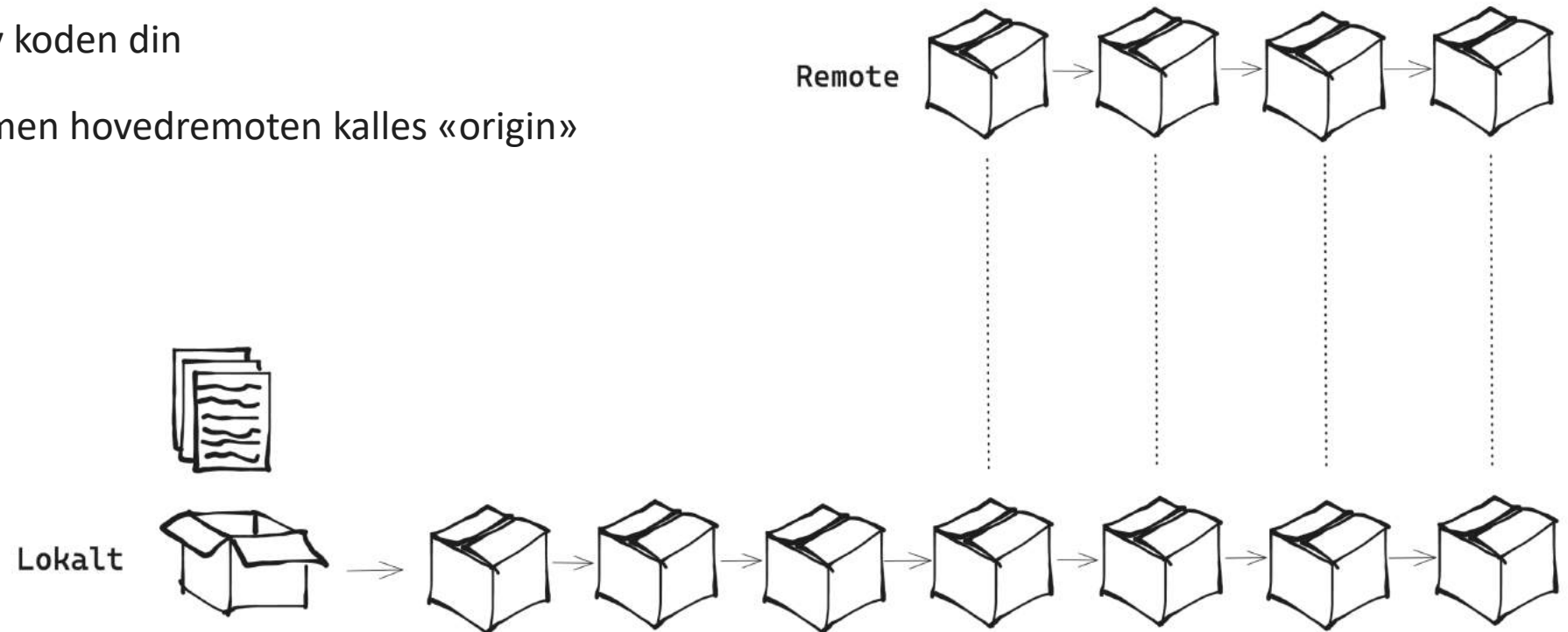
commit 2c203f7b8fd8daad5af836040898d316466c81b8
Author: dependabot[bot] <49699333+dependabot[bot]@users.noreply.github.com>
Date: Mon Mar 8 20:27:44 2021 +0000

    Bump github/super-linter from v3.15.1 to v3.15.2 (#1596)

    Bumps [github/super-linter](https://github.com/github/super-linter) from v3.15.1 to v3.15.2.
    <details>
    <summary>Release notes</summary>
    <p><em>Sourced from <a href="https://github.com/github/super-linter/releases">github/super-linter</a></em></p>
    <blockquote>
    <h2>Release v3.15.2</h2>
    <h2>Changelog</h2>
    <ul>
    <li>Fix issue with grabbing history</li>
    </ul>
    <h2>Bugs</h2>
    <p>we found a bad one...</p>
    </blockquote>
    </details>
```

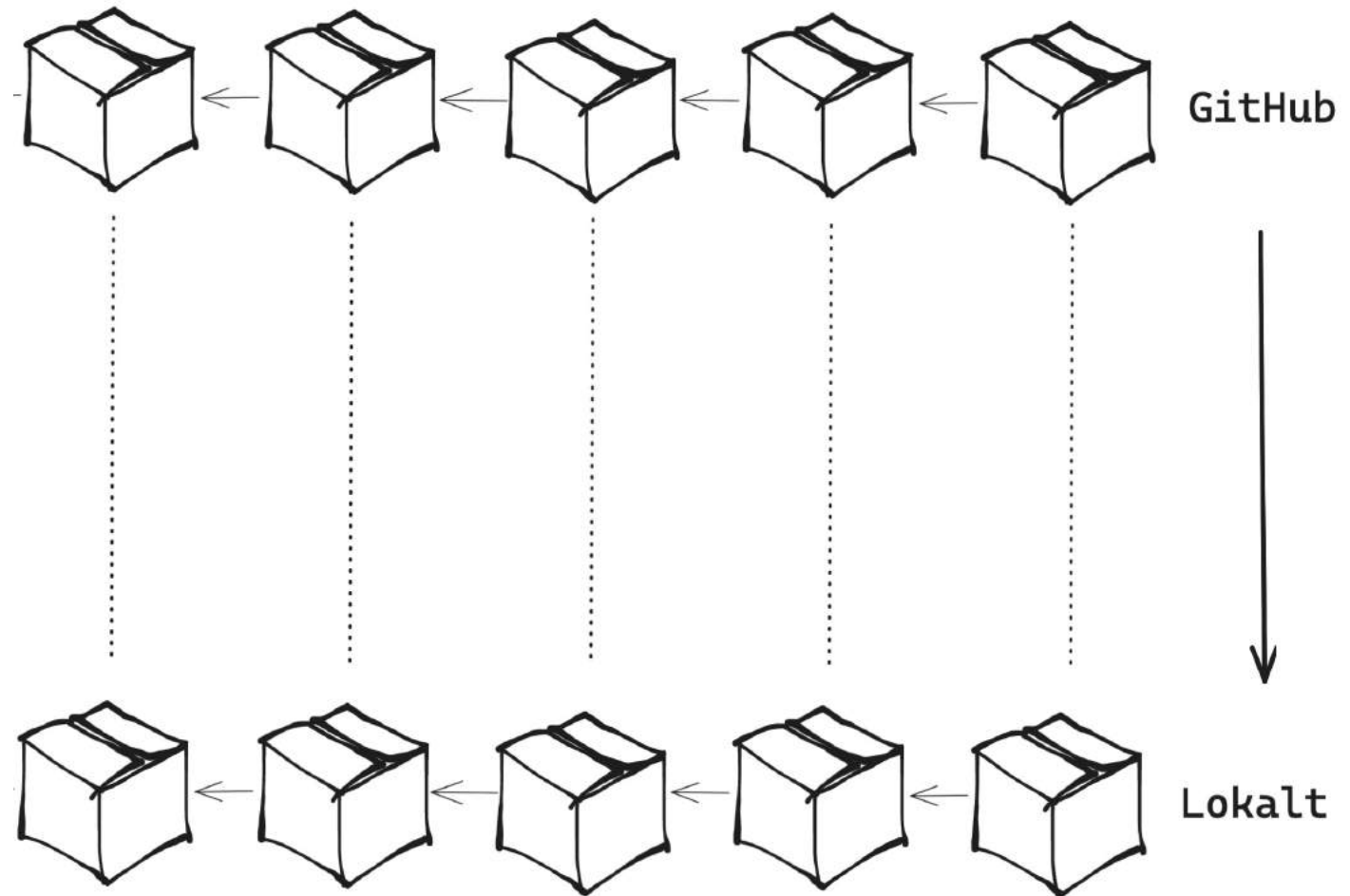
# Remotes

- `git remote -v`
- For at andre skal kunne få endringene dine må du dytte endringene til en remote (f.eks. GitHub)
- En remote holder en kopi av koden din
- Man kan ha flere remotes, men hovedremoten kalles «origin»



# Klone kode fra en remote

- `git clone <url>`
- Kopierer et prosjekt til din maskin
- Brukes ofte én gang når du starter på et prosjekt
- Eller om du vil få en «fresh start»



# Samhandling med remote

- git pull
  - Henter ned commits fra en remote repository (kodebase)
- git push
  - Dytter commits fra din maskin til en remote (origin om du ikke spesifiserer noe annet)

