



»brig«

Ein Werkzeug zur sicheren und verteilten

Dateisynchronisation

Um was geht's?

- Einführung
- Das Problem
- Demo
- Was hab ich da grad gesehen?
- Hilfe!
- ~~Applaus!~~ Fragen?

Wer ist'n das?

Chris Pahl.

Wer mehr über mich wissen will:

<https://sahib.github.io>

Es war einmal...



Das Problem

YOU WANT YOUR COUSIN TO SEND YOU A FILE? EASY.
HE CAN EMAIL IT TO— ... OH, IT'S 25 MB? HMM...

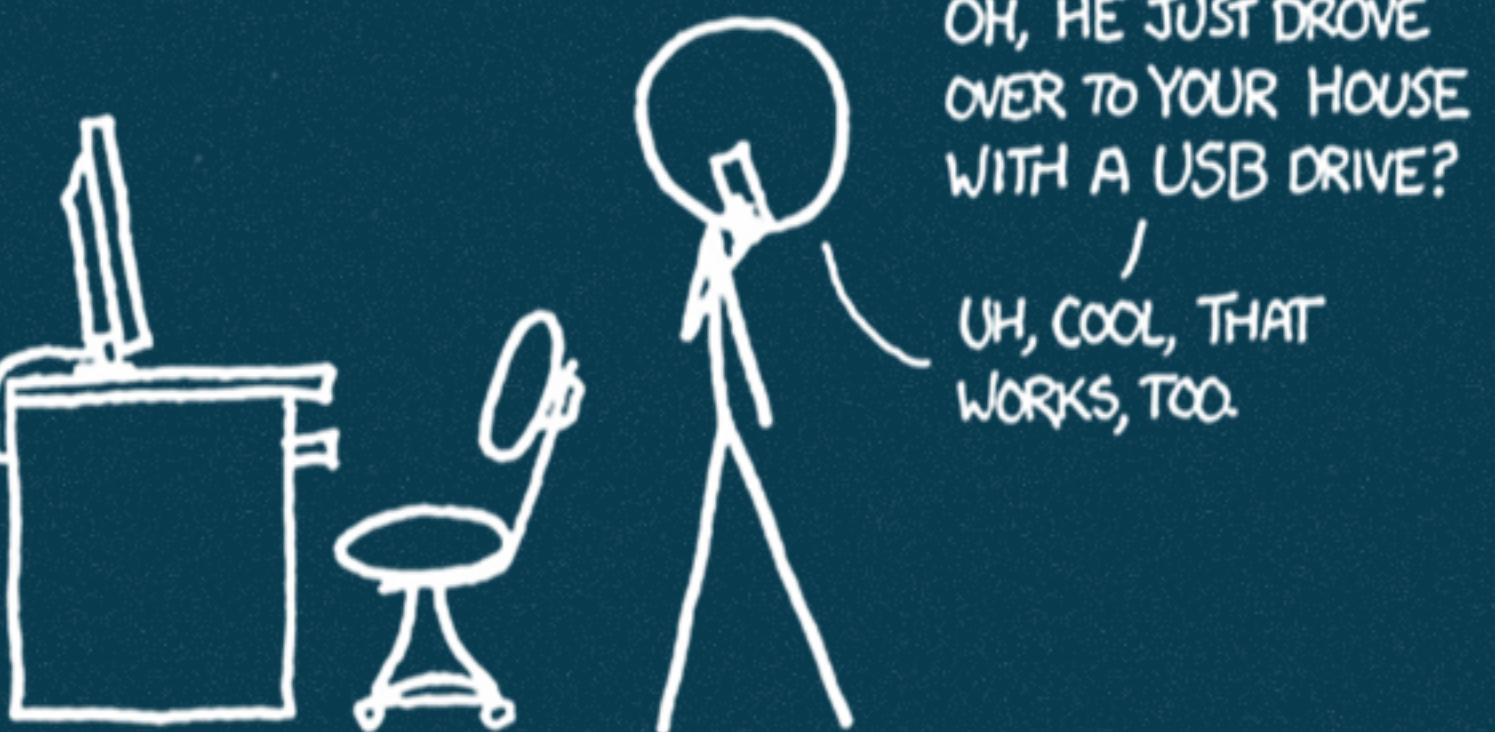
DO EITHER OF YOU HAVE AN FTP SERVER? NO, RIGHT.

IF YOU HAD WEB HOSTING, YOU COULD UPLOAD IT...

HMM. WE COULD TRY ONE OF THOSE MEGASHAREUPLOAD SITES,
BUT THEY'RE FLAKY AND FULL OF DELAYS AND PORN POPUPS.

HOW ABOUT AIM DIRECT CONNECT? ANYONE STILL USE THAT?

OH, WAIT, DROPBOX! IT'S THIS RECENT STARTUP FROM A FEW
YEARS BACK THAT SYNCs FOLDERS BETWEEN COMPUTERS.
YOU JUST NEED TO MAKE AN ACCOUNT, INSTALL THE—



OH, HE JUST DROVE
OVER TO YOUR HOUSE
WITH A USB DRIVE?

UH, COOL, THAT
WORKS, TOO.

Was ist das Ziel?



MAKE FILE SYNCING GREAT AGAIN!

Geht das auch detaillierter?

Dinge die Dateiaustausch sein sollte:

- Einfach.
- Sicher.
- Schnell.
- Versioniert.
- Dezentral.
- Frei.

Aber, aber...

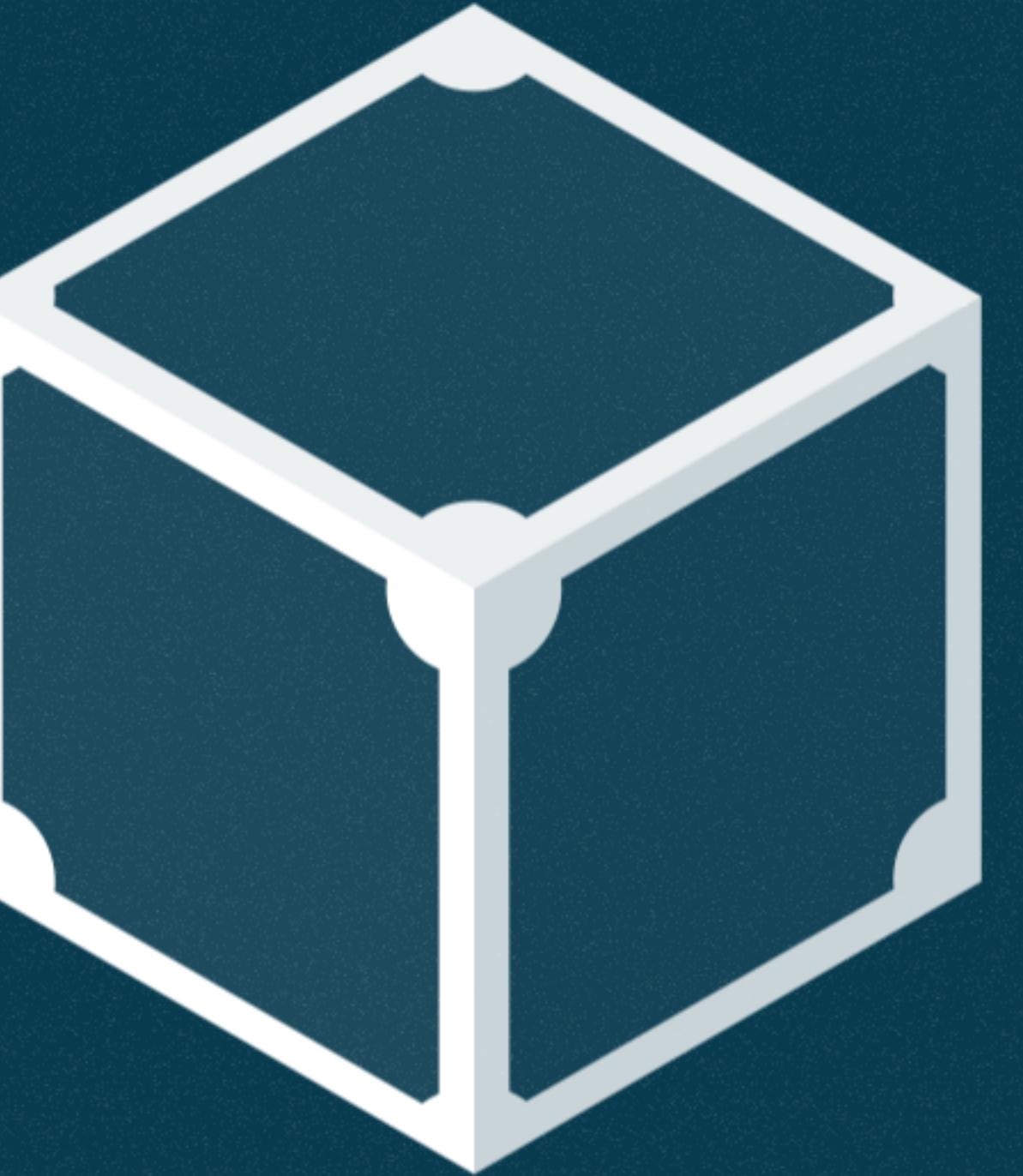
Ja, es gibt schon einige dezentrale Tools.



- Anderer Fokus
- Andere Basis
- Andere Features
- Gleiches Ziel.

(Siehe: <https://brig.readthedocs.org/comparison.html>)

IPFS



»Inter-Planetary-File-System«

Was kann das so?

```
$ echo 'Hallo Augsburg!' | ipfs add  
added QmbLr7bEQkC85EEGEmQk42dLz25VBy2L6iHyZQu
```

```
$ ipfs cat QmbLr7bEQkC85EEGEmQk42dLz25VBy2L6iHyZQu  
Hallo Augsburg!
```

```
$ ipfs id -f '<id>\n'  
QmeLNNcryy9Ky1dXnfnEPaDQ2KuJ6yafaSRZssjQ83ie84
```

»brig«



- Hash Nanny für ipfs.
- In GO geschrieben.
- **Zielgruppe:** Linux User. Erstmal.

Entwicklungsgeschichte:

- Ende 2015: *Masterprojekt*.
- Ende 2016: *Pausiert*.
- Ende 2017: *Hobbyprojekt*.
- Erster Beta Release heute!



Kurz gesagt: Fokus

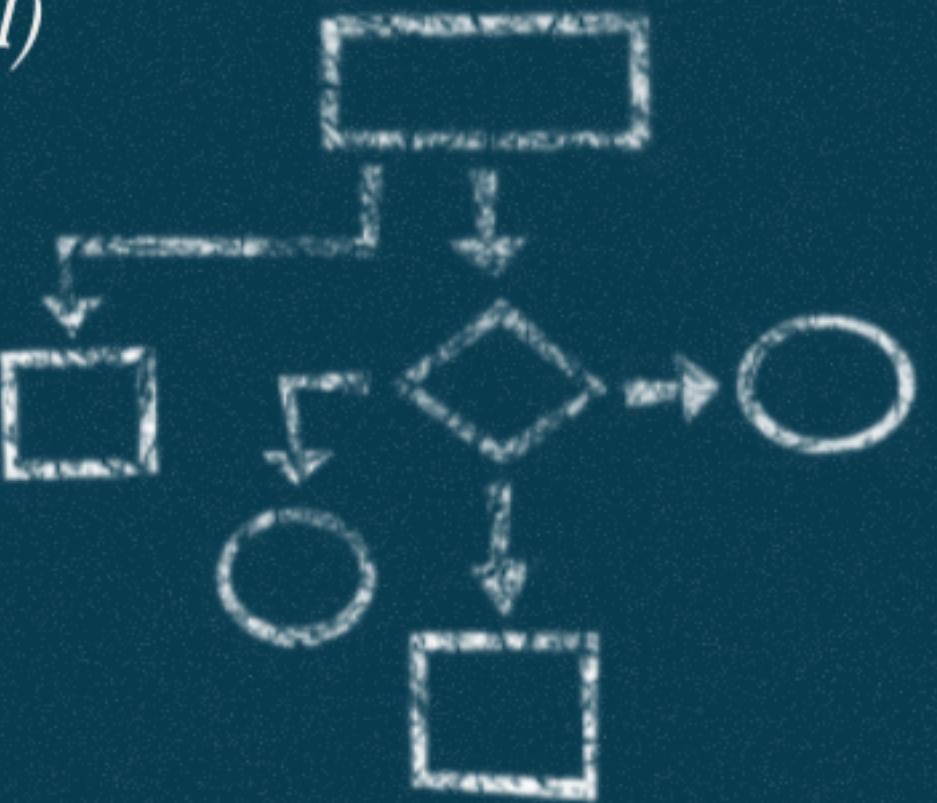
- Balance zwischen Sicherheit und Usability.
- Effizienz ist nett, aber kein primärer Fokus.
- Kompatibilität zu gewohnten Konzepten.
- Komplexität hinter einem einfachen Interface.

Demo

```
$ brig mv raiders twix  
# sonst ändert sich aber nix.
```

Workflow

- Initialer Setup (*nur einmal*)
- Remotes finden & hinzufügen (*nur einmal*)
- Diff anzeigen (*optional*)
- Synchronisieren
- Konflikte beheben



Disclaimer: Sicherheit?



Dezentralität



Nutzermanagement

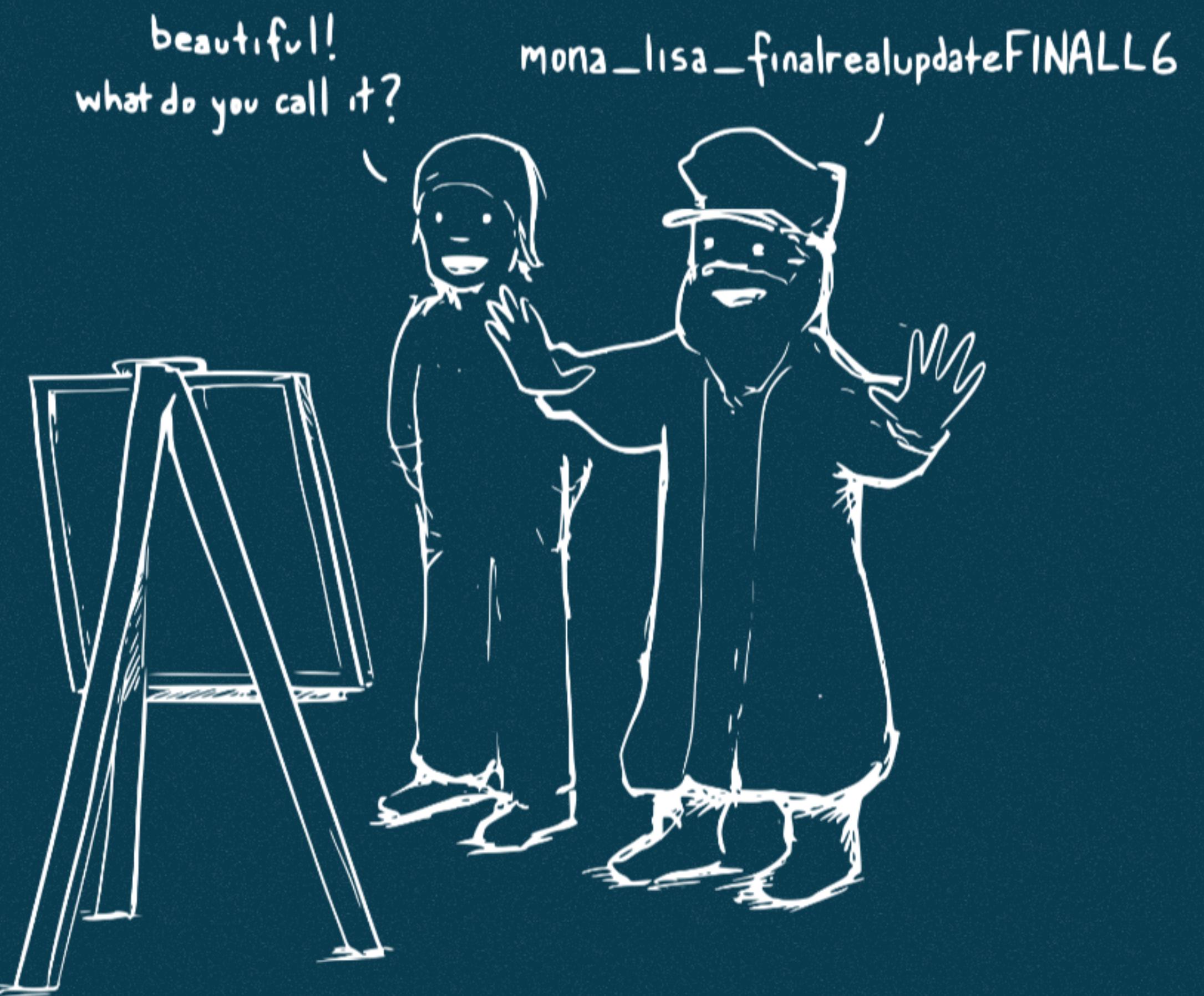
donald@whitehouse.gov/ovaloffice

The diagram illustrates the structure of the email address "donald@whitehouse.gov/ovaloffice". It is divided into three segments by horizontal arrows pointing to the right. The first segment, "donald", is labeled "Name". The second segment, "whitehouse.gov", is labeled "Domain". The third segment, "ovaloffice", is labeled "Resource".

Nutzen:

- Resourcen und Domains zur Discovery.
- Später (*optionale*) E-Mail Authentifizierung.

Versionierung

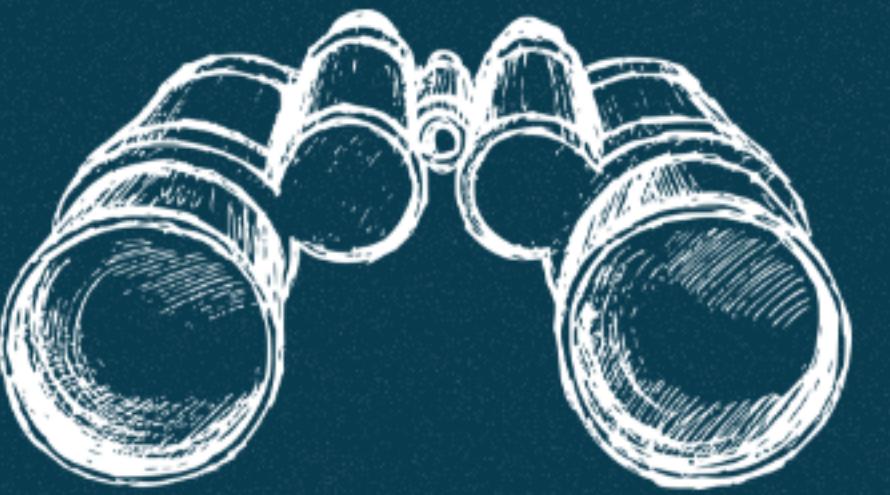


Pinning



Roadmap

- Selektives Sharing.
- Gateway für Hyperlinks.
- Archive Instances.
- (*optional*) Autosync.
- Performance, mehr Dokumentation...



Hauptproblem: Nur ein Entwickler.

Hilfe? Erwünscht.

Mithilfe via Experience Reports.

1. Was habt ihr gemacht?
2. Was habt ihr erwartet?
3. Warum hat das nicht funktioniert?

Und sonst?

- Bug reports. (`brig bug`)
- Pull requests.
- **Keine** Feature Requests!



Probem gelöst?

HOW STANDARDS PROLIFERATE:
(SEE: A/C CHARGERS, CHARACTER ENCODINGS, INSTANT MESSAGING, ETC)

SITUATION:
THERE ARE
14 COMPETING
STANDARDS.

14?! RIDICULOUS!
WE NEED TO DEVELOP
ONE UNIVERSAL STANDARD
THAT COVERS EVERYONE'S
USE CASES.



SOON:

SITUATION:
THERE ARE
15 COMPETING
STANDARDS.

Letzte Worte

 <http://brig.rtfd.org>

 github.com/sahib/brig

 <http://sahib.github.io/brig/public>

Fragen?