



# GetTogetherR!

## Git und R

---

Jonas Frost

[jonas.frost@studserv.uni-leipzig.de](mailto:jonas.frost@studserv.uni-leipzig.de)

Peter Kannewitz

[peter.kannewitz@uni-leipzig.de](mailto:peter.kannewitz@uni-leipzig.de)

12. April 2023

1. Kurzeinstieg: Git
2. Setup mit RStudio
3. Ressourcen und Hilfe

- monatliche Veranstaltung (jeden 2. Donnerstag im Monat)
- kurze Inputvorträge von uns oder unseren Gästen
- aktuelle Themen rund um R und RStudio:
  - R und Git (13.04.2023)
  - Netzwerkanalyse in R [Till Hovestadt] (11.05.2023)
  - ChatGPT in RStudio (08.06.2023)
  - Hausarbeiten mit RMarkdown (13.07.2023)
- aktueller Syllabus im [Moodlekurs](#)

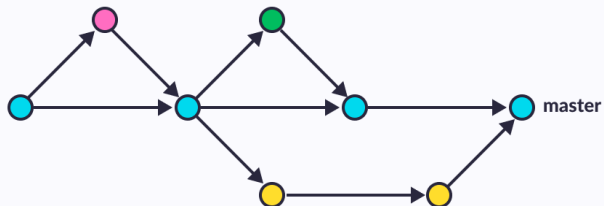
- Git entwickelt sich zum Standard für viele Anwendungsfelder
  - Kollaboration
  - Backup
  - Dokumentation
- RStudio bietet gute und einfache Möglichkeit zur Versionskontrolle via Git
- Funktion ist oft nicht Teil von Einführungskursen und wird daher wenig genutzt

## Kurzeinstieg: Git

---



- Programm zur Versionskontrolle
  - kann komplett lokal genutzt werden
  - Änderungen werden Schrittweise durchgeführt (Commits)
  - Änderungen können Rückgängig gemacht werden (revert)
  - Ältere Versionen des Projekts können wieder hergestellt werden
  - Verschiedene Versionen des Projekts können gleichzeitig bearbeitet werden und später zusammengeführt (Branching)



**Figure 1:** Inkrementale Arbeit an einem Projekt

# Git Hoster (Remote Git-Repository)

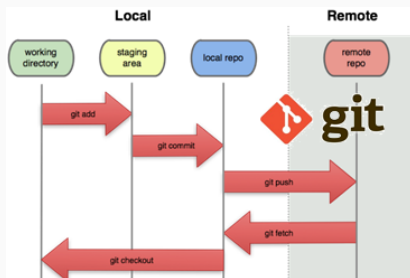


Figure 2: Zusammenarbeit über ein Remote Repository

- Git-Hoster: [GitHub](#), [GitLab](#), [Gitea](#), ...



## Setup mit RStudio

---

- Was braucht man alles?
  - Git auf dem Rechner installiert
  - Account bei einem Git-Hoster (bspw. GitHub)
  - bestehendes Git-Repository
- Schritte:
  - neues R-Projekt erstellen
  - Version Control
  - URL des Repositorys angeben



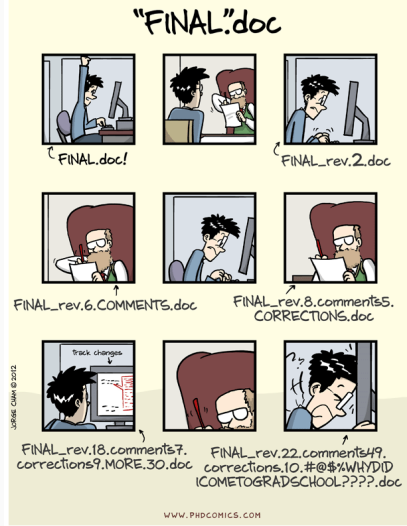
- GitHub ist nicht für große Daten gedacht (Filelimit 100MB)
- man muss Änderungen regelmäßig pushen, sonst hat man nichts gewonnen
- bei Problemen (gerade beim Kollaborieren) muss man sich näher mit Git auseinandersetzen (merge conflicts)

# Warum sich die Mühe machen?

- Änderungen tracken, dokumentieren und rückgängig machen
- gleichzeitig im Team zusammenarbeiten und Änderungen zusammenführen
- Projekte sicher abspeichern
- GitHub bietet unglaublich viel Potential für weitere Anwendungsfälle (Aufgabenplanung, Release, Continuous Integration)

Piled Higher and Deeper by Jorge Cham

www.phdcomics.com



## Ressourcen und Hilfe

---

- <https://r-bio.github.io/intro-git-rstudio/> (Einstiegs Tutorial)
- <https://posit.co/resources/videos/managing-part-2-github-and-rstudio/> (Video zum Einstieg)
- <https://happygitwithr.com/> (Umfassende Referenz)

- Welche Erfahrung habt Ihr mit Git und RStudio? Tipps? Tricks?
- Habt Ihr Ideen für eigene Anwendungsfälle? Wofür wollt Ihr die RStudio-Integration von Git benutzen?



- Ausprobieren?

- [https://geniusee.com/storage/app/media/blog/blog223\\_git\\_branching\\_model/GitHub\\_Flow.png](https://geniusee.com/storage/app/media/blog/blog223_git_branching_model/GitHub_Flow.png)
- <https://git-scm.com/images/logo@2x.png>
- <https://www.inflectra.com/Images/Product-Imagery/Git.png>
- [https://www.freepik.com/free-icon/business-presentation\\_772083.htm](https://www.freepik.com/free-icon/business-presentation_772083.htm)
- [https://phdcomics.com/comics/archive\\_print.php?comid=1531](https://phdcomics.com/comics/archive_print.php?comid=1531)