

Version Controlling mit Git + Github + RStudio

Claudia Neuendorf & Benjamin Becker

18. August 2021

Interner IQB Workshop

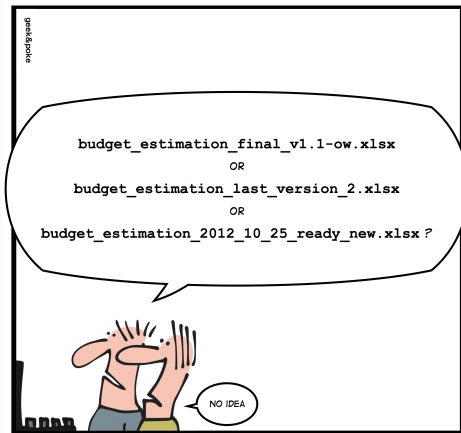
Agenda

- Motivation
- Setup
- Work Flows
- Empfehlungen
- Ressourcen

Disclaimer

Motivation

SIMPLY EXPLAINED



VERSION CONTROL

Solo-Projekte

- Implementation einer 'Change History'
 - Was wurde geändert?
 - Wann wurde es geändert?
- Keine verrückten Dateinamen
- Kein Archiv-Ordner
- Zugänglichkeit für andere ('Open Science')
- Sicherung/Archivierung
- ...

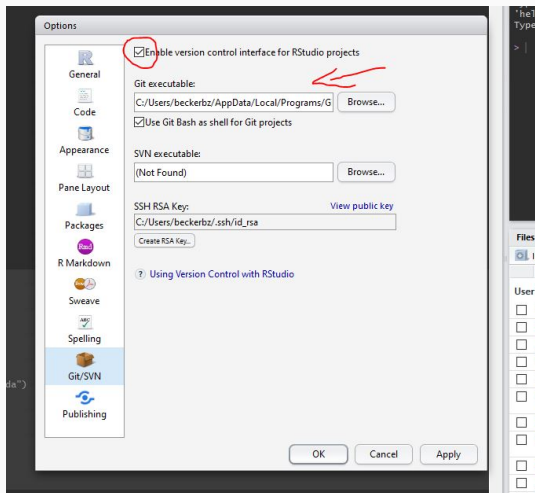
Kollaborationen

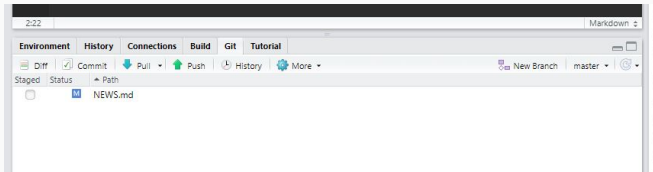
- Wer hat wann was geändert?
- Eindeutiger, aktueller Projektstand
- Keine nervigen Mail-Anhänge oder Austausch-Plattformen
- Erlaubt einfaches, paralleles Arbeiten
- Implementiert hierarchische Projekt-Strukturen (Absegnen von Änderungen)
- ...



Setup

- Git-Installation
- RStudio-Installation (optional)
 - Alternativen: Gitkraken, SmartGit, ...
- Account anlegen für Github
 - Alternativen: Bitbucket, Gitlab, ...
- Alles verbinden





Work Flows

Erstellen eines Repositories

- Erstellen eines Online-Repositorys (z.B. auf Github)
 - Benutzen einer R-spezifischen .gitignore
 - Initialisieren mit einer kurzen Readme-Datei (.md)
- Repository lokal auf den PC clonen
- (optional) R-Projekt in bestehendes Repository einfügen

- Einfache Text-Datei
- Welche Dateien sollen nicht archiviert werden?
→ Diese dann nur lokal in ihrer aktuellen Version vorhanden!
- Optionen
 - Einzelne Dateien
 - Ordner
 - Bestimmte Datei-Typen
 - Kombinationen daraus
- Anwendungs-Beispiele
 - Große Dateien (Datensätze, Bilder, ...)
 - Hilfsdateien (z.B. bei LaTeX-Kompilierungen)

Arbeit an einem bestehenden Repository

- Davor: Lokales Repo synchronisieren (**pullen**)
- Änderungen im lokalen Repository
→ Dateien erstellen/überschreiben/löschen
- Änderungen **stagen**
- Gestagede Änderungen **commiten** (aka neue Version)
- Einen oder mehrere Commits **pushen** (Online-Repository updaten)

Empfehlungen

- So simpel wie möglich halten!
 - Wenn möglich keine Branches/Forks/Pull Requests
- Inhaltliche bedeutungsvolle Commits
- Repository schlank halten (keine großen Dateien)

Git + RStudio Ressourcen

- Small Intro
(<https://r-bio.github.io/intro-git-rstudio/>)
- Happy Git with R (<https://happygitwithr.com/>)
- R Packages and Git (<https://r-pkgs.org/git.html>)

Allgemeine Git Ressourcen

- Git Book (<http://git-scm.com/book/en/v2>)

Vielen Dank fuer Eure Aufmerksamkeit!

Vielen Dank fuer Eure Aufmerksamkeit!

Fragen? Anmerkungen?