

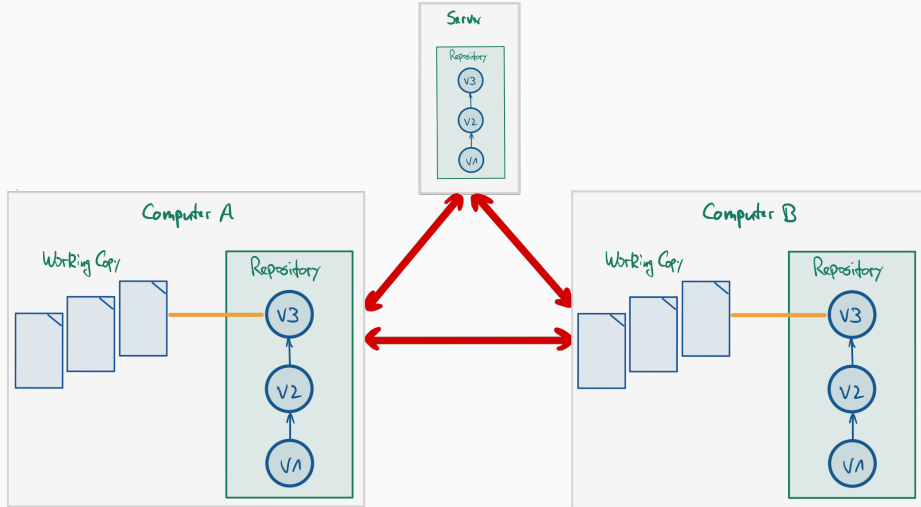
# Zusammenarbeit: Git-Workflows und Merge-/Pull-Requests

---

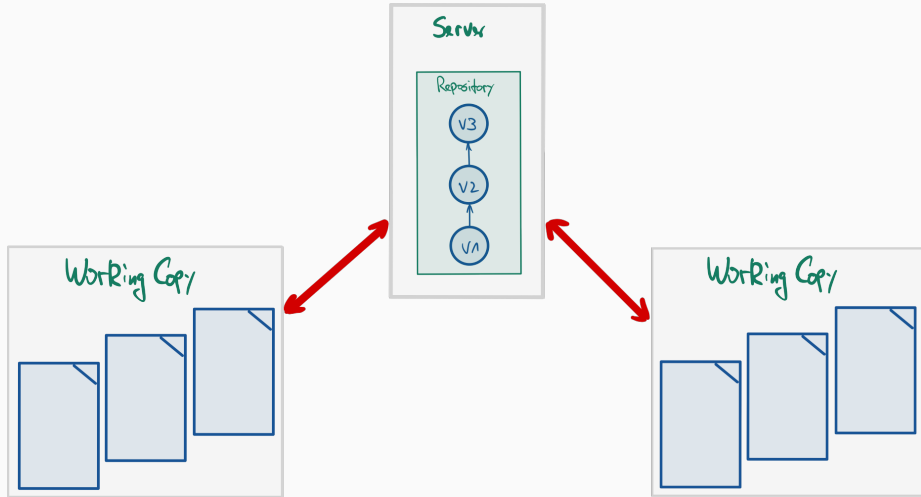
Carsten Gips (HSBI)

Unless otherwise noted, this work is licensed under CC BY-SA 4.0.

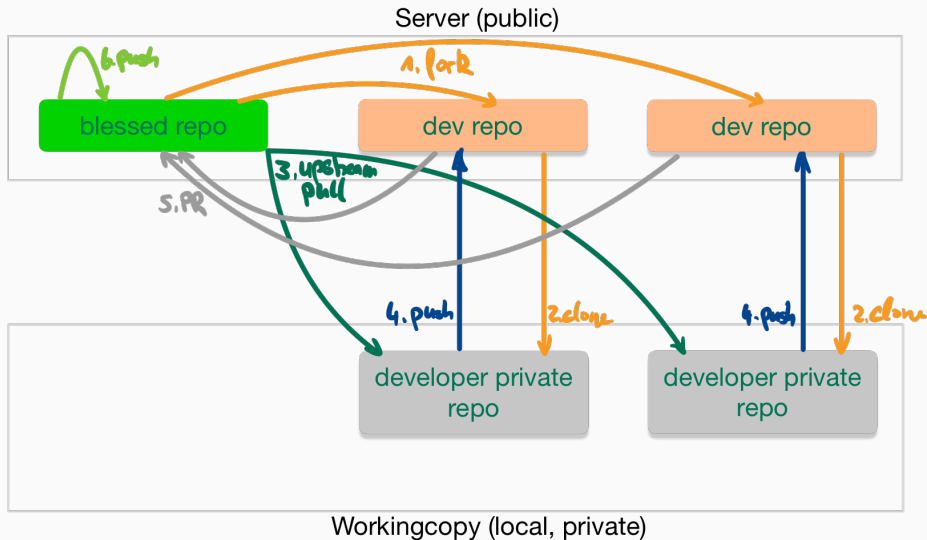
# Nutzung von Git in Projekten: Verteiltes Git (und Workflows)



# Zusammenarbeit: Zentraler Workflow mit Git (analog zu SVN)



# Zusammenarbeit: Einfacher verteilter Workflow mit Git



## Feature-Branches aktualisieren: Mergen mit *master* vs. Rebase auf *master*

```
$ git checkout master      # Workingcopy auf master
$ git pull upstream master # Aktualisiere lokalen master mit master aus Vorgabe-Repo
$ git checkout feature     # Workingcopy auf feature
$ git merge master         # Aktualisiere feature: Merge master in feature
$ git push origin feature  # Push aktuellen feature ins Team-Repo
```

Besser einen Pull/Rebase für den Feature-Branch machen!

# Kommunikation: Merge- bzw. Pull-Requests

The screenshot shows a GitHub Pull Request (PR) titled "Vorlesung: Git Remotes #261". The PR is in "Draft" status and is intended to merge 11 commits from the "remotes" branch into the "master" branch. The interface includes a file tree on the left, a list of changed files in the center, and a detailed view of the selected file, "markdown/git/remotes.md", on the right.

**PR Details:**

- Title:** Vorlesung: Git Remotes #261
- Status:** Draft
- Commits:** 11
- Files changed:** 3
- Changes:** +354 -25

**File Tree:**

- markdown/git
  - remotes.images
    - distributed.png
    - screenshot\_branches.png
    - remotes.md

**Changed Files:**

- BIN +176 KB markdown/git/remotes.images/distributed.png
- BIN +5.43 MB markdown/git/remotes.images/screenshot\_branches.png
- 379 B markdown/git/remotes.md

**File Content: markdown/git/remotes.md**

```
@@ -6,14 +6,40 @@ author: "Carsten Gips (FH Bielefeld)"
6 6 weight: 5
7 7 readings:
8 8 - key: "Chacon2014"
9 - comment: "Kapitel 2 und 3"
9 + comment: "Kapitel 3"
10 + - key: "AtlassianGit"
11 + - key: "GitCheatSheet"
10 12 tldr: |
11 - hier kommt eine tolle inline-zusammenfassung!
12 - Formatierung _könnte_ auch **gehen**?
13 + Eine der Stärken von Git ist das Arbeiten mit verteilten Repositories. Zu jeder Workingcopy gehört
14 + eine Kopie des Repositories, wodurch jederzeit alle Informationen einsehbar sind und auch offline
15 + gearbeitet werden kann. Allerdings muss man für die Zusammenarbeit mit anderen Entwicklern die lokalen
16 + Repos mit den "entfernten" Repos (auf dem Server oder anderen Entwicklungsrechnern) synchronisieren.
17 +
18 + Beim Klonen eines Repositories mit 'git clone <url>' wird das fremde Repo mit dem Namen 'origin'
19 + im lokalen Repo festgehalten. Dieser Name wird auch als Präfix für die Branches in diesem Repo genutzt,
20 + d.h. die Branches im Remote-Repo tauchen als 'origin/<branch>' im lokalen Repo auf. Diese Remote-Branches
21 + kann man nicht direkt bearbeiten, sondern man muss diese Remote-Branches in einem lokalen Branch auschecken
22 + und dann darin weiterarbeiten. Es können beliebig viele weitere Remotes dem eigenen Repository hinzugefügt
23 + werden.
24 +
```

- Git-Workflows für die Zusammenarbeit:
  - einfacher zentraler Ansatz für kleine Arbeitsgruppen vs.
  - einfacher verteilter Ansatz mit einem “blessed” Repo
- Aktualisieren Ihres Clones und Ihres Forks mit Änderungen aus dem “blessed” Repo
- Unterschied zwischen einem Pull/Merge und einem Pull/Rebase
- Erstellen von Beiträgen zu einem Projekt über Merge-Requests
  - Welche Commits werden Bestandteil eines Merge-Requests (und warum)
  - Diskussion über den Code in Merge-Requests



Unless otherwise noted, this work is licensed under CC BY-SA 4.0.