

ScaDS.AI
DRESDEN LEIPZIG

CENTER FOR SCALABLE DATA ANALYTICS AND
ARTIFICIAL INTELLIGENCE



NFDI4
BIOIMAGE

NATIONAL RESEARCH DATA MANAGEMENT INFRASTRUCTURE
FOR MICROSCOPY AND BIOIMAGE ANALYSIS



GLOBAL BIOIMAGE
ANALYST'S SOCIETY

Kollaboratives Arbeiten und Versionskontrolle mit Git

Robert Haase

ScaDS.AI
DRESDEN LEIPZIG

Robert Haase
@haesleinhuepf
Collaborative work / git
DataWeek Leipzig
April 15th 2024

These slides can be reused under the terms of the
[CC-BY 4.0](#) license unless mentioned otherwise.
<https://zenodo.org/doi/10.5281/zenodo.10966173>



GEFÖRDERT VOM



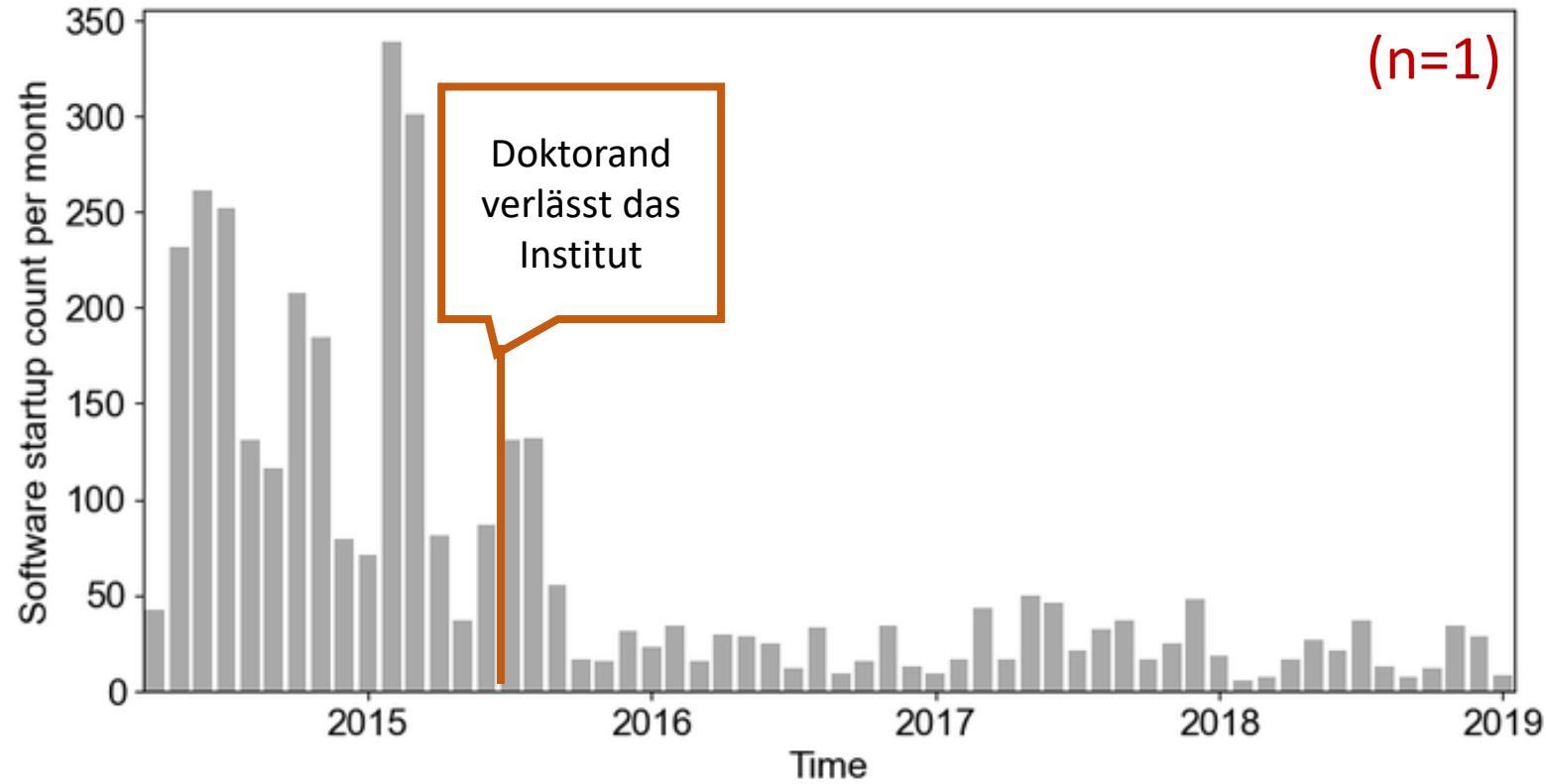
Bundesministerium
für Bildung
und Forschung



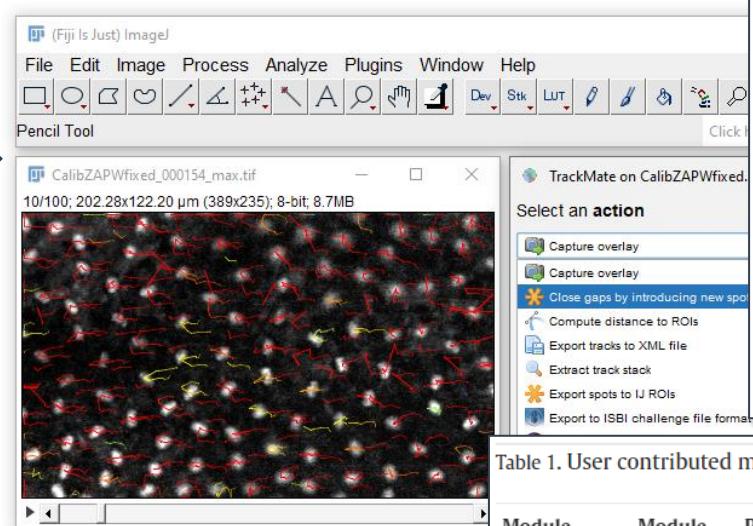
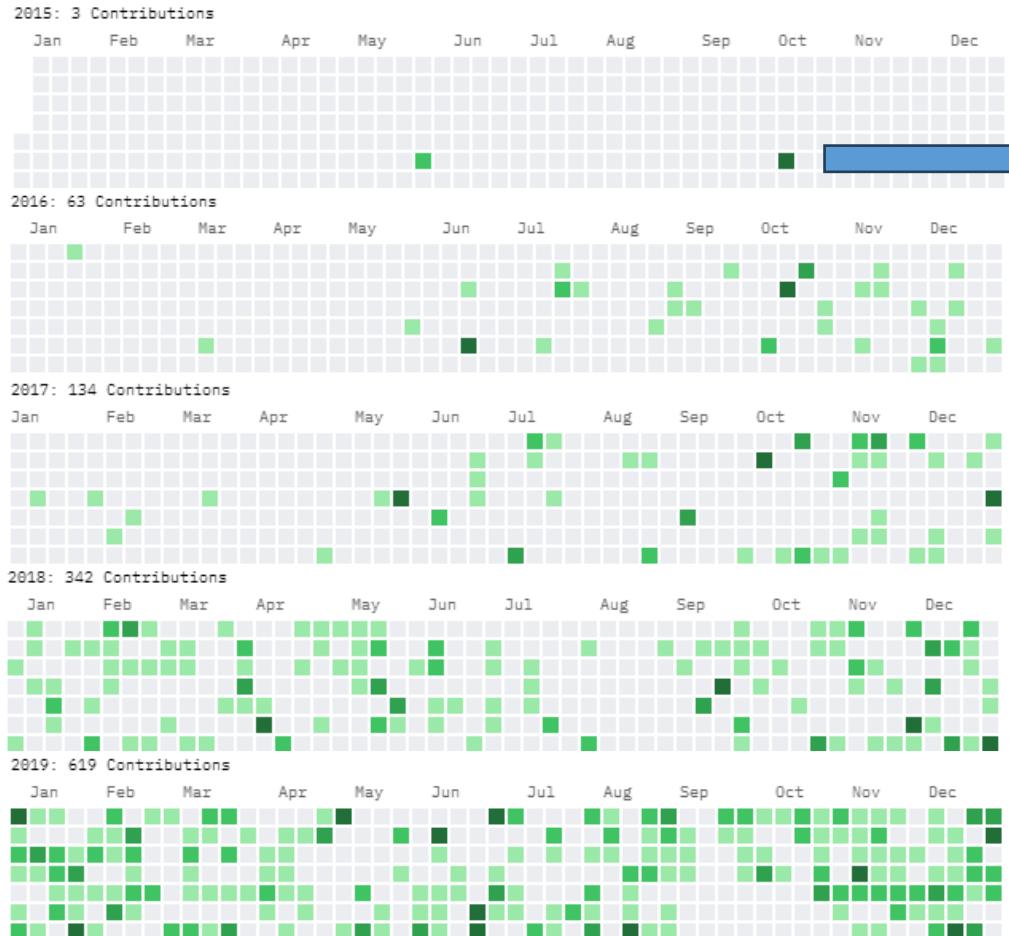
Diese Maßnahme wird gefördert durch die Bundesregierung
aufgrund eines Beschlusses des Deutschen Bundestages.
Diese Maßnahme wird mitfinanziert durch Steuermittel auf
der Grundlage des von den Abgeordneten des Sächsischen
Landtags beschlossenen Haushaltes.

Nachhaltigkeit von Softwareprojekten

- Was passiert wenn der Doktorand/Mitarbeiter das Institut verlässt?



Mitarbeit an Open Source Software



Methods
Volume 115, 15 February 2017, Pages 80-90

TrackMate: An open and extensible platform for single-particle tracking

Jean-Yves Tinevez^a, Nick Perry^a, Johannes Schindelin^b, Genevieve M. Hoopes^c, Gregory D. Reynolds^c, Emmanuel Laplantine^d, Sebastian Y. Bednarek^c, Spencer L. Shute^a, Kevin W. Eliceiri^b

Show more ▾

+ Add to Mendeley Share Cite

<https://doi.org/10.1016/j.jymeth.2016.09.016>

Get rights and content ▾

open access

Table 1. User contributed modules of TrackMate v3.4.0.

Module name	Module type	Purpose	Author	Location
Linear tracker	Particle-linking	Linking transported particles by extrapolating their velocity	Ronny Sczech	https://github.com/chicorrony/RonnyTrackMate
Batch mode	Plugin	Runs TrackMate in batch reading		
Close gaps	Generic action	Close gaps in tracks by creating spots in missing frame by linear interpolation of their coordinates	Robert Haase	Integrated into TrackMate v3.4.0

Note: The colour-code is not consistent between the years

Quiz

- Wie oft nutzt Ihr git oder Github?

Nie



Hin und wieder

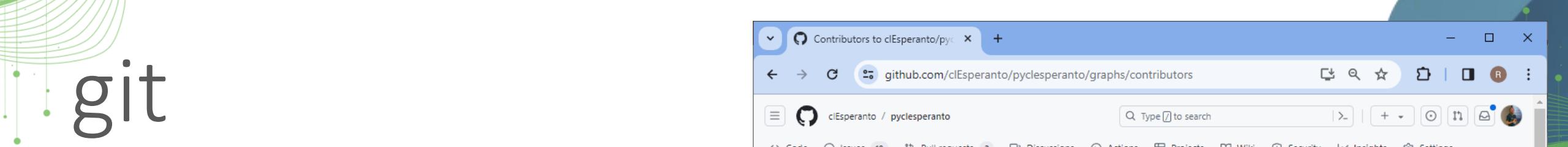


1/Woche



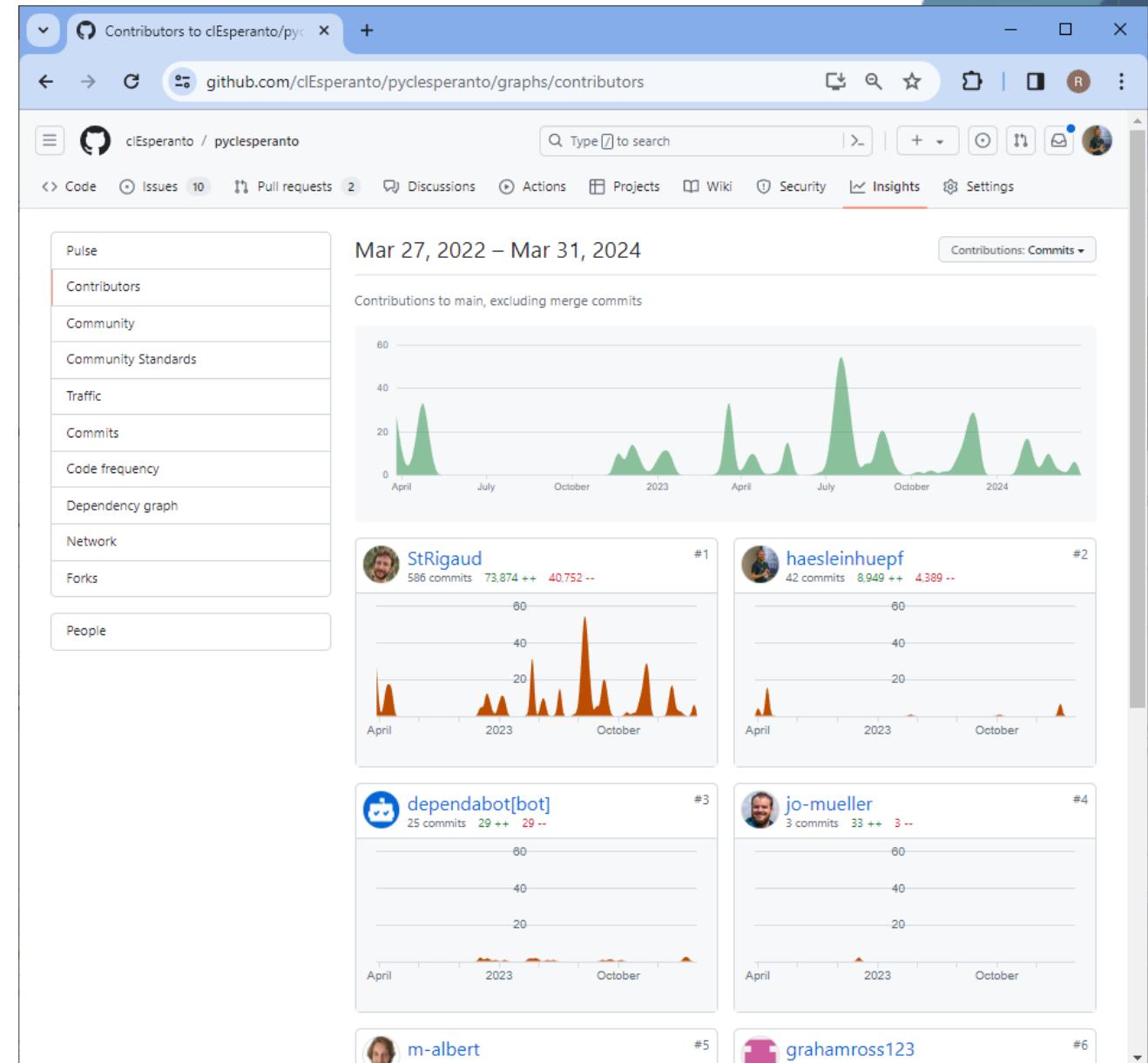
Täglich





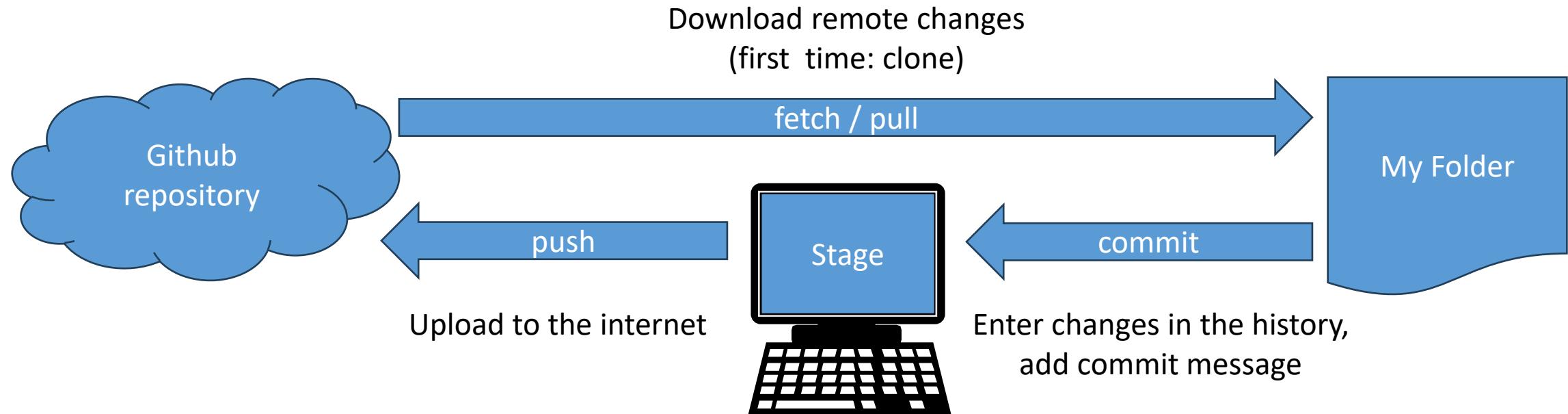
git

- Versionskontrolle ist ein wichtiges Werkzeug fuer Softwareentwickler und Datenwissenschaftler
- Verteiltes Dateisystem mit Logbuchfunktion
- Kontrolle was Teil des Ganzen wird – und was nicht



git

- Git verändert das Arbeiten: Es überführt Änderungen in einen aktiveren Prozess
(Wodurch wir mehr darüber nachdenken)



git

- Wer hat diesen Code geschrieben?
- Wann?
- Wieso?

The screenshot shows two GitHub browser windows side-by-side. The left window displays the commit history for the repository 'haesleinhuepf/example_image_analysis_script'. It lists several commits, with the commit 'bugfix: threshold_otsu' highlighted by a large orange arrow pointing from the list to its detailed diff view in the right window. The right window shows the detailed diff for the file 'my_library.py'. The diff highlights changes made to the 'segment_image' function, specifically the addition of the 'threshold_otsu' step. The commit message 'bugfix: threshold_otsu' is visible at the top of the diff view.

```
diff --git a/my_library.py b/my_library.py
index 3e3f3d..65c074 100644
--- a/my_library.py
+++ b/my_library.py
@@ -6,7 +6,8 @@ def segment_image(image):
 6   6     blurred = gaussian(image, sigma=2)
 7   7
 8   8     # binarize the image
 9 - 9     binary = threshold_otsu(blurred)
 9 + 9     threshold = threshold_otsu(blurred)
10 + 10    binary = blurred > threshold
11   11
12   12     # label connected components
13   13     result = label(binary)
```

git

- History
- Track recent changes

The screenshot shows two browser windows side-by-side. The left window displays the GitHub repository 'ScaDS/BIDS-lecture-2024' with the 'Commits' page. It lists three commits made by 'haesleinhuepf' yesterday:

- add backwards compatibility exercise
- move pull-request exercise from week 2 to week 3
- fix issue with ../../ folder locations

A blue arrow points from the third commit to the right window, which shows a detailed view of the commit 'fix issue with ../../ folder locations'. The commit message is 'fix issue with ../../ folder location'. The diff view shows the changes made to the file '02a_remote_files/exploring_bioimage_archive.ipynb'. The changes are as follows:

```
@@ -233,9 +233,9 @@
233 233     "    if not os.path.exists(folder_path):\n",
234 234         "        os.makedirs(folder_path)\n",
235 235     "\n",
236 -     "base_folder = f'../../data/{accession}'\n",
237 -     "raw_folder = f'../../data/{accession}/images'\n",
238 -     "groundtruth_folder = f'../../data/{accession}/groundtruth'\n",
236 +     "base_folder = f'data/{accession}'\n",
237 +     "raw_folder = f'data/{accession}/images'\n",
238 +     "groundtruth_folder = f'data/{accession}/groundtruth'\n",
239 239     "\n",
240 240     "ensure_folder_exists(base_folder)\n"
```

Quiz

- Was ist kein FAIR-Prinzip?

Findable



Accessible



Interoperable



Reproducible



Die FAIR-Prinzipien

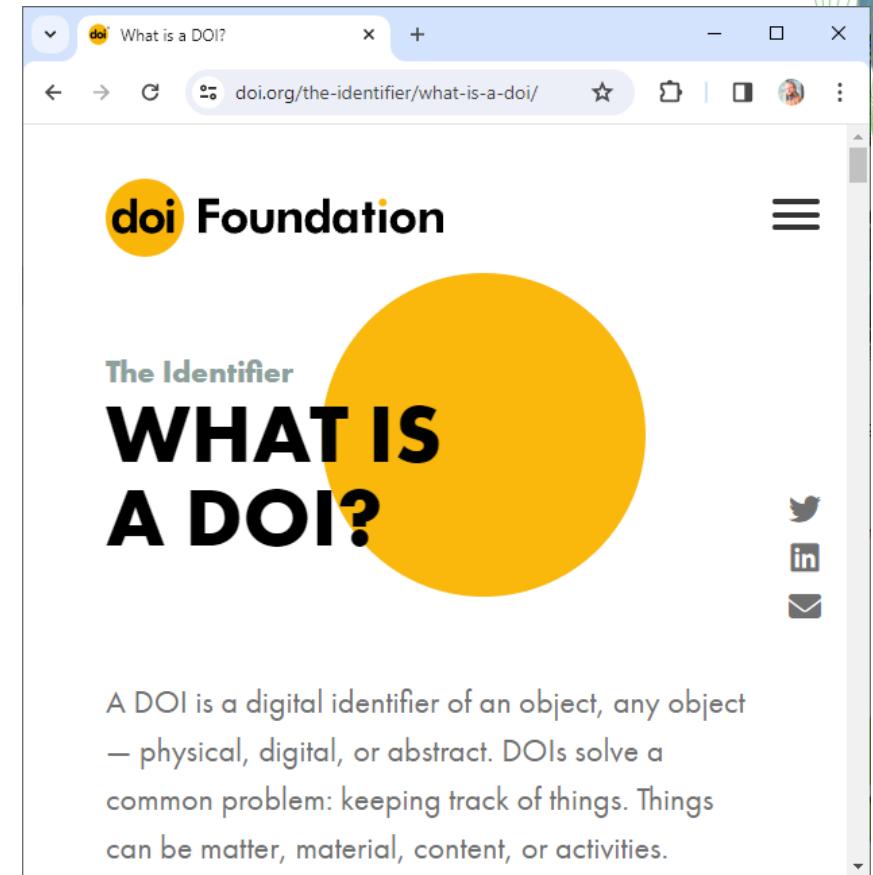
- Findable
- Accessible
- Interoperable
- Reusable



Die FAIR-Prinzipien

Findable / Findbarkeit

- F1. (Meta)daten sind verbunden mit einem global eindeutigem Identifier (DOI)
- F2. Daten sind mit “reichen Metadaten” beschrieben
- F3. Metadaten beinhalten die DOI, die sie beschreiben
- F4. (Meta)daten sind in einer durchsuchbaren Resource registriert



Github: Eine durchsuchbare Resource

The screenshot shows two browser windows. The left window is the GitHub homepage with a large banner reading 'Let's build from here' and the tagline 'The world's leading AI-powered developer platform.' A blue callout box points to the search bar with the text 'Suche: "Prompt Engineering Tutorial"'. The right window shows the search results for 'prompt%20engineering%20tutorial&type=repositories'. It includes a sidebar for filtering by Code, Repositories (selected), Issues, Pull requests, Discussions, Users, and More. Under Languages, Jupyter Notebook, Python, and Shell are listed. Three repository cards are visible: 1. 'Step-by-Step tutorial that teaches you how to use Azure Prompt Flow to streamline the workflow for prompt engineering and expedite the pr...' (Jupyter Notebook, 2 stars, updated Feb 26). 2. 'ScaDS/prompt-engineering-tutorial' (Jupyter Notebook, 3 stars, updated Dec 15, 2023). 3. 'skandavivek/DSPy-blog' (A tutorial on DSPy and whether automated prompt engineering lives up to the hype).

Suche: "Prompt Engineering Tutorial"

Filter by

- Code
- Repositories** 30
- Issues 2k
- Pull requests 368
- Discussions 81
- Users 0
- More

Languages

- Jupyter Notebook
- Python
- Shell

Step-by-Step tutorial that teaches you how to use Azure Prompt Flow to streamline the workflow for prompt engineering and expedite the pr...
Jupyter Notebook · ⭐ 2 · Updated on Feb 26

ScaDS/prompt-engineering-tutorial
This Jupyter Book contains notebooks demonstrating OpenAI's API for using chatGPT.
Jupyter Notebook · ⭐ 3 · Updated on Dec 15, 2023

skandavivek/DSPy-blog
A tutorial on DSPy and whether automated prompt engineering lives up to the hype
Star

Platz 5

Nutzen: Findbarkeit -> Sichtbarkeit

- YouTube
- Github

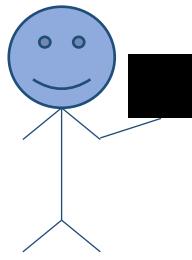
Offenes & FAIRes Teilen
ist ein PR-Instrument

... fuehrt zu

- mehr Softwarenutzer:innen
- neuenw Kollaborationen

Offenheitslevel

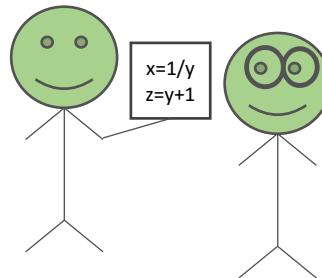
Closed source



- Open to collaborations
- “Black box”
- Compiled code (e.g. C/C++)
- Good for protecting intellectual properties (\$\$\$)

Hardware device drivers

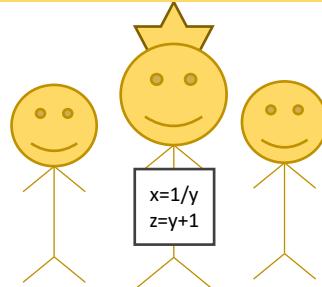
Open source



- Code available to read
- Not necessarily executable code
- No maintenance / support efforts

Custom image analysis scripts

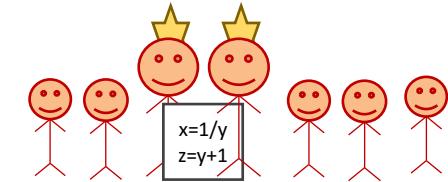
Benevolent dictatorship



- Open to contributions
- Single maintainer, often overwhelmed
- Efficient decision making
- Bus factor ≈1

TrackMate, SNT, MorpholibJ, CLIJ

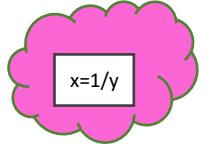
Community driven



- Open to contributions
- Partially democratic
- Board of maintainers (core developers)
- Long-winded decision making

scikit-image, scipy, OpenCL

Openly extensible



- Openly extensible; without maintainers involved
- Partially community driven

ImageJ, Python, numpy

Die FAIR-Prinzipien

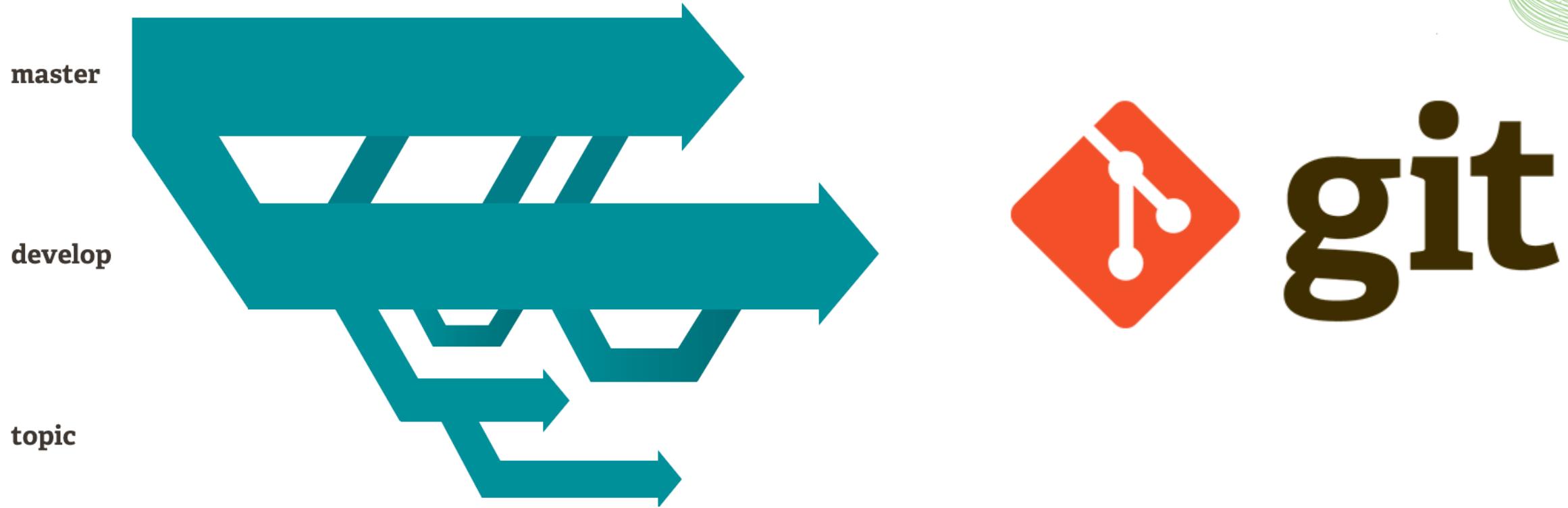
- Accessible / Verfügbar / Zugreifbar
- A1. (Meta)daten können über ein Standardisiertes Protokoll empfangen werden
 - A1.1 Das Protokoll ist offen, frei und universell implementierbar
 - A1.2 Das Protokoll erlaubt Authentifizierung und Authorisierung, wenn erforderlich
- A2. Metadaten sind verfügbar, auch wenn die Daten selbst nicht mehr verfügbar sind

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



git: standardized communication protocol

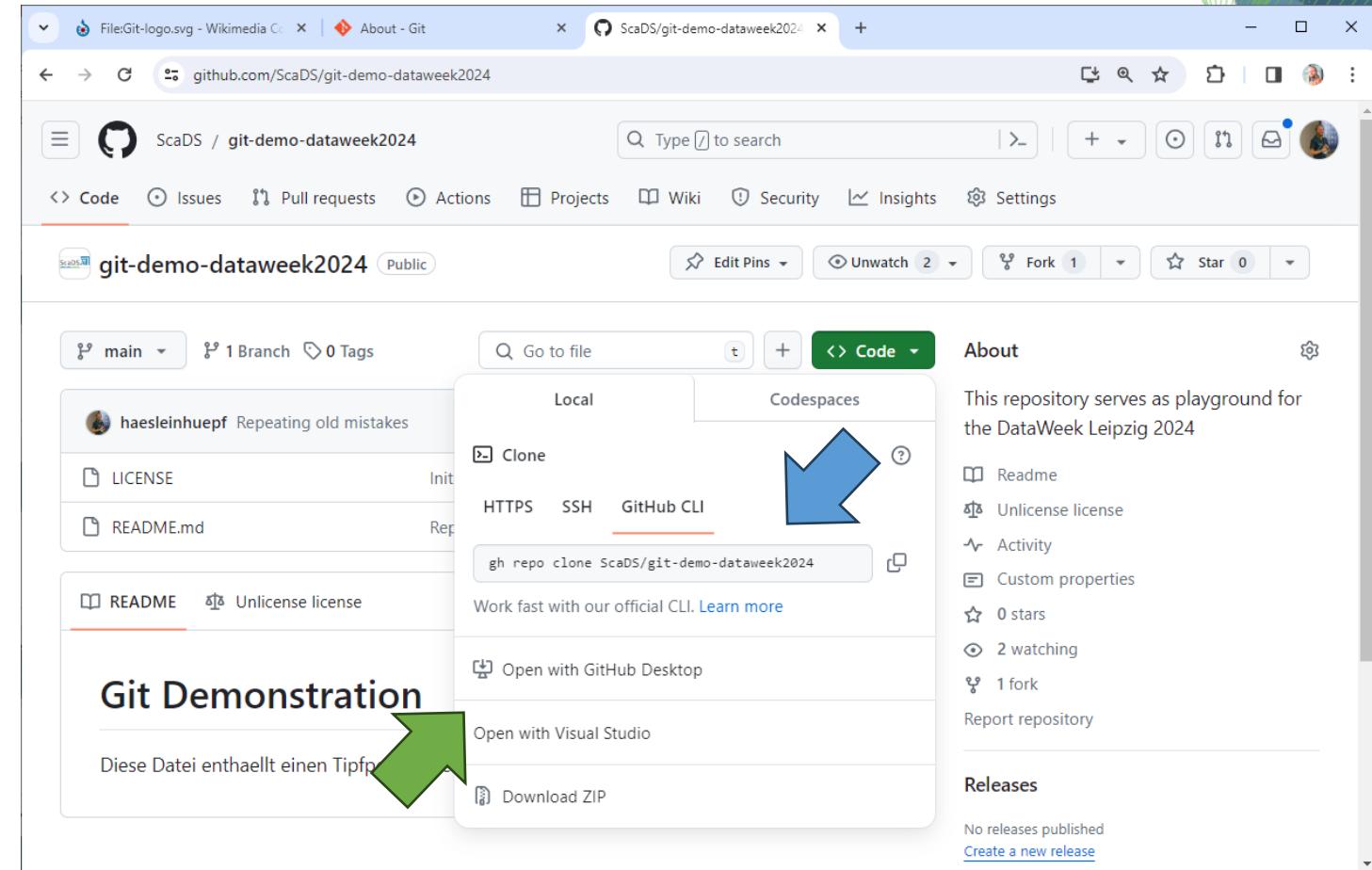
- Git nutzt “Branches” um paralleles Arbeiten zu erlauben



git

Accessibility

- Möglichkeit fuer Menschen und Computer, Daten herunterzuladen



Restricted Access

- Das A in FAIR steht nicht zwingend fuer Open-Access

The image shows two screenshots of a GitHub repository. The left screenshot shows the main repository page for 'git-demo-dataweek-private-2024'. The word 'Private' is highlighted with a red box. The right screenshot shows the 'Manage access' page for the same repository. The 'Collaborators and teams' tab is selected. It displays that the repository is a 'PRIVATE REPOSITORY' where 'Only those with access to this repository can view it.' Below this, there is a section titled 'Manage access' with a message: 'You haven't added any teams or people yet' and buttons for 'Add people' and 'Add teams'.

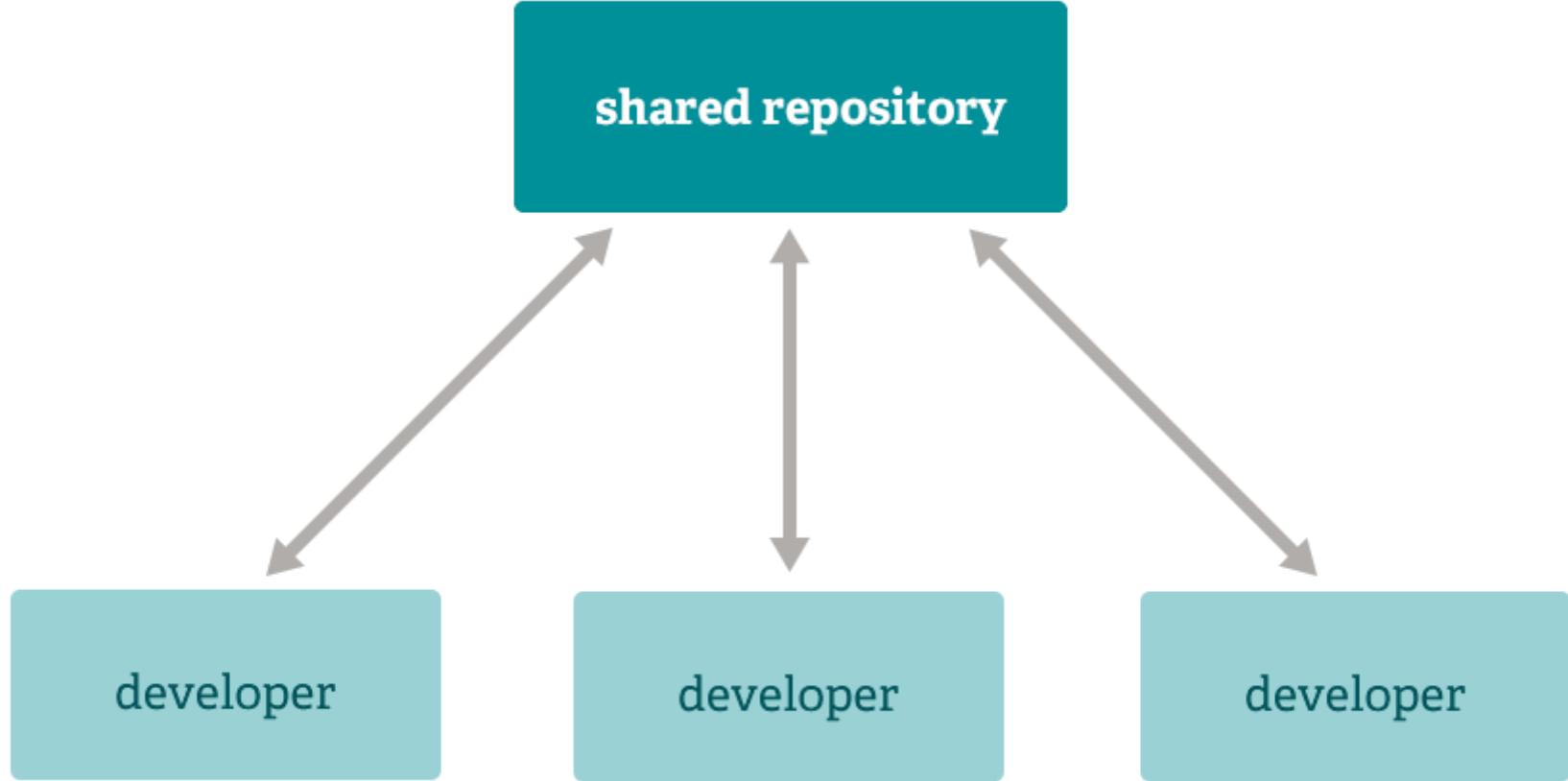
Die FAIR-Prinzipien

- Interoperable
 - I1. (Meta)daten sind formalisiert in einer zugaenglichen, gemeinsamen, breit angewandten Sprache, geeignet fuer Wissensrepraesentation
 - I2. (Meta)daten nutzen ein Vokabular, dass ebenfalls den FAIR-Prinzipien unterliegt
 - I3. (Meta)daten referenzieren andere qualifizierte (Meta)daten



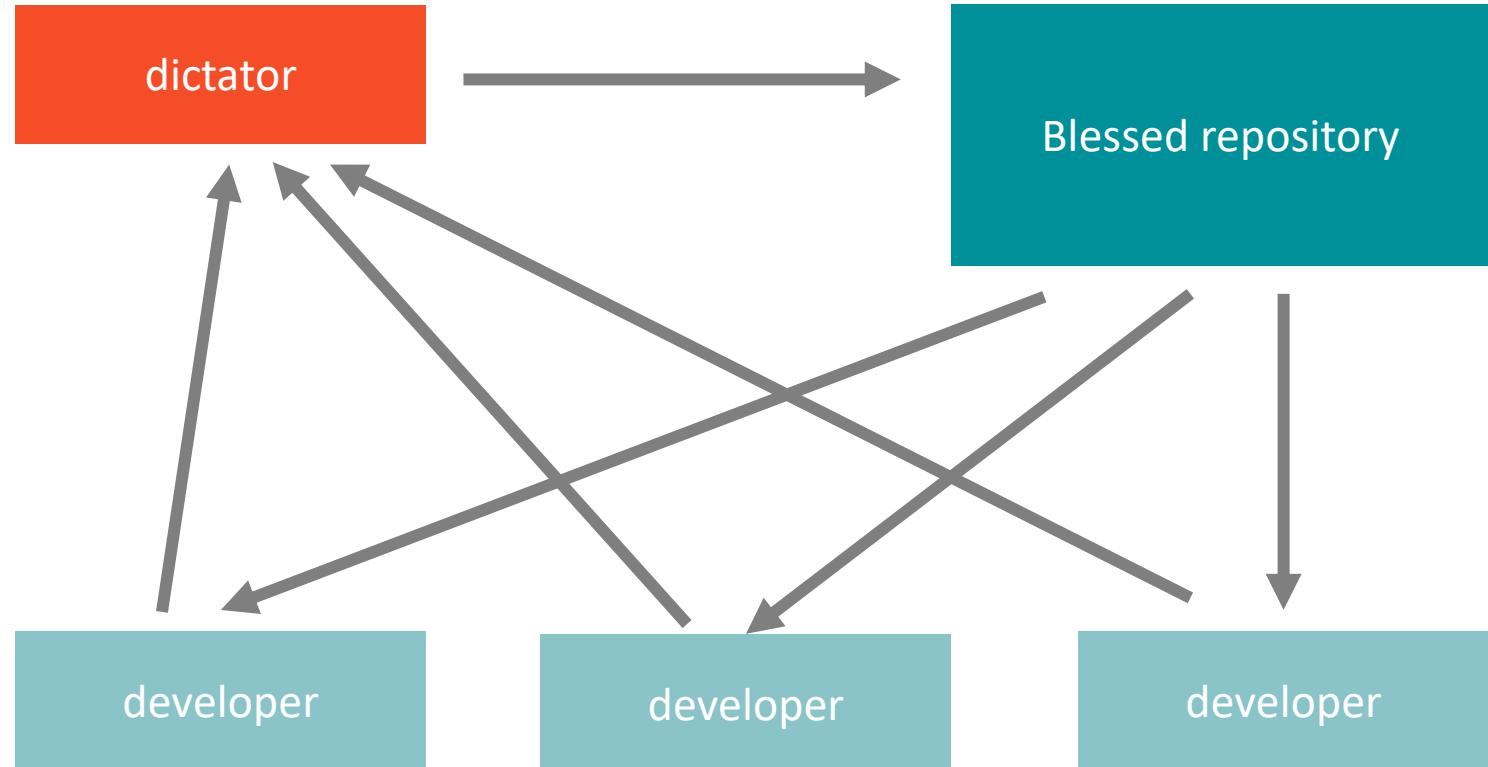
git: Workflows

- Subversion-style



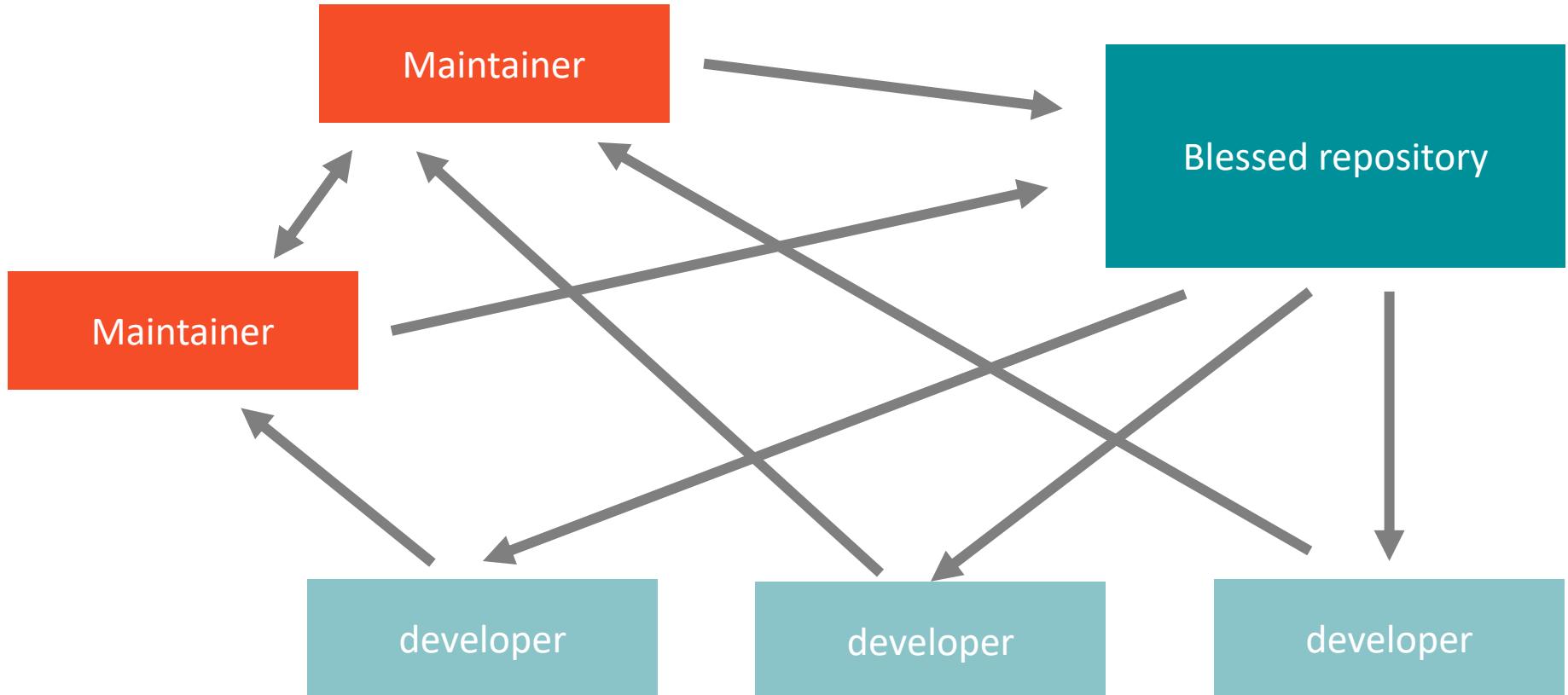
git: Workflows

- Benevolent dictatorship workflow



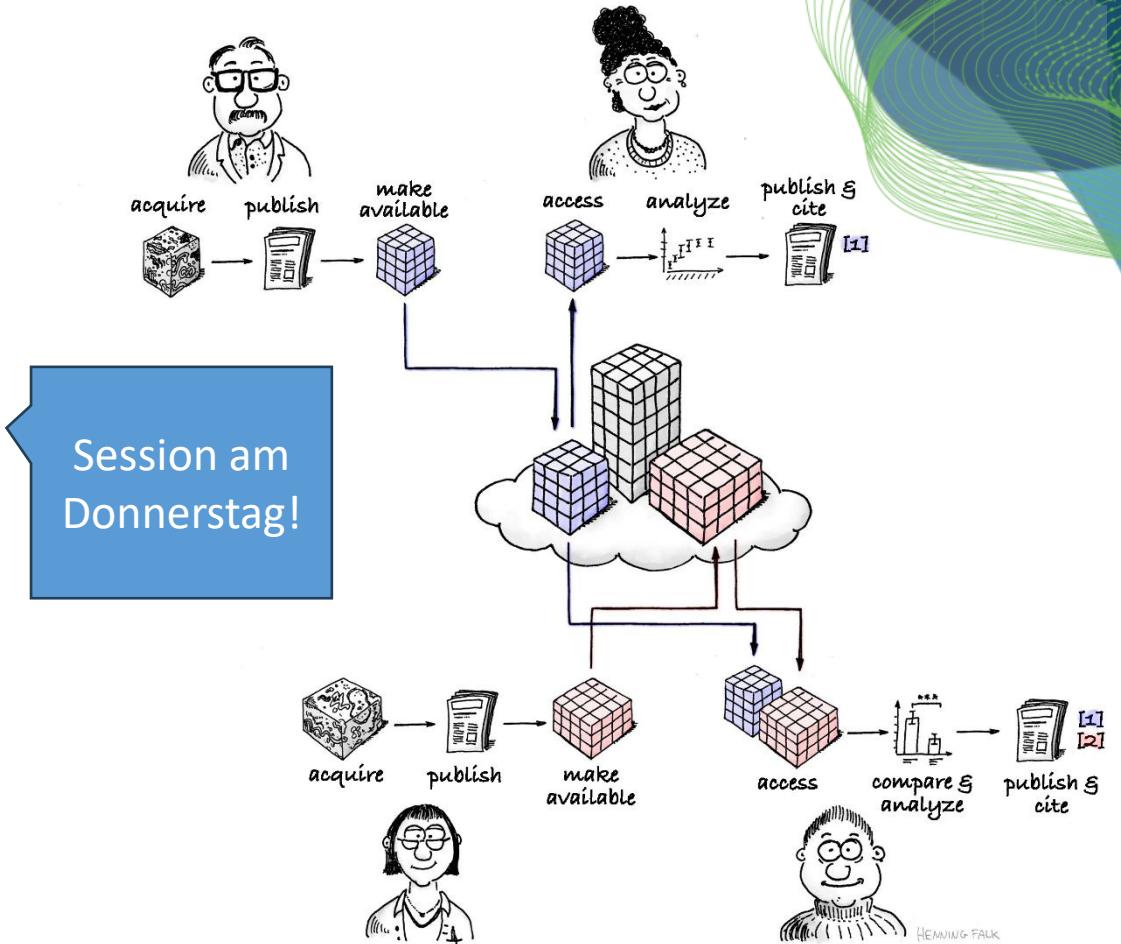
git: Workflows

- Joint maintainers (may implement some form of democracy)



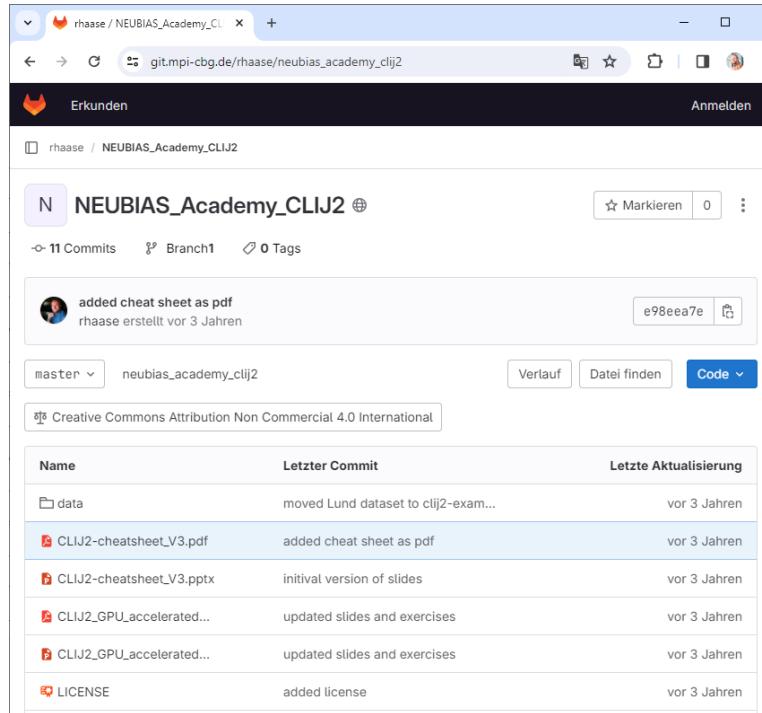
The FAIR-principles

- Reusable / Wiederverwendbar
 - R1. (Meta)daten sind reich an vielfältigen, akkurate und relevanten Attributen
 - R1.1. (Meta)daten werden mit einer klaren und verfügbaren Nutzungslizenz versehen
 - R1.2. (Meta)daten sind stets mit der detaillierten Herkunft erfasst
 - R1.3. (Meta)daten folgen gemeinschaftlich definierten Standards



Teilen, aber wo?

- Gitlab / Github:
Ideal fuer Kollaboratives Arbeiten,
bei Open Source Projekten



The screenshot shows a GitHub repository page for 'stackview'. It displays a list of 14 branches and 32 tags. One file is highlighted: 'fixed documentation' by 'haesleinhuepf' from last week. The repository contains files like 'docs', 'stackview', '.gitignore', 'LICENSE', 'README.md', and 'setup.py'. A table at the bottom lists files with their last commit details.

Name	Letzter Commit	Letzte Aktualisierung
docs	fixed documentation	last week
stackview	update contact email, bump version	last week
.gitignore	ignore deployment script	5 months ago
LICENSE	Initial commit	3 years ago
README.md	fixed documentation	last week
setup.py	update contact email, bump version	last week

- Zenodo:
archiving / citing

The screenshot shows a Zenodo record page for 'haesleinhuepf/stackview: 0.7.6'. It provides a summary of the release, including the date (March 30, 2024), version (0.7.6), and authors (Robert Haase¹, Jordão Bragantini², Oren Amsalem). It also includes sections for 'What's Changed' (listing changes like 'add imshow (moved over, and updated from pyclesperanto_prototype) by @haesleinhuepf in https://github.com/haesleinhuepf/stackview/pull/37' and 'fix deprecation warning in switch by @haesleinhuepf in https://github.com/haesleinhuepf/stackview/pull/38') and a 'Full Changelog' link.

Zenodo

- öffentlich geförderte Infrastruktur @ CERN / Schweiz

The image displays two side-by-side screenshots of the Zenodo website. The left screenshot shows the homepage with a blue header and a 'Featured communities' section featuring the European Climate and Modelling Forum, which is highlighted with a large, colorful circular graphic. The right screenshot shows the main navigation menu with links for 'About', 'Blog', 'Help', 'Developers', 'Contribute', and various API and documentation links. Both screenshots show the URL zenodo.org in the browser address bar.

Quiz

- Wo ist Open-Source Software am besten sichtbar?

Git server an der
Universität



Zenodo.org



Github.com

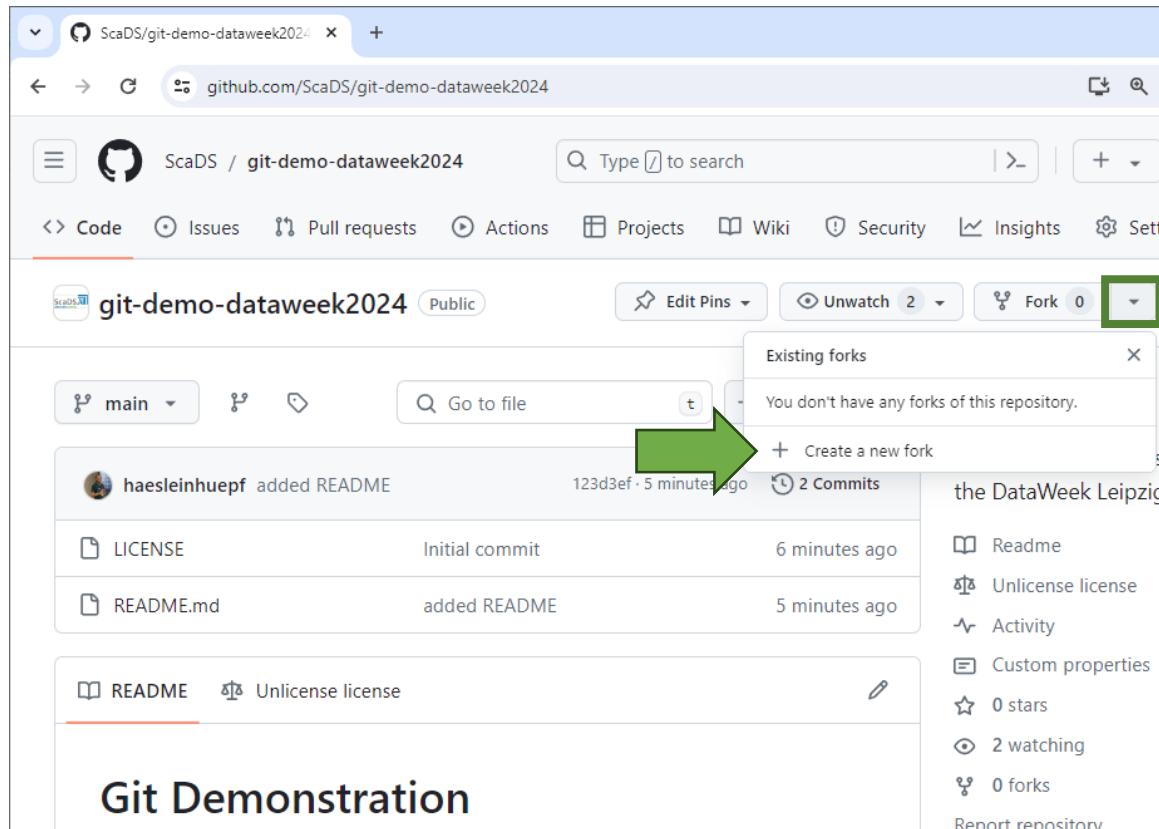


Firmen- / Instituts-
Webseite

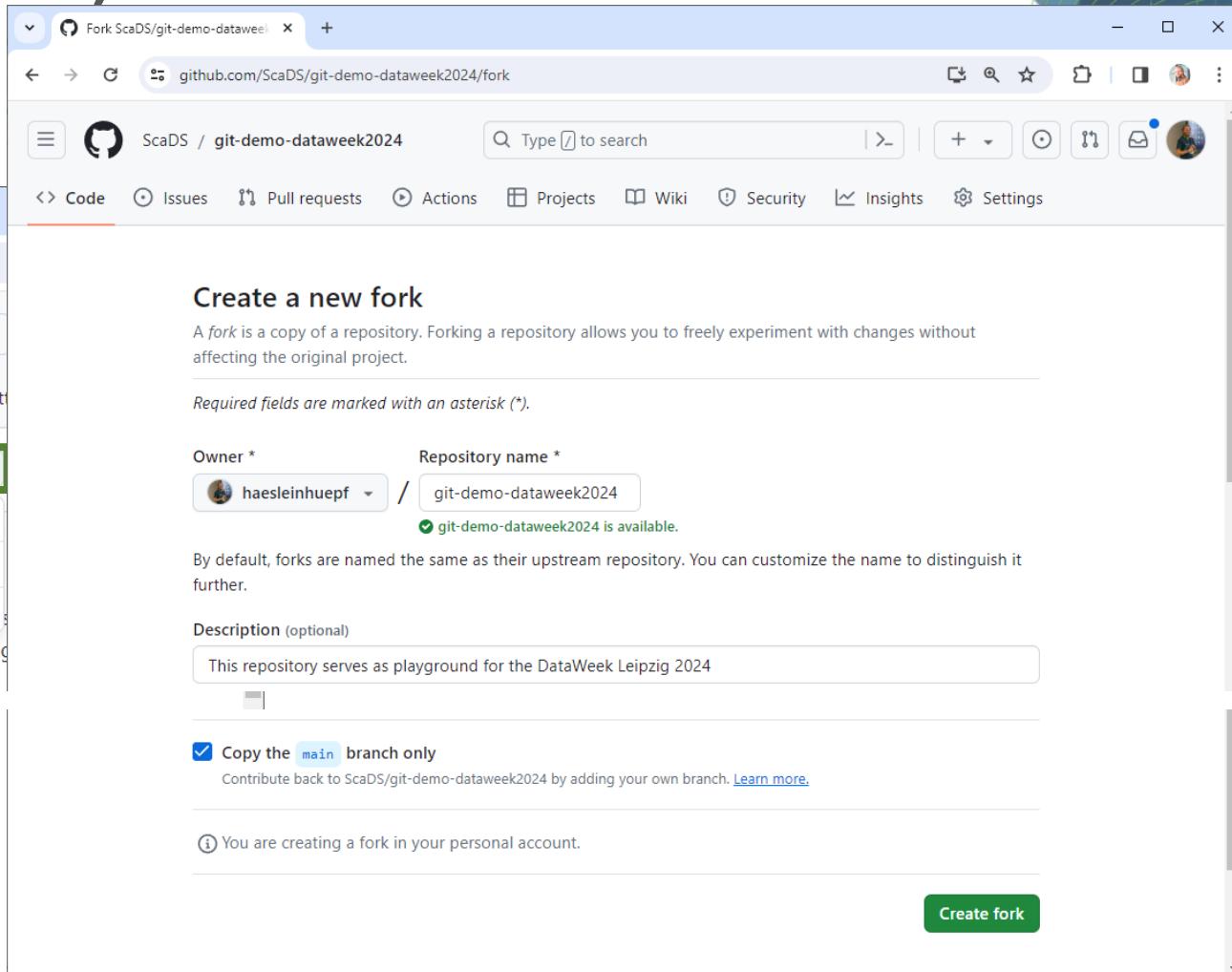


Quick start: git(hub)

- *Forken einer repository*



A screenshot of a GitHub repository page for "git-demo-dataweek2024". The page shows a commit from "haesleinhuepf" adding a README file. A green arrow points to the "Fork" button in the top right corner of the repository card.



A screenshot of the "Create a new fork" dialog box on GitHub. It shows the "Owner" dropdown set to "haesleinhuepf" and the "Repository name" field set to "git-demo-dataweek2024". A note says "git-demo-dataweek2024 is available". The "Description (optional)" field contains "This repository serves as playground for the DataWeek Leipzig 2024". A checked checkbox says "Copy the main branch only". A note at the bottom says "You are creating a fork in your personal account". A green "Create fork" button is at the bottom right.

Quick start: git(hub)

- In einen Fork haben wir Schreibrechte...

The image displays two side-by-side screenshots of a GitHub repository page. Both screenshots show the repository `haesleinhuepf/git-demo-dataweek2024`.

Left Screenshot (Initial State):

- The repository is public.
- It was forked from `ScaDS/git-demo-dataweek2024`.
- The main branch is up-to-date with the upstream branch.
- Recent commits include:
 - `haesleinhuepf added README` (123d3ef · 8 minutes ago)
 - `LICENSE` (Initial commit · 9 minutes ago)
 - `README.md` (added README · 8 minutes ago)

Right Screenshot (After Commit):

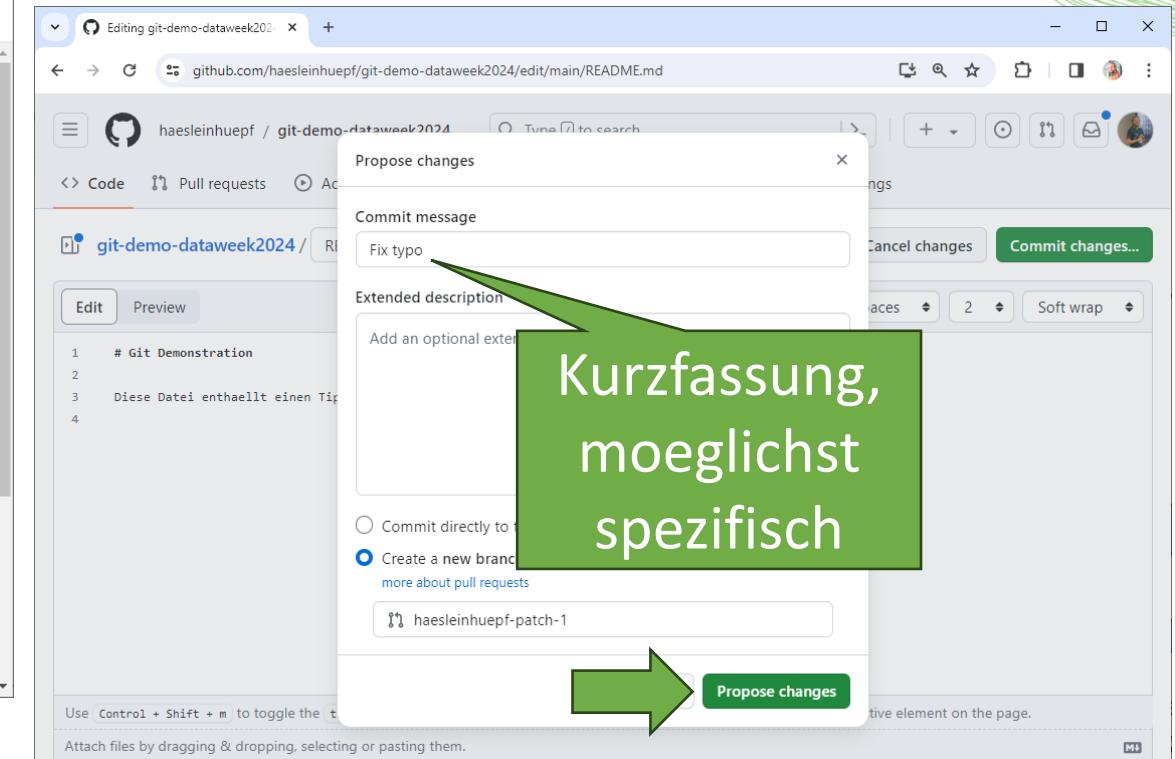
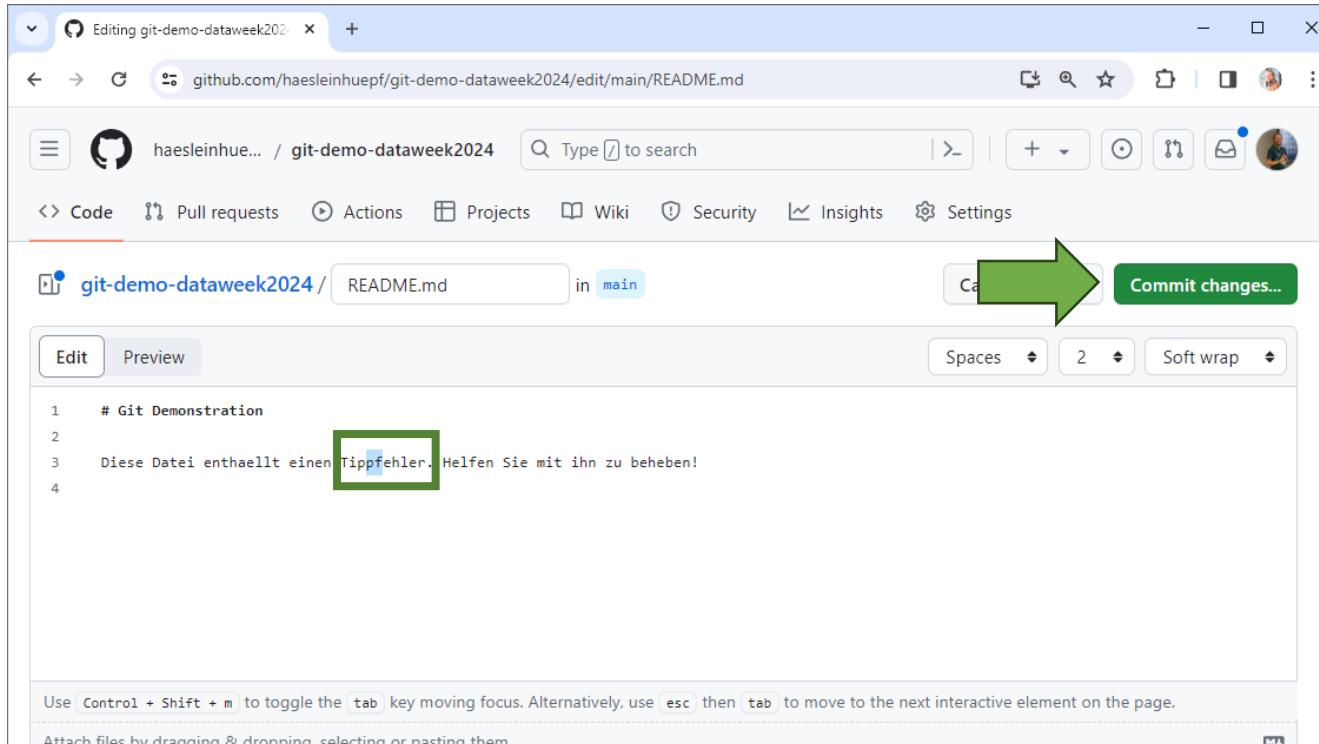
- The repository is public.
- It was forked from `ScaDS/git-demo-dataweek2024`.
- The main branch is up-to-date with the upstream branch.
- Recent commits include:
 - `haesleinhuepf added README` (123d3ef · 8 minutes ago)
 - `LICENSE` (Initial commit · 9 minutes ago)
 - `README.md` (added README · 8 minutes ago)
- The README file content is shown:

```
Git Demonstration

Diese Datei enthaelt einen Tipfehler. Helfen Sie mit ihn zu beheben!
```
- The sidebar on the right shows repository statistics:
 - Unlicense license
 - Activity
 - 0 stars
 - 0 watching
 - 0 forks

Quick start: git(hub)

- In einen Fork haben wir Schreibrechte...



Quick start: git(hub)

- Pull-requests sind der Mechanismus zur Zusammenarbeit

The screenshot shows three browser tabs comparing branches on the GitHub repository `haesleinhuepf/git-demo-dataweek2024`:

- Left Tab:** Compares `main` vs `haesleinhuepf-patch-1`. It shows 1 commit and 1 file changed. A green arrow points from this tab to the dropdown menu in the middle tab.
- Middle Tab:** Compares `main` vs `ScaDS/git-demo-dataweek2024`. The dropdown menu is open, showing `base repository: haesleinhuepf/git-demo-dataweek2024` and `compare: haesleinhuepf-patch-1`. A green arrow points from the left tab to this dropdown.
- Right Tab:** Shows the pull request creation dialog. It includes fields for "Add a title" (Fix typo), "Add a description" (containing "Hi Robert, this just fixes a typo."), and "Create pull request". A large green speech bubble labeled "Freundliche Nachricht" (Friendly Message) points to the "Add a description" field.

Quick start: git(hub)

- Reviewer-Perspektive

The screenshot shows two browser windows side-by-side, both displaying a GitHub pull request page for a repository named "ScaDS / git-demo-dataweek2024".

Left Window (Reviewer Perspective):

- The title is "Fix typo #1".
- Status: Open.
- Details: haesleinhuepf wants to merge 1 commit into `ScaDS:main` from `haesleinhuepf:haesleinhuepf-patch-1`.
- Conversation: 0 messages.
- Commits: 1.
- Checks: 0.
- Files changed: 1.
- Content: A comment from user `haesleinhuepf` saying "Hi Robert, this just fixes a typo. Best, Robert".
- Commit details: `493ea42` (Verified).
- Instructions: "Add more commits by pushing to the `haesleinhuepf-patch-1` branch on `haesleinhuepf/git-demo-dataweek2024`".
- Review sidebar: Shows "No reviews" and "Still in progress? Convert to...".

Right Window (Code View):

- The title is "Fix typo by haesleinhuepf - Pull".
- Details: `github.com/ScaDS/git-demo-dataweek2024/pull/1/files`.
- Conversation: 0 messages.
- Commits: 1.
- Checks: 0.
- Files changed: 1.
- Content: A diff view of `README.md`.
 - Line 1: `1 1 @@ -1,3 +1,3 @@`
 - Line 2: `2 2 # Git Demonstration`
 - Line 3: `3 - Diese Datei enthaelt einen Tippfehler. Helfen Sie mit ihn zu beheben!`
 - Line 4: `3 + Diese Datei enthaelt einen Tippfehler. Helfen Sie mit ihn zu beheben!`
- Review buttons: "Review in codespace" and "Review changes".

A large green arrow points from the left window's sidebar towards the right window's code view.

Quick start: git(hub)

- Reviewer-Perspektive

The image displays three sequential screenshots of a GitHub pull request review process:

- Screenshot 1: Pull Request Opened**
The pull request is titled "Fix typo by haesleinhuepf · Pull #1". It shows a green "Open" button and a note: "Continuous integration has not been set up. GitHub Actions and several other apps can be used to automatically enforce style." Below this, a green checkmark indicates "This branch has no conflicts with the base branch. Merging can be performed automatically." A "Merge pull request" button is present.
- Screenshot 2: Merge Confirmation**
A modal window titled "Merge pull request #1 from haesleinhuepf/haesleinhuepf-pa..." shows the commit message "Fix typo". It includes "Confirm merge" and "Cancel" buttons. The GitHub Actions status bar at the bottom shows "Fix typo" and "Merge pull request #1 from haesleinhuepf/haesleinhuepf-patch-1".
- Screenshot 3: Pull Request Merged**
The pull request is now merged, indicated by a purple "Merged" button and the message "Fix typo #1 haesleinhuepf merged 1 commit into ScaDS:main from haesleinhuepf:haesleinhuepf-patch-1 now". A purple box highlights the message "Pull request successfully merged and closed". The GitHub Actions status bar at the bottom shows "Development" and "Notifications". A comment from "Robert" is visible in the "Add a comment" section.



CENTER FOR SCALABLE DATA ANALYTICS AND
ARTIFICIAL INTELLIGENCE



NFDI 4
BIOIMAGE

NATIONAL RESEARCH DATA MANAGEMENT INFRASTRUCTURE
FOR MICROSCOPY AND BIOIMAGE ANALYSIS



GLOBAL BIOIMAGE
ANALYST'S SOCIETY

Praktikum

Robert Haase

GEFÖRDERT VOM



Bundesministerium
für Bildung
und Forschung



Diese Maßnahme wird gefördert durch die Bundesregierung aufgrund eines Beschlusses des Deutschen Bundestages.
Diese Maßnahme wird mitfinanziert durch Steuermittel auf der Grundlage des von den Abgeordneten des Sächsischen Landtags beschlossenen Haushaltes.



Robert Haase
@haesleinhuepf
Collaborative work / git
DataWeek Leipzig
April 15th 2024

These slides can be reused under the terms of the
[CC-BY 4.0](#) license unless mentioned otherwise.

34



TECHNISCHE
UNIVERSITÄT
DRESDEN



UNIVERSITÄT
LEIPZIG

Praktikum

- Helfen Sie den Tippfehler zu beheben
 - Forken Sie die Repository

<https://github.com/ScaDS/git-demo-dataweek2024>

- Editieren Sie readme.md
- Senden Sie einen pull-request

Quiz

- Die Bitte Änderungen zu übernehmen ist ein...

Fork



Pull-request



Push-request



Commit





ScaDS.**AI**
DRESDEN LEIPZIG

CENTER FOR SCALABLE DATA ANALYTICS AND
ARTIFICIAL INTELLIGENCE



NFDI4
BIOIMAGE

NATIONAL RESEARCH DATA MANAGEMENT INFRASTRUCTURE
FOR MICROSCOPY AND BIOIMAGE ANALYSIS



GLOBAL BIOIMAGE
ANALYST'S SOCIETY

Kollaboratives Arbeiten und Versionskontrolle mit Git II

Robert Haase

GEFÖRDERT VOM



Diese Maßnahme wird gefördert durch die Bundesregierung aufgrund eines Beschlusses des Deutschen Bundestages. Diese Maßnahme wird mitfinanziert durch Steuermittel auf der Grundlage des von den Abgeordneten des Sächsischen Landtags beschlossenen Haushaltes.

ScaDS.**AI**
DRESDEN LEIPZIG

Robert Haase
@haesleinhuepf
Collaborative work / git
DataWeek Leipzig
April 15th 2024

These slides can be reused under the terms of the
[CC-BY 4.0](#) license unless mentioned otherwise.

37



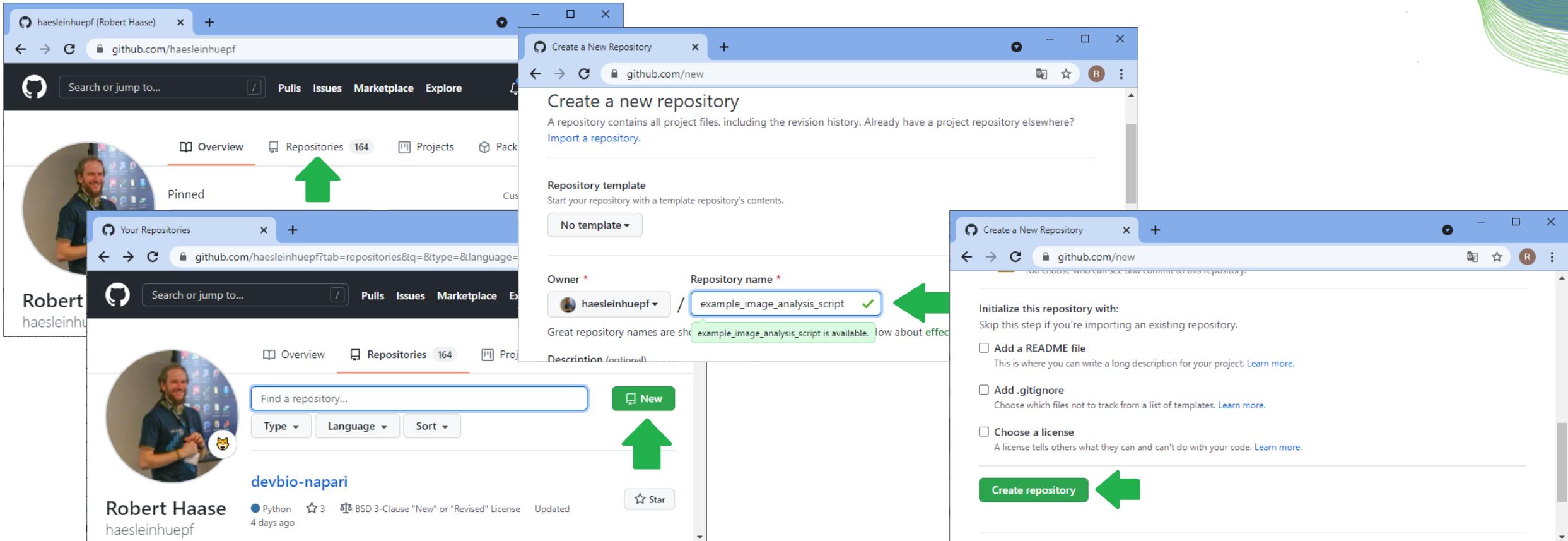
**TECHNISCHE
UNIVERSITÄT
DRESDEN**



UNIVERSITÄT
LEIPZIG

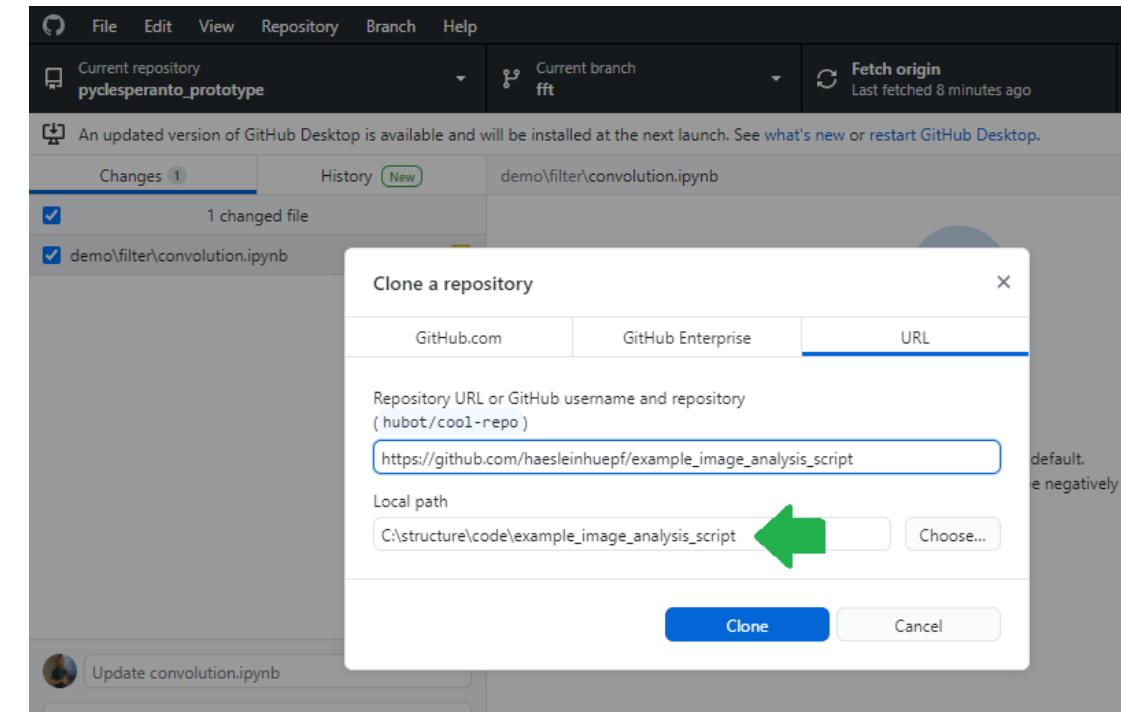
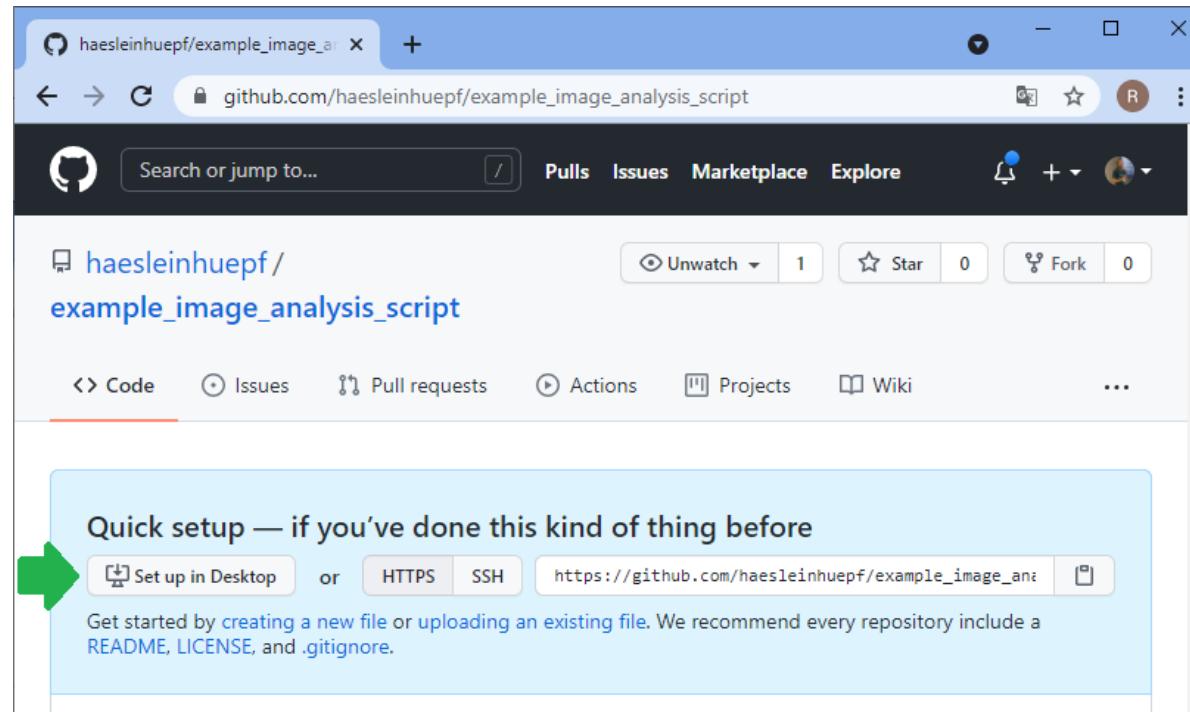
Github – Repositories anlegen

- Anlegen einer neuen Repository



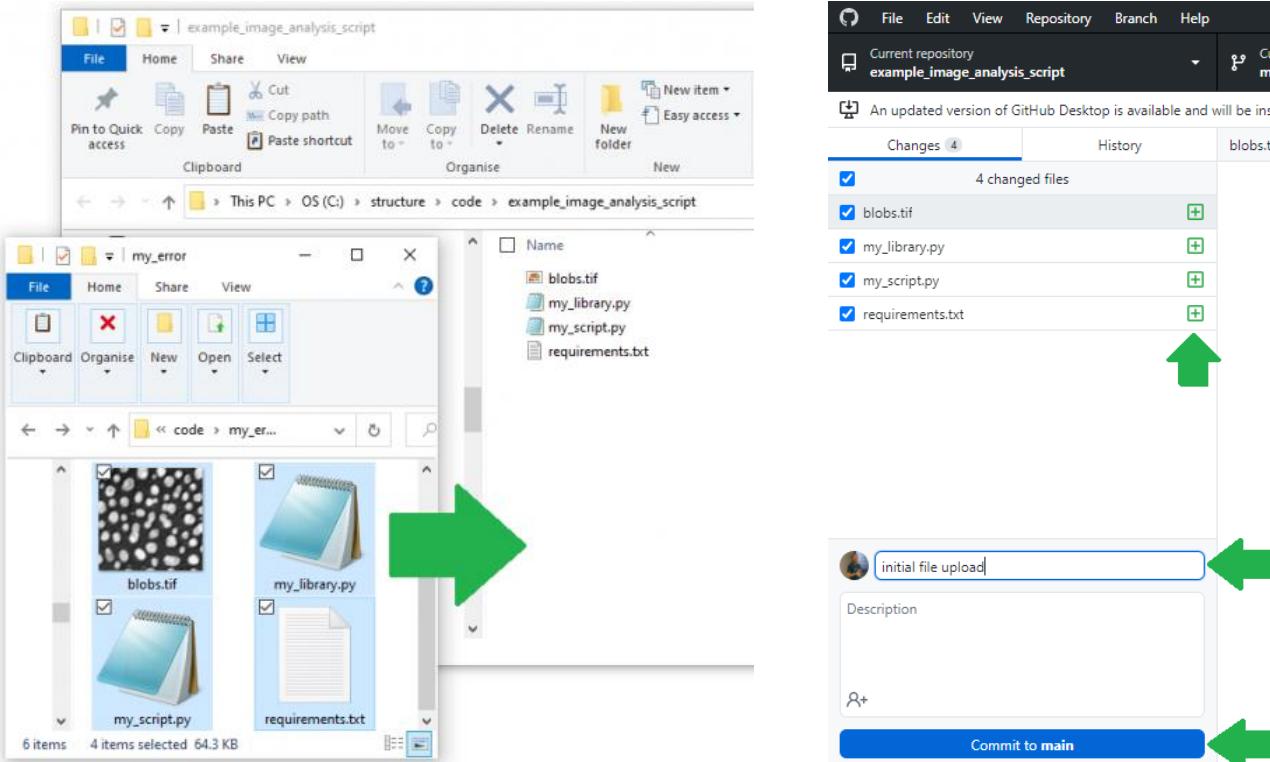
Github – repositories clonen

- git clone <https://github.com/organization/repository>
- Alternativ: Github Desktop



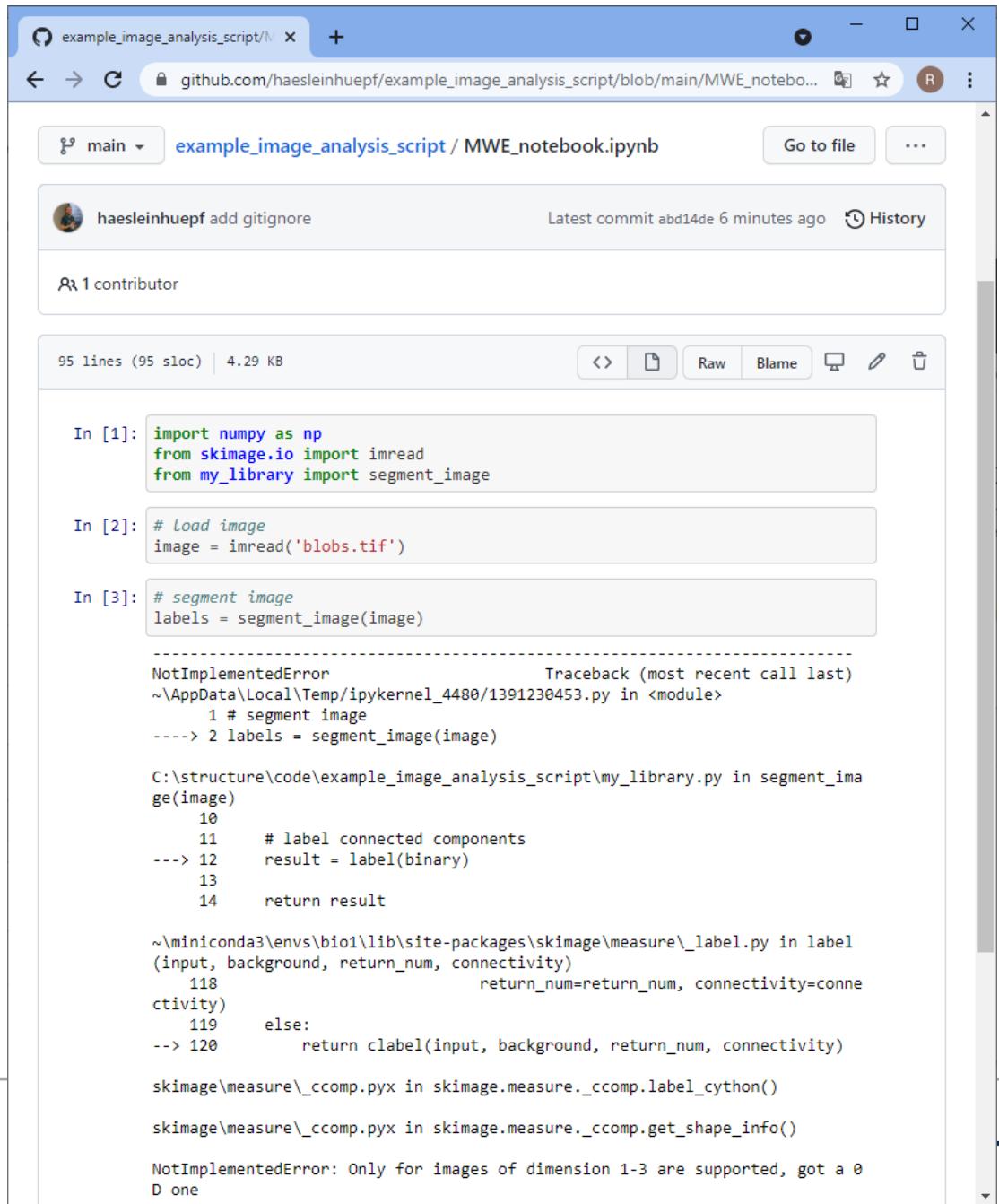
Github - Upload

- git [add], commit, push



Github

- Python Notebooks online lesbar
- Fehlersuche einfach, ohne etwas herunter laden zu müssen



The screenshot shows a GitHub browser interface for a Python notebook named 'MWE_notebook.ipynb' from the repository 'example_image_analysis_script'. The notebook has 95 lines of code (95 sloc) and a size of 4.29 KB. It contains three code cells:

```
In [1]: import numpy as np
from skimage.io import imread
from my_library import segment_image

In [2]: # Load image
image = imread('blobs.tif')

In [3]: # segment image
labels = segment_image(image)

-----  
NotImplementedError Traceback (most recent call last)
~\AppData\Local\Temp\ipykernel_4480\1391230453.py in <module>
      1 # segment image
----> 2 labels = segment_image(image)

C:\structure\code\example_image_analysis_script\my_library.py in segment_im
age
      10
      11     # label connected components
----> 12     result = label(binary)
      13
      14     return result

~\miniconda3\envs\bio1\lib\site-packages\skimage\measure\_label.py in label
(input, background, return_num, connectivity)
      118                                         return_num=return_num, connectivity=conne
ctivity)
      119     else:
--> 120         return clabel(input, background, return_num, connectivity)

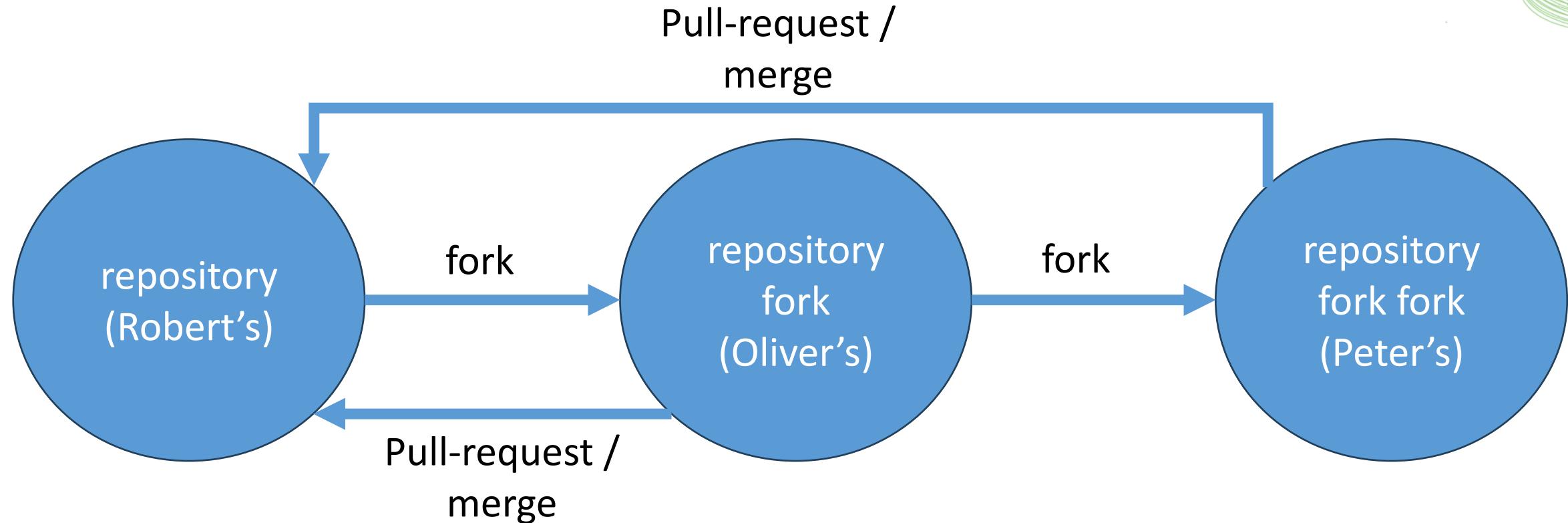
skimage\measure\_ccomp.pyx in skimage.measure._ccomp.label_cython()

skimage\measure\_ccomp.pyx in skimage.measure._ccomp.get_shape_info()

NotImplementedError: Only for images of dimension 1-3 are supported, got a 0
D one
```

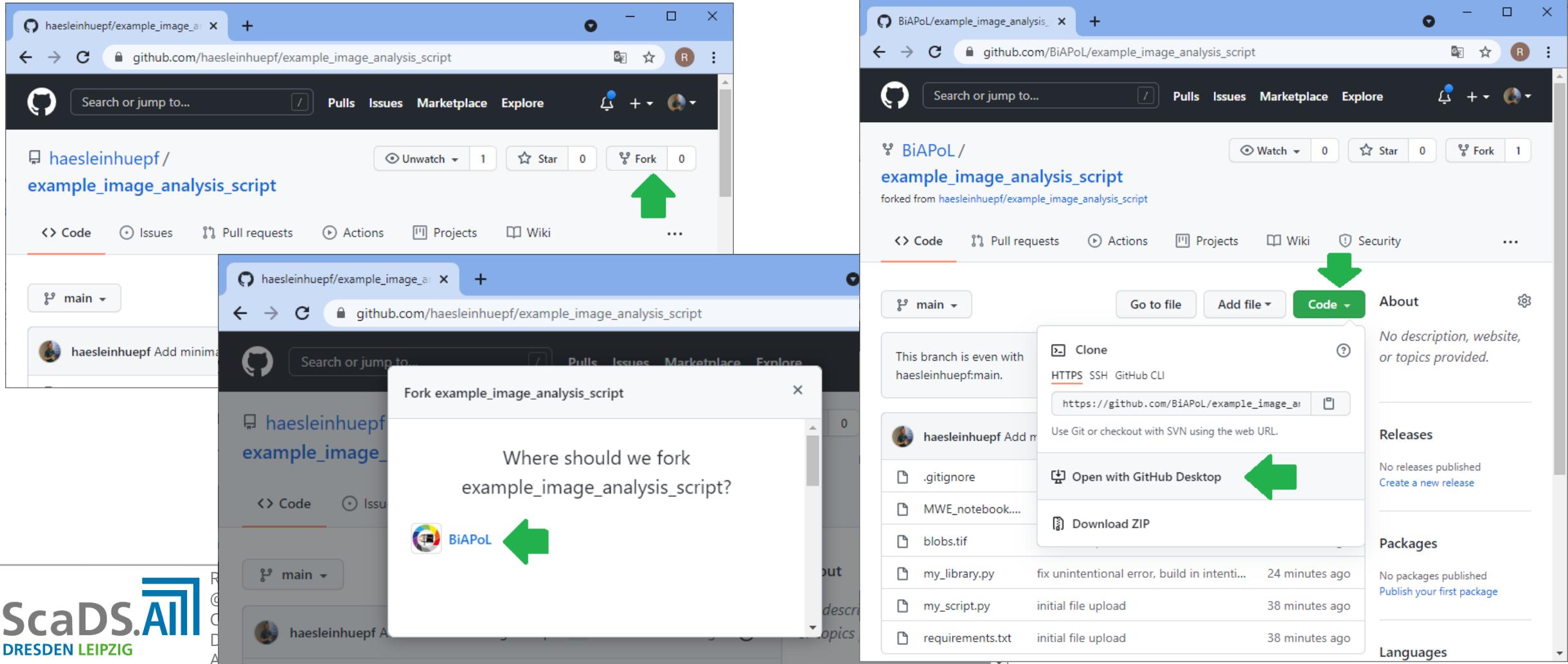
git - forking

- Ein Fork ist eine Kopie (die wir editieren können)



github - forking

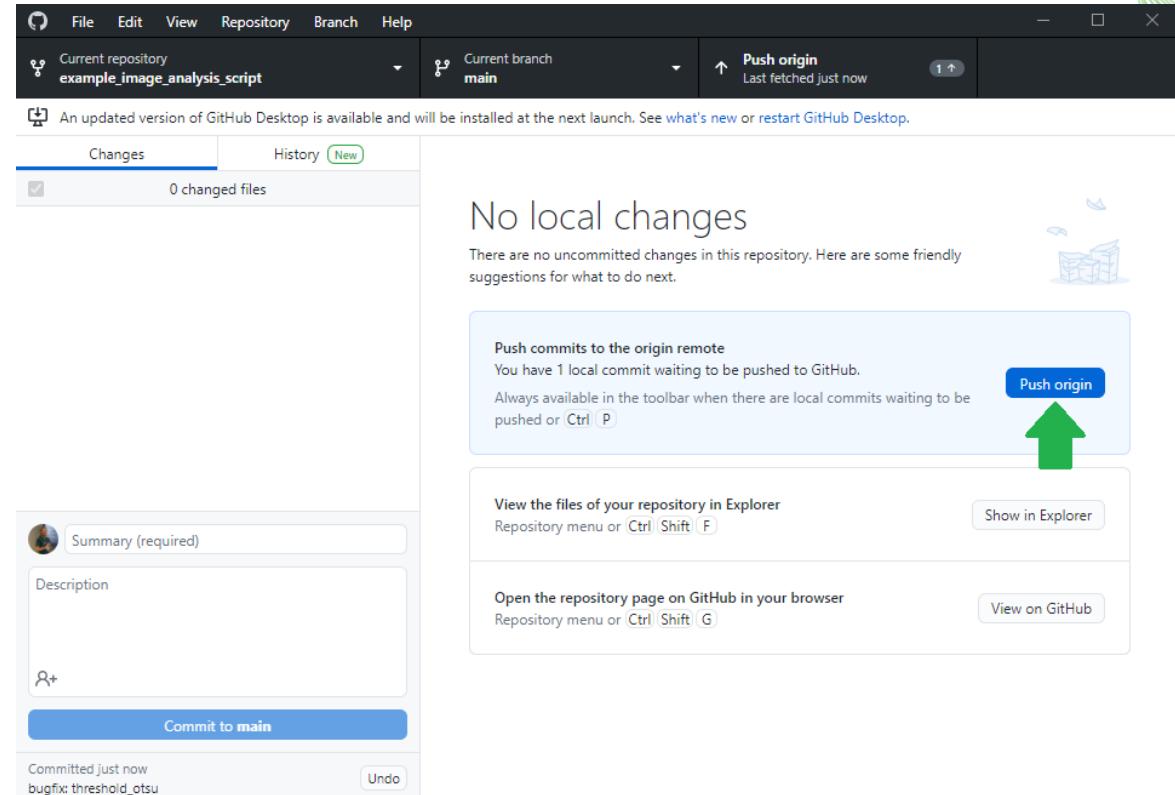
- Ein Fork ist eine Kopie (die wir editieren können)



github – Upload

- Unsere Änderungen werden zu unserem Fork hochgeladen

The screenshot shows the GitHub Desktop application interface. On the left, there's a sidebar with repository and branch selection. The main area displays a code diff for a file named `my_library.py`. A green arrow points from the bottom of the code editor towards the commit message. The commit message is titled "bugfix: threshold_otsu" and contains a note: "threshold.otsu delivers a number (the threshold), not a binary image. For thresholding the image, an additional step is necessary." A blue button at the bottom right of the commit message area is labeled "Commit to main".



Github – Pull-requests

- Der formale Akt etwas beizutragen

The image shows two screenshots of the GitHub interface illustrating the creation of a pull request.

Screenshot 1: Repository Overview

This screenshot shows the repository `BiAPoL/example_image_analysis_script`. The main branch is `main`. A green arrow points to the `Open pull request` button, which is highlighted in red. Another green arrow points to the `Upstream` dropdown menu, also highlighted in red.

Screenshot 2: Pull Request Creation Form

This screenshot shows the `Open a pull request` form. It includes fields for `base repository`, `head repository`, and `compare`. Below the form, a message from Robert states:

Dear Robert,
here comes a bug fix for your image segmentation function. `threshold_otsu` delivers a number (the threshold), not a binary image. For thresholding the image, an additional step is necessary.
Best,
Robert

Implizit: Ich
stimme Euren
Nutzungbedingu
ngen (Lizenz,
Copyright) zu

Github – Pull-requests

- Reviewer-Perspektive

The image displays two side-by-side screenshots of a GitHub pull request interface. On the left, a pull request titled "bugfix: threshold_otsu #1" is shown, where Robert Haase has merged one commit from the "BiAPoL:main" branch into the "haesleinhuepf:main" branch. The code diff shows changes in "my_library.py". A green arrow points to the "Files changed" section. On the right, the review interface for the same pull request is shown. It includes a message from the reviewer, Robert Haase, saying "Thank you Robert! That's great 😊". The interface also shows merge status, notifications, and other repository details.

Github – pull requests

- Reviewer perspective

A screenshot of a GitHub pull request page. The title is "bugfix: threshold_otsu #1". A purple "Merged" button is visible. The commit message is: "haesleinhuepf merged 1 commit into haesleinhuepf:main from BIAPOL:main now". Below the commit message, there is a comment from "haesleinhuepf" that reads:
Dear Robert,
here comes a bug fix for your image segmentation function. `threshold_otsu` delivers a number (the threshold), not a binary image. For thresholding the image, an additional step is necessary.
Best,
Robert

The commit message for the merged commit is: "bugfix: threshold_otsu ... 65c074a". A "Revert" button is present next to it. Another comment from "haesleinhuepf" says: "haesleinhuepf merged commit 44a28d7 into haesleinhuepf:main now". A "Revert" button is also present here. A final comment from "haesleinhuepf" says: "haesleinhuepf commented now Thank you Robert! That's great 😊".

A screenshot of a GitHub repository page for "haesleinhuepf/example_image_analysis_script". The repository has 1 unwatched, 0 stars, and 1 fork. The "Pull requests" tab is selected. A pull request titled "bugfix: threshold_otsu ..." is shown, merged into "haesleinhuepf:main" from "BIAPOL:main". The latest commit was made 17 minutes ago by "haesleinhuepf". The commit message is: "bugfix: threshold_otsu ...".
A green box highlights the commit message with the text: "Problem behoben :-)"
The repository contains a Jupyter notebook named "MWE_notebook.ipynb". The code in the notebook is:

```
In [1]: import numpy as np
from skimage.io import imread
from my_library import segment_image

In [2]: # Load image
image = imread('blobs.tif')

In [3]: # segment image
labels = segment_image(image)

In [4]: # count objects
number_of_objects = labels.max()
print('Number of objects', number_of_objects)
```

The output of the last cell is: "Number of objects 61".

Github

- Falls das zu schnell war...

The screenshot shows a web browser window displaying a blog post from the FocalPlane website. The header features the FocalPlane logo and the text "Where biology meets microscopy". Below the header is a navigation bar with links to Home, About us, Topics, Gallery, Jobs, Events, Resources, Network, Contact us, and Log in/register. A search icon is also present. The main content area shows a breadcrumb trail: Home / How to / Collaborative bio-image analysis script ... The title of the post is "Collaborative bio-image analysis script editing with git". It was posted by Robert Haase on 4 September 2021. The post begins with a TL;DR summary: "I'm a computer scientist who often collaborates with biologists on bio-image analysis scripts. We are using more and more git, a version control program, for working on code collaboratively. When using git, we speak about repositories, commits and pushing to the origin. We also make forks, send pull-requests and merge code. This blog post explains these terms and demonstrates how a typical collaborative bio-image analysis scripting project looks like." The text continues with a personal anecdote about writing a script that counts cells and needing help from experts.

The screenshot shows a web browser window displaying a GitHub profile page for the user "haesleinhuepf". The profile picture is a portrait of a smiling man. The GitHub interface includes a search bar, navigation links for Pulls, Issues, Marketplace, and Explore, and tabs for Overview, Repositories (164), Projects, and Packages. A green arrow points upwards towards the "Repositories" tab, likely indicating where to click to access the repository for sharing code.

Quiz

- Ein Fork ist ...

Eine Kopie



Ein Ordner



Eine private
Repository



Eine Datei



Mittagspause!

13:30 geht es weiter



CENTER FOR SCALABLE DATA ANALYTICS AND
ARTIFICIAL INTELLIGENCE



NFDI 4
BIOIMAGE

NATIONAL RESEARCH DATA MANAGEMENT INFRASTRUCTURE
FOR MICROSCOPY AND BIOIMAGE ANALYSIS



GLOBAL BIOIMAGE
ANALYST'S SOCIETY

Praktikum

Robert Haase



Robert Haase
@haesleinhuepf
Collaborative work / git
DataWeek Leipzig
April 15th 2024

These slides can be reused under the terms of the
[CC-BY 4.0](#) license unless mentioned otherwise.

51



TECHNISCHE
UNIVERSITÄT
DRESDEN



UNIVERSITÄT
LEIPZIG

GEFÖRDERT VOM



Bundesministerium
für Bildung
und Forschung



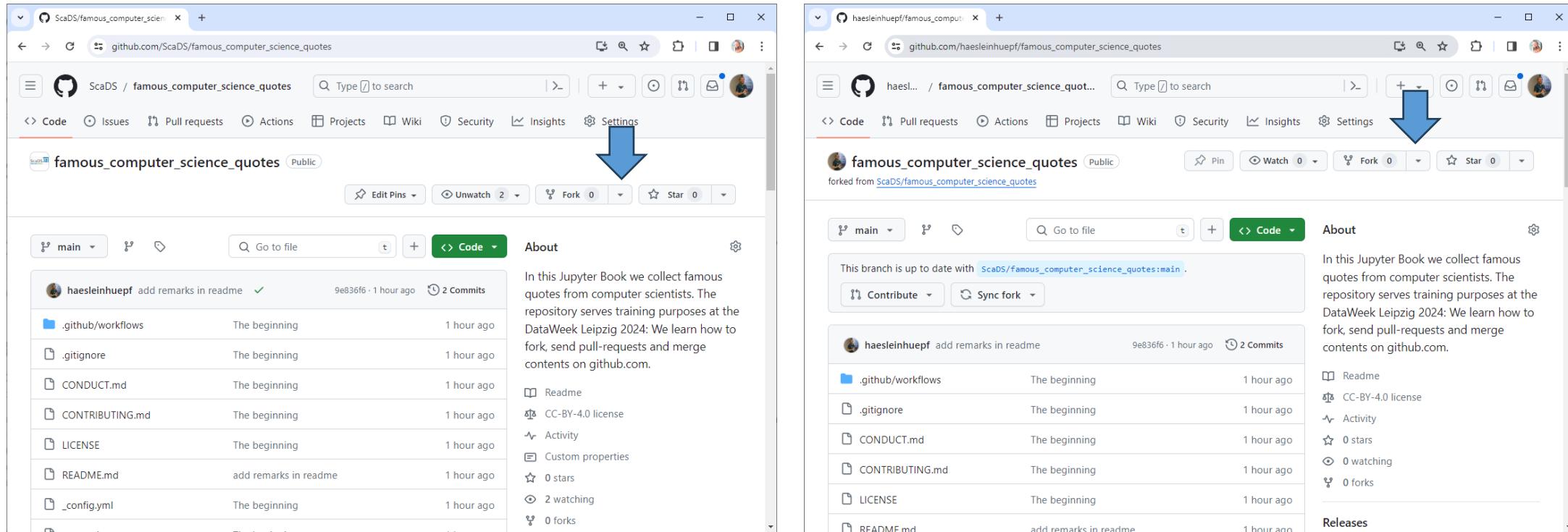
Diese Maßnahme wird gefördert durch die Bundesregierung
aufgrund eines Beschlusses des Deutschen Bundestages.
Diese Maßnahme wird mitfinanziert durch Steuermittel auf
der Grundlage des von den Abgeordneten des Sächsischen
Landtags beschlossenen Haushaltes.

Praktikum: Merge-Konflikte

- Ziel: Erzeugt und löst einen Merge-Konflikt
- Arbeitet zu zweit oder in kleinen Gruppen

Praktikum: Merge-Konflikte

- Schritt 1: Forkt diese Repository
- Schritt 2: Forkt den Fork Eurer Nachbarn



Praktikum: Merge-Konflikte

- Euer task: Ändert eine Datei in Eurem Fork
- Task der Nachbarn: Ändert die gleiche Datei in deren Fork

The image shows two side-by-side GitHub commit comparison interfaces. The left window shows a commit from 'haesleinhuepf' on April 7, 2024, adding a quote from Bill Gates. The right window shows a commit from 'haesleinhuepf' changing the order of quotes in a file named 'computer_science.md'. A large blue callout box in the bottom center of the interface contains the text: 'Modifiziert die gleiche Datei. Erzeugt einen Konflikt!' (Modifies the same file. Creates a conflict!).

Comparing main...haesleinhuepf | +

github.com/haesleinhuepf/famous_computer_science_quotes/compare/main...haesleinhuepf-patch-1

-o 1 commit | 1 file changed | Ax 1 contributor

Commits on Apr 7, 2024

Add Bill Gates quote
haesleinhuepf committed now

Showing 1 changed file with 2 additions and 0 deletions.

computer_science.md

@@ -2,6 +2,8 @@
2 2
3 3 * "There are two ways to write error-free programs; only the third one works." Alan Perlis [source] (https://www.brainyquote.com/quotes/alan_perlis_177353)
4 4
5 + * "The computer was born to solve problems that did not exist before." – Bill Gates [source] (<https://www.create-learn.us/blog/computer-science-quotes/>)
6 +
7 * "There are two methods in software design. One is to make the program so simple, there are obviously no errors. The other is to make it so complicated, there are no obvious errors." Tony Hoare [source] (https://www.brainyquote.com/quotes/tony_hoare_620783)
8
9

reordered quotes - haesleinhuepf | +

github.com/haesleinhuepf/famous_computer_science_quotes/commit/e03fc6dc46d50191a7c6e411fcc9fc65ec756c1

reordered quotes

main

haesleinhuepf committed now Verified

1 parent 9e836f6 commit e03fc6d

Show 1 changed file with 1 addition and 2 deletions.

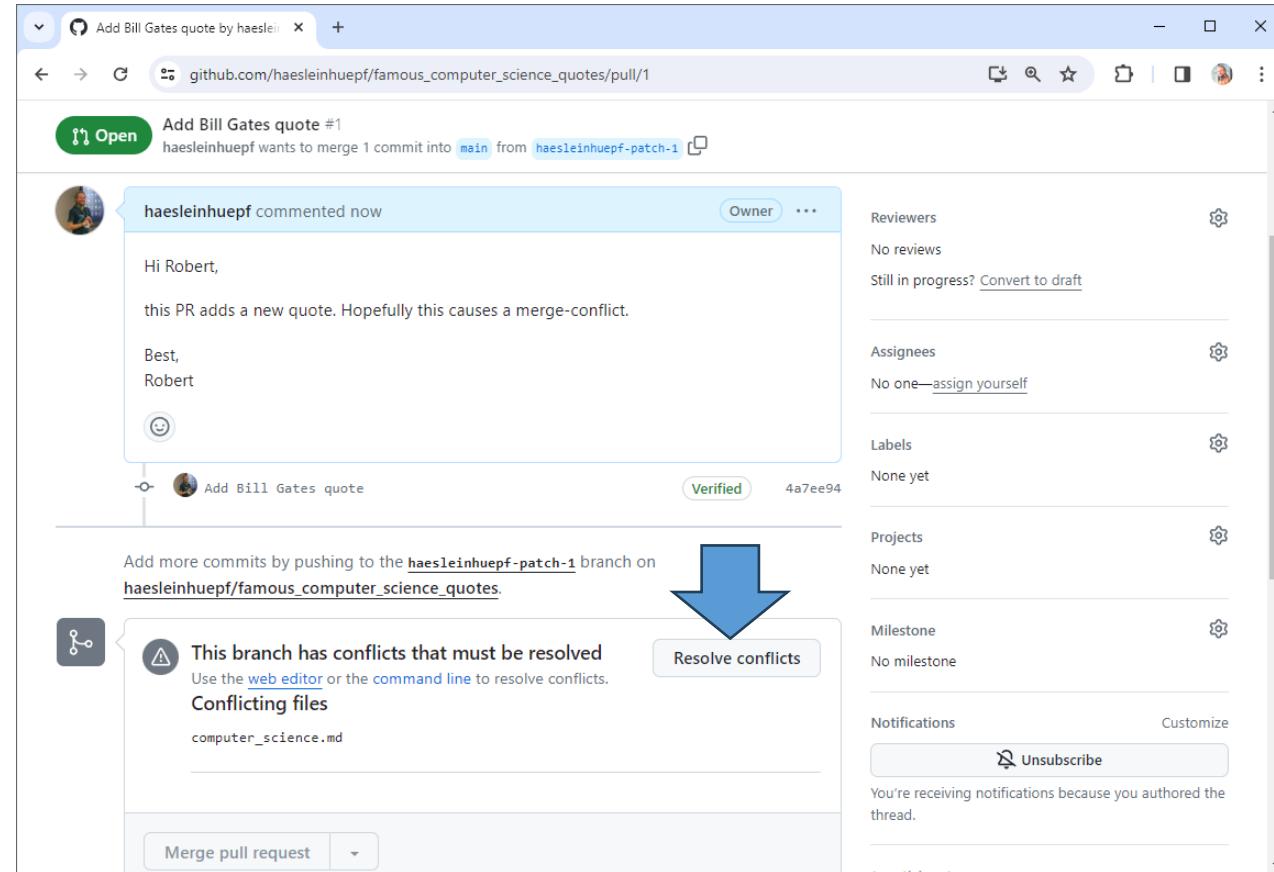
computer_science.md

@@ -1,9 +1,8 @@
1 1 # Computer Science Quotes
2 2
3 - * "There are two ways to write error-free programs; only the third one works." Alan Perlis [source] (https://www.brainyquote.com/quotes/alan_perlis_177353)
4 -
5 + * "There are two methods in software design. One is to make the program so simple, there are obviously no errors. The other is to make it so complicated, there are no obvious errors." Tony Hoare [source] (https://www.brainyquote.com/quotes/tony_hoare_620783)
6 +
7 + * "There are two ways to write error-free programs; only the third one works." Alan Perlis [source] (https://www.brainyquote.com/quotes/alan_perlis_177353)
8
9

Modifiziert die gleiche Datei.
Erzeugt einen Konflikt!

Praktikum: Merge-Konflikte

- Sendet einen Pull-request zu Euren Nachbarn
- Löst den Konflikt





ScaDS.AI
DRESDEN LEIPZIG

CENTER FOR SCALABLE DATA ANALYTICS AND
ARTIFICIAL INTELLIGENCE



NFDI4
BIOIMAGE

NATIONAL RESEARCH DATA MANAGEMENT INFRASTRUCTURE
FOR MICROSCOPY AND BIOIMAGE ANALYSIS



GLOBAL BIOIMAGE
ANALYST'S SOCIETY

Github-Zenodo-Integration & Github Pages

Robert Haase

GEFÖRDERT VOM



Bundesministerium
für Bildung
und Forschung



Diese Maßnahme wird gefördert durch die Bundesregierung aufgrund eines Beschlusses des Deutschen Bundestages.
Diese Maßnahme wird mitfinanziert durch Steuermittel auf der Grundlage des von den Abgeordneten des Sächsischen Landtags beschlossenen Haushaltes.

ScaDS.AI
DRESDEN LEIPZIG

Robert Haase
@haesleinhuepf
Collaborative work / git
DataWeek Leipzig
April 15th 2024

These slides can be reused under the terms of the
[CC-BY 4.0](#) license unless mentioned otherwise.

56



**TECHNISCHE
UNIVERSITÄT
DRESDEN**



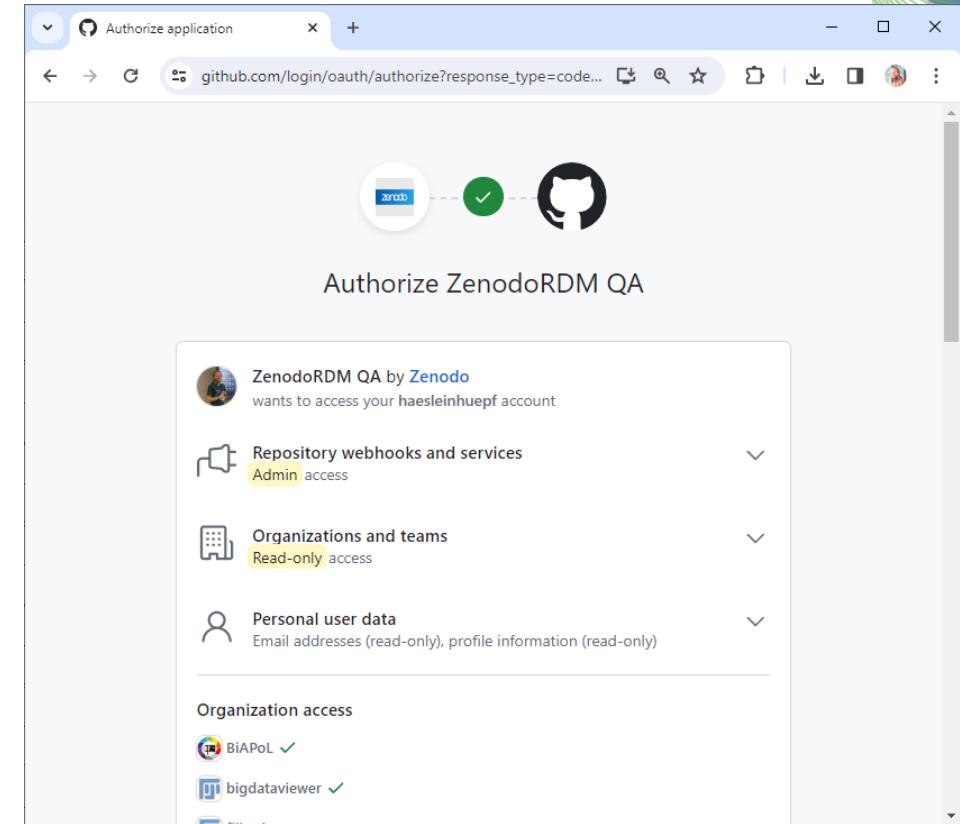
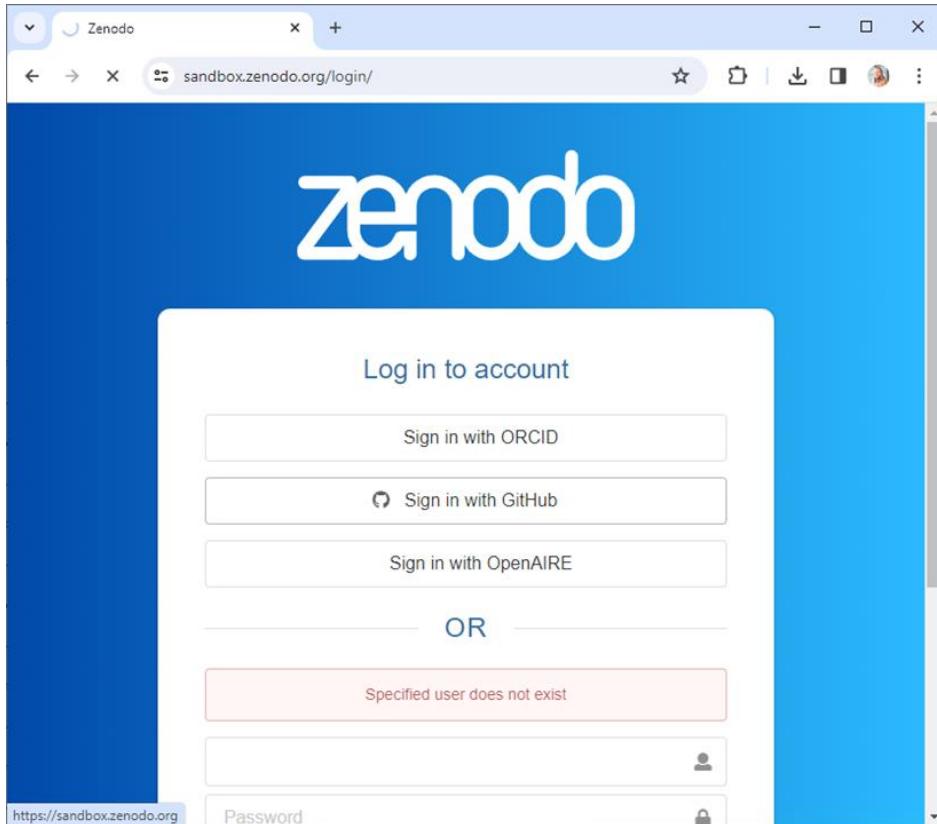
UNIVERSITÄT
LEIPZIG

Github versus Zenodo

- Github
 - Gemeinsames Arbeiten
 - Kleine Änderungen werden einzeln erfasst
 - Exaktes Logbuch
- Zenodo
 - öffentlich finanzierte Infrastruktur
 - Archivlösung
 - Nur Releases werden archiviert.

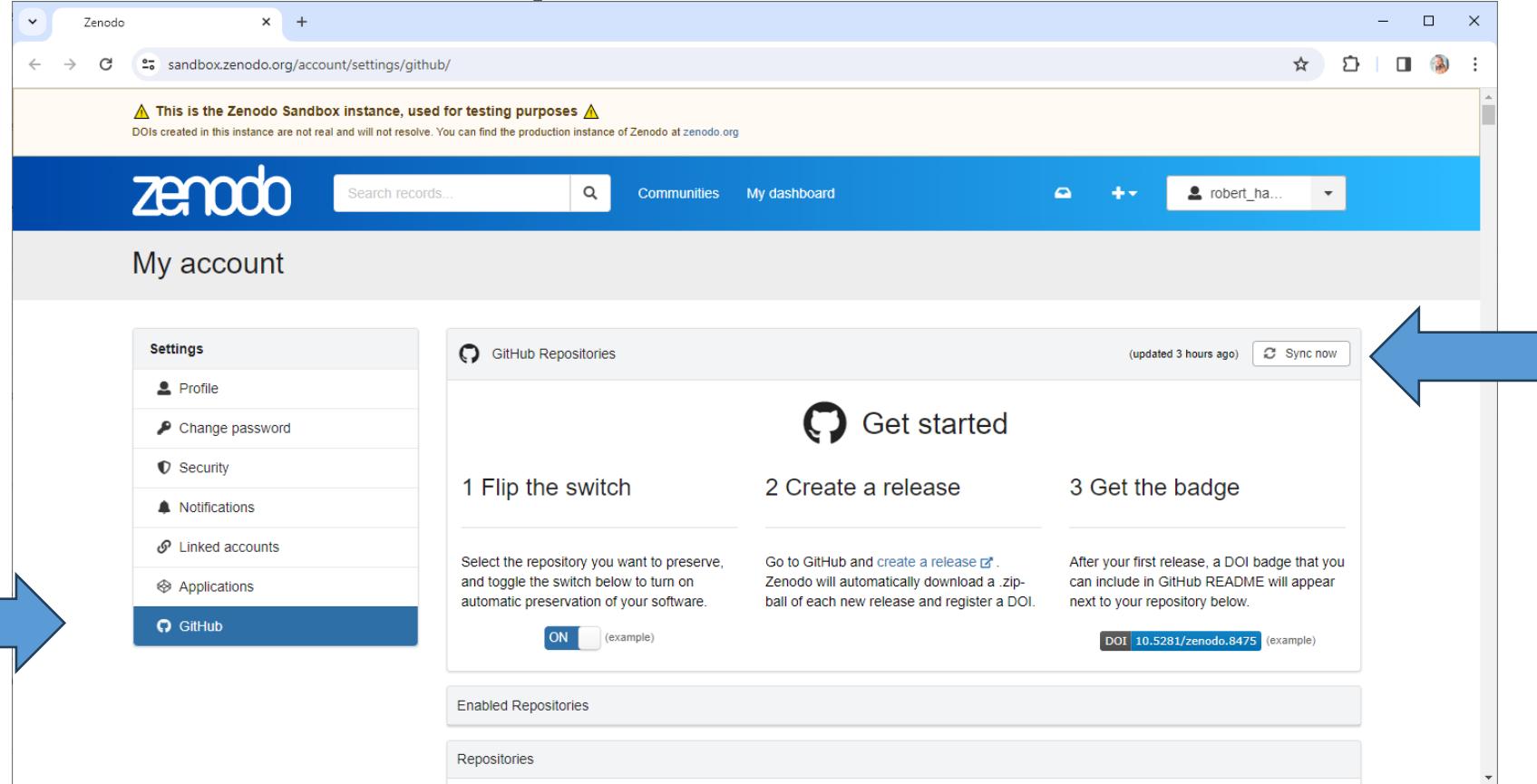
Zenodo-Integration

- Login bei Zenodo mit dem Github-Account



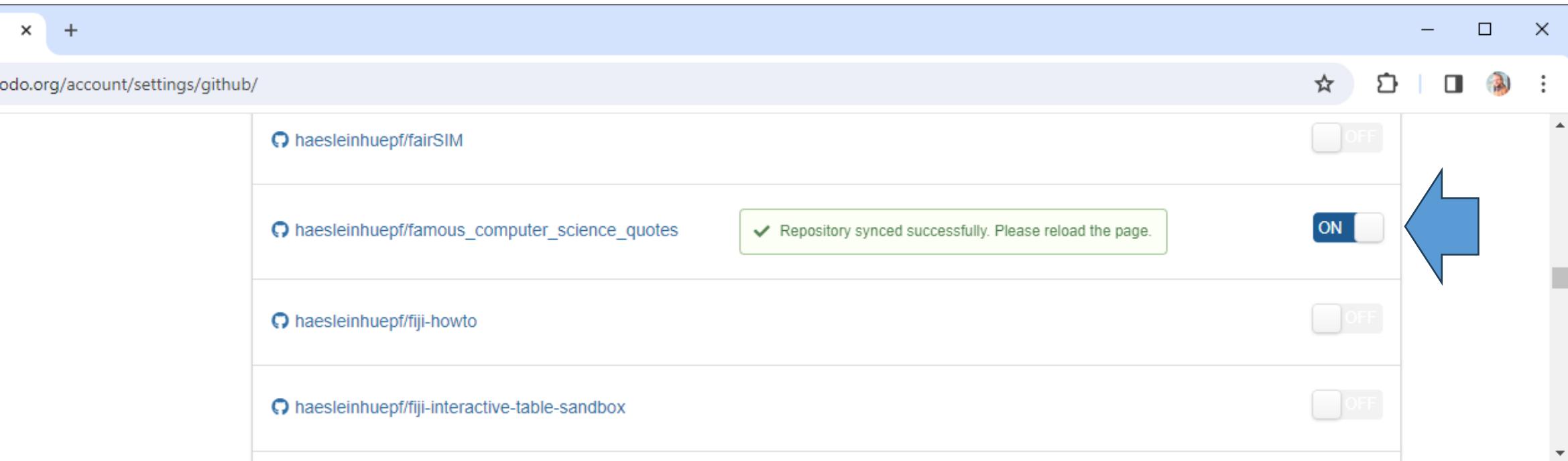
Zenodo-Integration

- Github mit Zenodo synchronisieren



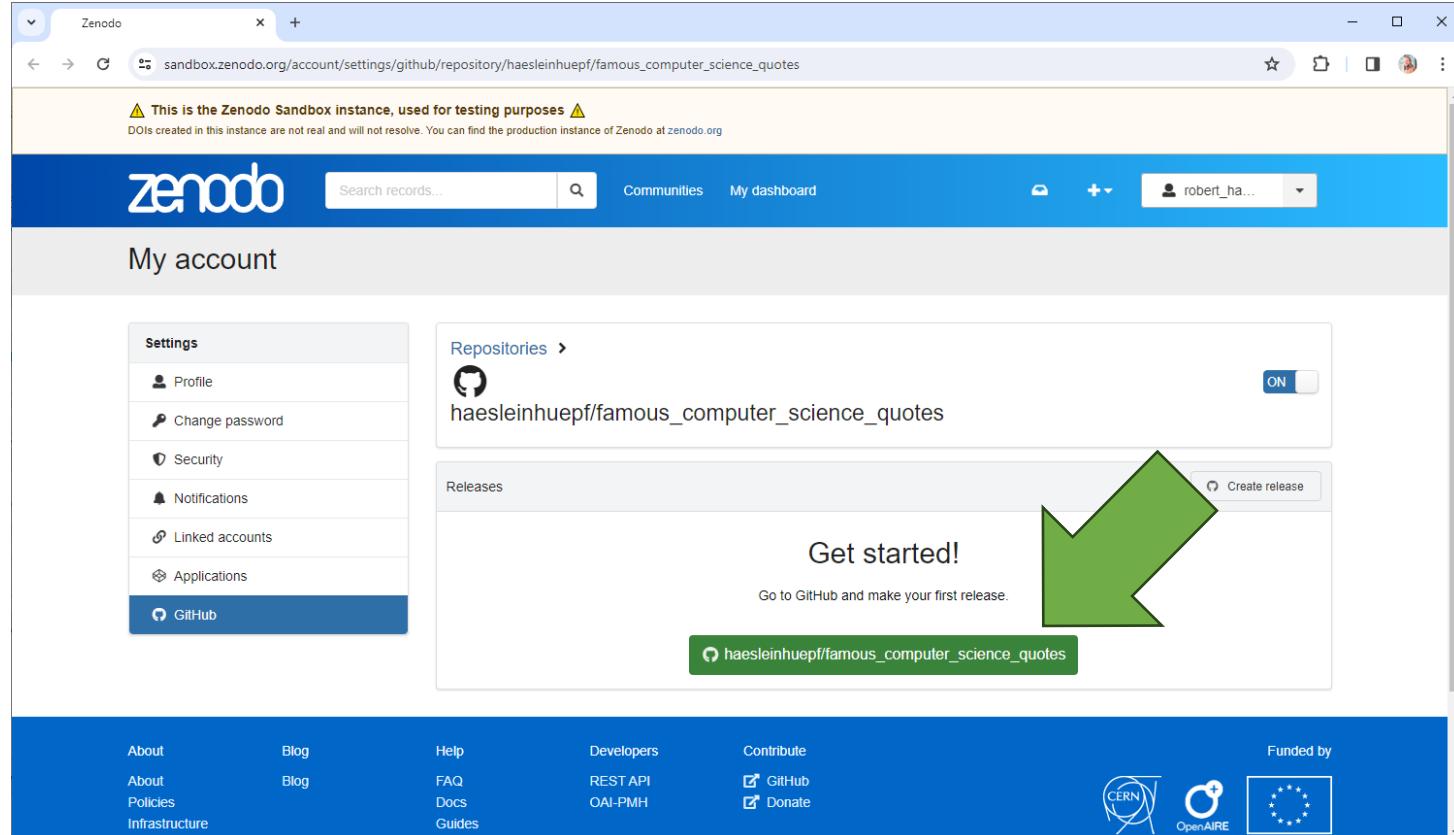
Zenodo integration

- Suche nach dem Projekt, das archiviert werden soll
- Schalte Archivierung ein



Zenodo integration

- Klick auf das Repository um Details zu sehen



Zenodo integration

- Erzeugt einen Versionstag (bspw. "v1")

The screenshot shows two side-by-side views of the GitHub 'Releases' page for a repository named 'famous_computer_science_quotes'. Both views show a release titled 'v1'.

Left View: This view is from a browser tab titled 'New release · haesleinhuepf/famous_computer_science_quotes/releases/new'. The 'Write' tab is selected, showing a rich text editor with the placeholder text: 'This is the first release. We use it to generate a DOI on Zenodo (sandbox)'. Below the editor, there's a note: 'Markdown is supported' and a file upload area: 'Paste, drop, or click to add files'. At the bottom, there are buttons for 'Publish release' and 'Save draft'.

Right View: This view is from a browser tab titled 'GitHub · New release · haesleinhuepf/famous_computer_science_quotes/releases/new'. It shows the same release details and content as the left view. A blue arrow points from the right side of the left view's editor area to the right view's editor area, highlighting the consistency of the text.

Zenodo integration

- Erkundet den Release auf Github und Zenodo

The image displays two browser windows illustrating the integration of GitHub and Zenodo. The left window shows a GitHub release page for the repository 'haesleinhuepf/famous_computer_science_quotes'. A new release 'v1' has been created, indicated by a green 'Latest' badge. The release notes state: 'This is the first release. We use it to generate a DOI on Zenodo (sandbox.)'. Below the release, there are two assets: 'Source code (zip)' and 'Source code (tar.gz)', both uploaded 30 minutes ago. The right window shows a Zenodo account page for the user 'robert_ha...'. Under 'Repositories', the repository 'haesleinhuepf/famous_computer_science_quotes' is listed. In the 'Releases' section, a release 'v1' is shown with a DOI of 10.5072/zenodo.43520, marked as 'Published' just now.

Zenodo integration

- Zurück auf dem Dashboard wird der neue Eintrag angezeigt

The screenshot shows the Zenodo dashboard with a blue header. A banner at the top states: "This is the Zenodo Sandbox instance, used for testing purposes. DOIs created in this instance are not real and will not resolve. You can find the production instance of Zenodo at zenodo.org". Below the banner, the user's profile picture and name "Robert Haase" are displayed. A green "New upload" button is visible. The main area shows a list of uploads under "My uploads". One entry is highlighted: "haesleinhuepf/famous_computer_science_quotes: v1" by Robert Haase, published on April 7, 2024. It is categorized as Software and has an "Open" link. Below the entry, it says: "This is the first release. We use it to generate a DOI on Zenodo (sandbox)".

The screenshot shows the Zenodo record page for the uploaded file. The URL is "haesleinhuepf/famous_computer_science_quotes: v1". The page header includes the Zenodo logo and a "Manage record" button. It displays the publication information: "Published April 7, 2024 | Version v1" and "haesleinhuepf/famous_computer_science_quotes: v1" by Robert Haase. A "Show affiliations" button is present. The "Files" section shows the contents of the uploaded zip file: "haesleinhuepf/famous_computer_science_quotes-v1.zip" containing "haesleinhuepf-famous_computer_science_quotes-e03fc6d", ".github", "workflows", "book.yml", and ".gitignore". File sizes are listed as 817 Bytes and 65 Bytes.

Github pages

- Kostenloses Webhosting, weitgehend automatisch

The image shows two side-by-side browser windows. Both windows have a blue header bar with the URL `scads.github.io/famous_computer_science_quotes/intro.html`. The left window displays the 'intro.html' page, which features the ScaDS.AI logo and navigation links for 'Computer Science Quotes', 'Chapters', 'Data Science Quotes', 'Artificial Intelligence Quotes', and 'Imprint'. The right window displays the 'computer_science.html' page, which also features the ScaDS.AI logo and navigation links. This page contains a list of computer science quotes, such as "There are two ways to write error-free programs; only the third one works." by Alan Perlis, and "There are two methods in software design. One is to make the program so simple, there are obviously no errors. The other is to make it so complicated, there are no obvious errors." by Tony Hoare. Navigation arrows for 'Previous' and 'Next' are visible at the bottom of the right-hand page.

https://scads.github.io/famous_computer_science_quotes

65

Github pages: Jupyter Books

- Basiert auf dem Jupyter Books Projekt

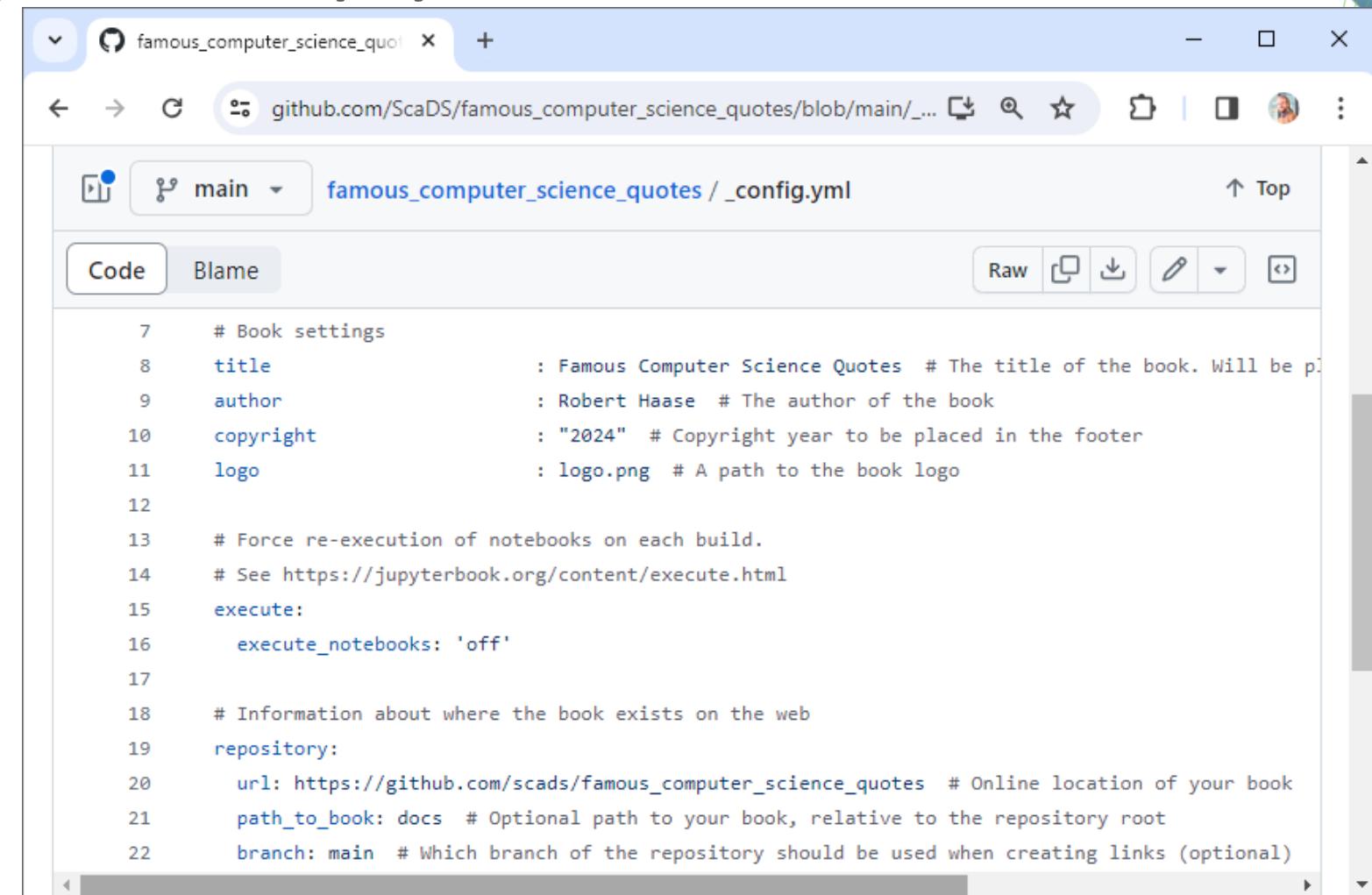
The image shows two side-by-side screenshots of the Jupyter Book documentation website.

Left Screenshot: The title is "Create your first book". It features a sidebar with "Tutorials" and "Create your first book" sections. The main content area has a heading "Create your first book" and a sub-section "What you should already know" which says: "In order to complete this tutorial, you should be relatively familiar with using a text editor." Below it is a note for Windows users: "Jupyter Book is now also tested against Windows OS 😊 However, there is a known incompatibility for notebook execution when using...".

Right Screenshot: The title is "Publish your book on the internet". It also has a sidebar with "Structure and organize content", "Write narrative content", "Write executable content", and "Build and publish outputs" sections. The main content area has a heading "Publish your book on the internet" and a "Tip" section: "When publishing to online locations, it may be useful to activate error logging on the terminal in MyST-NB. This allows you to see tracebacks that happen as a result of executing notebooks as part of your build process. To do so, see [Execution tracebacks in the terminal](#)." It also lists "GitHub Pages and Actions" and "Netlify" options.

Github pages: Jupyter Books

- Configuration:
`_config.yml`

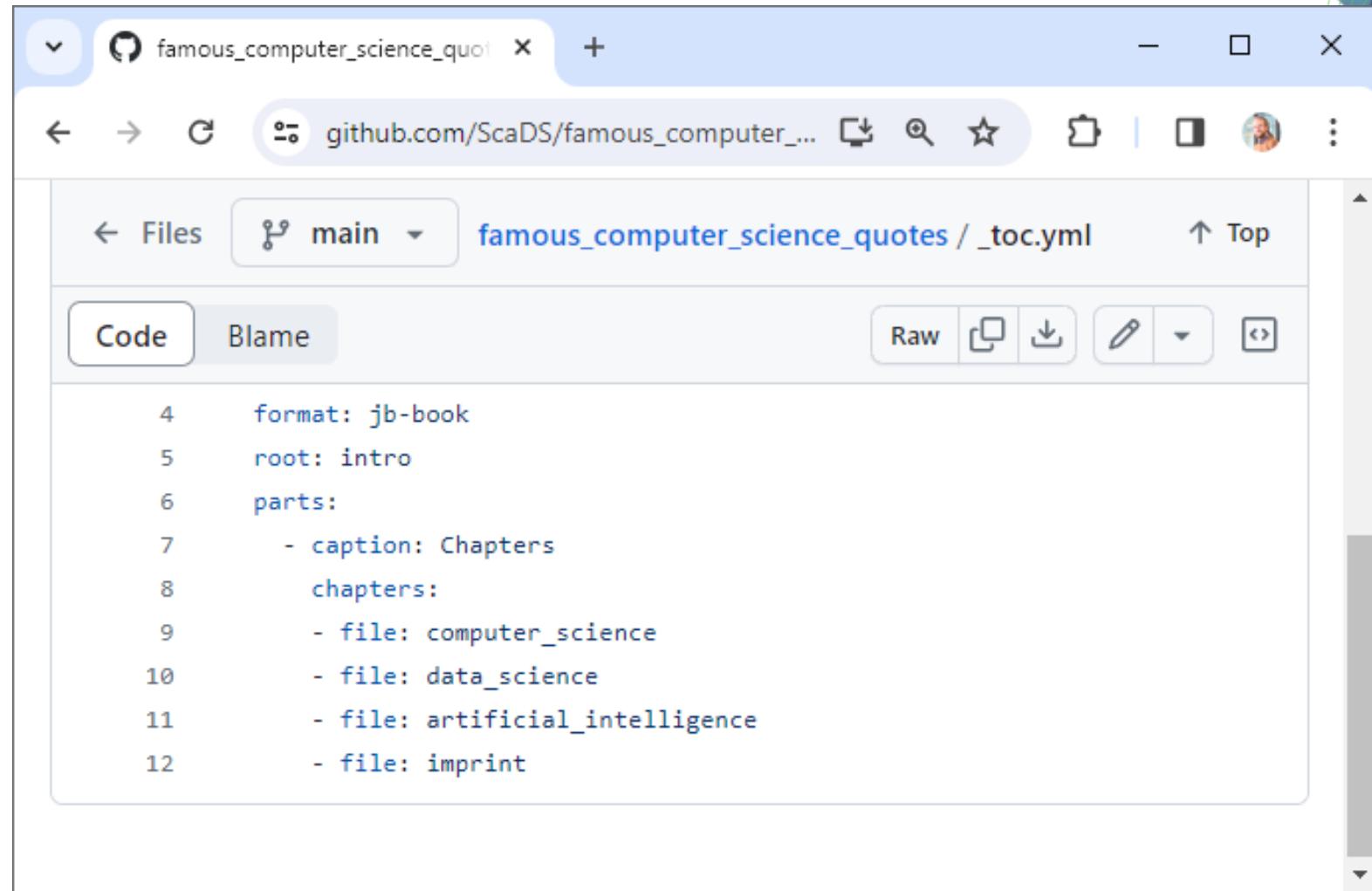


The screenshot shows a GitHub browser window with the URL github.com/ScaDS/famous_computer_science_quotes/blob/main/_config.yml. The page displays the `_config.yml` configuration file for a Jupyter Book. The code defines book settings like title, author, and copyright, and specifies build options such as execute_notebooks: 'off'. It also includes information about the book's online location and path_to_book.

```
7  # Book settings
8  title           : Famous Computer Science Quotes # The title of the book. Will be p:
9  author          : Robert Haase # The author of the book
10 copyright       : "2024" # Copyright year to be placed in the footer
11 logo            : logo.png # A path to the book logo
12
13 # Force re-execution of notebooks on each build.
14 # See https://jupyterbook.org/content/execute.html
15 execute:
16   execute_notebooks: 'off'
17
18 # Information about where the book exists on the web
19 repository:
20   url: https://github.com/scads/famous_computer_science_quotes # Online location of your book
21   path_to_book: docs # Optional path to your book, relative to the repository root
22   branch: main # Which branch of the repository should be used when creating links (optional)
```

Github pages: Jupyter Books

- Configuration:
`_toc.yml`

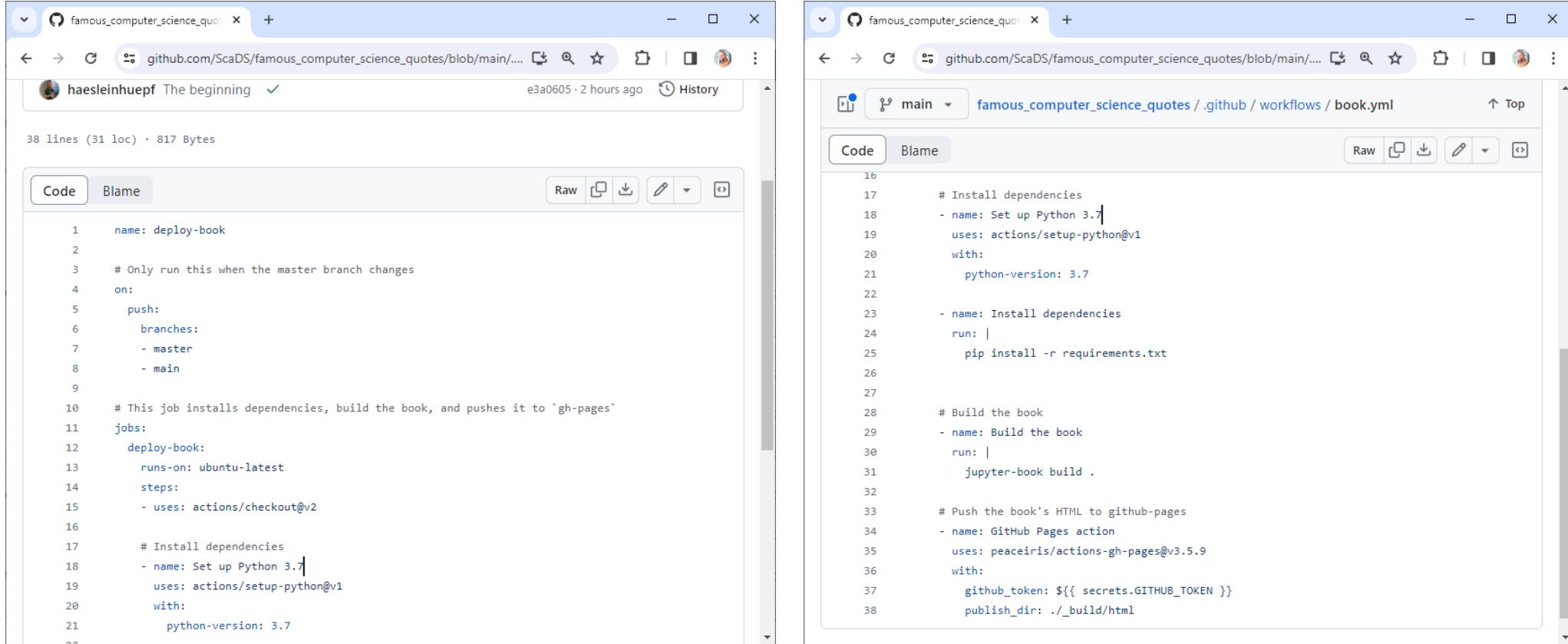


The screenshot shows a GitHub browser window with the URL github.com/ScaDS/famous_computer_science_quotes. The current file is `_toc.yml`, which contains the following YAML configuration:

```
format: jb-book
root: intro
parts:
- caption: Chapters
  chapters:
  - file: computer_science
  - file: data_science
  - file: artificial_intelligence
  - file: imprint
```

Github pages: Jupyter Books

- Behind the scenes: Github workflows



The image shows two side-by-side screenshots of a GitHub code editor interface. Both windows have the URL https://github.com/ScaDS/famous_computer_science_quotes/blob/main/.github/workflows/book.yml in the address bar.

Screenshot 1 (Left): This window shows the first half of the workflow file. It includes sections for defining a job named 'deploy-book' that runs on the 'master' and 'main' branches. The job installs dependencies using Python 3.7, runs a checkout step, and then performs a 'jupyter-book build' command.

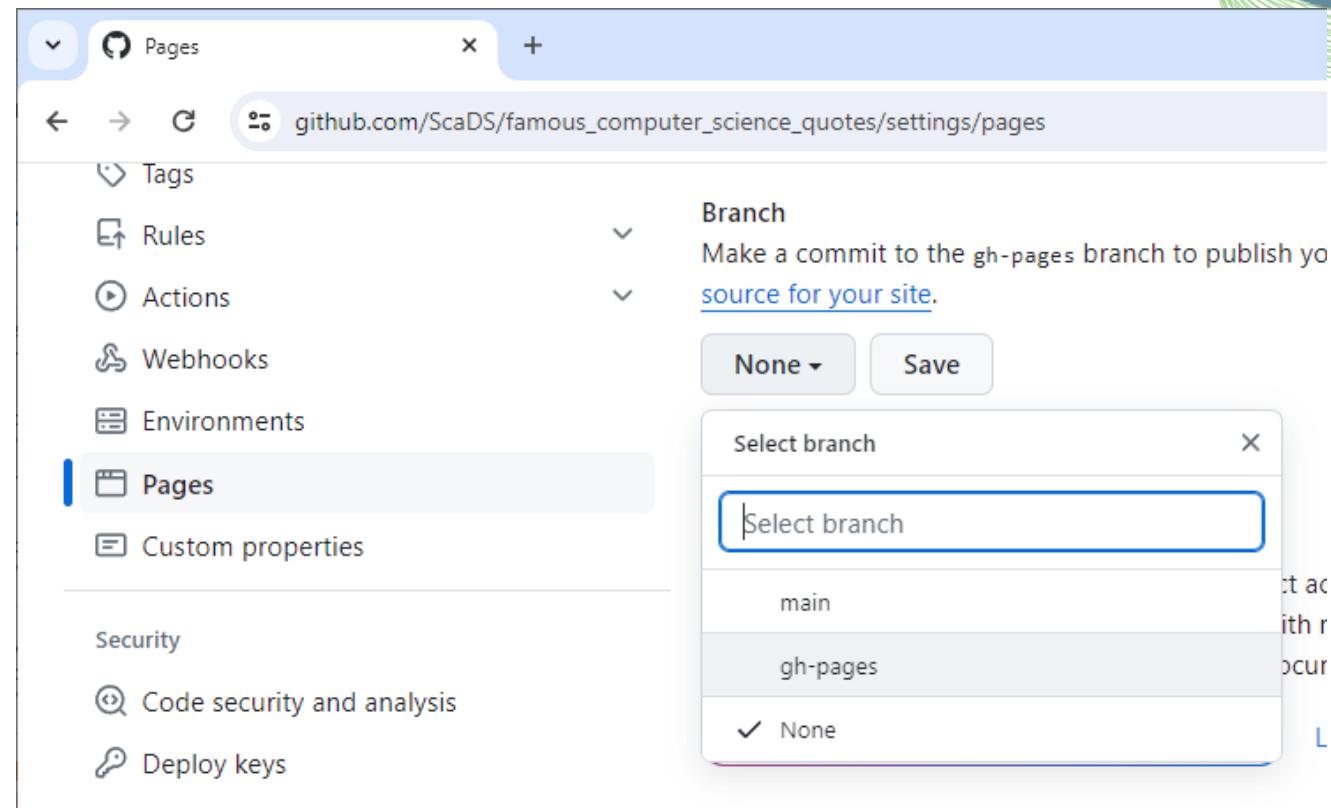
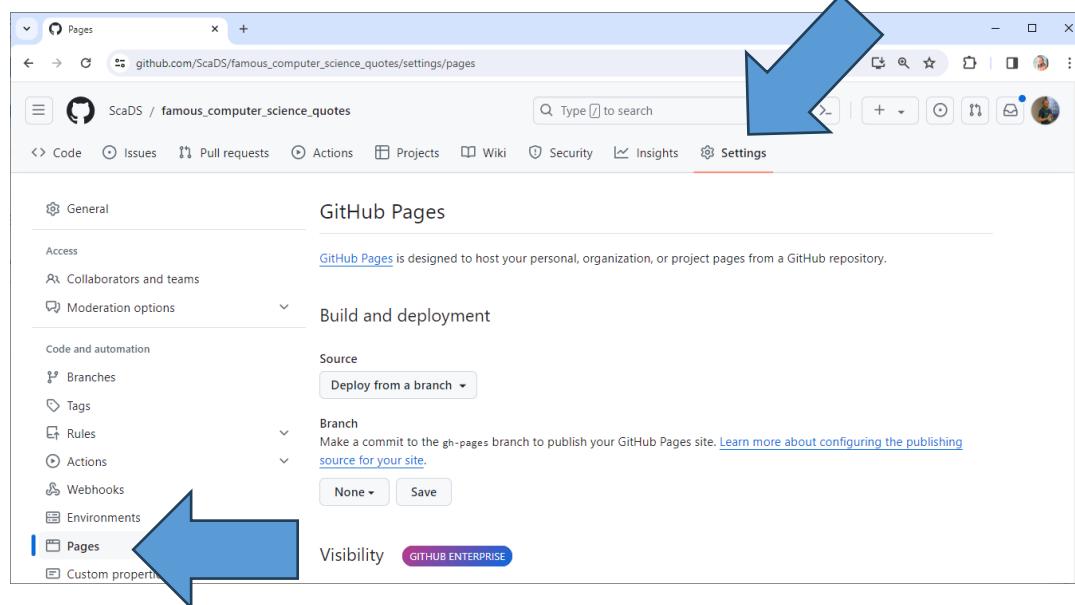
```
1 name: deploy-book
2
3 # Only run this when the master branch changes
4 on:
5   push:
6     branches:
7       - master
8       - main
9
10 # This job installs dependencies, build the book, and pushes it to `gh-pages`
11 jobs:
12   deploy-book:
13     runs-on: ubuntu-latest
14     steps:
15       - uses: actions/checkout@v2
16
17     # Install dependencies
18     - name: Set up Python 3.7
19       uses: actions/setup-python@v1
20       with:
21         python-version: 3.7
```

Screenshot 2 (Right): This window shows the second half of the workflow file. It continues from the previous section, adding a step to install dependencies using pip, followed by another 'jupyter-book build' command, and finally pushing the generated HTML files to the 'gh-pages' branch using the GitHub Pages action.

```
16
17   # Install dependencies
18   - name: Set up Python 3.7
19     uses: actions/setup-python@v1
20     with:
21       python-version: 3.7
22
23   - name: Install dependencies
24     run: |
25       pip install -r requirements.txt
26
27   # Build the book
28   - name: Build the book
29     run: |
30       jupyter-book build .
31
32   # Push the book's HTML to github-pages
33   - name: GitHub Pages action
34     uses: peaceiris/actions-gh-pages@v3.5.9
35     with:
36       github_token: ${{ secrets.GITHUB_TOKEN }}
37       publish_dir: ./_build/html
```

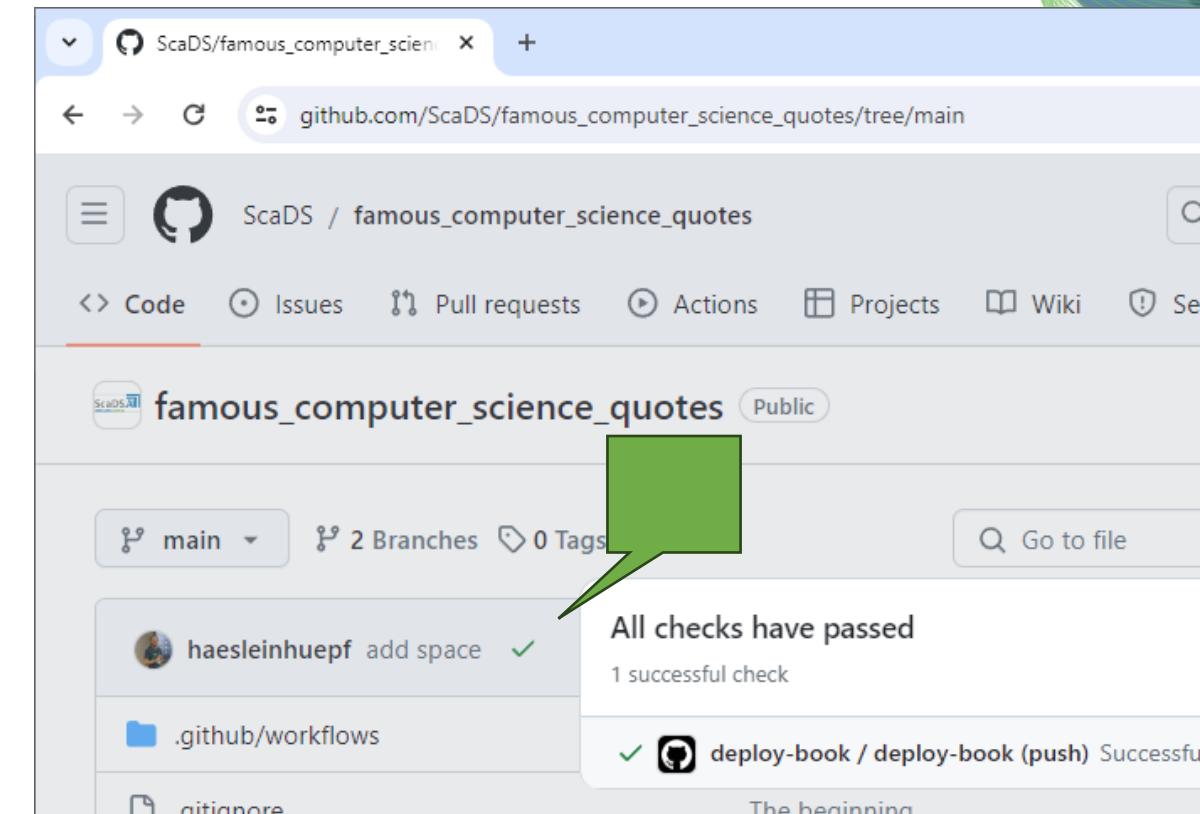
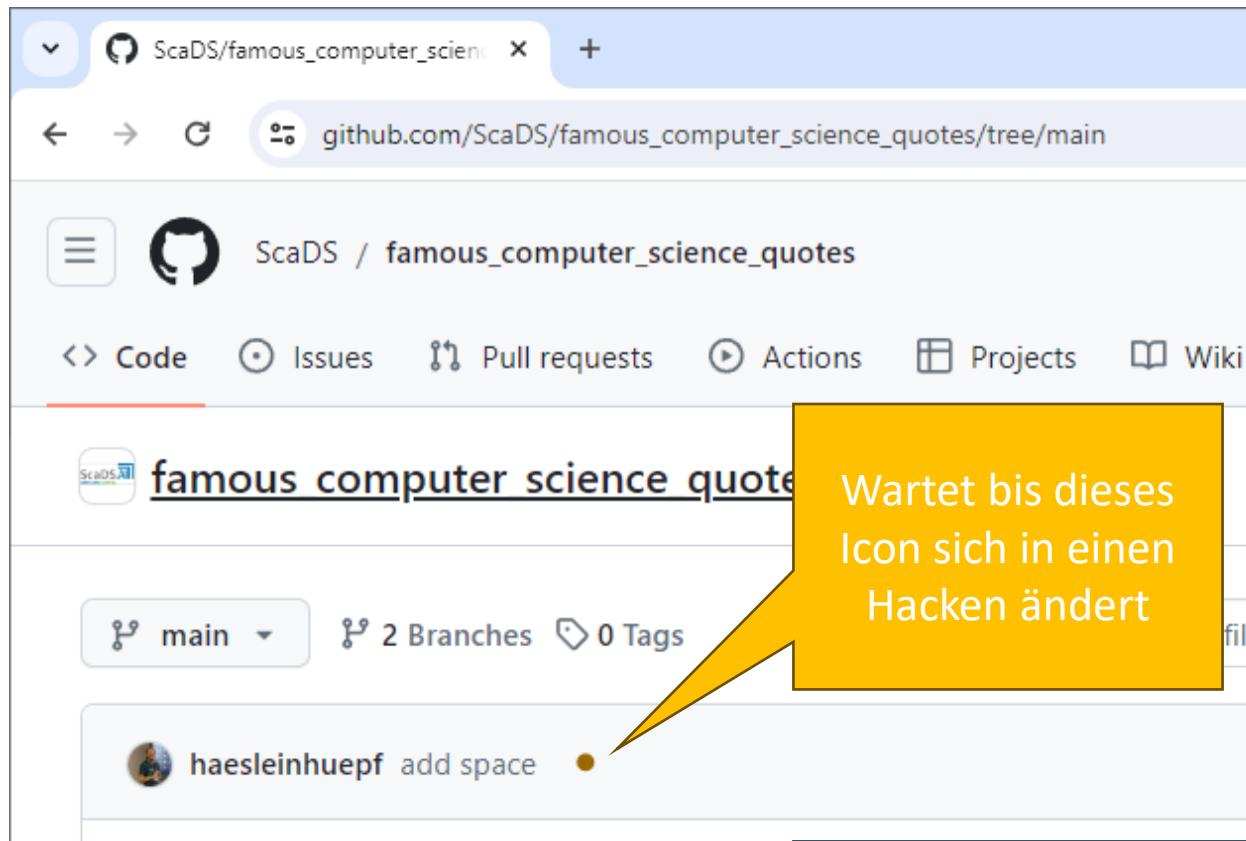
Github pages

- Konfiguration in Github: “gh-pages” bei Pages / Branch auswaehlen



Github pages

- Warten 😊



Das wird automatisch passieren,
wenn man eine Datei ändert

Zenodo integration & Github pages

- In case this was too fast...

The screenshot shows a browser window with the URL coderefinery.github.io/github-without-command-line/doi/#step-2-activate-the-repository-on-zendodo-sandbox. The page title is "Step 2: Activate the repository on Zenodo (sandbox)". On the left, there's a sidebar with "THE LESSON" and "Making your project citable" sections. The main content area has a green header bar with "We will exercise in the Zenodo sandbox". It explains that practice will be on <https://sandbox.zenodo.org/> instead of the "real" <https://zenodo.org/> to avoid creating permanent DOIs. Below this, there are two bullet points:

- Visit <https://sandbox.zenodo.org/account/settings/github/>:
- Select the repository you wish to preserve:

The right side of the screenshot shows a screenshot of the GitHub account settings page for "GitHub Repositories". It features a "Get started" section with three steps: 1. Flip the switch (with a toggle switch), 2. Create a release, and 3. Get the badge. A DOI badge for "10.5281/zenodo.8475 (example)" is shown.



CENTER FOR SCALABLE DATA ANALYTICS AND
ARTIFICIAL INTELLIGENCE



NFDI 4
BIOIMAGE

NATIONAL RESEARCH DATA MANAGEMENT INFRASTRUCTURE
FOR MICROSCOPY AND BIOIMAGE ANALYSIS



GLOBAL BIOIMAGE
ANALYST'S SOCIETY

Praktikum

Robert Haase



Robert Haase
@haesleinhuepf
Collaborative work / git
DataWeek Leipzig
April 15th 2024

These slides can be reused under the terms of the
[CC-BY 4.0](#) license unless mentioned otherwise.

73



TECHNISCHE
UNIVERSITÄT
DRESDEN



UNIVERSITÄT
LEIPZIG

GEFÖRDERT VOM



Bundesministerium
für Bildung
und Forschung



Diese Maßnahme wird gefördert durch die Bundesregierung
aufgrund eines Beschlusses des Deutschen Bundestages.
Diese Maßnahme wird mitfinanziert durch Steuermittel auf
der Grundlage des von den Abgeordneten des Sächsischen
Landtags beschlossenen Haushaltes.

Exercise: Zenodo integration

- Legt eine neue Repository an
 - Ladet die Dateien der Famous Computer Science Quotes Repository hoch
 - Schaltet die Archiv-Funktion in der Zenodo(sandbox) frei
 - Erzeugt einen Release auf Eurer Github Repository

Exercise: Github pages

- Arbeitet weiter in der gleichen Repository
(Es wird nicht mit einem Fork funktionieren)
- Aktiviert die github-pages in den Einstellungen
- Ändert Titel, Logo, Autoren und die Inhalte.
(hint: _config.yml and _toc.yml)

Acknowledgements

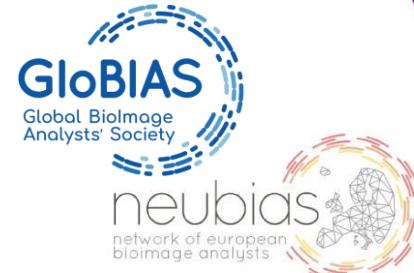
Communities & platforms



NFDI4
BIOIMAGE



image.sc



BiAPoL team

- Marcelo Zoccoler
 - Johannes Soltwedel
 - Maleeha Hassan
 - Stefan Hahmann
- Former lab members:
- Ryan George Savill
 - Laura Zigutyte
 - Mara Lampert
 - Allyson Ryan
 - Conni Wetzker
 - Somashekhar Kulkarni
 - Till Korten



Funding



Bundesministerium
für Bildung
und Forschung



Diese Maßnahme wird gefördert durch die Bundesregierung
aufgrund eines Beschlusses des Deutschen Bundestages.
Diese Maßnahme wird mitfinanziert durch Steuermittel auf
der Grundlage des von den Abgeordneten des Sächsischen
Landtags beschlossenen Haushaltens.

Chan
Zuckerberg
Initiative CZI

DFG

Deutsche
Forschungsgemeinschaft

Some Figures were
generated using



DALL-E