

Kollaboratives Arbeiten und Versionskontrolle mit Git

Robert Haase



GEFÖRDERT VOM



Bundesministerium
für Bildung
und Forschung

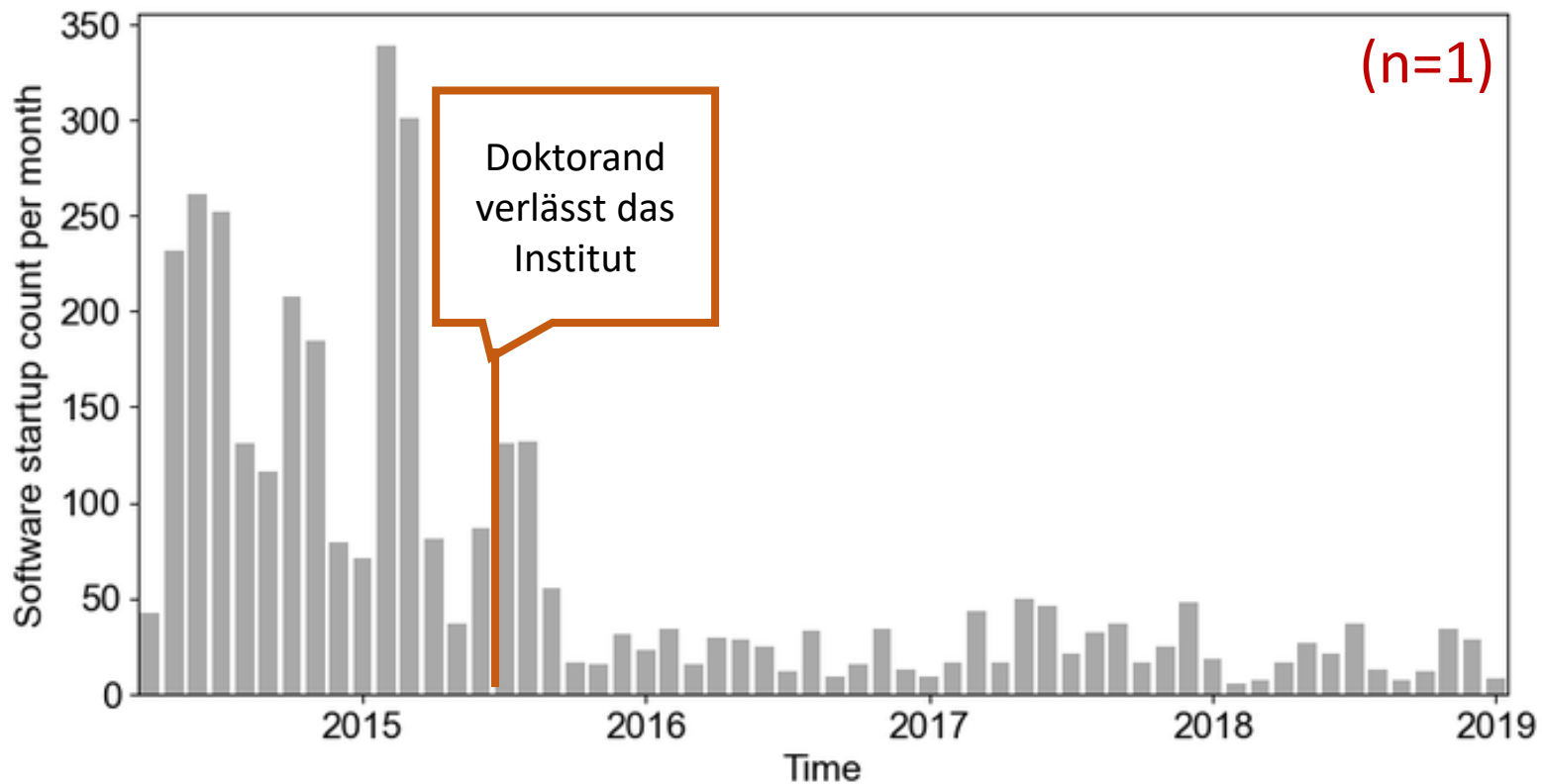
SACHSEN



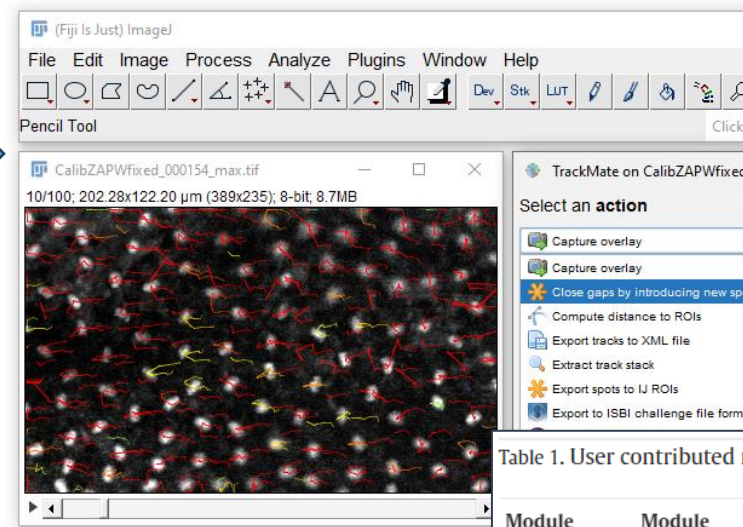
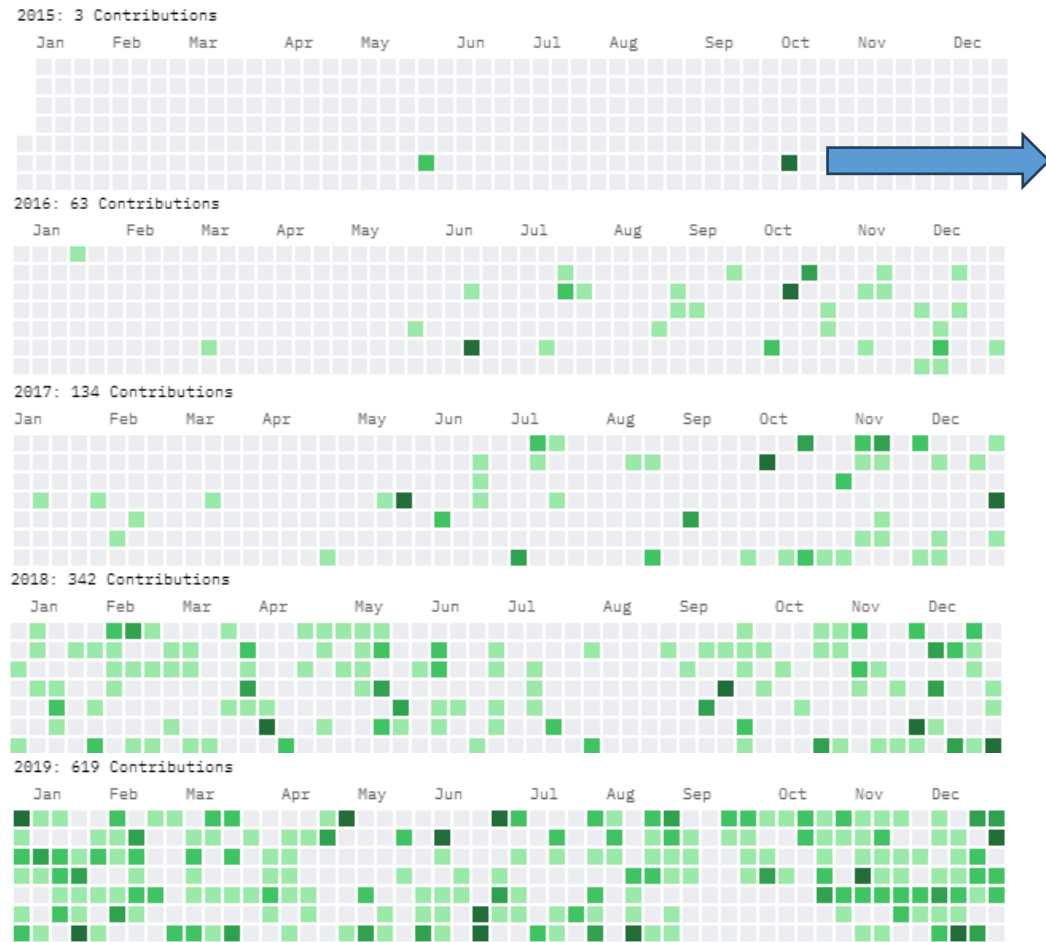
Diese Maßnahme wird gefördert durch die Bundesregierung
aufgrund eines Beschlusses des Deutschen Bundestages.
Diese Maßnahme wird mitfinanziert durch Steuermittel auf
der Grundlage des von den Abgeordneten des Sächsischen
Landtags beschlossenen Haushaltes.

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,1}, Johannes Schindelin^{b,2}, Genevieve M. Hoopes^c, Gregory D. Reynolds^c, Emmanuel Laplantine^d, Sebastian Y. Bednarek^e, Spencer L. Shorte^a, Kevin W. Eliceiri^{b,*}

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

[Add to Mendeley](#) [Share](#) [Cite](#)

[Under a Creative Commons license](#)

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/chiconronny/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

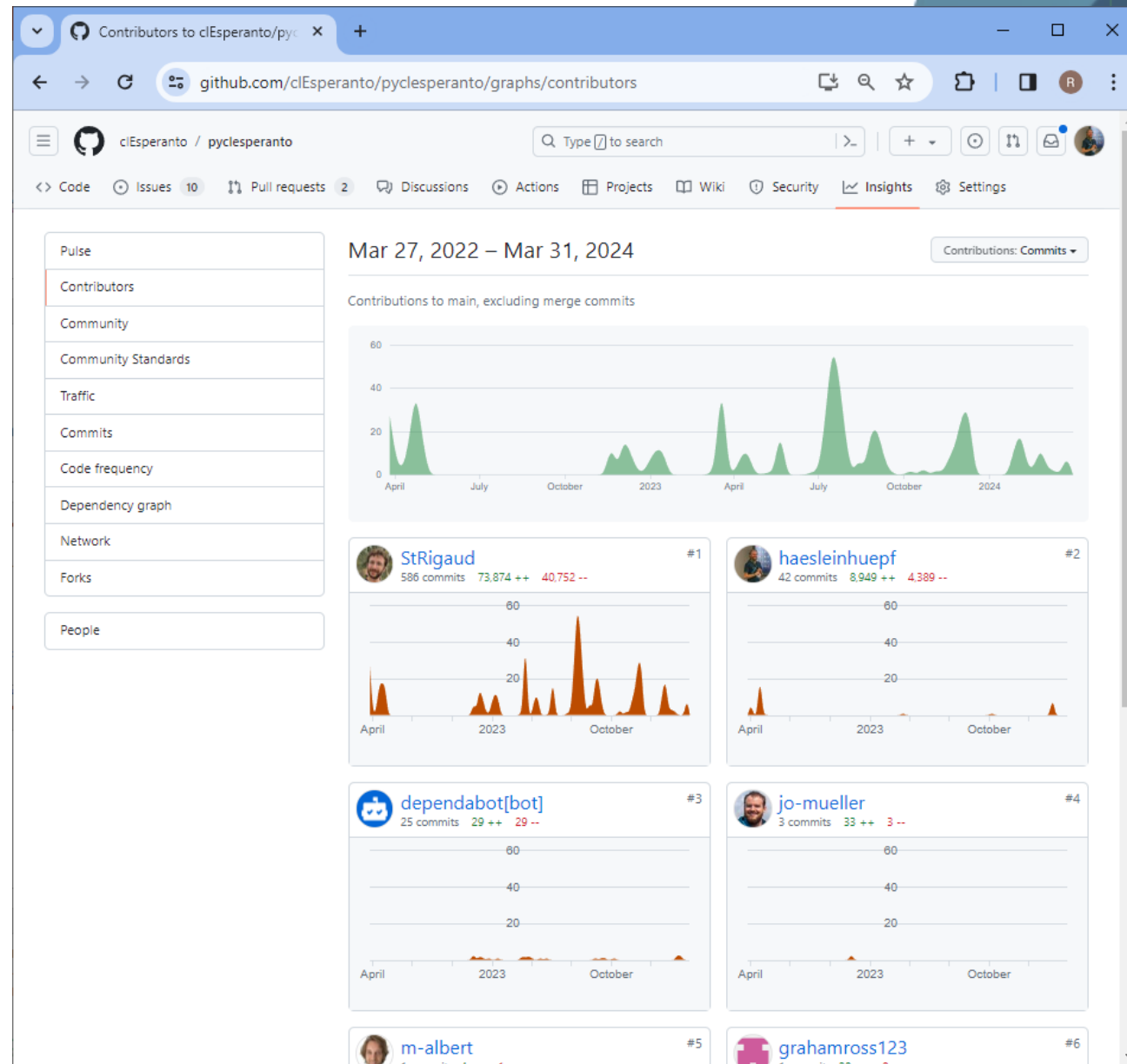


Täglich



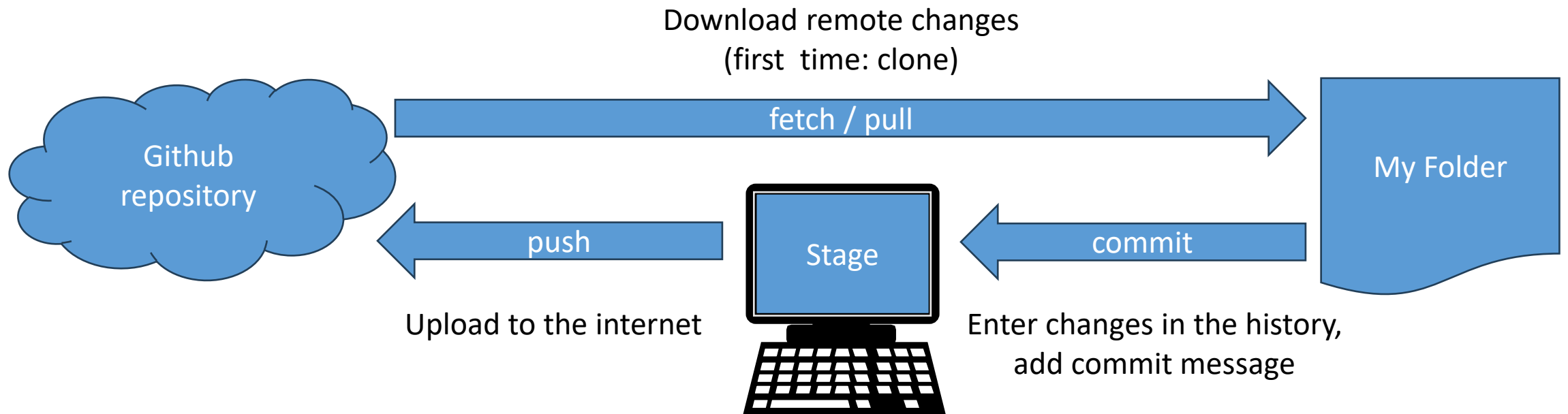
git

- 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 image shows two browser windows from GitHub. The left window displays the commit history for the repository `haesleinhuepf/example_image_analysis_script`. The commit list includes:

- Merge pull request #1 from BiAPoL/main
- bugfix: threshold_otsu** (highlighted with an orange arrow)
- Add minimal working example
- add gitignore

The right window shows the diff for the commit `bugfix: threshold_otsu`. It displays changes to the file `my_library.py`. The diff shows a function `segment_image` with several lines of code. Lines 9 and 10 are highlighted in green, indicating additions:

```
9 + binary = threshold_otsu(blurred)
10 + threshold = threshold_otsu(blurred)
11 + binary = blurred > threshold
```

git

- History
- Track recent changes

The screenshot displays the GitHub interface for the repository 'ScaDS / BIDS-lecture-2024'. The left sidebar shows the commit history for the 'main' branch, with three recent commits by 'haesleinhuepf' on April 12, 2024. The commit titled 'fix issue with ../../ folder locations' is highlighted with a blue arrow. The main content area shows a diff view for this commit, comparing the current state with the previous one. The diff shows two changed files with 16 additions and 14 deletions. The file '02a_remote_files/exploring_bioimage_archive.ipynb' is highlighted, showing changes in the 'base_folder', 'raw_folder', and 'groundtruth_folder' paths. The diff view is split into two panes, showing the original code (left) and the modified code (right).

Commits

main

Commits on Apr 12, 2024

- add backwards compatibility exercise
haesleinhuepf committed yesterday
- move pull-request exercise from week 2 to week 1
haesleinhuepf committed yesterday
- fix issue with ../../ folder locations
haesleinhuepf committed yesterday

Commits on Apr 9, 2024

Showing 2 changed files with 16 additions and 14 deletions.

Whitespace Ignore whitespace Split Unified

28 02a_remote_files/exploring_bioimage_archive.ipynb

↑	@@ -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

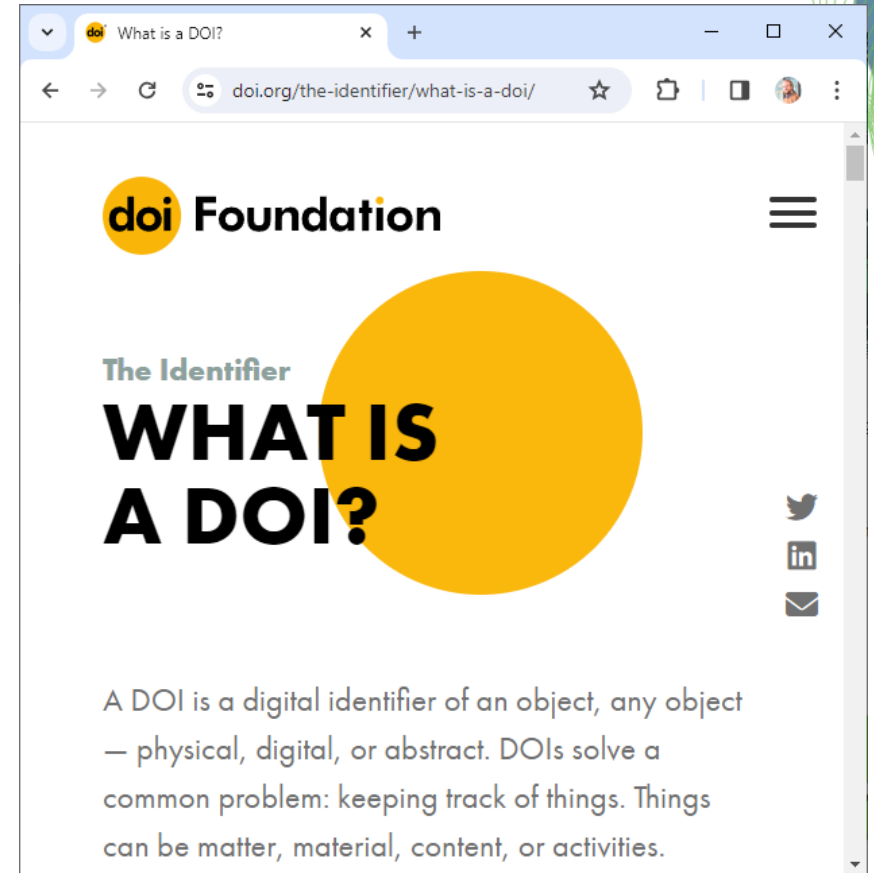
- Findable
- Accessible
- Interoperable
- Reusable



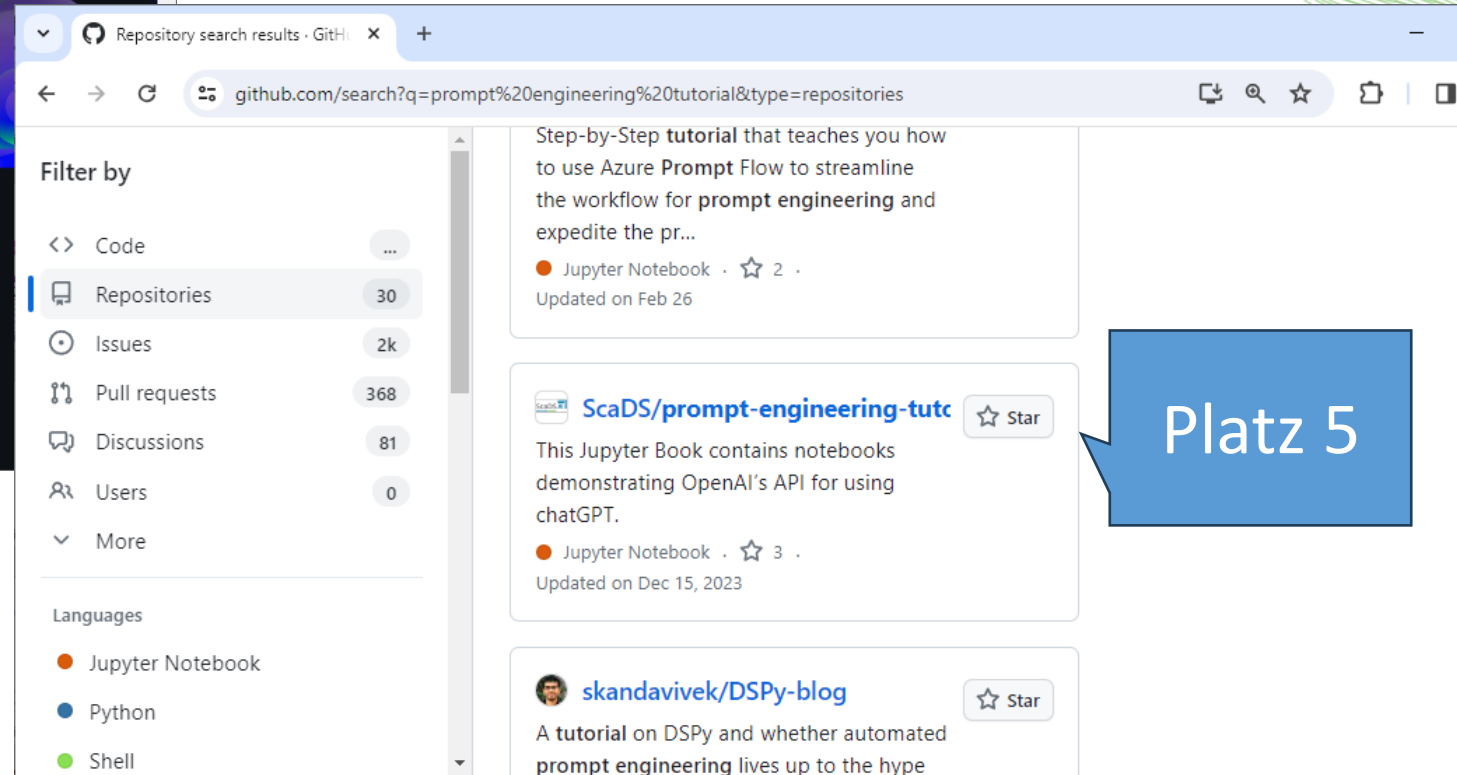
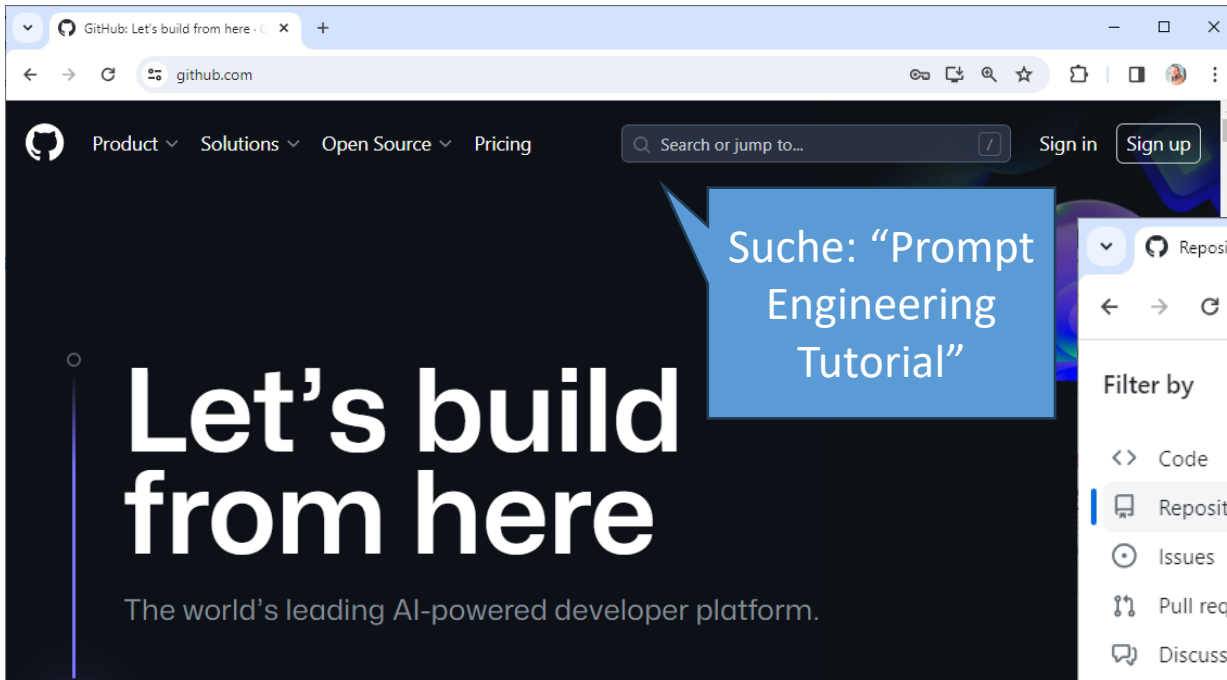
Die FAIR-Prinzipien

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

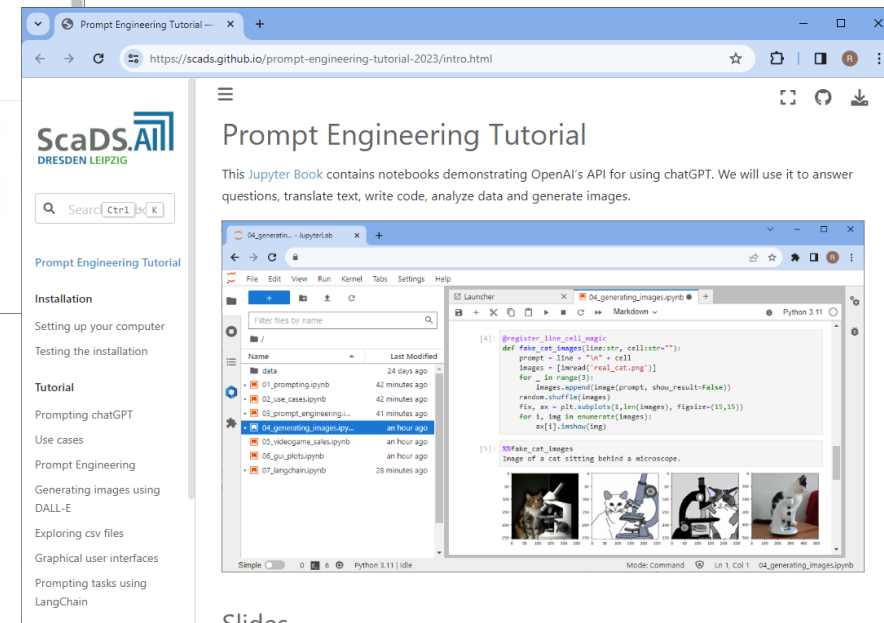
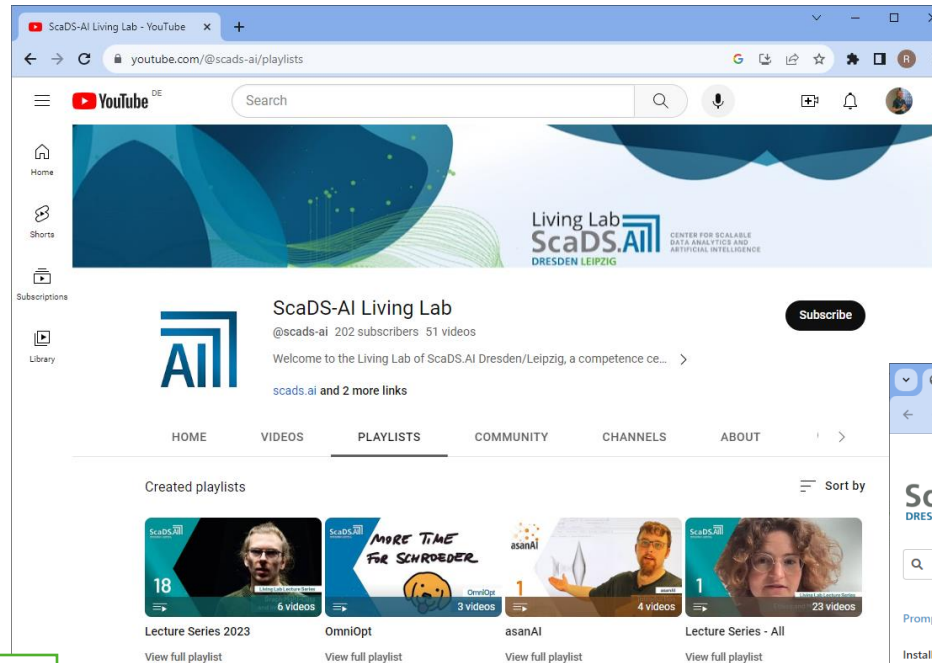


GitHub: Eine durchsuchbare Resource



Nutzen: Findbarkeit -> Sichtbarkeit

- YouTube
- Github



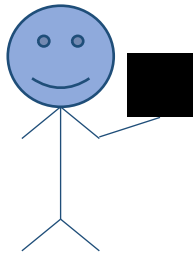
Offenes & FAIRes Teilen
ist ein PR-Instrument

... fuehrt zu

- mehr Softwarenutzer:innen
- neuenw Kollaborationen

Offenheitslevel

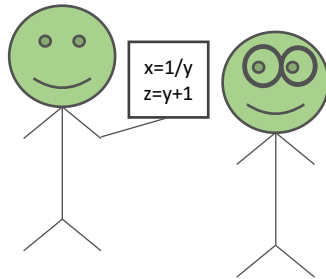
Closed source



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

Hardware device drivers

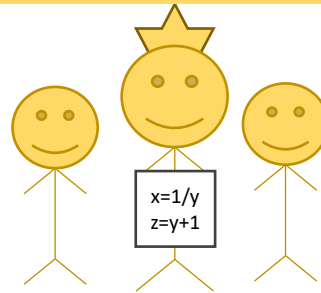
Open source



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

Custom image analysis scripts

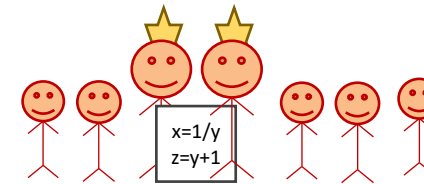
Benevolent dictatorship



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

TrackMate, SNT, MorpholibJ, CLIJ

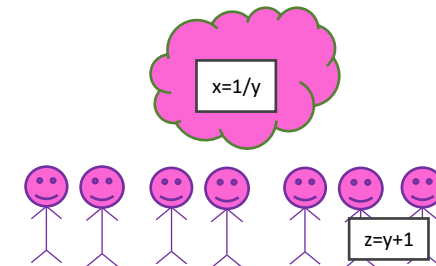
Community driven



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

scikit-image, scipy, OpenCL

Openly extensible

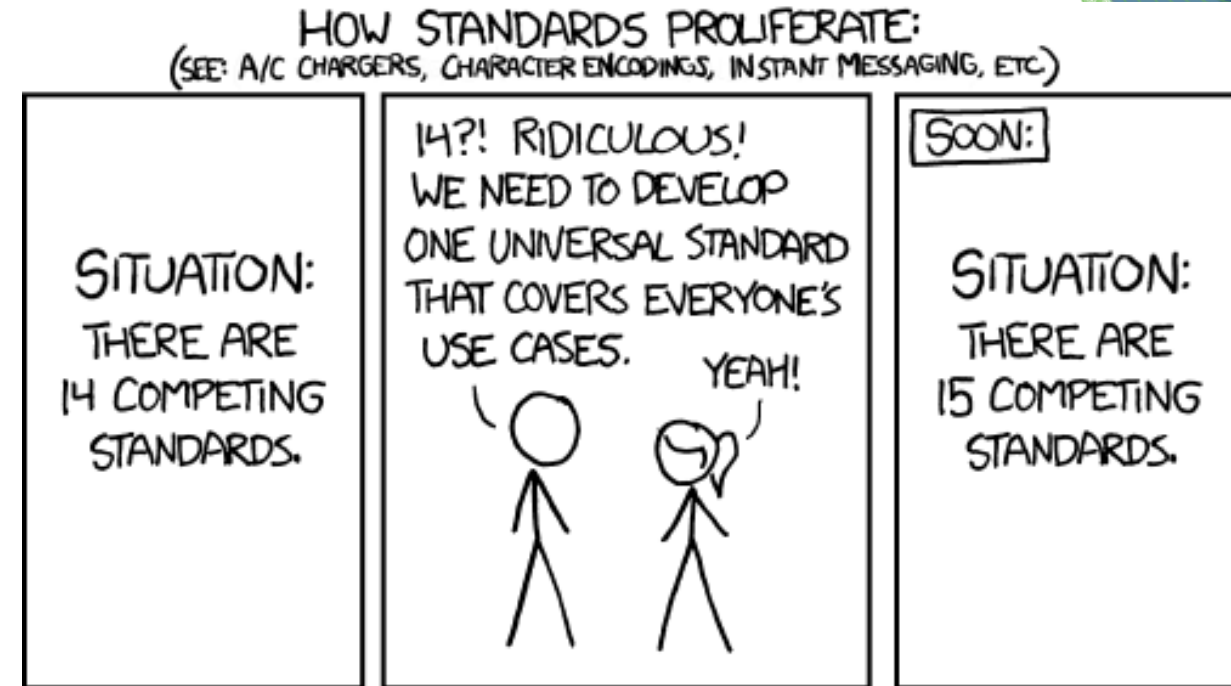


- Openly extensible; without maintainers involved
- Partially community driven

ImageJ, Python, numpy

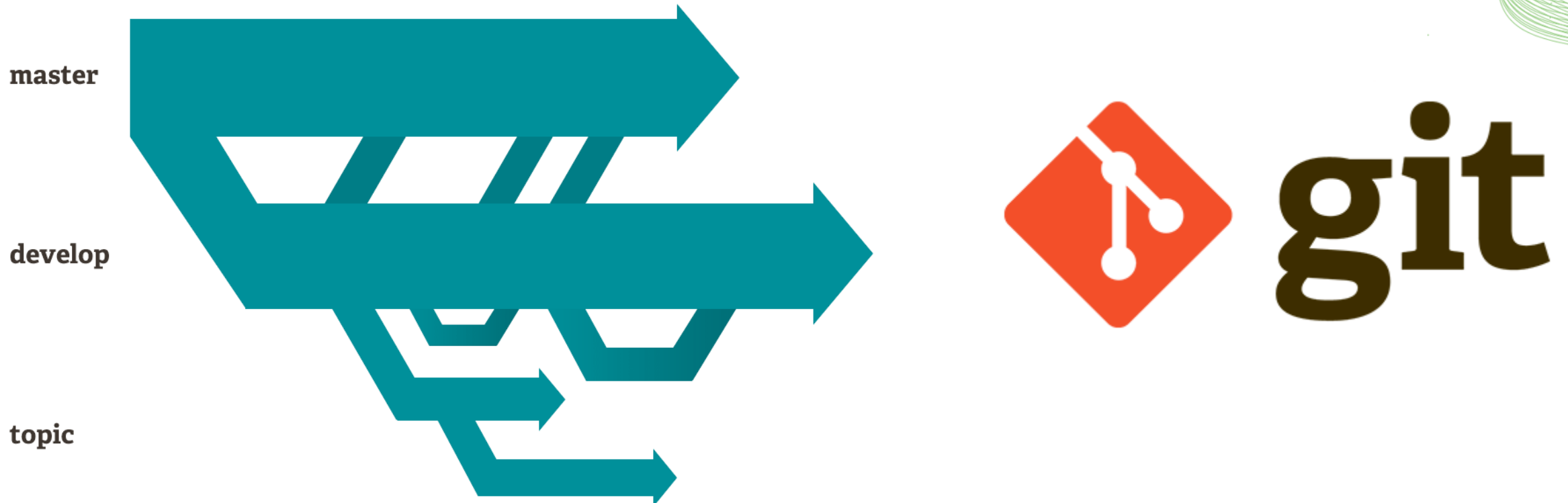
Die FAIR-Prinzipien

- Accessible / Verfügbar / Zugreifbar
- A1. (Meta)daten koennen ueber 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 verfuegbar, auch wenn die Daten selbst nicht mehr verfuegbar sind



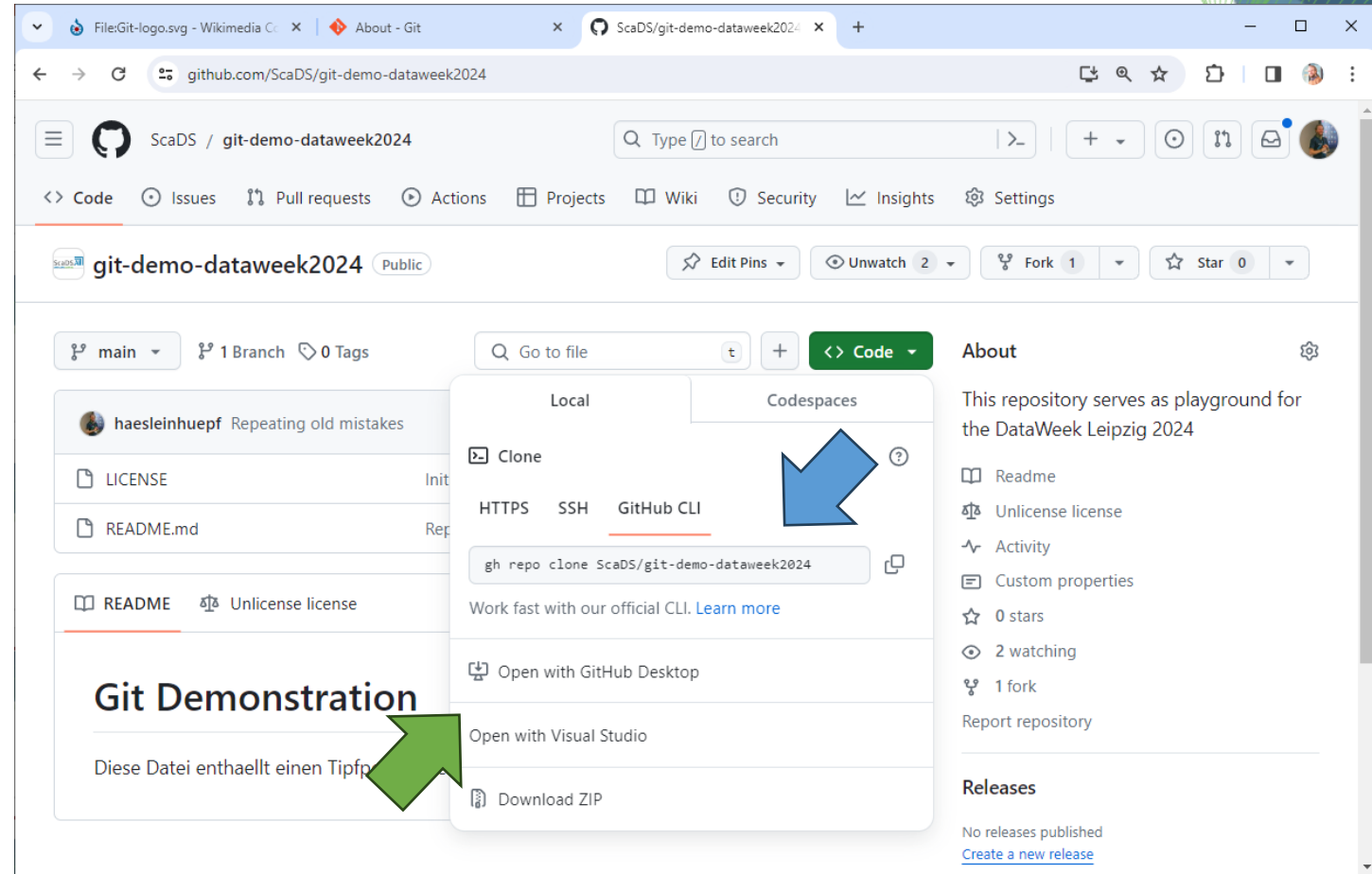
git: standardized communication protocol

- Git nutzt “Branches” um paralleles Arbeiten zu erlauben



Accessibility

- Möglichkeit fuer Menschen und Computer, Daten herunterzuladen



Restricted Access

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

github.com/ScaDS/git-demo-dataweek-private-2024

ScaDS / git-demo-dataweek-private-2024

Code Issues Pull requests Actions Projects Security Insights Settings

git-demo-dataweek-private-2024 **Private** Edit Pins Unwatch 2 Fork 0 Star 0

main 1 Branch 0 Tags

haesleinhuepf added README e0beff8 · now 2 Commits

LICENSE Initial commit 1 minute ago

README.md added README now

README Unlicense license

A private repository

This is a private repository for demonstrating how to give others access to it without sharing it publicly

Readme Unlicense license Activity Custom properties 0 stars 2 watching 0 forks

Releases No releases published Create a new release

github.com/ScaDS/git-demo-dataweek-private-2024/settings/access

General

Who has access

Access

Collaborators and teams

Code and automation

- Branches
- Tags
- Rules
- Actions
- Webhooks
- Pages
- Custom properties

Security

- Code security and analysis
- Deploy keys
- Secrets and variables

Integrations

- GitHub Apps

PRIVATE REPOSITORY (locked)

Only those with access to this repository can view it. [Manage](#)

BASE ROLE (Read)

All 15 members can access this repository. [Manage](#)

DIRECT ACCESS

0 teams or members have access to this repository. Only [Owners](#) can contribute to this repository.

[Create team](#)

Manage access

You haven't added any teams or people yet

Organization owners can manage individual and team access to the organization's repositories. Team maintainers can also manage a team's repository access. [Learn more about organization access](#)

[Add people](#) [Add teams](#)

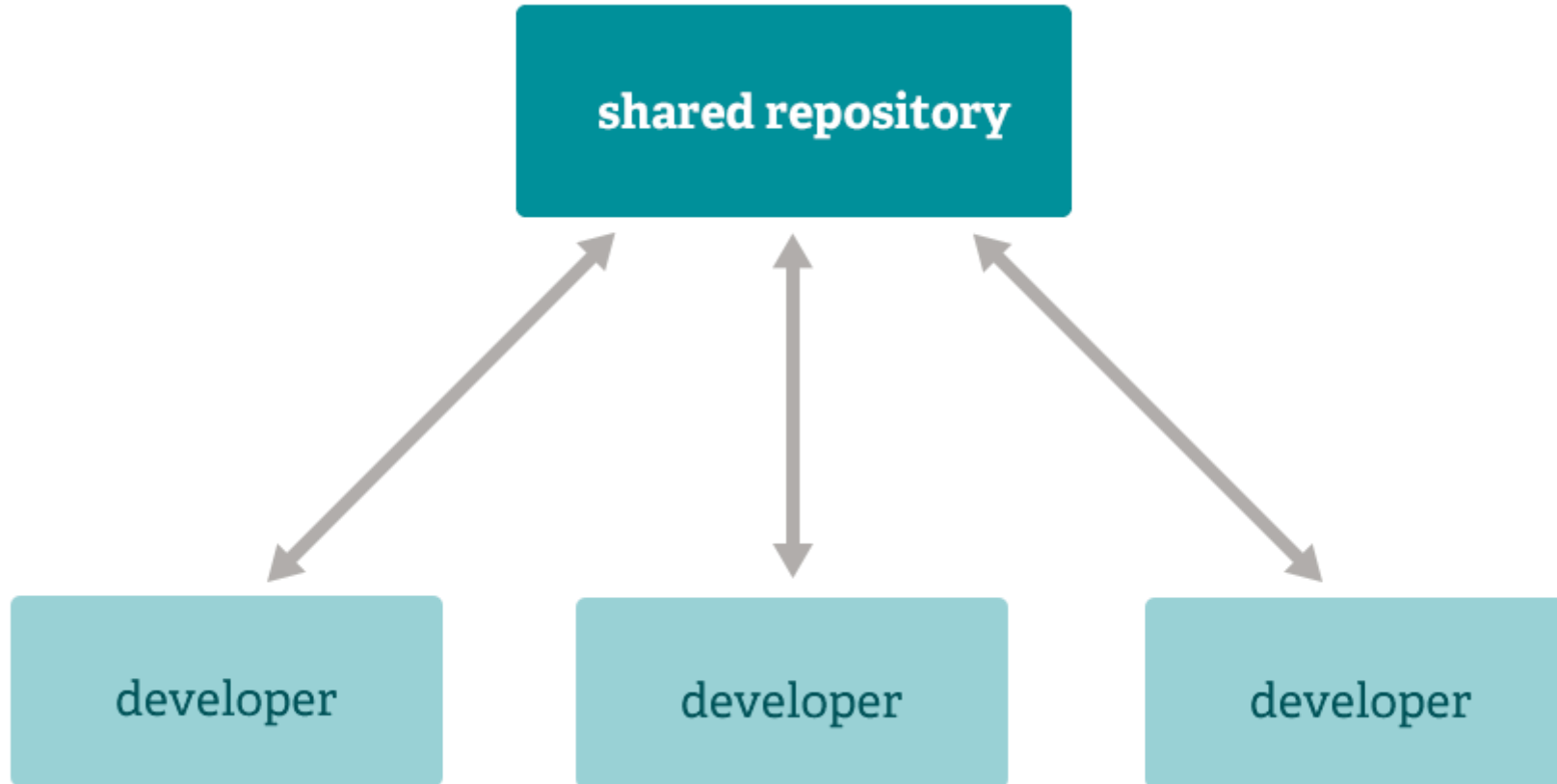
Die FAIR-Prinzipien

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



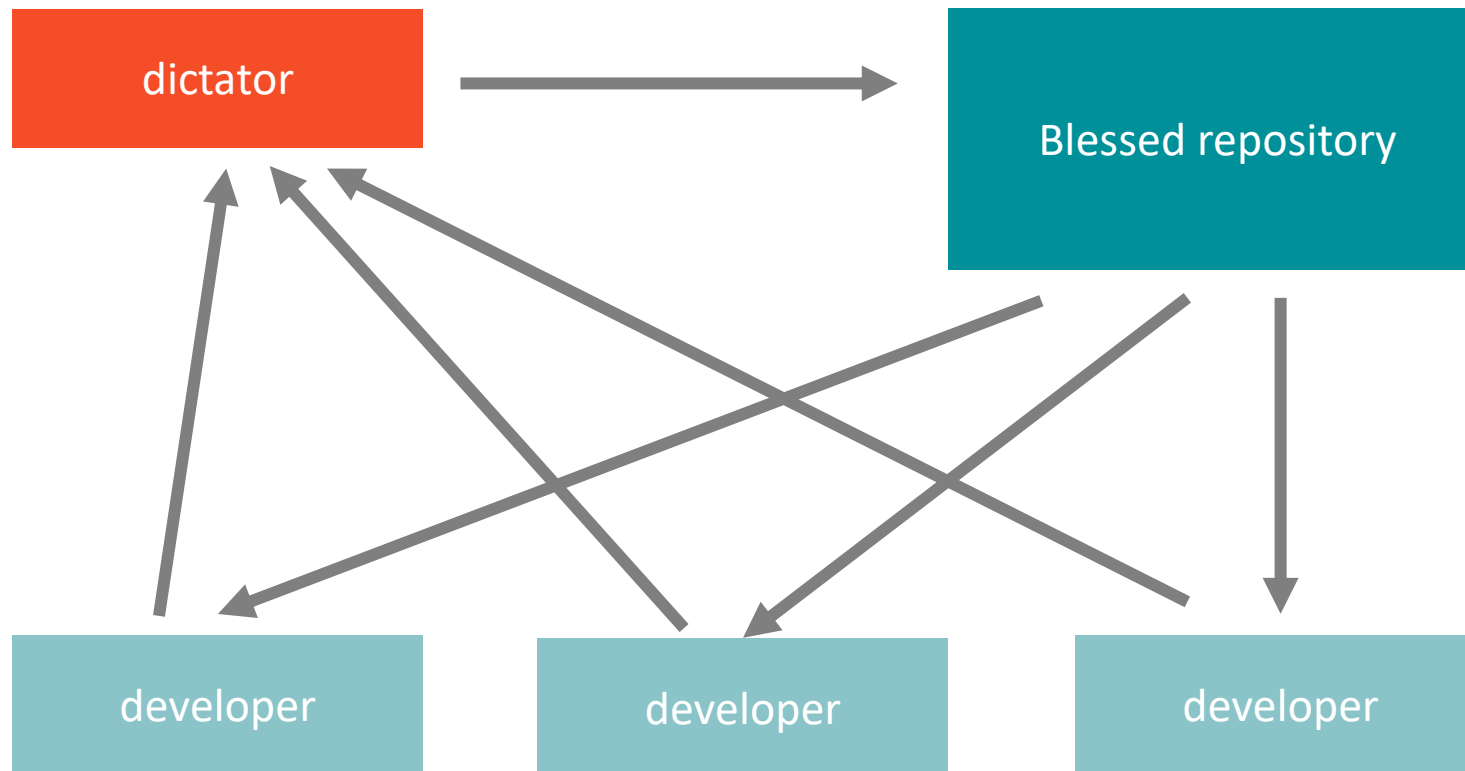
git: Workflows

- Subversion-style



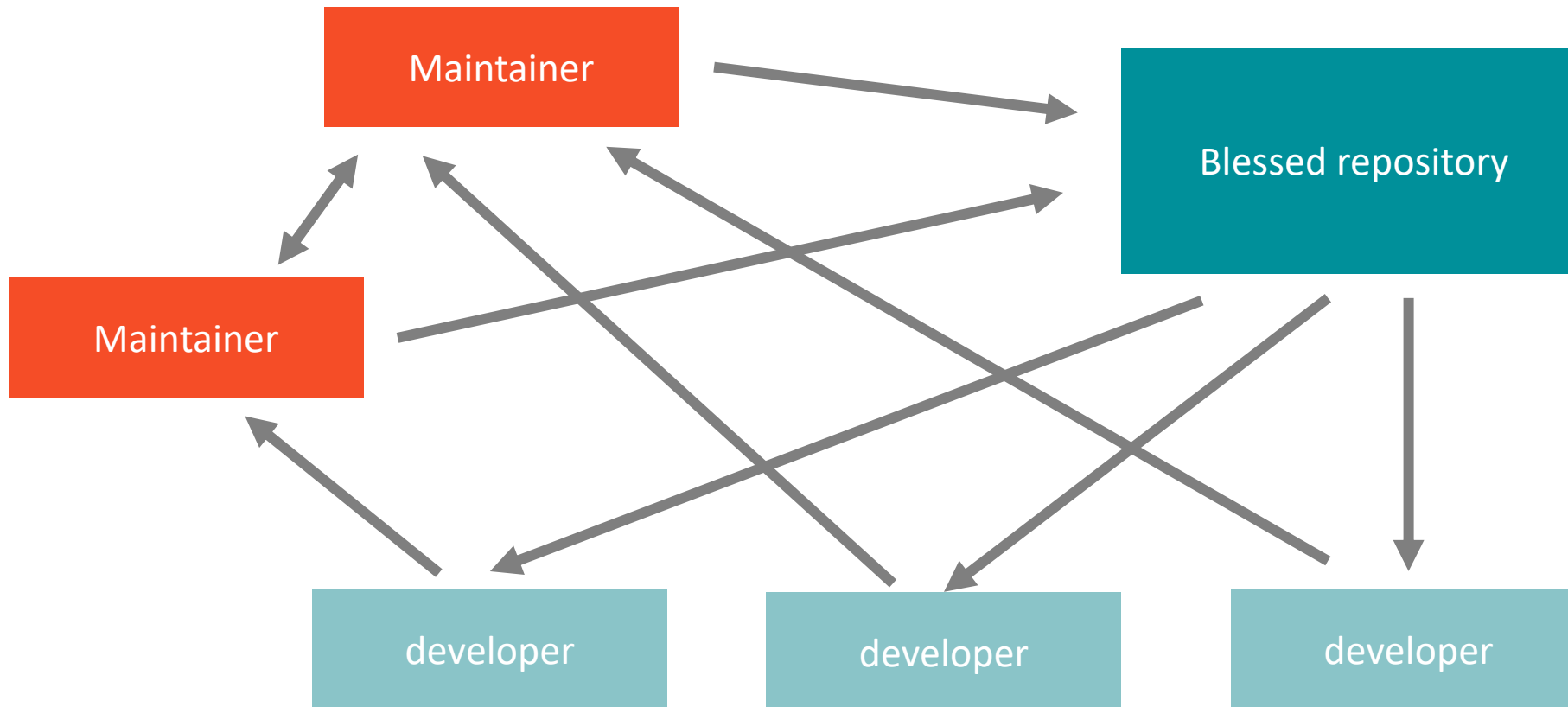
git: Workflows

- Benevolent dictatorship workflow



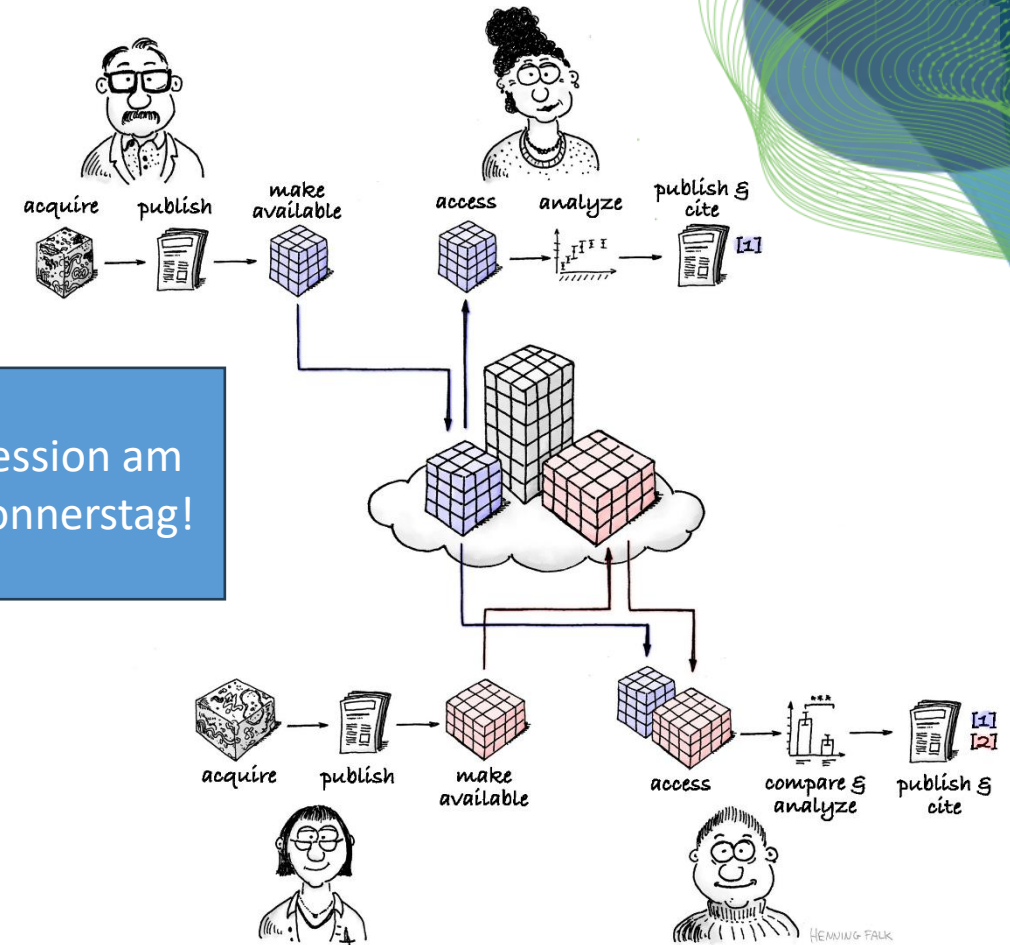
git: Workflows

- Joint maintainers (may implement some form of democracy)



The FAIR-principles

- Reusable / Wiederverwendbar
- R1. (Meta)daten sind reich an vielfaeltigen, akkuraten und relevanten Attributen
- R1.1. (Meta)daten werden mit einer klaren und verfuegbaren 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

- Zenodo:
archiving / citing

The screenshot shows the GitLab interface for the repository 'rhaase / NEUBIAS_Academy_CLI2'. It displays the repository name, commit count (11), and a table of recent commits. The most recent commit is 'added cheat sheet as pdf' by rhaase, dated 3 years ago.

Name	Letzter Commit	Letzte Aktualisierung
data	moved Lund dataset to clij2-exam...	vor 3 Jahren
CLIJ2-cheatsheet_V3.pdf	added cheat sheet as pdf	vor 3 Jahren
CLIJ2-cheatsheet_V3.pptx	initial version of slides	vor 3 Jahren
CLIJ2_GPU_accelerated...	updated slides and exercises	vor 3 Jahren
CLIJ2_GPU_accelerated...	updated slides and exercises	vor 3 Jahren
LICENSE	added license	vor 3 Jahren

The screenshot shows the GitHub interface for the repository 'haesleinhuepf / stackview'. It displays the repository name, commit count (209), and a list of files. The most recent commit is 'fixed documentation' by haesleinhuepf, dated last week.

File Name	Description	Last Updated
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

The screenshot shows the Zenodo record page for the version 0.7.6 of the 'haesleinhuepf/stackview' repository. It displays the publication date (March 30, 2024), the version number (0.7.6), and the authors (Robert Haase, Jordão Bragantini, Oren Amsalem). The page also includes a 'What's Changed' section and a 'Full Changelog' link.

haesleinhuepf/stackview: 0.7.6
Published March 30, 2024 | Version 0.7.6

Robert Haase¹; Jordão Bragantini²; Oren Amsalem

What's Changed

- add imshow (moved over, and updated from pyclesperanto_prototype) by @haesleinhuepf in <https://github.com/haesleinhuepf/stackview/pull/37>
- fix deprecation warning in switch by @haesleinhuepf in <https://github.com/haesleinhuepf/stackview/pull/38>

Full Changelog: <https://github.com/haesleinhuepf/stackview/compare/0.7.5...0.7.6>

Zenodo

- öffentlich geförderte Infrastruktur @ CERN / Schweiz

The image shows two overlapping browser windows of the Zenodo website. The left window displays the 'Featured communities' section, highlighting the 'European Climate and Modelling Forum' (ECMF) with a colorful abstract logo and a 'Browse' button. The right window shows the site's navigation menu, which includes links for 'About', 'Blog', 'Help', 'Developer s', 'Contribute', and 'Funded by'. The 'Funded by' section features logos for CERN, OpenAIRE, and the European Union. At the bottom of the right window, there are links for 'Status', 'Privacy policy', 'Cookie policy', 'Terms of Use', and 'Support', along with the text 'Powered by CERN Data Centre & InvenioRDM'.

Quiz

- Wo ist Open-Source Software am besten sichtbar?

Git server an der
Universität



Zenodo.org



Github.com

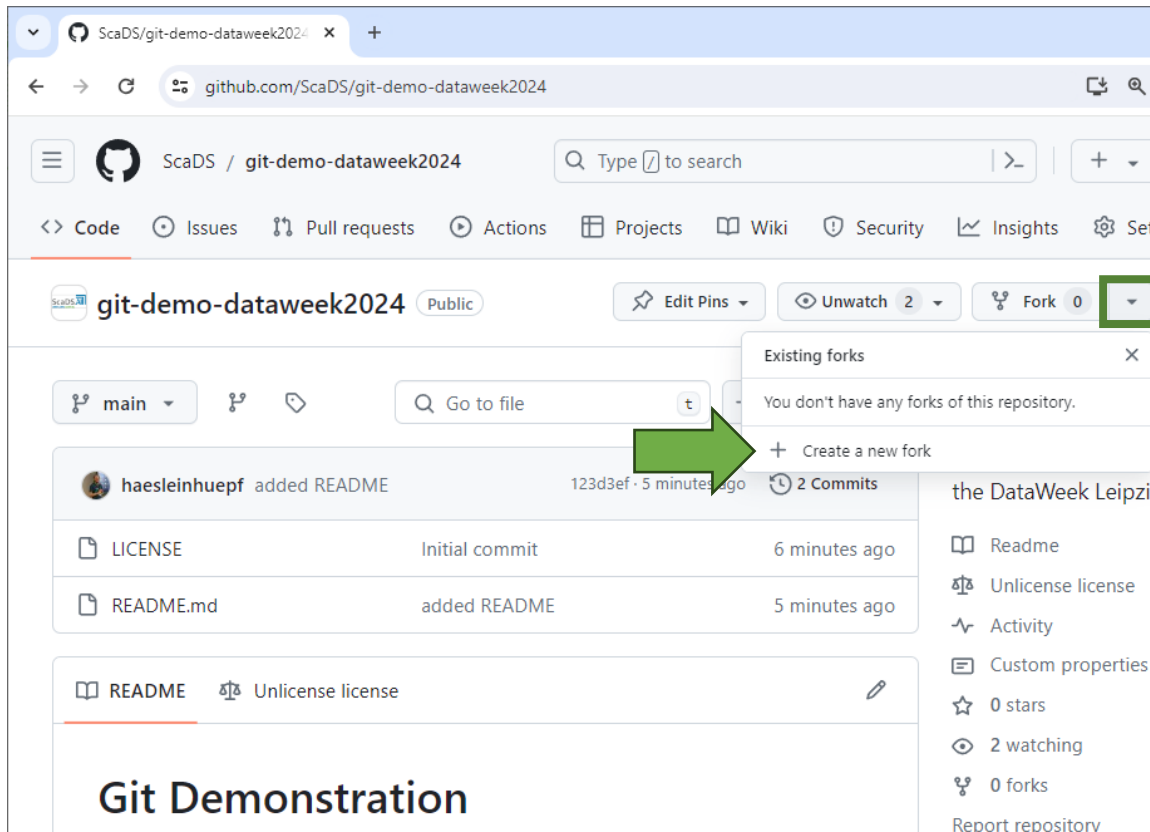


Firmen- / Instituts-
Webseite

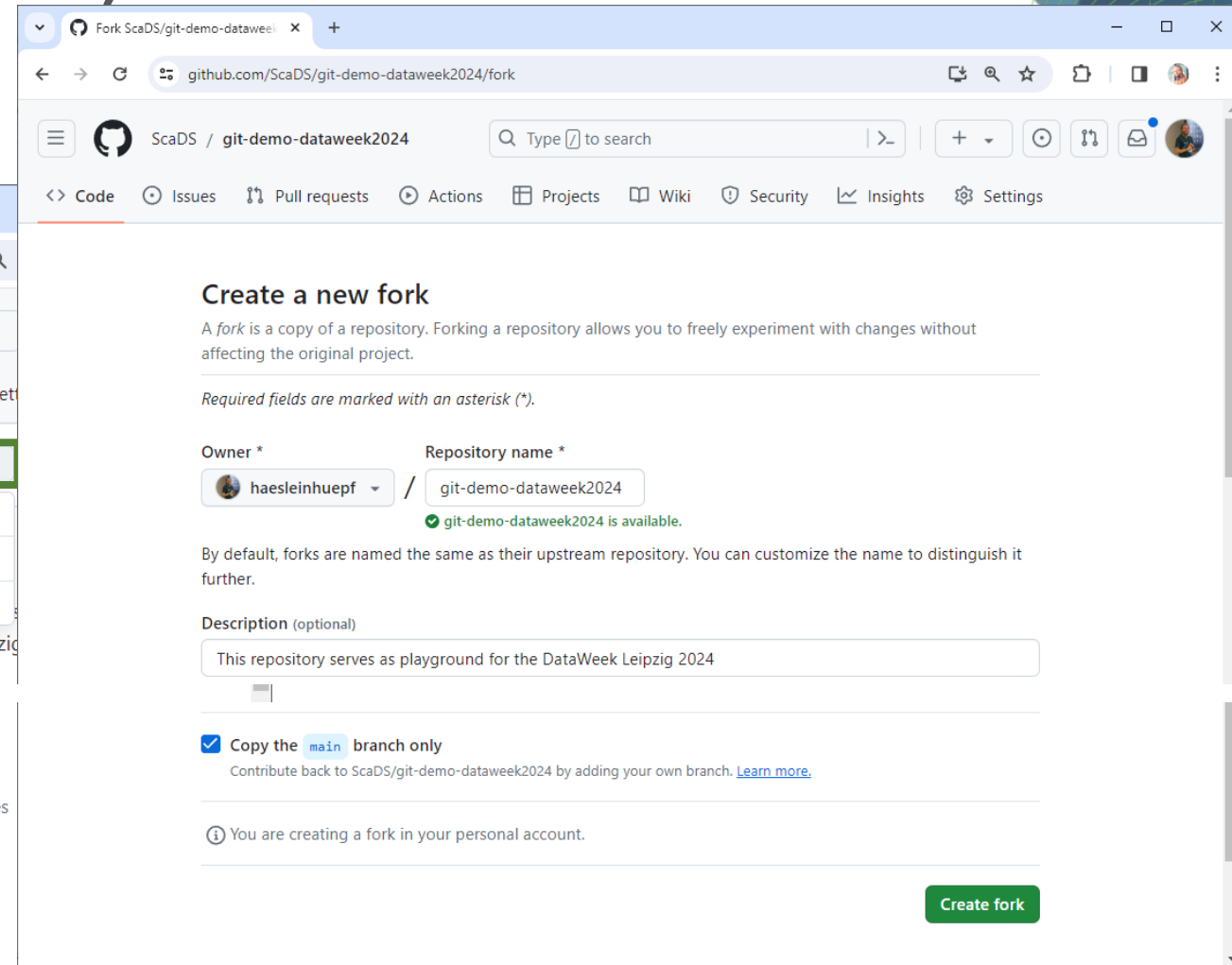


Quick start: git(hub)

- *Forken* einer repository



The screenshot shows the GitHub repository page for 'ScaDS / git-demo-dataweek2024'. The 'Fork' button is highlighted with a green box, and a green arrow points to it. The repository details show it is public, has 0 forks, and 2 commits. The commit history is visible, showing the initial commit and the addition of the README file. The repository title is 'Git Demonstration'.



The screenshot shows the 'Create a new fork' form on GitHub. The form is titled 'Create a new fork' and includes a description: 'A fork is a copy of a repository. Forking a repository allows you to freely experiment with changes without affecting the original project.' The form fields are: Owner (haesleinhuepf), Repository name (git-demo-dataweek2024), and Description (optional) (This repository serves as playground for the DataWeek Leipzig 2024). There is a checkbox for 'Copy the main branch only' which is checked. A green 'Create fork' button is at the bottom right.

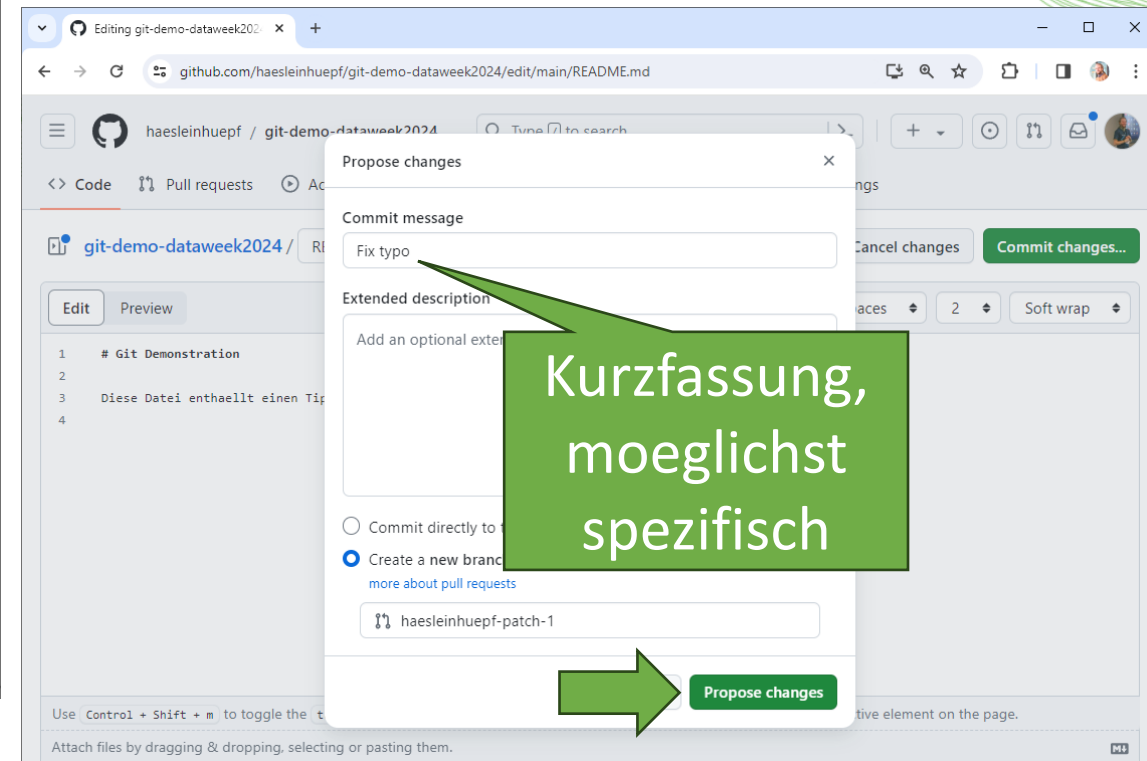
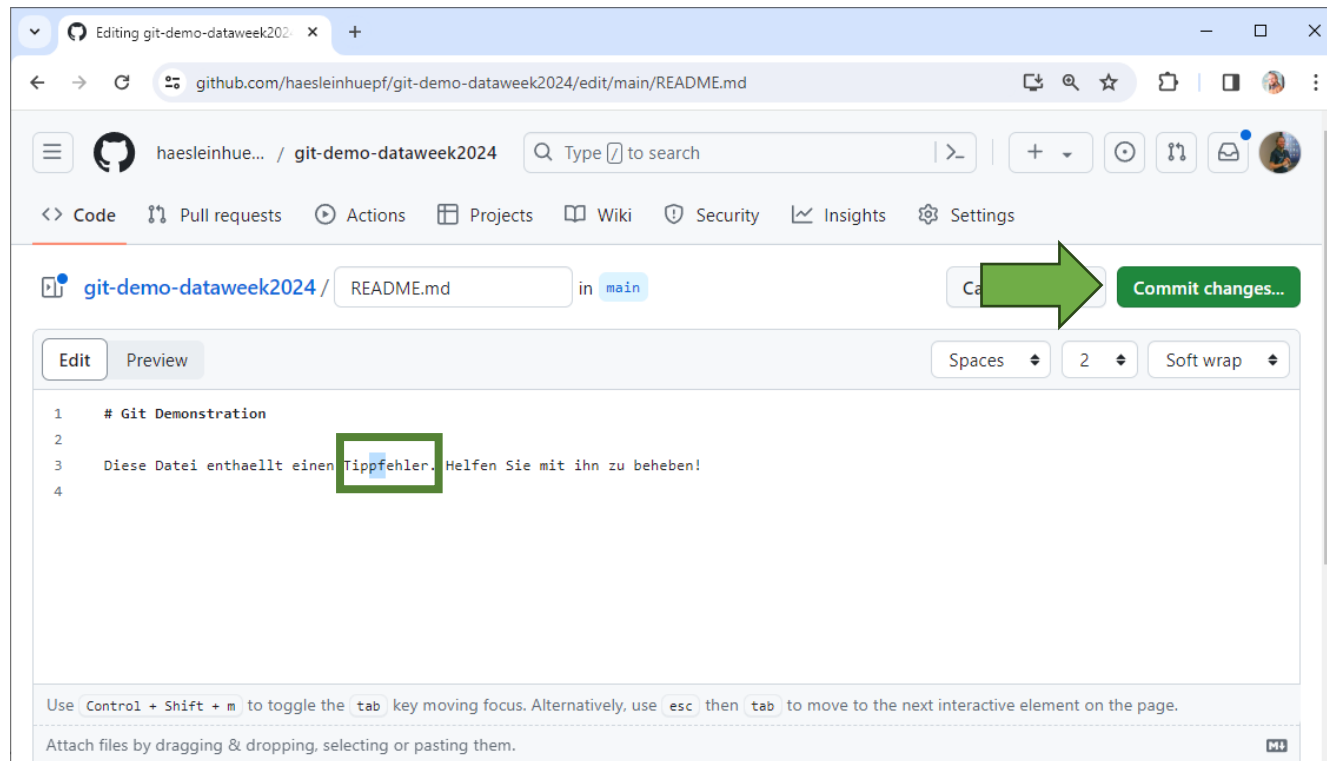
Quick start: git(hub)

- In einen Fork haben wir Schreibrechte...

The image shows two browser windows side-by-side. The left window displays the GitHub repository page for 'git-demo-dataweek2024', which is a fork of 'ScaDS/git-demo-dataweek2024'. The repository is public and has 0 stars, 0 watches, and 0 forks. The file list shows three files: README (added 8 minutes ago), LICENSE (initial commit, 9 minutes ago), and README.md (added 8 minutes ago). The right window shows the same repository but with the README file selected. The README content is 'Git Demonstration' and 'Diese Datei enthaelt einen Tipfpehler. Helfen Sie mit ihn zu beheben!'. A green box highlights the edit icon in the top right corner of the README preview.

Quick start: git(hub)

- In einen Fork haben wir Schreibrechte...



Quick start: git(hub)

- *Pull-requests* sind der Mechanismus zur Zusammenarbeit

Comparing changes

Choose two branches to see what's changed or to start a new pull request. If you need to, you can also [compare across forks](#) or [learn about comparisons](#).

base: main ← compare: haesleinhuepf-patch-1 ✓ Able to merge. These branches can be automatically merged.

Discuss and review the changes in this comparison with others. [Learn about pull requests](#)

1 commit 1 file changed

Commits on Apr 6, 2024

Fix typo
haesleinhuepf committed now

Showing 1 changed file with 1 addition and 1 deletion.

so compare across forks

Comparing changes

Choose two branches to see what's changed or to start a new pull request. If you need to, you can also [compare across forks](#) or [learn about comparisons](#).

base repository: haesleinhuepf/git-demo-dataweek2024 base: main ← compare: haesleinhuepf-patch-1

Choose a Base Repository

Filter repos

ScaDS/git-demo-dataweek2024

haesleinhuepf/git-demo-dataweek2024

Discuss and review the changes in this comparison with others. [Learn about pull requests](#)

1 commit 1 file changed 1 contributor

Commits on Apr 6, 2024

Fix typo

Comparing ScaDS/main...haesleinhuepf/git-demo-dataweek2024

base repository: ScaDS/git-demo-dataweek2024 head repository: haesleinhuepf/git-demo-dataweek2024 ✓ Able to merge. These branches can be automatically merged.

Add a title

Fix typo

Add a description

Write Preview H B I

Hi Robert,
this just fixes a typo.
Best,
Robert

Markdown is supported Paste, drop, or click to add files

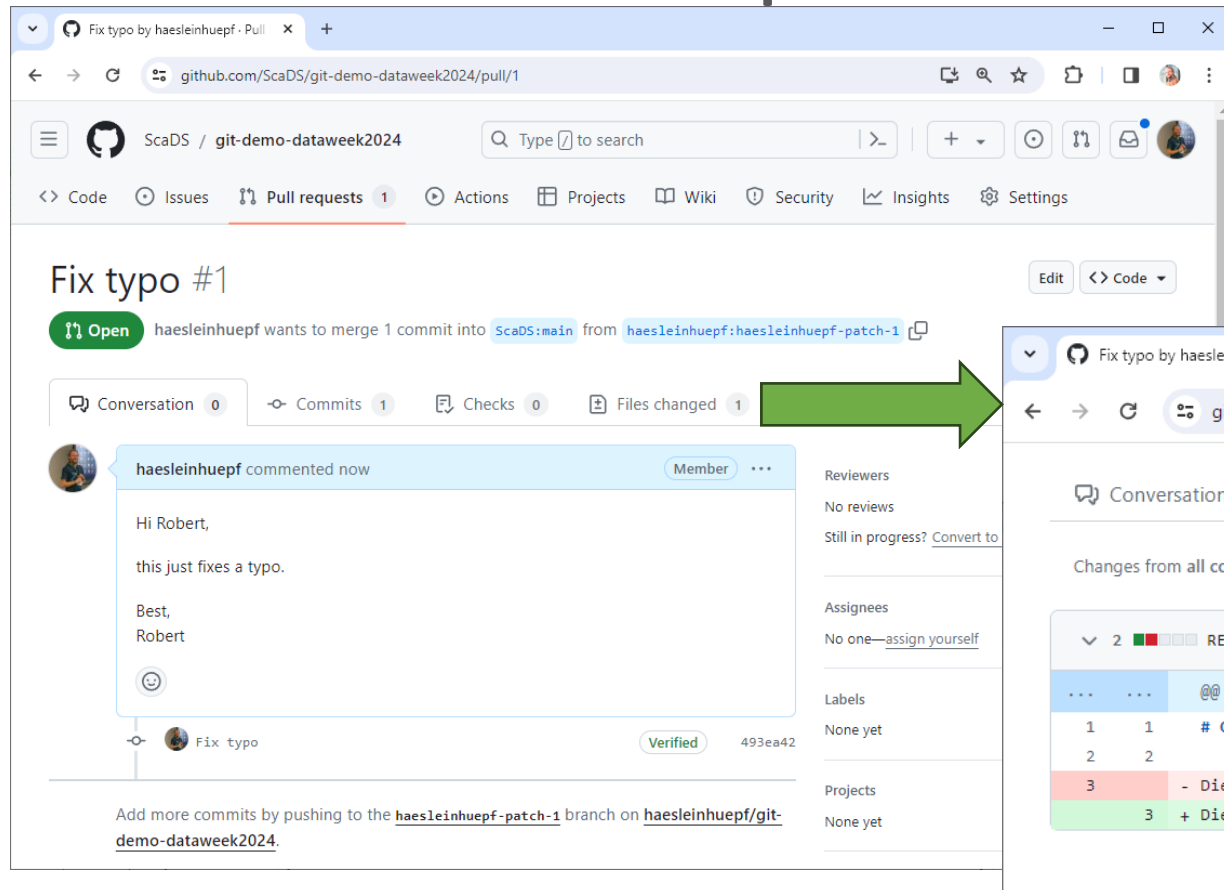
Allow edits to this pull request

Create pull request

Freundliche Nachricht

Quick start: git(hub)

• Reviewer-Perspektive



Fix typo by haesleinhuepf · Pull

github.com/ScaDS/git-demo-dataweek2024/pull/1

ScaDS / git-demo-dataweek2024

Fix typo #1

haesleinhuepf wants to merge 1 commit into ScaDS:main from haesleinhuepf:haesleinhuepf-patch-1

Conversation 0 Commits 1 Checks 0 Files changed 1

haesleinhuepf commented now

Hi Robert,

this just fixes a typo.

Best,
Robert

Fix typo

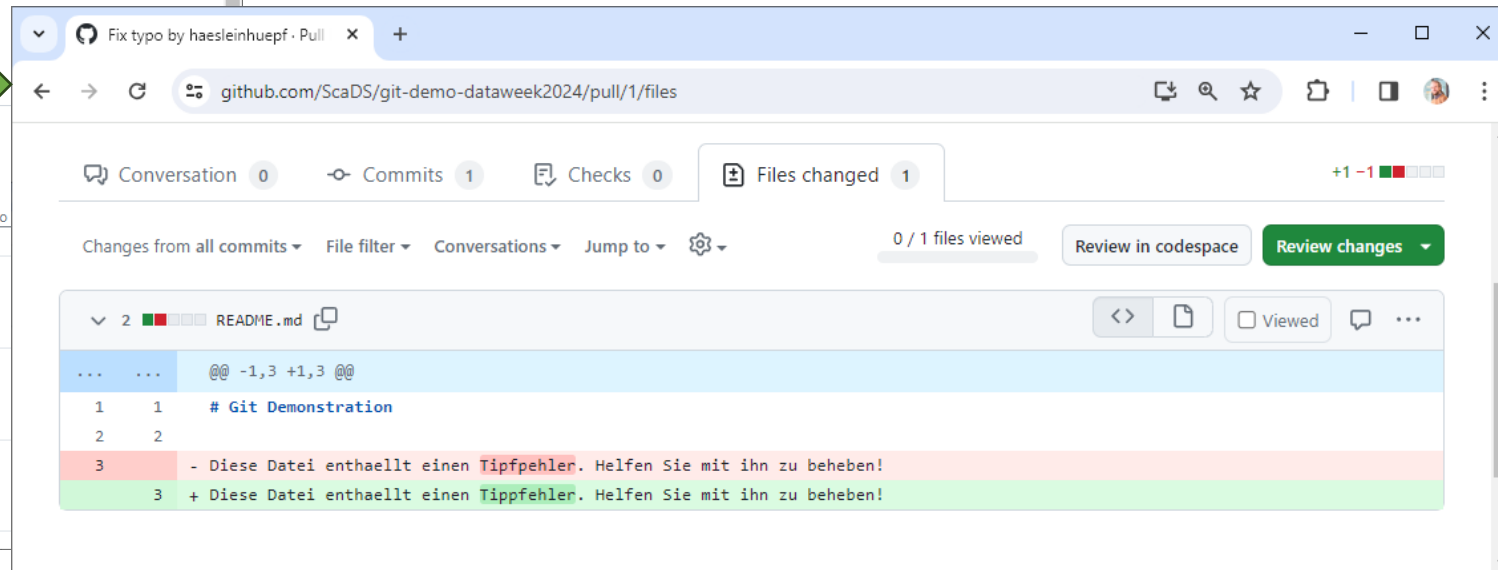
Reviewers: No reviews

Assignees: No one—assign yourself

Labels: None yet

Projects: None yet

Add more commits by pushing to the haesleinhuepf-patch-1 branch on haesleinhuepf/git-demo-dataweek2024.



Fix typo by haesleinhuepf · Pull

github.com/ScaDS/git-demo-dataweek2024/pull/1/files

Conversation 0 Commits 1 Checks 0 Files changed 1

Changes from all commits File filter Conversations Jump to

0 / 1 files viewed Review in codespace Review changes

2 README.md

```
@@ -1,3 +1,3 @@  
1 1 # Git Demonstration  
2 2  
3 - Diese Datei enthaellt einen Tippfehler. Helfen Sie mit ihn zu beheben!  
3 + Diese Datei enthaellt einen Tippfehler. Helfen Sie mit ihn zu beheben!
```

Quick start: git(hub)

• Reviewer-Perspektive

The image displays three overlapping screenshots of a GitHub pull request interface, illustrating the reviewer's perspective during the merge process.

- Left Screenshot:** Shows the pull request details for "Fix typo #1". It includes a status bar indicating "Continuous integration has not been set up" and "This branch has no conflicts with the base branch". A prominent green "Merge pull request" button is visible.
- Middle Screenshot:** Shows the "Merge pull request #1" dialog box. The commit message "Fix typo" is visible, and the "Confirm merge" button is highlighted.
- Right Screenshot:** Shows the pull request after it has been successfully merged and closed. A message states: "Pull request successfully merged and closed. You're all set — the haesleinhuepf:haesleinhuepf-patch-1 branch can be safely deleted." A "Delete branch" button is present. Below this, a comment from the reviewer is visible: "Hi Robert, that's awesome. Big Thanks! Best, Robert".

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.

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



Kollaboratives Arbeiten und Versionskontrolle mit Git II

Robert Haase

GEFÖRDERT VOM



Bundesministerium
für Bildung
und Forschung



SACHSEN

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.

Github – Repositories anlegen

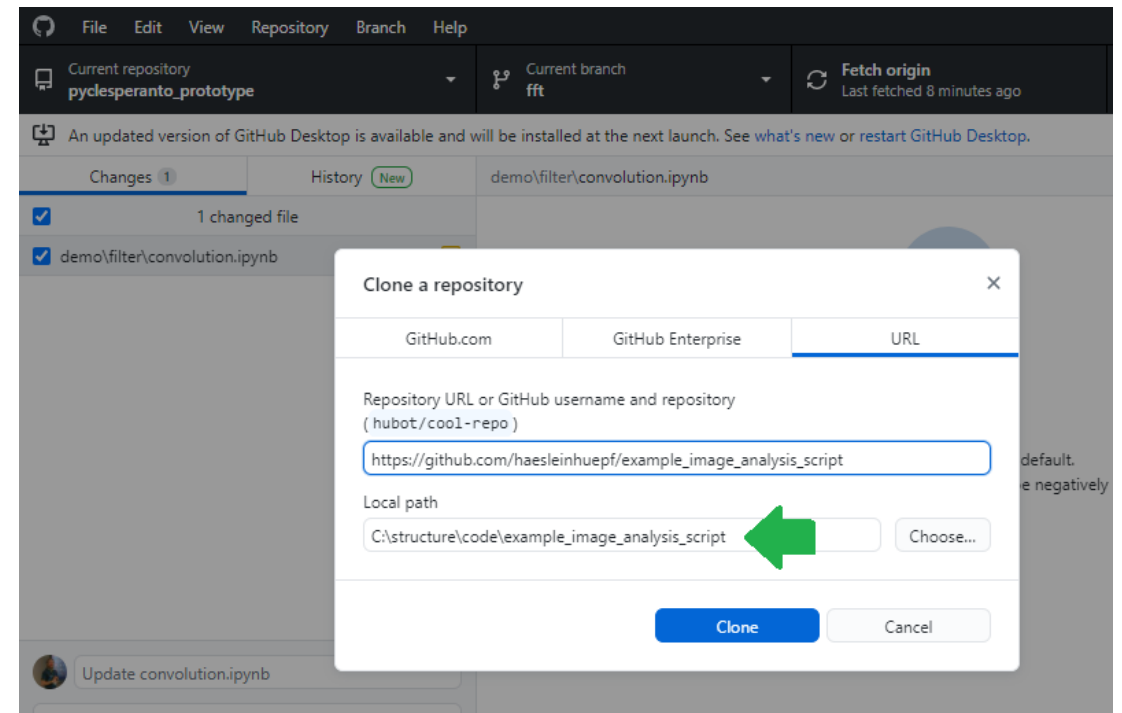
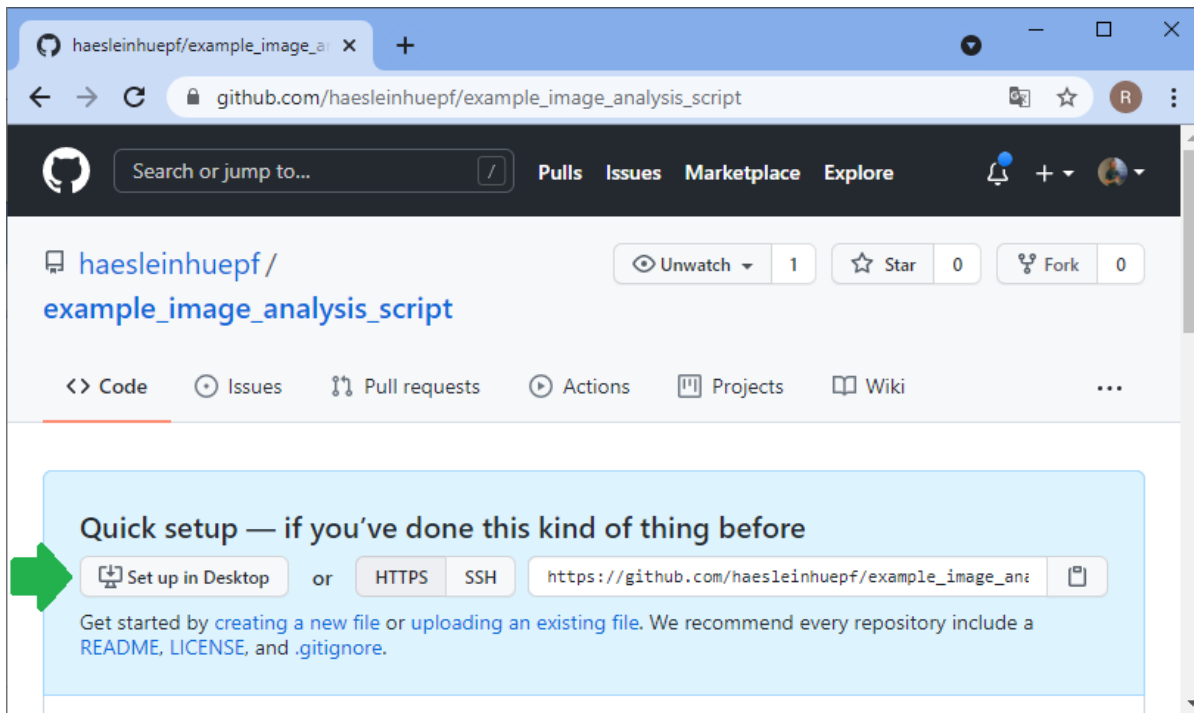
- Anlegen einer neuen Repository

The image illustrates the process of creating a new repository on GitHub through three sequential browser screenshots:

- Step 1:** The user's profile page on GitHub. A green arrow points to the 'Repositories' tab in the navigation bar, which shows 164 repositories.
- Step 2:** The 'Create a New Repository' page. A green arrow points to the 'Repository name' field, which contains 'example_image_analysis_script' and is marked as available.
- Step 3:** The same 'Create a New Repository' page, but with a green arrow pointing to the 'Create repository' button at the bottom.

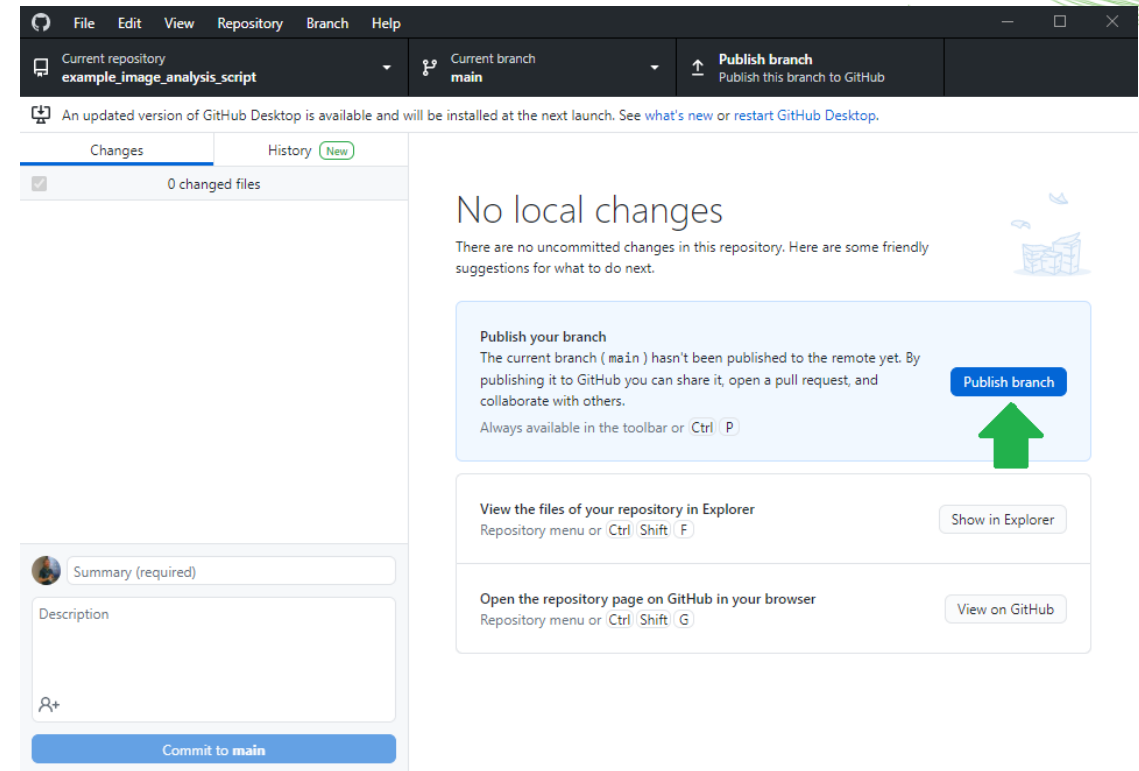
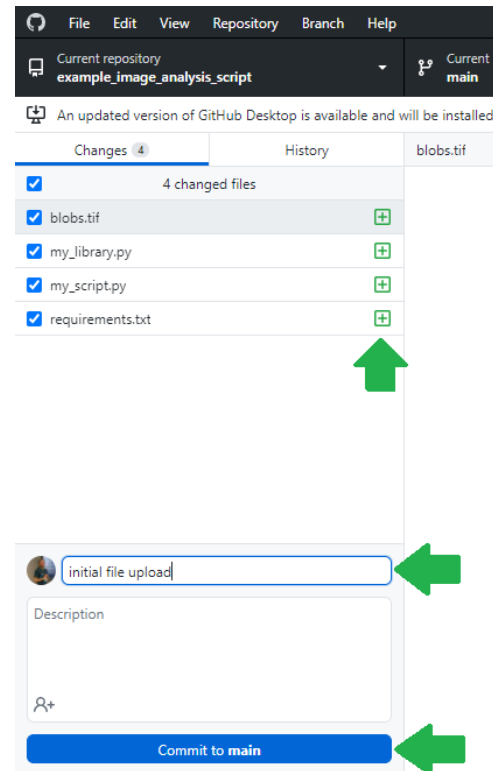
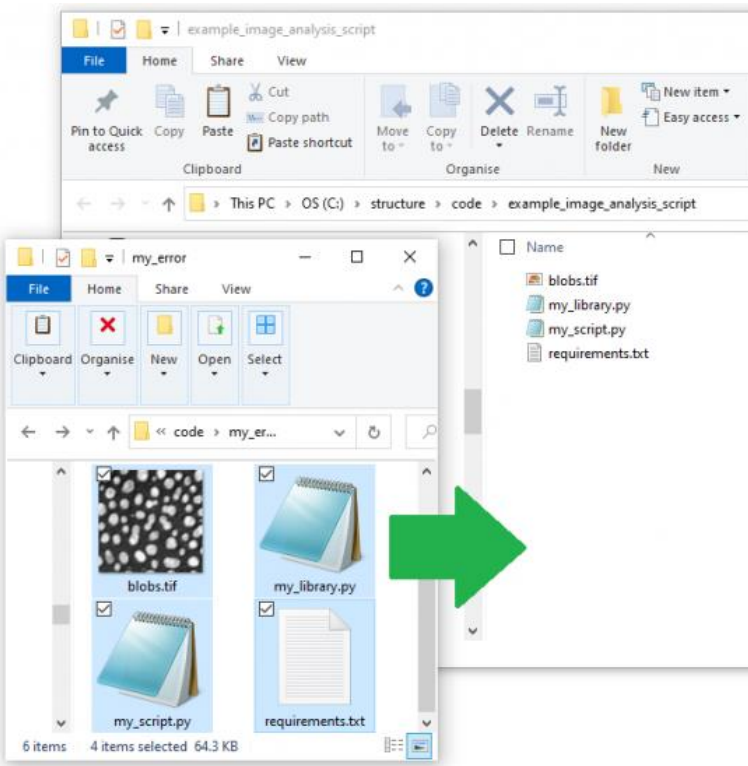
Github – repositories clonen

- `git clone https://github.com/organization/repository`
- **Alternativ: Github Desktop**



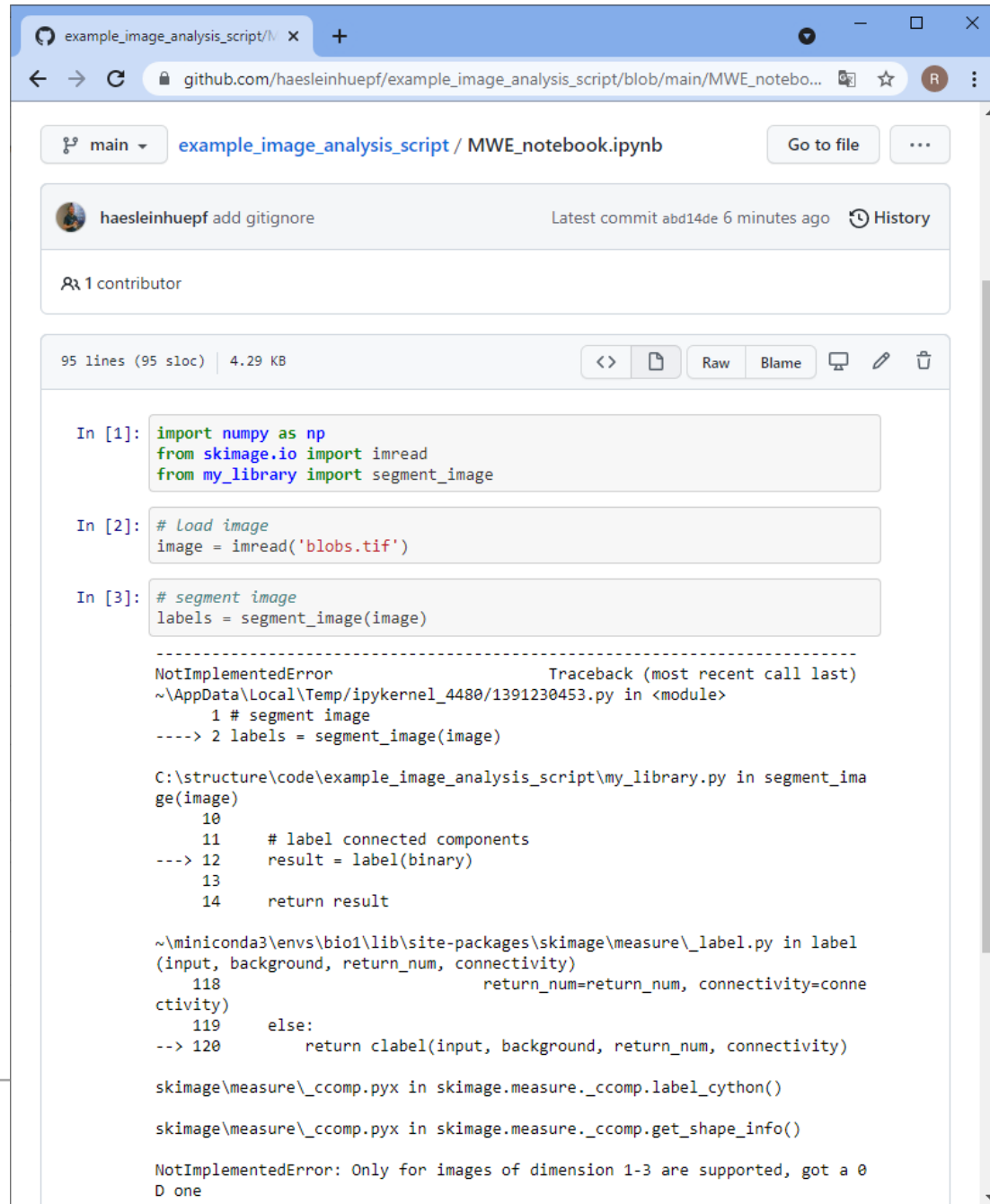
Github - Upload

- git [add], commit, push



Github

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



The screenshot shows a GitHub repository page for 'example_image_analysis_script' by user 'haesleinhuepf'. The file 'MWE_notebook.ipynb' is displayed, showing three input cells with Python code. The third cell contains a function call that results in a 'NotImplementedError'.

```
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_image(image)
     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=connectivity)
    119     else:
--> 120         return clabel(input, background, return_num, connectivity)

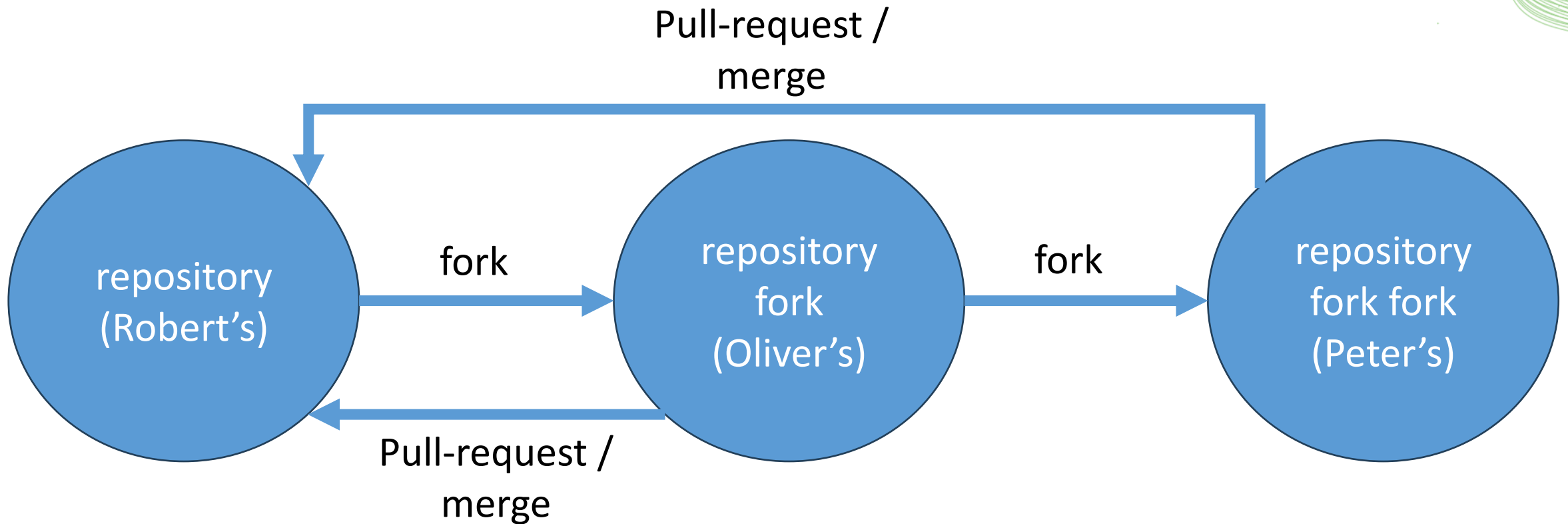
skimage\measure\_ccomp.pyx in skimage.measure._ccomp.label_cython()

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
```

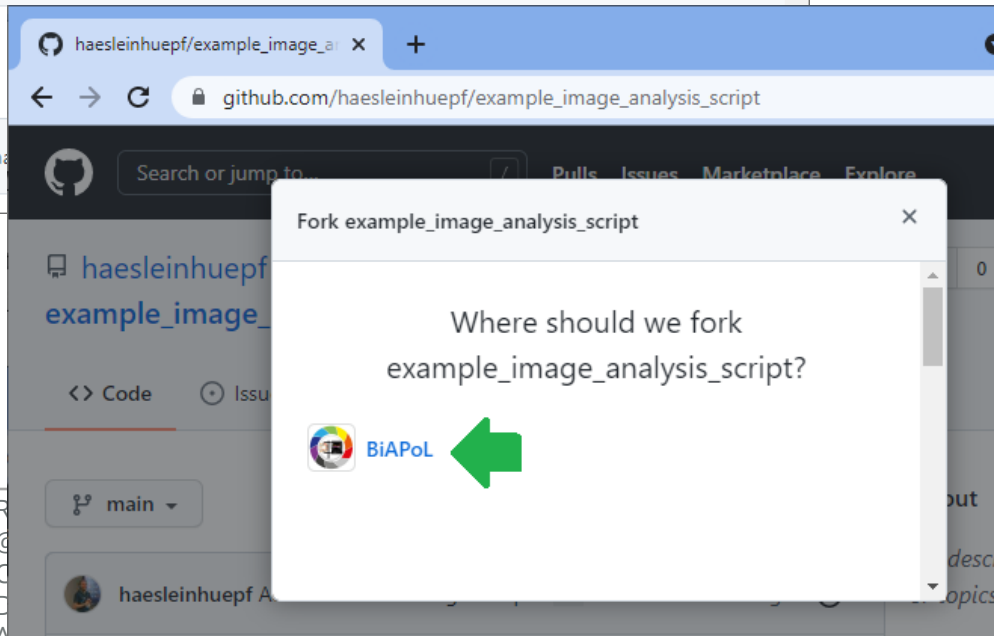
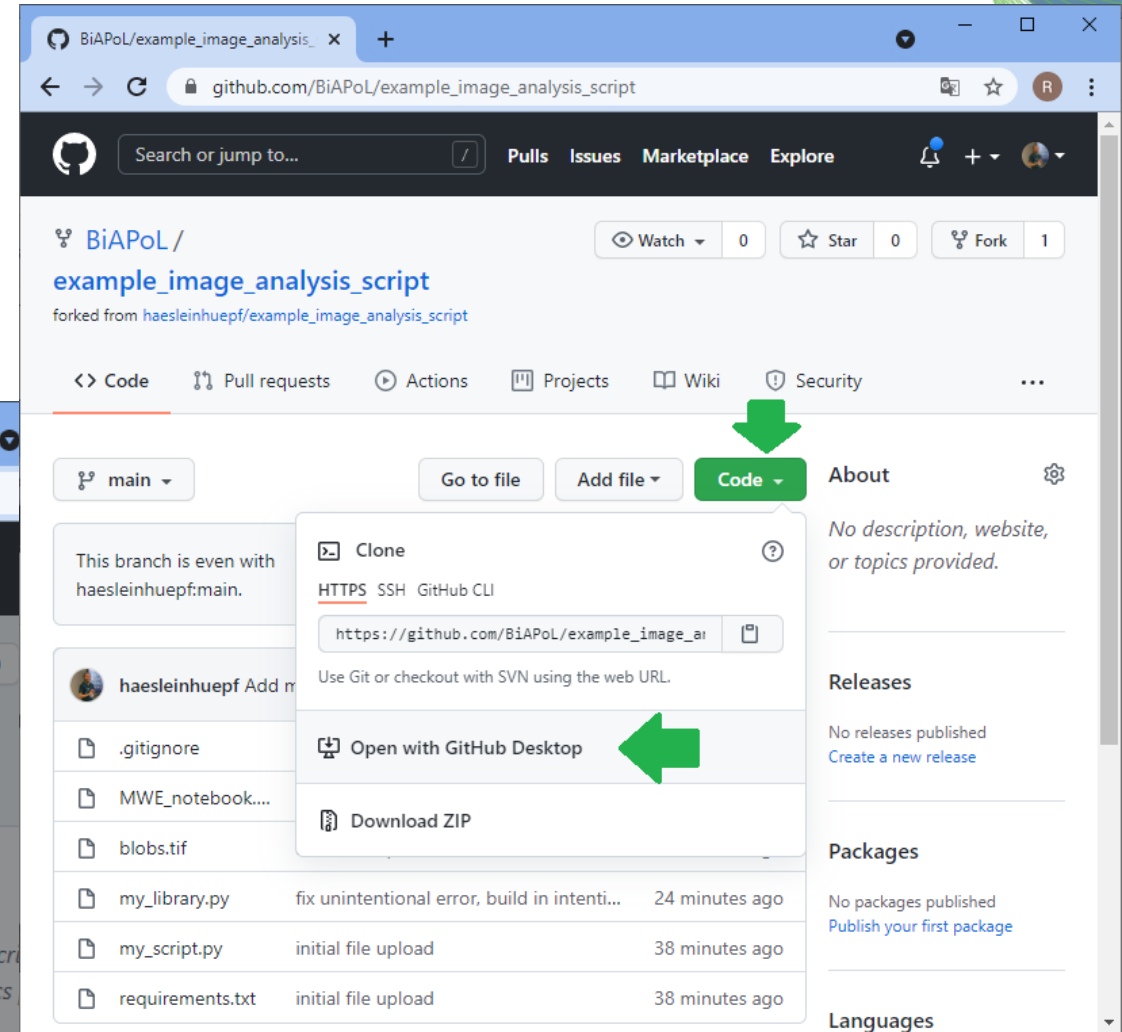
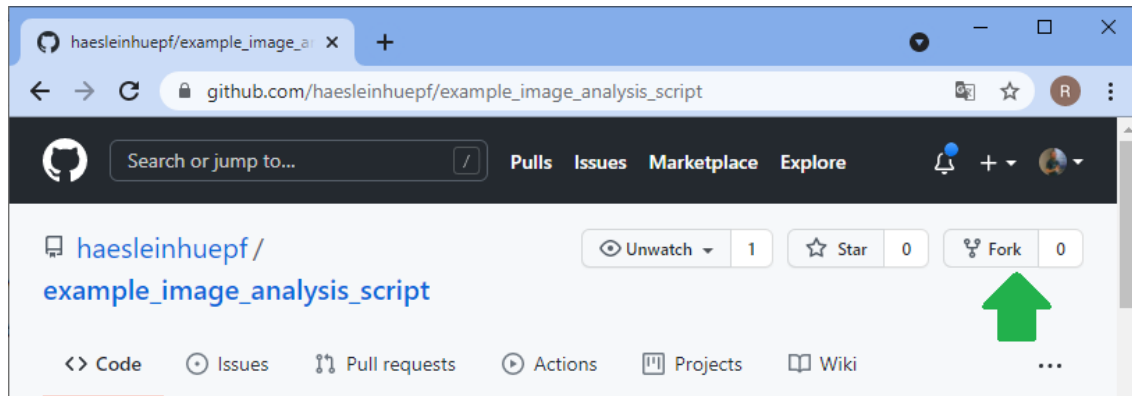

git - forking

- 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

File Edit View Repository Branch Help

Current repository: example_image_analysis_script | Current branch: main | Fetch origin: Never fetched

Changes: 2 | History: New | 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
10 11
11 12     # label connected components
12 13     result = label(binary)
@@ @@
```

bugfix: threshold_otsu

threshold_otsu delivers a number (the threshold), not a binary image. For thresholding the image, an additional step is necessary.

Commit to main

File Edit View Repository Branch Help

Current repository: example_image_analysis_script | Current branch: main | Push origin: Last fetched just now

Changes: 0 | History: New

No local changes

There are no uncommitted changes in this repository. Here are some friendly suggestions for what to do next.

Push commits to the origin remote

You have 1 local commit waiting to be pushed to GitHub.

Always available in the toolbar when there are local commits waiting to be pushed or **Ctrl | P**

Push origin

View the files of your repository in Explorer

Repository menu or **Ctrl | Shift | F**

Show in Explorer

Open the repository page on GitHub in your browser

Repository menu or **Ctrl | Shift | G**

View on GitHub

Summary (required)

Description

Commit to main

Committed just now

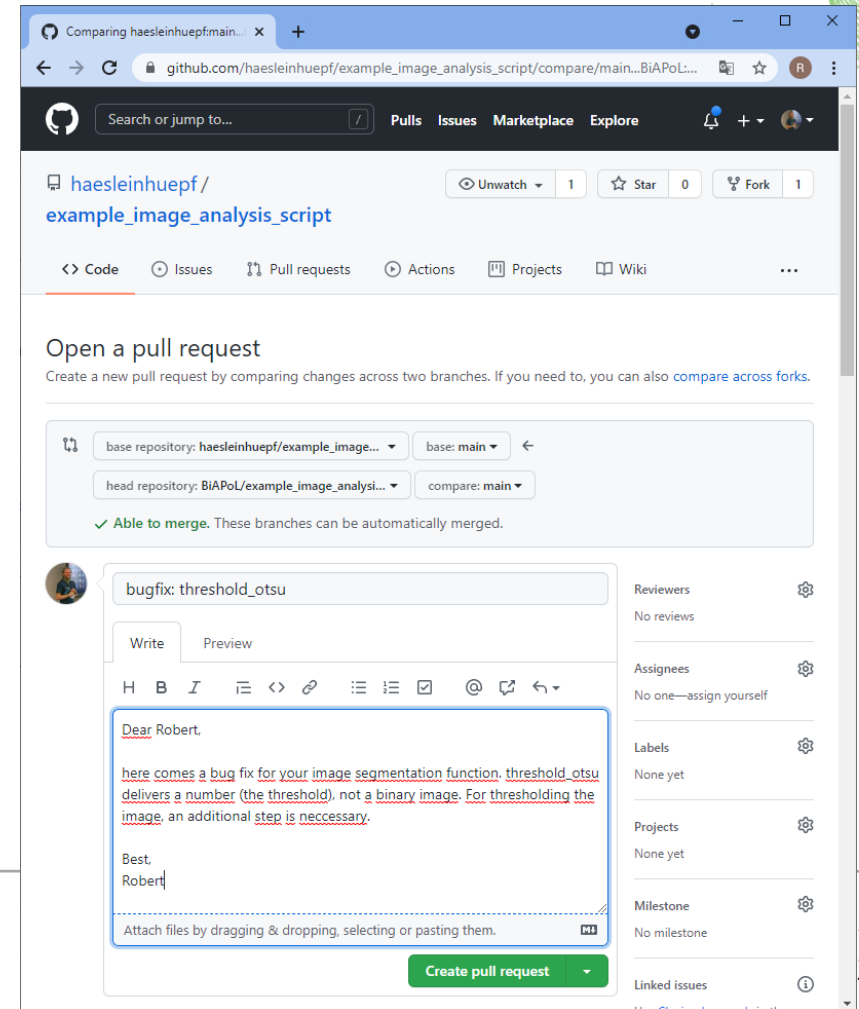
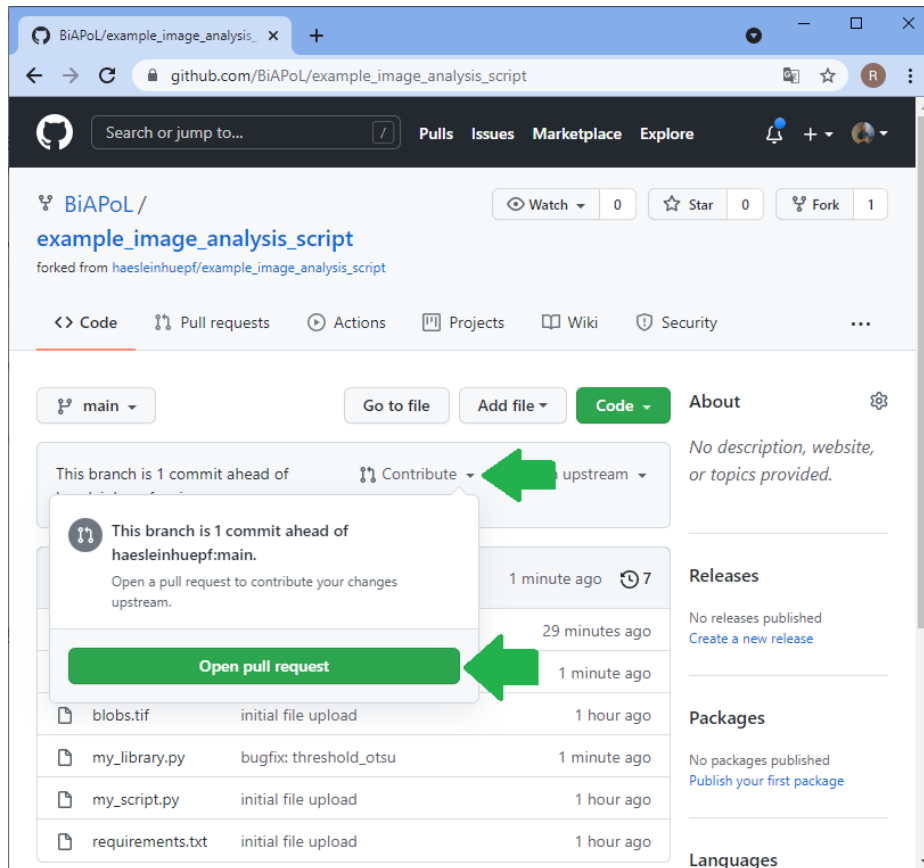
bugfix: threshold_otsu

Undo

Github – Pull-requests

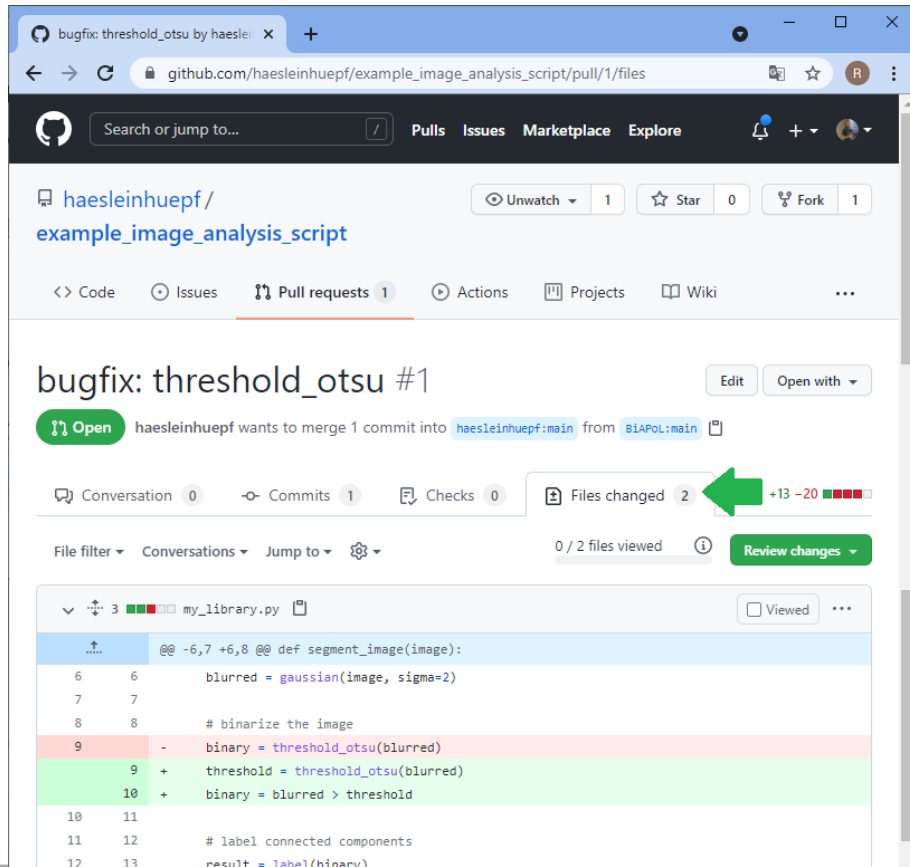
- Der formale Akt etwas beizutragen

Implizit: Ich stimme Euren Nutzungsbedingungen (Lizenz, Copyright) zu



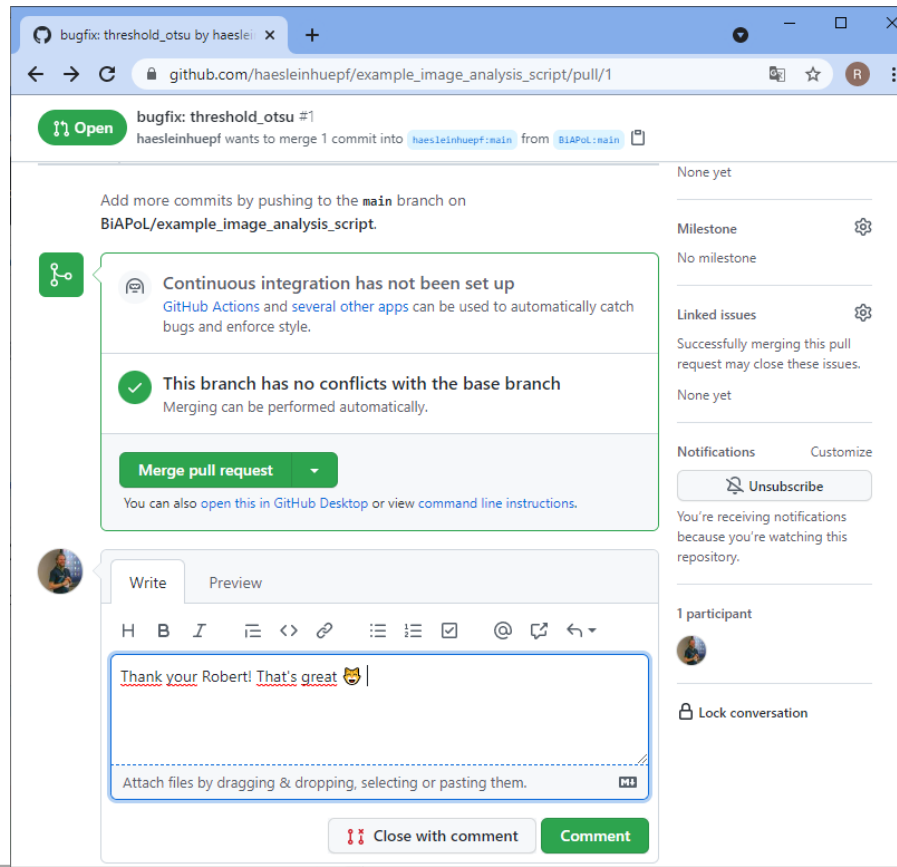
Github – Pull-requests

• Reviewer-Perspektive



The screenshot shows the GitHub pull request interface for a repository named 'example_image_analysis_script'. The pull request is titled 'bugfix: threshold_otsu #1' and is being reviewed by 'haesleinhuepf'. The interface shows the 'Files changed' section with a diff view of the file 'my_library.py'. A green arrow points to the 'Files changed' section, which shows 2 files changed, with a total of +13 lines added and -20 lines removed. The diff view shows the following changes:

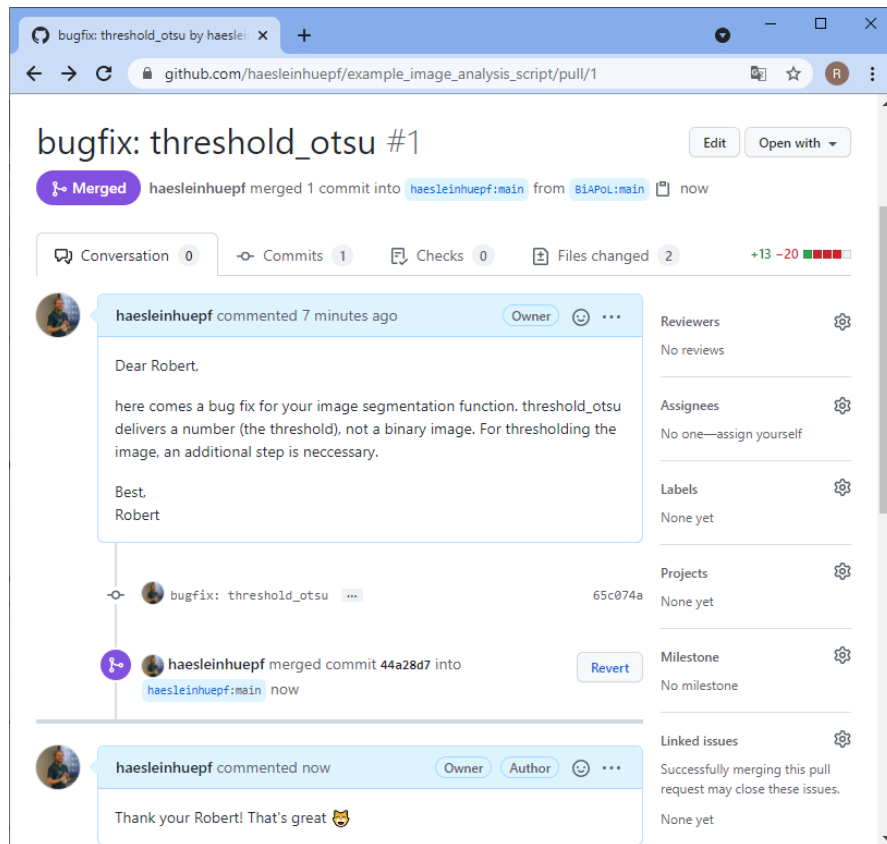
```
@@ -6,7 +6,8 @@ def segment_image(image):
6 6     blurred = gaussian(image, sigma=2)
7 7
8 8     # binarize the image
9 -     binary = threshold_otsu(blurred)
9 +     threshold = threshold_otsu(blurred)
10 +     binary = blurred > threshold
10 11
11 12     # label connected components
12 13     result = label(binary)
```



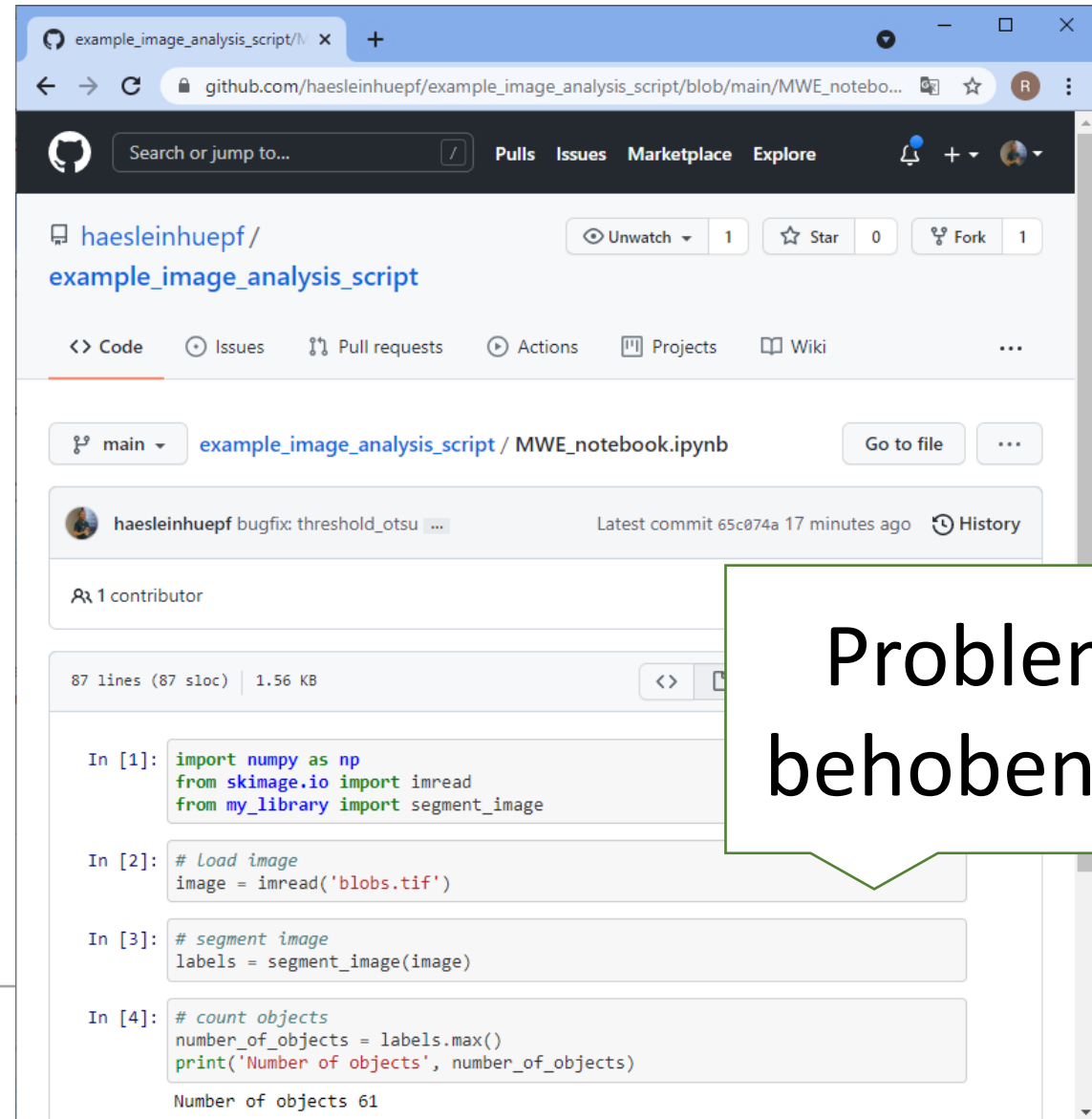
The screenshot shows the GitHub pull request interface for a repository named 'example_image_analysis_script'. The pull request is titled 'bugfix: threshold_otsu #1' and is being reviewed by 'haesleinhuepf'. The interface shows the 'Merge pull request' button, which is highlighted in green. The 'Merge pull request' button is accompanied by a green checkmark and the text 'This branch has no conflicts with the base branch. Merging can be performed automatically.' Below the 'Merge pull request' button, there is a text input field with the text 'Thank your Robert! That's great 🐱'. The 'Merge pull request' button is accompanied by a green checkmark and the text 'This branch has no conflicts with the base branch. Merging can be performed automatically.' Below the 'Merge pull request' button, there is a text input field with the text 'Thank your Robert! That's great 🐱'. The 'Merge pull request' button is accompanied by a green checkmark and the text 'This branch has no conflicts with the base branch. Merging can be performed automatically.' Below the 'Merge pull request' button, there is a text input field with the text 'Thank your Robert! That's great 🐱'.

Github – pull requests

- Reviewer perspective



The screenshot shows a pull request titled "bugfix: threshold_otsu #1" in a repository named "example_image_analysis_script". The pull request is marked as "Merged" and shows that the user "haesleinhuepf" merged 1 commit into the "main" branch from the "BIAPOL:main" branch. The conversation includes a comment from the reviewer, haesleinhuepf, dated 7 minutes ago, addressed to Robert. The comment 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 pull request also shows a commit history with a commit by haesleinhuepf merged into main, and a final comment from haesleinhuepf saying "Thank your Robert! That's great 🐱".



The screenshot shows the GitHub repository page for "haesleinhuepf/example_image_analysis_script". The file "MWE_notebook.ipynb" is selected, showing 87 lines of code. The code is as follows:

```
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)

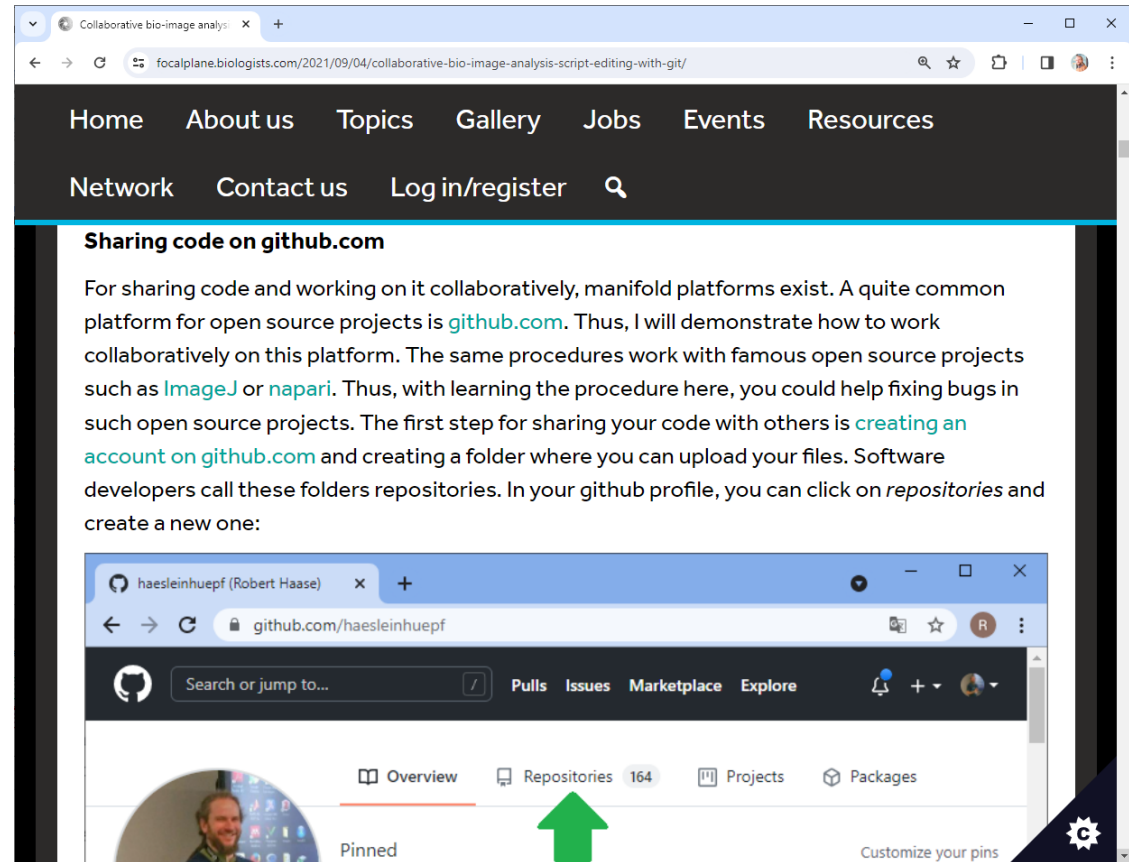
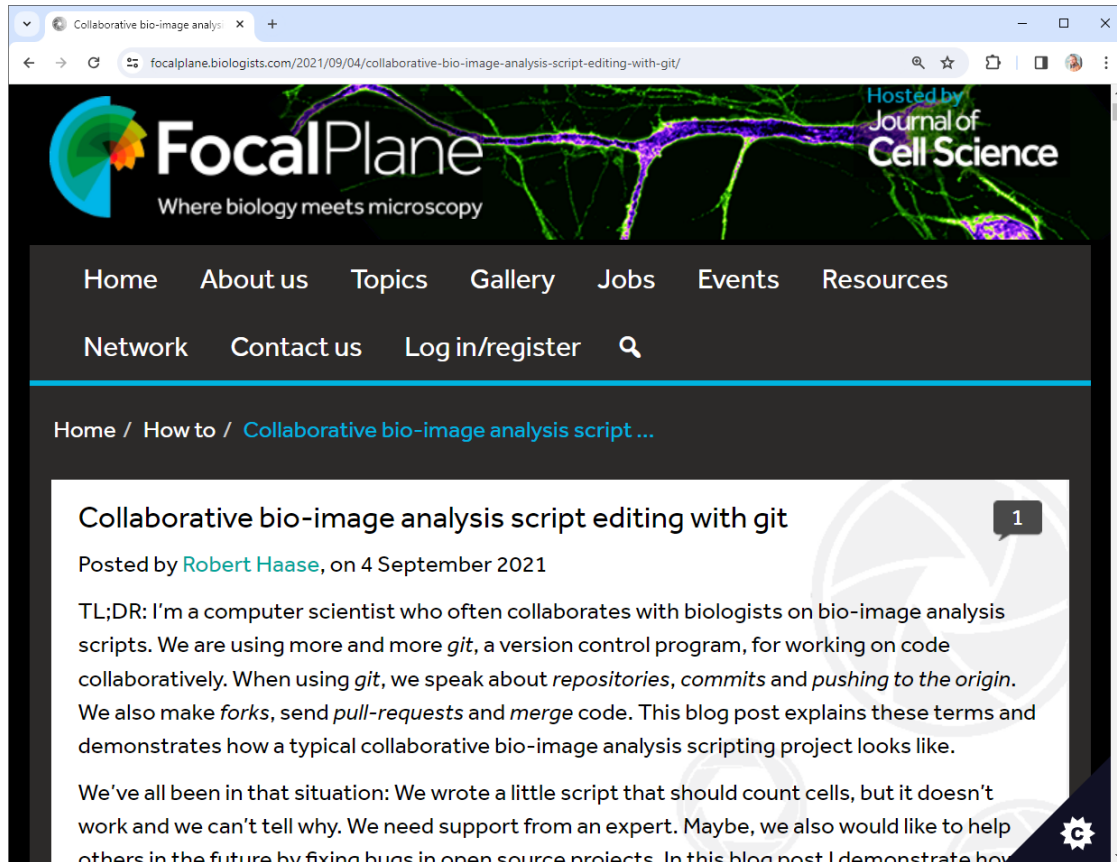
In [4]: # count objects
        number_of_objects = labels.max()
        print('Number of objects', number_of_objects)

Number of objects 61
```

A callout box with a green border and a speech bubble tail points to the code, containing the text "Problem behoben :-)".

GitHub

- Falls das zu schnell war...



Quiz

- Ein Fork ist ...

Eine Kopie



Ein Ordner



Eine private
Repository



Eine Datei





Mittagspause!

13:30 geht es weiter

Praktikum

Robert Haase

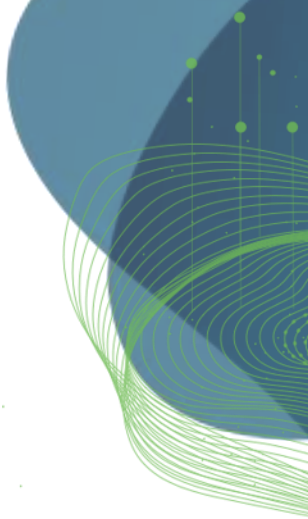
GEFÖRDERT VOM



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.

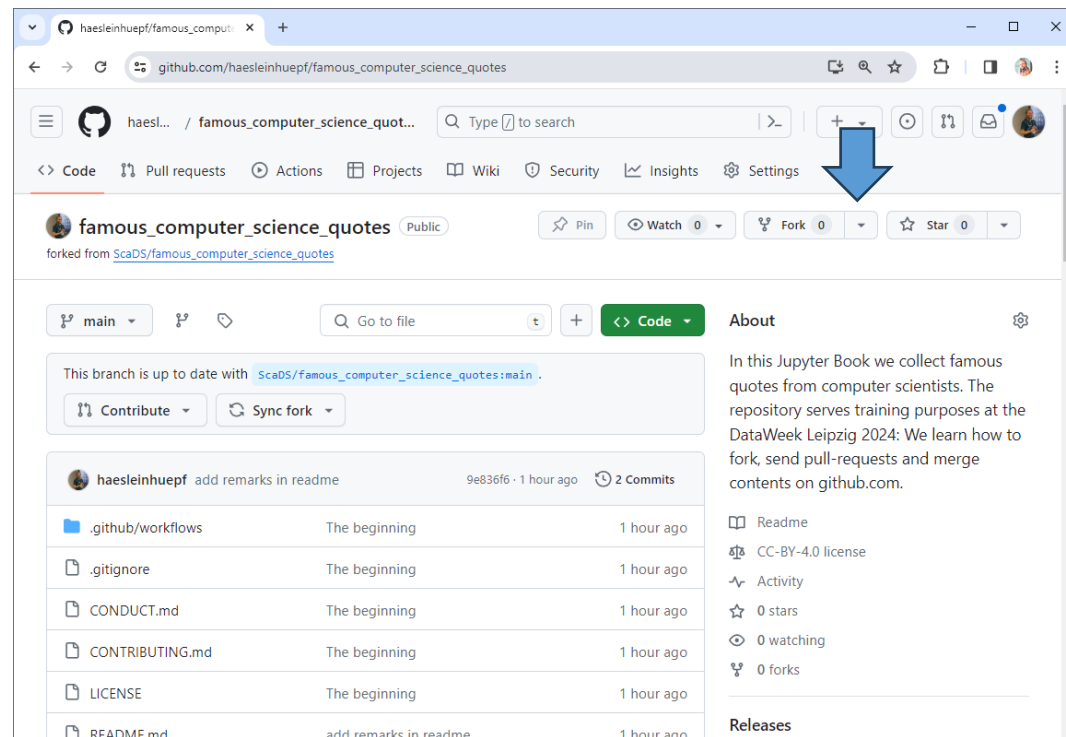
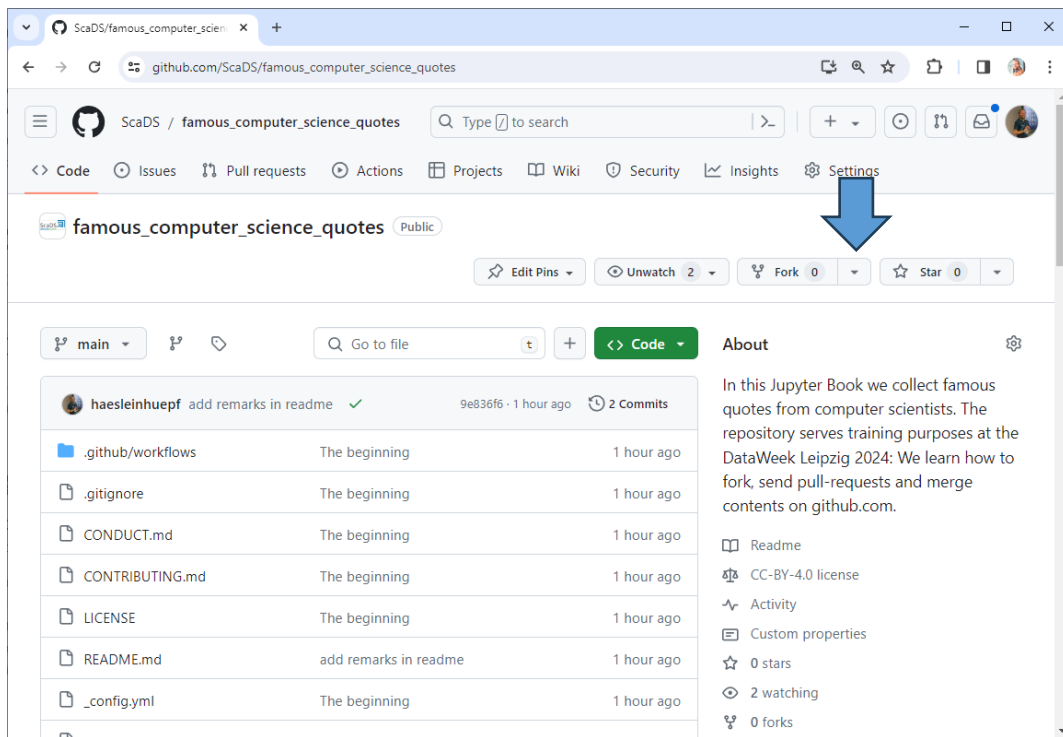


Praktikum: Merge-Konflikte

- 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

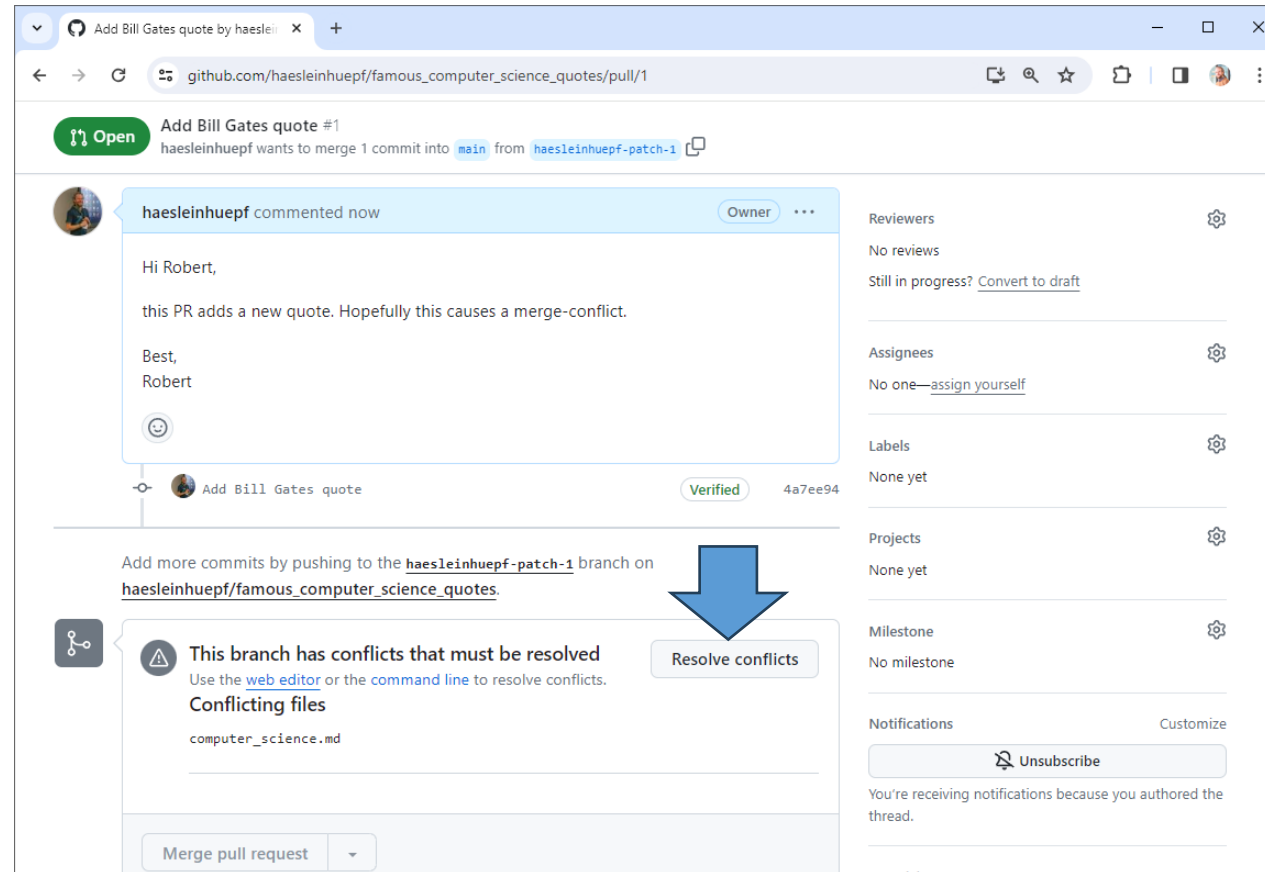
```
@@ -2,6 +2,8 @@
2 2
3 3 * "There are two ways to write error-free programs; only the third one works." Alan Perlis [source]
   (https://www.brainyquote.com/quotes/alan_perlis_177352)
4 4
5 + * "The computer was born to solve problems that did not exist before." - Bill Gates [source](https://www.create-learn.us/blog/computer-science-quotes/)
6 +
5 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)
6 8
7 9
```

```
@@ -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_177352)
4 -
5 3 * "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 4
5 + * "There are two ways to write error-free programs; only the third one works." Alan Perlis [source]
   (https://www.brainyquote.com/quotes/alan_perlis_177352)
7 6
8 7
9 8
```

Modifiziert die gleiche Datei.
Erzeugt einen Konflikt!

Praktikum: Merge-Konflikte

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



Github-Zenodo-Integration & Github Pages

Robert Haase

GEFÖRDERT VOM



Bundesministerium
für Bildung
und Forschung



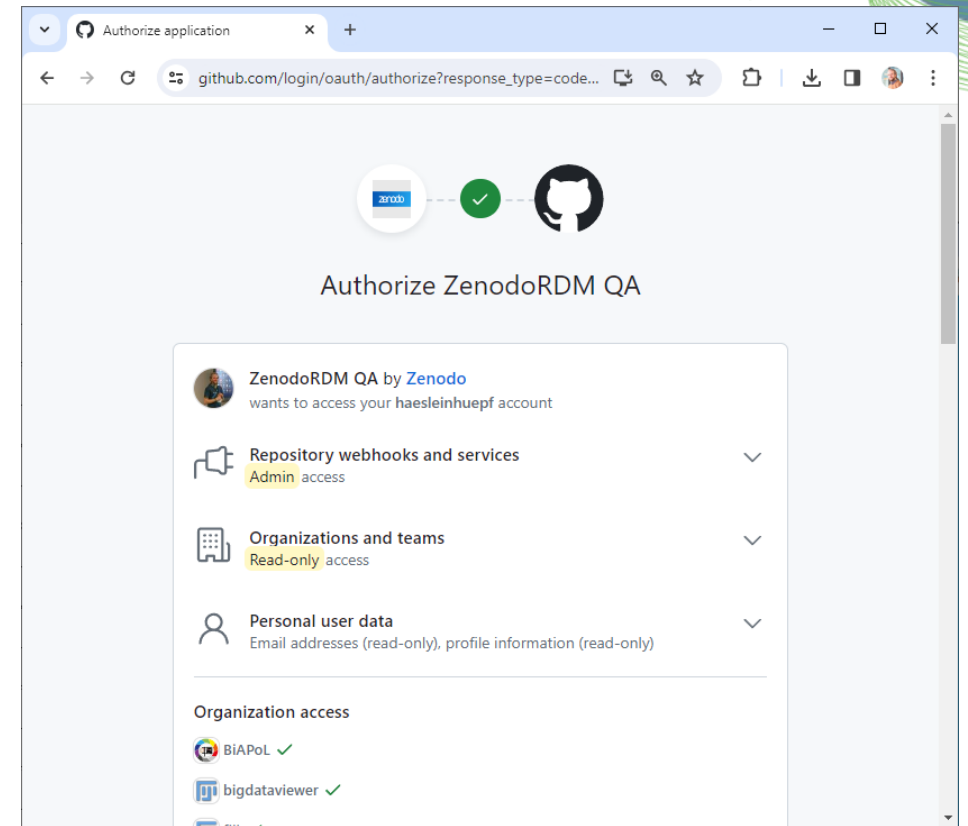
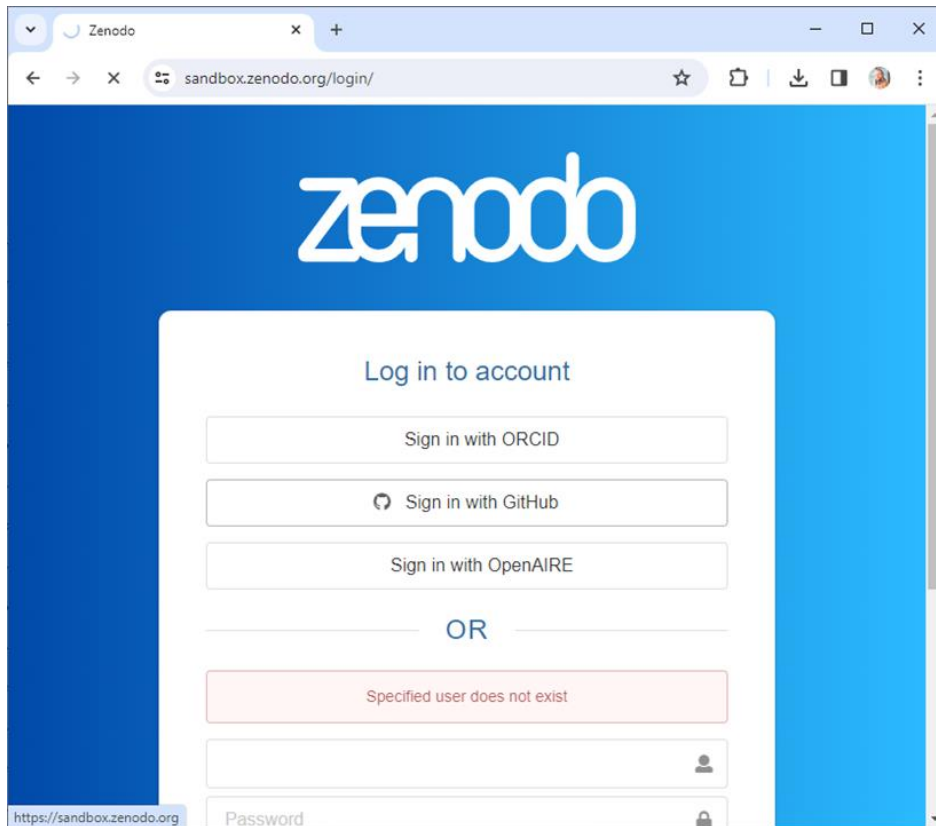
SACHSEN
Diese Maßnahme wird gefördert durch die Bundesregierung
aufgrund eines Beschlusses des Deutschen Bundestages.
Diese Maßnahme wird mitfinanziert durch Steuermittel auf
der Grundlage des von den Abgeordneten des Sächsischen
Landtags beschlossenen Haushaltes.

Github versus Zenodo

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

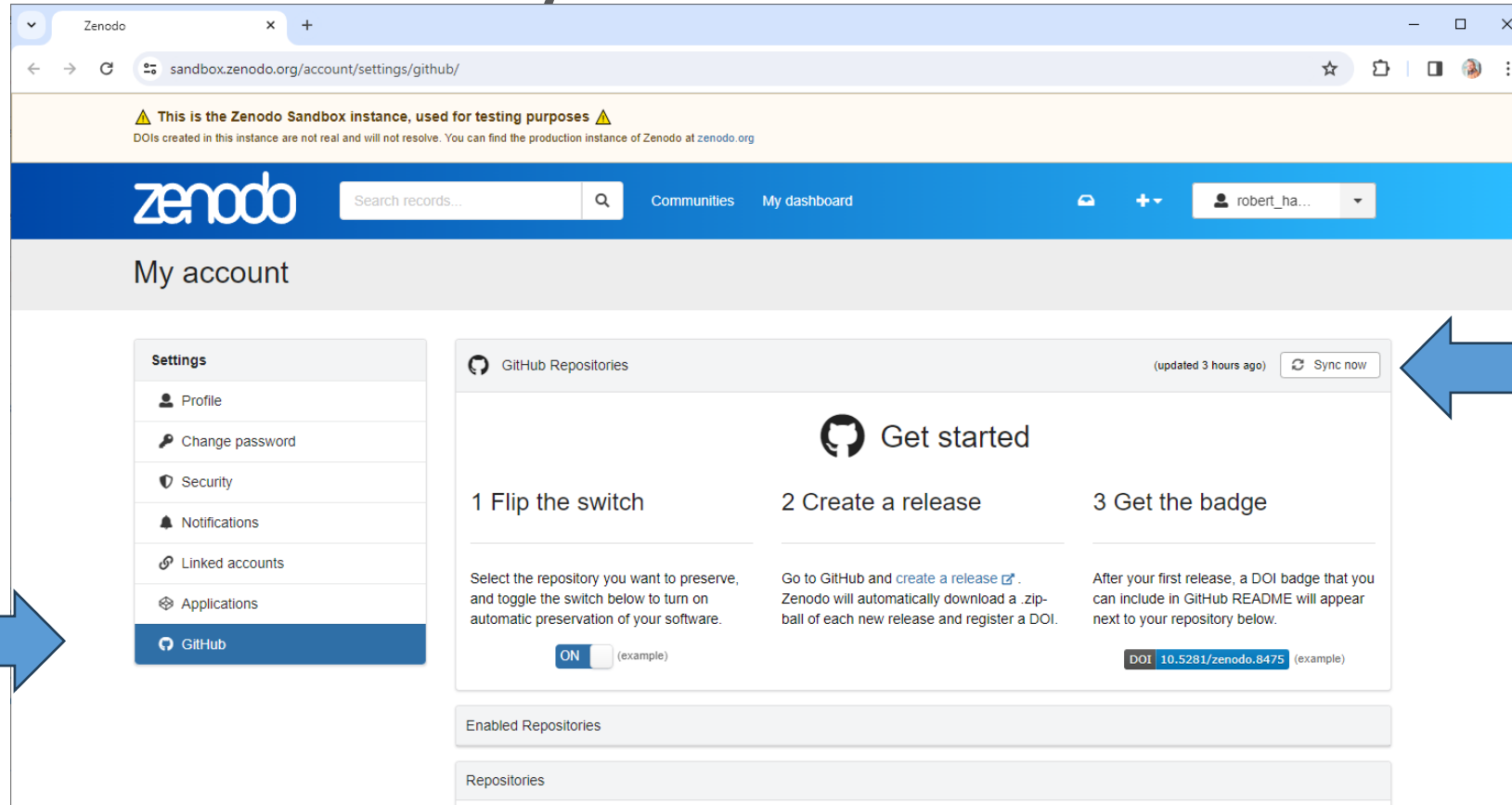
Zenodo-Integration

- Login bei Zenodo mit dem Github-Account



Zenodo-Integration

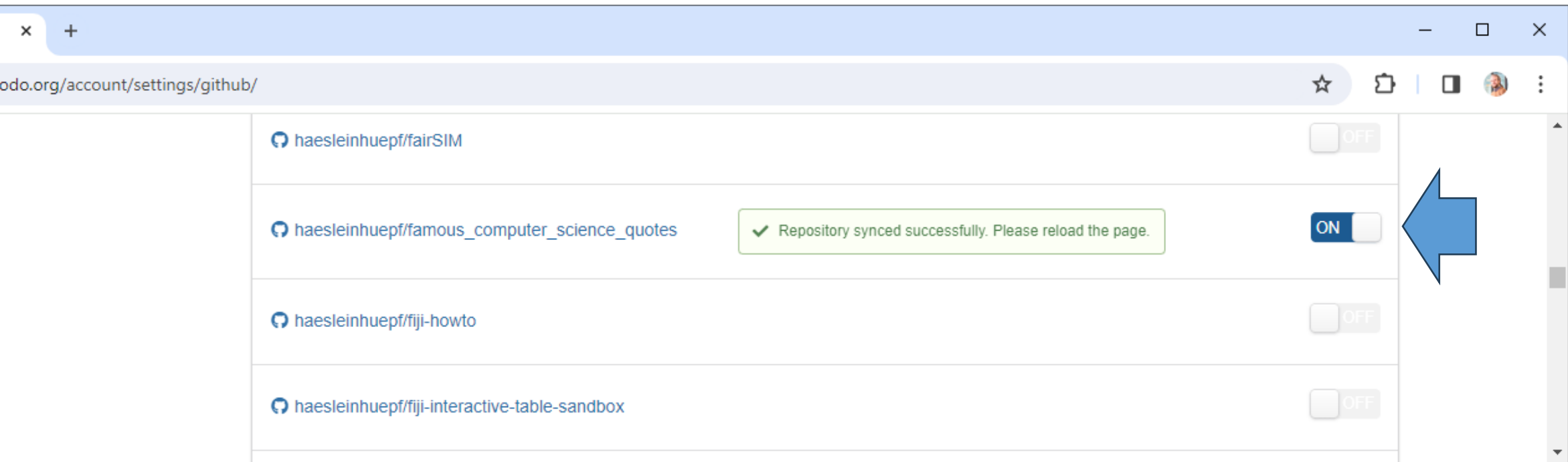
- Github mit Zenodo synchronisieren



The screenshot shows the Zenodo account settings page for GitHub integration. The browser address bar displays `sandbox.zenodo.org/account/settings/github/`. A warning banner at the top states: "This is the Zenodo Sandbox instance, used for testing purposes". The Zenodo logo and navigation menu are visible at the top. The page title is "My account". On the left, a "Settings" sidebar lists options: Profile, Change password, Security, Notifications, Linked accounts, Applications, and GitHub (highlighted with a blue arrow). The main content area is titled "GitHub Repositories" and includes a "Sync now" button (highlighted with a blue arrow). Below this, a "Get started" section provides a three-step guide: 1. Flip the switch (with an "ON" toggle and "(example)" text), 2. Create a release (with a link to "create a release"), and 3. Get the badge (with an example DOI: `DOI 10.5281/zenodo.8475` and "(example)" text). Below the guide are sections for "Enabled Repositories" and "Repositories".

Zenodo integration

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



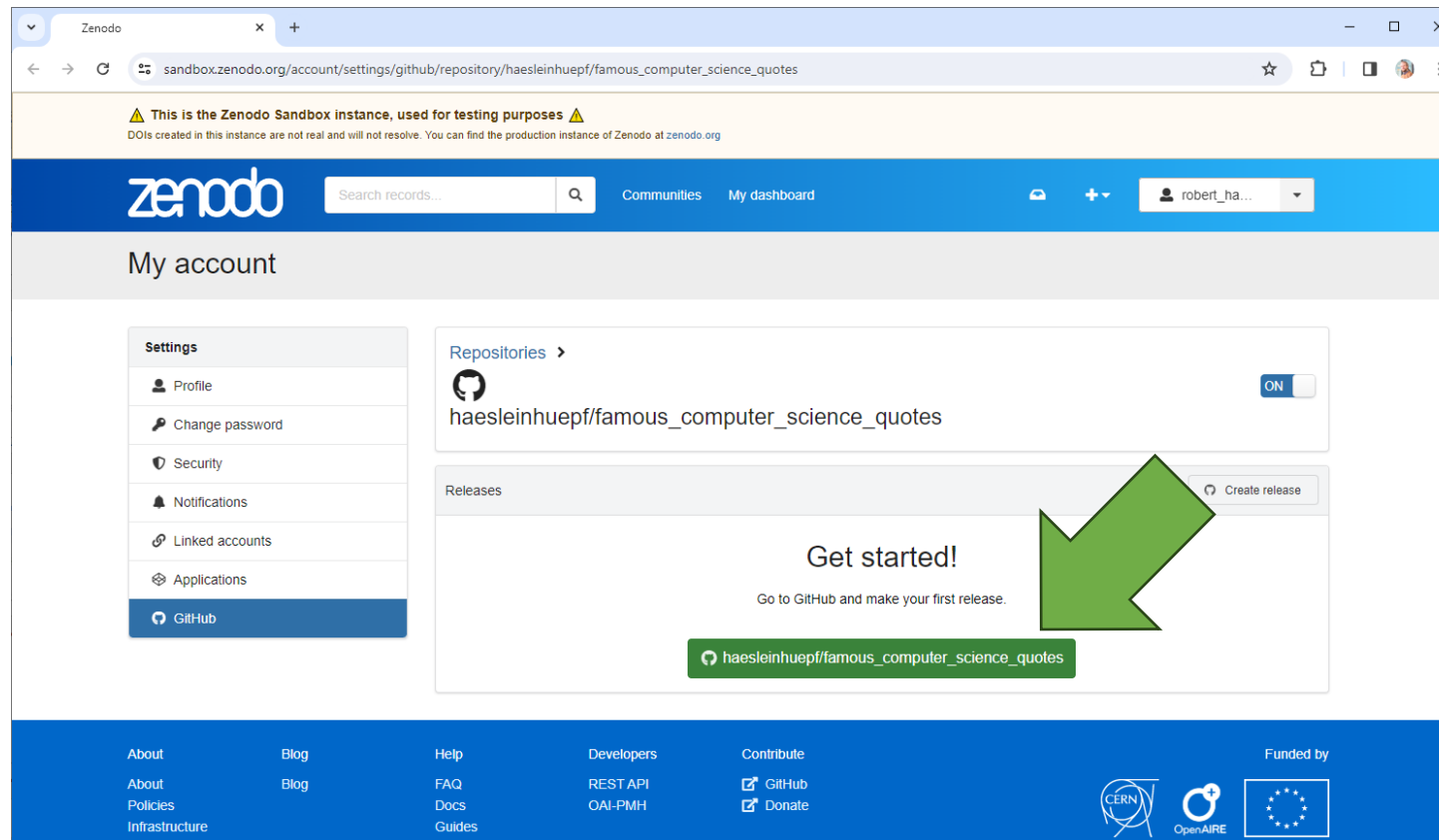
The screenshot shows the Zenodo account settings page for GitHub integration. The URL in the browser is `zenodo.org/account/settings/github/`. The page lists four repositories with their respective GitHub integration status:

Repository	Status
haesleinhuepf/fairSIM	OFF
haesleinhuepf/famous_computer_science_quotes	ON
haesleinhuepf/fiji-howto	OFF
haesleinhuepf/fiji-interactive-table-sandbox	OFF

A green notification box above the 'famous_computer_science_quotes' repository indicates: "✓ Repository synced successfully. Please reload the page." A blue arrow points to the 'ON' toggle switch for this repository.

Zenodo integration

- Klick auf das Repository um Details zu sehen



Zenodo integration

- Erzeugt einen Versionstag (bspw. “v1”)

haesleinhuepf / famous_computer_science_quotes

Releases Tags

v1 Target: main Previous tag: auto Generate release notes

Excellent! This tag will be created from the target when you publish this release.

v1

Write Preview H B I `<> @`

This is the first release. We use it to generate a DOI on Zenodo (sandbox)|

Markdown is supported Paste, drop, or click to add files

Attach binaries by dropping them here or selecting them.

Set as a pre-release This release will be labeled as non-production ready

Publish release Save draft

v1 Target: main Previous tag: auto Genera

Excellent! This tag will be created from the target when you publish this release.

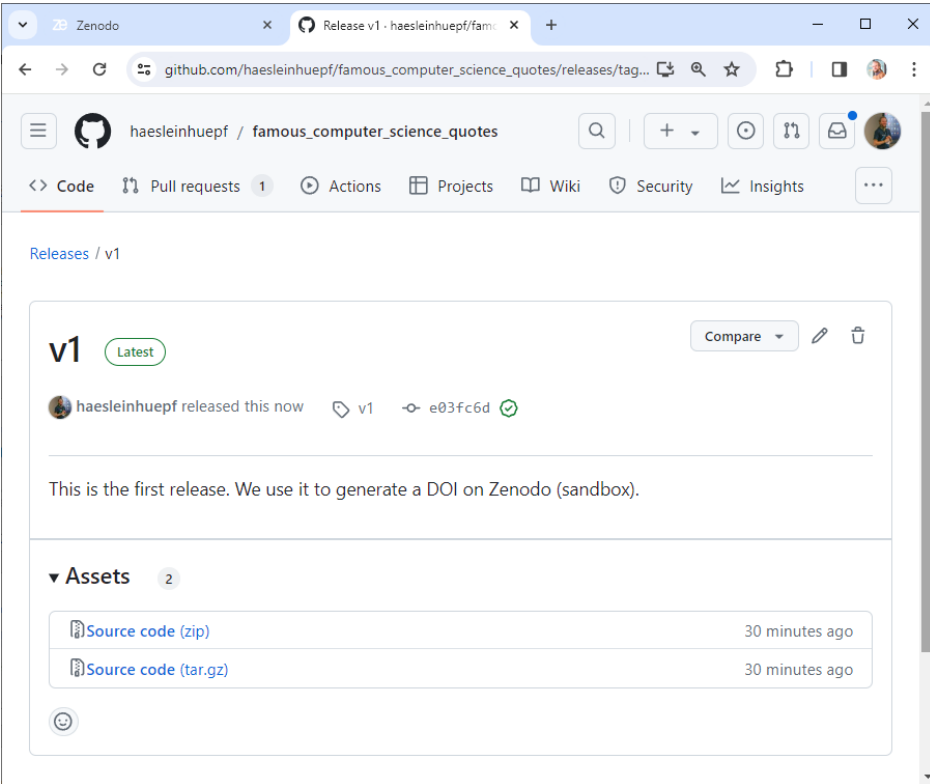
v1

Write Preview H

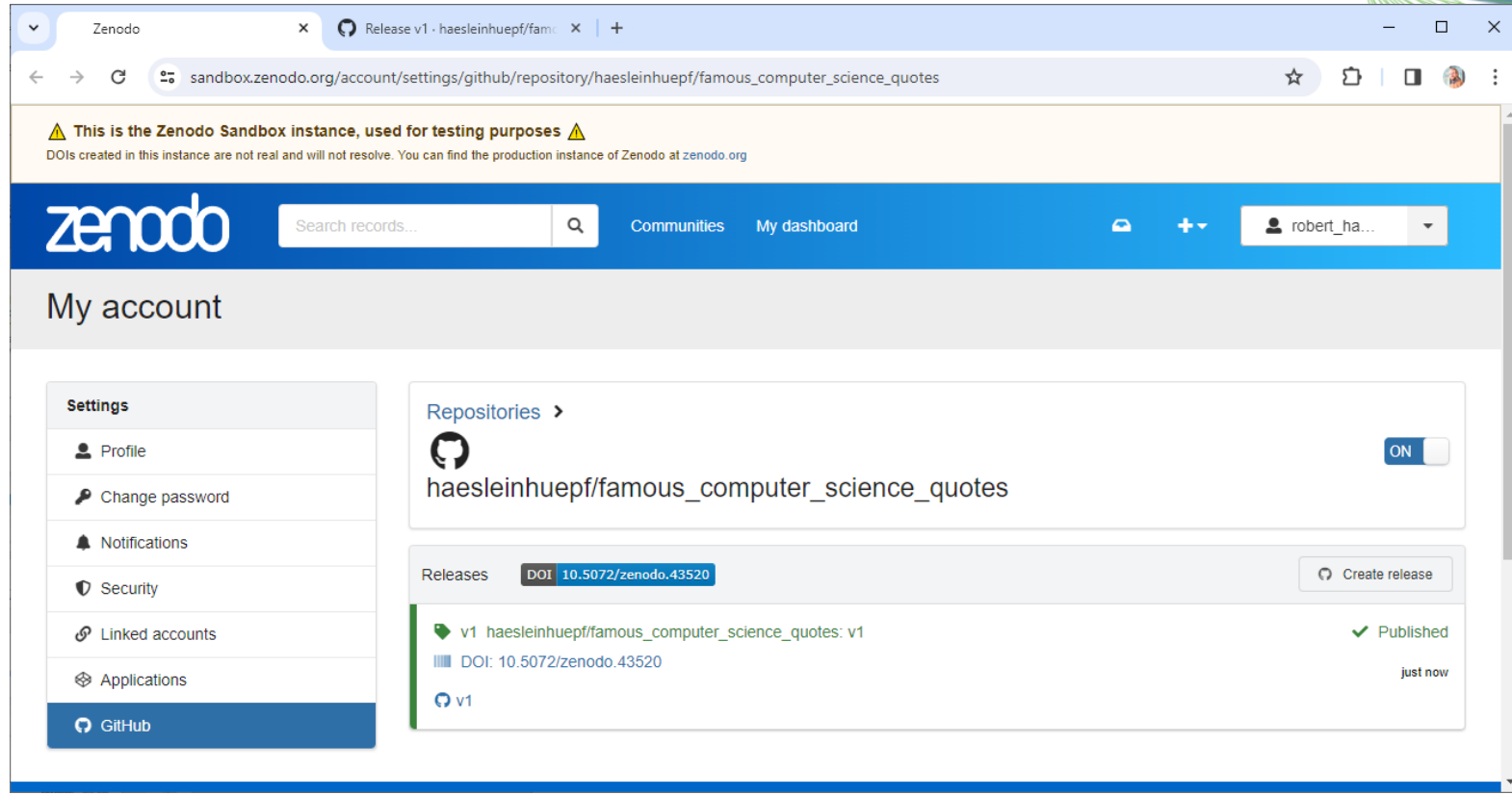
This is the first release. We use it to generate a DOI on Zenodo (sandbox)|

Zenodo integration

- Erkundet den Release auf Github und Zenodo



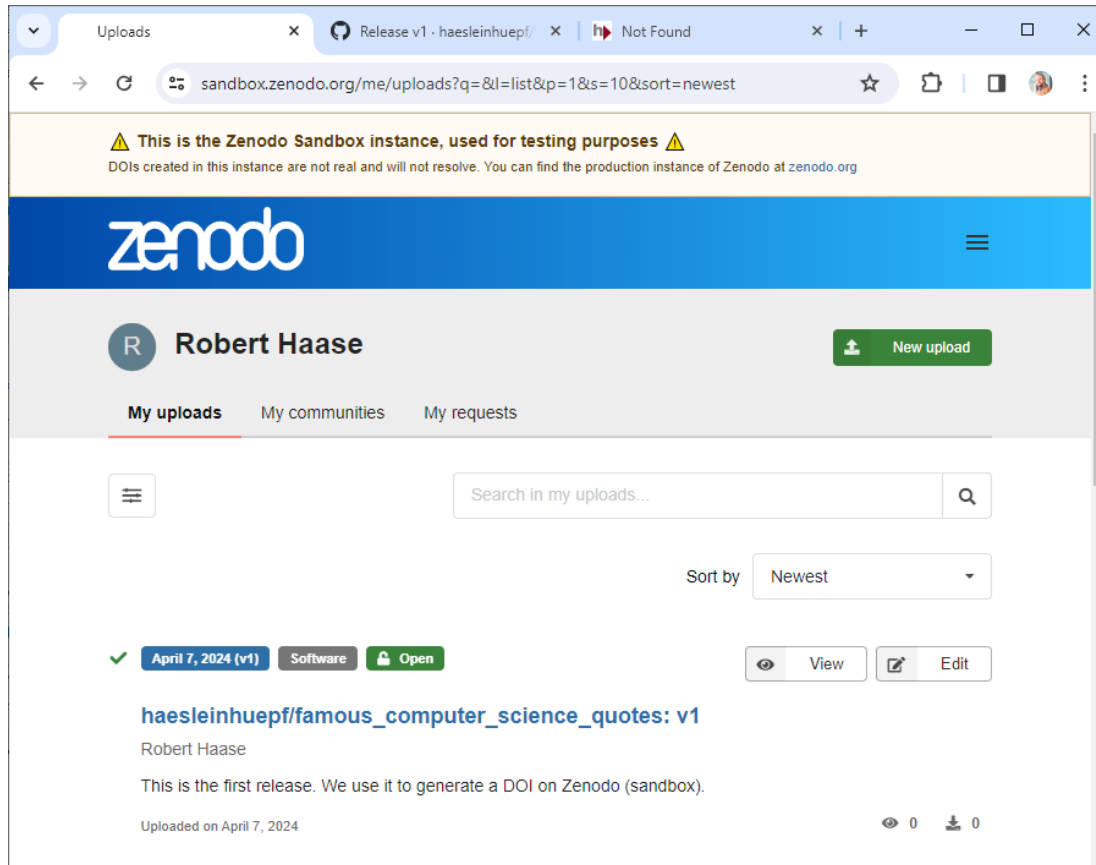
The screenshot shows the GitHub interface for the repository 'haesleinhuepf / famous_computer_science_quotes'. The 'Releases' section is active, displaying the latest release 'v1'. The release description states: 'This is the first release. We use it to generate a DOI on Zenodo (sandbox)'. Under the 'Assets' section, two source code files are listed: 'Source code (zip)' and 'Source code (tar.gz)', both uploaded 30 minutes ago.



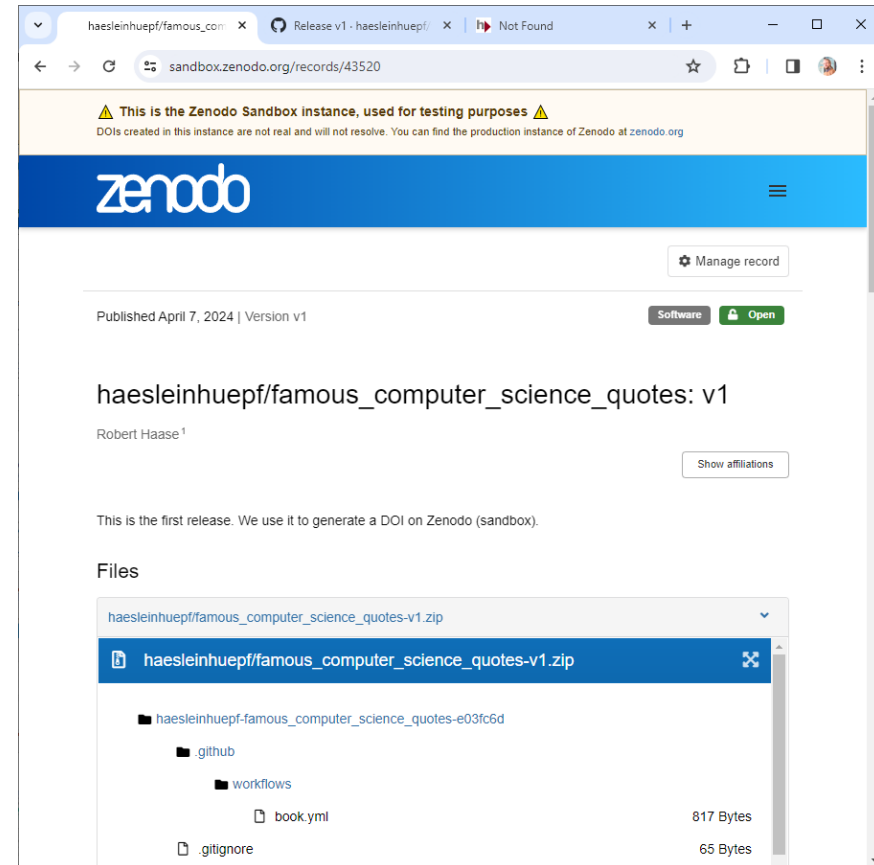
The screenshot shows the Zenodo 'My account' settings page for the repository 'haesleinhuepf/famous_computer_science_quotes'. A warning banner at the top indicates this is a Zenodo Sandbox instance. The 'Releases' section shows a single release 'v1' with the DOI '10.5072/zenodo.43520', which has been published 'just now'. The 'Settings' sidebar on the left includes options for Profile, Change password, Notifications, Security, Linked accounts, Applications, and GitHub.

Zenodo integration

- Zurück auf dem Dashboard wird der neue Eintrag angezeigt



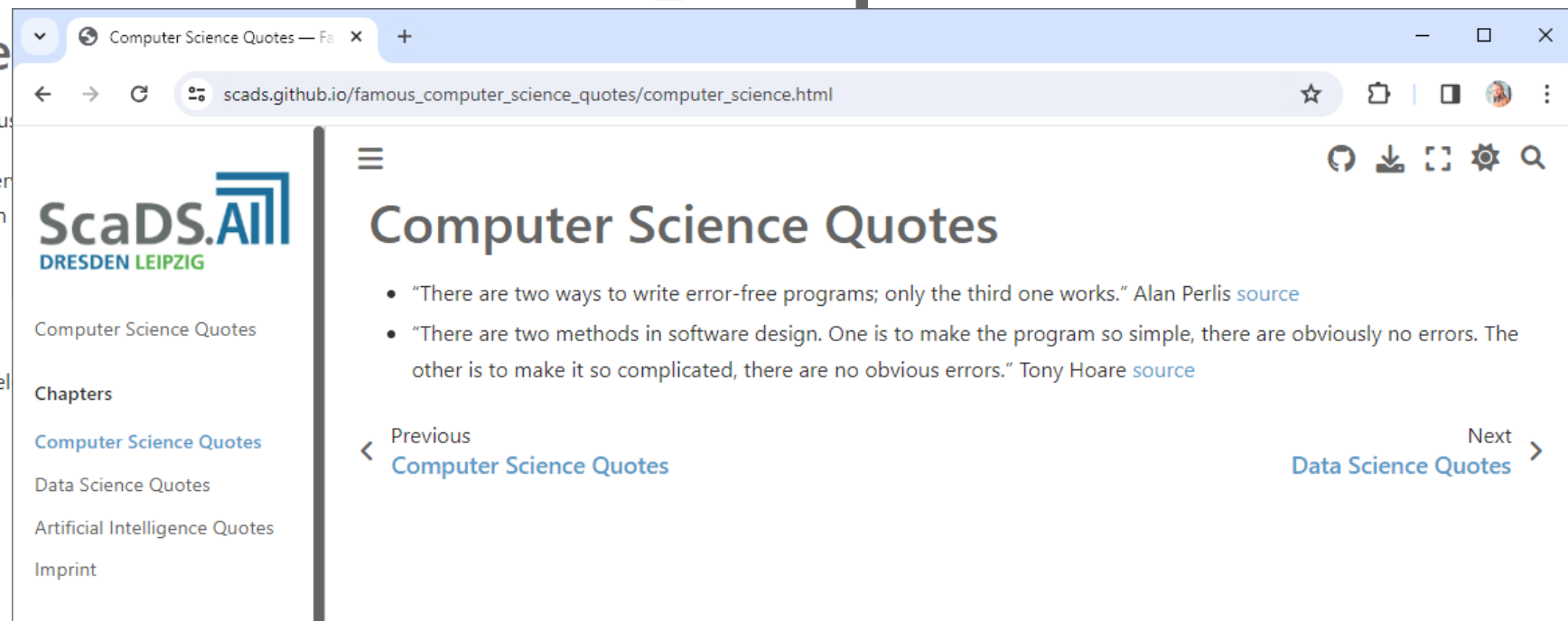
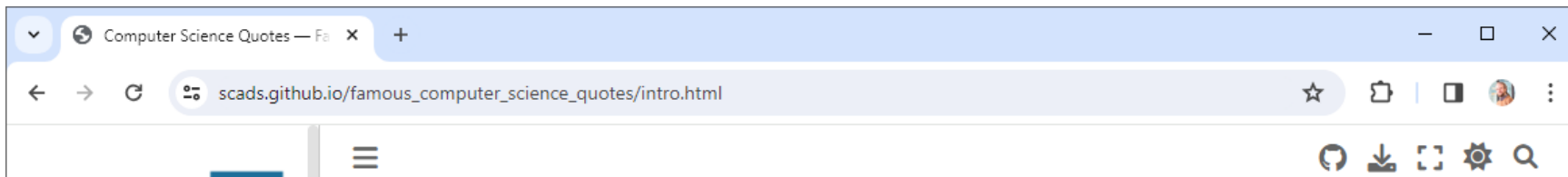
The screenshot shows the Zenodo dashboard for user Robert Haase. At the top, there is a warning banner: "This is the Zenodo Sandbox instance, used for testing purposes". Below the banner is the Zenodo logo and the user's name "Robert Haase" with a "New upload" button. The main content area shows a list of uploads, with the most recent one highlighted: "haesleinhuepf/famous_computer_science_quotes: v1". This entry is dated "April 7, 2024 (v1)" and is labeled as "Software" and "Open". It includes a description: "This is the first release. We use it to generate a DOI on Zenodo (sandbox)." and shows it was uploaded on April 7, 2024.



The screenshot shows the Zenodo record page for the upload "haesleinhuepf/famous_computer_science_quotes: v1". At the top, there is a warning banner: "This is the Zenodo Sandbox instance, used for testing purposes". Below the banner is the Zenodo logo and the record title "haesleinhuepf/famous_computer_science_quotes: v1" by Robert Haase. The record is dated "Published April 7, 2024 | Version v1" and is labeled as "Software" and "Open". It includes a description: "This is the first release. We use it to generate a DOI on Zenodo (sandbox)." and a "Files" section showing a zip file "haesleinhuepf/famous_computer_science_quotes-v1.zip" containing a directory "haesleinhuepf-famous_computer_science_quotes-e03fc6d" with subdirectories ".github" and "workflows", and files "book.yml" (817 Bytes) and ".gitignore" (65 Bytes).

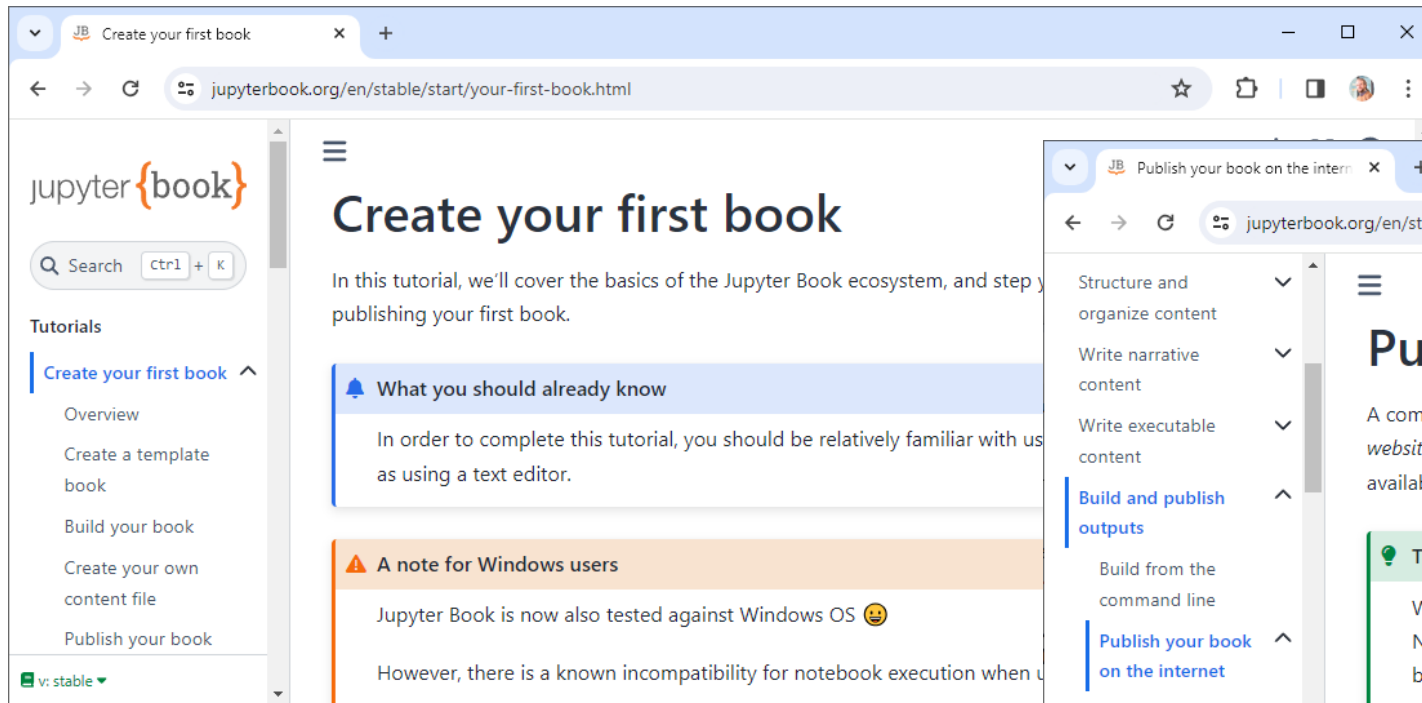
Github pages

- Kostenloses Webhosting, weitgehend automatisch

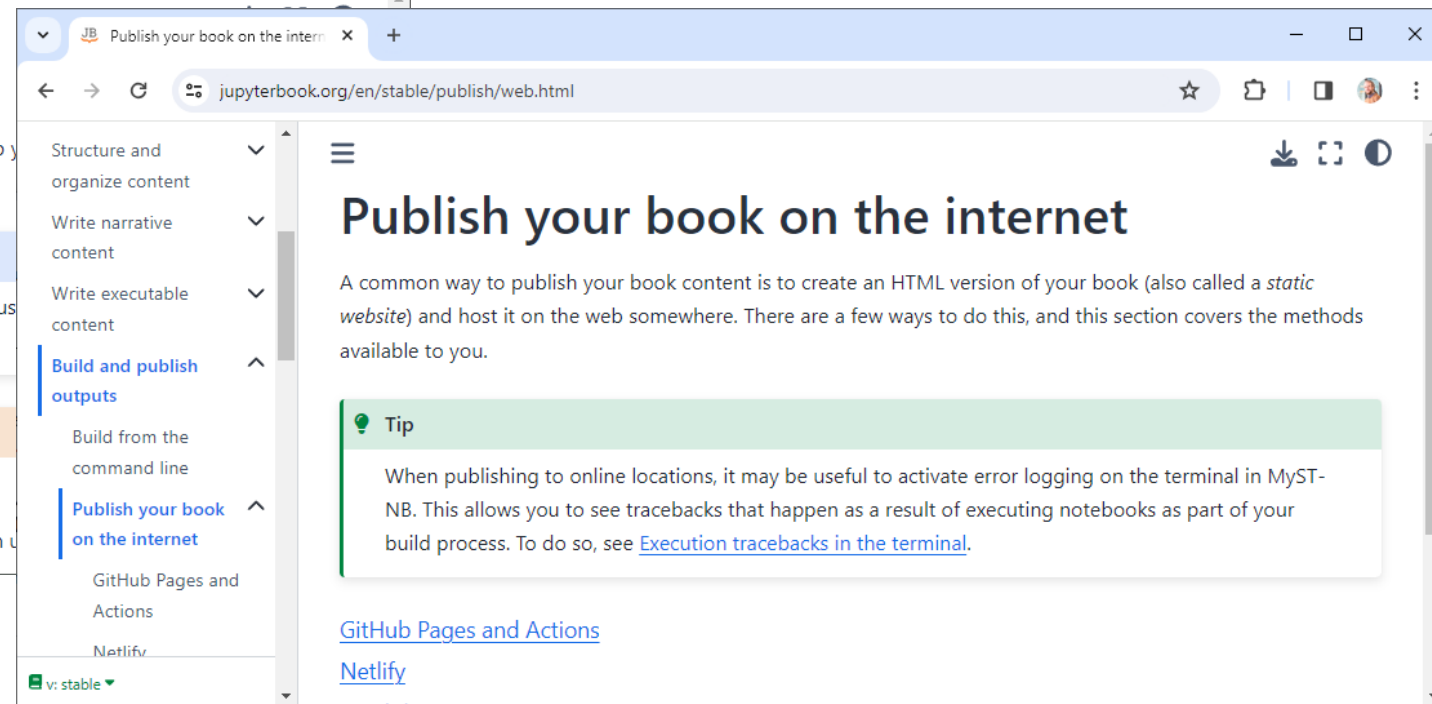


Github pages: Jupyter Books

- Basiert auf dem Jupyter Books Projekt



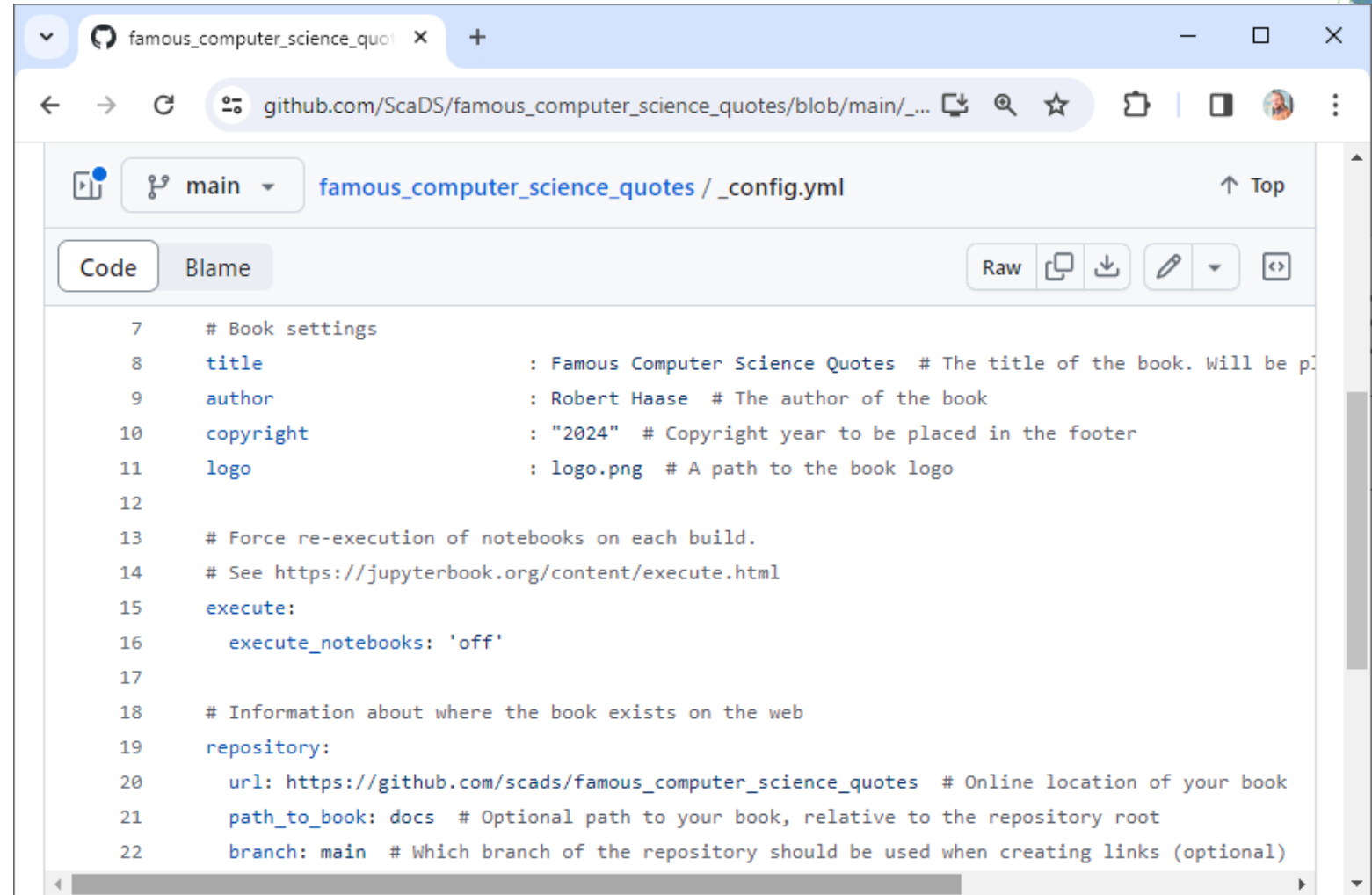
The screenshot shows a web browser window with the URL `jupyterbook.org/en/stable/start/your-first-book.html`. The page title is "Create your first book". The main content area contains the text: "In this tutorial, we'll cover the basics of the Jupyter Book ecosystem, and step by step publishing your first book." Below this, there are two highlighted sections: "What you should already know" (blue background) and "A note for Windows users" (orange background). The left sidebar contains a search bar and a "Tutorials" menu with items like "Create your first book", "Overview", "Create a template book", "Build your book", "Create your own content file", and "Publish your book".



The screenshot shows a web browser window with the URL `jupyterbook.org/en/stable/publish/web.html`. The page title is "Publish your book on the internet". The main content area contains the text: "A common way to publish your book content is to create an HTML version of your book (also called a *static website*) and host it on the web somewhere. There are a few ways to do this, and this section covers the methods available to you." Below this, there is a "Tip" box with a light green background, containing the text: "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](#)." The left sidebar contains a table of contents with items like "Structure and organize content", "Write narrative content", "Write executable content", "Build and publish outputs", "Publish your book on the internet", "GitHub Pages and Actions", and "Netlify".

Github pages: Jupyter Books

- Configuration: `_config.yml`

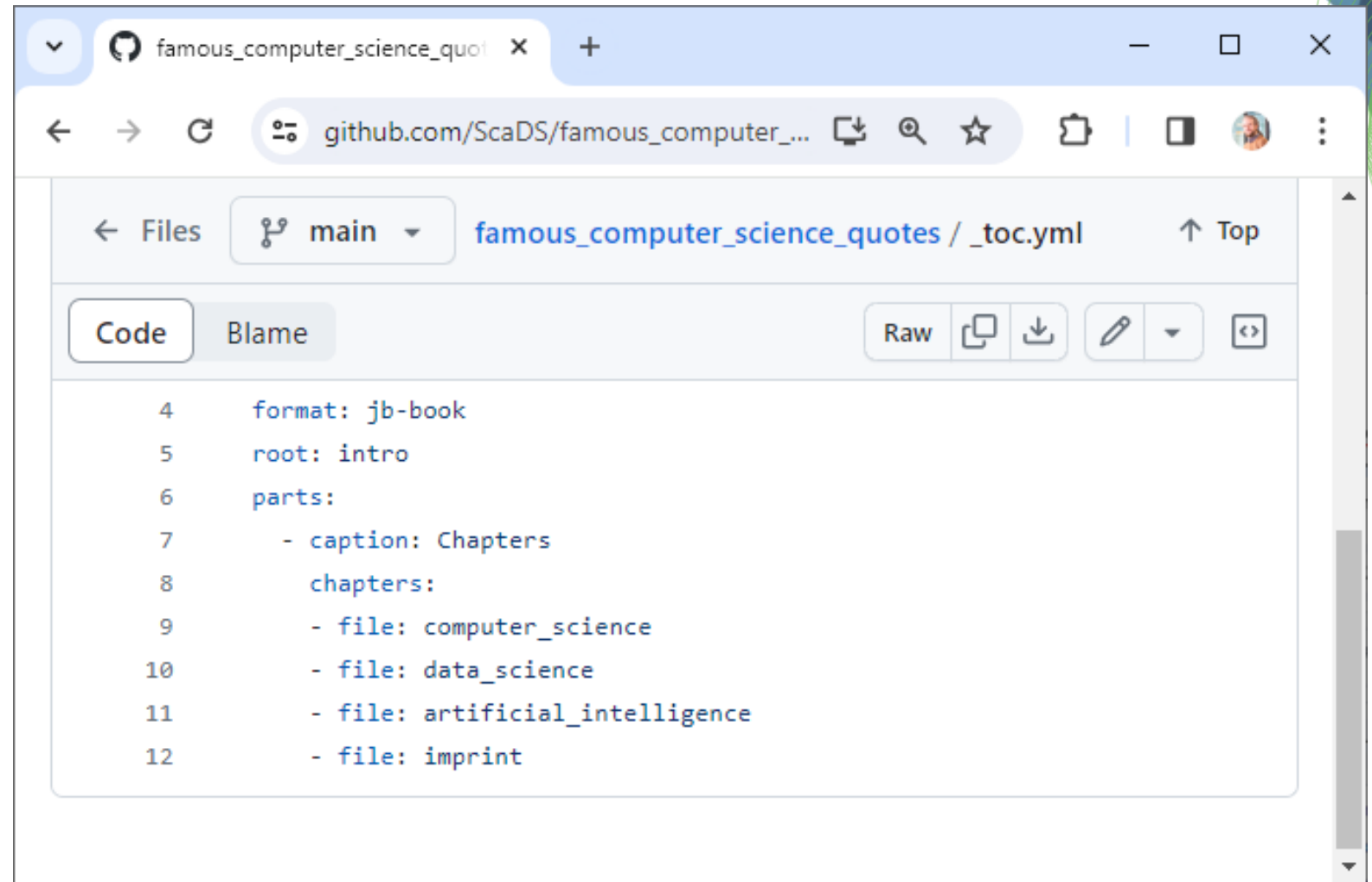


The screenshot shows a web browser displaying the GitHub page for the file `_config.yml` in the repository `famous_computer_science_quotes`. The file content is as follows:

```
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`

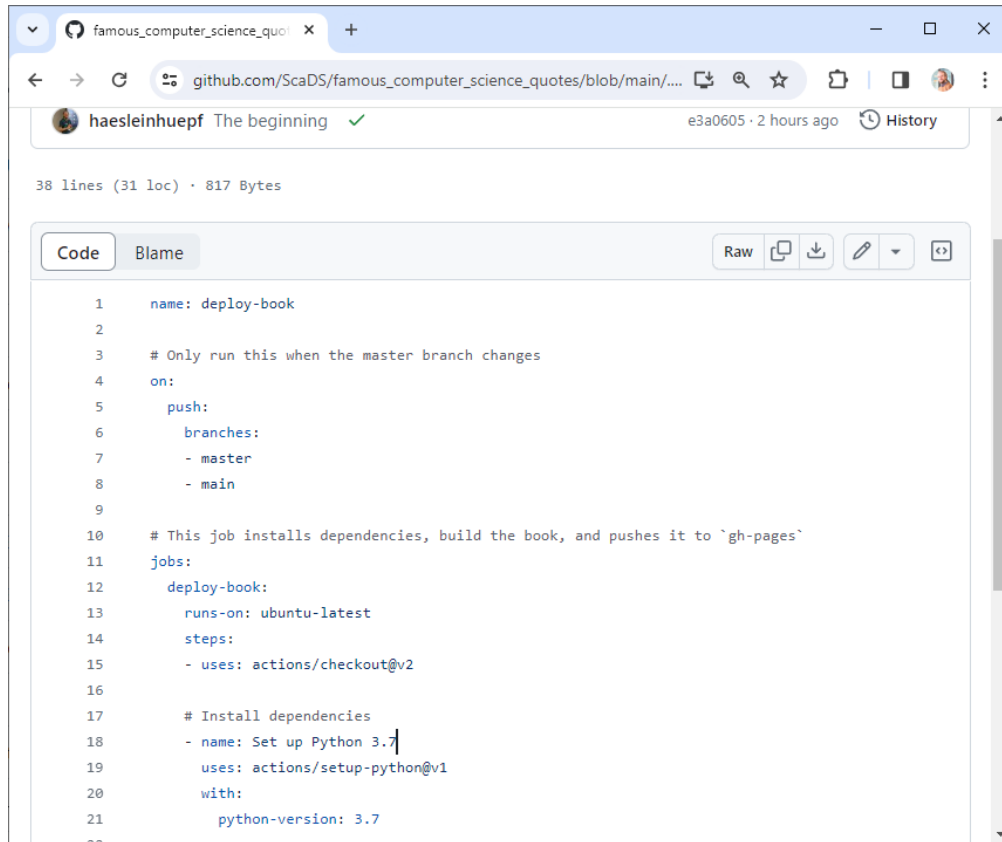


The screenshot shows a web browser window displaying a GitHub repository page. The browser's address bar shows the URL `github.com/ScaDS/famous_computer_...`. The page title is `famous_computer_science_quotes / _toc.yml`. The repository is on the `main` branch. The file `_toc.yml` is selected, and the `Code` tab is active. The content of the file is as follows:

```
4     format: jb-book
5     root: intro
6     parts:
7       - caption: Chapters
8         chapters:
9           - file: computer_science
10          - file: data_science
11          - file: artificial_intelligence
12          - file: imprint
```

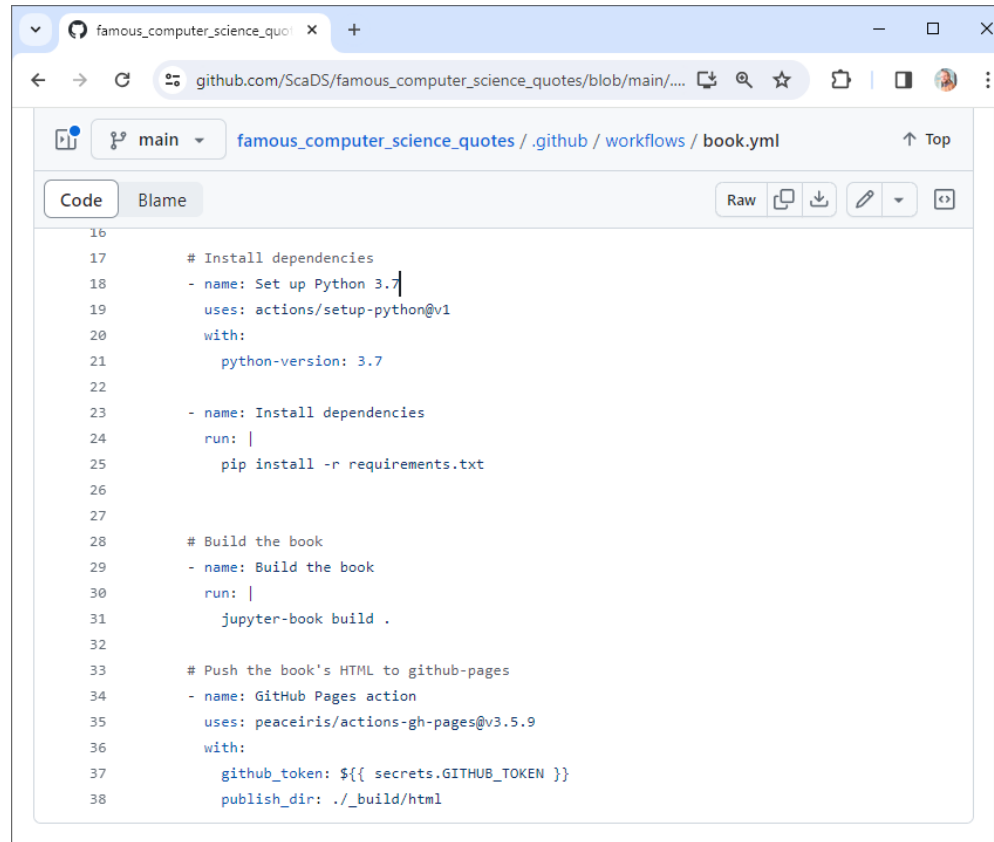
Github pages: Jupyter Books

- Behind the scenes: Github workflows



A screenshot of a web browser showing a GitHub repository page. The URL is `github.com/ScaDS/famous_computer_science_quotes/blob/main/...`. The page shows a workflow file named `book.yml` with 38 lines of code. The code is displayed in a light blue editor with line numbers on the left. The workflow is triggered on push to the master or main branch and consists of two jobs: `deploy-book` and `install-dependencies`. The `deploy-book` job runs on the latest Ubuntu and uses the `actions/setup-python@v1` action to set up Python 3.7, followed by `actions/checkout@v2`. The `install-dependencies` job also runs on the latest Ubuntu and uses `actions/setup-python@v1` to set up Python 3.7, followed by `pip install -r requirements.txt`.

```
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
```

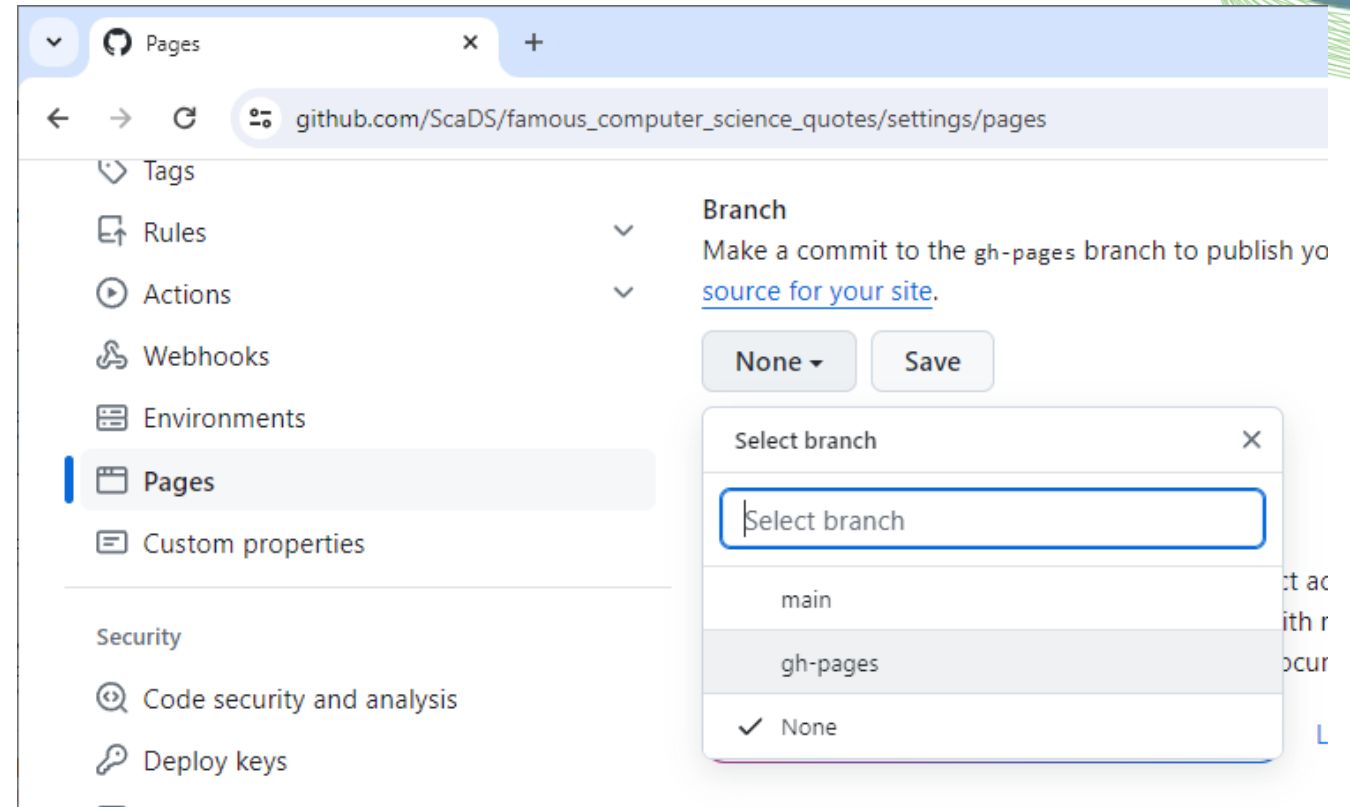
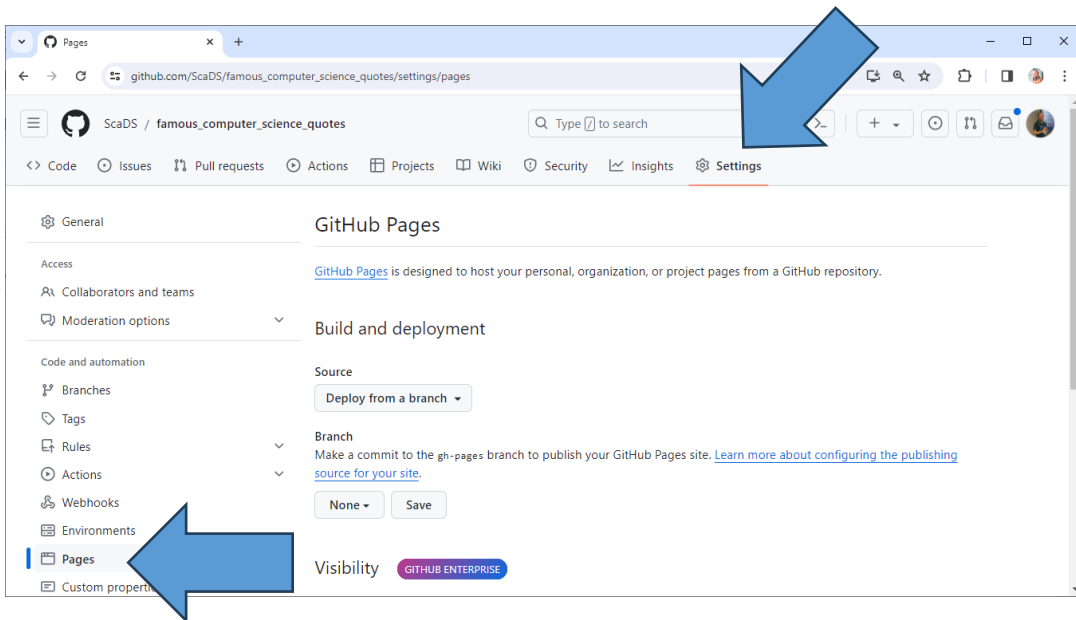


A screenshot of a web browser showing a GitHub repository page. The URL is `github.com/ScaDS/famous_computer_science_quotes/blob/main/...`. The page shows a workflow file named `book.yml` with 38 lines of code. The code is displayed in a light blue editor with line numbers on the left. The workflow is triggered on push to the master or main branch and consists of two jobs: `install-dependencies` and `build-book`. The `install-dependencies` job runs on the latest Ubuntu and uses the `actions/setup-python@v1` action to set up Python 3.7, followed by `pip install -r requirements.txt`. The `build-book` job also runs on the latest Ubuntu and uses `actions/setup-python@v1` to set up Python 3.7, followed by `jupyter-book build .`. The `push-gh-pages` job runs on the latest Ubuntu and uses the `peaceiris/actions-gh-pages@v3.5.9` action to push the book's HTML to GitHub Pages.

```
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
28 # Build the book
29 - name: Build the book
30   run: |
31     jupyter-book build .
32
33 # Push the book's HTML to github-pages
34 - name: GitHub Pages action
35   uses: peaceiris/actions-gh-pages@v3.5.9
36   with:
37     github_token: ${ secrets.GITHUB_TOKEN }
38     publish_dir: ./_build/html
```

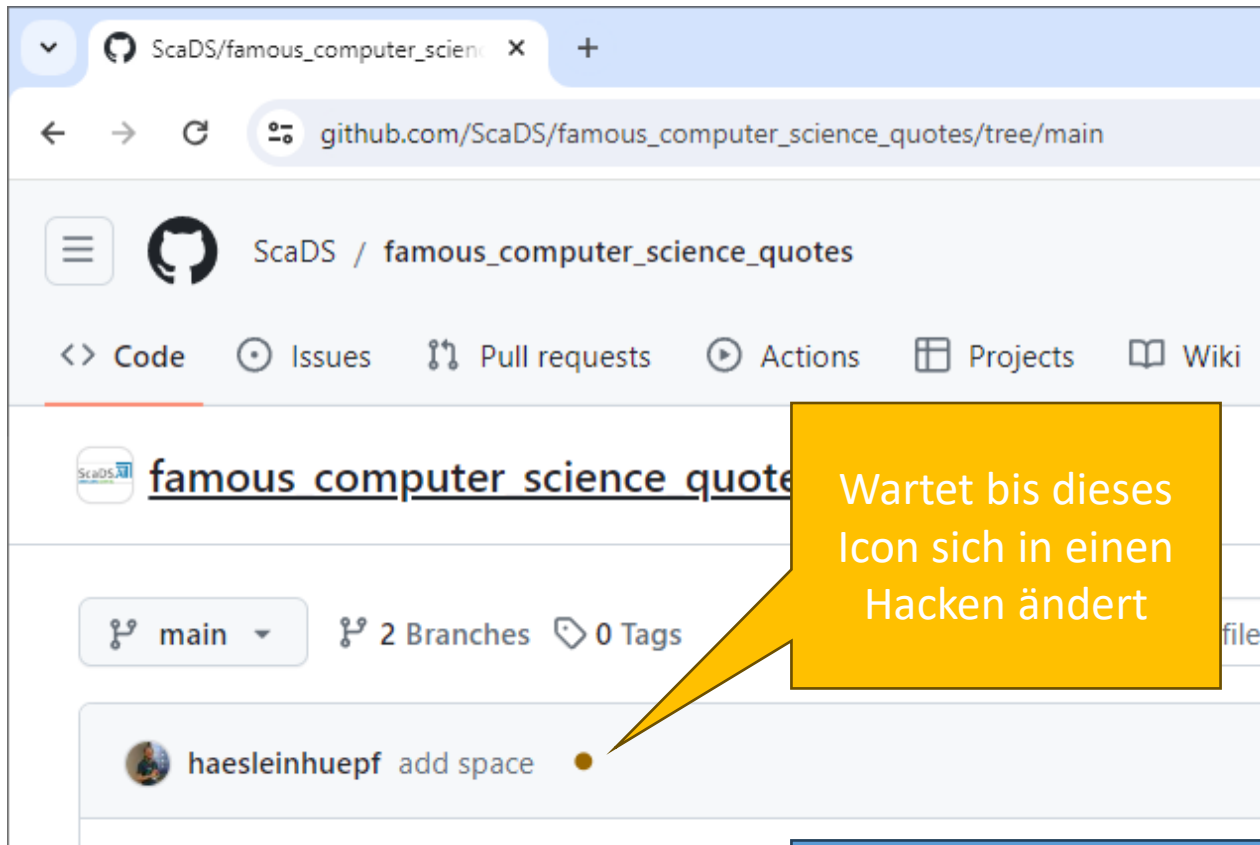
Github pages

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



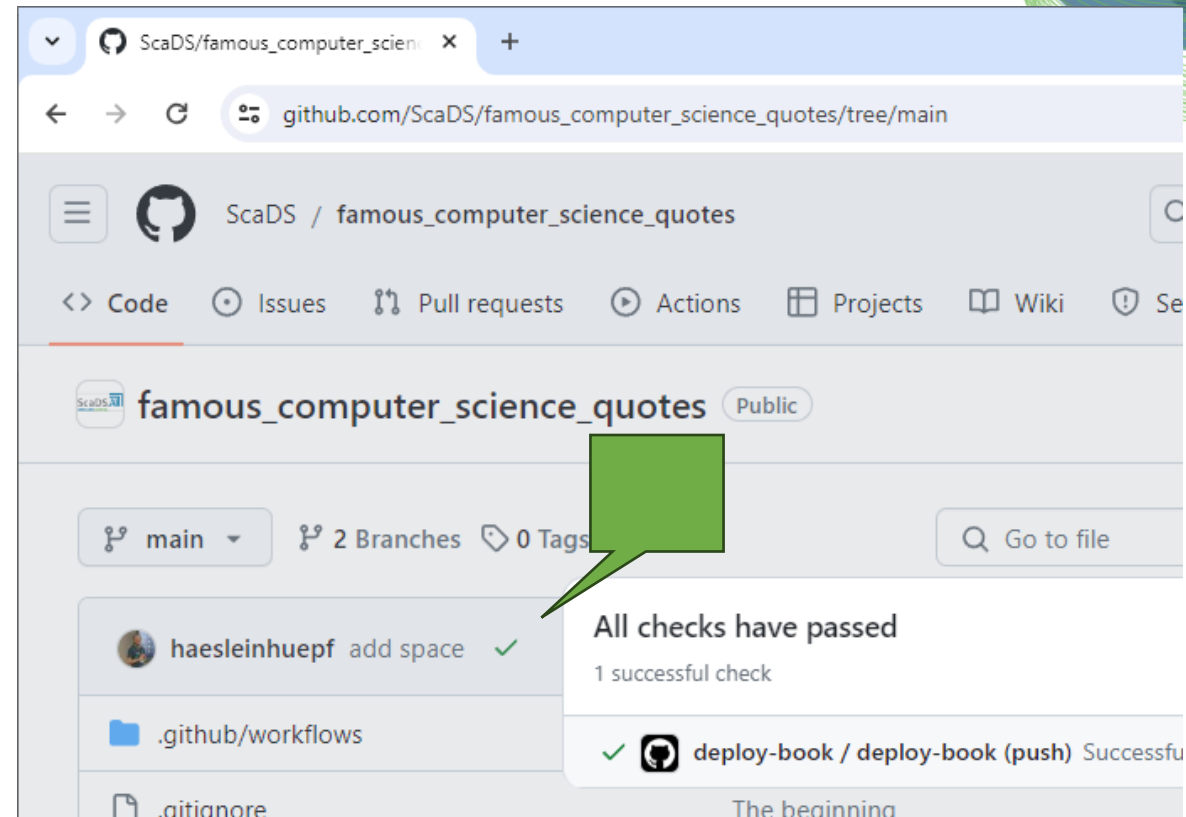
Github pages

- Warten 😊



The screenshot shows the GitHub repository page for 'ScaDS / famous_computer_science_quotes'. The commit 'haesleinhuepf add space' is visible, and a yellow callout bubble points to a small orange dot icon next to the commit, indicating a pending status.

Wartet bis dieses Icon sich in einen Hacken ändert



The screenshot shows the same GitHub repository page, but the commit 'haesleinhuepf add space' now has a green checkmark icon next to it, indicating that all checks have passed. A green callout bubble points to the checkmark.

All checks have passed
1 successful check
deploy-book / deploy-book (push) Successful

Das wird automatisch passieren,
wenn man eine Datei ändert

Zenodo integration & Github pages

- In case this was too fast...

The screenshot shows a web browser window with the URL coderefinery.github.io/github-without-command-line/doi/#step-2-activate-the-repository-on-zenodo-sandbox. The page is titled "Step 2: Activate the repository on Zenodo (sandbox)".

GitHub without the command line

Search docs

THE LESSON

- Basics and motivation
- Creating repositories using the web interface
- Creating repositories using GitHub Desktop
- Contributing to existing repositories using pull requests
- How to organize a group's work

Making your project citable

- Step 1: Prepare an example repository
- Step 2: Activate the repository on Zenodo (sandbox)**
- Step 3: Create a "release" and get a DOI
- Step 4: Add a DOI badge to your repository

Step 2: Activate the repository on Zenodo (sandbox)

We will exercise in the Zenodo sandbox

We will practice on <https://sandbox.zenodo.org/> and not on the "real" <https://zenodo.org/> to make sure we do not create "real" DOIs which we cannot remove.

- Visit <https://sandbox.zenodo.org/account/settings/github/>:

Home / Account / GitHub

Settings

- Profile
- Change password
- Security
- Linked accounts
- Applications
- Shared links
- GitHub**

GitHub Repositories (updated now) Sync now ...

Get started

- 1 Flip the switch**
Select the repository you want to preserve, and toggle the switch below to turn on automatic preservation of your software.
- 2 Create a release**
Go to GitHub and create a release. Zenodo will automatically download a .zip-ball of each new release and register a DOI.
- 3 Get the badge**
After your first release, a DOI badge that you can include in GitHub README will appear next to your repository below.
DOI: [10.5281/zenodo.8475](https://doi.org/10.5281/zenodo.8475) (example)

- Select the repository you wish to preserve:

Praktikum

Robert Haase

GEFÖRDERT VOM



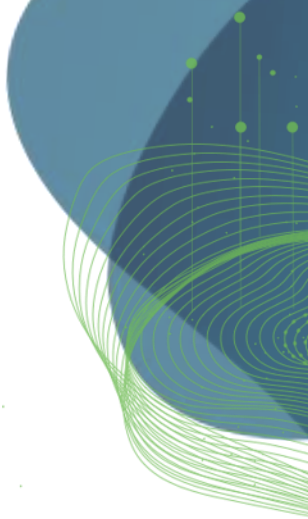
Bundesministerium
für Bildung
und Forschung



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.

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

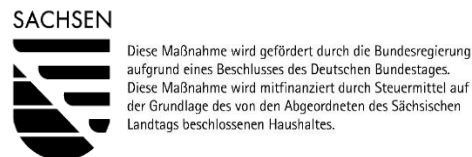


BiAPoL team

- 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



Some Figures were generated using

