

#OpenScience

#OpenSource

Good practice in scientific programming

Robert Haase

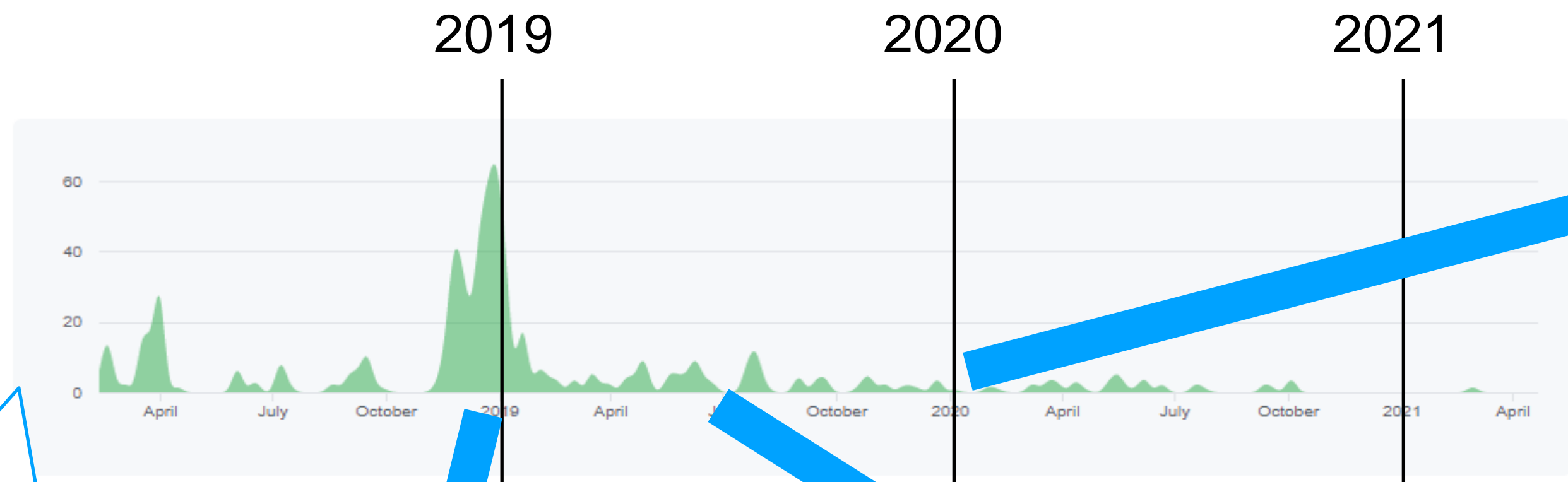
October 2022

Accessible

Reusable

Sustainable

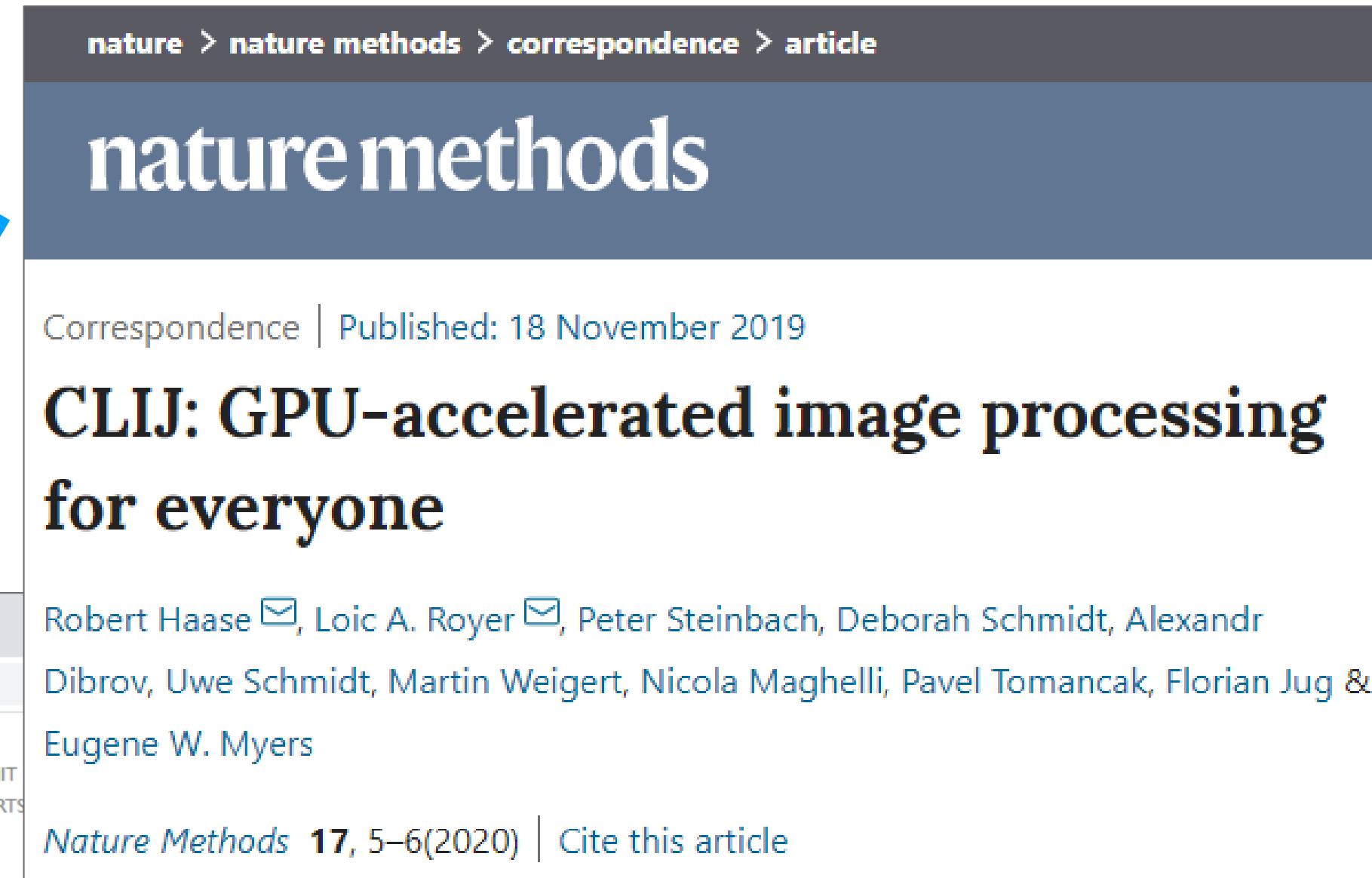
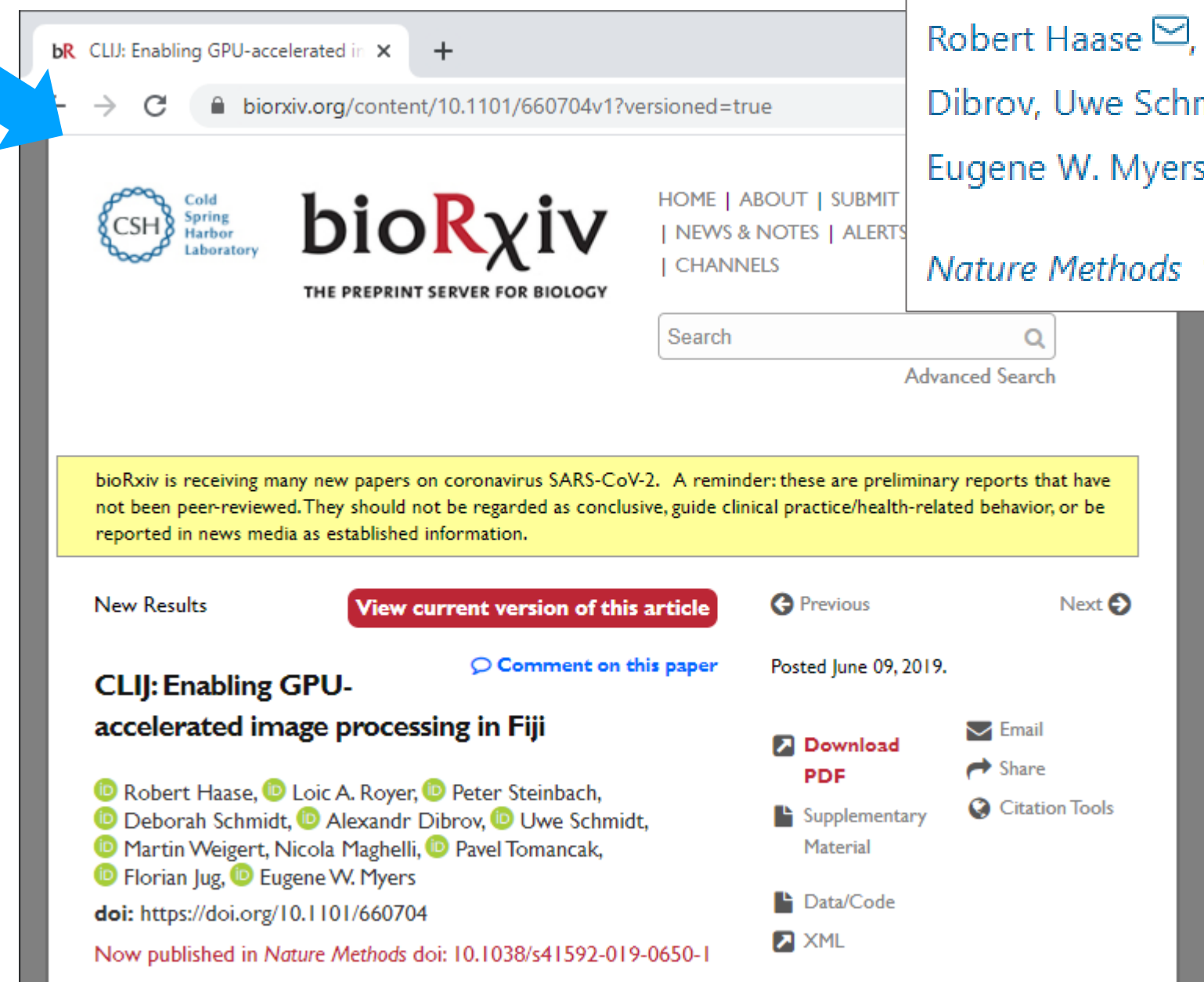
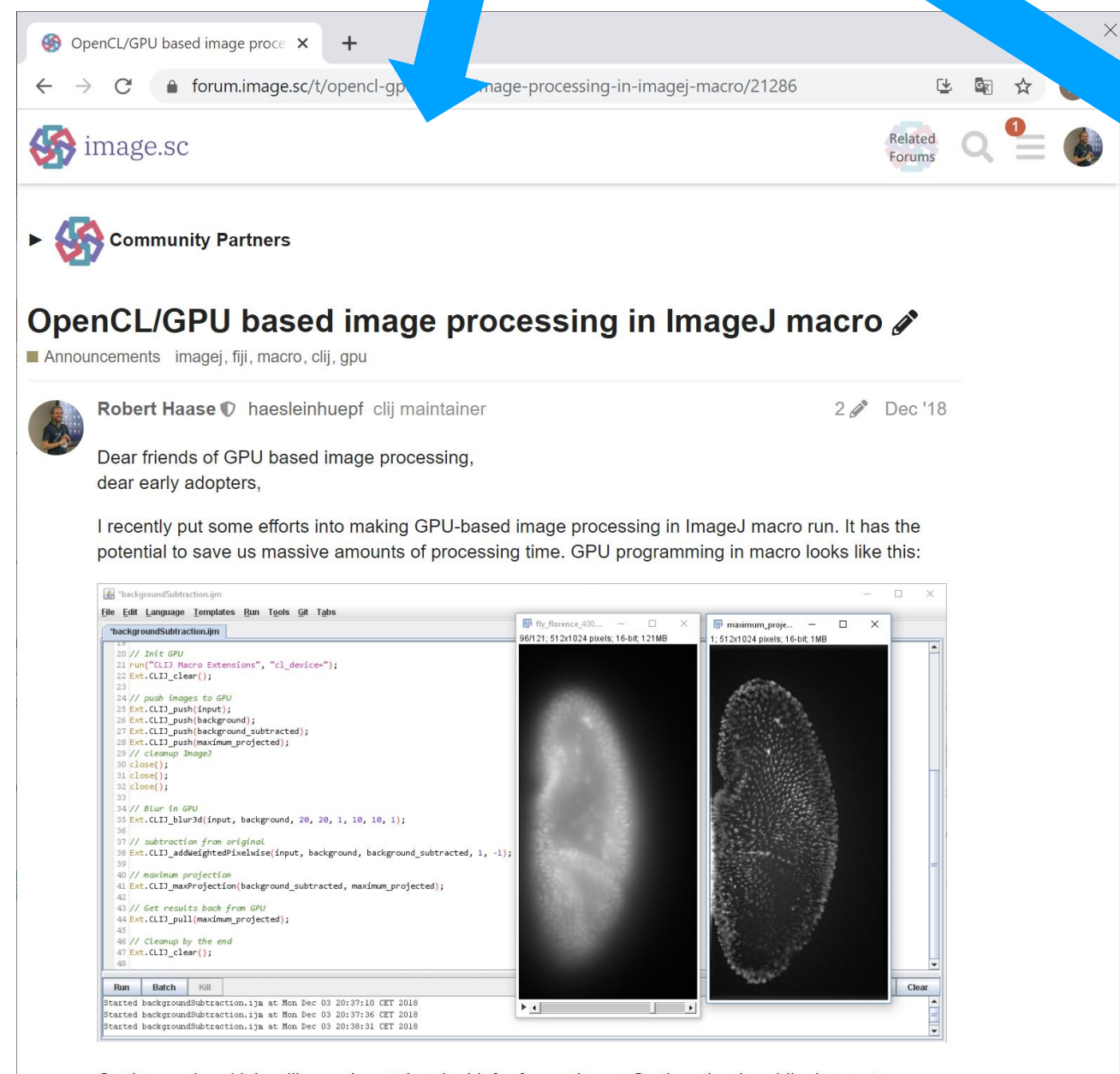
An open science story



Nov. 2017: I took over microscopy control software from Loic and “found” a GPU-accelerated image processing part



Loic A. Royer
(CZ Biohub)
@loicaroyer



<https://github.com/clij/clij/graphs/contributors> <https://forum.image.sc/t/opencl-gpu-based-image-processing-in-imagej-macro/21286> <https://www.biorxiv.org/content/10.1101/660704v1?versioned=true>
<https://www.nature.com/articles/s41592-019-0650-1>



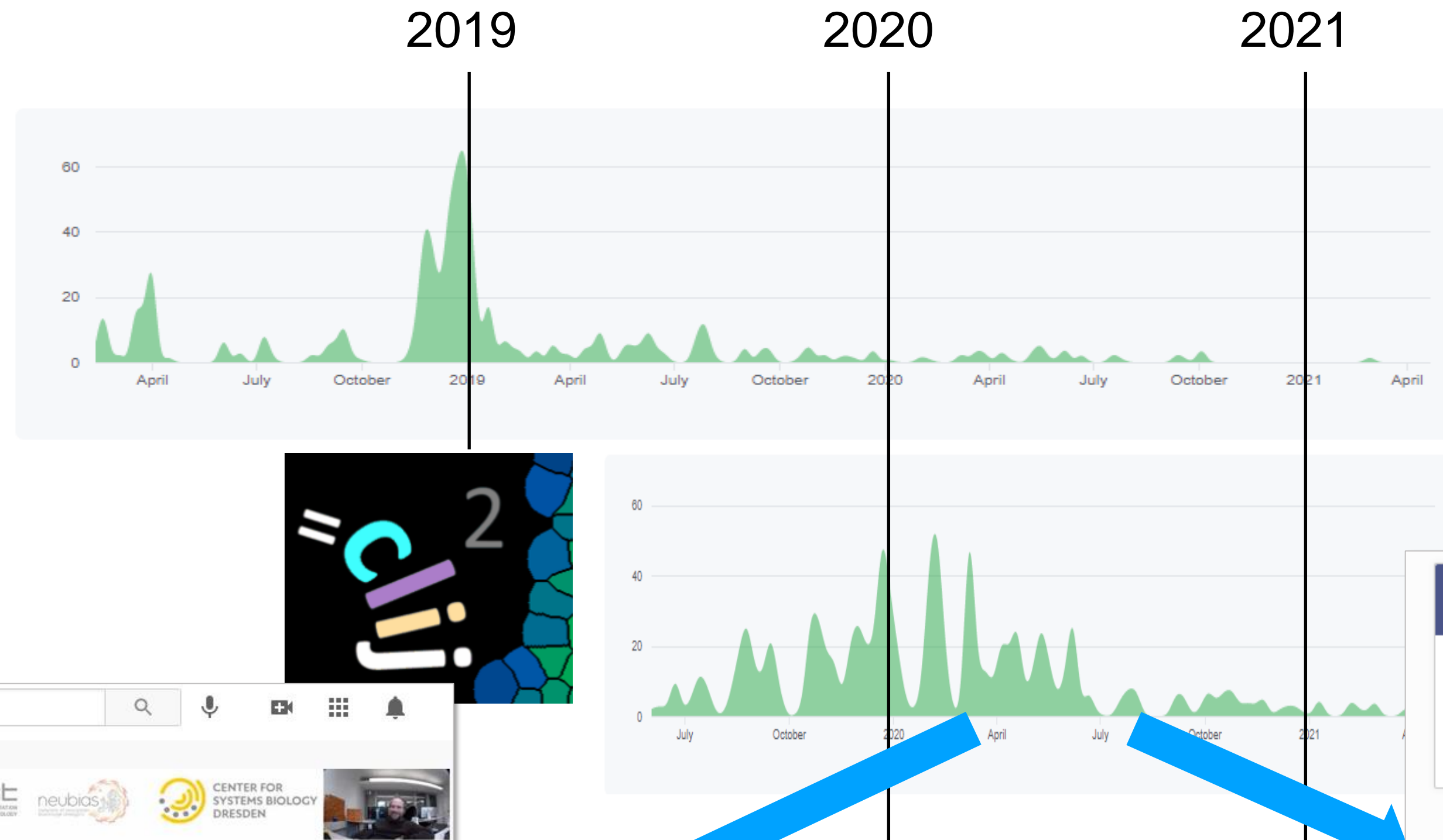
**TECHNISCHE
UNIVERSITÄT
DRESDEN**

DRESDEN
concept



@haesleinhuepf
@PoLDresden

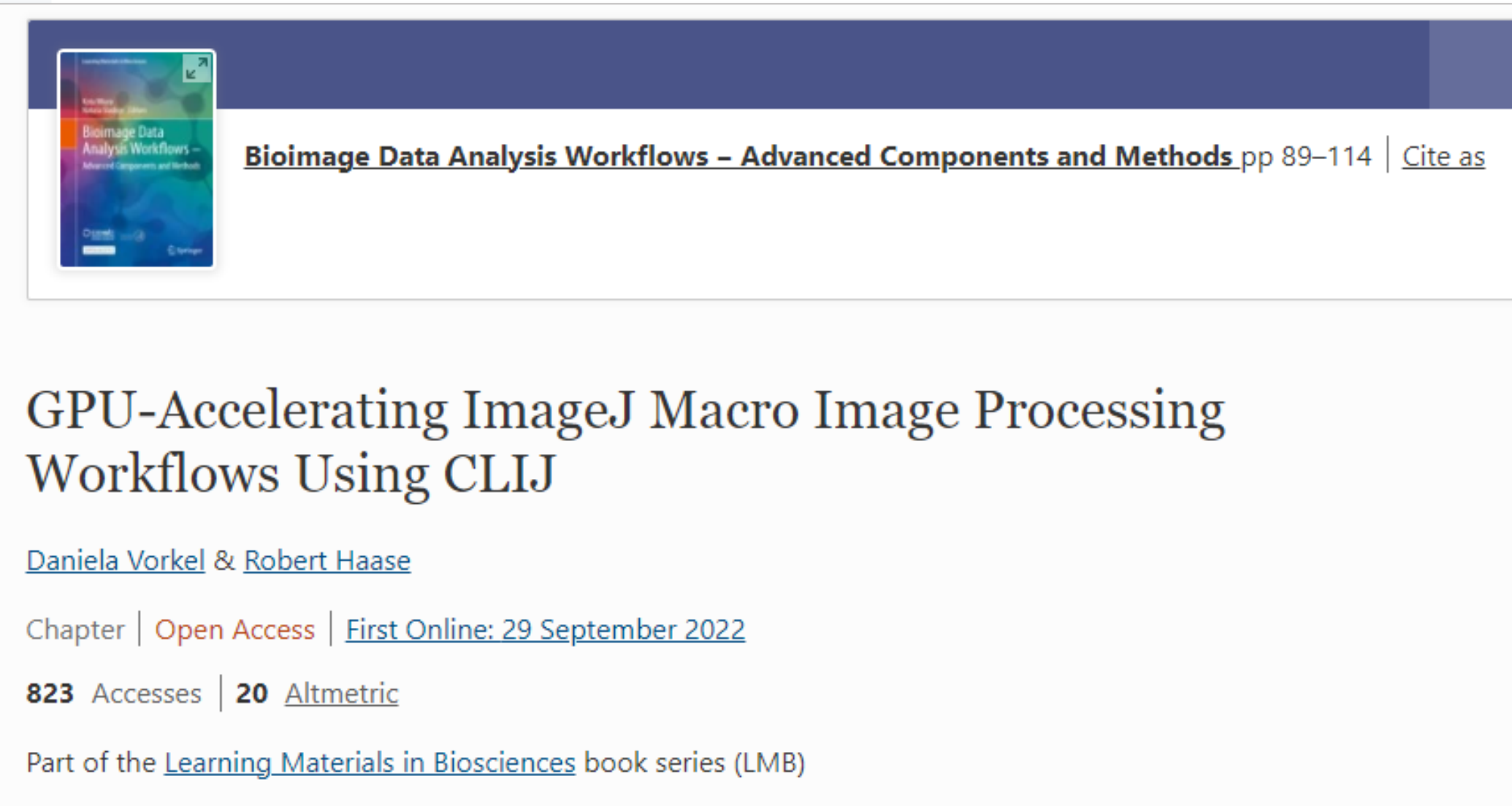
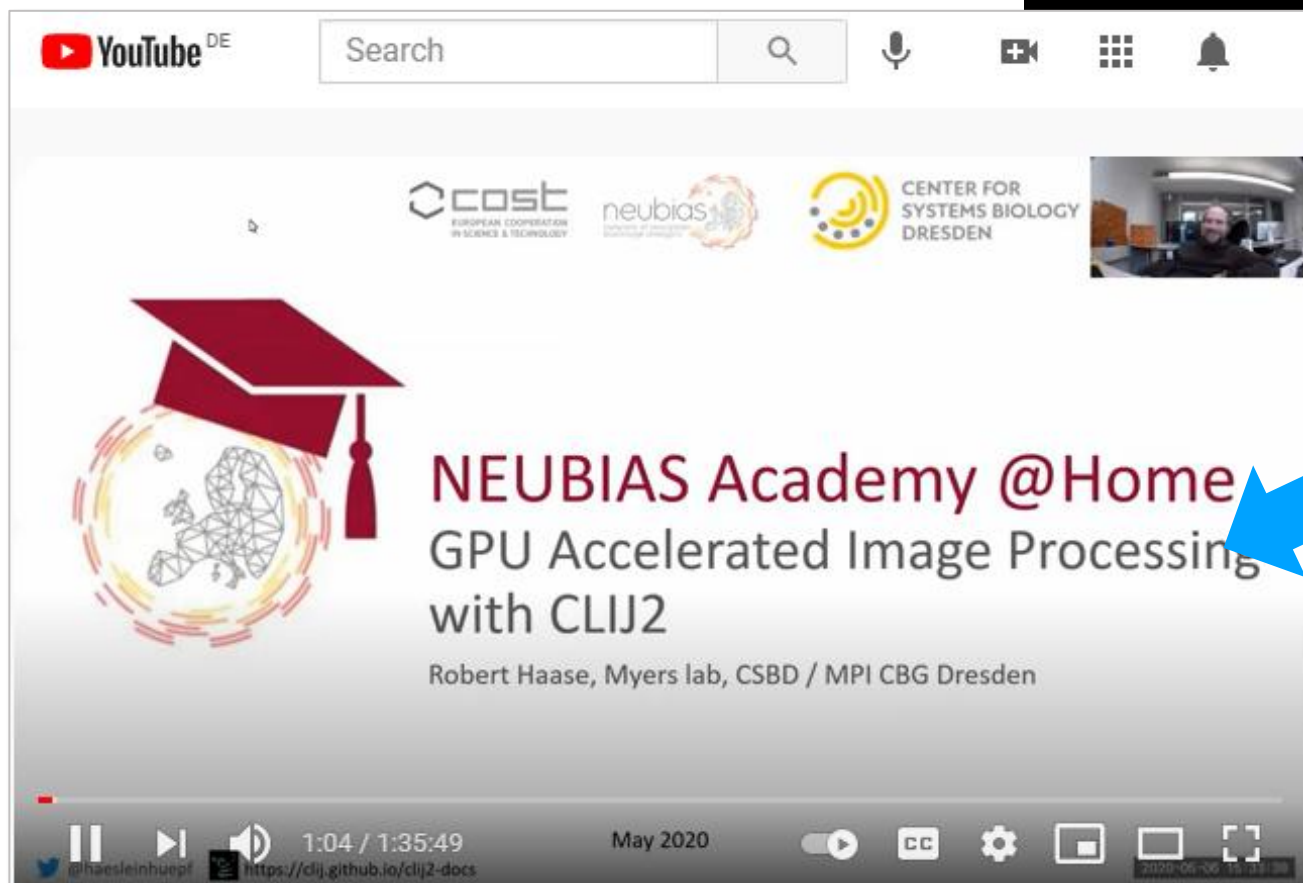
An open science story



Best co-author



Dani Vorkel
(Myers lab)
@happifocus

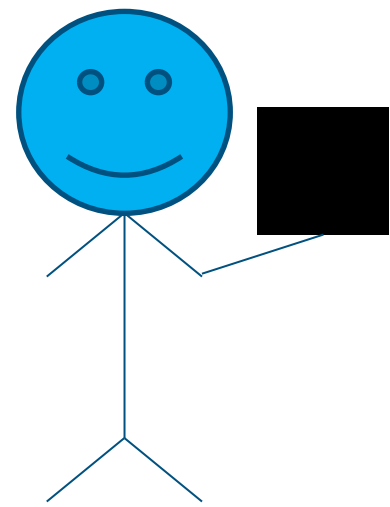


Openness of software / projects

Communication
is key!

Choose your project's level wisely, and communicate it clearly

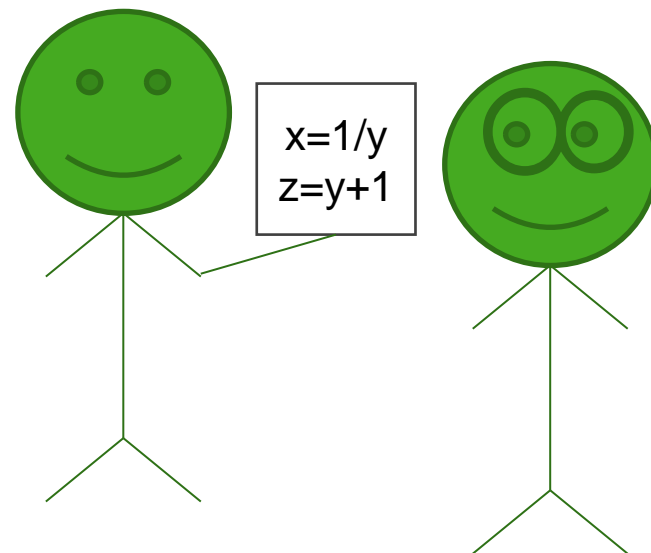
Closed source



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

Hardware device
drivers

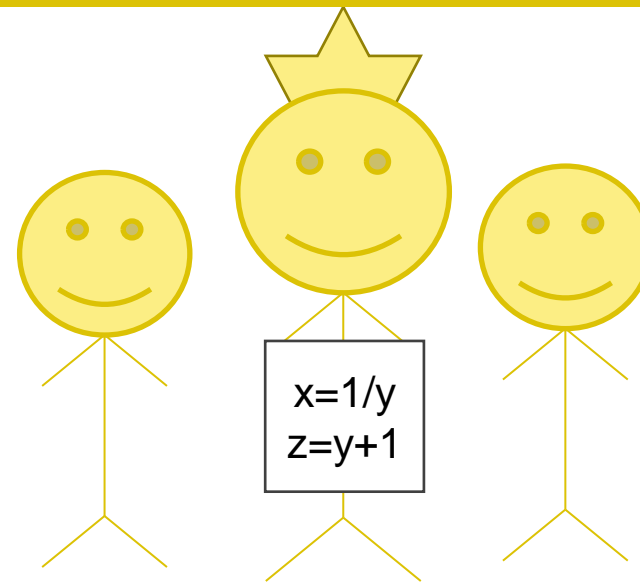
Open source



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

Custom image
analysis scripts

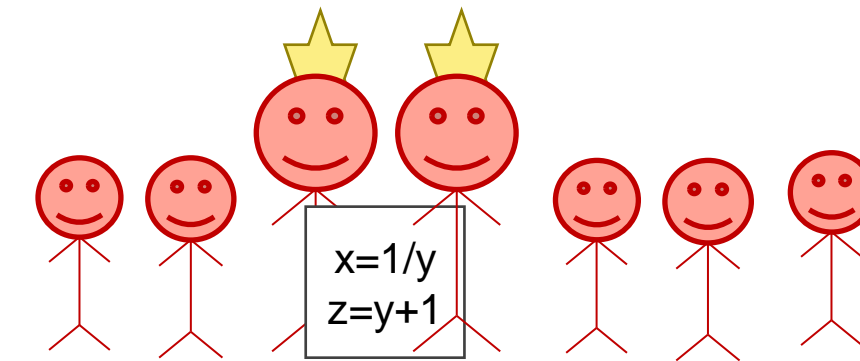
Benevolent dictatorship



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

TrackMate, SNT,
MorpholibJ, CLIJ

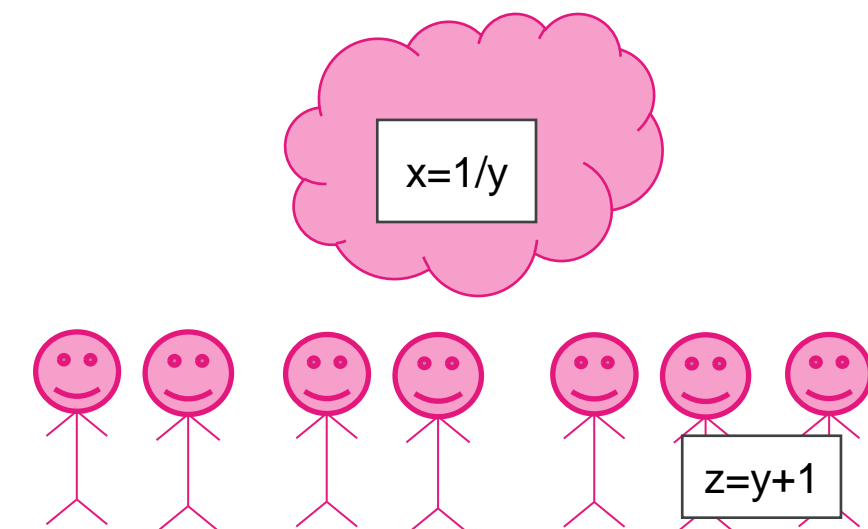
Community driven



- 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

Target audience

Documentation should tell who is the target audience and how far it is developed

Communication is key!

CLIJ2

GPU accelerated image processing for everyone

CLIJ2 home

Installation in Fiji | Installation in Icy | Installation in Matlab

Reference | Cheat sheets | Source code

This project is maintained by haesleinhuepf

Imprint

CLIJ2: GPU-accelerated image processing for everyone

Introduction

CLIJ2 is a GPU-accelerated image processing library for ImageJ/Fiji, Icy, Matlab and Java. It comes with hundreds of operations for filtering, binarizing, labeling, measuring in images, projections, transformations and mathematical operations for images. While most of these are classical image processing operations, CLIJ2 also allows performing operations on matrices potentially representing neighborhood relationships between cells and pixels.

py-clesperanto

forum 18 topics | website up | pypi v0.17.3 | contributors 7 | downloads 237k/month | Stars 80

Forks 19 | license BSD-3-Clause | python 3.7 | 3.8 | 3.9 | tests | codecov 93% | status alpha

DOI 10.5281/zenodo.6757655

py-clesperanto is a prototype for clesperanto - a multi-platform multi-language framework for GPU-accelerated image processing. We mostly use it in the life sciences for analysing 3/4-dimensional microscopy data, e.g. as we face it developmental biology when segmenting cells and studying their individual properties as well as properties of compounds of cells forming tissues.

Image data source: Daniela Vorkel, Myers lab, MPI-CBG, rendered using napari

clesperanto uses OpenCL kernels from CLIJ. Since version 0.11.1 py-clesperanto comes with a yet experimental cupy-based CUDA backend.

For users convenience, there are code generators available for napari and Fiji. Also check out the napari workflow optimizer for semi-automatic parameter tuning of clesperanto-functions.

Note: This is a decommissioned napari plugin

This project is discontinued. You can still install and use it (version 0.1.2 was last tested with napari 0.4.12) but this project is no longer maintained and supported.

```
conda install pyopenc1
pip install napari-mahotas-image-processing==0.1.2
```

Check out these napari plugins which have similar functionality:

- napari-zelda
- napari-allencell-segmenter
- napari-simpleitk-image-processing
- napari-segment-blobs-and-things-with-membranes
- napari_pyclesperanto_assistant

napari-mahotas-image-processing (n-mahotas)

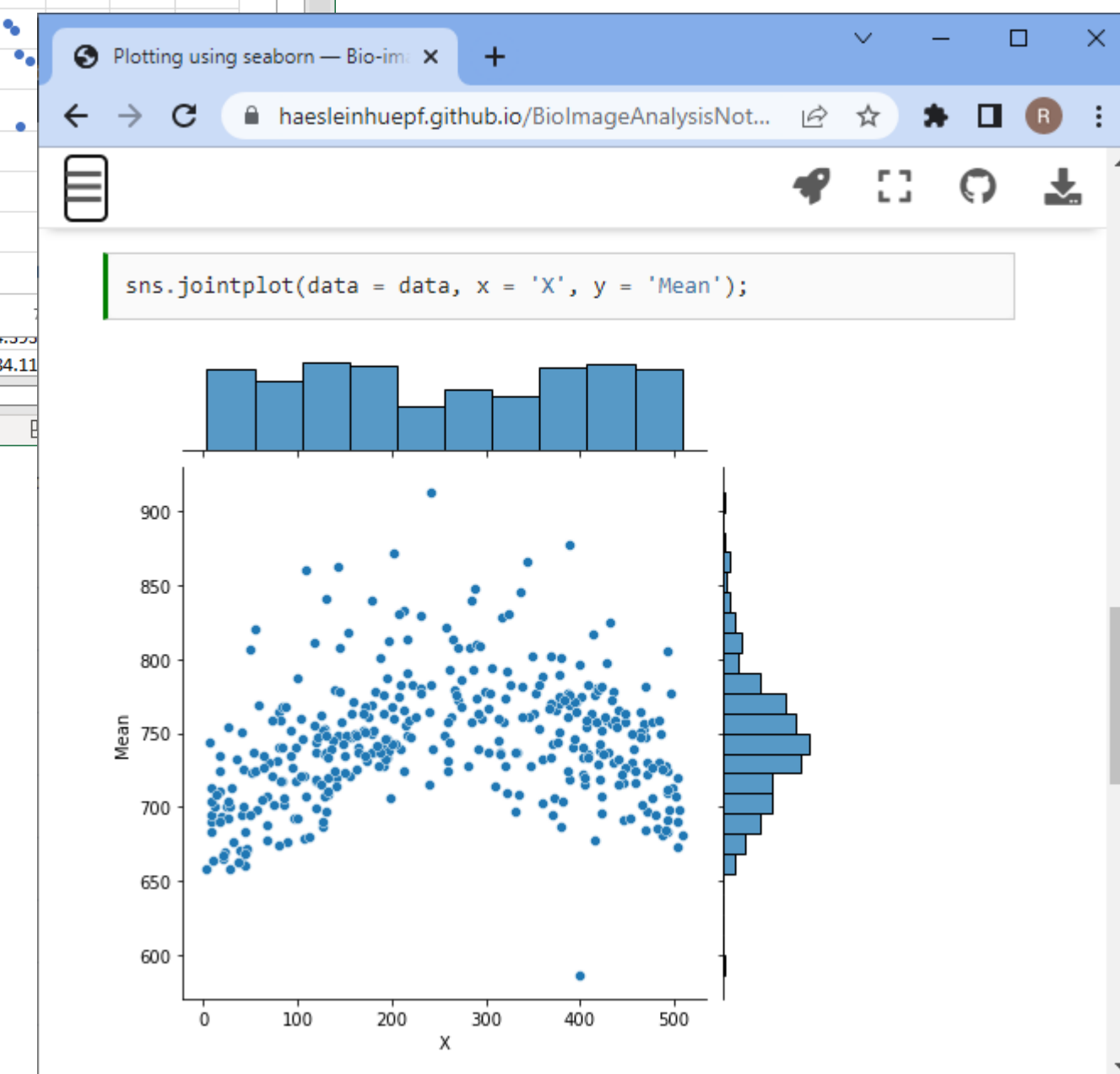
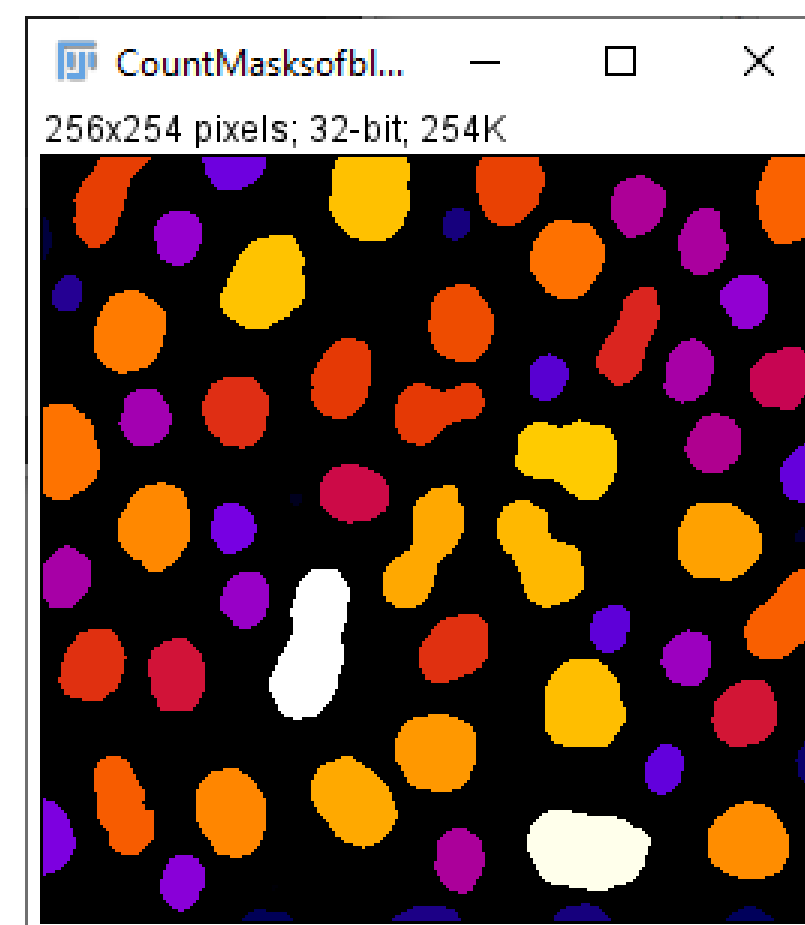
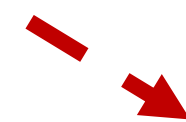
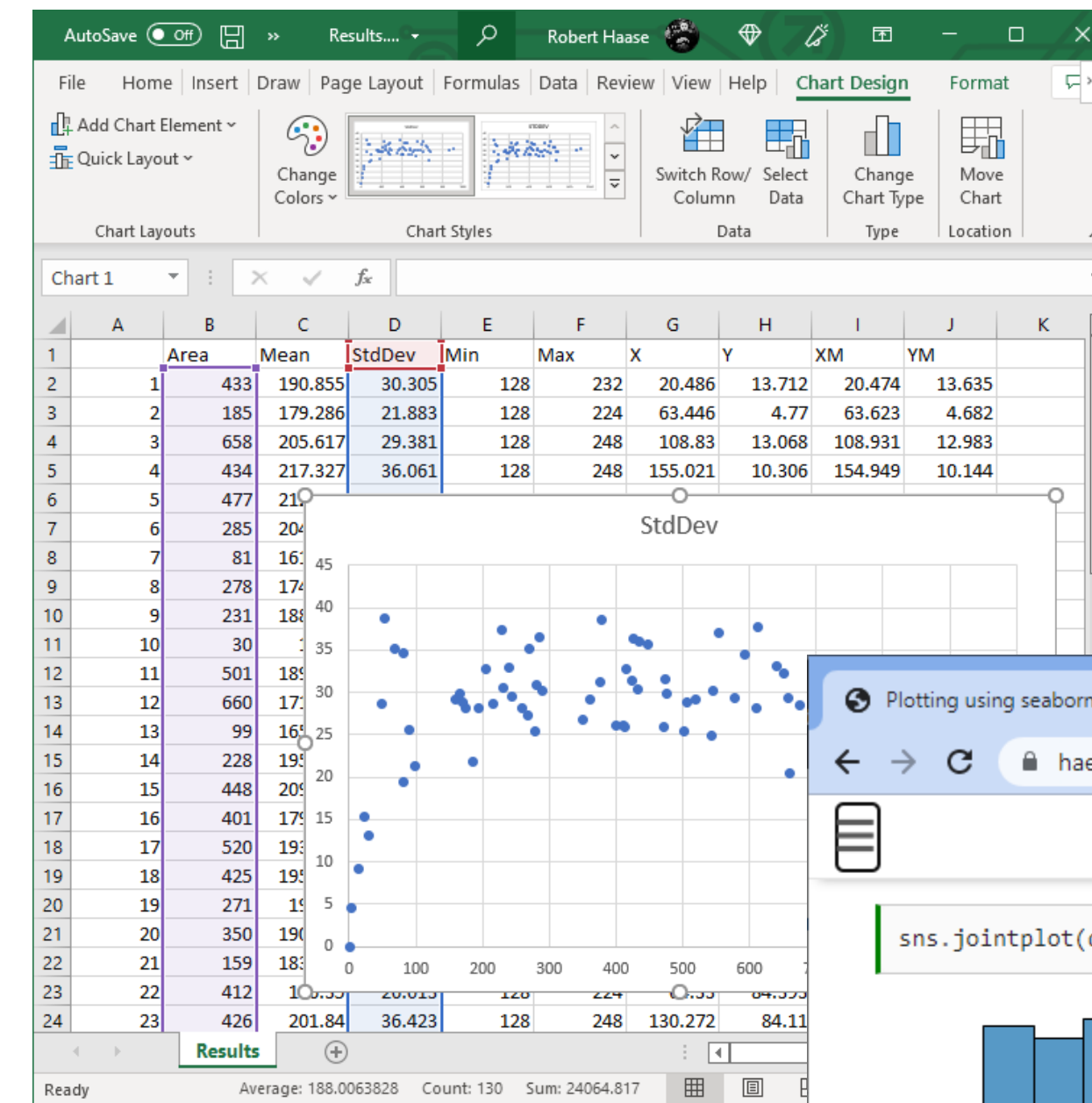
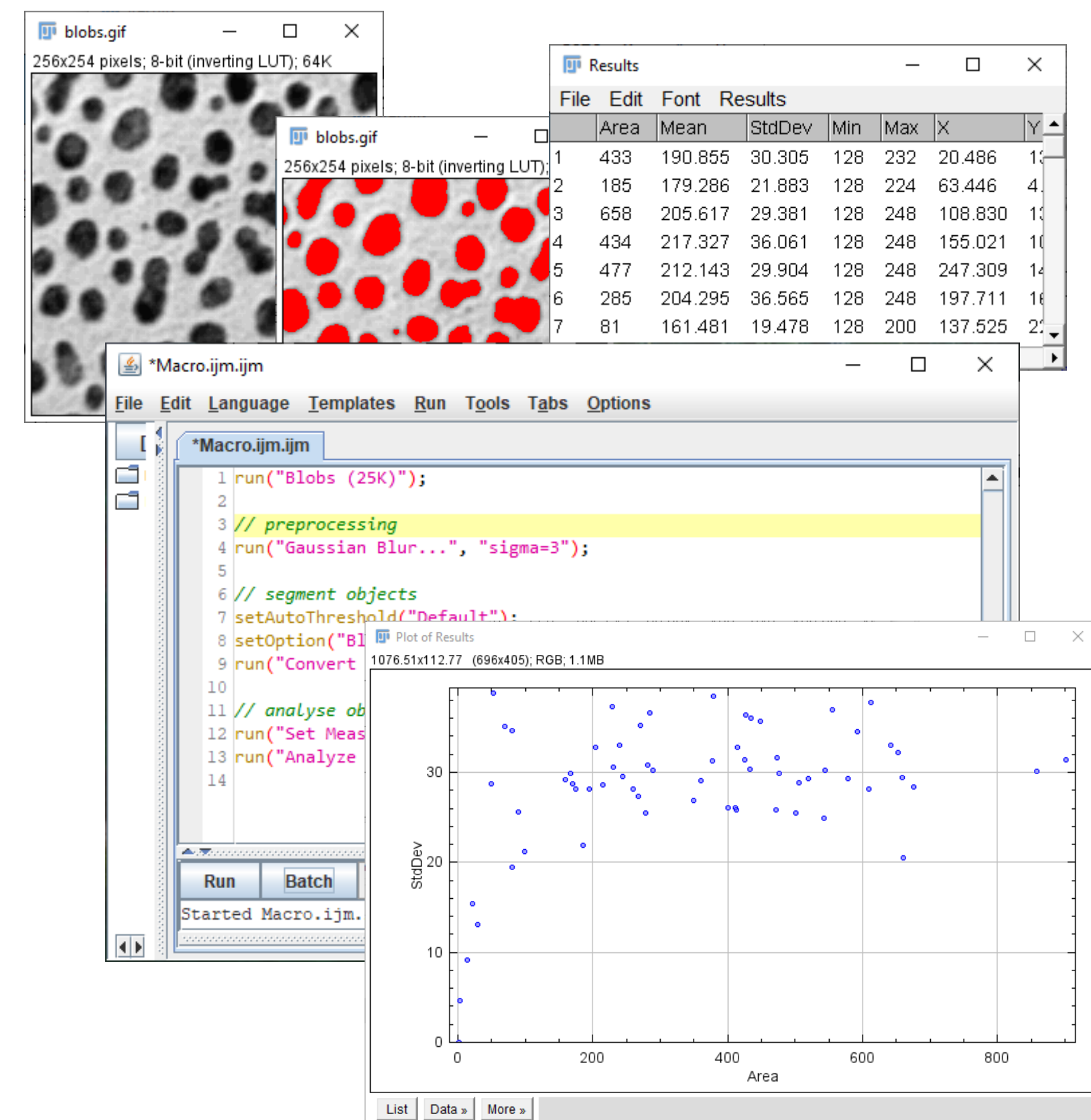
license BSD-3-Clause | pypi v0.1.3 | python 3.7 | 3.8 | 3.9 | tests failing | codecov unknown

* napari hub plugin not found

Image processing based using the Mahotas library in napari

Image data science is cross-disciplinary

... and uses diverse toolsets

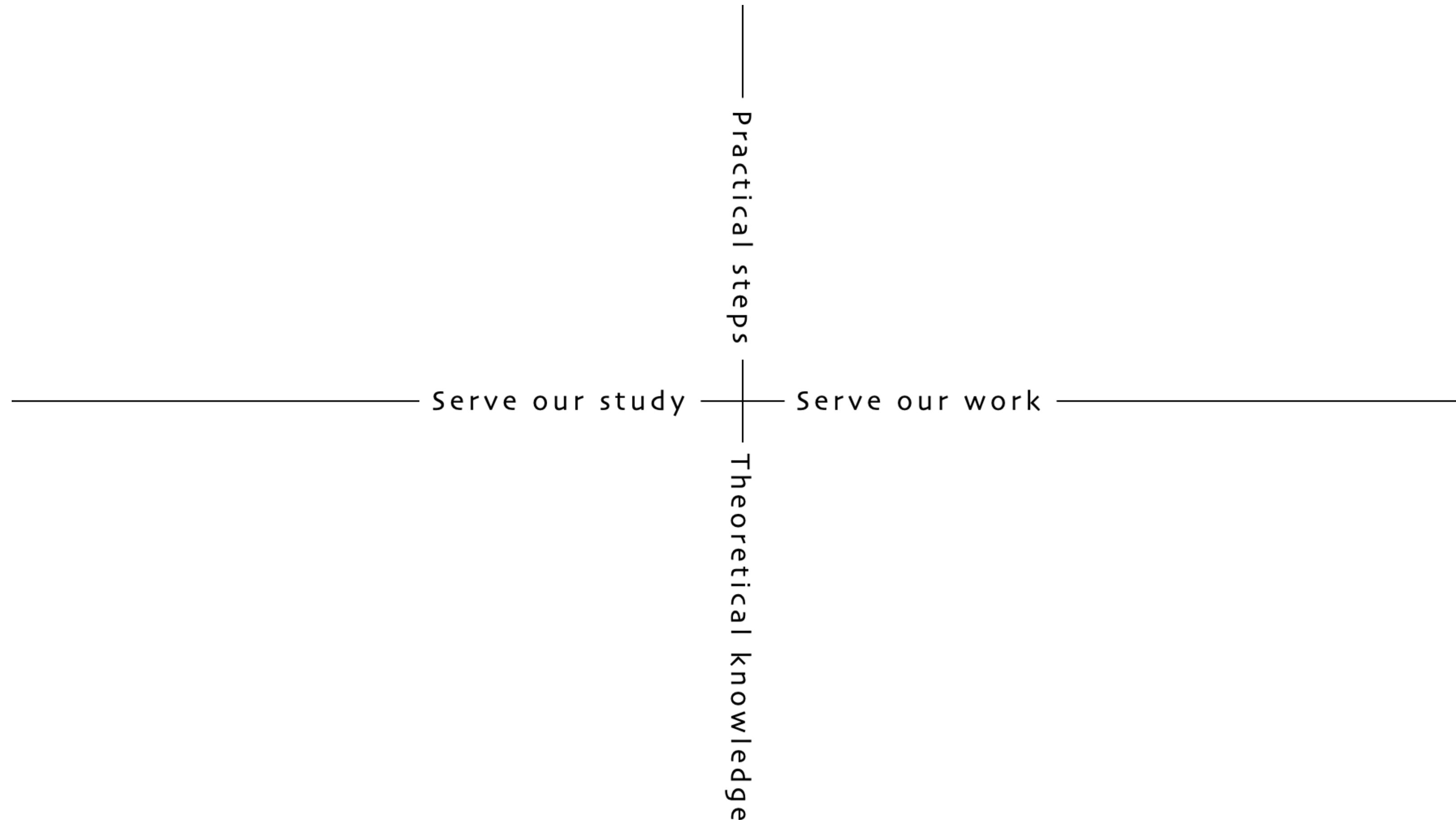


Documentation is key

... to make code reusable

Documentation best-practices

The four modes of documentation

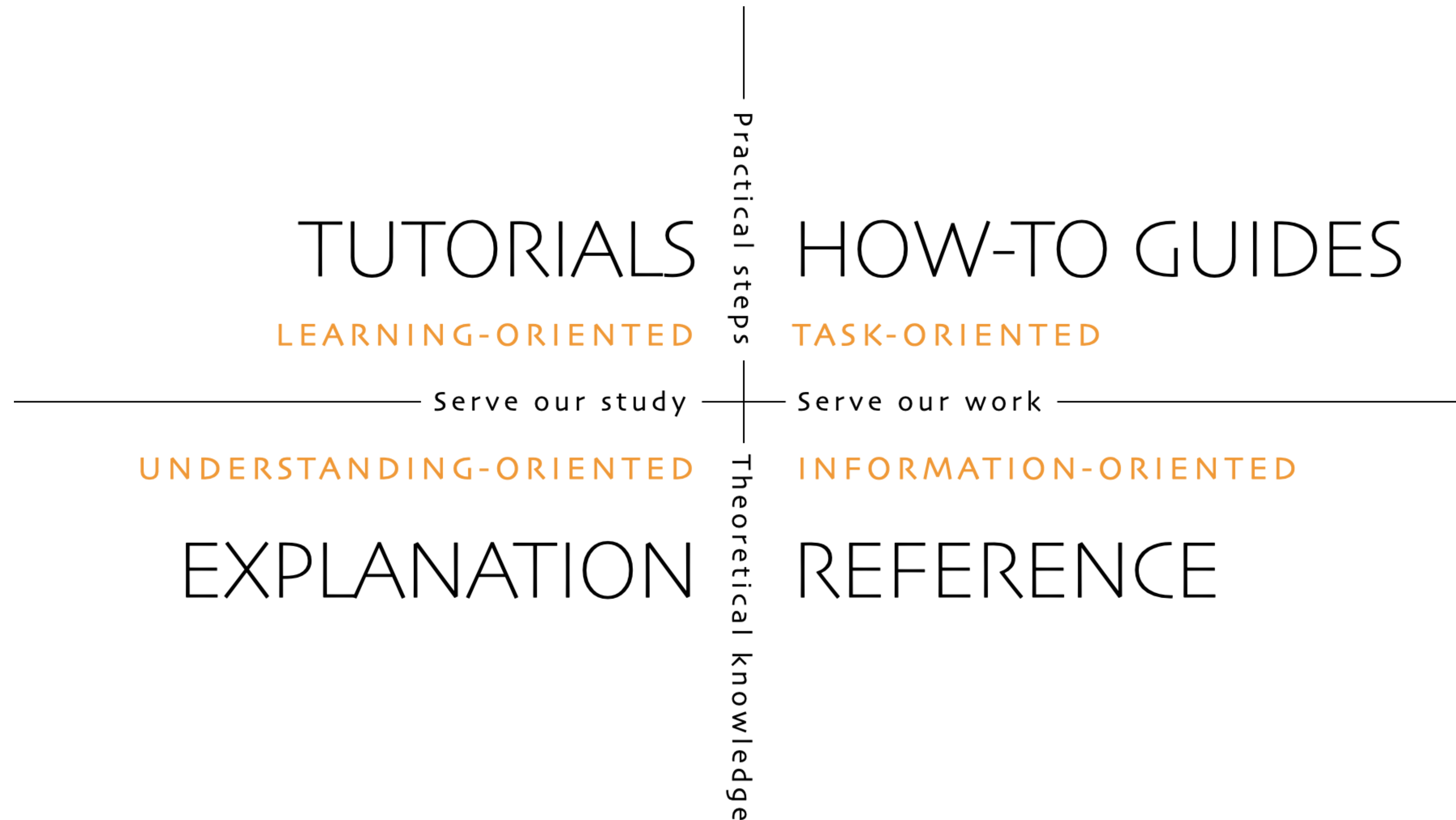


<https://www.writethedocs.org/videos/eu/2017/the-four-kinds-of-documentation-and-why-you-need-to-understand-what-they-are-daniele-procida/>

The figure is licensed [CC-BY-SA 4.0](https://creativecommons.org/licenses/by-sa/4.0/) by Daniele Procida et al. <https://diataxis.fr/>

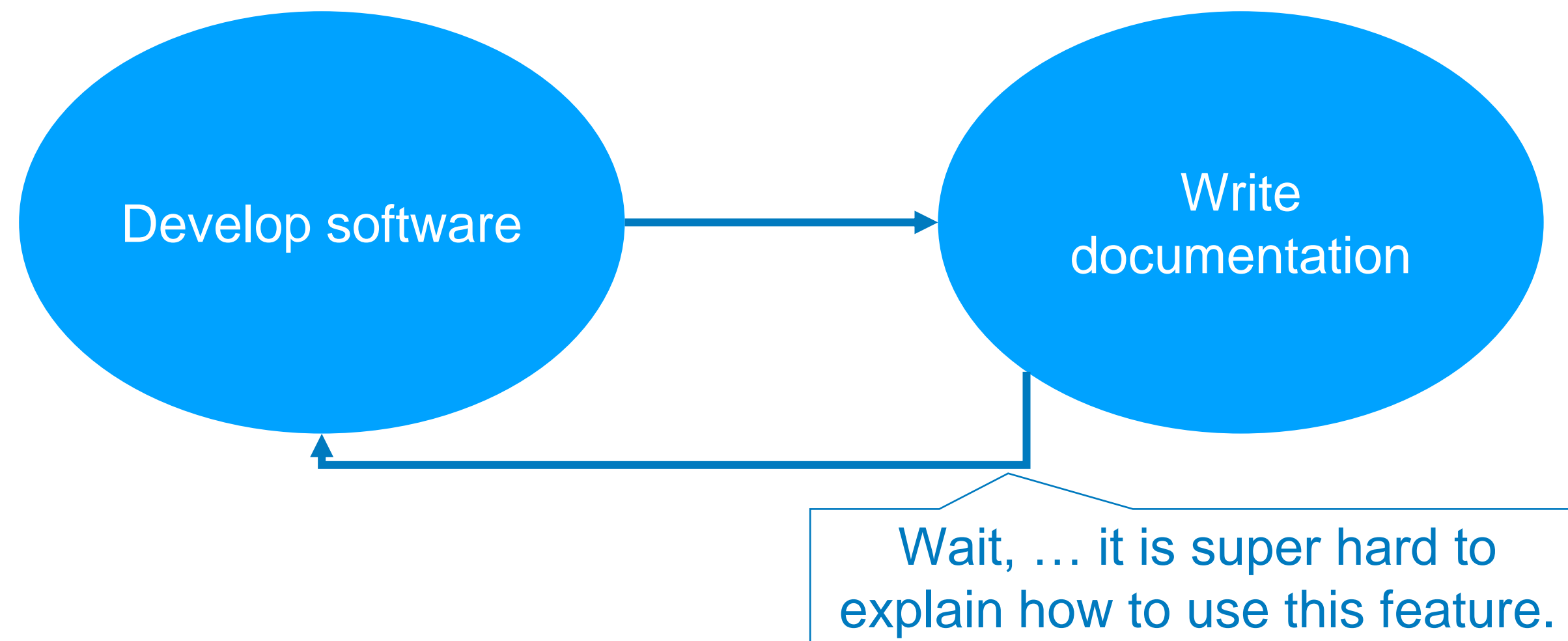
Documentation best-practices

The four modes of documentation



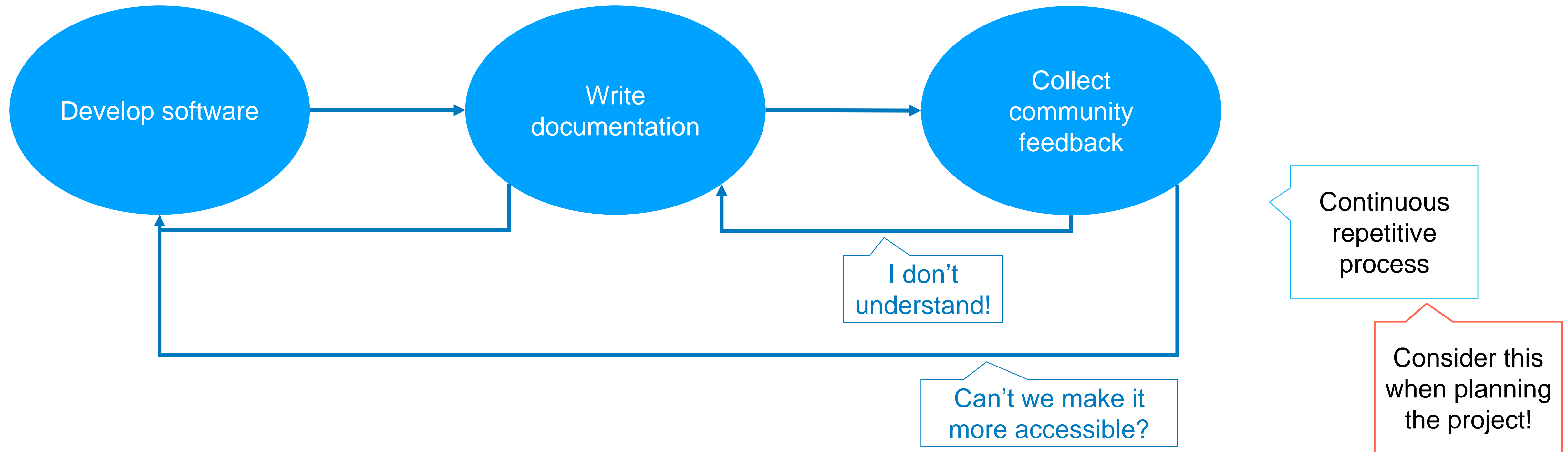
Documentation best-practices

Documentation is a process, not a final step



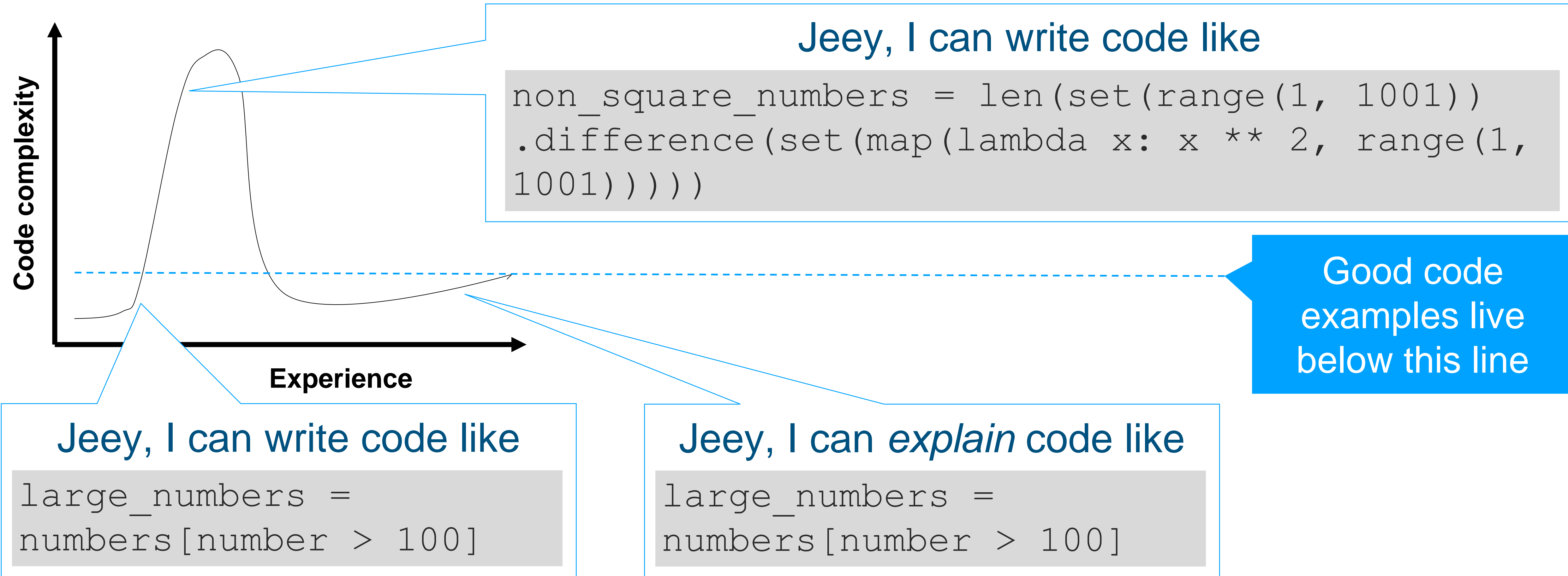
Documentation best-practices

Documentation is a process, not a final step



Documentation best-practices

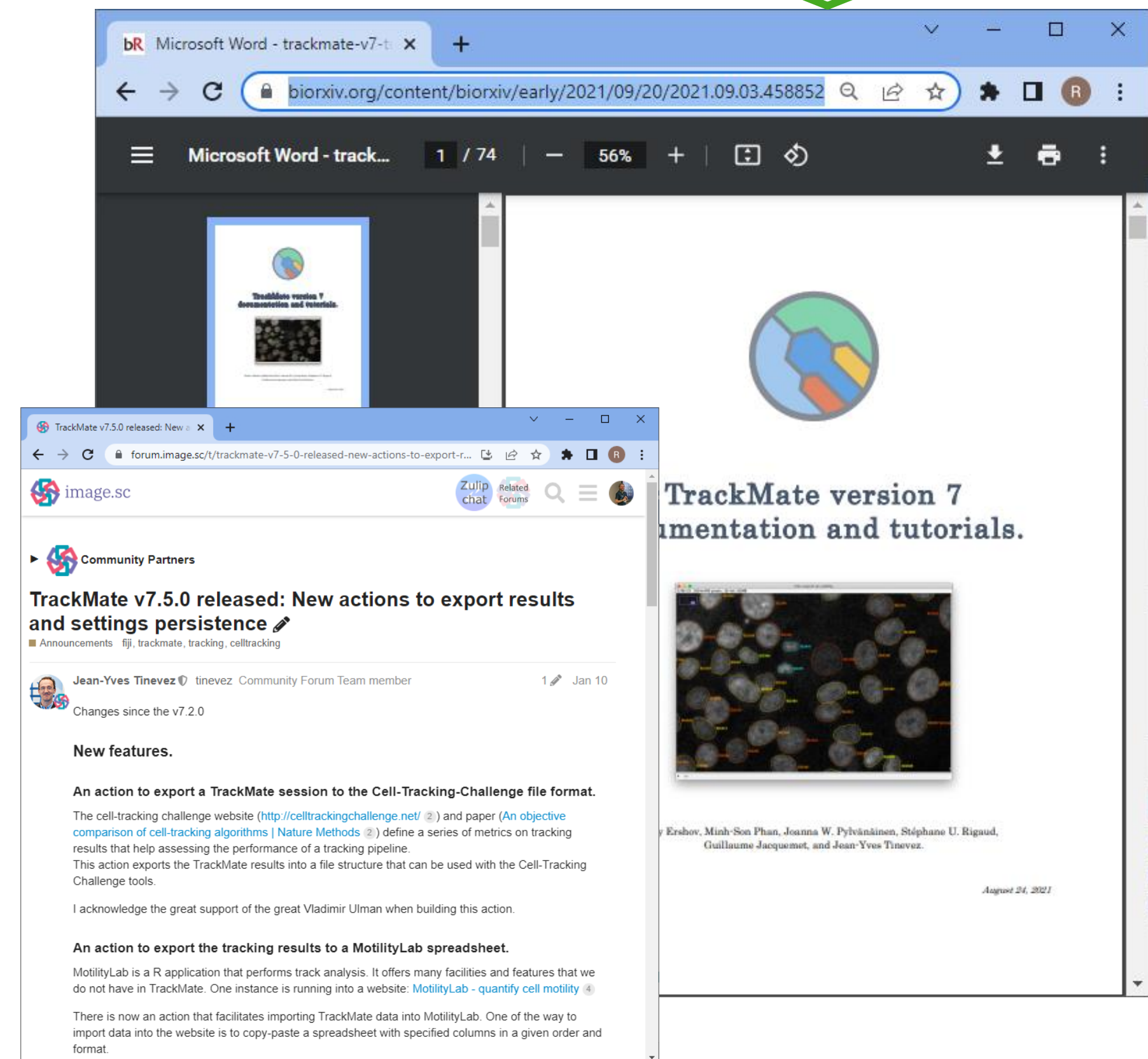
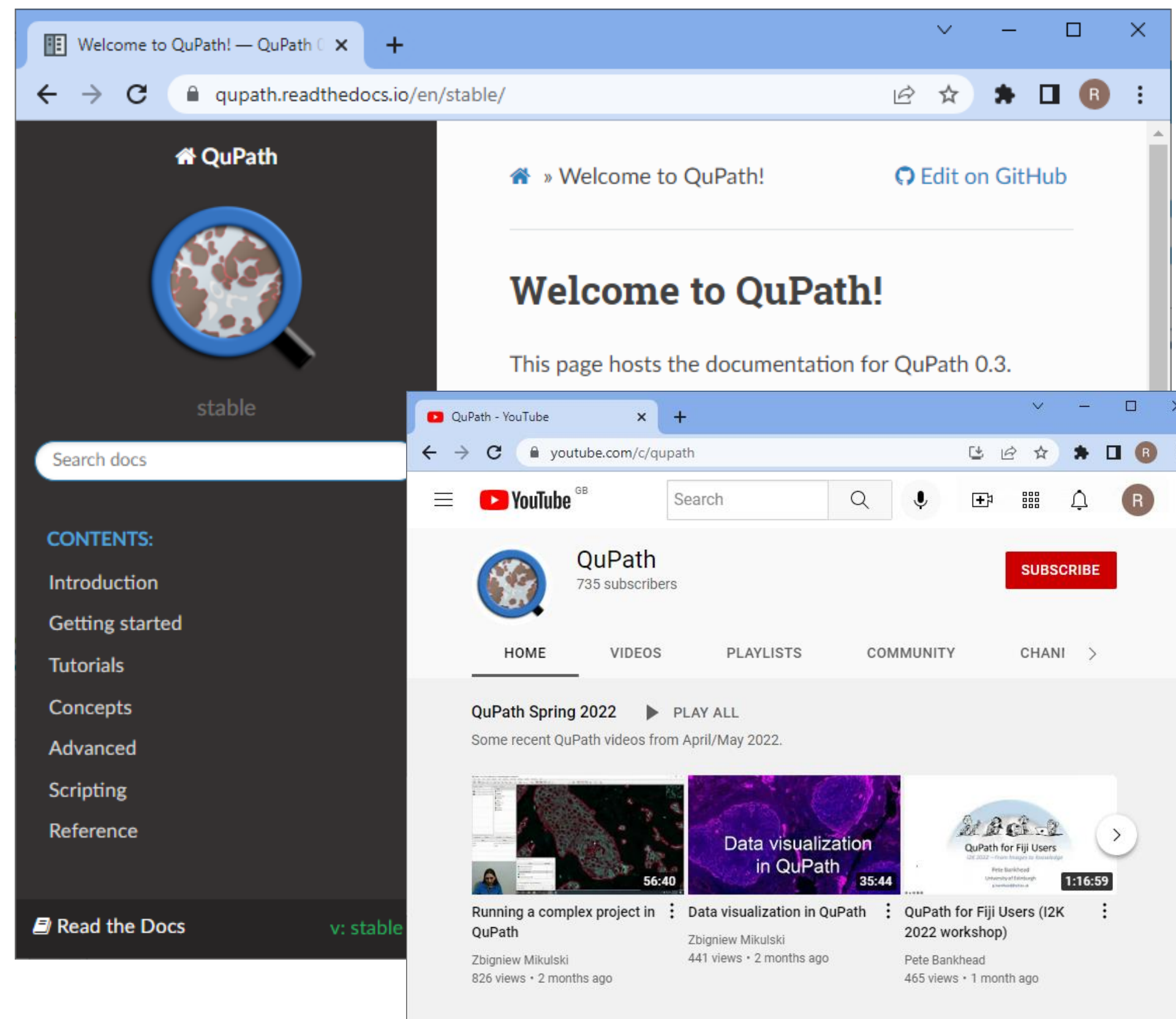
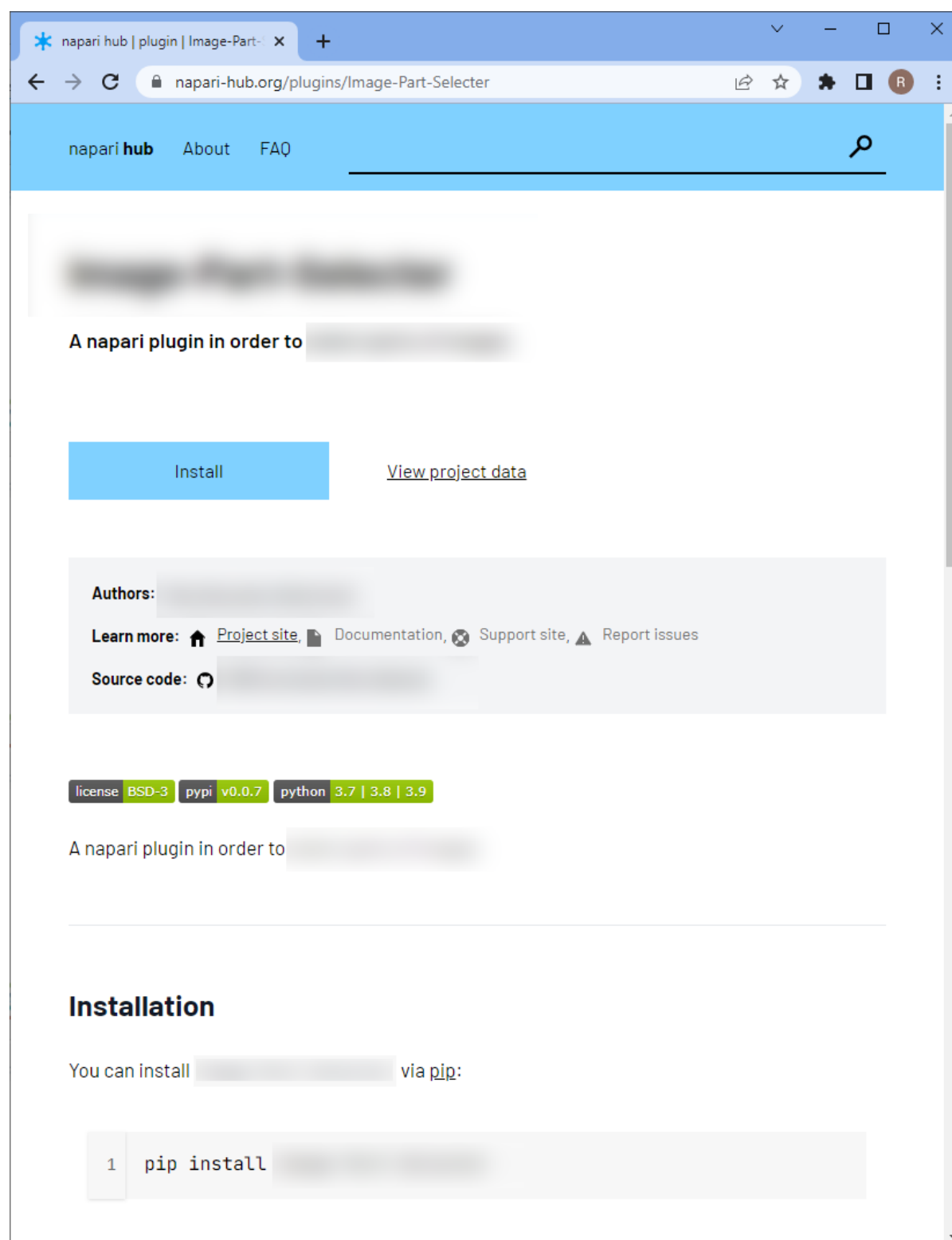
Interpretation of “High quality code” changes with experience



Documentation best-practices

Use your project's website to show others how to use your software.
Otherwise, they may not use it.

Communication
is key!



Shipping scientific software

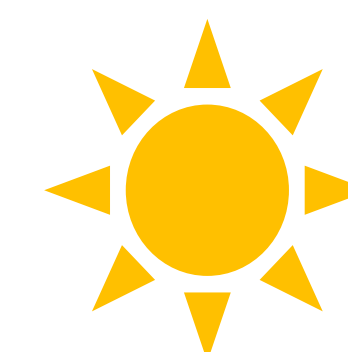
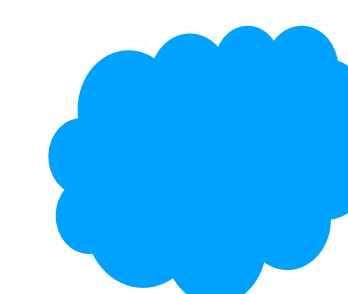
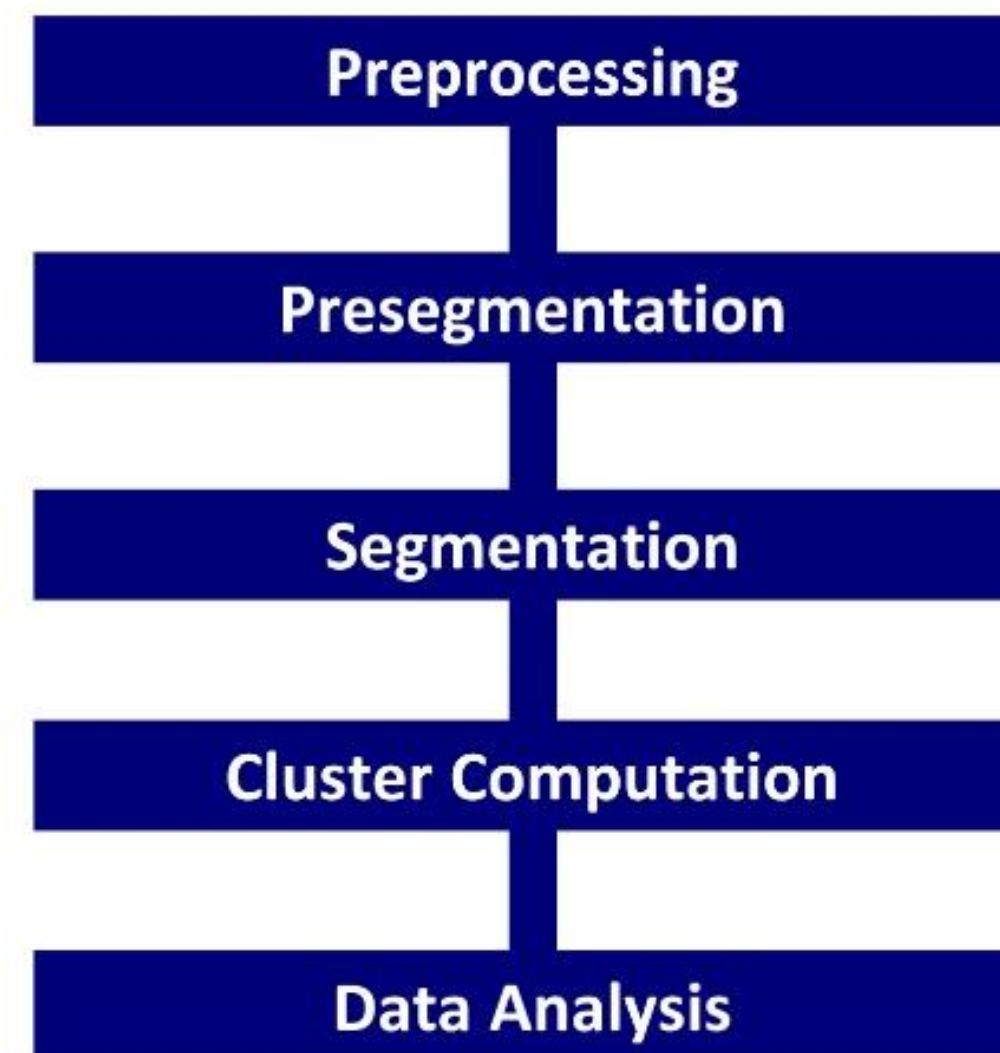
... so that others can reuse it

Deployment

Method / algorithm / software developers live from making things that are useful for others.

“... is available as
user-friendly
standalone software.”

“... is available as
user-friendly Python
library.”



Installation *can* be made
easy.

Ensuring interoperability
with other software comes
with high effort.

Combining standalone
software is often painful and
not *user-friendly*.

Think of workflow developers,
a.k.a. “users”.

In life-sciences, a Python
library may not be
perceived as *user-friendly*.

Installation and dependency
management is often tricky.

Interoperability comes
basically for free.

Deployment

Ship new methods as plugins for established platforms

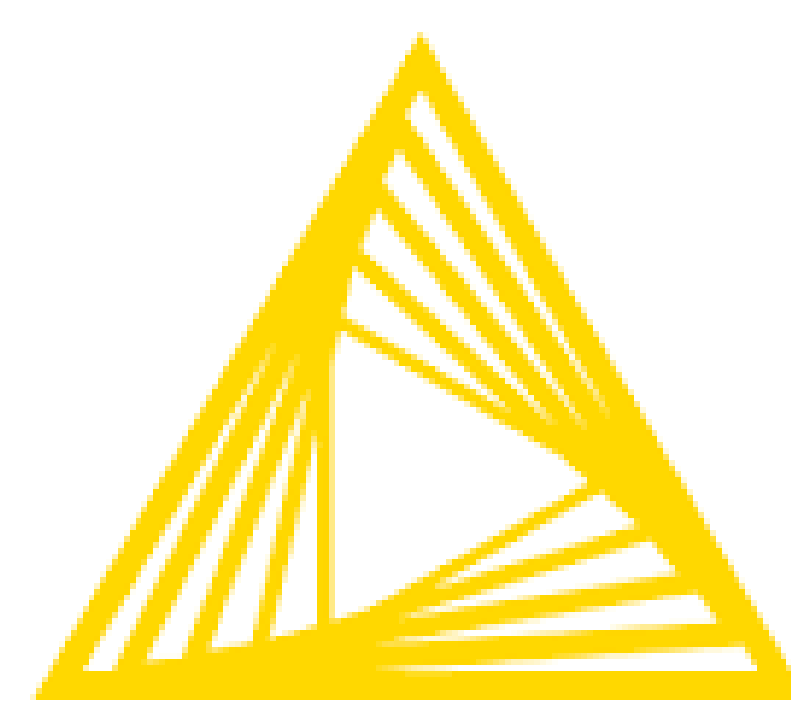
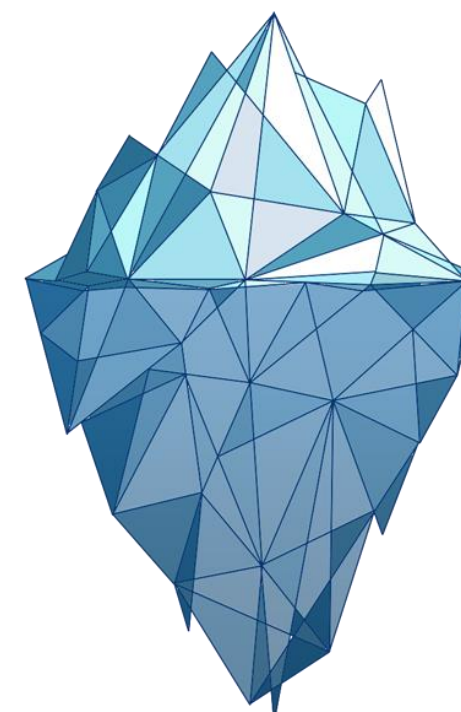
minimizing

and

maximizing

- effort to ensure interoperability with other tools and file-formats,
- entrance-level burden for new users and
- installation efforts.

- visibility,
- accessibility,
- community support and finally,
- adoption.



Logos may be subject to copyright of the respective communities

<https://imagej.nih.gov/ij/index.html> <https://fiji.sc/> <https://napari.org/>
<https://icy.bioimageanalysis.org/> <https://cellprofiler.org/> <https://www.knime.com/>

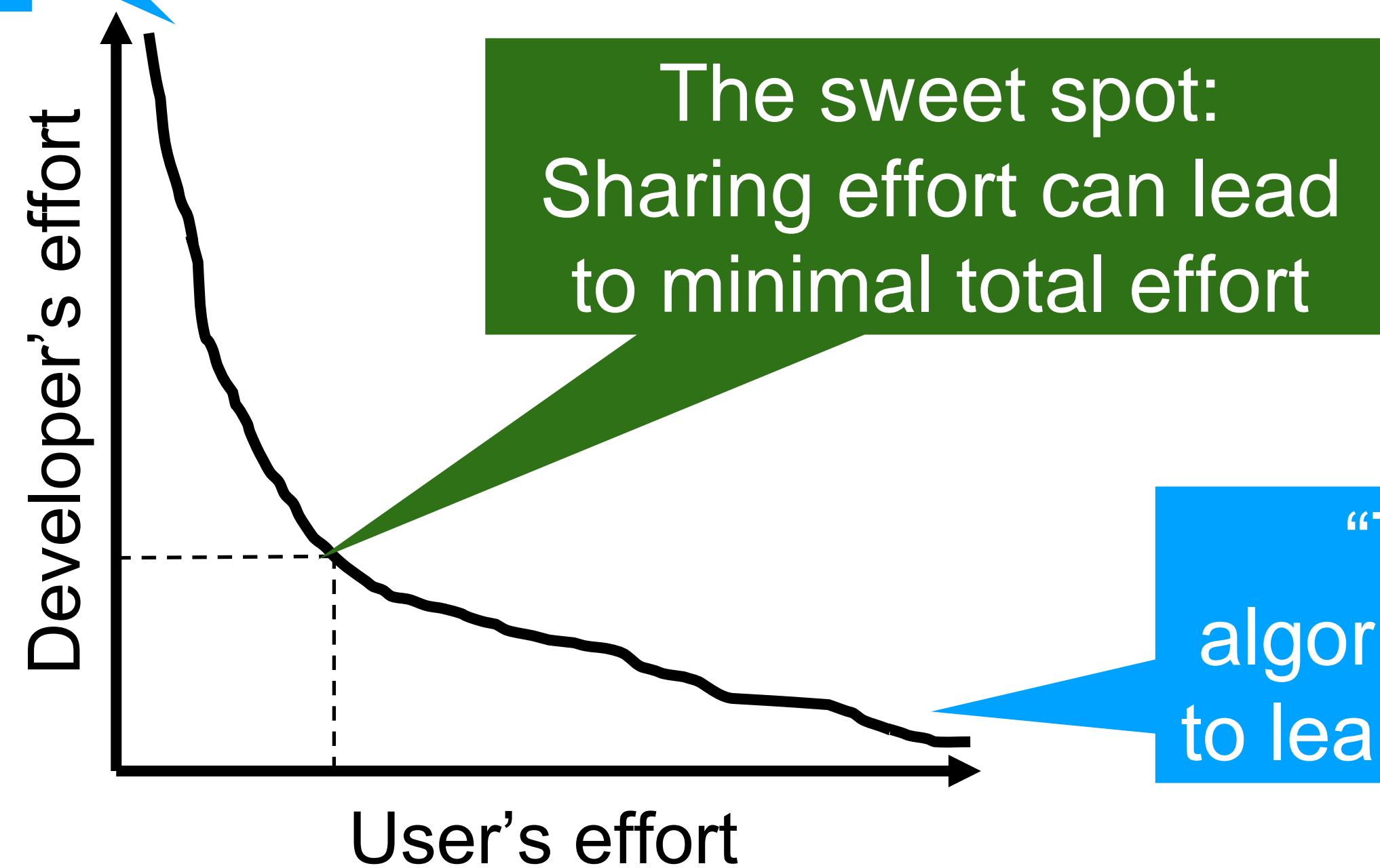
Interdisciplinary collaboration

... of life scientists and developers joining forces.

Why we need to work together

- ... to minimize effort

Making the user interface self-explaining and easily accessible



“To use our algorithm, you need to learn Python first.”

Why we need to work together

- ... to be successful in our projects (and grant applications)

Computer scientist: “I would like to develop software that allows to unravel the underlying principles of embryo development.”

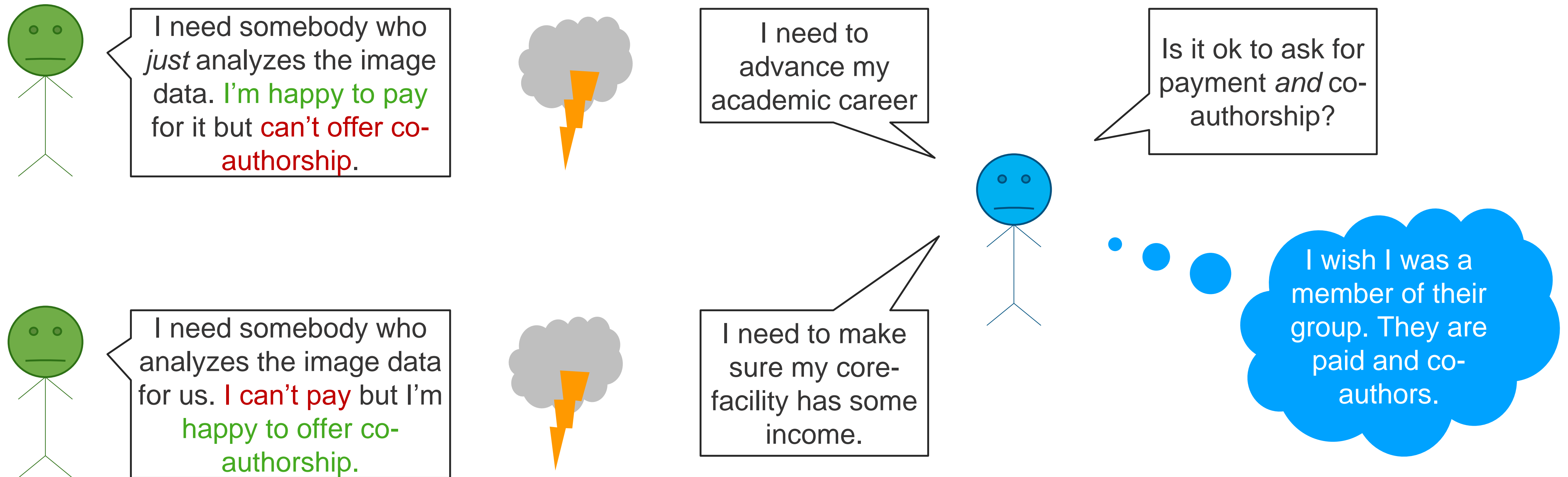
Funding agency: “But you’re lacking the developmental biology skills to make this happen.”

Biologist: “I would like to do quantitative image data science unravel the underlying principles of embryo development.”

Funding agency: “But you’re lacking the bioinformatics skills to make this happen.”

Matching expectations

... and *meeting in the middle* for successful collaborations



Matching expectations

Speak out your expectations and conditions early.

Communication
is key!

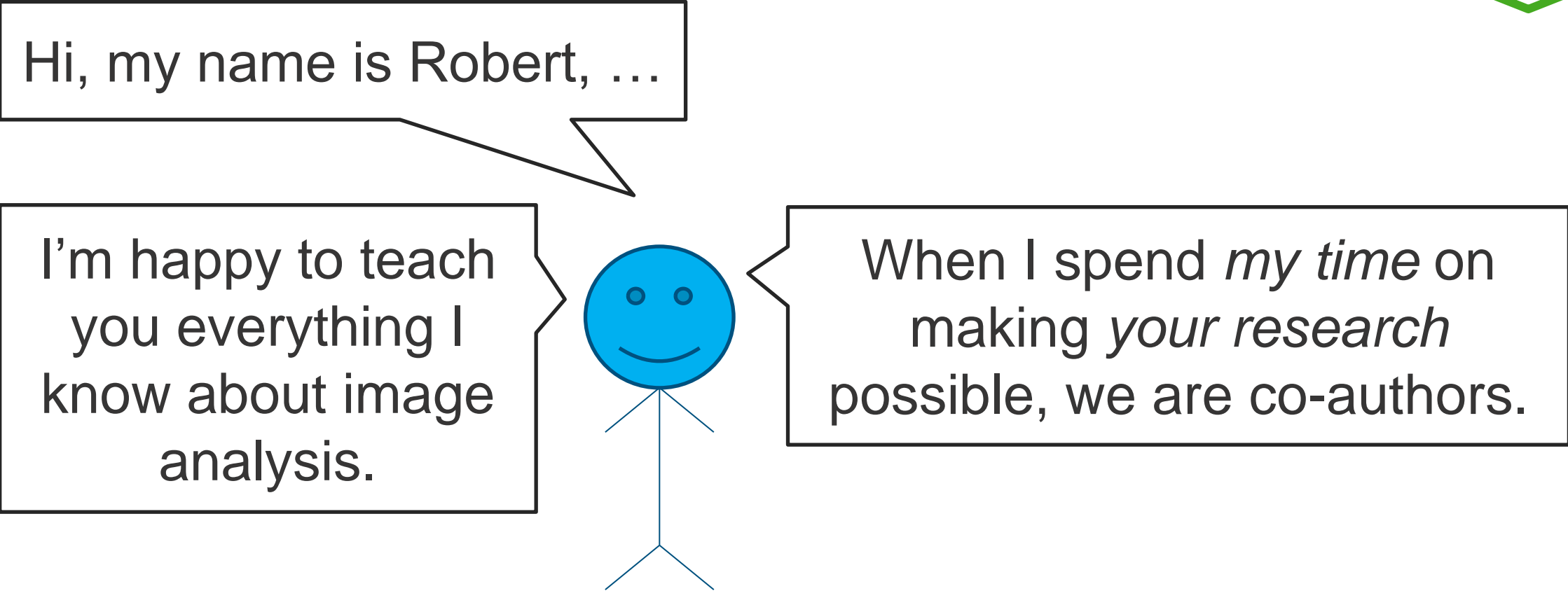


Image Analysis	Recommendation of analysis software and tools. Basic data analysis help and advice.	Simple acknowledgement
	Constructive data analysis and interpretation. Creation of complex custom image analysis tools.	Inclusion of specific facility member on author list

Imaging Facility Guidelines for Acknowledgement

- 1 All publications resulting from the use of instruments within the facility should acknowledge the facility as a whole, e.g. 'the authors gratefully acknowledge the [core facility name] for their support & assistance in this work' and the facility should be informed of the publication.
- 2 Specific grants that have funded the facility instruments used for the work to be published must be acknowledged if the data was acquired during the active period of that grant. Facility staff will advise users of such grant codes.
- 3 Assistance above the technical or routine level, with any facility staff providing scientific input and expertise in experimental set-up, acquisition or analysis, should be recognised through co-authorship on resulting publications. Please discuss acknowledgements with facility staff prior to manuscript submission.

Example scenarios with baseline recommendations:

Sample Preparation	Fast, routine sample preparation with standard protocol.	Simple acknowledgement
	Development of new sample preparation protocols. Optimisation of existing protocols for specific samples.	Inclusion of specific facility member on author list
Image Acquisition	Training of users to acquire images themselves. Simple acquisition of raw data.	Simple acknowledgement
	Operational image acquisition with input and decisions dependent on expertise. Design or re-design of experimental conditions.	Inclusion of specific facility member on author list
Image Analysis	Recommendation of analysis software and tools. Basic data analysis help and advice.	Simple acknowledgement
	Constructive data analysis and interpretation. Creation of complex custom image analysis tools.	Inclusion of specific facility member on author list

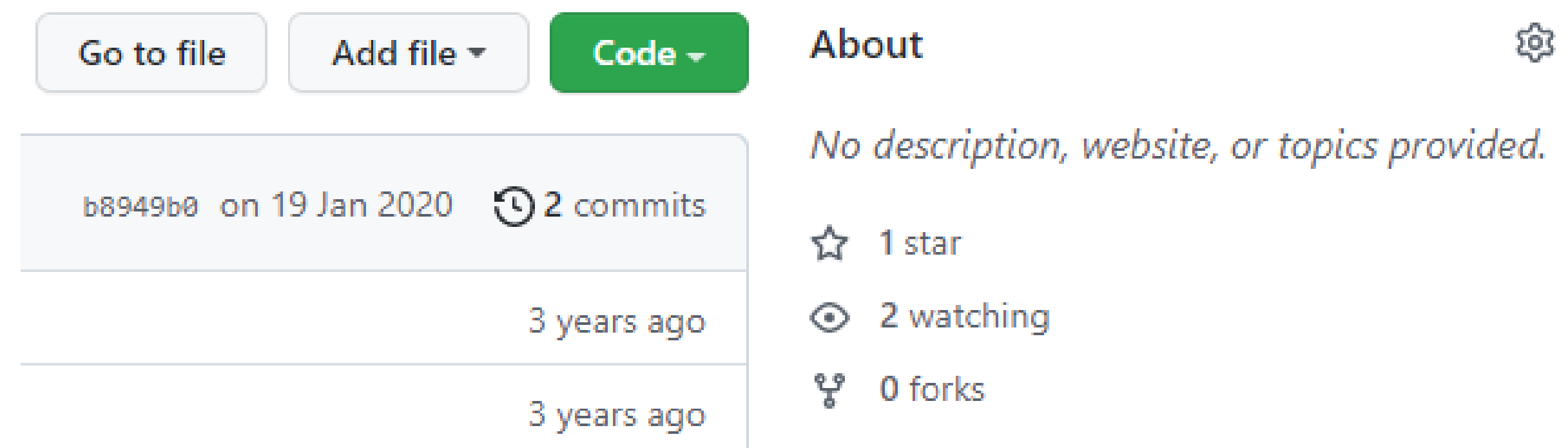
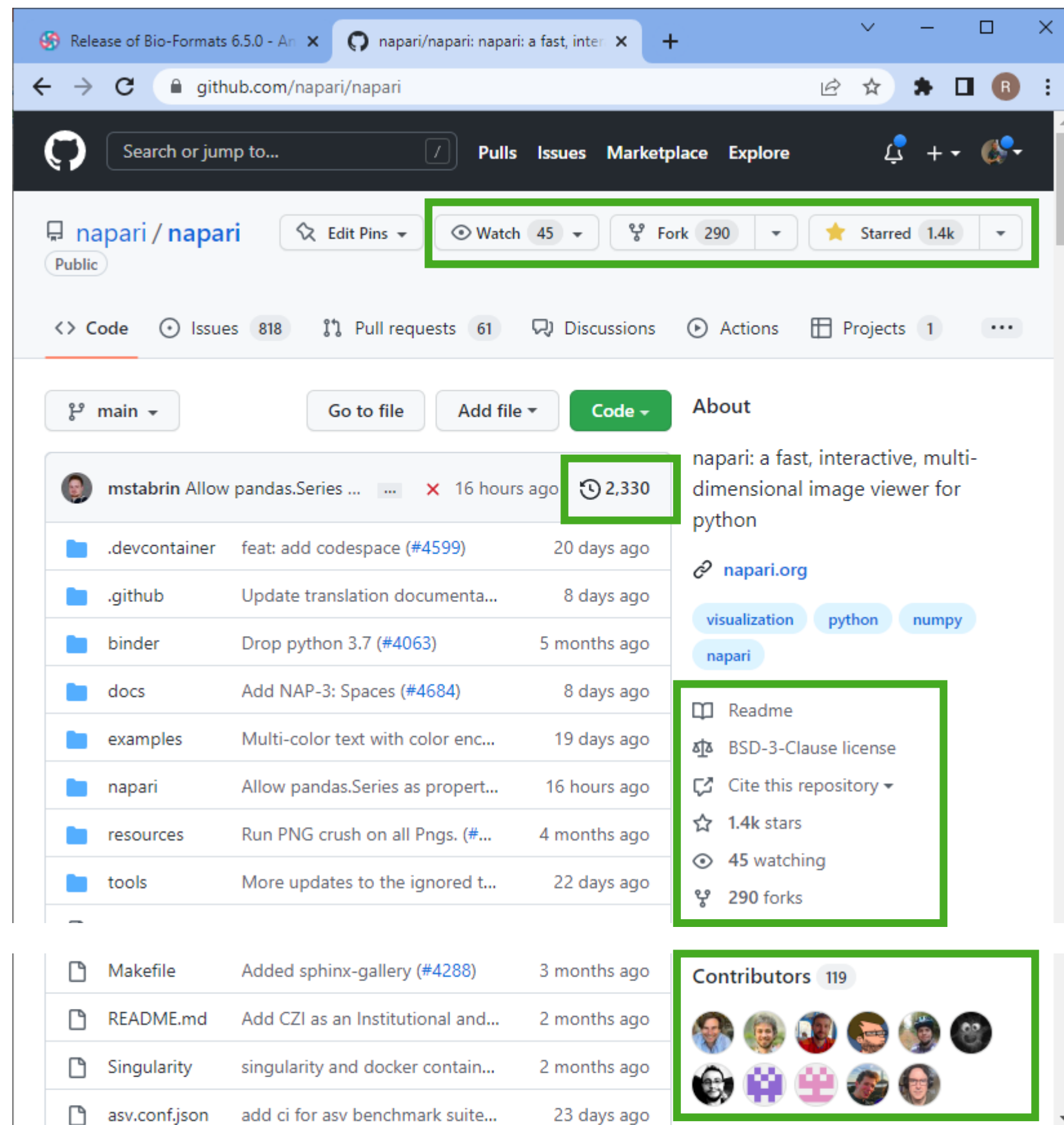
Based on the publication policy compiled by Natasha Stephen, Plymouth Electron Microscopy Centre, after discussions with the RMS EM-UK community



Software quality indicators

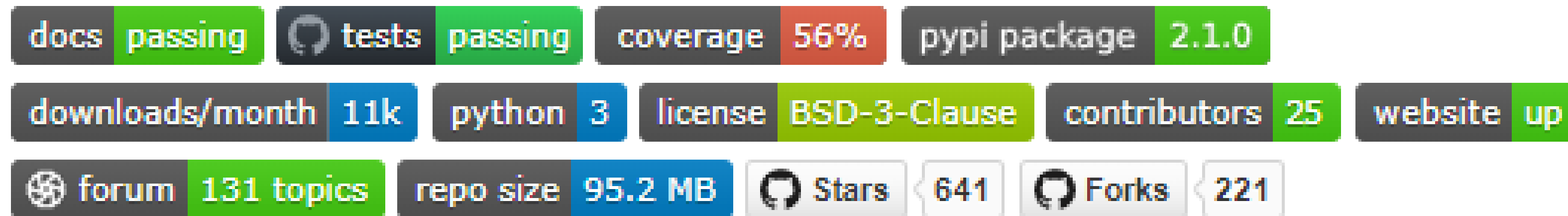
Visit the project's github or gitlab page and review indicators.

- **Stars:** People like software, similarly to tweets on Twitter
- **Watching:** People receive updates for new releases
- **Forks:** People made a copy of the code, e.g. to contribute to the project
- **Contributors:** People who contributed to the code
- **Commits:** Changes to the code



Software quality indicators

Visit the project's github or gitlab page and review indicators.



Note, github
badges not
deserved.



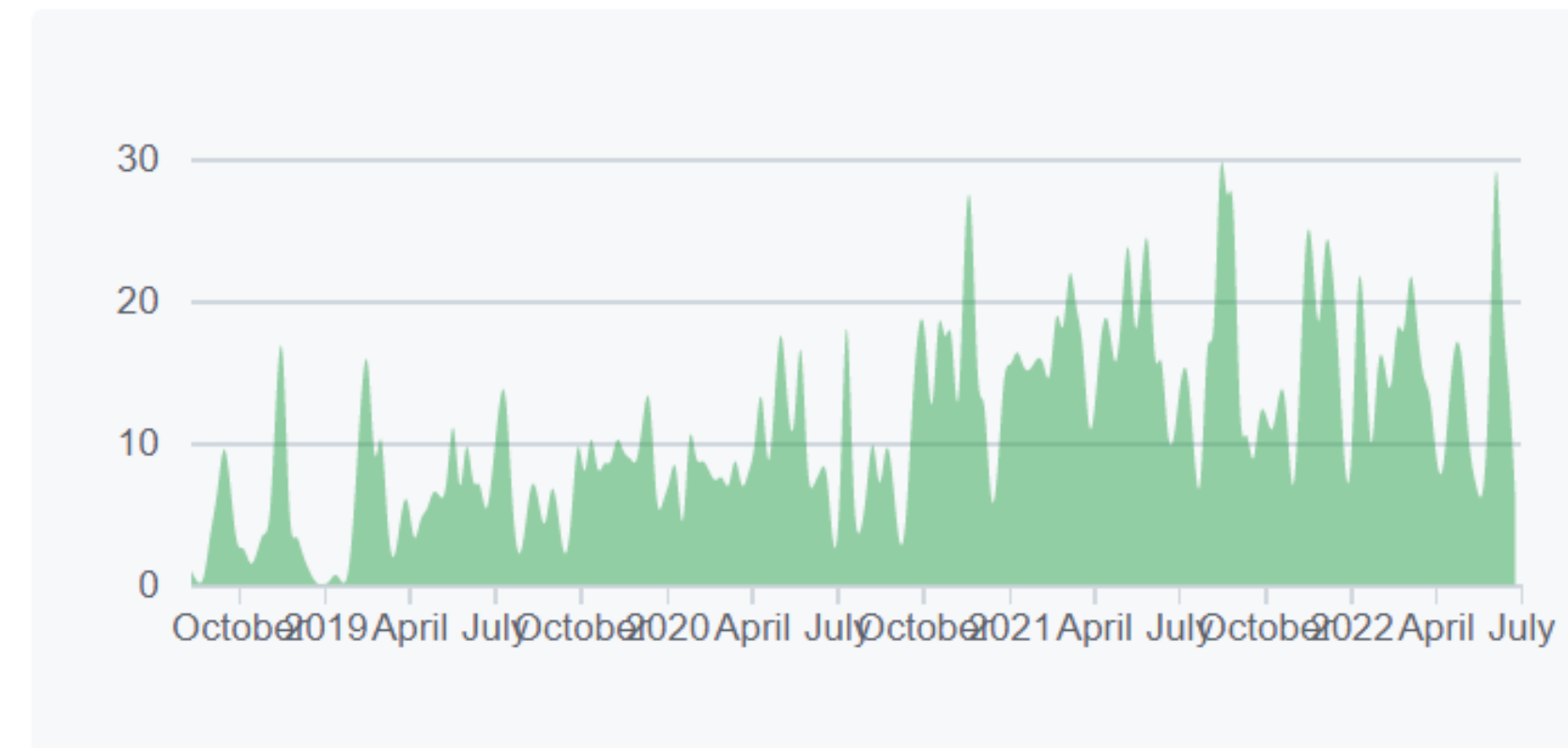
Software quality indicators

Visit the project's github or gitlab page and review indicators.

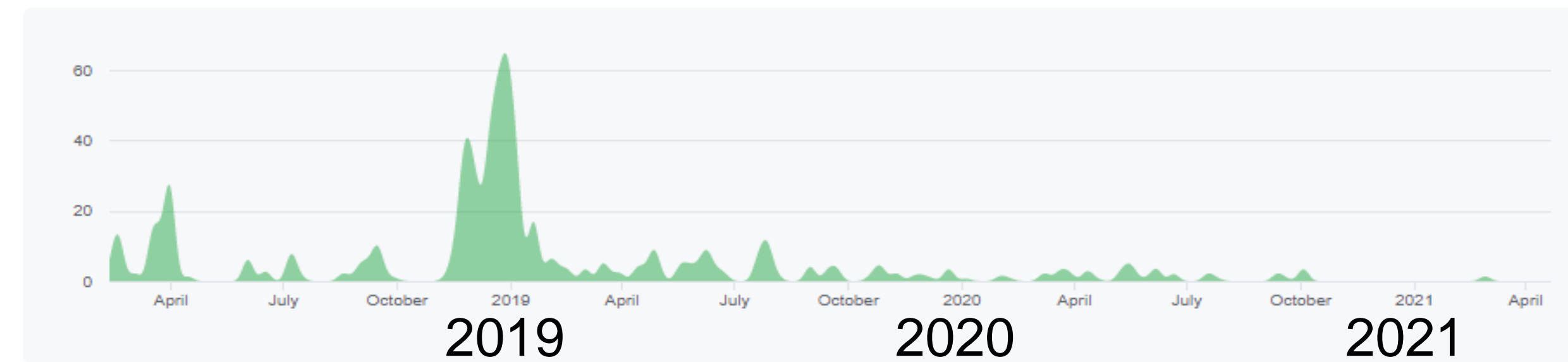
Aug 12, 2018 – Jul 1, 2022

Contributions: Commits ▼

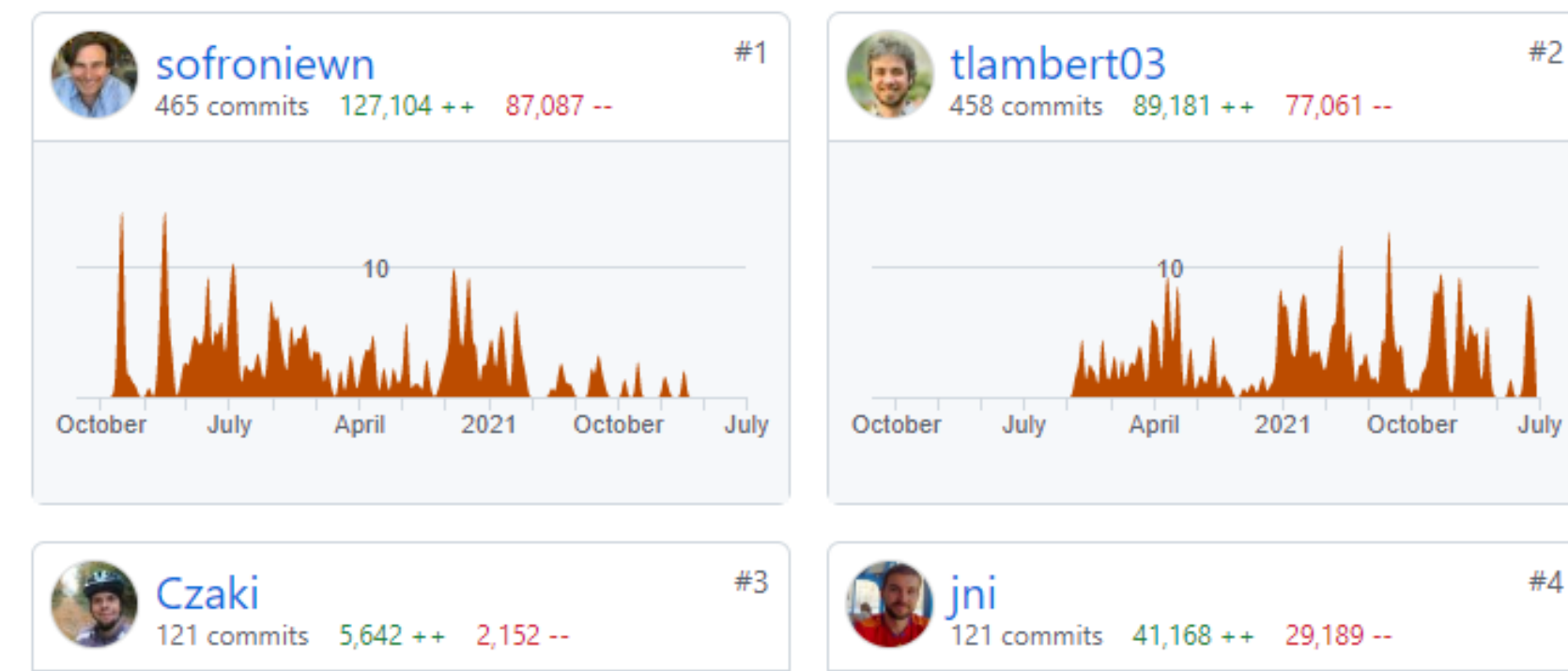
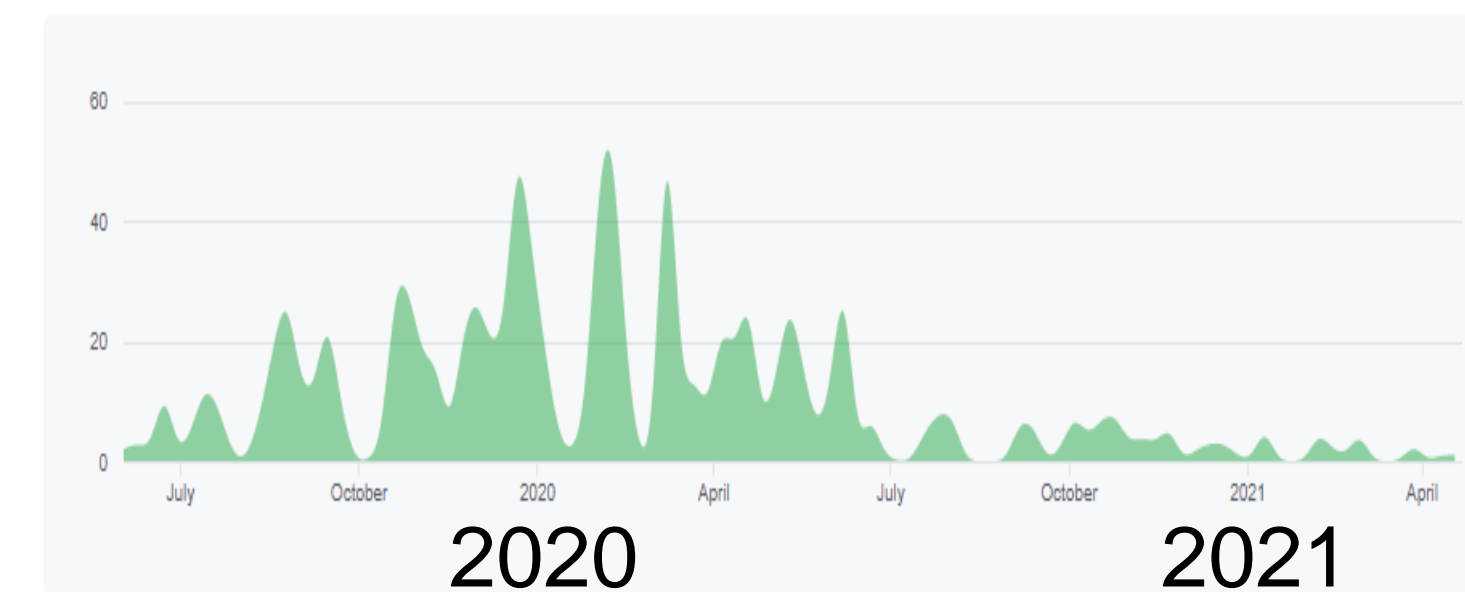
Contributions to main, excluding merge commits and bot accounts



Be a bit careful when development appears to have halted



And check related projects



@haesleinhuepf
@PoLDresden

<https://github.com/napari/napari>
<https://github.com/napari/napari/graphs/contributors>
<https://github.com/cli/cli/graphs/contributors>
<https://github.com/cli/cli2/graphs/contributors>



TECHNISCHE
UNIVERSITÄT
DRESDEN

DRESDEN
concept



Crowd sourcing

... ideas, feedback, improvements

Market research

Engage with the community

- Use twitter to reach out
- Learn about who needs what

Communication
is key!

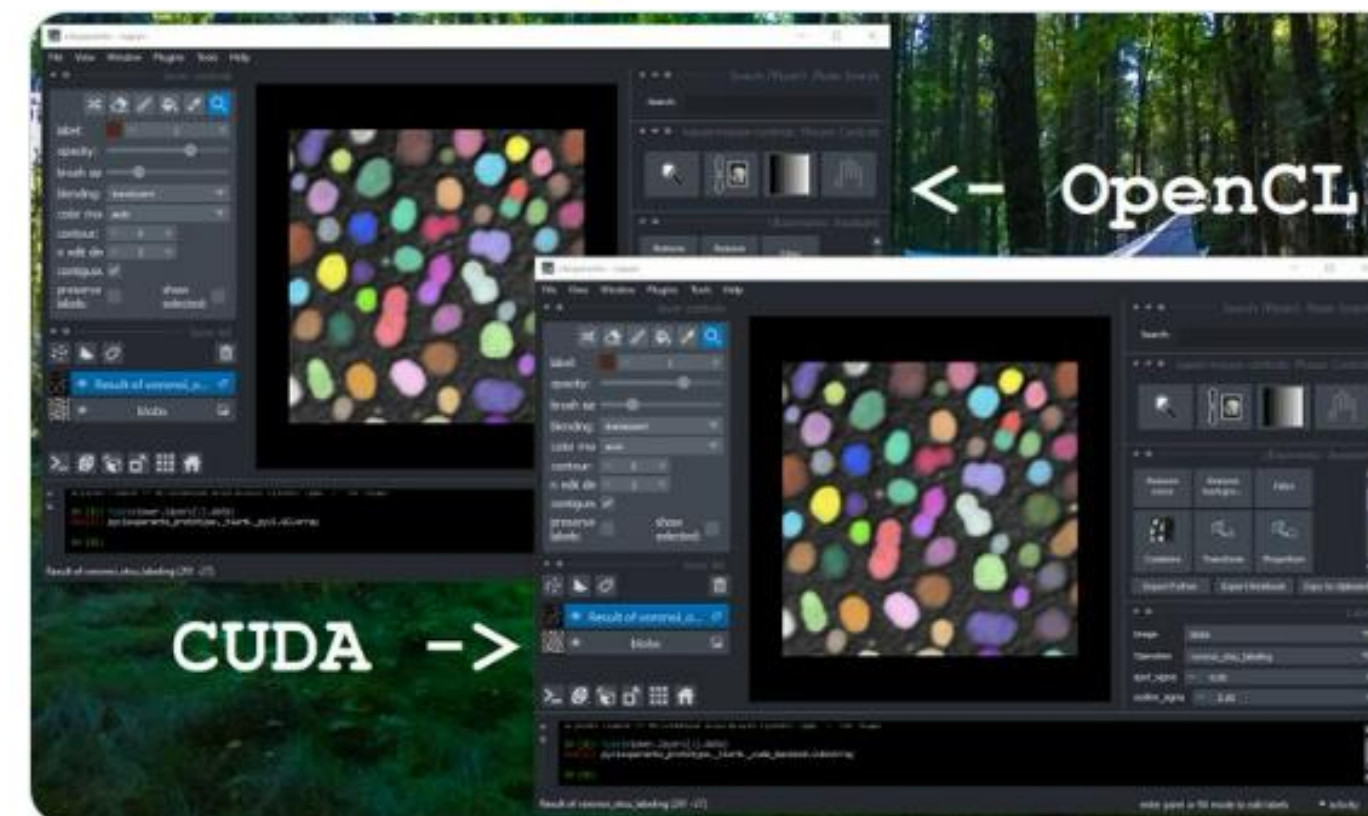


Robert Haase
@haesleinhuepf

... and we will incorporate [#CUDA](#)/[@CuPy_Team](#) with [#OpenCL](#)/[#clesperanto](#) to find new friends who use GPUs differently and learn from them 🧐💻🚀

You wonder what will be different between [#clesperanto](#)-OpenCL and [#clesperanto](#)-CUDA based image processing? 🤔

I hope not a single bit 😎



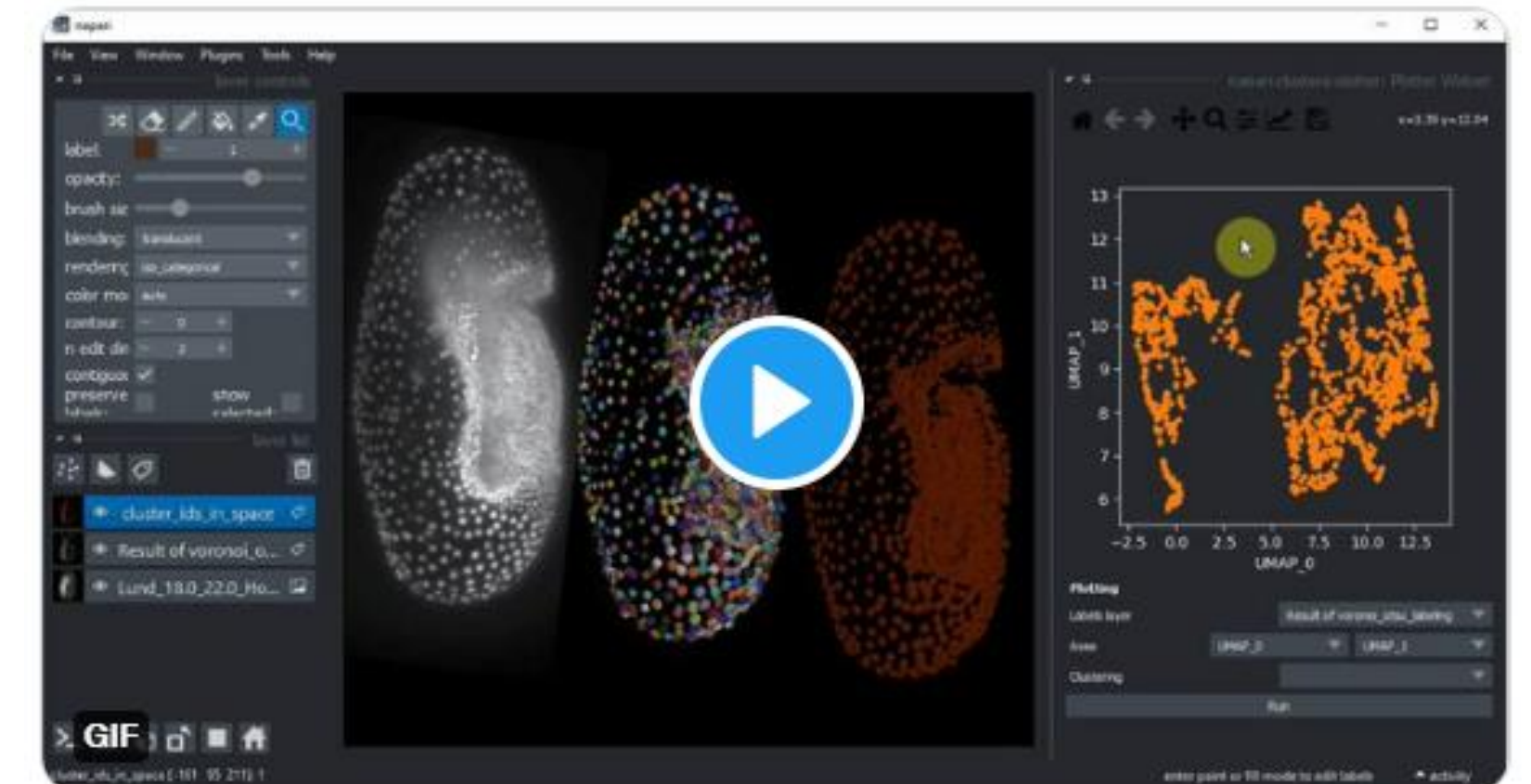
5:55 PM · Nov 15, 2021 · Twitter Web App

2 Retweets 32 Likes



Laura Žigutytė
@zigutyte

My first [@napari_imaging](#) plugin is out now! It can measure region properties, plot them, perform k-means or manual clustering. Huge thanks to [#biapol](#) group members for amazing teamwork and all the help [@RyanSavill4](#), [@zoccolermarcelo](#) and [@haesleinhuepf](#) 🌟
[napari-hub.org/plugins/napari...](#)



2:56 PM · Nov 15, 2021 · Twitter Web App

74 Retweets 12 Quote Tweets 403 Likes

Community involvement

forum.image.sc/t/scroll-through-multiple-hyperstacks-interlocked/...

image.sc

Community Partners

Scroll through multiple hyperstacks interlocked

Image Analysis imagej, fiji

Robert Haase haesleinhuepf clij & clesperanto maintainer 3d

Hi all,

we usually have two hyperstacks open that belong to two different channels of the recording.
-it would be great if one could scroll the two hyperstacks together (having them interlocked)
Do anybody have an idea how this might work?

Asking for a friend.

Thanks!

Best,
Robert

✓ Solved by [lmurphy](#) in post #2

Hi Robert, Perhaps I don't understand but for similar stuff I use Analyze > Tools > Synchronise Windows (important: not Sync Windows!). [forForum]

forum.image.sc/t/scroll-through-multiple-hyperstacks-interlocke...

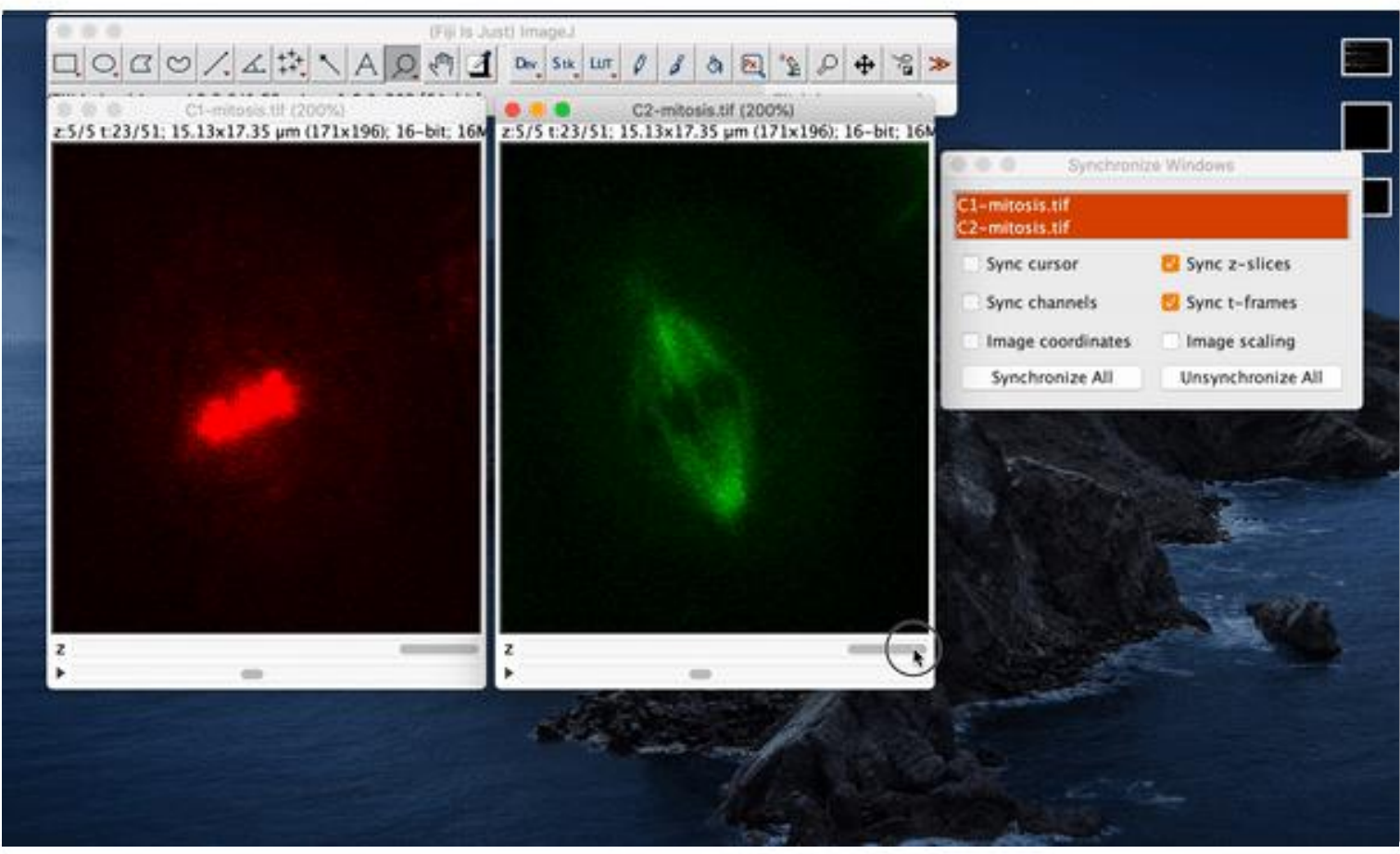
Scroll through multiple hyperstacks int...

Image Analysis imagej, fiji

Laura Murphy Imurphy Leader 3d

Hi Robert,

Perhaps I don't understand but for similar stuff I use Analyze > Tools > Synchronise Windows (important: not Sync Windows!).



✓ Solution 6 ❤️ 🔗 ... ↩ Reply

Robert Haase haesleinhuepf clij & clesperanto maintainer 2/3

Reply after 18 minutes!

Community involvement

The screenshot displays the image.sc forum interface. The main thread title is "interlocked hyperstacks". A sidebar on the right, titled "Your topic is similar to...", lists related topics. A yellow arrow points from the thread title to the sidebar, and a green arrow points from the sidebar back to the thread title. A green callout bubble on the right says "Reply after 18 minutes!".

Thread Title: interlocked hyperstacks

Author: Robert Haase

Content: Hi all, we usually have two hyperstacks open that belong to two different channels of the recording. -it would be great if one could scroll the two hyperstacks together (having them interlocked) Do... Asking for a friend. Thanks! Best, Robert

Similar Topics:

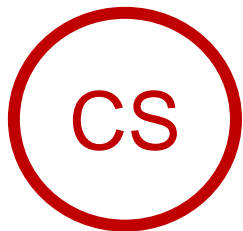
- ☒ Scroll through multiple hyperstacks interlocked
- ☒ Bug in "Re-order Hyperstack..."
- ☒ Make substack from hyperstack
- ☒ Intensity Projection on a Multichannel hyperstack

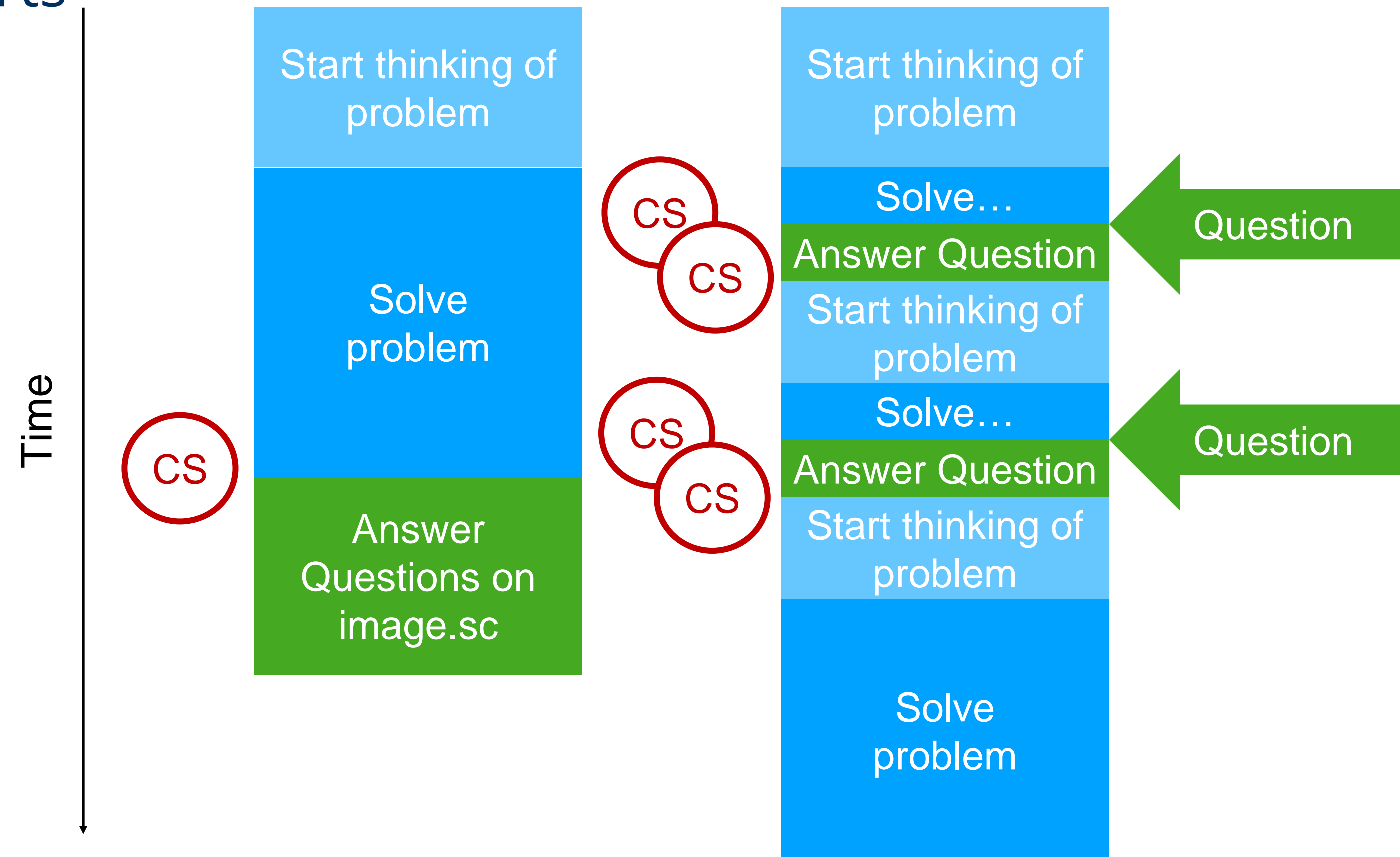
Why public support channels

User's perspective

- Receive feedback from multiple experts
- Also if one expert is on vacation

Developer's perspective

- Reduce repetitive basic questions
- Distribute workload
- Distribute knowledge
- Asynchronous Q&A
- Minimize context-switches 



Related publications

F1000Research






BROWSE GATEWAYS & COLLECTIONS HOW TO PUBLISH ▾ ABOUT ▾ BLOG

Home » Browse » Developing open-source software for bioimage analysis: opportunities...

OPINION ARTICLE

Check for updates

Developing open-source software for bioimage analysis: opportunities and challenges [version 1; peer review: 2 approved]

✉ Florian Levett ^{1,2}, Anne E. Carpenter ³, Kevin W. Eliceiri ⁴, Anna Kreshuk⁵, Peter Bankhead ⁶, Robert Haase ⁷

Author details

ALL METRICS


1683 VIEWS

223 DOWNLOADS

This article is included in the NEUBIAS - the Bioinformatics Gateway

This article is included in the Bioinformatics Gateway

This article is included in the Research on Res

 Cornell University

arXiv > q-bio > arXiv:2110.13951

Quantitative Biology > Other Quantitative Biology

[Submitted on 26 Oct 2021]

Open microscopy in the life sciences: Quo Vadis?

Johannes Hohlbein, Benedict Diederich, Barbora Marsikova, Emmanuel G. Reynaud, Seamus Holden, Wiebke Jahr, Robert Haase, Kirti Prakash

PERSPECTIVE article

Front. Bioinform., 14 April 2022 | <https://doi.org/10.3389/fbinf.2022.889755>


Meeting in the Middle: Towards Successful Multidisciplinary Bioimage Analysis Collaboration

 Anjalie Schlaeppi^{1,2*},  Wilson Adams^{3,4†},  Robert Haase^{5†},  Jan Huiskens^{1,6†},  Ryan B. MacDonald^{7†},  Kevin W. Eliceiri^{1,8†} and  Elisabeth C. Kugler^{7*†}

¹Morgridge Institute for Research, Madison, WI, United States
²BioImaging and Optics Platform, École Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland
³Department of Biomedical Engineering, Vanderbilt University, Nashville, TN, United States
⁴Department of Pharmacology, Vanderbilt University, Nashville, TN, United States
⁵DFG Cluster of Excellence "Physics of Life", Germany and Center for Systems Biology Dresden, TU Dresden, Dresden, Germany
⁶Department of Biology and Psychology, Georg-August-University Göttingen, Göttingen, Germany
⁷Faculty of Brain Sciences, Institute of Ophthalmology, University College London, London, United Kingdom
⁸Center for Quantitative Cell Imaging, University of Wisconsin-Madison, Madison, WI, United States

TOTAL VIEWS **1,085**

Check for updates

 Cornell University

arXiv > q-bio > arXiv:2204.07547

Quantitative Biology > Quantitative Methods

[Submitted on 15 Apr 2022]

A Hitchhiker's Guide through the Bio-image Analysis Software Universe

Robert Haase, Elnaz Fazeli, David Legland, Michael Doube, Siân Culley, Ilya Belevich, Eija Jokitalo, Martin Schorb, Anna Klemm, Christian Tischer

Modern research in the life sciences is unthinkable without computational methods for extracting, quantifying and visualizing information derived from biological microscopy imaging data. In the past decade, we observed a dramatic increase in available software packages for these purposes. As it is increasingly difficult to keep track of the number of available image analysis platforms, tool collections, components and emerging technologies, we provide a conservative overview of software we use in daily routine and give insights into emerging new tools. We give guidance on which aspects to consider when choosing the right platform, including aspects such as image data type, skills of the team, infrastructure and community at the institute and availability of time and budget.