

CCP-volumeEM Show & Tell

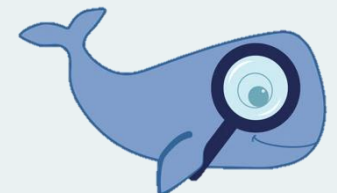
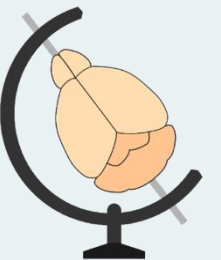
Good practices for image  
analysis software

# Quick about me

- Research software engineer
- UCL's Advanced Research Computing Centre (ARC)
- All kinds of projects including many biological / medical + imaging-related
- + contributing to open-source projects

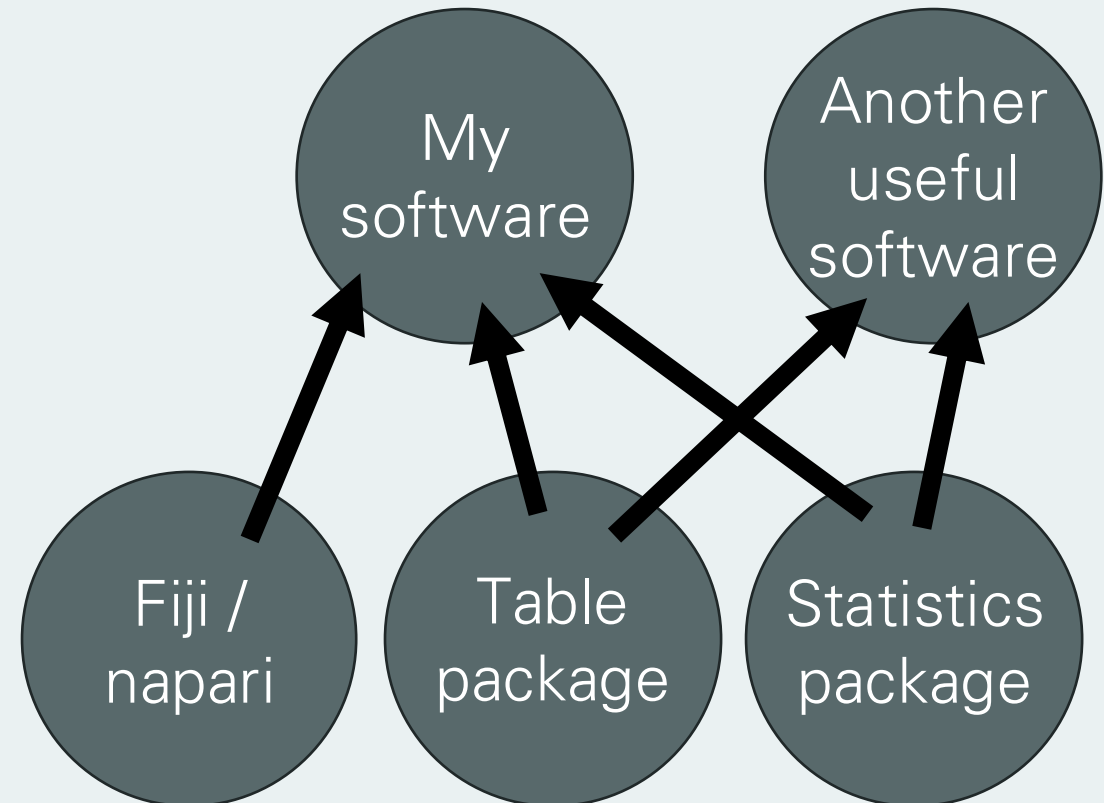


Crosshair  
(no logo ? )



# Software maintenance – why?

- Software needs continued work to remain up-to-date
- Bug fixes
- Adding new features
- Updating documentation
- Keeping up with dependencies:
  - Other packages
  - The framework you are building in:  
e.g. Fiji / napari



# Software maintenance – how?

- Often limited resources:
  - Small teams
  - Limited funding
- Make maintenance as efficient as possible
- Steps beyond making the software work
- Focus on what has the most impact for your project!



# Version control - Git and GitHub

- Git = a version control system
- GitHub = a website that integrates with git + lets you collaborate / share your code with others
- There are more options: GitLab, Bitbucket...
- Let's look at napari's GitHub...



# Automated dependency updates

- How do I know when a new version is out?
- Detect and automatically open PRs
- Detect security vulnerabilities
- Works for python, java...



Dependabot



Renovate  
(Mend)

# Guidelines for dependency support

- Spec 0 for scientific python: <https://scientific-python.org/specs/spec-0000/>

# Testing

- Automated testing of your software
- Python (napari) = pytest

<https://docs.pytest.org/>

- Java (Fiji): JUnit

<https://junit.org/>





# Running remotely

- Usually we want to run when a new PR is opened
- 'Continuous integration': integrating new code changes frequently, usually with automated tests / builds etc.
- GitHub actions (or GitLab CI/CD...)

<https://docs.github.com/en/actions>



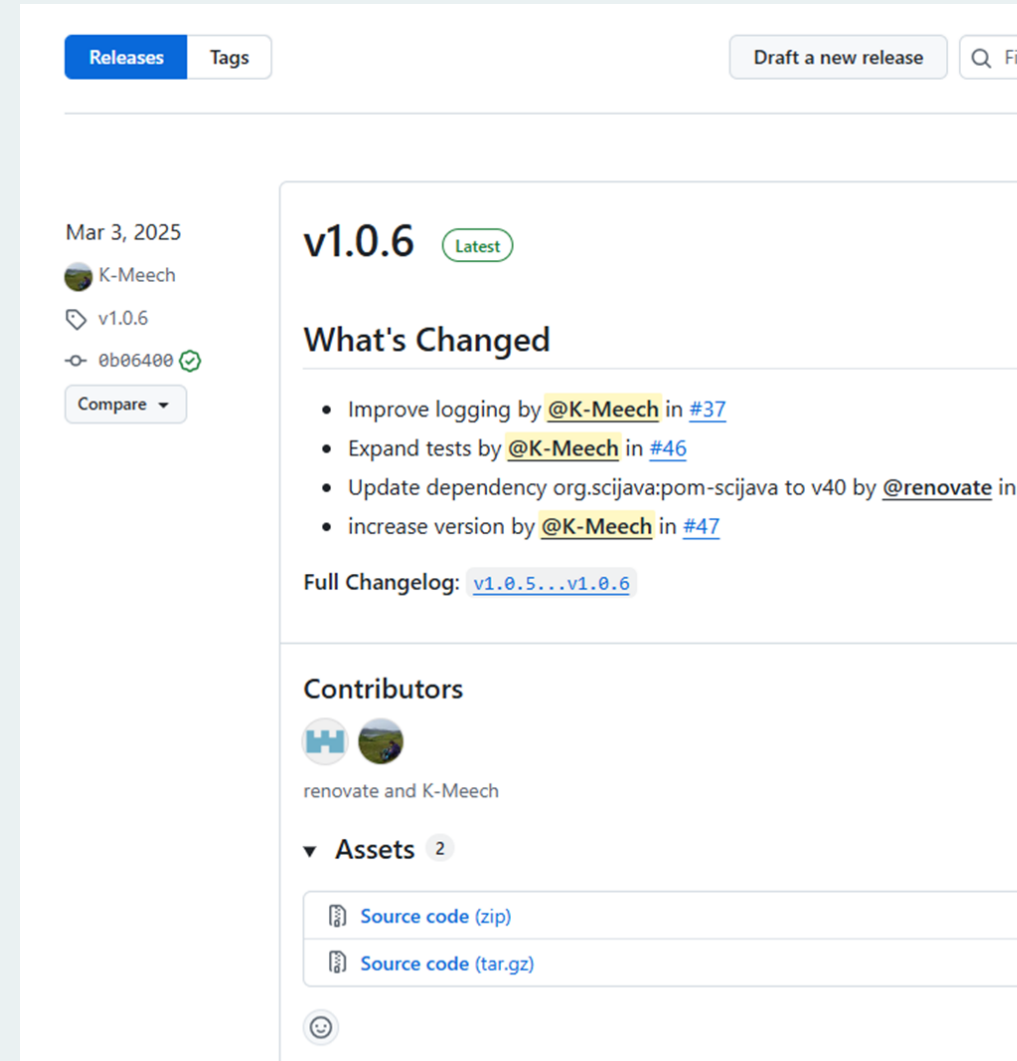
# Code coverage

- % of source code executed by automated tests
- Can't have good testing without good code coverage
- BUT can have good code coverage with bad testing
- Automated tools e.g. Codecov



# Releases / versioning

- Versioning helps other people use your package + understand when it changes
- When new code is made available to users of your software : new release
- Semantic versioning: <https://semver.org/>
- EffVer: <https://jacobtomlinson.dev/effver/>



The screenshot shows a GitHub release page for version v1.0.6. At the top, there are tabs for 'Releases' and 'Tags', and a button to 'Draft a new release'. The release is dated 'Mar 3, 2025' and is by user 'K-Meech'. It shows a commit hash '0b06400' with a green checkmark and a 'Compare' dropdown. The version 'v1.0.6' is marked as 'Latest'. The 'What's Changed' section lists four updates: 'Improve logging by @K-Meech in #37', 'Expand tests by @K-Meech in #46', 'Update dependency org.scijava:pom-scijava to v40 by @renovate in #45', and 'increase version by @K-Meech in #47'. The 'Full Changelog' link shows the range 'v1.0.5...v1.0.6'. The 'Contributors' section lists 'renovate and K-Meech'. The 'Assets' section shows two files: 'Source code (zip)' and 'Source code (tar.gz)'. A smiley face icon is at the bottom.

Releases Tags Draft a new release

Mar 3, 2025 K-Meech v1.0.6 0b06400 ✓ Compare

**v1.0.6** Latest

**What's Changed**

- Improve logging by @K-Meech in #37
- Expand tests by @K-Meech in #46
- Update dependency org.scijava:pom-scijava to v40 by @renovate in #45
- increase version by @K-Meech in #47

Full Changelog: [v1.0.5...v1.0.6](#)

**Contributors**

renovate and K-Meech

▼ Assets 2

- Source code (zip)
- Source code (tar.gz)

# Release notes

- Help yourself + others understand what has changed

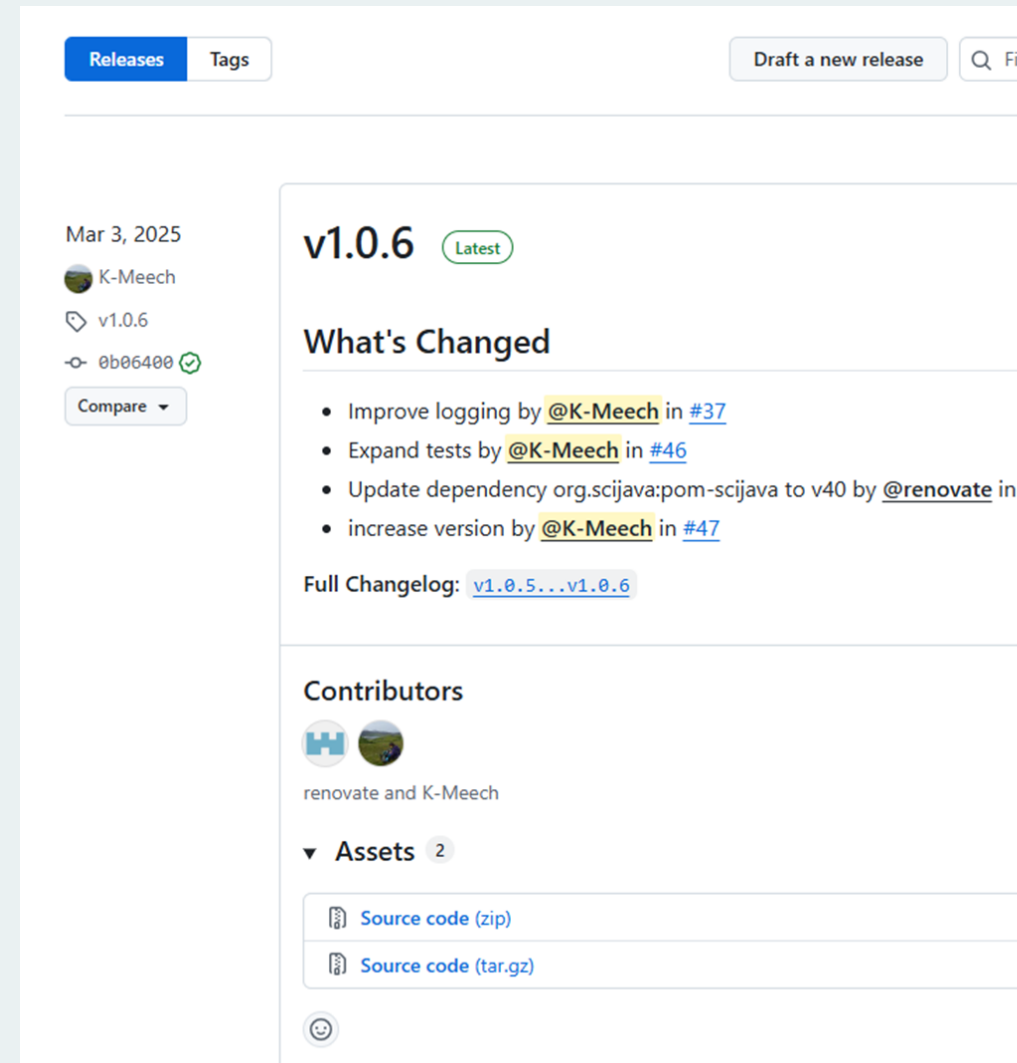
- GitHub auto-release notes

- Nicely edited Github release notes:

<https://github.com/napari/napari/releases>

- Release notes on a web-page:

[https://napari.org/dev/release/release\\_0\\_7\\_0.html](https://napari.org/dev/release/release_0_7_0.html)



The screenshot shows the GitHub release page for version 1.0.6. At the top, there are tabs for 'Releases' and 'Tags', and a button to 'Draft a new release'. The release is dated 'Mar 3, 2025' and was created by 'K-Meech'. The commit hash '0b06400' is shown with a green checkmark. A 'Compare' button is also visible. The release title is 'v1.0.6' with a 'Latest' badge. The 'What's Changed' section lists four items: 'Improve logging by @K-Meech in #37', 'Expand tests by @K-Meech in #46', 'Update dependency org.scijava:pom-scijava to v40 by @renovate in #47', and 'increase version by @K-Meech in #47'. The 'Full Changelog' link shows the range 'v1.0.5...v1.0.6'. The 'Contributors' section lists 'renovate' and 'K-Meech'. The 'Assets' section shows two files: 'Source code (zip)' and 'Source code (tar.gz)'. A smiley face icon is at the bottom left of the release card.

Releases Tags Draft a new release

Mar 3, 2025  
K-Meech  
v1.0.6  
0b06400 ✓  
Compare

**v1.0.6** Latest

**What's Changed**

- Improve logging by @K-Meech in #37
- Expand tests by @K-Meech in #46
- Update dependency org.scijava:pom-scijava to v40 by @renovate in #47
- increase version by @K-Meech in #47

Full Changelog: [v1.0.5...v1.0.6](#)

**Contributors**




renovate and K-Meech

▼ Assets 2

- Source code (zip)
- Source code (tar.gz)

# Documentation

- Make it easier for yourself + others to use your code
- GitHub readme
- GitHub wiki
- GitHub pages site
- Have user and developer focused docs
- Automated documentation building with GitHub Actions

 BrainGlobe

## BrainGlobe #


The BrainGlobe Initiative exists to facilitate the development of interoperable Python-based tools for computational neuroanatomy.

We have three aims:

- Develop specialist software for specific analysis and visualisation needs, such as [cellfinder](#) and [brainrender](#).
- Develop core tools to facilitate [others to build interoperable tools in Python](#), e.g., the [BrainGlobe Atlas API](#).
- Build a community of neuroscientists and developers to share knowledge, build software and engage with the scientific, and open-source community (e.g., by organising hackathons).

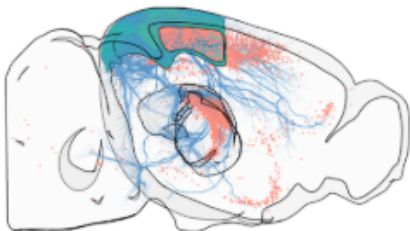
### ? About

What is the BrainGlobe Initiative?



### Tutorials

How do I get started with these tools?



### Documentation




Find out more about these

### Get in touch

Questions, feedback, or want

# Documentation in code

- Python docstrings
  - Different standards: numpydoc style  
<https://numpydoc.readthedocs.io/en/latest/format.html>
- Javadoc
- Provide auto-generated API docs:
  - <https://napari.org/dev/api/index.html>
  - <https://javadoc.scijava.org/ImgLib2/>

 BrainGlobe

## BrainGlobe #


The BrainGlobe Initiative exists to facilitate the development of interoperable Python-based tools for computational neuroanatomy.

We have three aims:

- Develop specialist software for specific analysis and visualisation needs, such as [cellfinder](#) and [brainrender](#).
- Develop core tools to facilitate [others to build interoperable tools in Python](#), e.g., the [BrainGlobe Atlas API](#).
- Build a community of neuroscientists and developers to share knowledge, build software and engage with the scientific, and open-source community (e.g., by organising hackathons).

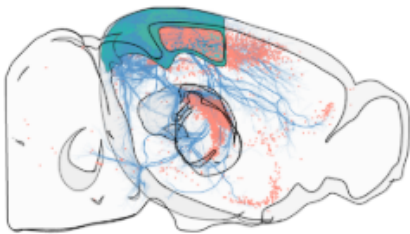
### ? About

What is the BrainGlobe Initiative?



### Tutorials

How do I get started with these tools?



### Documentation

Find out more about these

### Get in touch

Questions, feedback, or want

# Summary / further resources

- Many features you can add to help make a project more sustainable long-term
- It's a lot! Focus on what will have most impact for your project
- Ideally, add these early into a project

# Further resources

- ARC python tooling: <https://github-pages.arc.ucl.ac.uk/python-tooling/>
- Turing way book: <https://book.the-turing-way.org/>
- Software carpentry:
  - Git intro: <https://swcarpentry.github.io/git-novice/>
  - Intermediate research software development: <https://carpentries-incubator.github.io/python-intermediate-development/>

- Would be great to make a resource specifically for fiji/napari plugins