UNIVERSITY OF CAMBRIDGE

Institute of Computing for Climate Science

Software Reproducibility

What is means, why it matters, and how to do it.

Marion Weinzierl

Institute of Computing for Climate Science (ICCS), Cambridge University

What do I mean by Software Reproducibility?

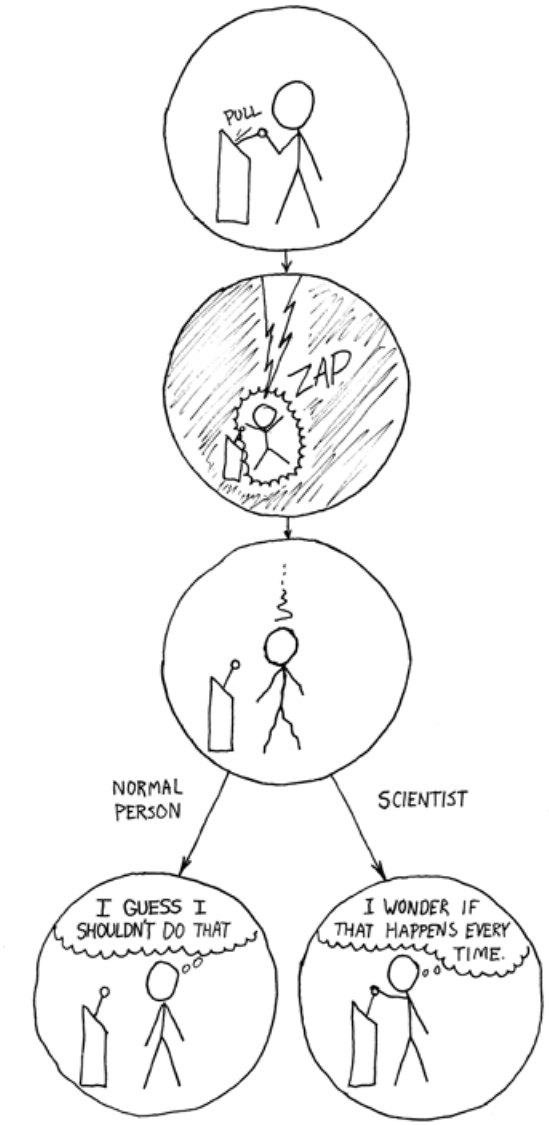


Image by xkcd.com

What do I mean by Software Reproducibility?

If I would go to your code repository, download the code and follow the instructions in the README/documentation, would I be able to reproduce the results from your publication?

What do I mean by Software Reproducibility?

If I would go to your code repository, download the code and follow the instructions in the README/documentation, would I be able to reproduce the results from your publication?

Will I even be able to run your code?

What do I mean by Software Reproducibility?

If I would go to your code repository, download the code and follow the instructions in the README/documentation, would I be able to reproduce the results from your publication?

Will I even be able to run your code?

Or compile it?

What do I mean by Software Reproducibility?

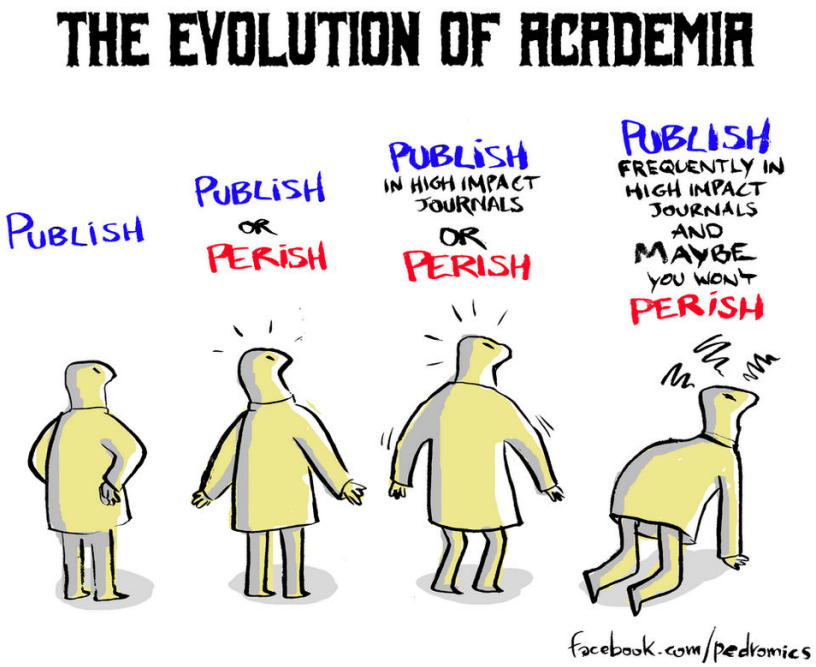
If I would go to your code repository, download the code and follow the instructions in the README/documentation, would I be able to reproduce the results from your publication?

Will I even be able to run your code?

Or compile it?

Do you even have a repository that I can find?

Why is Software Reproducibility an issue?



- Pressure to publish/write grants/teach/administrate... in academia
  - → Software is not a priority
  - → Researchers are (mostly) not software engineers
  - → Better software is no progression criterion
- Not just a problem in software-based research

→ Replication/Reproducibility crisis

Reproducibility Crisis

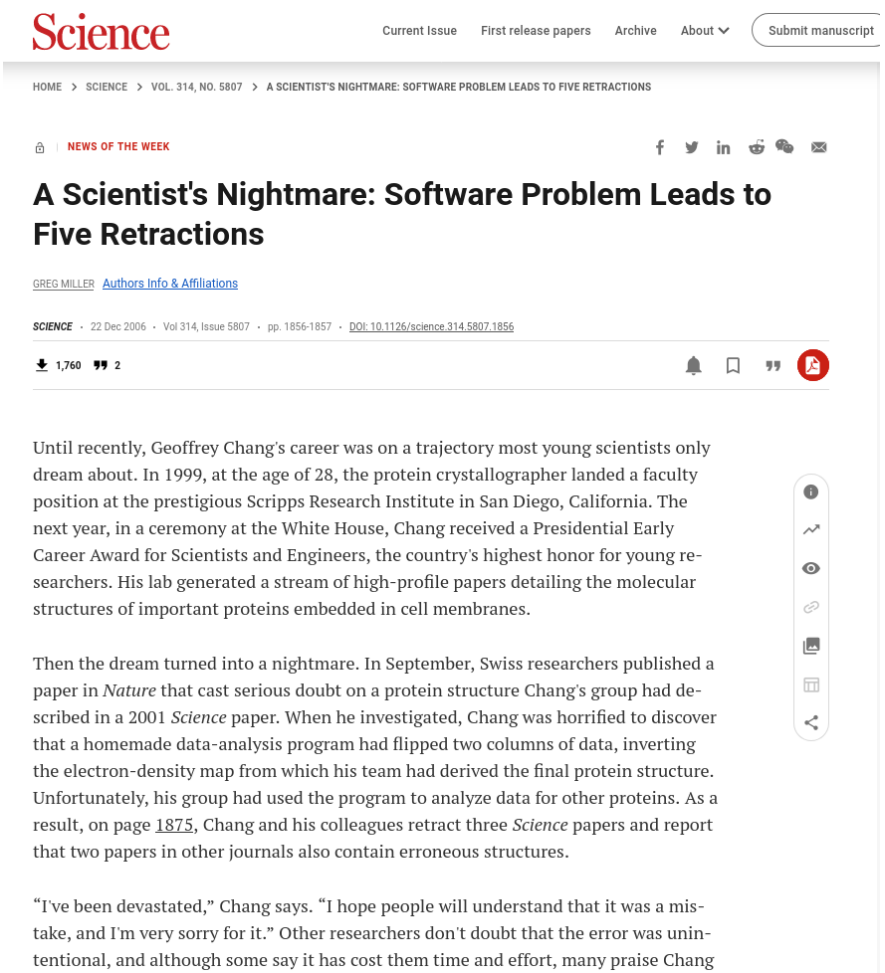
- *Why Most Published Research Findings Are False*, John Ioannidis, 2005
- Several controversies in the 2010s about influential results from studies which could not be replicated
- Criticism of research practices

$\Rightarrow$ But what are the incentives?

Fear...

Un-checked, un-tested, un-documented low-quality code can lead to...

- paper retractions
- undetected false results
- software sustainability issues
- decisions taken based on false software results
- lack of trust



Science

Current Issue First release papers Archive About Submit manuscript

HOME > SCIENCE > VOL. 314, NO. 5807 > A SCIENTIST'S NIGHTMARE: SOFTWARE PROBLEM LEADS TO FIVE RETRACTIONS

NEWS OF THE WEEK

A Scientist's Nightmare: Software Problem Leads to Five Retractions

GREG MILLER Authors Info & Affiliations

SCIENCE · 22 Dec 2006 · Vol 314, Issue 5807 · pp. 1856-1857 · DOI: 10.1126/science.314.5807.1856

1,760 2

Until recently, Geoffrey Chang's career was on a trajectory most young scientists only dream about. In 1999, at the age of 28, the protein crystallographer landed a faculty position at the prestigious Scripps Research Institute in San Diego, California. The next year, in a ceremony at the White House, Chang received a Presidential Early Career Award for Scientists and Engineers, the country's highest honor for young researchers. His lab generated a stream of high-profile papers detailing the molecular structures of important proteins embedded in cell membranes.

Then the dream turned into a nightmare. In September, Swiss researchers published a paper in *Nature* that cast serious doubt on a protein structure Chang's group had described in a 2001 *Science* paper. When he investigated, Chang was horrified to discover that a homemade data-analysis program had flipped two columns of data, inverting the electron-density map from which his team had derived the final protein structure. Unfortunately, his group had used the program to analyze data for other proteins. As a result, on page 1875, Chang and his colleagues retract three *Science* papers and report that two papers in other journals also contain erroneous structures.

"I've been devastated," Chang says. "I hope people will understand that it was a mistake, and I'm very sorry for it." Other researchers don't doubt that the error was unintentional, and although some say it has cost them time and effort, many praise Chang

Positive Incentives

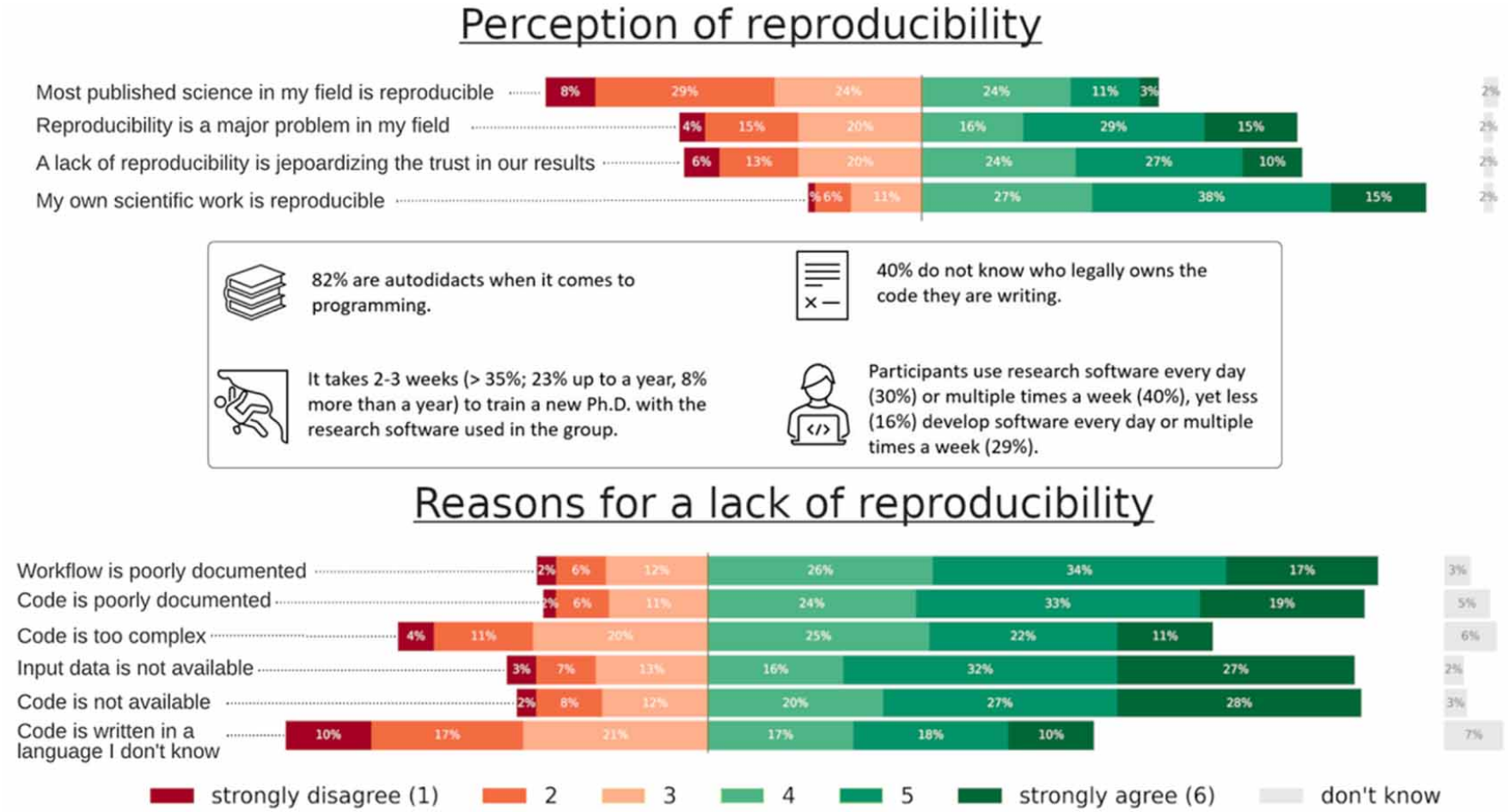
- Easier rerunning of experiments when something changes
- Reducing technical debt
- Saving time and computational resources (?)
- Smoother onboarding of new team members
- Better collaboration

Software Reproducibility in the Earth Science Community

The critical need to foster computational reproducibility,
Reinecke et al., 2022, Environ. Res. Lett. **17** 041005

"Our non-representative poll through a web-based survey revealed that:
(a) the lack of reproducibility is jeopardizing trust in computational research,
(b) that there is a considerable lack of knowledge on establishing software development methods, and
(c) that Open Science is still not widely practiced."

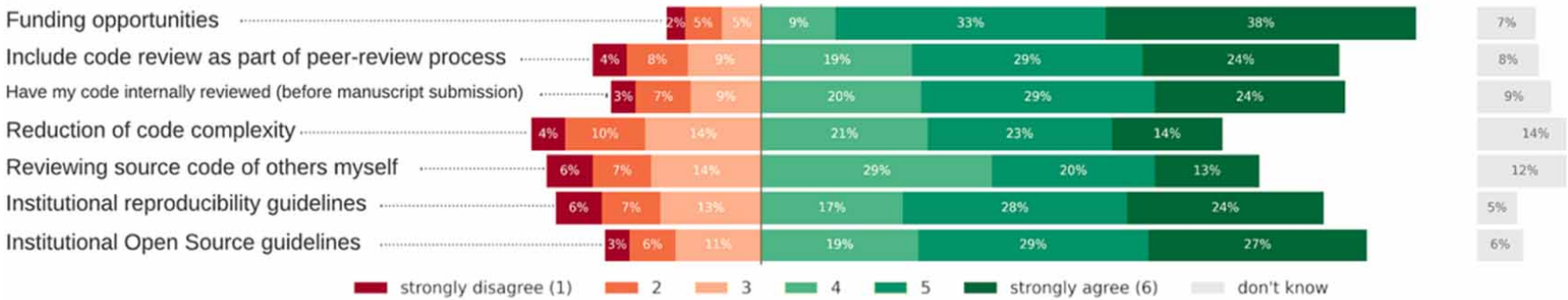
Software Reproducibility in the Earth Science Community



The critical need to foster computational reproducibility, Reinecke et al., 2022

Software Reproducibility in the Earth Science Community

Solutions



(1) Sharing is key, and journals should support this transition

"SHARE! Share your data, share your code!"

"Journals should move towards requiring experimental reproducibility as a part of criteria for publication. [..]" and "Journals should push for open and reproducible code and open data. [..]"

(2) We need to teach the suitable methods or require specialized staff

"[..] Versioning, packaging, long-form documentation, testing, continuous integration and deployment should become widely accepted standards in scientific software development as they have become in the private sector."

"Hire more research software engineers that oversee sustainable software development practices and help with standardising, testing, and publishing research software."

(3) We require a change in funding and recognition

"[..] it is impossible to get funding for 'redoing something that was already done before'"

"If you don't have a permanent position, putting effort into maintaining software for external users [..] means losing time to publish papers and keep your job."

(4) Some hurdles may not be easy to overcome

"In the climate community this is not very easily addressed. Reproducing the output of global climate models would be absurdly time and resource consuming. [..]"

"I am overwhelmed with the endless variety of licenses, programing languages, sharing repositories, the constant migration towards newer approaches that make older ones obsolete. [..]"

The critical need to foster computational reproducibility, Reinecke et al., 2022

Things change (in some parts of the scientific community)

- Various groups and organisations work for better reproducibility.
- Conferences and journals start to ask for software, data etc. to back up research findings.
- Software sustainability and research software engineering have become a thing (internationally).
- But still not widely known outside of the bubble!

UK Reproducibility Network

- Peer-led consortium within the UK
- National Steering Group, local and institutional groups
- Training, events
- Engaging with stakeholders
- `https://www.ukrn.org/international-networks/`

Multi-discipline

UKRN works across disciplines, ranging from the arts and humanities to the physical sciences, with a particular focus on the biomedical sciences.

Meta-research

UKRN supports research improvement through the study of research itself, and promoting evaluation of innovations to improve research quality.

Training support

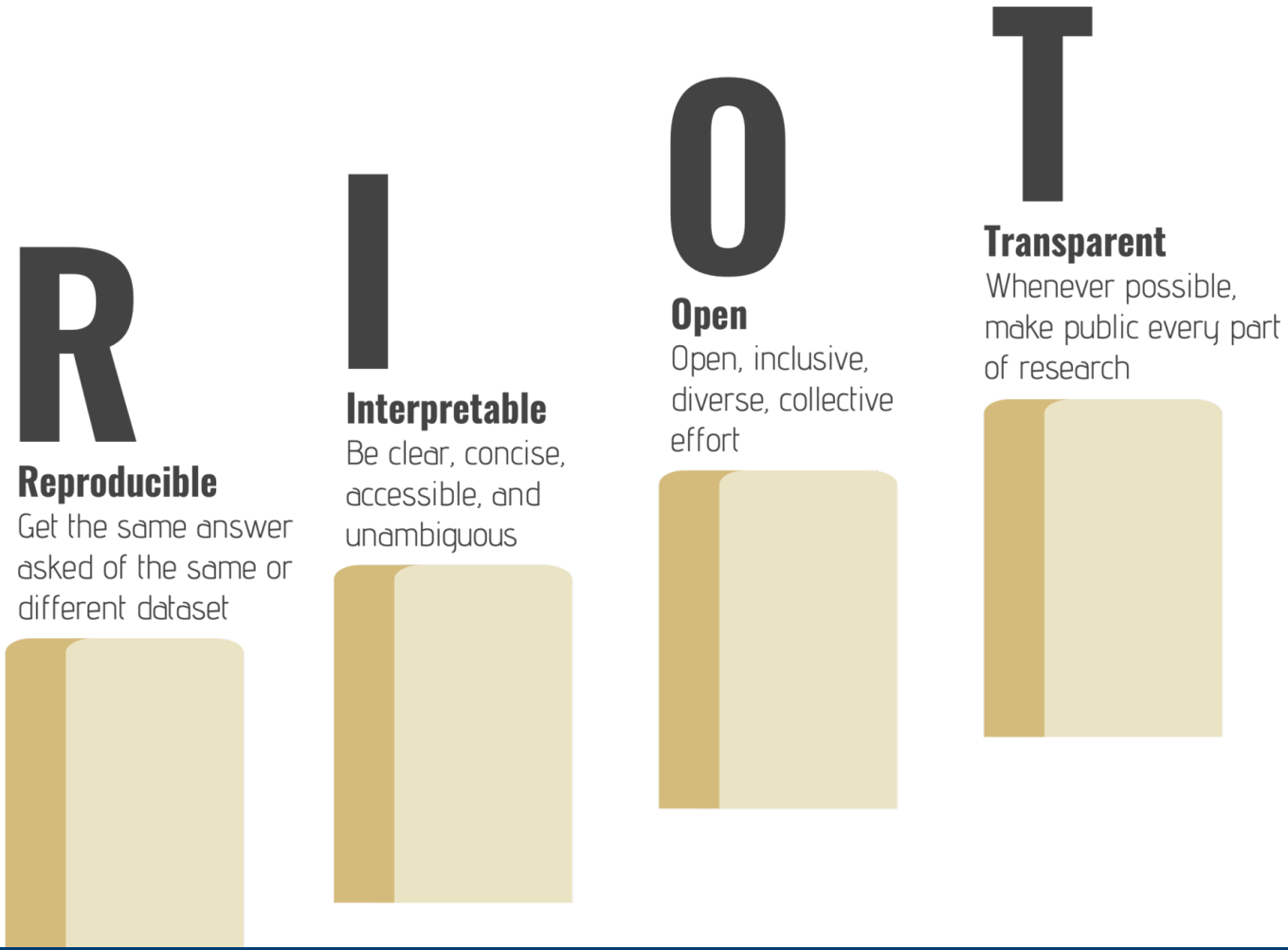
UKRN promotes training through the delivery of short courses, preparation of online materials and the coordination of virtual seminars and journal clubs.

Distributed network

UKRN follows a distributed model where Local Networks can be supported by individuals acting as advocates for the work of UKRN, or formally by their institution.

RIOT Science

- Groups at universities (mostly Psychology, mostly UK)
- Conferences, events, seminars open to everyone



The Turing Way

- Handbook
- Community
- Collaboration

- Findability: *Software, and its associated metadata, is easy for both humans and machines to find.*
- Accessibility: *Software, and its metadata, is retrievable via standardized protocols.*
- Interoperability: *Software interoperates with other software by exchanging data and/or metadata, and/or through interaction via application programming interfaces (APIs), described through standards.*
- Reusability: *Software is both usable (can be executed) and reusable (can be understood, modified, built upon, or incorporated into other software)*

ACM Reproducibility Badges

Association for Computing Machinery

SC Reproducibility Initiative

- SC (formerly Supercomputing), The International Conference for High Performance Computing Networking, Storage, and Analysis
- Initiative started in 2015, Artifact Descriptions (ADs) optional for the first years $\rightarrow$ used in Student Cluster Competition
- Then gradually made mandatory for more categories/prizes
- Computational Results Analysis (CRA) $\rightarrow$ Artifact Evaluation (AE) appendix still optional
- AD/AE committee evaluates appendices and recommends ACM badge awards (IEEE badges seem to have vanished)
- Reproducibility challenge introduced 2021

ReproHack

- Challenge: Reproduce the results of a paper in one day!
- Started in 2016 and 2017 as satellite events of OpenCon (inspired by a course by Owen Petchey)
- Developed further by Anna Krystalli in her SSI fellowship
- More events, a team formed, remote ReproHacks became a thing....
- ReproHack Hub launched in 2021
  - Material and checklists for organisers
  - Paper database
  - Evaluation forms
  - Events listing
  - Support through ReproHack Slack

And more!

- JOSS
- ReScience C
- CODECHECK
- ML Reproducibility Challenge
- Climate Informatics Reproducibility Challenge
- ...

Practical Steps

- **Code repository & version control**
- **Documentation**
- **Automatisation**
- Testing and Continuous Integration
- Notebooks and containers
- Reproducibility review / ReproHack

Photo by Estudio Bloom on Unsplash

Code repository

- Github,Gitlab, Bitbucket,...
- Learn how to use it for collaboration and project management!
- Tag code version for results.
- DOI e.g. through Github-Zenodo integration
- Add a license so that users know what they are allowed to do.

README and Documentation

- (In-code)
- What does the code do?
- Installation incl. prerequisites, libs and packages
- Quick-start guide
- Data
- Parameters
- Minimal working examples
- Expected output
- How to produce the figures
- (All the rest, like citation, contributing guidelines etc.)

(Semi-)Automatisation

- Scripts to run the simulation pipeline
- Scripts for figures
- Scripts for tests
- This will save you a lot of time in future!

Testing and Continuous Integration

- Using the before-mentioned scripts
- Regression tests
- Unit tests
- Integration tests
- CI through Github Actions, Gitlab Runners, Jenkins, CircleCI,...
- Performance Regression

Notebooks and Containers

- Containers: Docker, Singularity
  - Virtual operating system
  - Set up the system and do the installation for the user
- Jupyter notebooks and similar
  - Text blocks and code blocks
  - User just needs to click to execute
- Question: What level of reproducibility is that?

Reproducibility review / ReproHack

- If I would go to your code repository, download the code and follow the instructions in the README/documentation...?
- We don't want to look into the code to be able to run it.
- Ask a colleague who doesn't know your code.
- Make it part of the onboarding process.
- Attend a ReproHack/join a reproducibility initiative to learn by reviewing.

Support from the ICCS Research Software Engineers

Institute of Computing for Climate Science

- Code Reviews
- RSE support through ICCS Resource Allocation Process
- ICCS Climate Code Clinic
- ICCS ReproHack in March 2024 (probably March 12)
- Talk to us!

References

- https://sc23.supercomputing.org/program/papers/reproducibility-initiative/
- https://sc23.supercomputing.org/program/papers/reproducibility-appendices-badges/
- https://www.reprohack.org/
- https://codecheck.org.uk/project/
- https://www.ukrn.org/
- https://riotscience.co.uk/
- https://joss.theoj.org/
- http://rescience.github.io/
- https://www.acm.org/publications/policies/artifact-review-and-badging-current

References

- https://en.wikipedia.org/wiki/Replication_crisis
- https://www.software.ac.uk/blog/2017-02-20-software-reproducibility-possible-and-practical
- https://iopscience.iop.org/article/10.1088/1748-9326/ac5cf8
- https://www.nature.com/articles/s41597-022-01710-x

Thanks for listening! Questions?

Contact me

- in the VESRI+CliMA Communities Slack workspace
- at mw925@cam.ac.uk