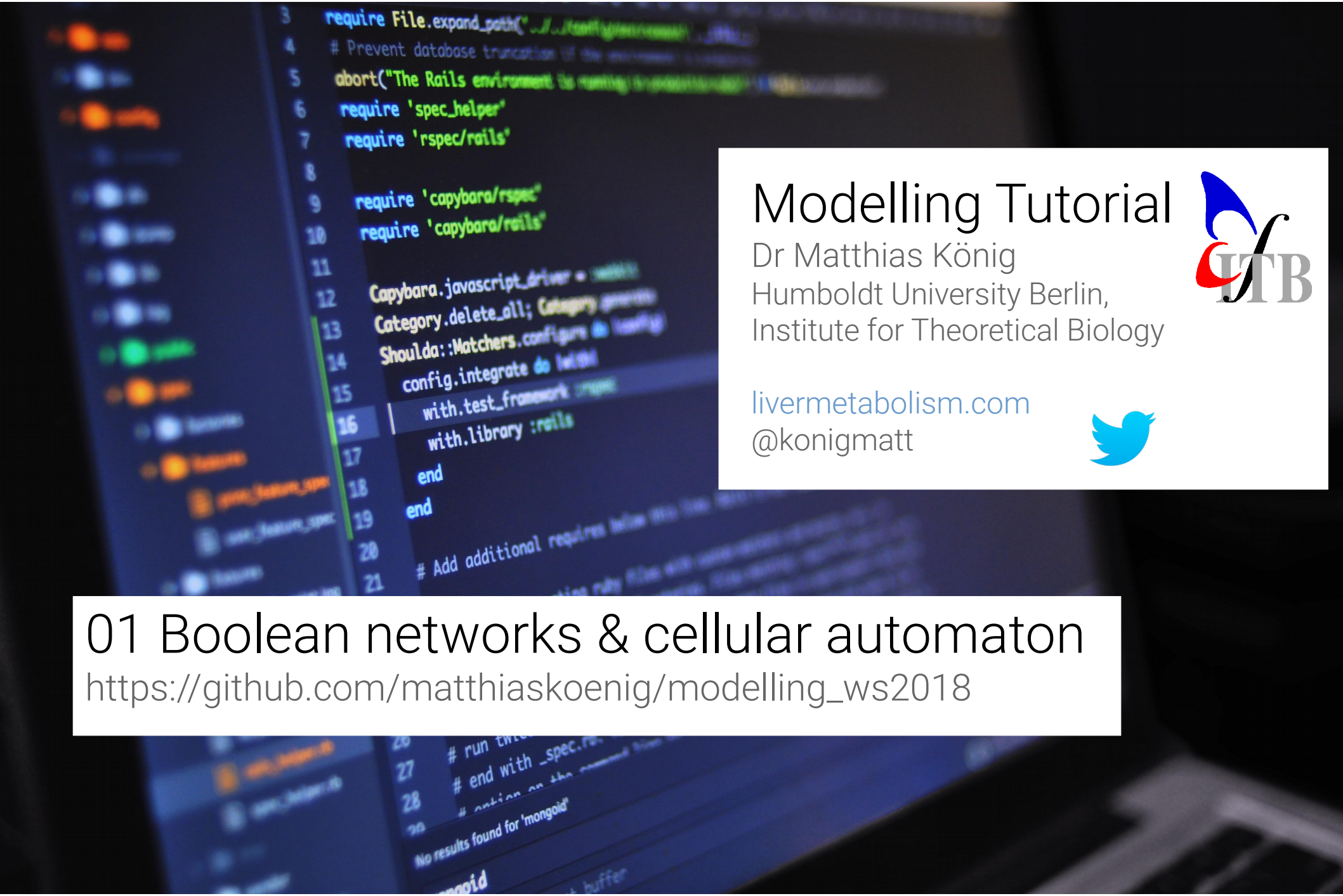


Modelling Tutorial

Dr Matthias König
Humboldt University Berlin,
Institute for Theoretical Biology

livermetabolism.com
@konigmatt

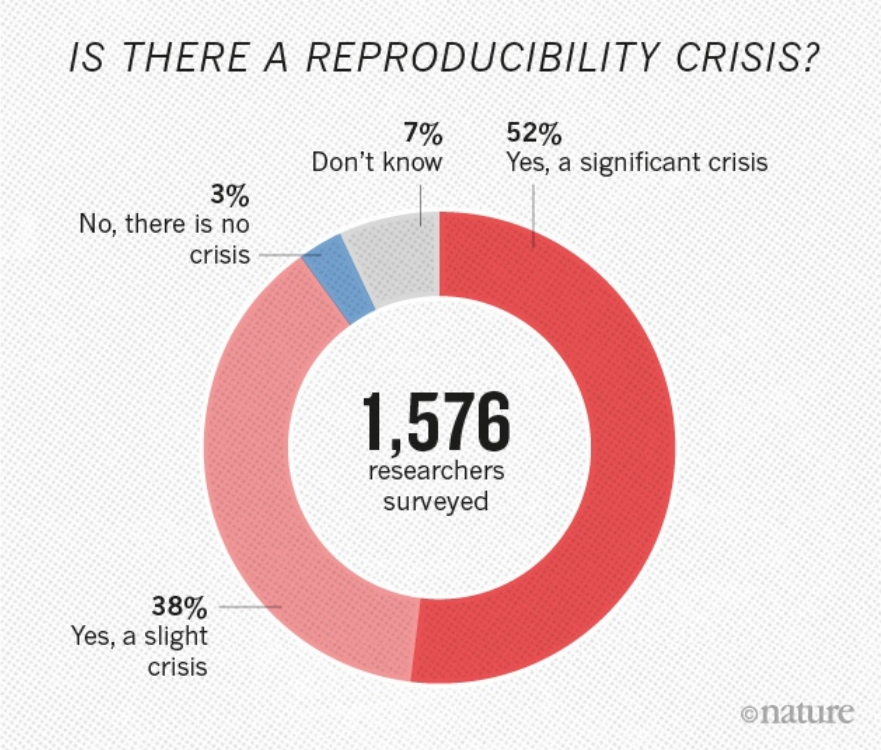
01 Boolean networks & cellular automaton

https://github.com/matthiaskoenig/modelling_ws2018

Overview

- Version control & git
- Python virtual environments
- Solution to the problems
  - Boolean networks
  - Cellular Automaton

Reproducibility/Reusability Crisis



"Really Reproducible Research" (1992) inspired by Stanford Professor Jon Claerbout:

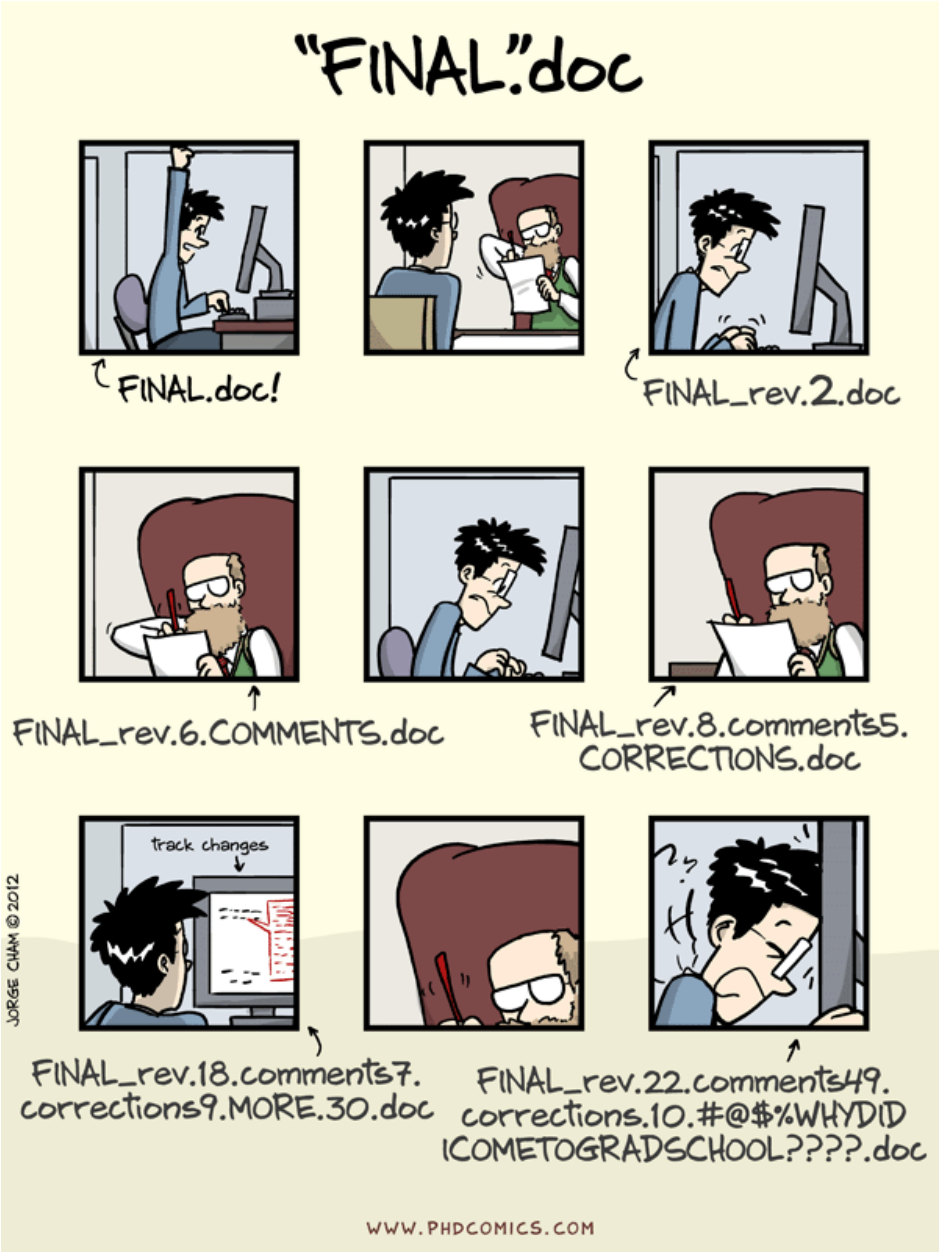
"The idea is: **An article** about computational science in a scientific publication is not the scholarship itself, it **is merely advertising of the scholarship**. The **actual scholarship is the complete ... set of instructions [and data]** which generated the figures."

David Donoho, 1998

How long would it take you to reproduce your results from your last programming project?

https://www.nature.com/news/1-500-scientists-lift-the-lid-on-reproducibility-1.19970
https://web.stanford.edu/~vcs/talks/MSKCC-Sept15-2015-STODDEN.pdf

Versioning Crisis



git to the rescue

- **Git** is a free and open source **distributed version control system**.
  - tiny footprint
  - lightning fast performance
  - works for everything from small to very large projects

- **GitHub** is a **code hosting platform for version control and collaboration**.
  - social coding
  - alternatives: GitLab, BitBucket, SourceForge

What can git do for you?

- Work anywhere & offline
- Decentralized backups
- Version control & track changes
  - diffs & branches
- Revert & experiment
  - revisions & branches
- Collaborative editing/work
  - Pull requests
- Reproducible research
- Releases & snapshots
  - citable code
- Issue tracker
- Continuous integration
  - unit tests
  - commit hooks

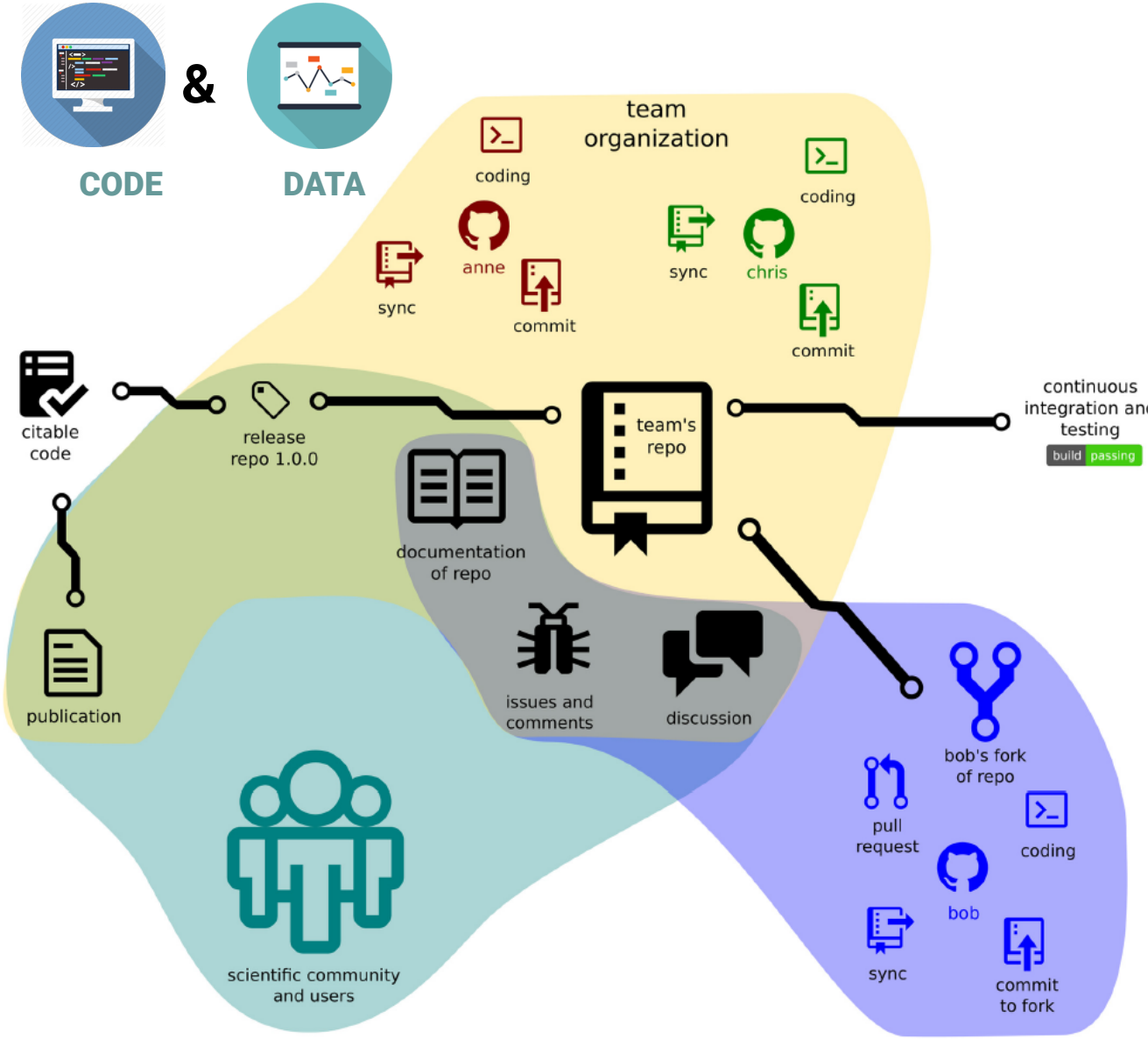
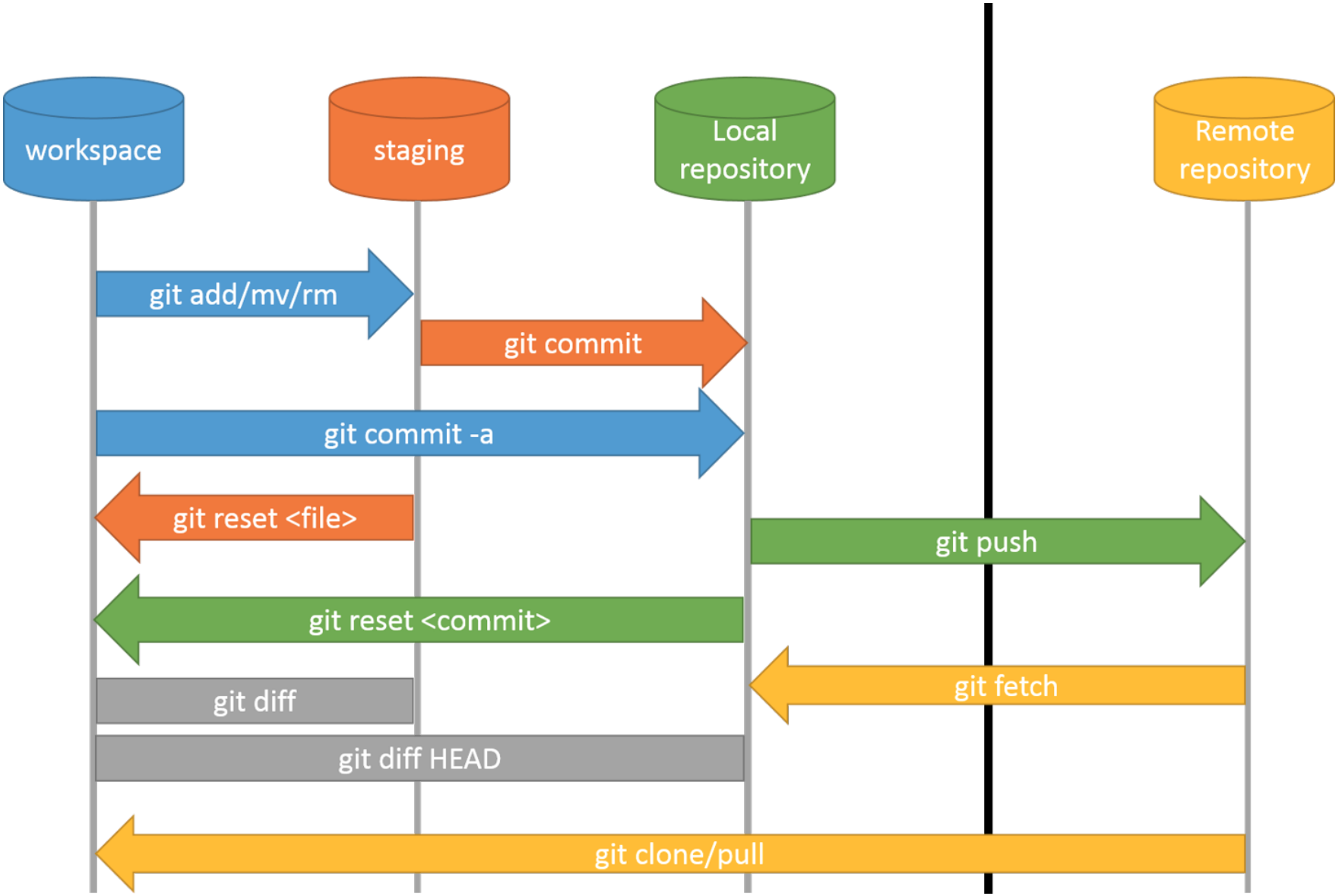


Fig 1. The structure of a GitHub-based project illustrating project structure and interactions with the community.

doi:10.1371/journal.pcbi.1004947.g001

Perez.Riverol2016

How does git work?



http://blog.podrezo.com/git-introduction-for-cvssvntfs-users/

Python Package Management

Collection of available packages/libraries in the environment

- starts out fine, overtime **things get messy**
- problematic for developing in teams/multiple computers (**sync issues**)
- not working if multiple projects require different versions (python/package)
- **often no control** over environment (compute server, deployment environments)
- **easy to break** things

"Earlier you said we are losing ground. Could you be more specific?"

Virtual environments

- **Virtual environment**
  Isolated python environment with own python executable and dependencies
  - reproducible (contains dependency & system setup)
  - isolated (does not affect system)
  - exchangable (computers & teams)

- **virtualenv** is a tool to create **python virtual environments**
  - virtualenv creates a folder which contains all the necessary executables to use the packages that a Python project would need.
- Other alternatives **conda**

Python in a Box

References

- Perez-Riverol, Y.; Gatto, L.; Wang, R.; Sachsenberg, T.; Uszkoreit, J.; Leprevost, F. d. V.; Fufezan, C.; Ternent, T.; Eglen, S. J.; Katz, D. S.; Pollard, T. J.; Konovalov, A.; Flight, R. M.; Blin, K. & Vizcaíno, J. A.
  Ten Simple Rules for Taking Advantage of Git and GitHub.
  PLoS computational biology, 2016, 12, e1004947

- Blischak, J. D.; Davenport, E. R. & Wilson, G.
  A Quick Introduction to Version Control with Git and GitHub.
  PLoS computational biology, 2016, 12, e1004668

- Ram, K.
  Git can facilitate greater reproducibility and increased transparency in science.
  Source code for biology and medicine, 2013, 8, 7

- **Tutorials**
  https://try.github.io/levels/1/challenges/1

- **Information/Books**
  https://git-scm.com/
  https://git-scm.com/book/en/v2

References

- http://docs.python-guide.org/en/latest/dev/virtualenvs/
- https://virtualenvwrapper.readthedocs.io/en/latest/
- https://www.slideshare.net/ryan_blunden/virtualenv-and-pip-isolated-python-environments