



“DevOps” for Systems Biology

Jun Allard

Associate Professor

Department of Mathematics

Department of Physics and Astronomy

Center for Complex Biological Systems

NSF-Simons Center for Multiscale Cell Fate

Director, Graduate Gateway program in Mathematical, Computational and Systems Biology

University of California, Irvine

Twitter: @allardlab

Recent worldwide events have challenged the Mathematical Biology community in two major ways:

First, a global pandemic has elevated mathematical biology, and mathematical biologists, to the world stage to assist with, e.g., predictive epidemiology. This has revealed that, if the field is to maintain impact and respect, e.g., with policy decision-makers, our standards of rigor in coding practices need to be elevated too. Our lack of rigor in coding practices is particularly troubling, since mathematical rigor has been a point of pride in the field. It also represents an opportunity.

Second, the rapid move of education, outreach and research to remote modes of collaboration has challenged old modes of producing science. This also represents an opportunity, particularly for the inclusion of students and community members who may thrive in remote modes and asynchronous collaborative modes, due to, e.g., geographic factors, or students with external responsibilities limiting their time flexibility.



neil_ferguson ✅
@neil_ferguson

I'm conscious that lots of people would like to see and run the pandemic simulation code we are using to model control measures against COVID-19. To explain the background - I wrote the code (thousands of lines of undocumented C) 13+ years ago to model flu pandemics...

2:13 PM · Mar 22, 2020 · Twitter for iPhone

1.4K Retweets 4.9K Likes



neil_ferguson ✅ @neil_ferguson · Mar 22

Replying to @neil_ferguson

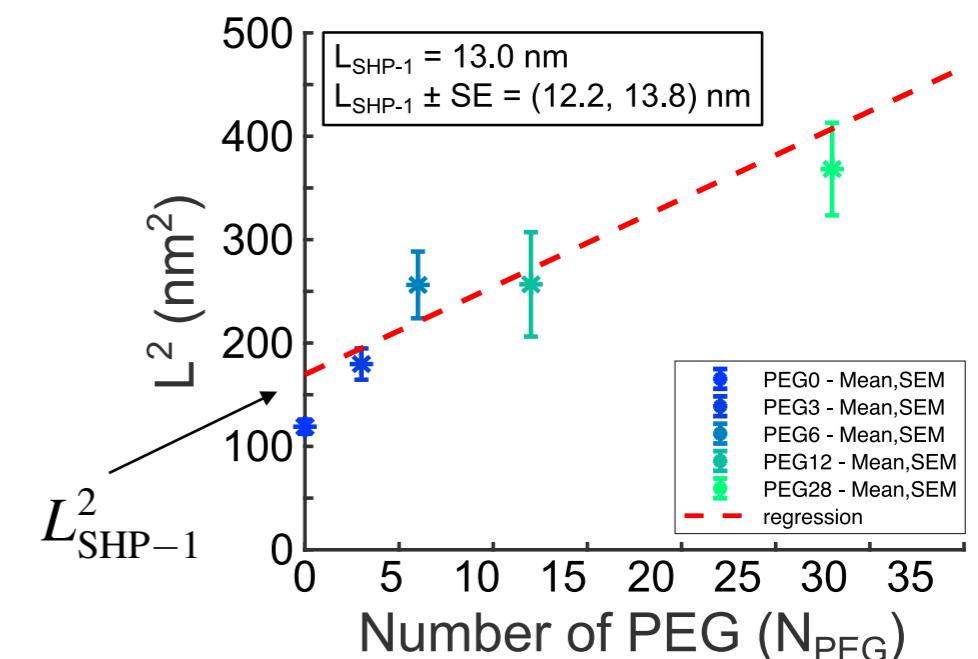
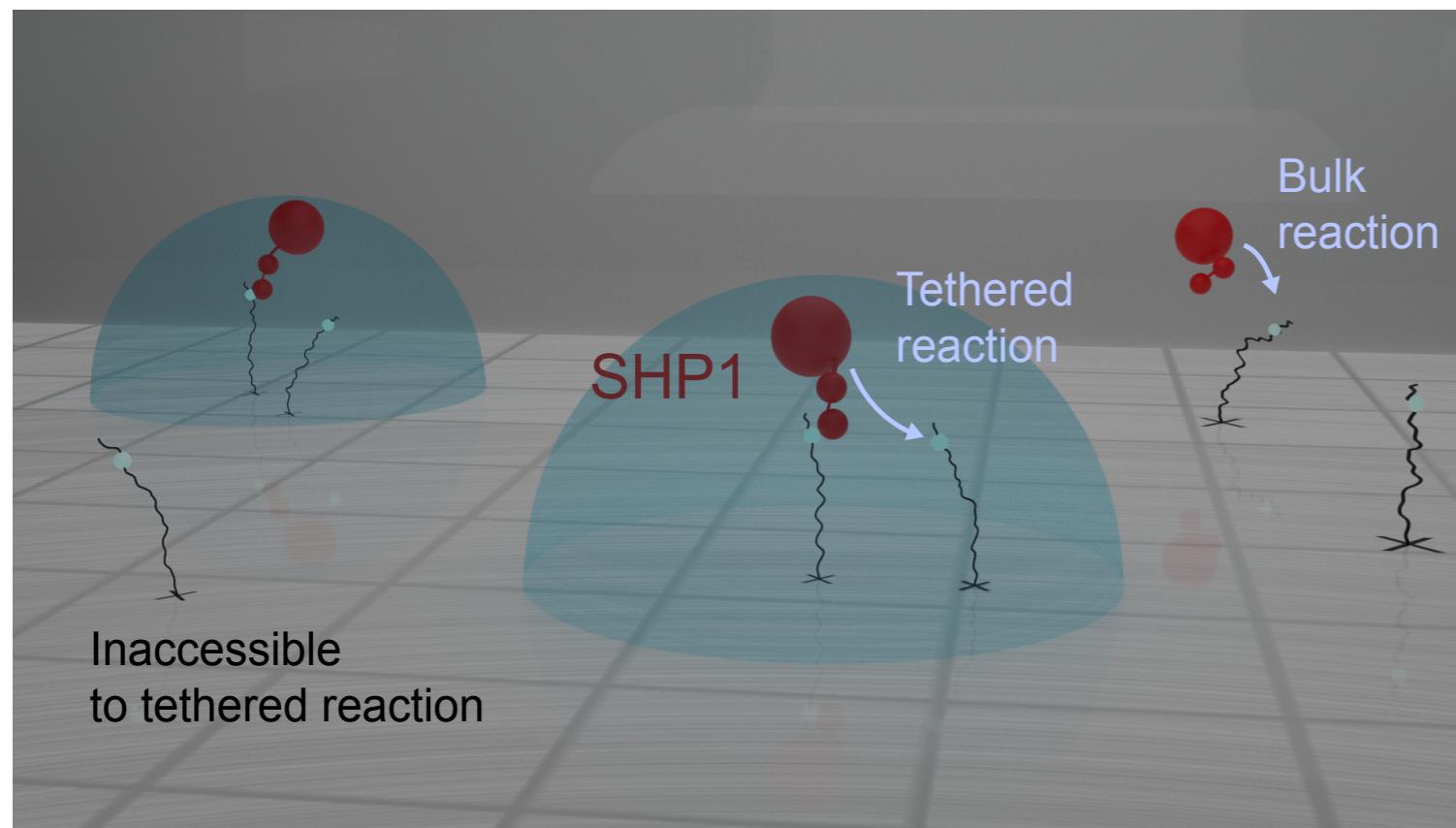
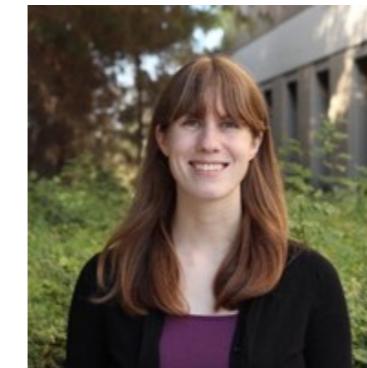
I am happy to say that @Microsoft and @GitHub are working with @Imperial_JIDEA and @MRC_Outbreak to document, refactor and extend the code to allow others to use without the multiple days training it would currently require (and which we don't have time to give)...

2054 Biophysical Journal 120, 2054–2066, May 18, 2021

Determination of the molecular reach of the protein tyrosine phosphatase SHP-1

Lara Clemens,¹ Mikhail Kutuzov,² Kristina Viktoria Bayer,² Jesse Goyette,^{3,4} Jun Allard,^{1,*} and Omer Dushek^{2,*}

¹Center for Complex Biological Systems, University of California Irvine, Irvine, California; ²Sir William Dunn School of Pathology, University of Oxford, Oxford, United Kingdom; ³EMBL Australia Node in Single Molecule Science, School of Medical Sciences University of New South Wales, Sydney, Australia and ⁴ARC Centre of Excellence in Advanced Molecular Imaging, University of New South Wales, Sydney, Australia



$$L^2 = 4 \times N_{\text{PEG}} \times l_{\text{PEG}} \times l_p + L_{\text{SHP-1}}^2$$

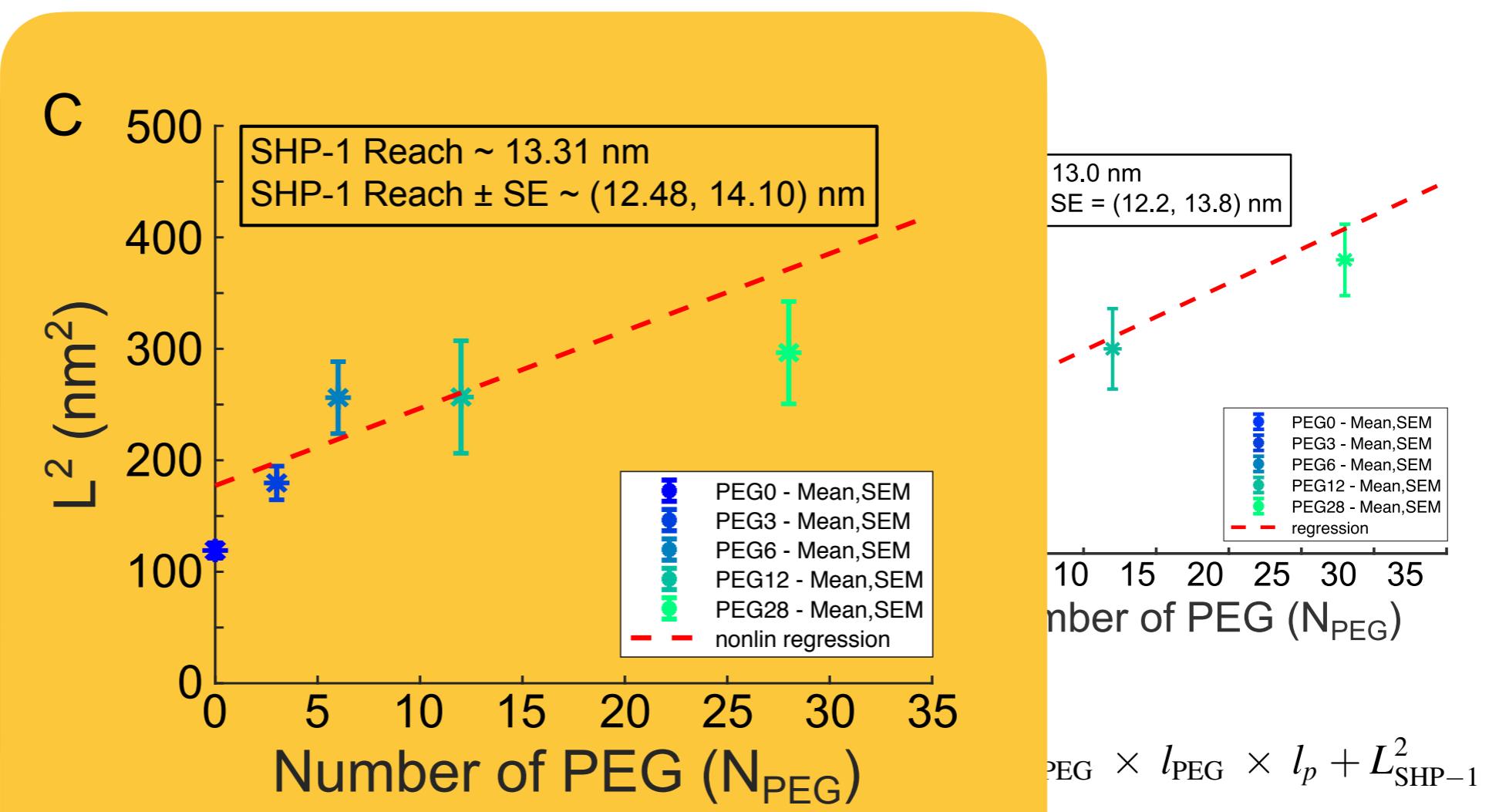
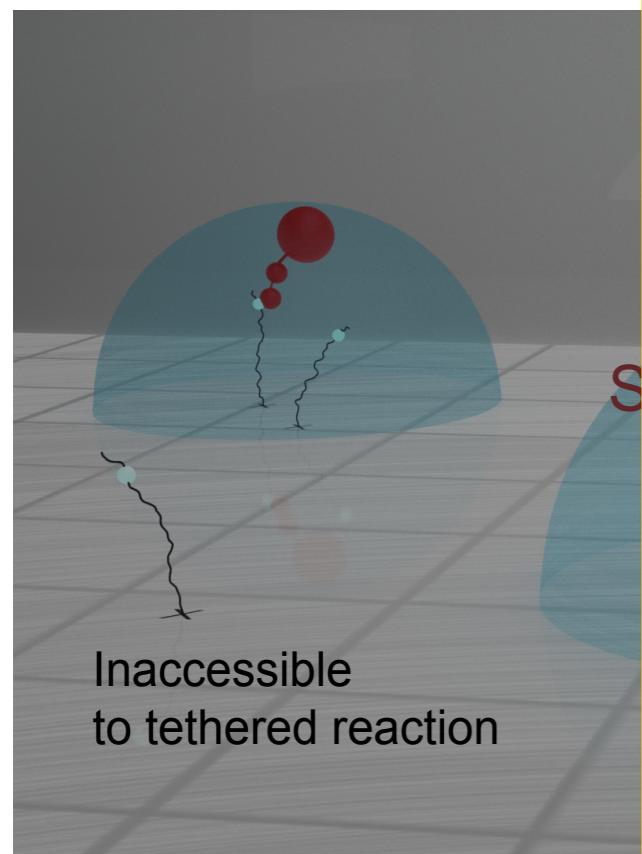
(L2 is a linear function of NPEG)

2054 Biophysical Journal 120, 2054–2066, May 18, 2021

Determination of the molecular reach of the protein tyrosine phosphatase SHP-1

Lara Clemens,¹ Mikhail Kutuzov,² Kristina Viktoria Bayer,² Jesse Goyette,^{3,4} Jun Allard,^{1,*} and Omer Dushek^{2,*}

¹Center for Complex Biological Systems, University of California Irvine, Irvine, California; ²Sir William Dunn School of Pathology, University of Oxford, Oxford, United Kingdom; ³EMBL Australia Node in Single Molecule Science, School of Medical Sciences University of New South Wales, Sydney, Australia and ⁴ARC Centre of Excellence in Advanced Molecular Imaging, University of New South Wales, Sydney, Australia



Software engineering

Software engineering



Software engineering



Software engineering



Product development

Software engineering



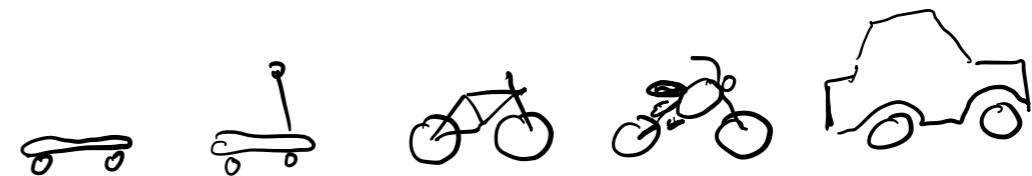
Product development



Software engineering



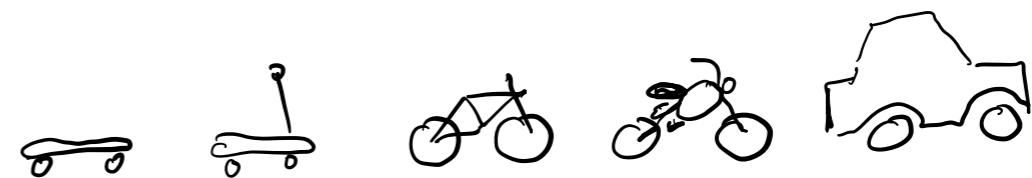
Product development



Software engineering



Product development

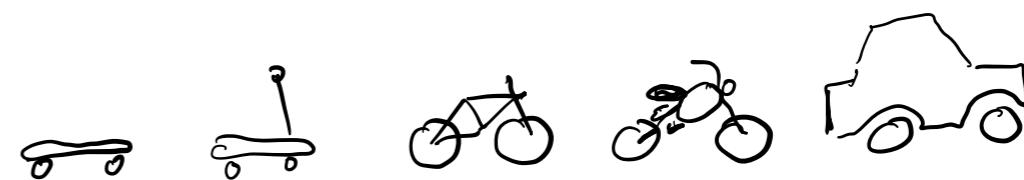


Henrik Kniberg

Software engineering



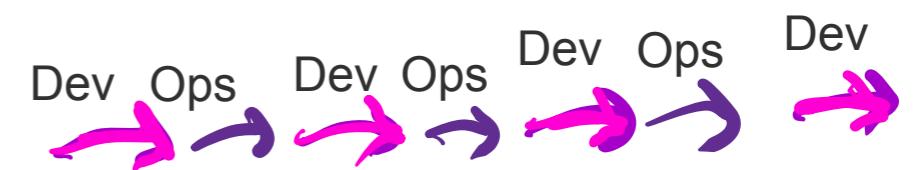
Product development



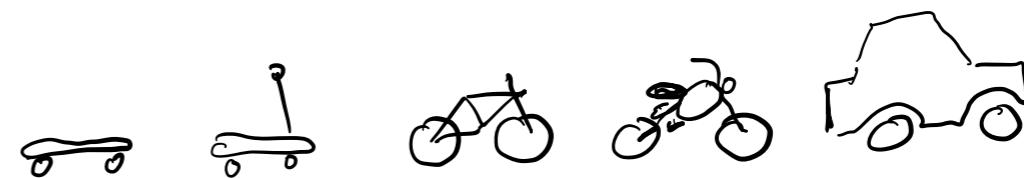
Henrik Kniberg

Science

Software engineering



Product development



Henrik Kniberg

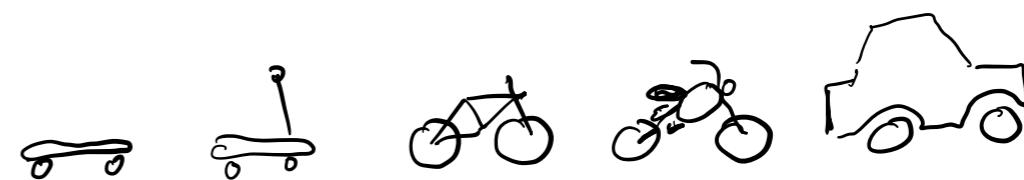
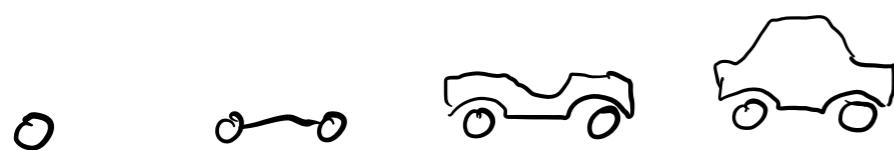
Science



Software engineering

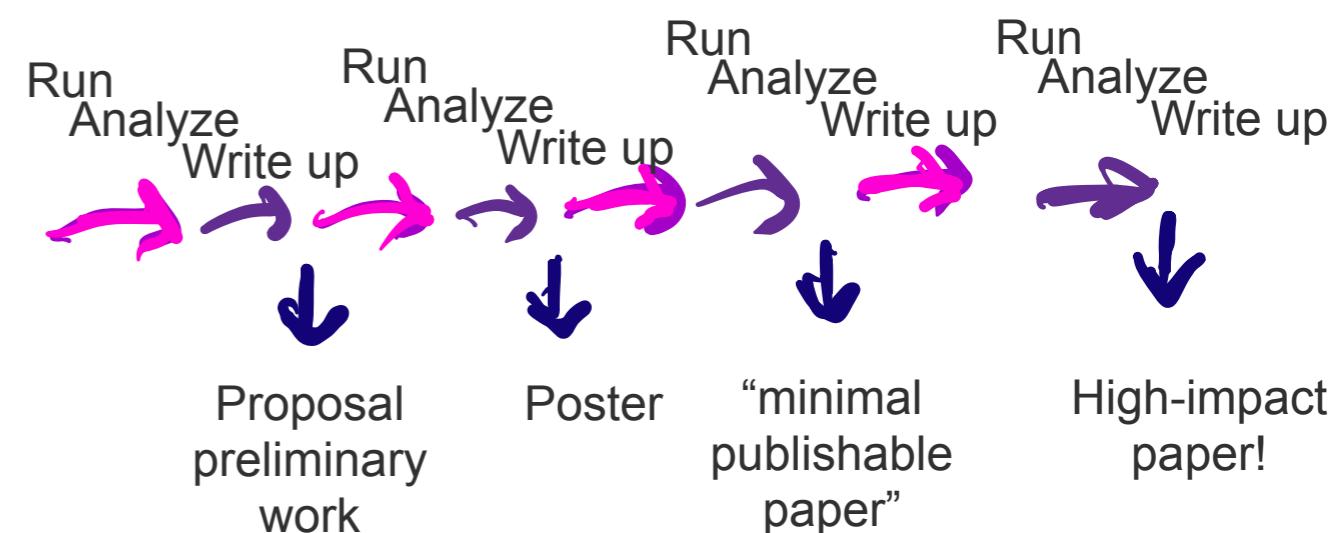
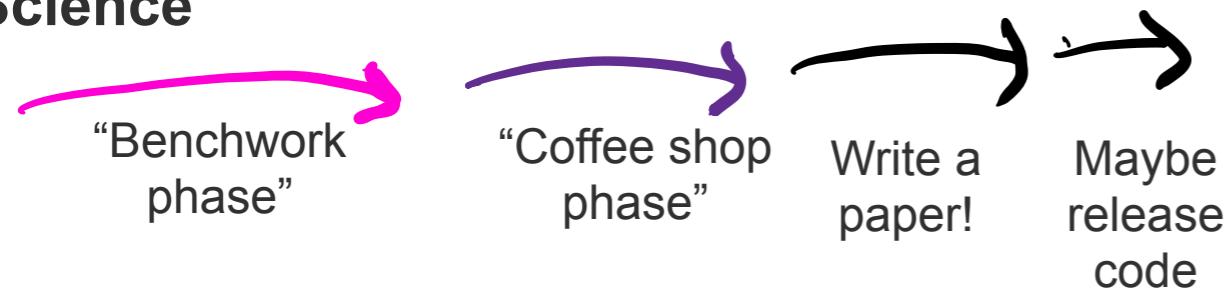


Product development



Henrik Kniberg

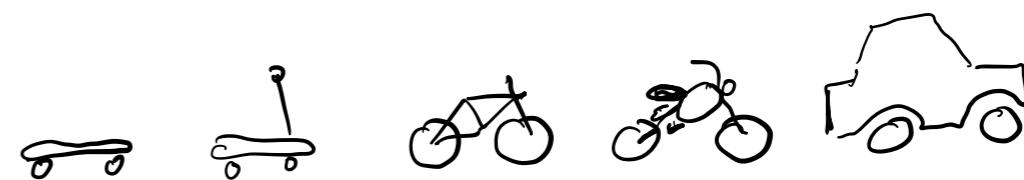
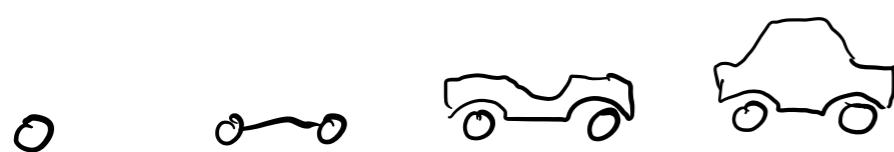
Science



Software engineering

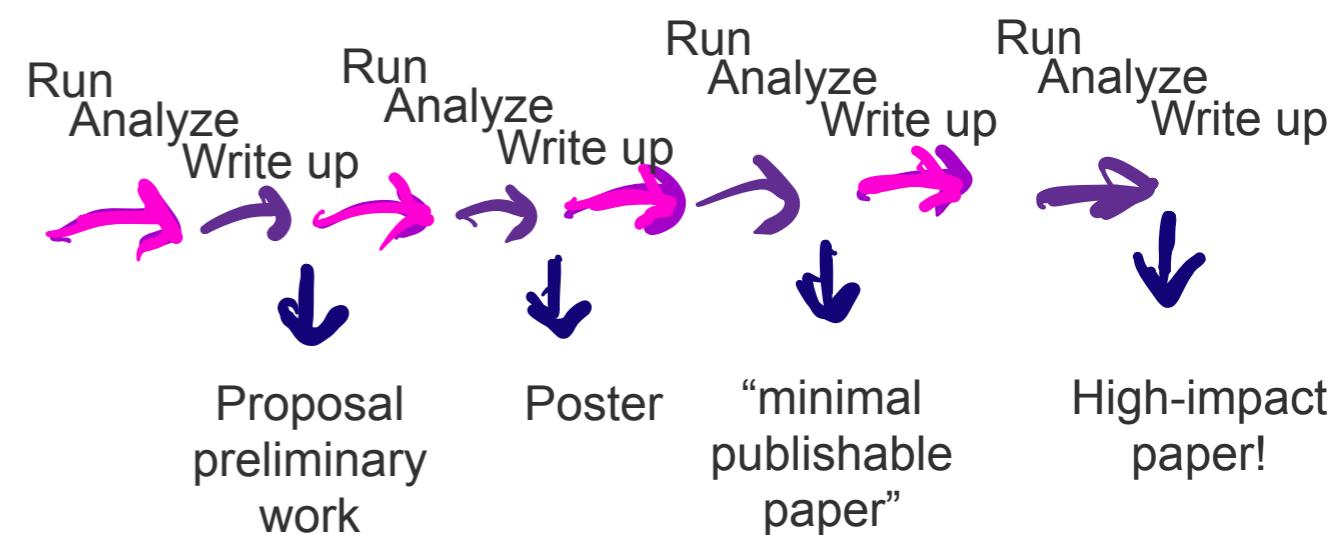


Product development



Henrik Kniberg

Science



The DevOps approach is to use **small batch** and **continuous improvement**.

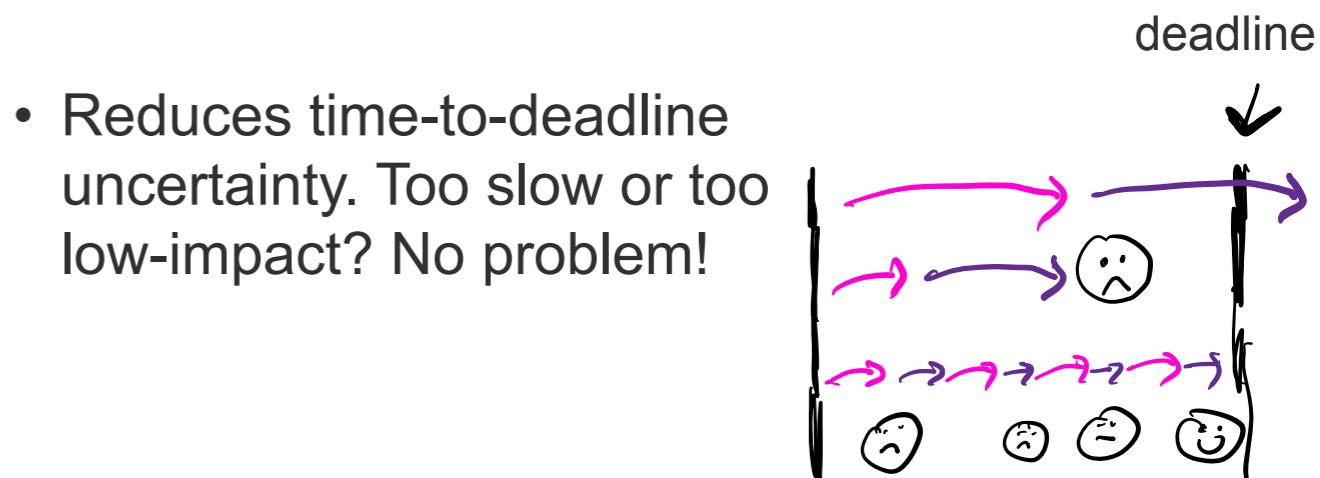
The DevOps approach is **small batch** and **continuous improvement**.

CONS

- Wasteful. You don't use a lot of what you do.
- Missing the advantage of scale. (What would Henry Ford think?!)
- Might be scientifically impossible to “loop back”, e.g., randomized control trial

PROS

- Empirically shown to **reduce worker burnout** (Humble & Kim 2018, Forsgren & Humble, 2016)
- Reduces time-to-deadline uncertainty. Too slow or too low-impact? No problem!



- For complex, high-uncertainty project (i.e., Science!), all the parts adapt to achieve a goal

The DevOps approach is **small batch** and **continuous improvement**.

CONS

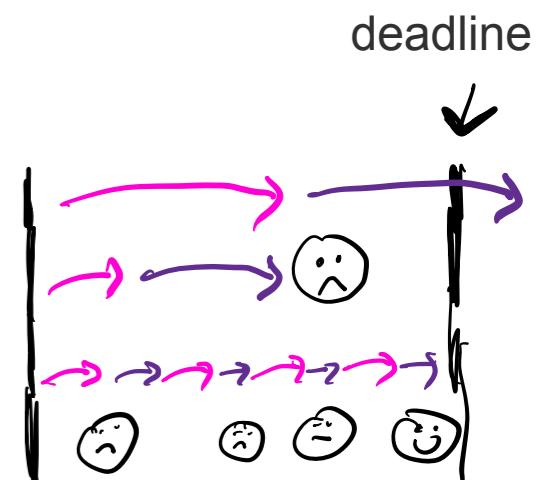
- Wasteful. You don't use a lot of what you do.
- Missing the advantage of scale. (What would Henry Ford think?!)
- Might be scientifically impossible to “loop back”, e.g., randomized control trial
- **A lot of time spend on keeping different versions organized.**

PROS

- Empirically shown to **reduce worker burnout** (Humble & Kim 2018, Forsgren & Humble, 2016)

- Reduces time-to-deadline uncertainty. Too slow or too low-impact? No problem!

- For complex, high-uncertainty project (i.e., Science!), all the parts adapt to achieve a goal



The DevOps approach is **small batch** and **continuous improvement**.

CONS

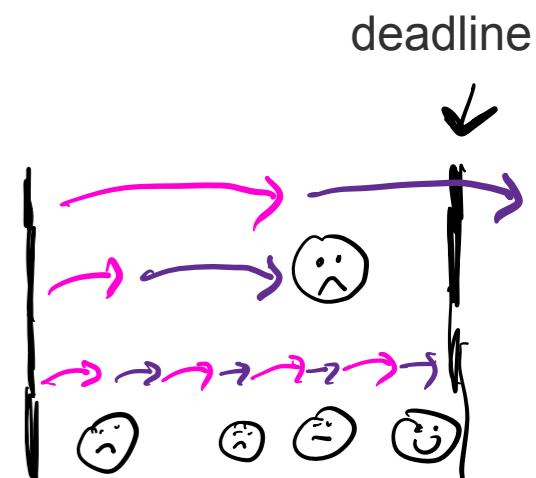
- Wasteful. You don't use a lot of what you do.
- Missing the advantage of scale. (What would Henry Ford think?!)
- Might be scientifically impossible to "loop back", e.g., randomized control trial
- **A lot of time spend on keeping different versions organized.**

PROS

- Empirically shown to **reduce worker burnout** (Humble & Kim 2018, Forsgren & Humble, 2016)

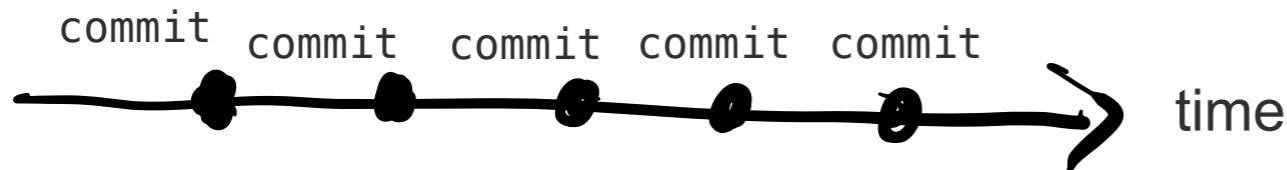
- Reduces time-to-deadline uncertainty. Too slow or too low-impact? No problem!

- For complex, high-uncertainty project (i.e., Science!), all the parts adapt to achieve a goal



→ We're going to need easy, **seamless version control!** →

git



git terminology

repository (repo)
commit hash

[GitHub.com](#)
(company)

git commands

add/stage
commit

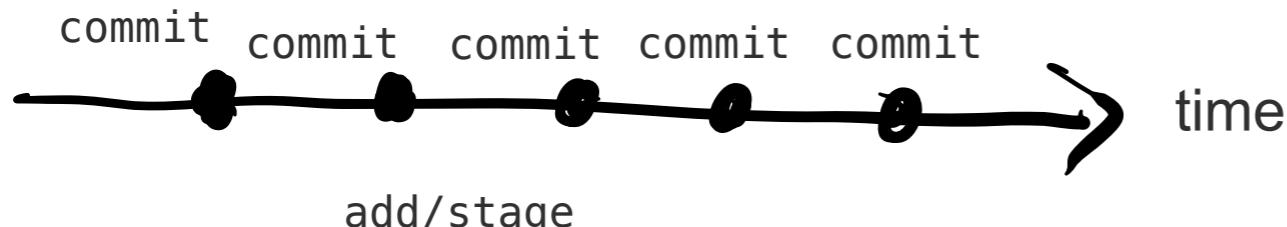
push
pull

clone
fork

pull request

Bisection-search for trouble/bugs

Advantages over automatic history (like Apple Time Machine or Google Docs)



commit
commit
commit
commit
commit
add/stage
commit message – required message summarizing changes
commit hash – unique code for state of the repo

git terminology

repository (repo)
commit hash

GitHub.com
(company)

git commands

add/stage
commit

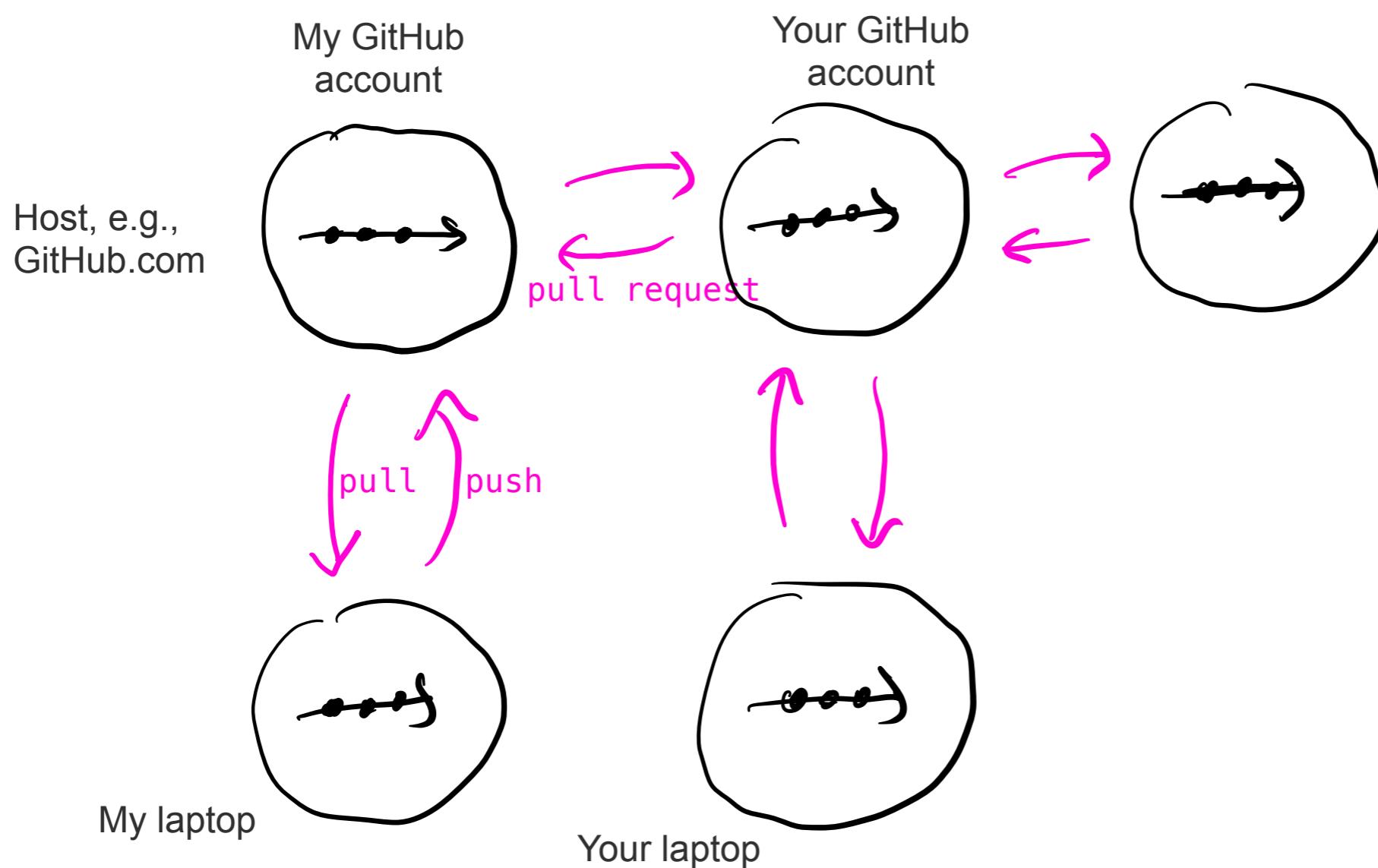
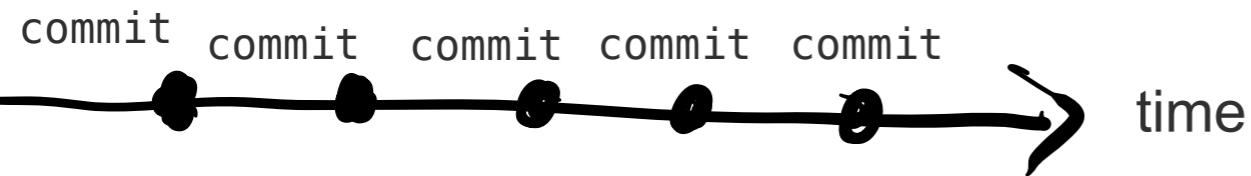
push
pull

clone
fork

pull request

Bisection-search for trouble/bugs

Advantages over automatic history (like Apple Time Machine or Google Docs)

**git terminology**

repository (repo)
commit hash

GitHub.com
(company)

git commands

add/stage
commit

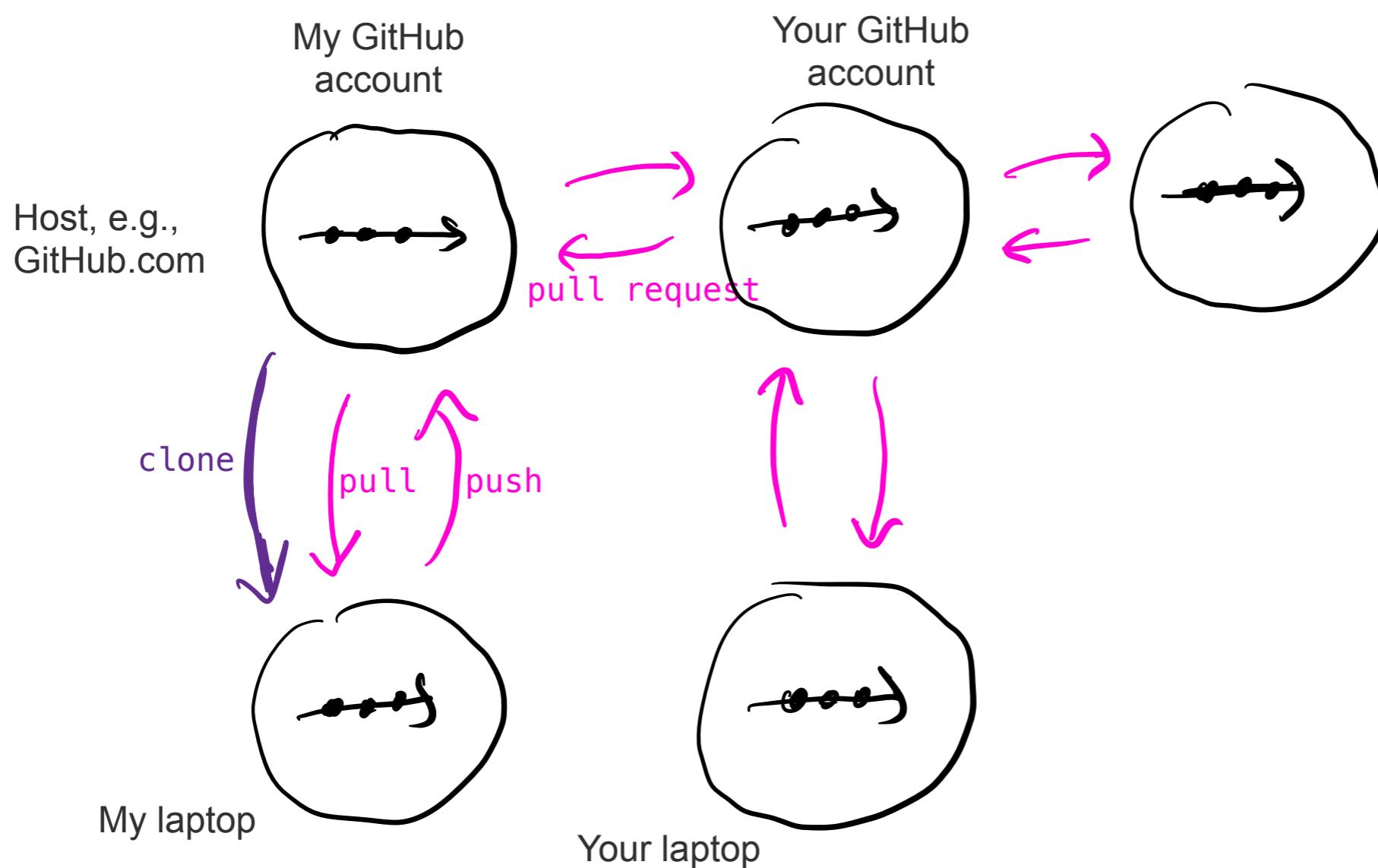
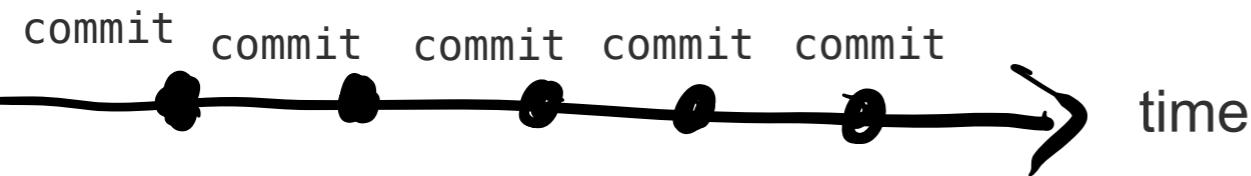
push
pull

clone
fork

pull request

Bisection-search for trouble/bugs

Advantages over automatic history (like Apple Time Machine or Google Docs)

**git terminology**

repository (repo)
commit hash

GitHub.com
(company)

git commands

add/stage
commit

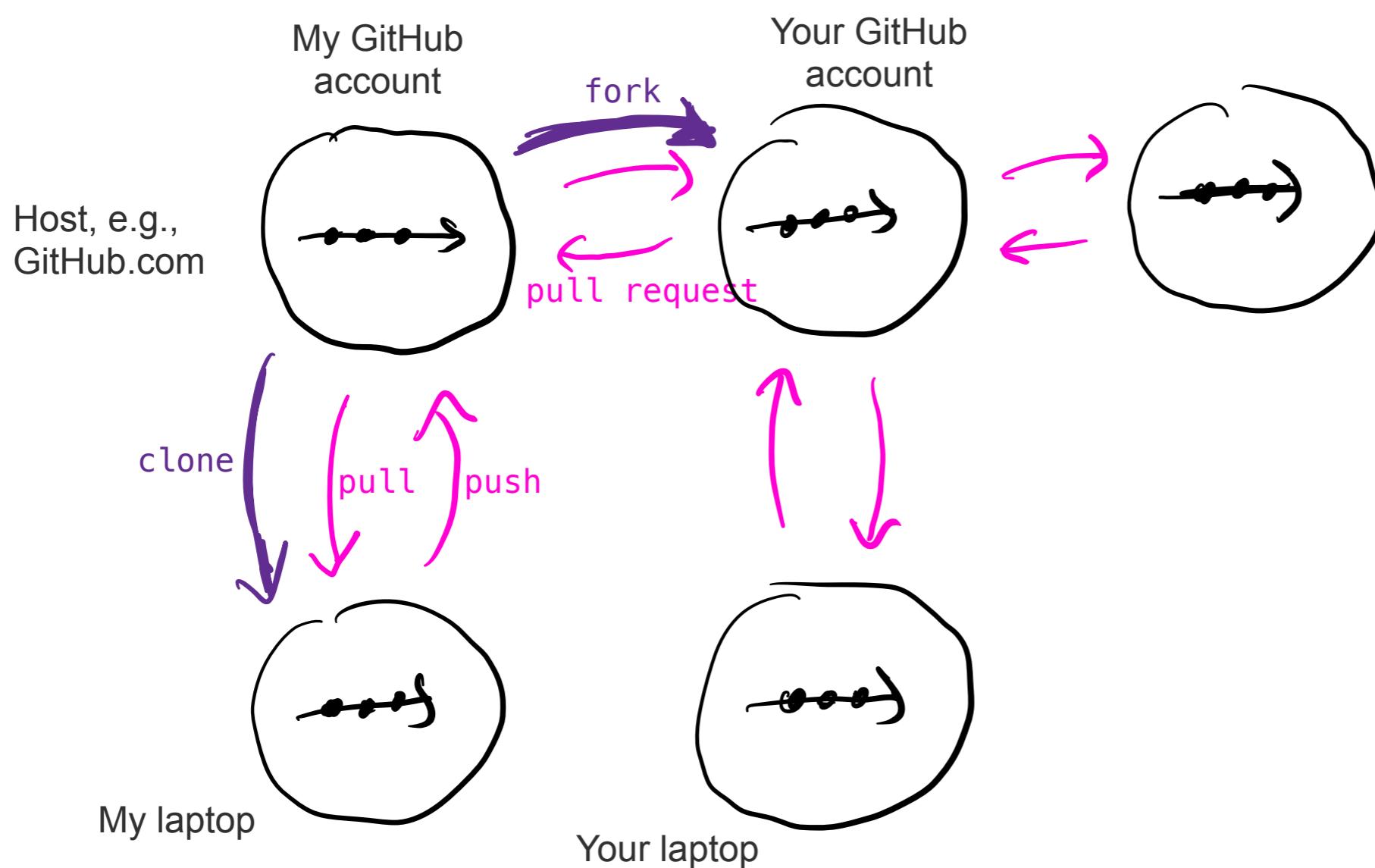
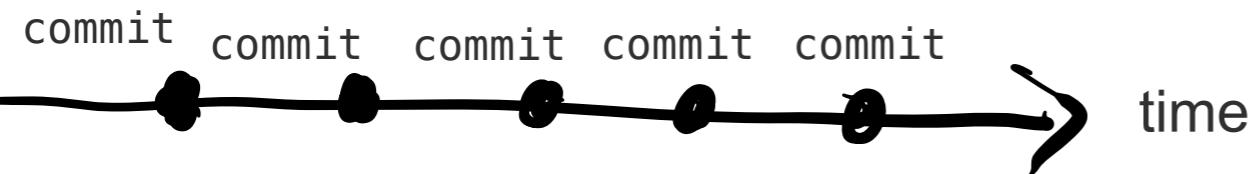
push
pull

clone
fork

pull request

Bisection-search for trouble/bugs

Advantages over automatic history (like Apple Time Machine or Google Docs)

**git terminology**

repository (repo)
commit hash

GitHub.com
(company)

git commands

add/stage
commit

push
pull

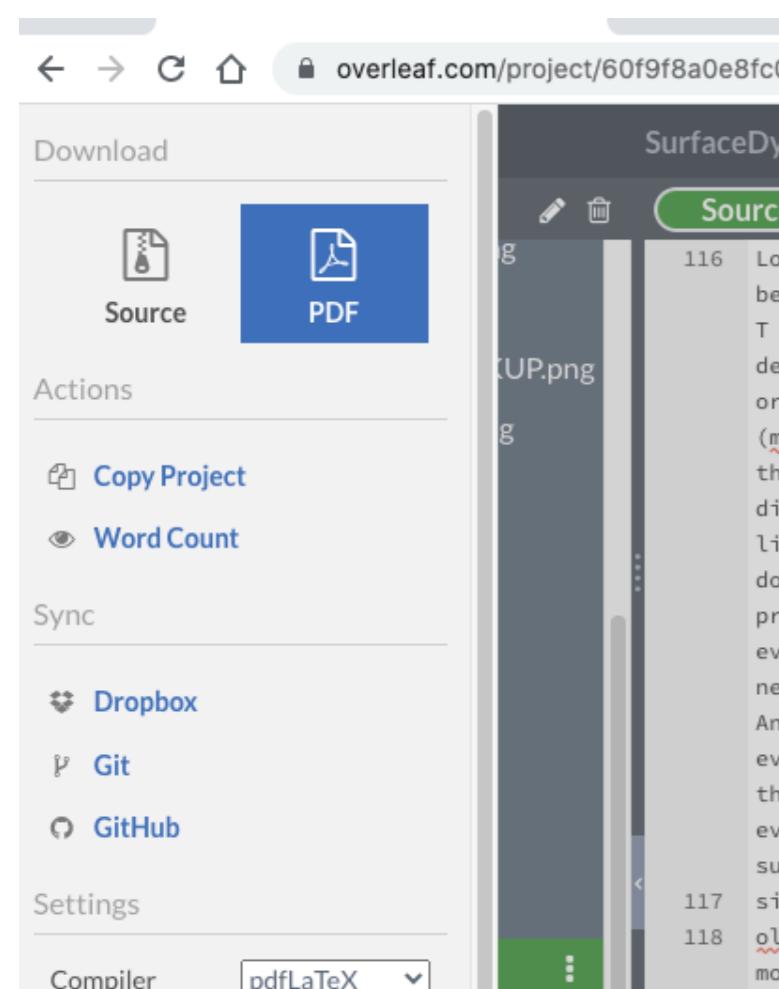
clone
fork

pull request

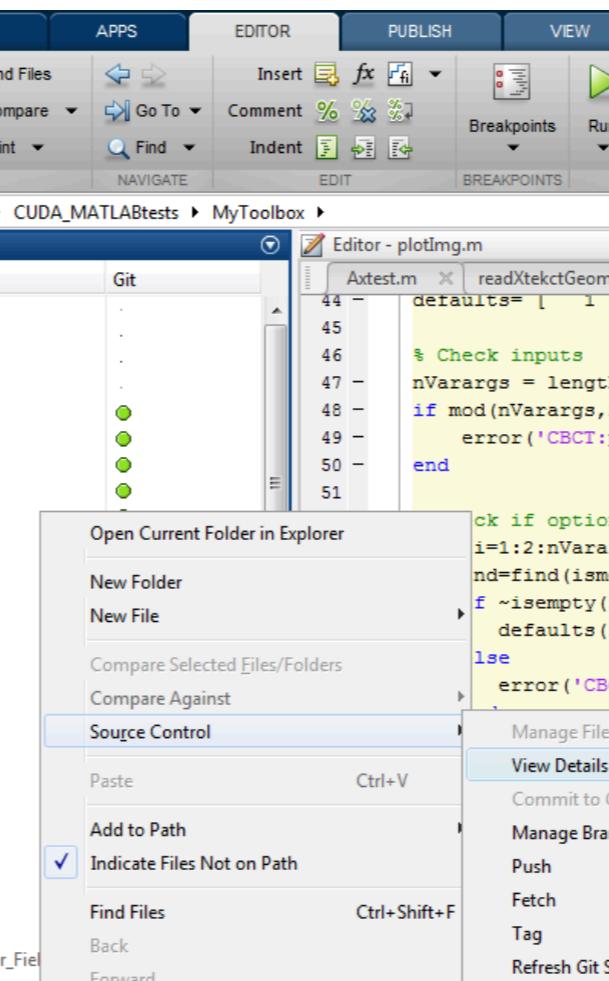
Bisection-search for trouble/bugs

Advantages over automatic history (like Apple Time Machine or Google Docs)

Overleaf (latex)



Matlab



Jupyter

The screenshot shows a Jupyter Notebook interface. At the top, it displays a git commit history: 'allardjun Bootstrap minor edits to text' by '1 contributor'. Below that, it shows a file size of '1.24 MB'. The main area contains a section titled 'Problem Set 9' and an example titled 'Example: predicting colon cancer from st...'. It includes two code snippets:

```
In [ ]: # plot settings
options(repr.plot.width=15, repr.plot.height=10)

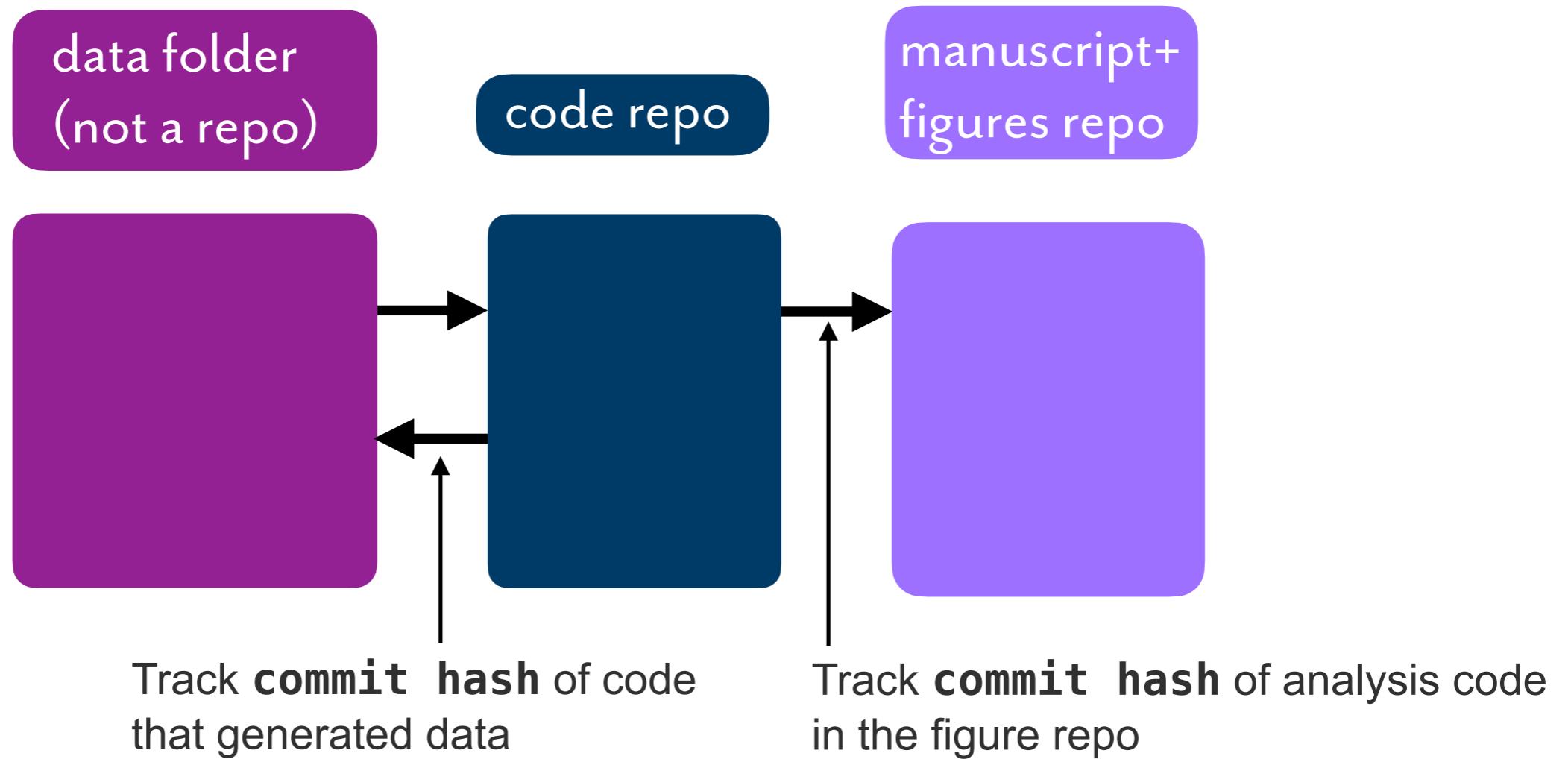
In [24]: # Install a package BioConductor ExperimentHub to access colon cancer dataset
if (!requireNamespace("BiocManager", quietly = TRUE))
  install.packages("BiocManager")
BiocManager::install()
BiocManager::install("ExperimentHub")

# Install glmnet for LASSO and Elastic Net regression
install.packages("glmnet")
```

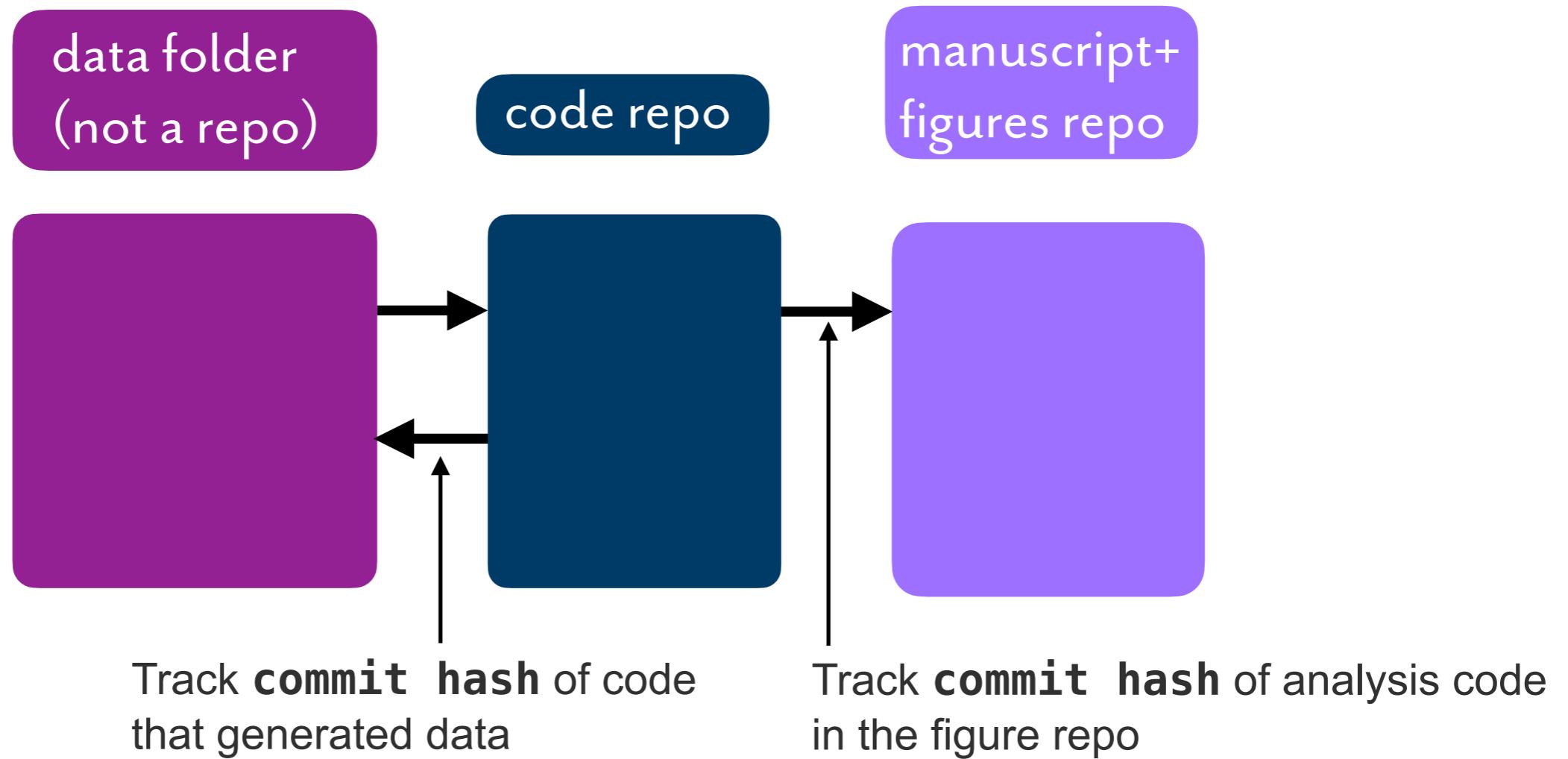
Microsoft Word & Mathematica: works but does not track line-by-line changes

Large files (~>100MB): Needs tricks, or keep in a separate folder.

Typical project setup in Allard Group projects



Typical project setup in Allard Group projects



UPSHOT: Version-control with git can be used smoothly for entire projects (analysis and write-ups, not just code). This allows you to use a continuous-improvement “DevOps” approach — which has pros and cons.

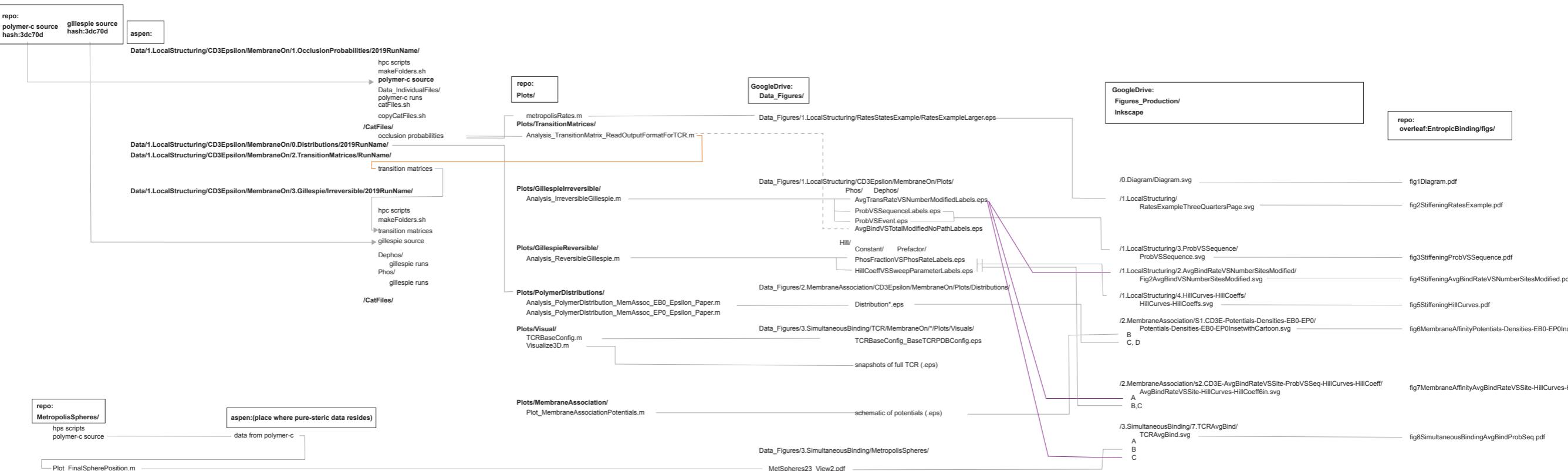
Intrinsic Disorder in the T Cell Receptor Creates Cooperativity and Controls ZAP70 Binding

Lara Clemens,¹ Omer Dushek,² and Jun Allard^{1,3,*}

¹Center for Complex Biological Systems, University of California Irvine, Irvine, California; ²Dunn School of Pathology, University of Oxford, Oxford, United Kingdom; and ³Department of Mathematics and Department of Physics and Astronomy, University of California Irvine, Irvine, California



Biophysical Journal 120, 379–392, January 19, 2021



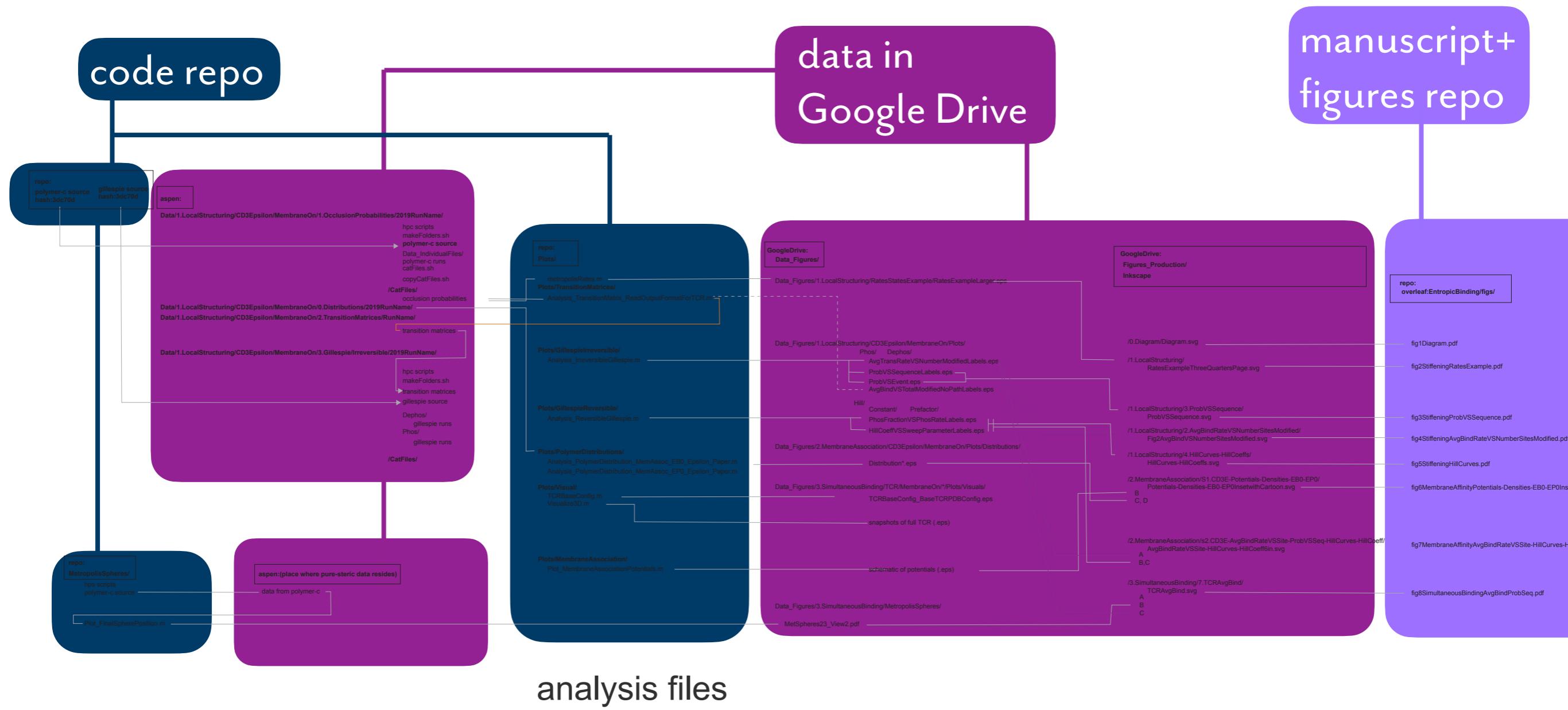
Intrinsic Disorder in the T Cell Receptor Creates Cooperativity and Controls ZAP70 Binding

Lara Clemens,¹ Omer Dushek,² and Jun Allard^{1,3,*}

¹Center for Complex Biological Systems, University of California Irvine, Irvine, California; ²Dunn School of Pathology, University of Oxford, Oxford, United Kingdom; and ³Department of Mathematics and Department of Physics and Astronomy, University of California Irvine, Irvine, California



Biophysical Journal 120, 379–392, January 19, 2021



Much more we didn't discuss!

- branches
- How to go back to previous versions
- How to deal with merge failures

git tutorials

Software Carpentry: <https://swcarpentry.github.io/git-novice/>

UPSHOT: Version-control with git can be used smoothly for entire projects (analysis and write-ups, not just code). This allows you to use a continuous-improvement “DevOps” approach — which has pros and cons.