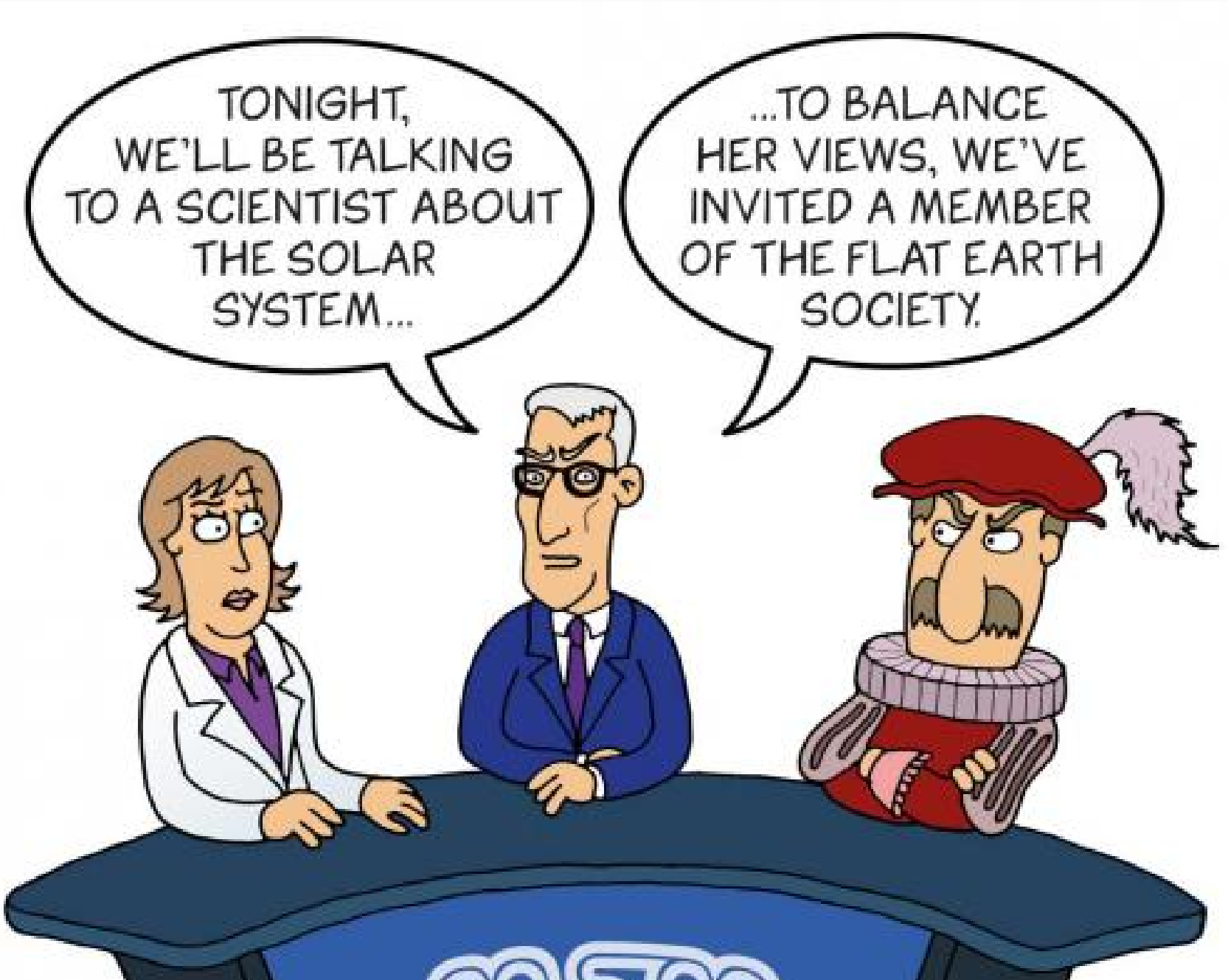
Reproducibility Research and Open Science

Arnaud Legrand

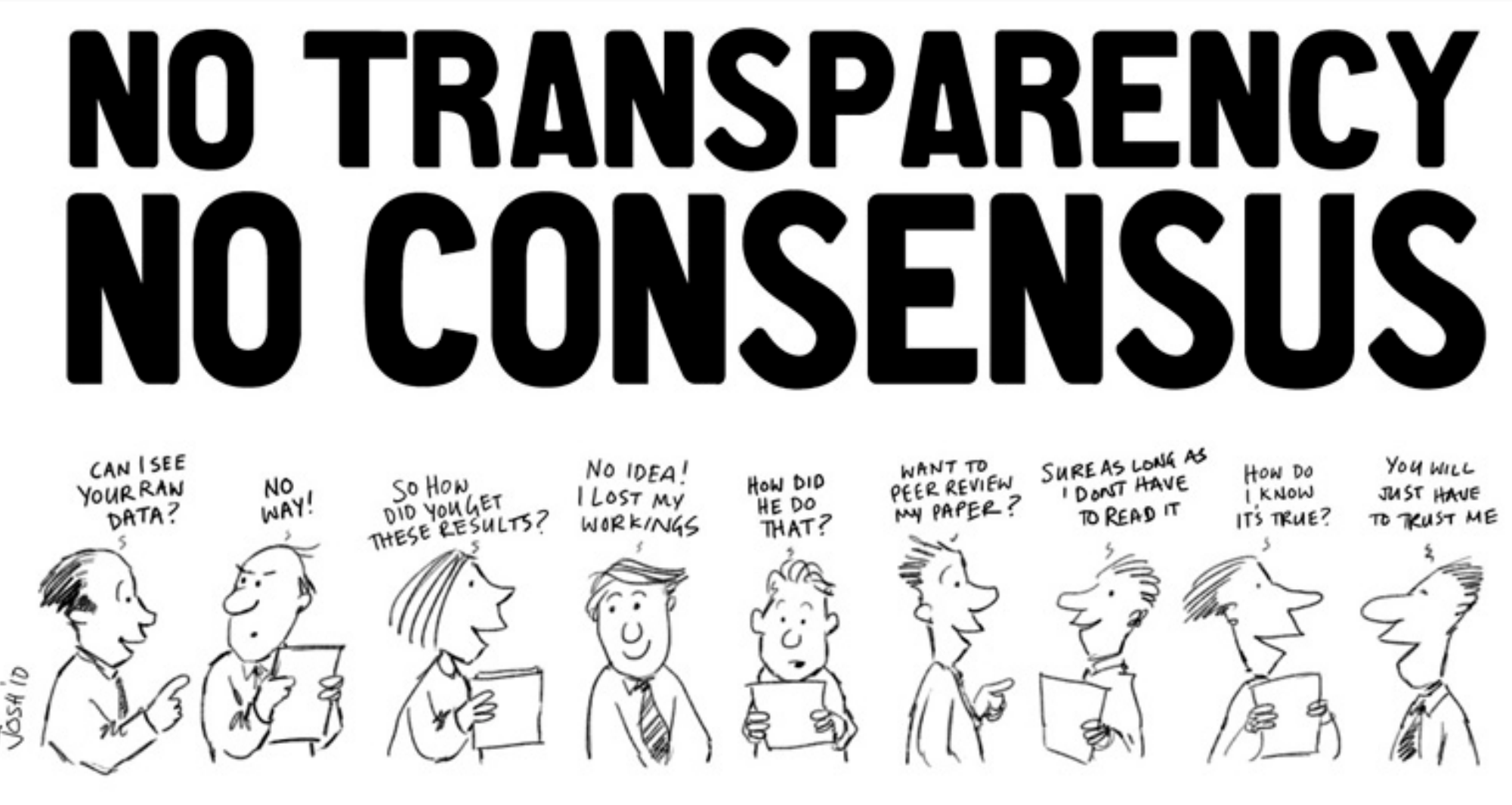
Doing a PhD, good practice and pitfalls to avoid

December 2025

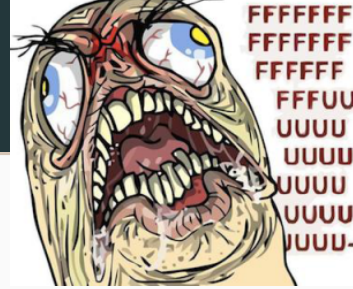
Scientific Consensus



Scientific Consensus



Common Horror Stories 1/4: *What did I do?*



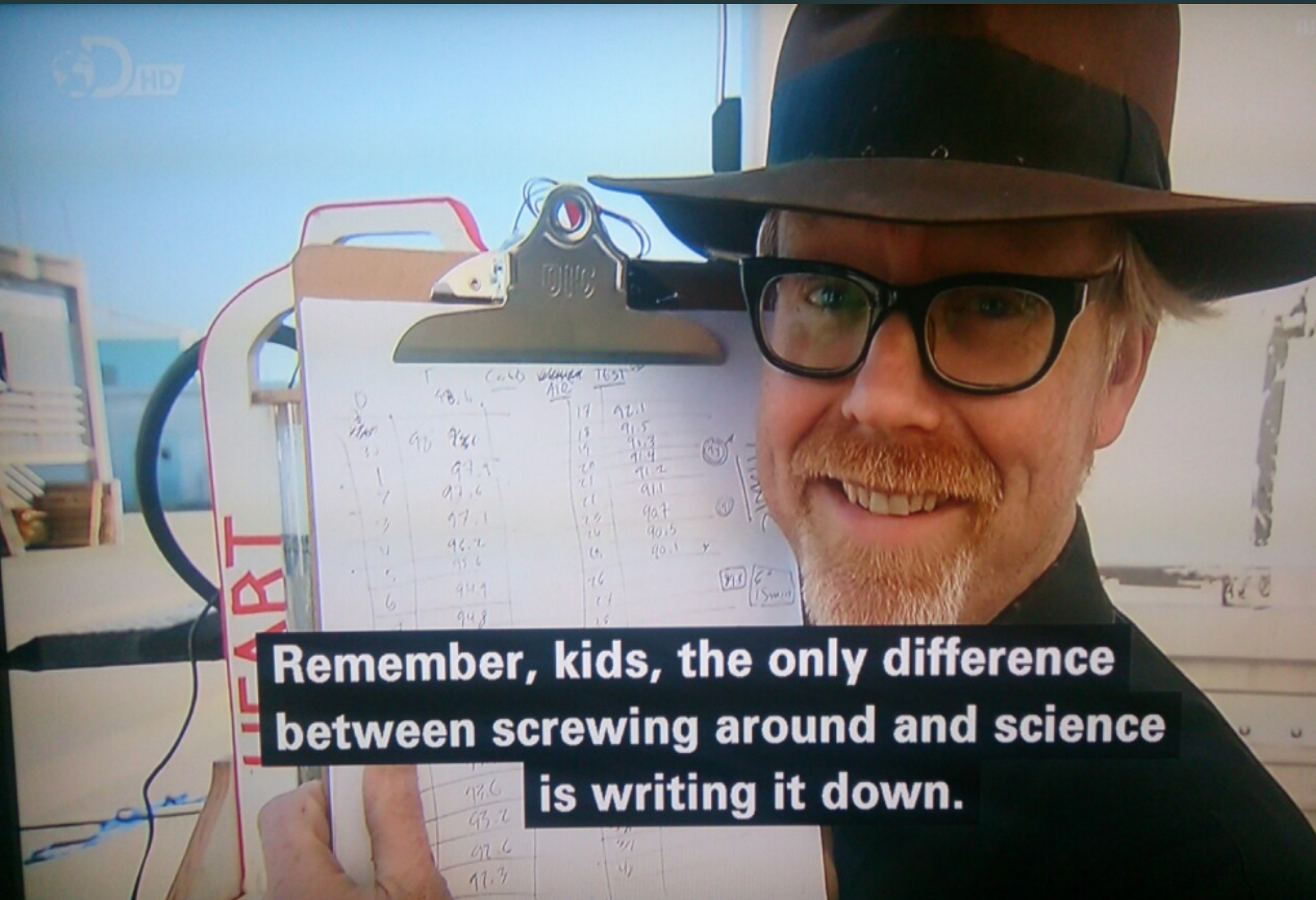
Author

- I thought I used the same parameters but I'm getting different results!
- The new student wants to compare with the method I proposed last year
- My advisor asked me whether I took care of setting this or this but I can't remember
- The damned fourth reviewer asked for a major revision and wants me to change Figure 3. Which code and which data set did I use?
- It worked yesterday! 6 months later: Why did I do that?

Reviewer

- As usual, there is no confidence interval, I wonder about the variability and whether the difference is significant or not
- That can't be true, I'm sure they removed some points
- Why is this graph in logscale? How would it look like otherwise? I'm not even sure of what this value means. If only I could access the generation script

Mythbusters: Science vs. Screwing Around



Computational Documents...

Document initial dans son environnement

jupyter example_pi (autosaved)

File Edit View Insert Cell Kernel Widgets Help Hide Code Python 3

Code CellToolbar

Hide Prompts Hide Code Hide Outputs

```
# Un document computationnel
```

Hide Prompts Hide Code Hide Outputs

```
Mon ordinateur m'indique que $\pi$ vaut *approximativement*
```

In [1]:

Hide Prompts Hide Code Hide Outputs

```
from math import *
print(pi)
```

```
3.141592653589793
```

Hide Prompts Hide Code Hide Outputs

```
Mais calculé avec la __méthode__ des [aiguilles de Buffon]
(https://fr.wikipedia.org/wiki/Aiguille_de_Buffon), on obtiendrait comme
__approximation__ :
```

In [2]:

Hide Prompts Hide Code Hide Outputs

```
import numpy as np
N = 1000000
x = np.random.uniform(size=N, low=0, high=1)
theta = np.random.uniform(size=N, low=0, high=pi/2)
2/(sum((x+np.sin(theta))>1)/N)
```

Out[2]: 3.1437198694098765

Hide Prompts Hide Code Hide Outputs

```
On peut inclure des formules mathématiques comme $\frac{1}{\sigma
\sqrt{2\pi}} \exp \left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$ et des
_dessins qui n'ont rien à voir_ avec $\pi$ (si ce n'est une constante de
normalisation... ☺).
```

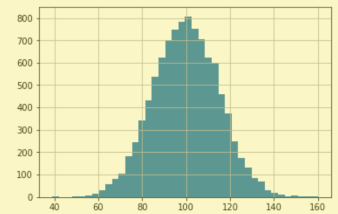
In [3]:

Hide Prompts Hide Code Hide Outputs

```
%matplotlib inline
import matplotlib.pyplot as plt

mu, sigma = 100, 15
x = mu + sigma*np.random.randn(10000)

plt.hist(x,40)
plt.grid(True)
plt.show()
```



Export

Document final

Un document computationnel

Mon ordinateur m'indique que π vaut *approximativement*

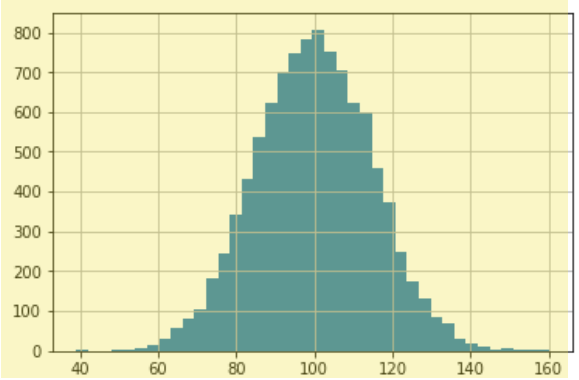
```
3.141592653589793
```

Mais calculé avec la **méthode** des aiguilles de Buffon, on obtiendrait comme **approximation** :

```
import numpy as np
N = 1000000
x = np.random.uniform(size=N, low=0, high=1)
theta = np.random.uniform(size=N, low=0, high=pi/2)
2/(sum((x+np.sin(theta))>1)/N)
```

```
3.1437198694098765
```

On peut inclure des formules mathématiques comme $\frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$ et des *dessins qui n'ont rien à voir* avec π (si ce n'est une constante de normalisation... ☺).



Computational Documents...

Document initial dans son environnement

jupyter example_pi (autosaved)

File Edit View Insert Cell Kernel Widgets Help Hide Code Python 3

Code CellToolbar

Hide Prompts Hide Code Hide Outputs

Un document computationnel

Mon ordinateur m'indique que π vaut *approximativement*

In [1]:

```
from math import *
print(pi)
```

```
3.141592653589793
```

Mais calculé avec la __méthode__ des [aiguilles de Buffon](https://fr.wikipedia.org/wiki/Aiguille_de_Buffon), on obtiendrait comme __approximation__ :

In [2]:

```
import numpy as np
N = 1000000
x = np.random.uniform(size=N, low=0, high=1)
theta = np.random.uniform(size=N, low=0, high=pi/2)
2/(sum((x+np.sin(theta))>1)/N)
```

Out[2]: 3.1437198694098765

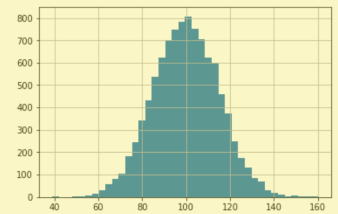
On peut inclure des formules mathématiques comme $\frac{1}{\sigma \sqrt{2\pi}} \exp \left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$ et des *dessins qui n'ont rien à voir* avec π (si ce n'est une constante de normalisation... ☺).

In [3]:

```
%matplotlib inline
import matplotlib.pyplot as plt

mu, sigma = 100, 15
x = mu + sigma*np.random.randn(10000)

plt.hist(x,40)
plt.grid(True)
plt.show()
```



Export

Document final

Un document computationnel

Mon ordinateur m'indique que π vaut *approximativement*

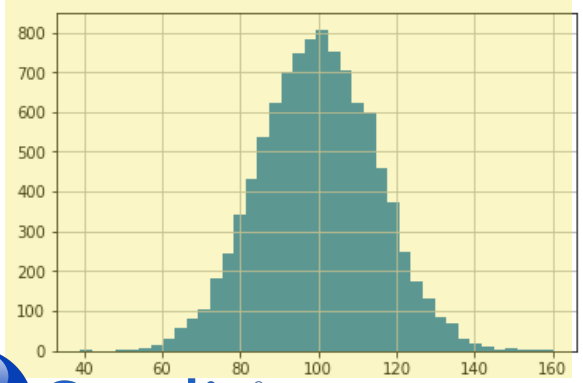
```
3.141592653589793
```

Mais calculé avec la **méthode** des aiguilles de Buffon, on obtiendrait comme **approximation** :

```
import numpy as np
N = 1000000
x = np.random.uniform(size=N, low=0, high=1)
theta = np.random.uniform(size=N, low=0, high=pi/2)
2/(sum((x+np.sin(theta))>1)/N)
```

```
3.1437198694098765
```

On peut inclure des formules mathématiques comme $\frac{1}{\sigma\sqrt{2\pi}}\exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$ et des *dessins qui n'ont rien à voir* avec π (si ce n'est une constante de normalisation... ☺).



... and Laboratory Notebooks

Subject: Fin des pièces jointes dans Mattermost (Inria)

> Mattermost est le principal outil de communication et d'échanges dans certaines équipes. Nous avons un canal par projet, un canal avec chaque doctorant, etc. Ce ne sont pas de simples chats "informels"... ce sont de vraiment outils de travail dans la durée !

*> Comme XXX, nous aussi l'utilisons beaucoup pour échanger des fichiers, des PDF de papiers, faire la *biblio*...*

- *Nous avons positionné Mattermost comme un service de discussion instantanée (ou asynchrone, mais pour des messages à court terme).*
- *Sa finalité ([..]) n'est ni la ~~gestion documentaire~~, ni ~~l'archivage de documents~~, ni le ~~suivi des expérimentations~~ (au sens carnet de laboratoire), même si, finalement, on peut l'utiliser de cette manière.*
- *Si des documents importants pour votre activité au sein d'Inria sont stockés dans Mattermost, et conservés seulement ici, c'est un risque pour Inria (et pour vous): que se passe-t-il en cas de départ des agents? remplacement du service par un autre ? évolution d'une équipe de recherche ? Qui est le propriétaire d'un document partagé ici ?*

Common Horror Stories 2/4: *Argh... damned computers*

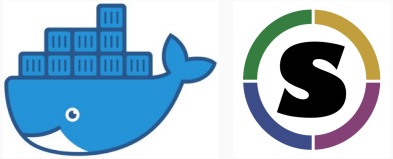
- Hey! Here is my code. It's on GitHub so feel free to play with it! I'm doing open science 🙂
  - **Alice**: I got 3.123123 **Bob**: I got segfault **Cal**: I got 3.123125
- Damned! It used to work!!! Whenever I upgrade my computer, things break so I try to stay away from this ☹
- Whenever trying the code of my colleague, I had to install `Foo` but I broke everything and now neither his code nor mine works! ☹

Seriously ? It's 21st century. 😉 How come all this is so painful ?

Containers and Package Managers

The good

The bad



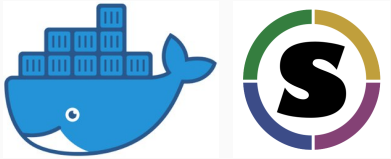
The uggly

Automatic tracking

Containers and Package Managers

The good

The bad



The uggly

Automatic tracking

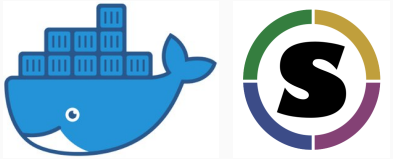
Containers

- Pros: Lightweight, Good isolation, Easy to use
  - Running as easy as `docker run <img> <cmd>`
  - Building images: `docker build -f <Dockerfile>`
  - Sharing through the Docker Hub: `docker pull/push <img>`

Containers and Package Managers

The good

The bad



The uggly

Automatic tracking

Containers

- Pros: Lightweight, Good isolation, Easy to use
- Cons: Opaque, Container build is generally not reproducible
  - Recipes rarely follow *reproducible good practices*

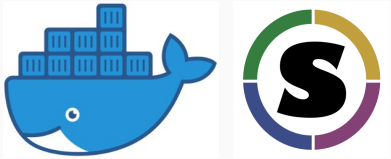
```
1  FROM ubuntu:20.04
2  RUN apt-get update
3       && apt-get upgrade -y
4       && apt-get install -y ...
```

  - Choose a stable image (and the smallest possible)
  - Include only the necessary libraries (e.g. no graphics libs)
  - Avoid system updates (instead freeze sources)

Containers and Package Managers

The good

The bad



The uggly

Automatic tracking

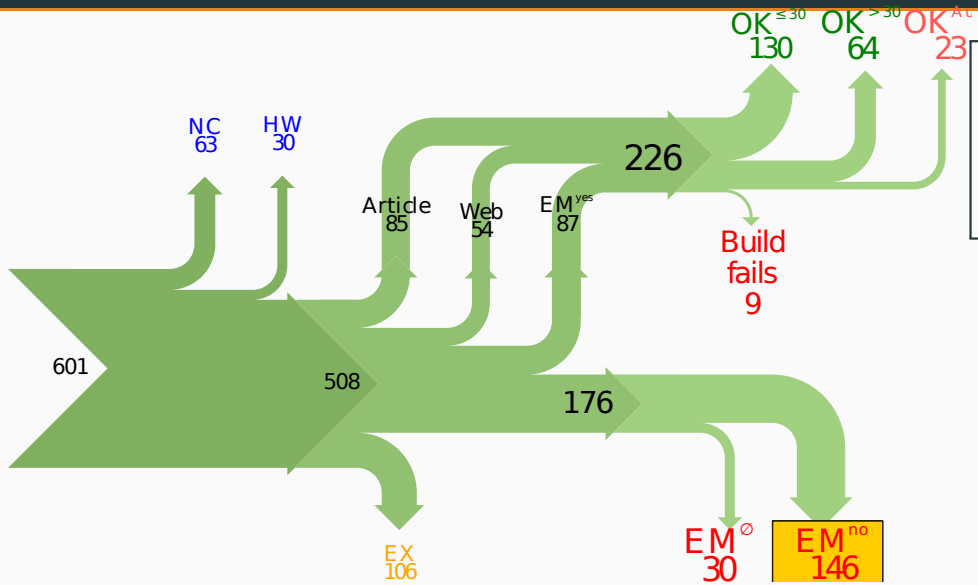
Containers

- Pros: Lightweight, Good isolation, Easy to use
- Cons: Opaque, Container build is generally not reproducible

Package managers (the uggly and the good)

- Language specific: `pip/pipenv/virtualenv`, `conda`, `CRAN/Bioconductor`
  - Limits: version management, durability, permeable, language centric
- GUIX/NiX = Full-fledged functional package manager
  - Native support for environment (*à la git*)
  - Isolation through `--pure`
  - Recompile from source (cache recommended)

Common Horror Stories 3/4: *Please hold on*



- Versionning Problems

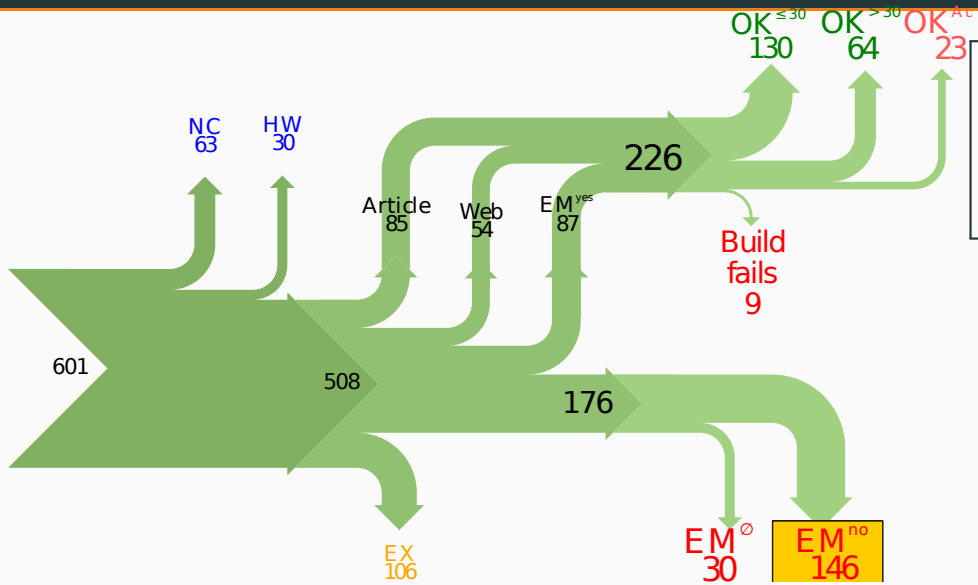
Collberg, Christian *et Al.*, *Measuring Reproducibility in Computer Systems Research*, http://reproducibility.cs.arizona.edu/ 2013

- 8 ACM conferences (ASPLOS'12, CCS'12, OOPSLA'12, OSDI'12, PLDI'12, SIGMOD'12, SOSP'11, VLDB'12) and 5 journals
- EM^{no} = the code cannot be provided

Thanks for your interest in the implementation of our paper. The good news is that I was able to find some code. I am just hoping that it is a stable working version of the code, and matches the implementation we finally used for the paper. Unfortunately, I have lost some data when my laptop was stolen last year. The bad news is that the code is not commented and/or clean.

Attached is the ⟨system⟩ source code of our algorithm. I'm not very sure whether it is the final version of the code used in our paper, but it should be at least 99% close. Hope it will help.

Common Horror Stories 3/4: *Please hold on*



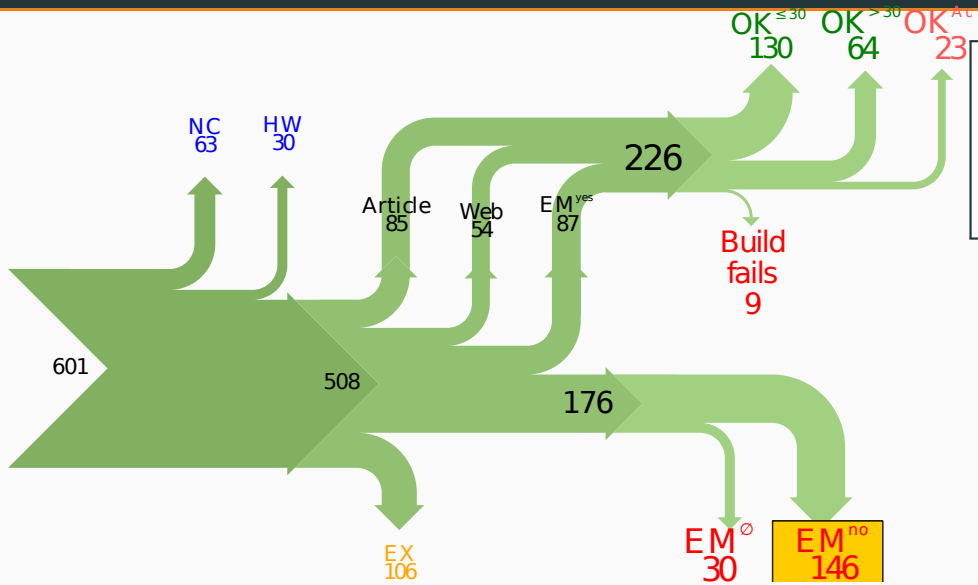
- Versionning Problems
- Bad Backup Practices

Collberg, Christian *et Al.*, *Measuring Reproducibility in Computer Systems Research*, http://reproducibility.cs.arizona.edu/ 2013

- 8 ACM conferences (ASPLOS'12, CCS'12, OOPSLA'12, OSDI'12, PLDI'12, SIGMOD'12, SOSP'11, VLDB'12) and 5 journals
- EM^{no} = the code cannot be provided

Unfortunately, the server in which my implementation was stored had a disk crash in April and three disks crashed simultaneously. While the help desk made significant effort to save the data, my entire implementation for this paper was not found.

Common Horror Stories 3/4: *Please hold on*



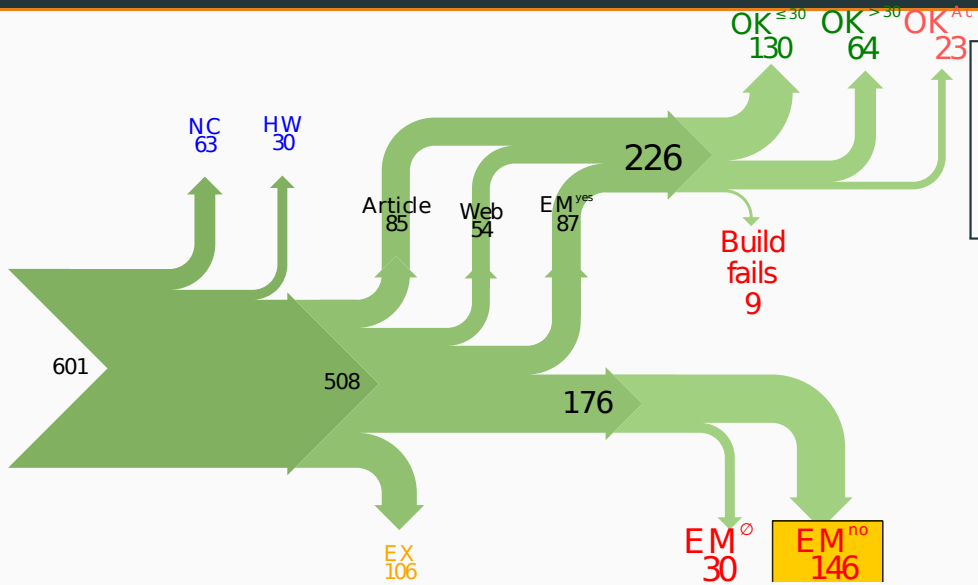
- Versionning Problems
- Bad Backup Practices
- Code Will be Available Soon

Collberg, Christian *et Al.*, *Measuring Reproducibility in Computer Systems Research*, http://reproducibility.cs.arizona.edu/ 2013

- 8 ACM conferences (ASPLOS'12, CCS'12, OOPSLA'12, OSDI'12, PLDI'12, SIGMOD'12, SOSP'11, VLDB'12) and 5 journals
- EM^{no} = the code cannot be provided

Unfortunately the current system is not mature enough at the moment, so it's not yet publicly available. We are actively working on a number of extensions and things are somewhat volatile. However, once things stabilize we plan to release it to outside users. At that point, we would be happy to send you a copy.

Common Horror Stories 3/4: *Please hold on*



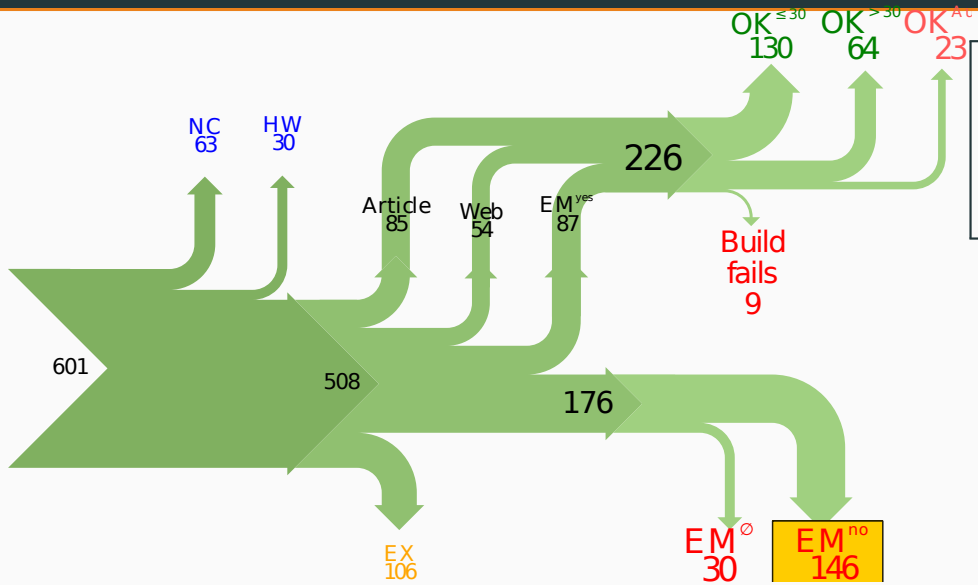
Collberg, Christian *et Al.*, *Measuring Reproducibility in Computer Systems Research*, http://reproducibility.cs.arizona.edu/ 2013

- 8 ACM conferences (ASPLOS'12, CCS'12, OOPSLA'12, OSDI'12, PLDI'12, SIGMOD'12, SOSP'11, VLDB'12) and 5 journals
- EM^{no} = the code cannot be provided

- Versionning Problems
- Bad Backup Practices
- Code Will be Available Soon
- No Intention to Release

I am afraid that the source code was never released. The code was never intended to be released so is not in any shape for general use.

Common Horror Stories 3/4: *Please hold on*



Collberg, Christian *et Al.*, *Measuring Reproducibility in Computer Systems Research*, http://reproducibility.cs.arizona.edu/ 2013

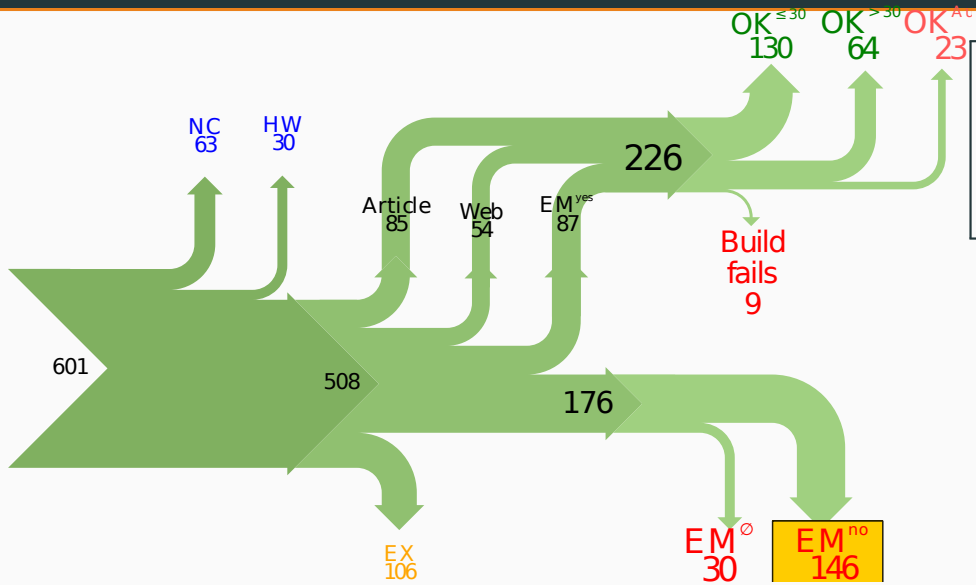
- 8 ACM conferences (ASPLOS'12, CCS'12, OOPSLA'12, OSDI'12, PLDI'12, SIGMOD'12, SOSP'11, VLDB'12) and 5 journals
- EM^{no} = the code cannot be provided
  - Programmer Left

- Versionning Problems
- Bad Backup Practices
- Code Will be Available Soon
- No Intention to Release

⟨STUDENT⟩ was a graduate student in our program but he left a while back so I am responding instead. For the paper we used a prototype that included many moving pieces that only ⟨STUDENT⟩ knew how to operate and we did not have the time to integrate them in a ready-to-share implementation before he left. Still, I hope you can build on the ideas/technique of the paper.

Unfortunately, the author who has done most of the coding for this paper has passed away and the code is no longer maintained.

Common Horror Stories 3/4: *Please hold on*



Collberg, Christian *et Al.*, *Measuring Reproducibility in Computer Systems Research*, http://reproducibility.cs.arizona.edu/ 2013

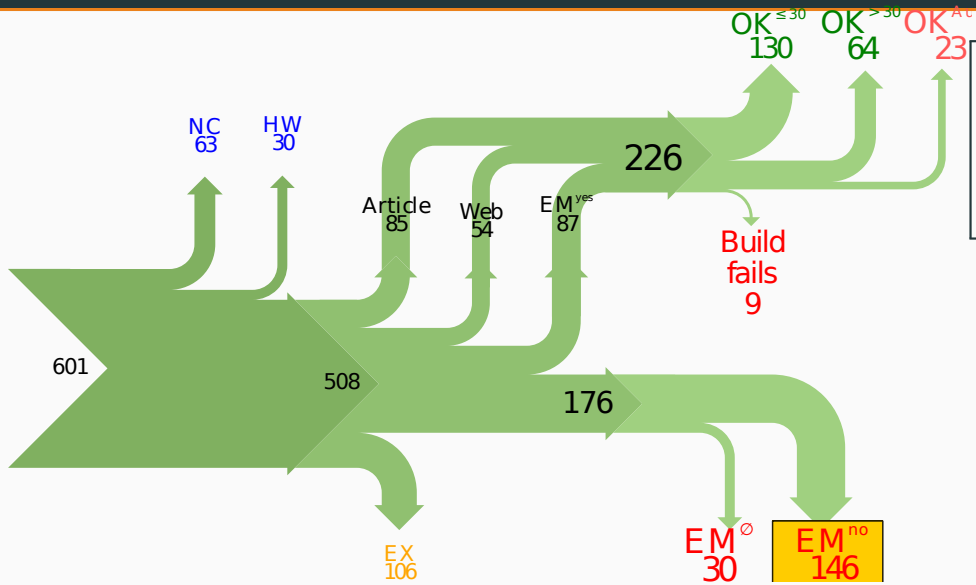
- 8 ACM conferences (ASPLOS'12, CCS'12, OOPSLA'12, OSDI'12, PLDI'12, SIGMOD'12, SOSP'11, VLDB'12) and 5 journals
- EM^{no} = the code cannot be provided
  - Programmer Left
  - Commercial Code

- Versionning Problems
- Bad Backup Practices
- Code Will be Available Soon
- No Intention to Release

Since this work has been done at ⟨COMPANY⟩ we don't open-source code unless there is a compelling business reason to do so. So unfortunately I don't think we'll be able to share it with you.

The code owned by ⟨COMPANY⟩, and AFAIK the code is not open-source. Your best bet is to reimplement :(Sorry.

Common Horror Stories 3/4: *Please hold on*



Collberg, Christian *et Al.*, *Measuring Reproducibility in Computer Systems Research*, http://reproducibility.cs.arizona.edu/ 2013

- 8 ACM conferences (ASPLOS'12, CCS'12, OOPSLA'12, OSDI'12, PLDI'12, SIGMOD'12, SOSP'11, VLDB'12) and 5 journals
- EM^{no} = the code cannot be provided
  - Programmer Left
  - Commercial Code
  - Proprietary Academic Code

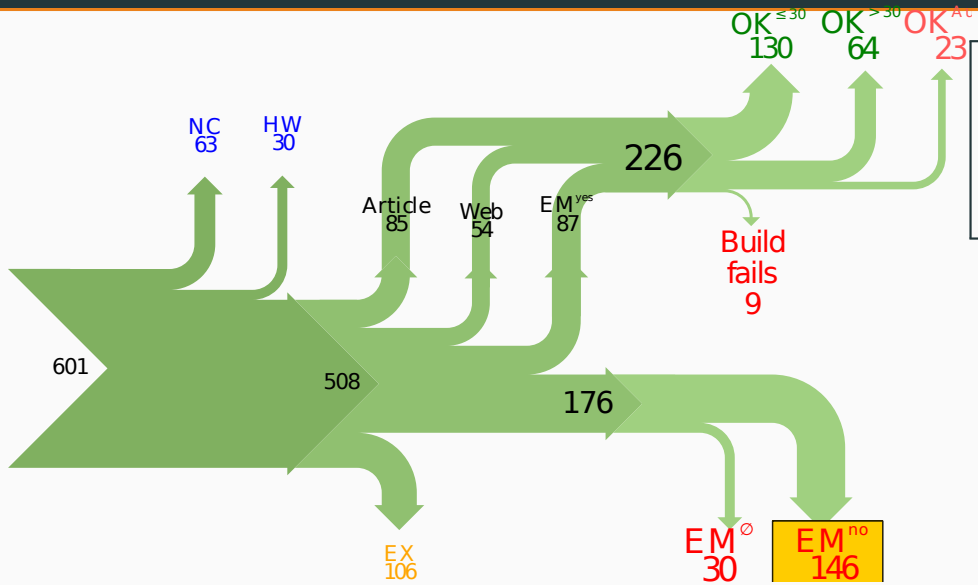
- Versionning Problems
- Bad Backup Practices
- Code Will be Available Soon
- No Intention to Release

Unfortunately, the ⟨SYSTEM⟩ sources are not meant to be opensource (the code is partially property of ⟨UNIVERSITY 1⟩, ⟨UNIVERSITY 2⟩ and ⟨UNIVERSITY 3⟩.)

If this will change I will let you know, albeit I do not think there is an intention to make the ⟨SYSTEM⟩ sources opensource in the near future.

If you're interested in obtaining the code, we only ask for a description of the research project that the code will be used in (which may lead to some joint research), and we also have a software license agreement that the University would need to sign.

Common Horror Stories 3/4: *Please hold on*



Collberg, Christian *et Al.*, *Measuring Reproducibility in Computer Systems Research*, http://reproducibility.cs.arizona.edu/ 2013

- 8 ACM conferences (ASPLOS'12, CCS'12, OOPSLA'12, OSDI'12, PLDI'12, SIGMOD'12, SOSP'11, VLDB'12) and 5 journals
- EM^{no} = the code cannot be provided
  - Versionning Problems
  - Bad Backup Practices
  - Code Will be Available Soon
  - No Intention to Release
  - Programmer Left
  - Commercial Code
  - Proprietary Academic Code
  - Research vs. Sharing

In the past when we attempted to share it, we found ourselves spending more time getting outsiders up to speed than on our own research. So I finally had to establish the policy that we will not provide the source code outside the group.

Changing Research Practices

Soft. Engineering, Statistics, and Reproducible Research in the curricula

Manifesto: *"I solemnly pledge"* (WSSSPE, Lorena Barba, FAIR)

1. I will teach my graduate students about reproducibility
2. All our research code (and writing) is under version control
3. We will always carry out verification and validation
4. We will share data, plotting script & figure under CC-BY
5. We will upload the preprint to arXiv at the time of submission of a paper
6. We will release code at the time of submission of a paper
7. We will add a "Reproducibility" declaration at the end of each paper
8. I will keep an up-to-date web presence

Learn and Teach using online resources like

- Software Carpentry, The Turing Way, ...

Changing Publishing Practices

Artifact evaluation and ACM badges

Major conferences

- Supercomputing: Artifact Description (AD) mandatory, Artifact Evaluation (AE) still optional, Double blind vs. RR
- NeurIPS, ICLR: open reviews, reproducibility challenge

Joelle Pineau @ NeurIPS'18

- ACM SIGMOD 2015-2019, Most Reproducible Paper Award...

Mentalitie are evolving people care, make stuff available, errors are found and fixed

Horror Stories 4/4: Fighting Information Loss with Archives

 or  = awesome collaborations ($\neq$ archive)

- D. Spinellis. *The Decay and Failures of URL References*. CACM, 46(1), 2003
  The half-life of a referenced URL is approximately 4 years
- P. Habibzadeh. *Decay of References to Web sites in Articles Published in General Medical Journals: Mainstream vs Small Journals*. Applied Clinical Informatics. 4 (4), 2013
  half life ranged from 2.2 years in EMHJ to 5.3 years in BMJ
- Discontinuated forges: Code Space, Gitorious, Google code, Inria Gforge

Horror Stories 4/4: Fighting Information Loss with Archives

or

= awesome collaborations (≠ archive)

- D. Spinellis. *The Decay and Failures of URL References*. CACM, 46(1), 2003
  The half-life of a referenced URL is approximately 4 years
- P. Habibzadeh. *Decay of References to Web sites in Articles Published in General Medical Journals: Mainstream vs Small Journals*. Applied Clinical Informatics. 4 (4), 2013
  half life ranged from 2.2 years in EMHJ to 5.3 years in BMJ
- Discontinuated forges: Code Space, Gitorious, Google code, Inria Gforge

Article archives arXiv.org HAL archives-ouvertes.fr

Data archives figshare zenodo

INTERNET ARCHIVE

Software Archive Software Heritage Collect/Preserve/Share

Plan for disaster with `git` and `git-annex` (not `git LFS`!)

Separation between articles, code, and data is not so simple though

Different Reproducibility Concerns in Modern Science

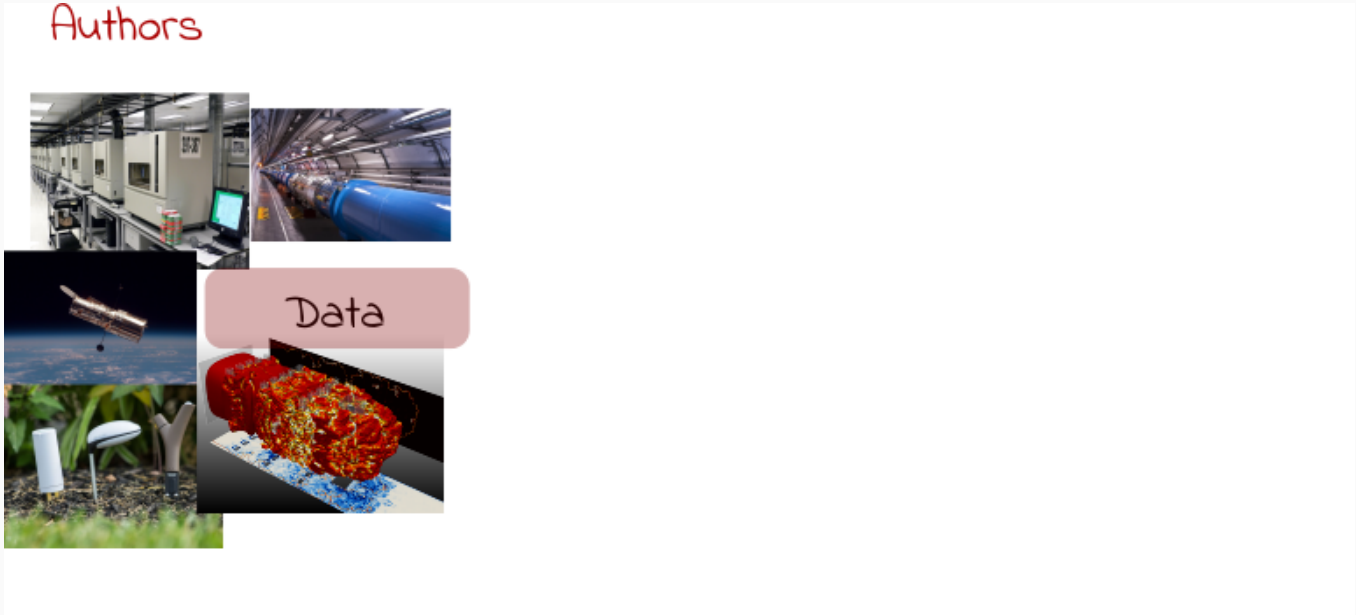
Social Sciences, Oncology, ... methodology, statistics, pre-registration

Genomics software engineering, computational reproducibility, provenance

Computational fluid dynamics numerical issues

Artificial Intelligence most of the above

The processing steps between raw observations and findings have gotten increasingly numerous and complex



Different Reproducibility Concerns in Modern Science

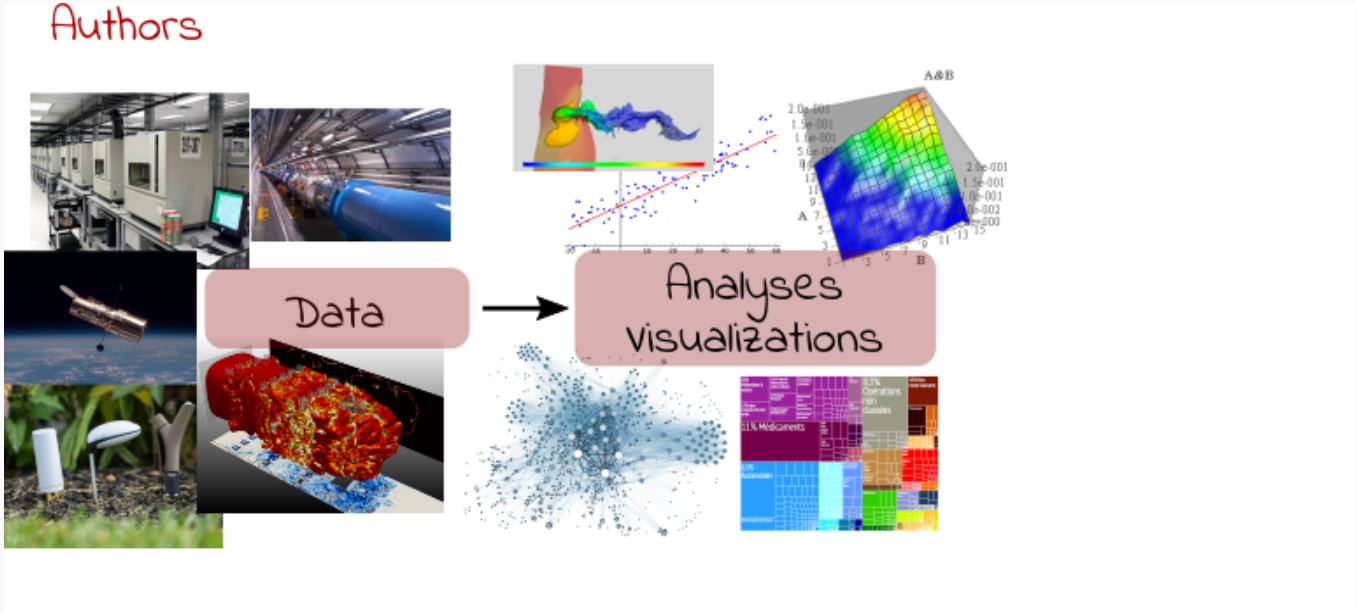
Social Sciences, Oncology, ... methodology, statistics, pre-registration

Genomics software engineering, computational reproducibility, provenance

Computational fluid dynamics numerical issues

Artificial Intelligence most of the above

The processing steps between raw observations and findings have gotten increasingly numerous and complex



Different Reproducibility Concerns in Modern Science

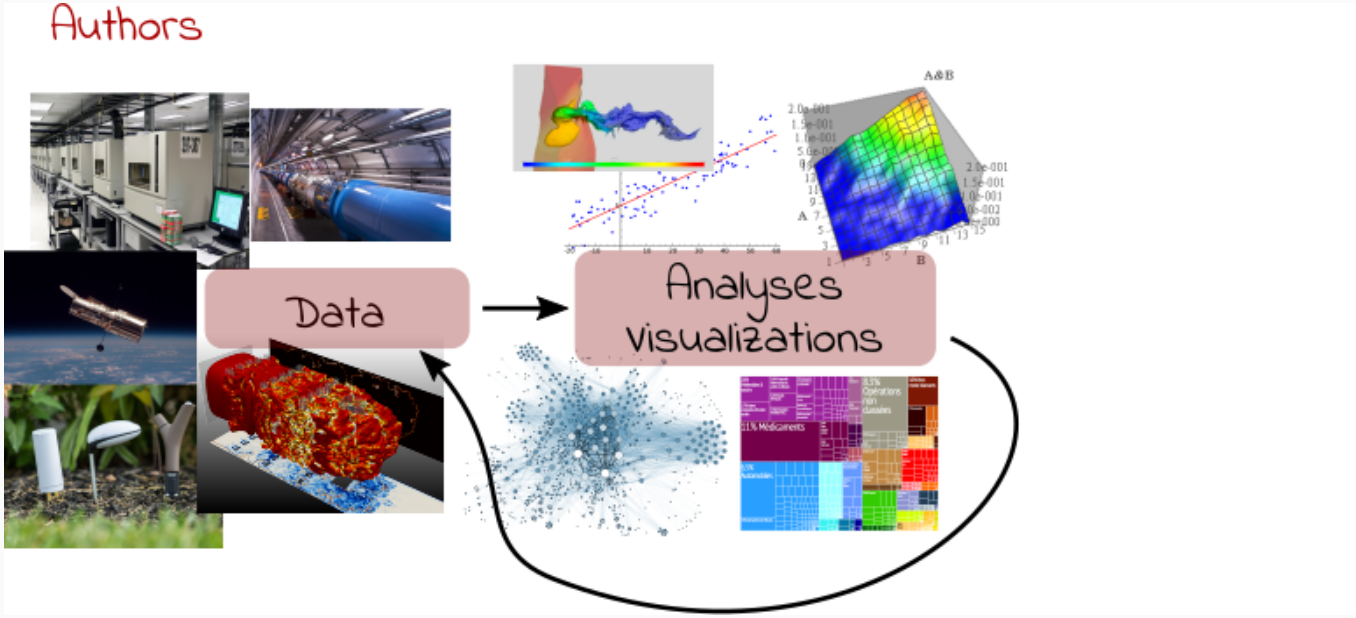
Social Sciences, Oncology, ... methodology, statistics, pre-registration

Genomics software engineering, computational reproducibility, provenance

Computational fluid dynamics numerical issues

Artificial Intelligence most of the above

The processing steps between raw observations and findings have gotten increasingly numerous and complex



Different Reproducibility Concerns in Modern Science

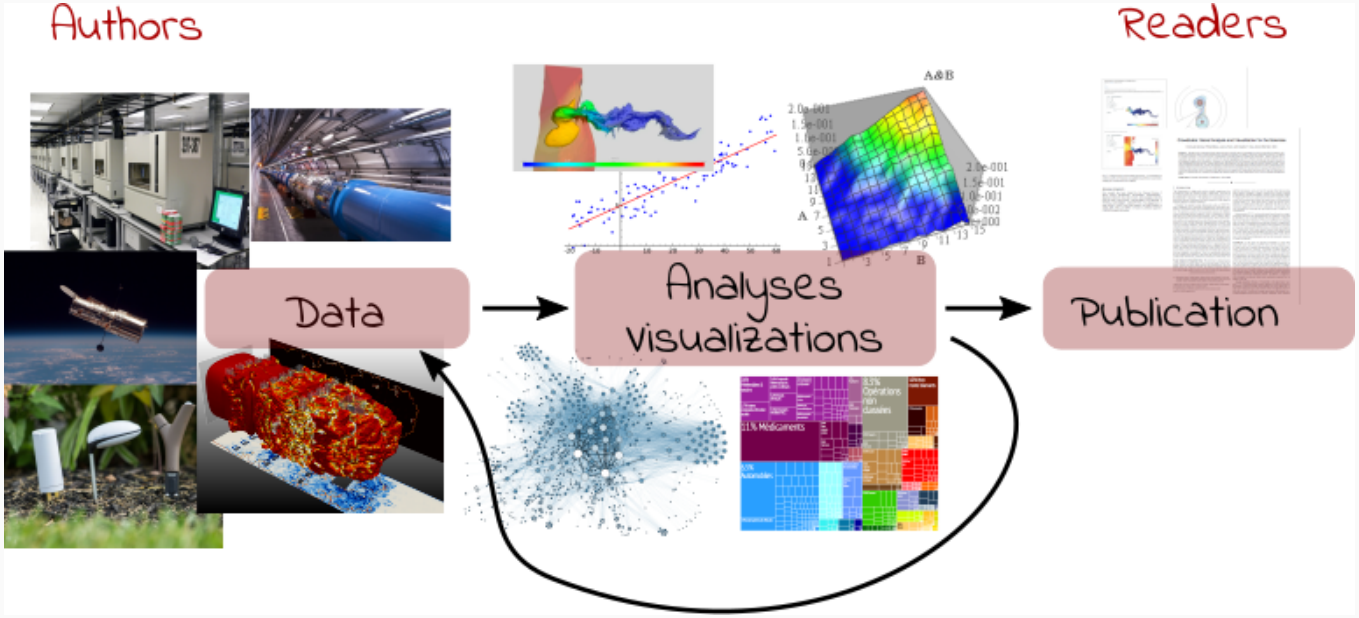
Social Sciences, Oncology, ... methodology, statistics, pre-registration

Genomics software engineering, computational reproducibility, provenance

Computational fluid dynamics numerical issues

Artificial Intelligence most of the above

The processing steps between raw observations and findings have gotten increasingly numerous and complex



Different Reproducibility Concerns in Modern Science

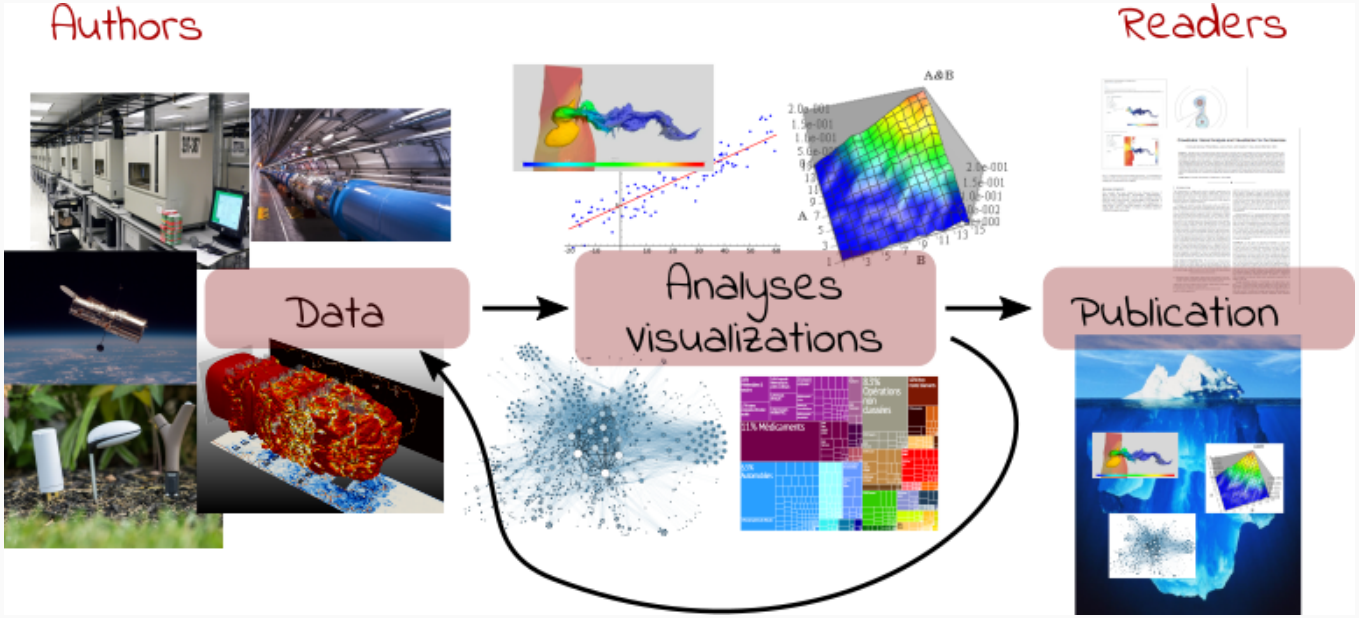
Social Sciences, Oncology, ... methodology, statistics, pre-registration

Genomics software engineering, computational reproducibility, provenance

Computational fluid dynamics numerical issues

Artificial Intelligence most of the above

The processing steps between raw observations and findings have gotten increasingly numerous and complex



Different Reproducibility Concerns in Modern Science

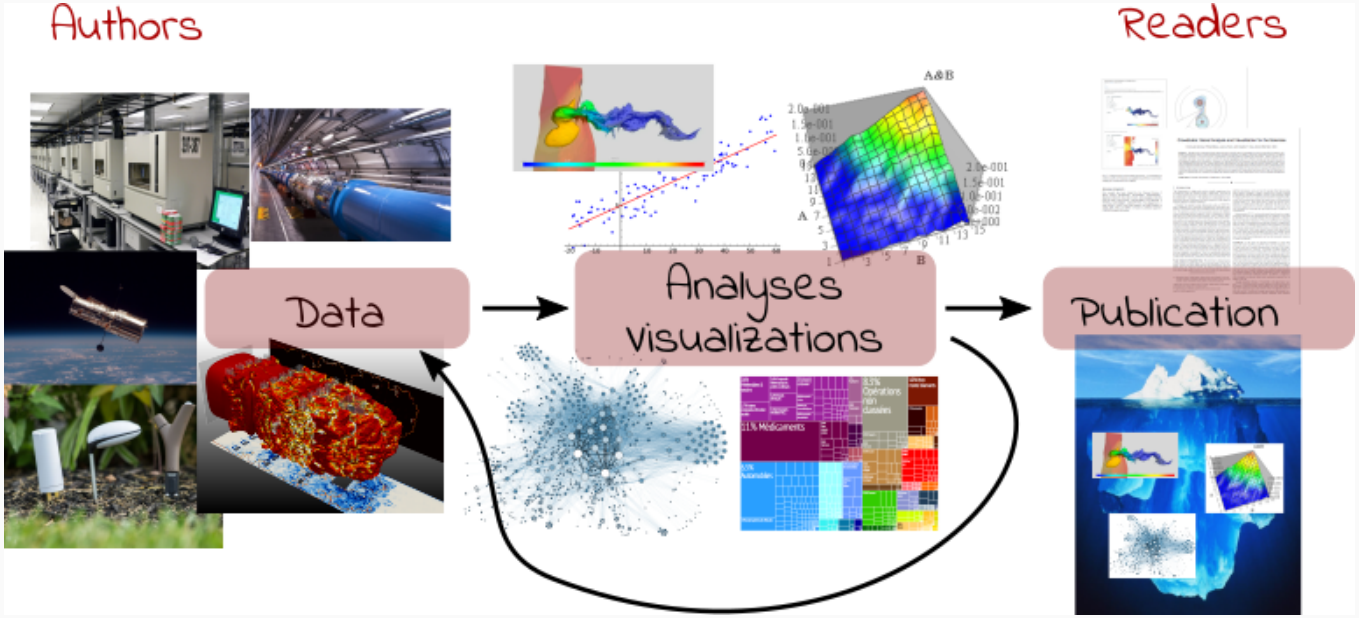
Social Sciences, Oncology, ... methodology, statistics, pre-registration

Genomics software engineering, computational reproducibility, provenance

Computational fluid dynamics numerical issues

Artificial Intelligence most of the above

The processing steps between raw observations and findings have gotten increasingly numerous and complex



Reproducible Research = Bridging the Gap by working Transparently

Reproducible Research ∼ Open Science ?

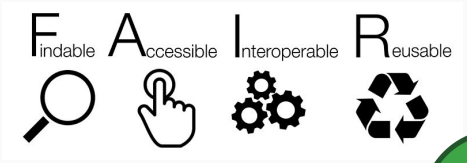
Plan National pour la Science Ouverte (BSN ⇝ CoSO)

- CNRS, Inria, INRAE, ...
- Many flavors: *Citizen Science*

Main pillars:

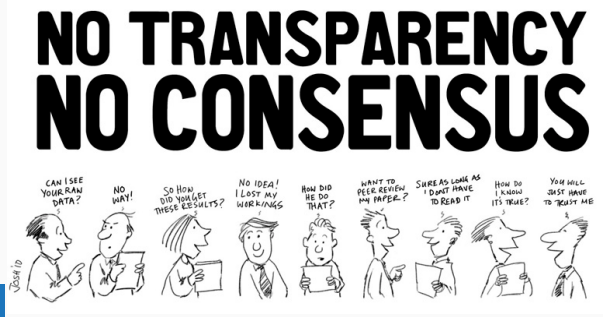
1. Open access

2. Open data



3. Open source
   - *Open hardware*

4. **Open methodology** (Reproducible Research)
   - *Open-notebook science*
   - *Open science infrastructures*
5. Open peer review (avoid collusion)
6. Open educational resources



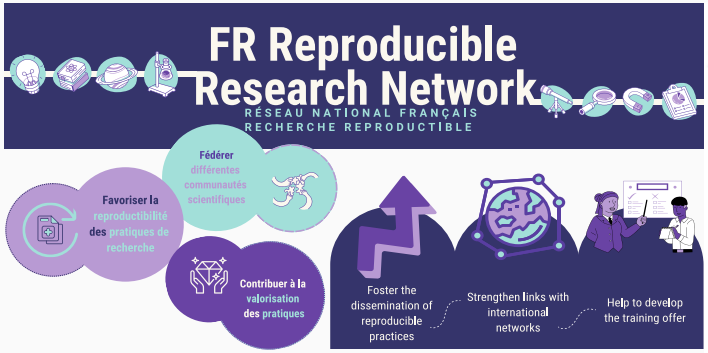
Obviously making code/data available for the reproduction of results from published papers has become the new norm

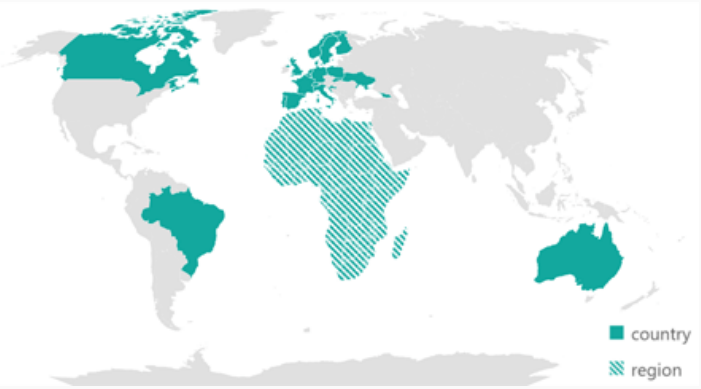
Resources and Acknowledgments

A non-technical introduction to reproducibility issues

Resources and Acknowledgments

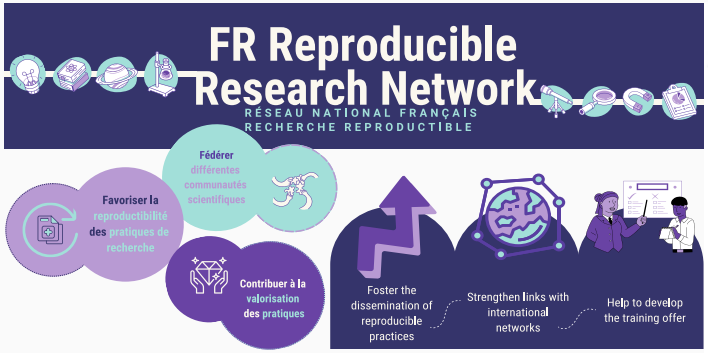
A non-technical introduction to reproducibility issues

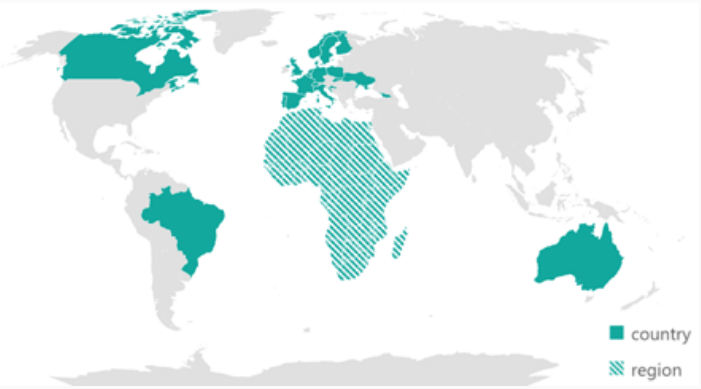




Resources and Acknowledgments

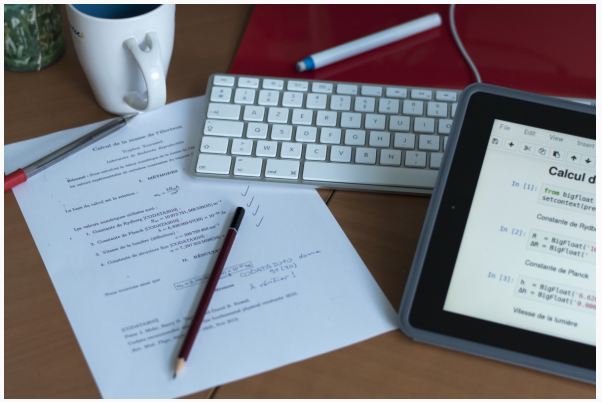
A non-technical introduction to reproducibility issues





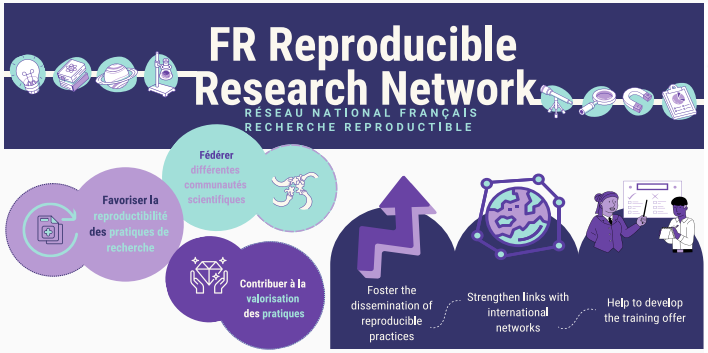
MOOC Reproducible Research: Methodological principles for a transparent science, Learning Lab

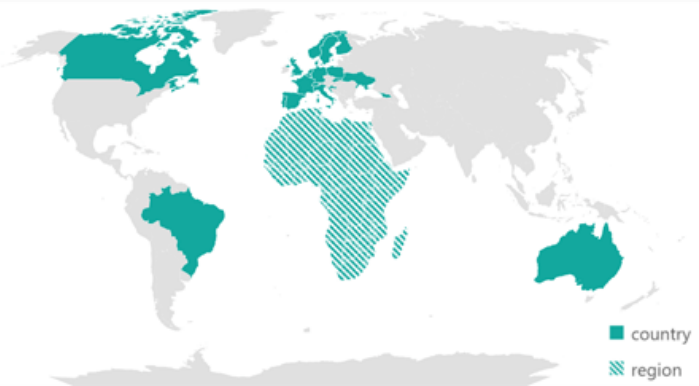
- Konrad Hinsen, Christophe Pouzat
- Laurence Farhi, Madeline Montigny, ...
- 2018 – ... (25,000+)



Resources and Acknowledgments

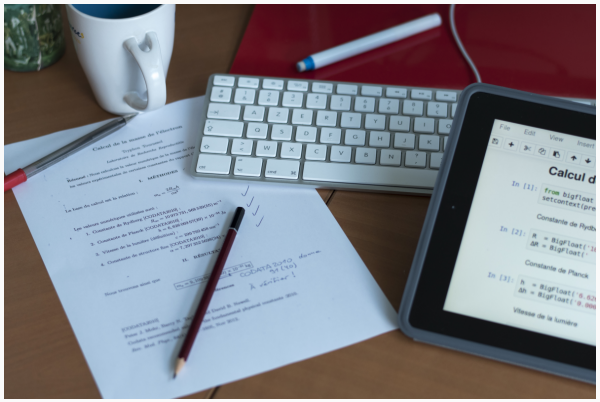
A non-technical introduction to reproducibility issues

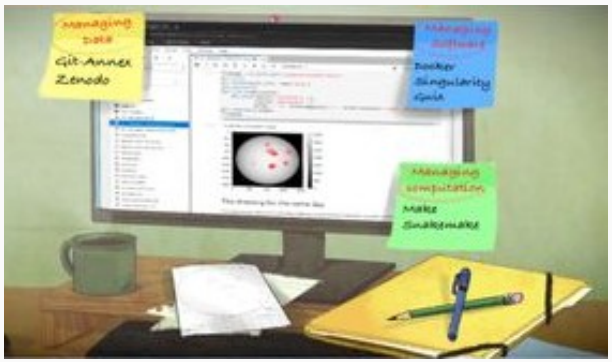




MOOC Reproducible Research: Methodological principles for a transparent science, Learning Lab

- Konrad Hinsen, Christophe Pouzat
- Laurence Farhi, Madeline Montigny, …
- 2018 – … (25,000+)





MOOC RR2: Practices and tools for managing computations and data 2024 – … (3,000+)

- Managing data (`git annex`, Zenodo, SWH)
- Software environment control (`docker`, `singularity`, `guix`)
- Scientific workflow (`make`, `snakemake`)

That's all Folks!