



# Obtaining Faithful/Reproducible Measurements on Modern CPUs

---

Arnaud Legrand and Tom Cornebize  
Univ. Grenoble Alpes, CNRS, Inria, Grenoble INP

Journée  – Reproductibilité de la recherche  
May 2021



# REPRODUCIBLE RESEARCH AND COMPUTER SCIENCE

---

# COMMON REPRODUCIBILITY CONCERNS IN MODERN SCIENCE

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

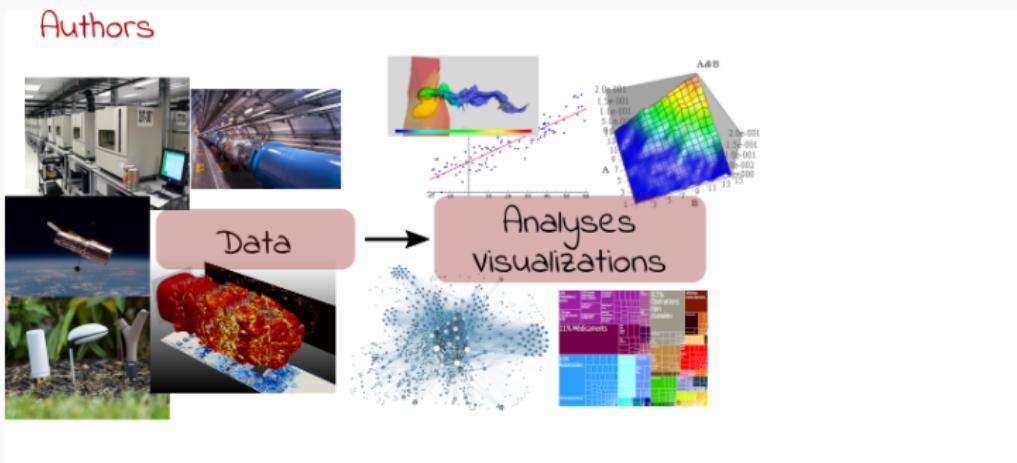
Authors



Data

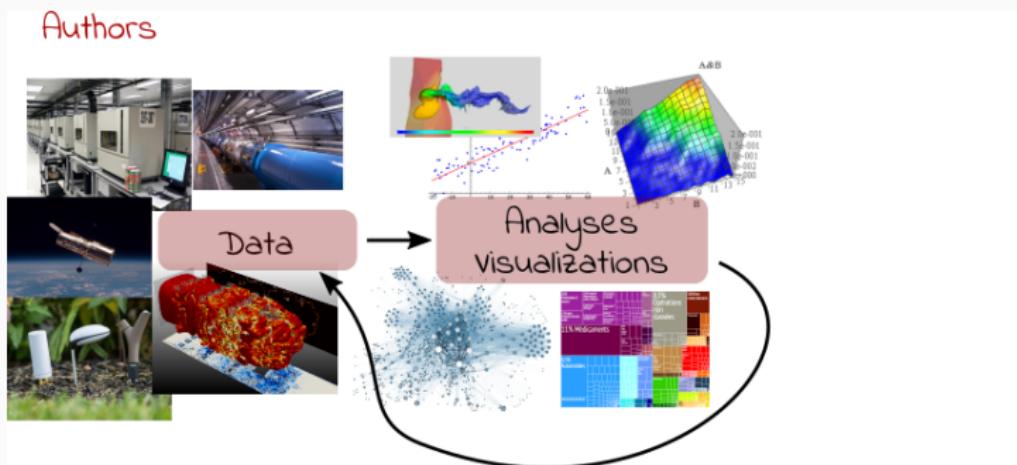
# COMMON REPRODUCIBILITY CONCERNS IN MODERN SCIENCE

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



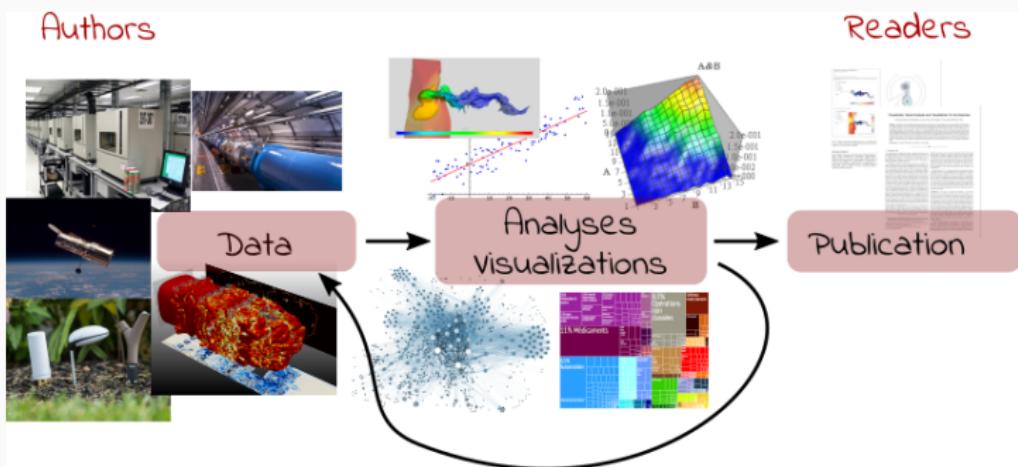
# COMMON REPRODUCIBILITY CONCERNS IN MODERN SCIENCE

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



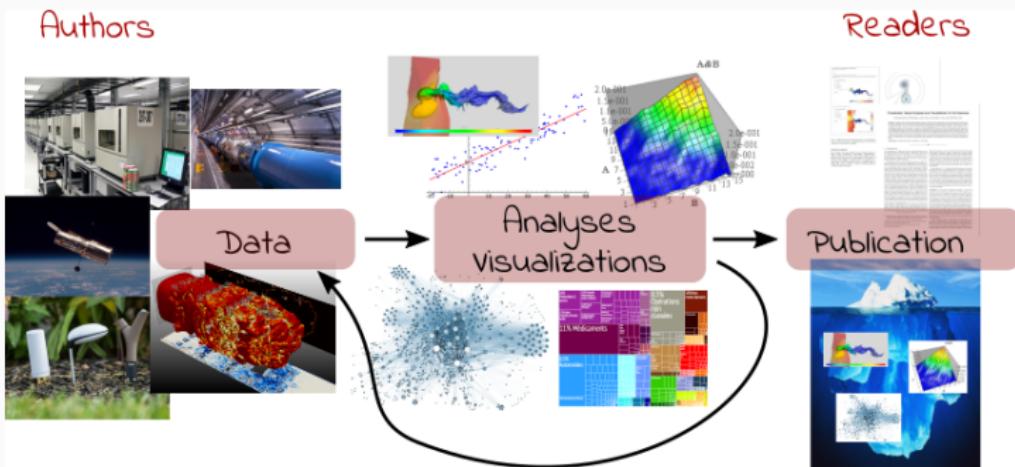
# COMMON REPRODUCIBILITY CONCERN IN MODERN SCIENCE

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



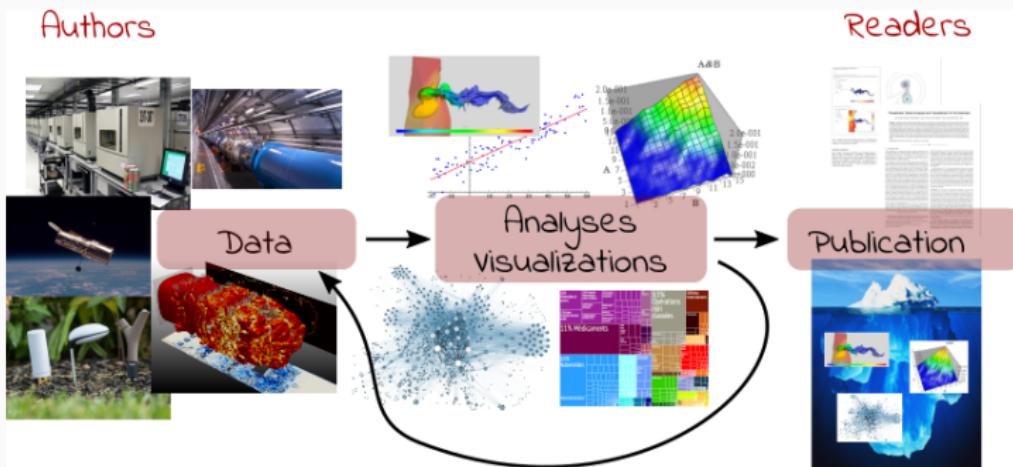
# COMMON REPRODUCIBILITY CONCERNS IN MODERN SCIENCE

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



# COMMON REPRODUCIBILITY CONCERNS IN MODERN SCIENCE

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



Reproducible Research = Bridging the Gap by working Transparently

# "REPRODUCIBLE RESEARCH": FIRST APPEARANCE

Claerbout & Karrenbach, meeting of the Society of Exploration Geophysics, 1992

## Electronic Documents Give Reproducible Research a New Meaning

RE1.3

Jon F. Claerbout and Martin Karrenbach, Stanford Univ.

### SUMMARY

A revolution in education and technology transfer follows from the marriage of word processing and software command scripts. In this marriage an author attaches to every figure caption a pushbutton or a name tag usable to recalculate the figure from all its data, parameters, and programs. This provides a new meaning of reproducibility in computer documents.

In 1990, we set this sequence of goals:

- Learn how to merge a publication with its underlying computational analysis.
- Teach researchers how to prepare a document in a form where they themselves can reproduce their own research results a year or more later by "pressing a single button".
- Learn how to leave finished work in a condition where coworkers can reproduce the calculation including the final illustration by pressing a button in its caption.
- Prepare a complete copy of our local software environment so that graduating students can take their work away with them to other sites, press a button, and reproduce their Stanford work.
- Merge electronic documents written by multiple authors (SEP reports).

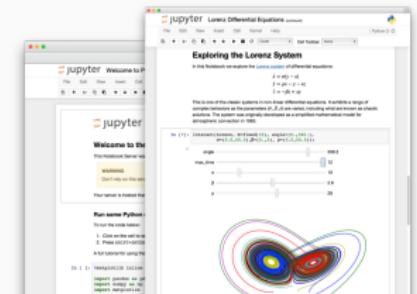
- make incremental improvements in electronic-document software
- seek partners for broadening standards (and making incremental improvements).

Our basic goal is reproducible research. The electronic document is our means to this end. In principle, reproducibility in research can be achieved without electronic documents and that is how we started. Our first nonelectronic reproducible document was a textbook in which the paper document contained the name of a program script in every figure caption. The program scripts were organized by book chapter and section so they could be correlated to an accompanying magnetic tape dump of the file system. The magnetic tape also contained all the necessary data to feed the program script.

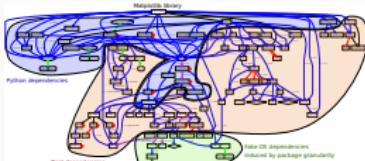
Now that we have begun using CD-ROM publication, we can go much further. Every figure caption contains a pushbutton that jumps to the appropriate science directory (folder) and initiates a figure rebuild command and then displays the figure, possibly as a movie or interactive program. We normally display seismic images of the earth's interior, but to reach wider audiences, Figure 1 shows a satellite weather picture which the pushbutton will animate as seen on commercial television. We include all our plot software as well as freely available software from many sources, including compilers and the L<sup>A</sup>T<sub>E</sub>X word processing systems. Naturally we cannot include licensed software, but with the exception

# EXISTING TOOLS, EMERGING STANDARDS

## Notebooks and workflows



## Software environments



## Sharing platforms



Scientific practices have greatly evolved, in particular since we rely on computers

How computers broke science – and what we can do about it

– Ben Marwick, The conversation, 2015



## 1. Computational science concerns:

**Genomics** software engineering, computational reproducibility, provenance

**Computational fluid dynamics** numerical issues

## 2. Statistical concerns:

**Social Psychology, Medical sciences, ...** methodology, statistics, pre-registration

What about Computer Science ?

Computer Science is young and inherits from Mathematics,  
Engineering, Linguistic, Nat. Sciences, ...

Purely theoretical scientists whose practice is close to mathematics  
may not be concerned (can't publish a math article without releasing  
the proofs).

- Have a look at talk by Vladimir Voevodsky in 2014 at Princeton 😊

*Les quatre concepts de l'informatique*, Gilles Dowek 2011:

- Algorithm, Machine, Language, Information

# WELL, I DESIGN ALGORITHMS!

---

- "Real" problems are all NP-hard, Log-APX, etc.
- Real workload = ~~NP-completeness-proof widgets~~, regularities and properties (difficult to formally state but that should be exploited)

Algorithms are evaluated on particular **workloads** that impact both their running time and the quality of the solutions

# WELL, I DESIGN ALGORITHMS!

- "Real" problems are all NP-hard, Log-APX, etc.
- Real workload = ~~NP-completeness-proof~~ widgets, regularities and properties (difficult to formally state but that should be exploited)

Algorithms are evaluated on particular **workloads** that impact both their running time and the quality of the solutions

**Image Processing:** True horror stories, E. Meinhardt-Llopis, CANUM 2016

- *The proposed multigrid algorithm converges to the solution of the problem in  $O(N)$  using biharmonic functions*
- Surprisingly, our naive multi-scale Gauss-Seidel converges much faster

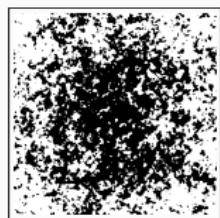
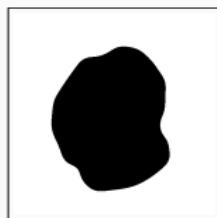
# WELL, I DESIGN ALGORITHMS!

- "Real" problems are all NP-hard, Log-APX, etc.
- Real workload = ~~NP-completeness-proof~~ widgets, regularities and properties (difficult to formally state but that should be exploited)

Algorithms are evaluated on particular **workloads** that impact both their running time and the quality of the solutions

**Image Processing:** True horror stories, E. Meinhardt-Llopis, CANUM 2016

- *The proposed multigrid algorithm converges to the solution of the problem in  $O(N)$  using biharmonic functions*
- Surprisingly, our naive multi-scale Gauss-Seidel converges much faster



~~ IPOL



Machine Learning: Trouble at the lab, The Economist 2013

*According to some estimates, three-quarters of published scientific papers in the field of machine learning are bunk because of this "overfitting".*

– Sandy Pentland (MIT)

NeurIPS, ICLR: open reviews, reproducibility challenges

→ Joelle Pineau @ NeurIPS'18



Every month in CACM, there is an article about the ethical consequences of Machine Learning on:

- Car driving, Autonomous guns, Law enforcement (risk assessment, predictive policing), ... It's Not the Algorithm, It's the Data (CACM, Feb. 2017)
- Advertising, Loan attribution, Selection at University, Organ transplant

Increasing society concern about **fairness** and **transparency**

# EXPERIMENTING WITH COMPUTERS

---

*Computer science is not more related to computers than Astronomy to telescopes*

– Dijkstra (mis-attributed)

Right, why should we care about computers? They are deterministic machines after all, right? 😊

# EXPERIMENTING WITH COMPUTERS

*Computer science is not more related to computers than Astronomy to telescopes*

– Dijkstra (mis-attributed)

Right, why should we care about computers? They are **deterministic** machines after all, right? 😊

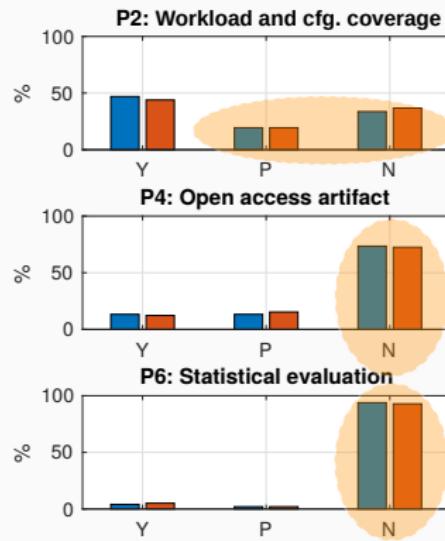
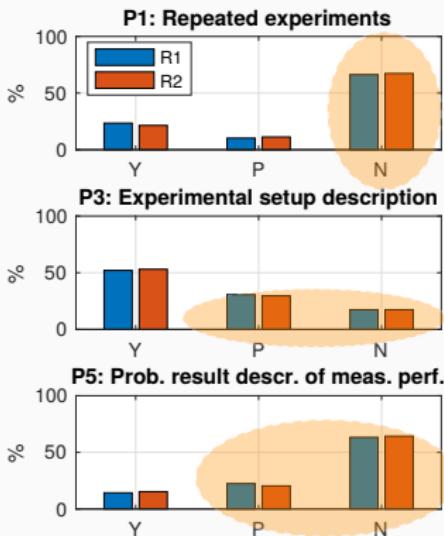
**Model ≠ Reality.** Although designed and built by human beings, computer systems are **so complex** that mistakes easily slip in...

**Our reality evolves!!!** The hardware keeps evolving so most results on old platforms quickly become obsolete (although, we keep building on such results 😊).

We need to regularly revisit and allow others to build on our work!

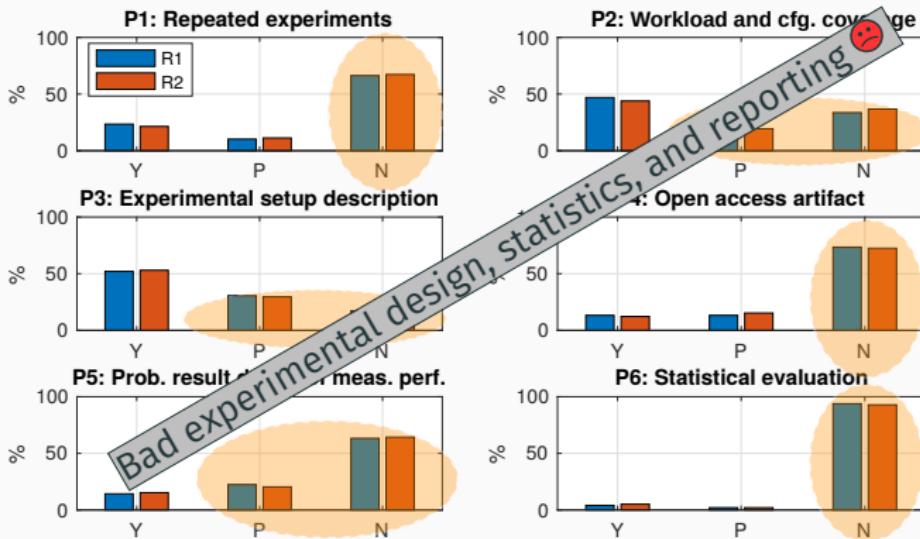
# KEY CONCERN FOR OUR COMMUNITY (ROOM FOR IMPROVEMENT)

How are cloud performance currently obtained and reported?, Methodological Principles for Reproducible Performance Evaluation in Cloud Computing, IEEE Trans. on Soft. Eng., July 2019



# KEY CONCERN FOR OUR COMMUNITY (ROOM FOR IMPROVEMENT)

How are cloud performance currently obtained and reported?, Methodological Principles for Reproducible Performance Evaluation in Cloud Computing, IEEE Trans. on Soft. Eng., July 2019



Key DoE principles:

1. Replicate to increase reliability.
2. Randomize to reduce bias  $\rightsquigarrow$  Evaluate statistical confidence.

## HORROR STORIES WHEN MEASURING CPU PERFORMANCE

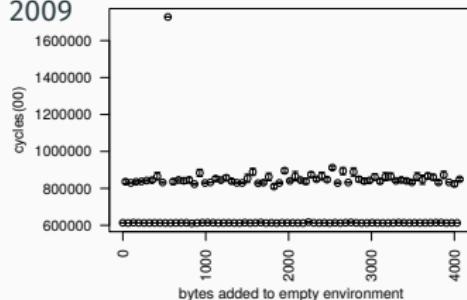
---

# MEASURING PERFORMANCE IS DIFFICULT

*Producing wrong data without doing anything obviously wrong!*

Mytkowicz et al. in ACM SIGPLAN Not. 44(3), March 2009

*changing the size of environment variables can trigger performance degradation as high as 300%; simply changing the link order of object files can cause performance to decrease by as much as 57%.*

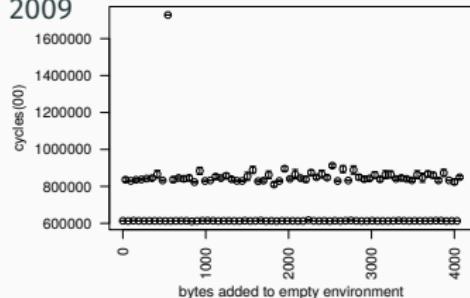


# MEASURING PERFORMANCE IS DIFFICULT

Producing wrong data without doing anything obviously wrong!

Mytkowicz et al. in ACM SIGPLAN Not. 44(3), March 2009

changing the size of *environment variables* can trigger performance degradation as high as 300%; simply changing the *link order* of object files can cause performance to decrease by as much as 57%.



Taming the Influence of Memory Layout. STABILIZER: Statistically Sound Performance Evaluation, C. Curtsinger and E. Berger in ASPLOS 2013

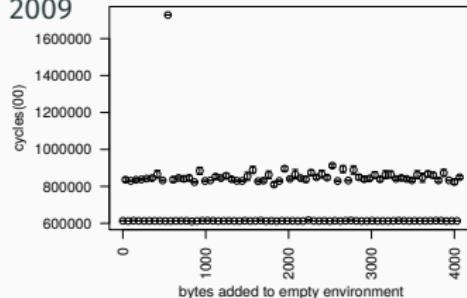
STABILIZER forces executions to sample the space of memory configurations by *repeatedly rerandomizing* layouts of code, stack, and heap objects at run-time. [...] Re-randomization ensures that layout effects *follow a Gaussian distribution*, enabling the use of statistical tests like ANOVA.

# MEASURING PERFORMANCE IS DIFFICULT

Producing wrong data without doing anything obviously wrong!

Mytkowicz et al. in ACM SIGPLAN Not. 44(3), March 2009

changing the size of *environment variables* can trigger performance degradation as high as 300%; simply changing the *link order* of object files can cause performance to decrease by as much as 57%.



Taming the Influence of Memory Layout. *STABILIZER: Statistically Sound Performance Evaluation*, C. Curtsinger and E. Berger in ASPLOS 2013

STABILIZER forces executions to sample the space of memory configurations by *repeatedly rerandomizing* layouts of code, stack, and heap objects at run-time. [...] Re-randomization ensures that layout effects *follow a Gaussian distribution*, enabling the use of statistical tests like ANOVA.

Randomization helps fighting bias incurred by:

1. specific configurations       $AA \dots A \rightarrow A_1 A_2 \dots A_n$  (*pseudo-replication*)
2. temporary perturbations       $AA \dots A BB \dots B \rightarrow ABBAAB \dots$

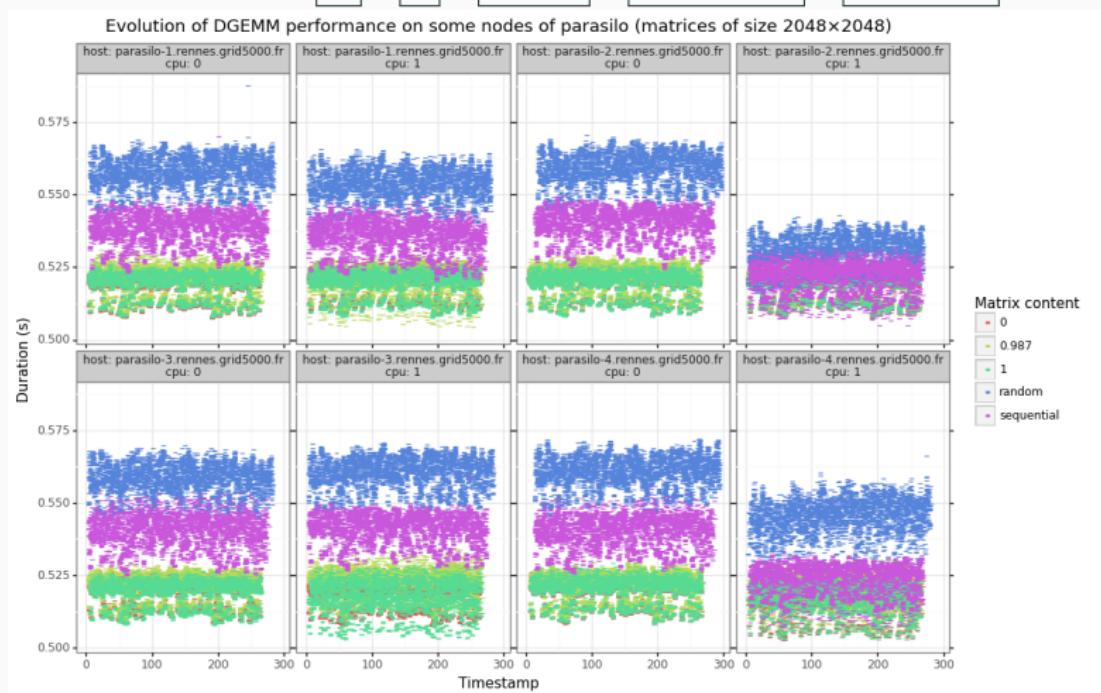
# ON THE IMPORTANCE OF CONTENT INITIALIZATION

- $C = A \times A$  ( $2048 \times 2048$ ), independant
- Time scale = 5 minutes
- A initialized with

# ON THE IMPORTANCE OF CONTENT INITIALIZATION

- $C = A \times A$  ( $2048 \times 2048$ ), independant
- $A$  initialized with 

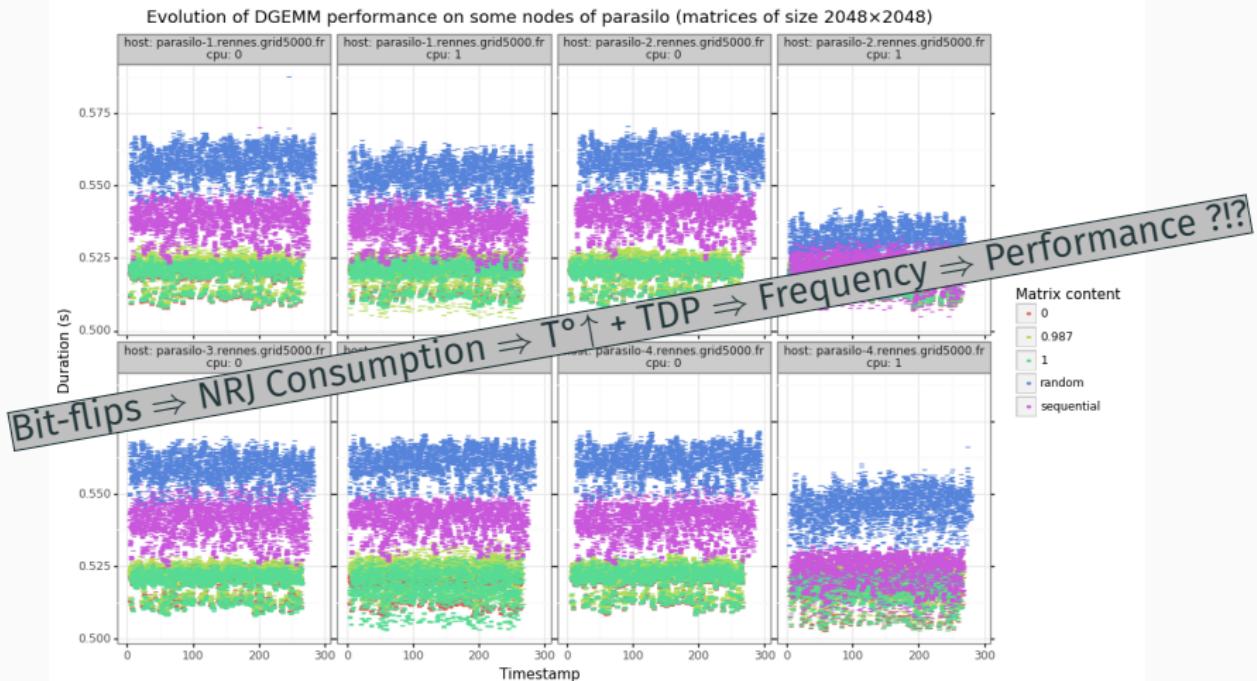
0	1	0.987	1, 2, 3, ...	random ?
---	---	-------	--------------	----------
- Time scale = 5 minutes



# ON THE IMPORTANCE OF CONTENT INITIALIZATION

- $C = A \times A$  ( $2048 \times 2048$ ), independant
- $A$  initialized with 

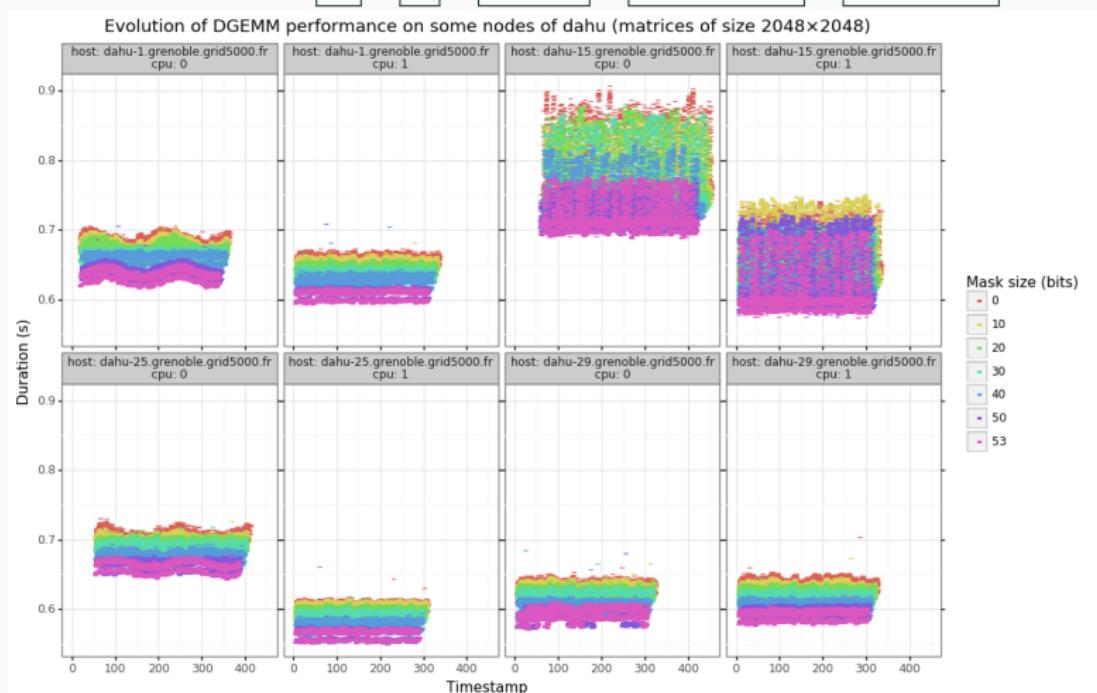
0	1	0.987	1, 2, 3, ...	random ?
---	---	-------	--------------	----------
- Time scale = 5 minutes



# ON THE IMPORTANCE OF CONTENT INITIALIZATION

- $C = A \times A$  ( $2048 \times 2048$ ), independant
- $A$  initialized with 

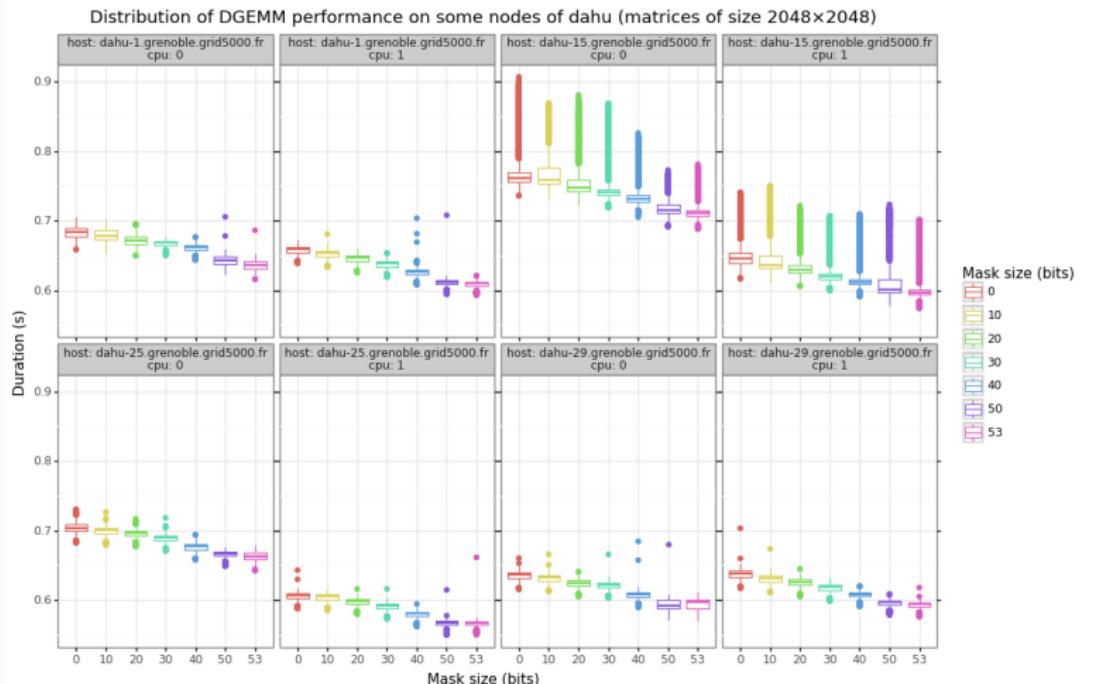
0	1	0.987	1, 2, 3, ...	random ?
---	---	-------	--------------	----------
- Time scale = 5 minutes



# ON THE IMPORTANCE OF CONTENT INITIALIZATION

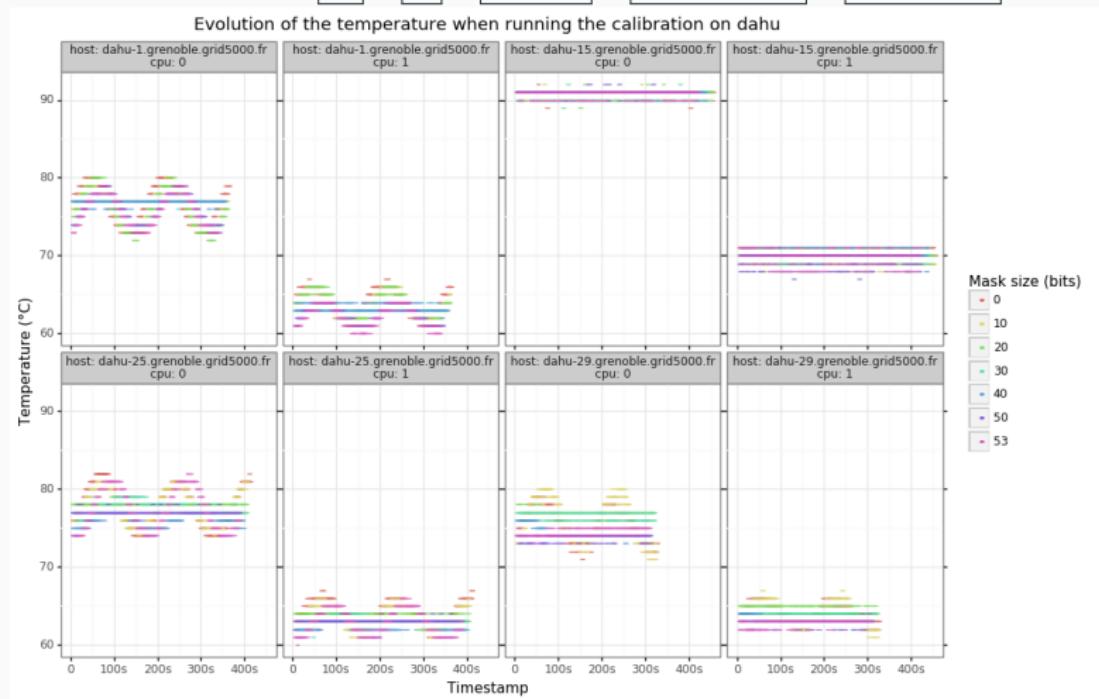
- $C = A \times A$  ( $2048 \times 2048$ ), independant
- $A$  initialized with 

0	1	0.987	1, 2, 3, ...	random ?
---	---	-------	--------------	----------
- Time scale = 5 minutes



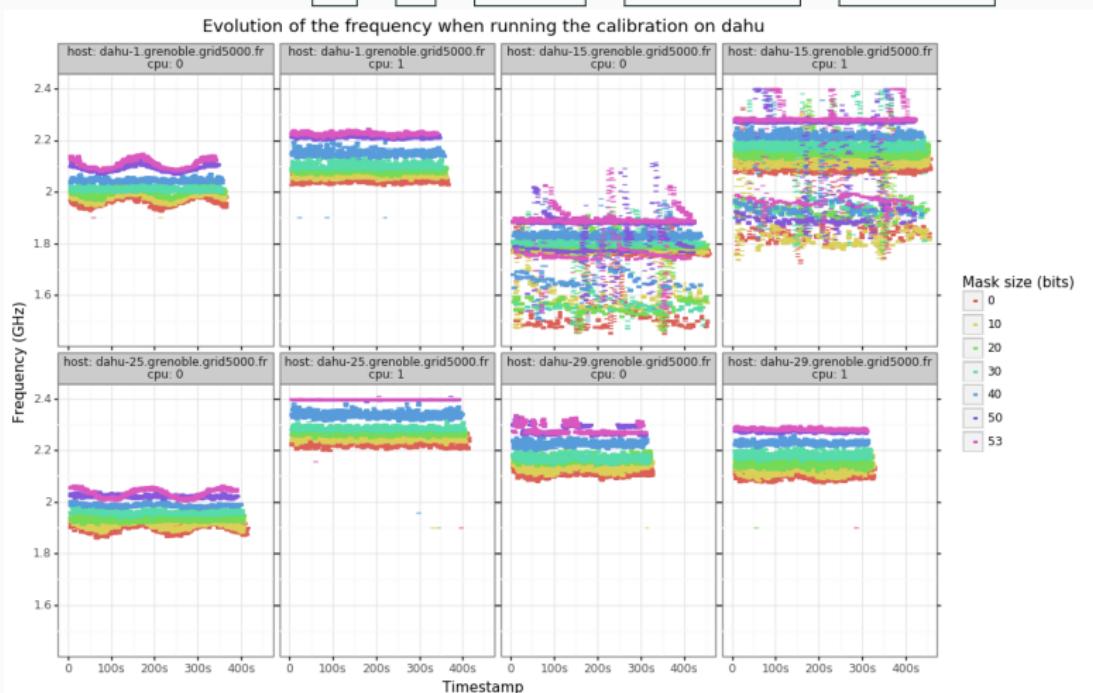
# ON THE IMPORTANCE OF CONTENT INITIALIZATION

- $C = A \times A$  ( $2048 \times 2048$ ), independant
- $A$  initialized with 0 1 0.987 1, 2, 3, ... random ?
- Time scale = 5 minutes



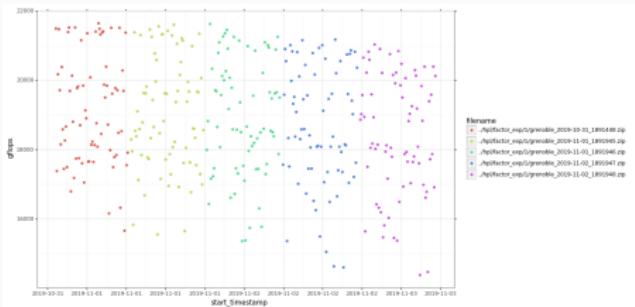
# ON THE IMPORTANCE OF CONTENT INITIALIZATION

- $C = A \times A$  ( $2048 \times 2048$ ), independant
- $A$  initialized with 0 1 0.987 1, 2, 3, ... random ?
- Time scale = 5 minutes



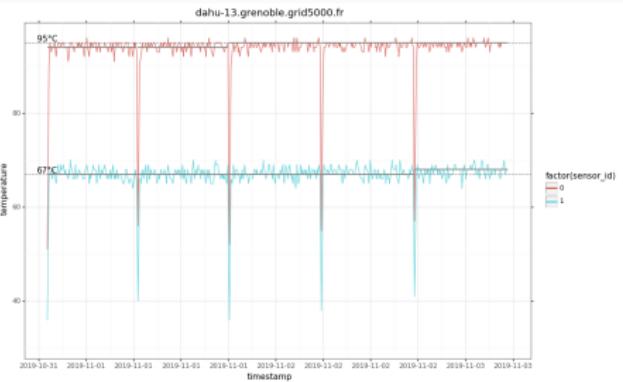
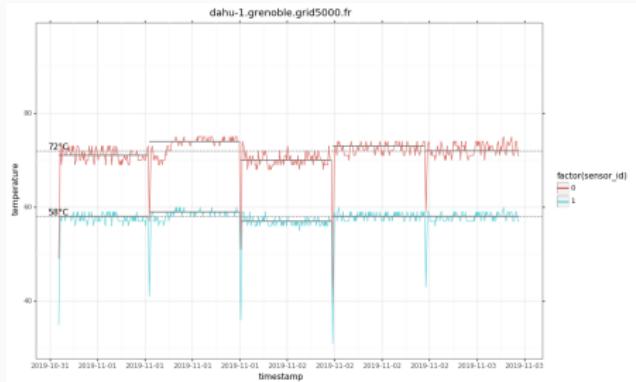
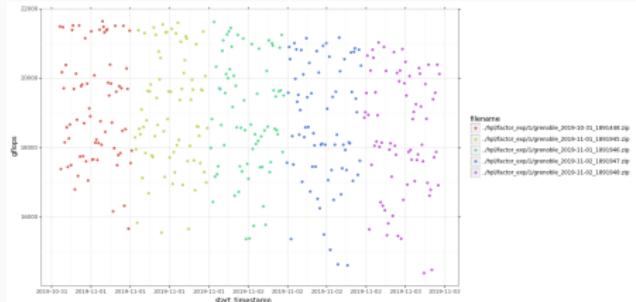
# AVOIDING "TEMPORARY" PERTURBATIONS (RANDOMIZING A FACTORIAL DESIGN)

- HPL performance (32 nodes, 70 cfg., 5 repetitions) • Time scale = 3 days



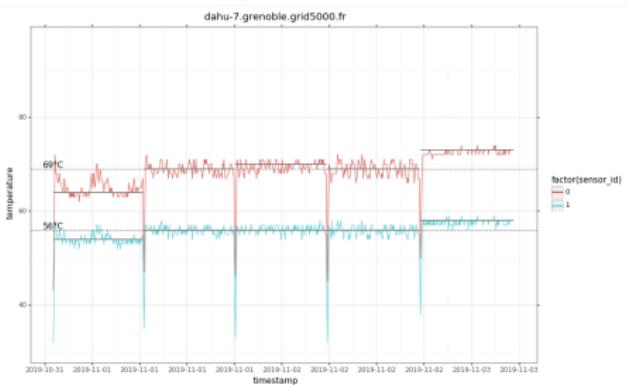
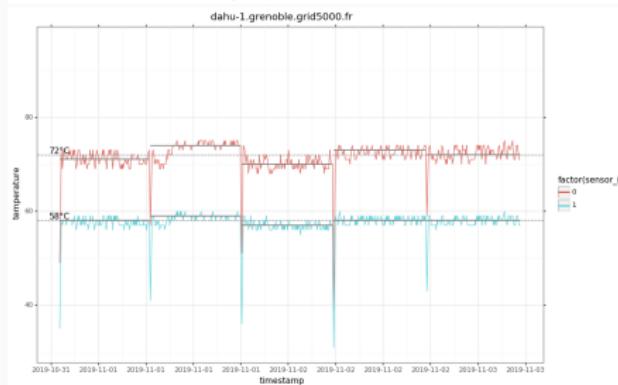
# AVOIDING "TEMPORARY" PERTURBATIONS (RANDOMIZING A FACTORIAL DESIGN)

- HPL performance (32 nodes, 70 cfg., 5 repetitions) • Time scale = 3 days

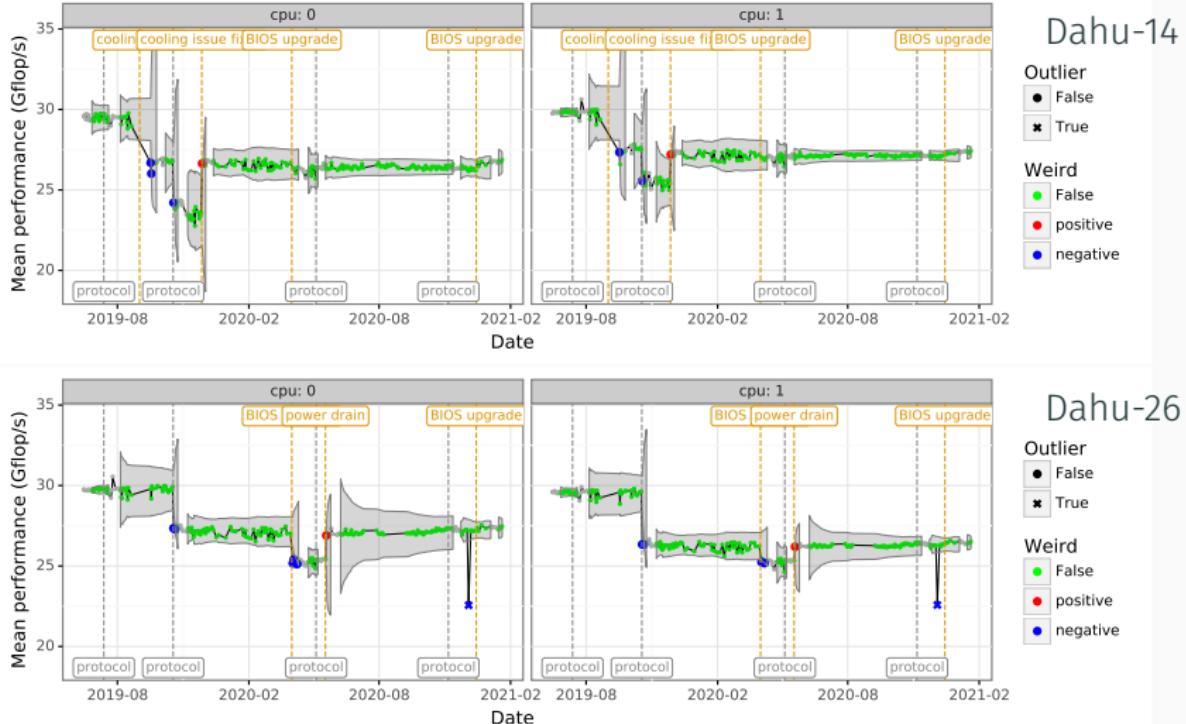


# AVOIDING "TEMPORARY" PERTURBATIONS (RANDOMIZING A FACTORIAL DESIGN)

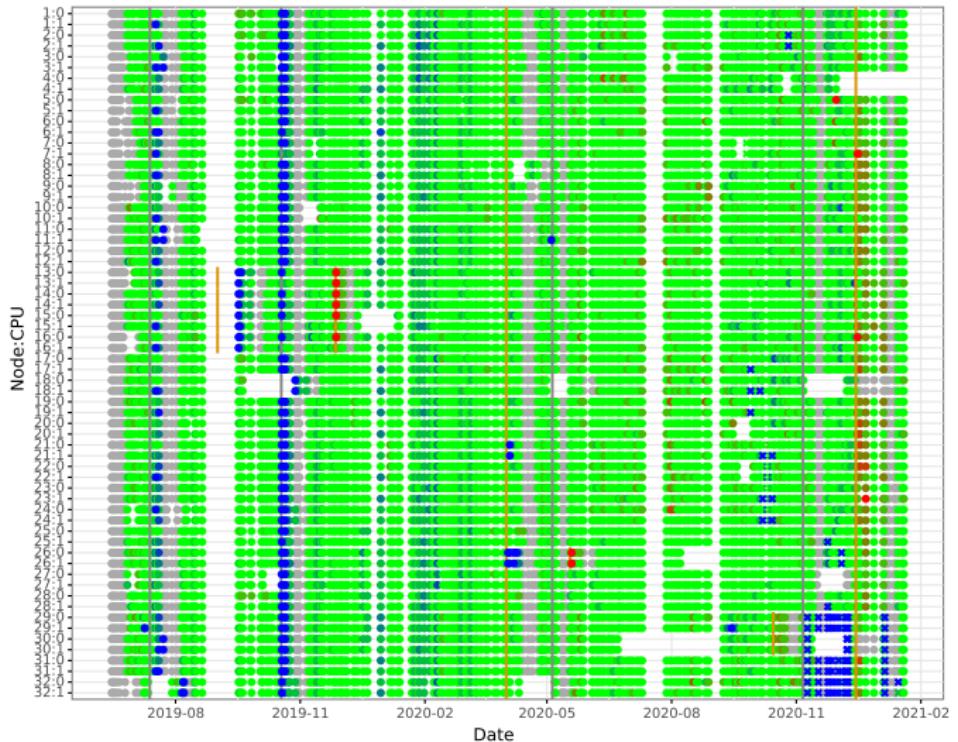
- HPL performance (32 nodes, 70 cfg., 5 repetitions) • Time scale = 3 days



# PLATFORM EVOLUTION OVER A LONG PERIOD



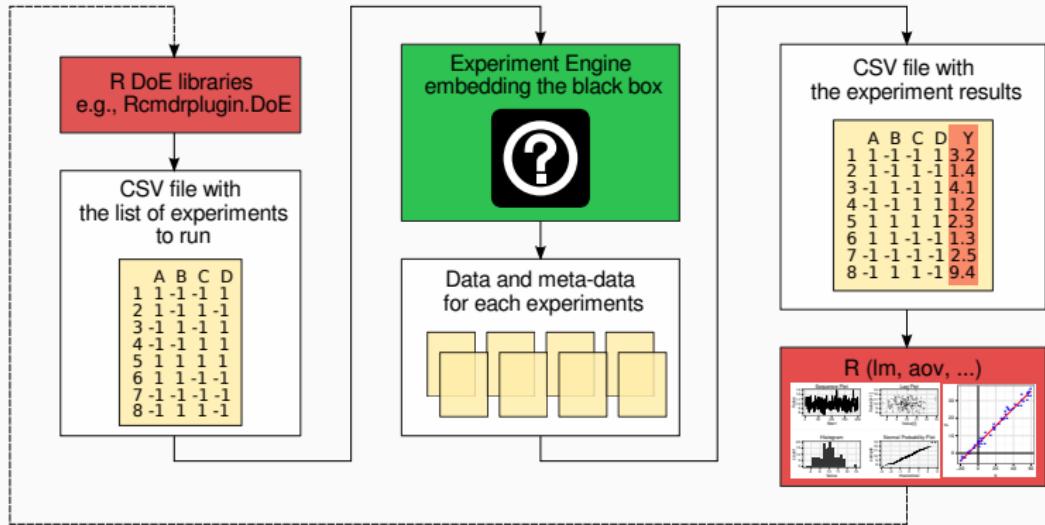
# PLATFORM EVOLUTION OVER A LONG PERIOD



## CONCLUSION

---

# EXPERIMENTAL METHODOLOGY: NOTICING THE UNEXPECTED



1. A **separation of concerns**
  - Transparent Measurement Procedure and Analysis Procedure
2. Randomized and Designed Experiments allowing to both:
  - Check the model and Instantiate it
3. Careful **recording of all experimental parameters** (before and during XPs)

# REPRODUCIBLE RESEARCH = RIGOR AND TRANSPARENCY

To err is human. Good research requires time and resources

1. Train yourself and your students: RR, statistics, experiments
  - Beware of checklists and norms
  - Understand what's at stake



#RRMooc 3rd Edition: ≈ Feb. 2020

A new MOOC: "Advanced RR" (Oct 2021?)

- Managing data (HDF5, archiving)
- Software environment control (Docker, GUIX)
- Scientific workflow (snakemake)

# REPRODUCIBLE RESEARCH = RIGOR AND TRANSPARENCY

To err is human. Good research requires time and resources

1. Train yourself and your students: RR, statistics, experiments
  - Beware of checklists and norms
  - Understand what's at stake



#RRMooc 3rd Edition: ≈ Feb. 2020

A new MOOC: "Advanced RR" (Oct 2021?)

- Managing data (HDF5, archiving)
- Software environment control (Docker, GUIX)
- Scientific workflow (snakemake)

2. Change the norm: make publication practices evolve
  - Require data, code, environment, XP protocol, ...
3. Incentive: consider RR/open science when hiring/promoting

# REPRODUCIBLE RESEARCH = RIGOR AND TRANSPARENCY

To err is human. Good research requires time and resources

1. Train yourself and your students: RR, statistics, experiments
  - Beware of checklists and norms
  - Understand what's at stake



#RRMooc 3rd Edition: ≈ Feb. 2020

A new MOOC: "Advanced RR" (Oct 2021?)

- Managing data (HDF5, archiving)
- Software environment control (Docker, GUIX)
- Scientific workflow (snakemake)

2. Change the norm: make publication practices evolve

- Require data, code, environment, XP protocol, ...

3. Incentive: consider RR/open science when hiring/promoting

4. Prepare the Future: How to share Experiments?

- Reuse, reuse, reuse!
  - Shared and controlled testbeds (e.g., Grid'5000/FIT-IoTLab)
  - Toward **literate experimentation?**
- 4–8 October, 2021 @ Strasbourg  
16th GDR RSD Fall School: Reproductibilité et recherche expérimentale en réseaux et en systèmes <https://rsd-ecole.cnrs.fr/>