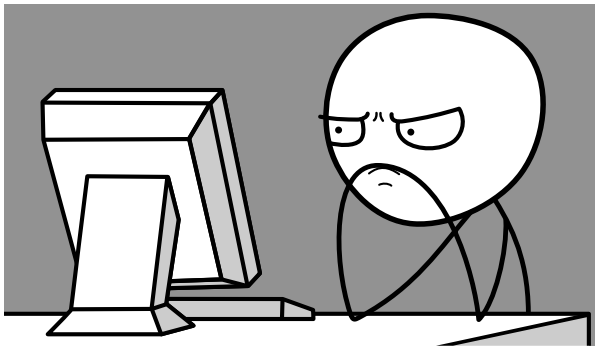
Quality Assurance of a HPC Cluster: Testing for Performance non-Regression

Tom Cornebize, Arnaud Legrand
Laboratoire d'Informatique de Grenoble

November 5, 2019

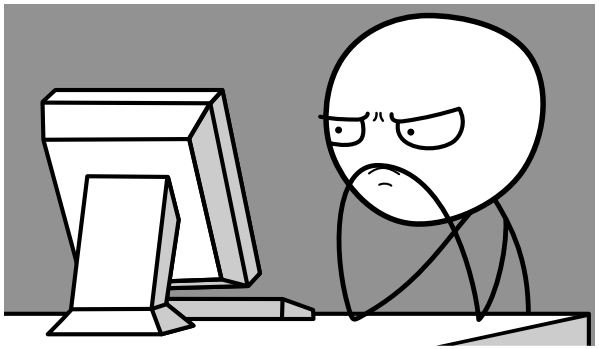
Context

Typical Performance Evaluation Questions (Given my application and a supercomputer)



- **Before** running
  - How many nodes?
  - For how long?
  - Which parameters?

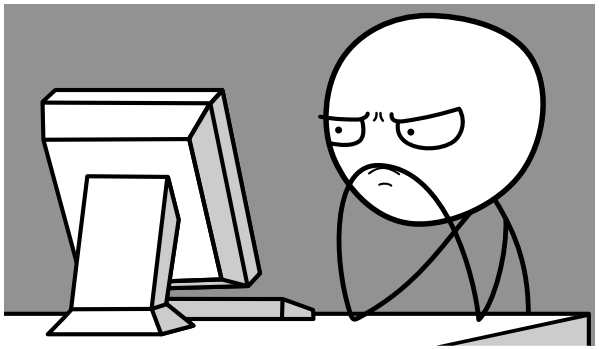
Typical Performance Evaluation Questions (Given my application and a supercomputer)



- **Before** running
  - How many nodes?
  - For how long?
  - Which parameters?
- **After** running
  - Performance as “expected”?
  - Problem in the app or the platform?

Context

Typical Performance Evaluation Questions (Given my application and a supercomputer)

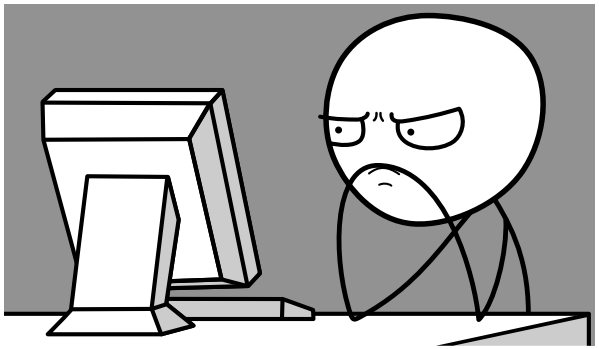


- **Before** running
  - How many nodes?
  - For how long?
  - Which parameters?
- **After** running
  - Performance as “expected”?
  - Problem in the app or the platform?

So many large-scale runs, solely to tune performance?!?

Context

Typical Performance Evaluation Questions (Given my application and a supercomputer)



- **Before** running
  - How many nodes?
  - For how long?
  - Which parameters?
- **After** running
  - Performance as "expected"?
  - Problem in the app or the platform?

So many large-scale runs, solely to tune performance?!?

Holy Grail: Predictive Simulation on a "Laptop"

Performance measures

Building a predictive model of the durations:

- Computations (`dgemm`, ...)
- Communications (`MPI_send`, ...)

A lot of measures, with different input sizes

Performance measures

Building a predictive model of the durations:

- Computations (`dgemm`, ...)
- Communications (`MPI_send`, ...)

A lot of measures, with different input sizes

Some troubles, wrong predictions
$\Rightarrow$ Needed to investigate.

Down the rabbit hole (1)

Possible issues

- Most of the problems are human mistakes (wrong library version, wrong options, ...)
- A lot of transient phenomena: OS scheduler, temperature changes, core frequencies oscillation...
- A measure can have an impact (positive or negative) on the next measure (cache effects...)

Down the rabbit hole (1)

Possible issues

- Most of the problems are human mistakes (wrong library version, wrong options, ...)
- A lot of transient phenomena: OS scheduler, temperature changes, core frequencies oscillation...
- A measure can have an impact (positive or negative) on the next measure (cache effects...)

Design of experiments

- Randomizing the sequence of measures to reduce bias
- Tools: ad-hoc scripts to generate *experiment files*

Down the rabbit hole (2)

Automating the setup

- Job submission, deployment, software stack installation, experiment execution...
- Tools: OAR, Kadeploy, Peanut

Down the rabbit hole (2)

Automating the setup

- Job submission, deployment, software stack installation, experiment execution...
- Tools: OAR, Kadeploy, Peanut

Automating the metadata collection

- Date, kernel and library versions, output of every command, CPU temperature, core frequencies...
- Tools: Peanut, custom scripts

Down the rabbit hole (2)

Automating the setup

- Job submission, deployment, software stack installation, experiment execution...
- Tools: OAR, Kadeploy, Peanut

Automating the metadata collection

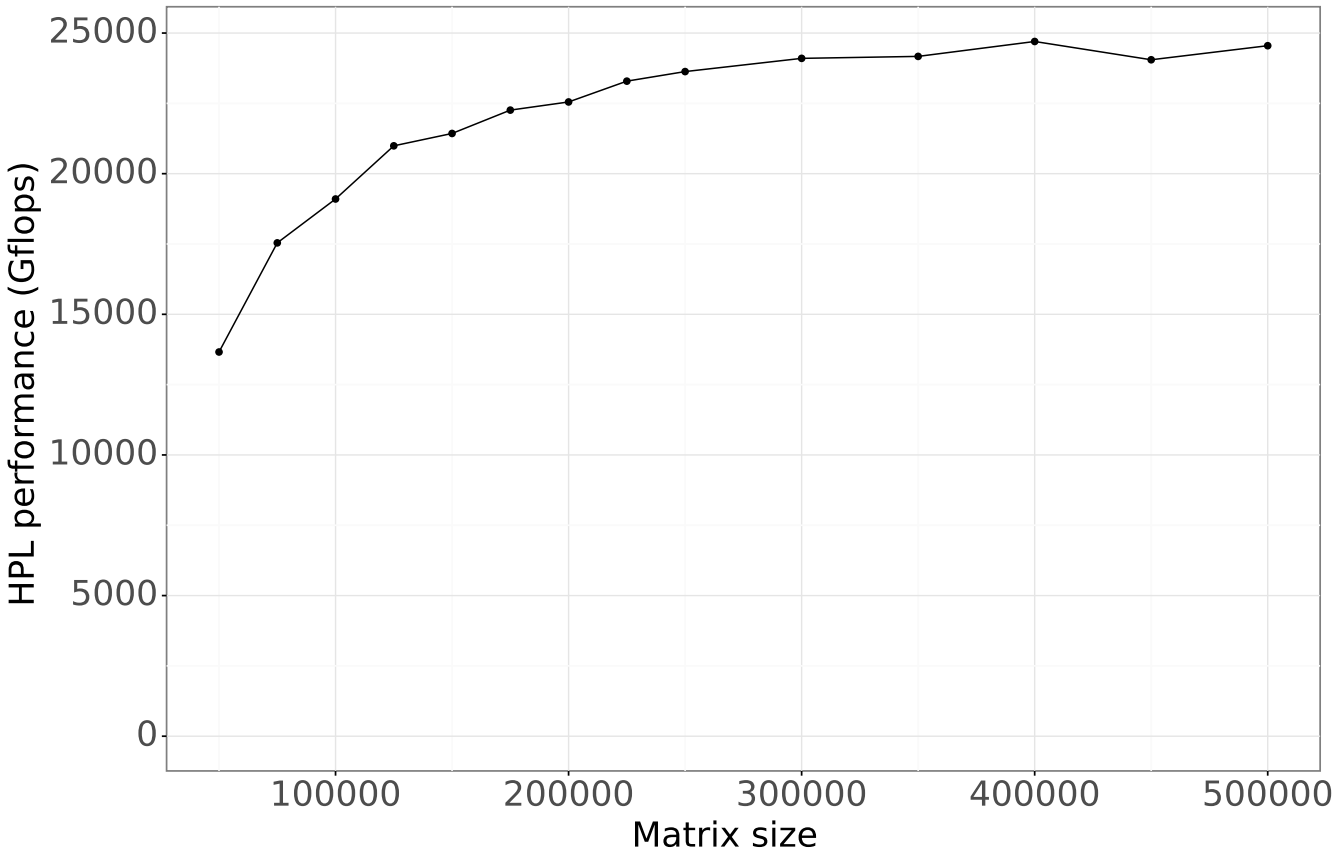
- Date, kernel and library versions, output of every command, CPU temperature, core frequencies...
- Tools: Peanut, custom scripts

Data analysis

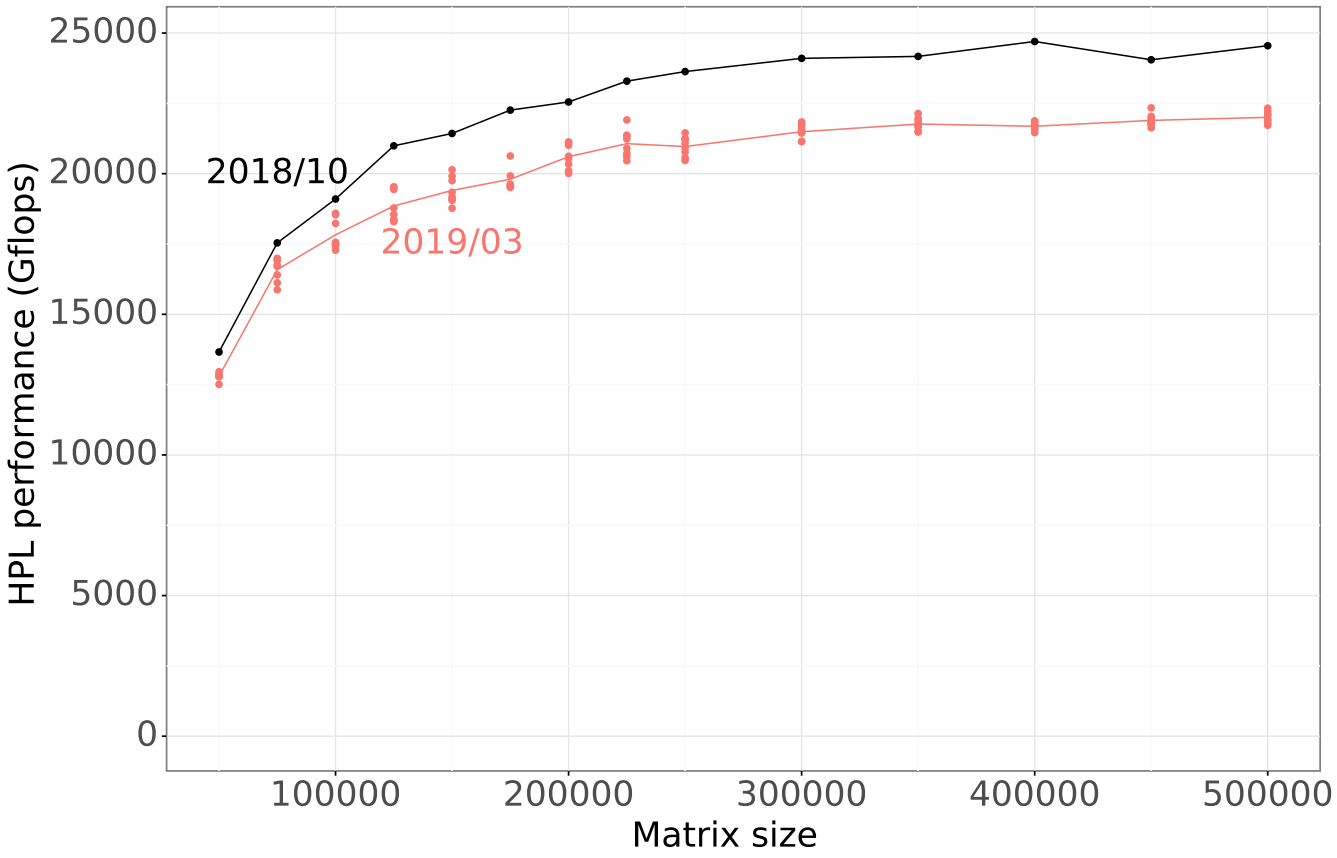
- Data visualization (correlations, temporal patterns, distributions)
- Statistics (linear regressions, ANOVA)
- Tools: Python & R with Jupyter, ggplot...

Dahu@Grid'5000

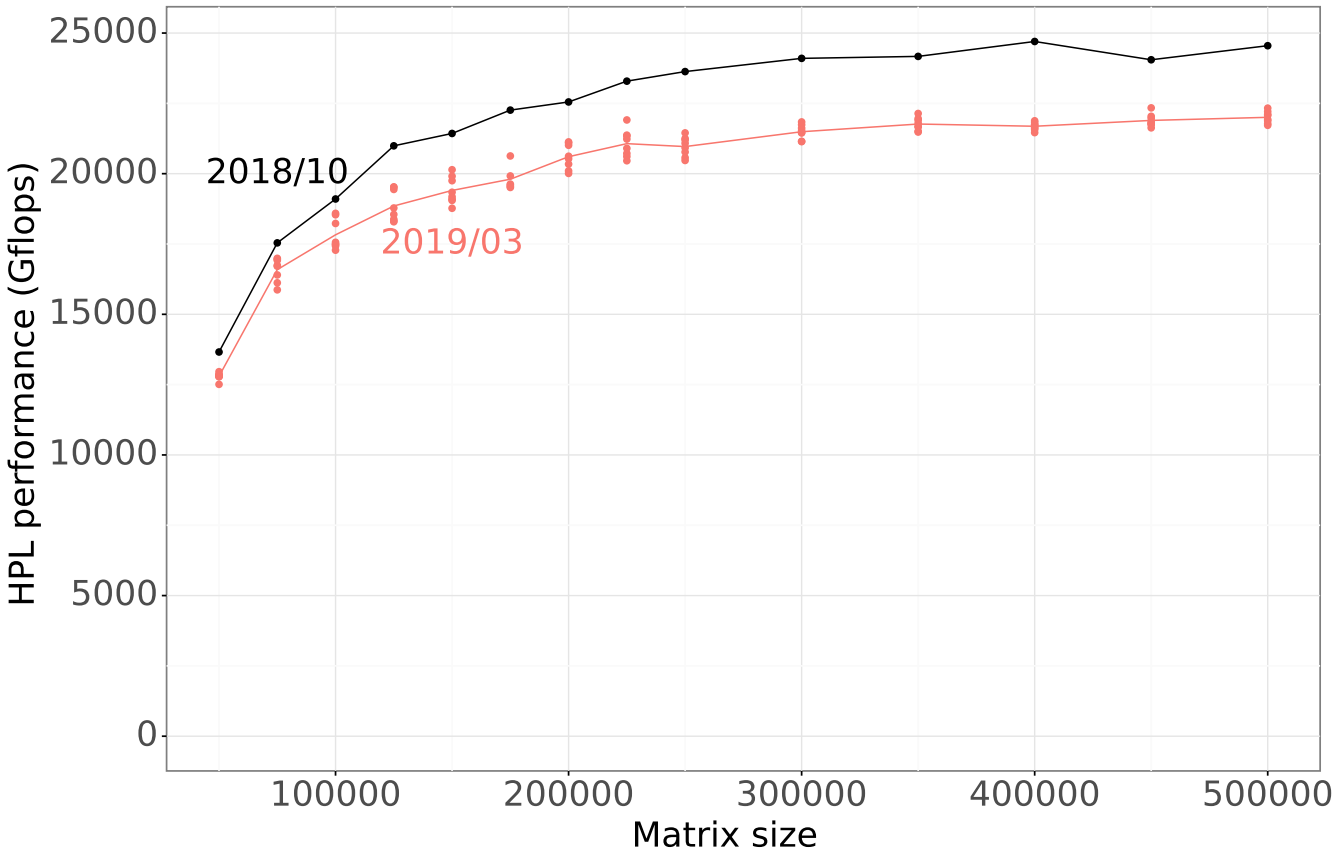
Performance of the whole cluster: parallel application



Performance of the whole cluster: parallel application

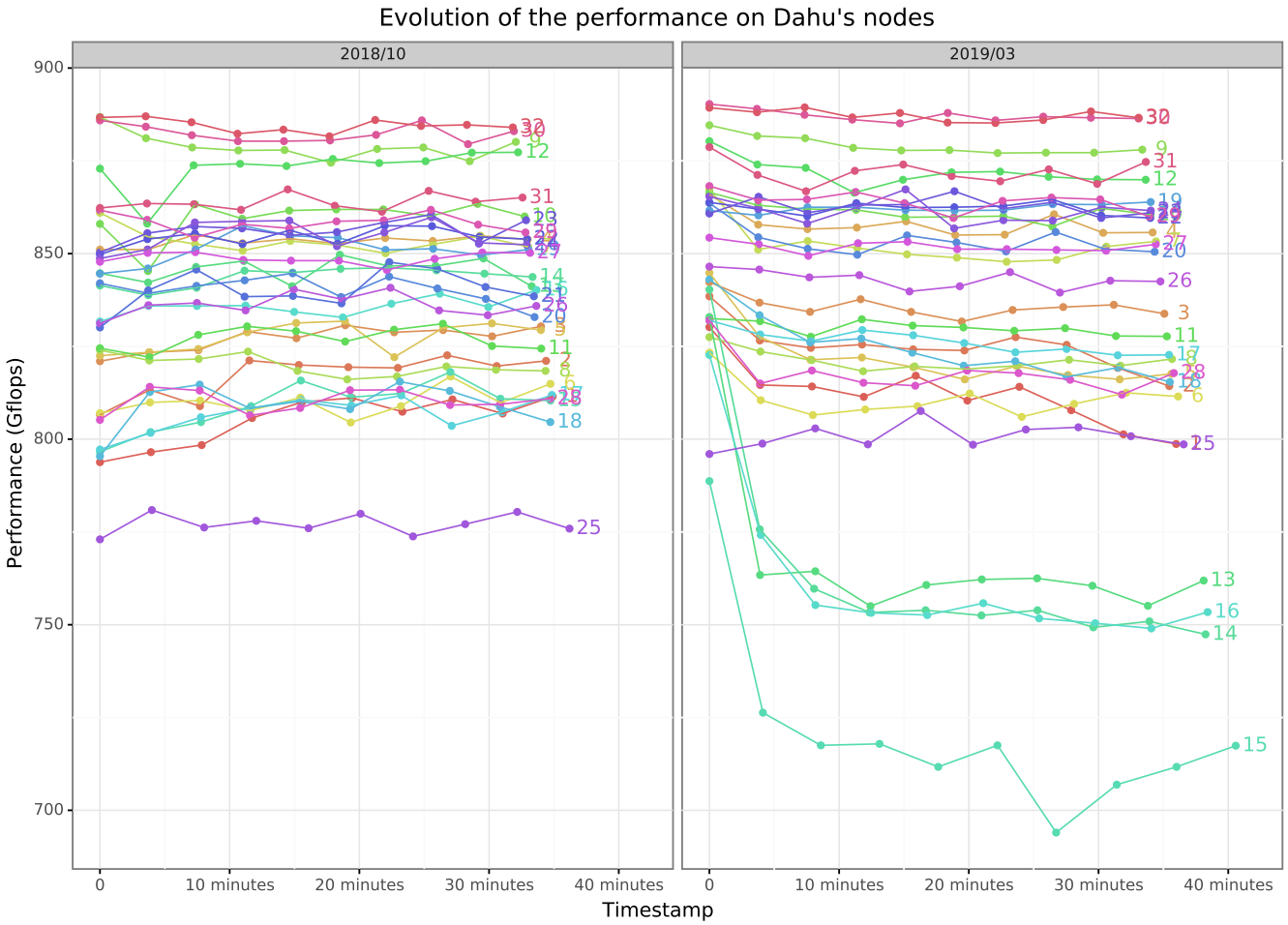


Performance of the whole cluster: parallel application



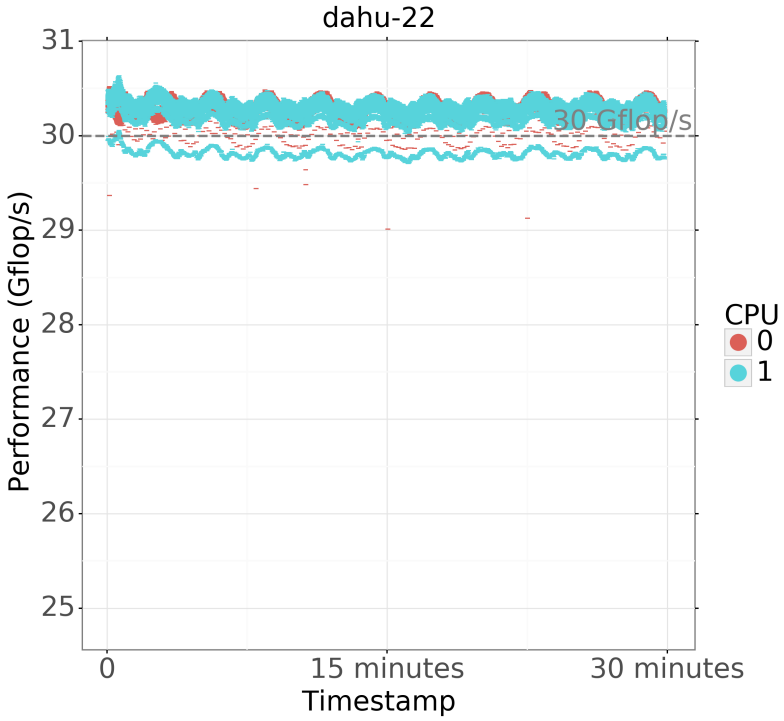
Same software, same hardware, 10% performance drop

Single-node performance

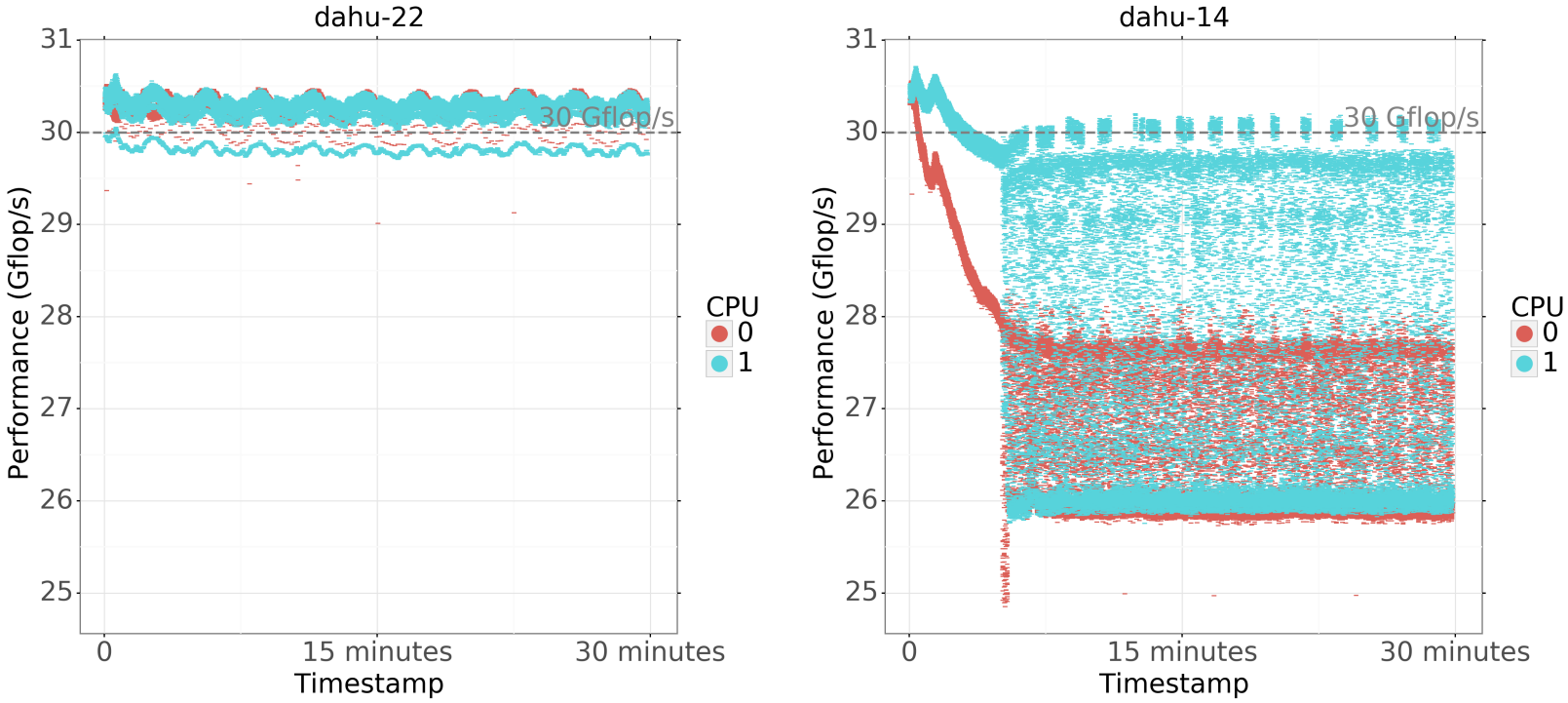


Performance drop for dahu-{13...16} after a few minutes

The Good & the Bad: Performance evolution

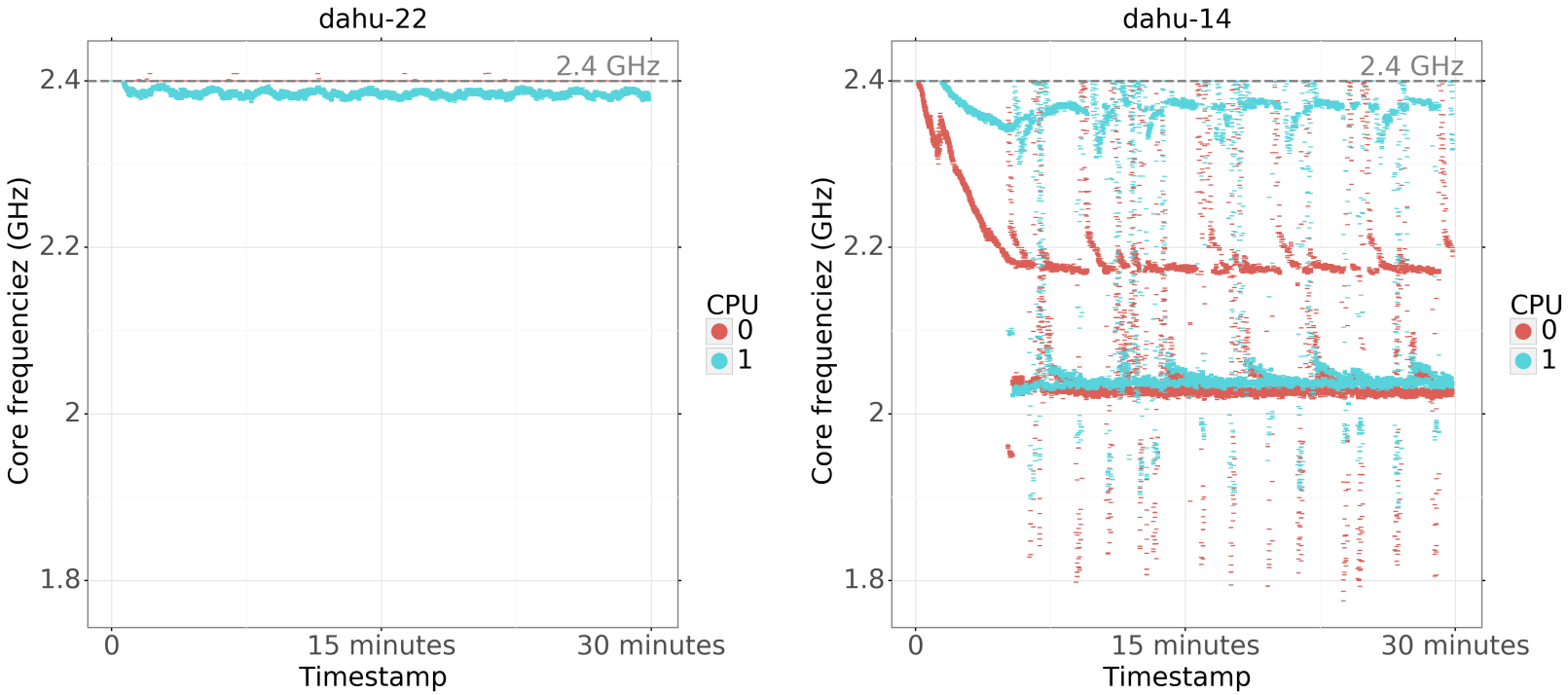


The Good & the Bad: Performance evolution



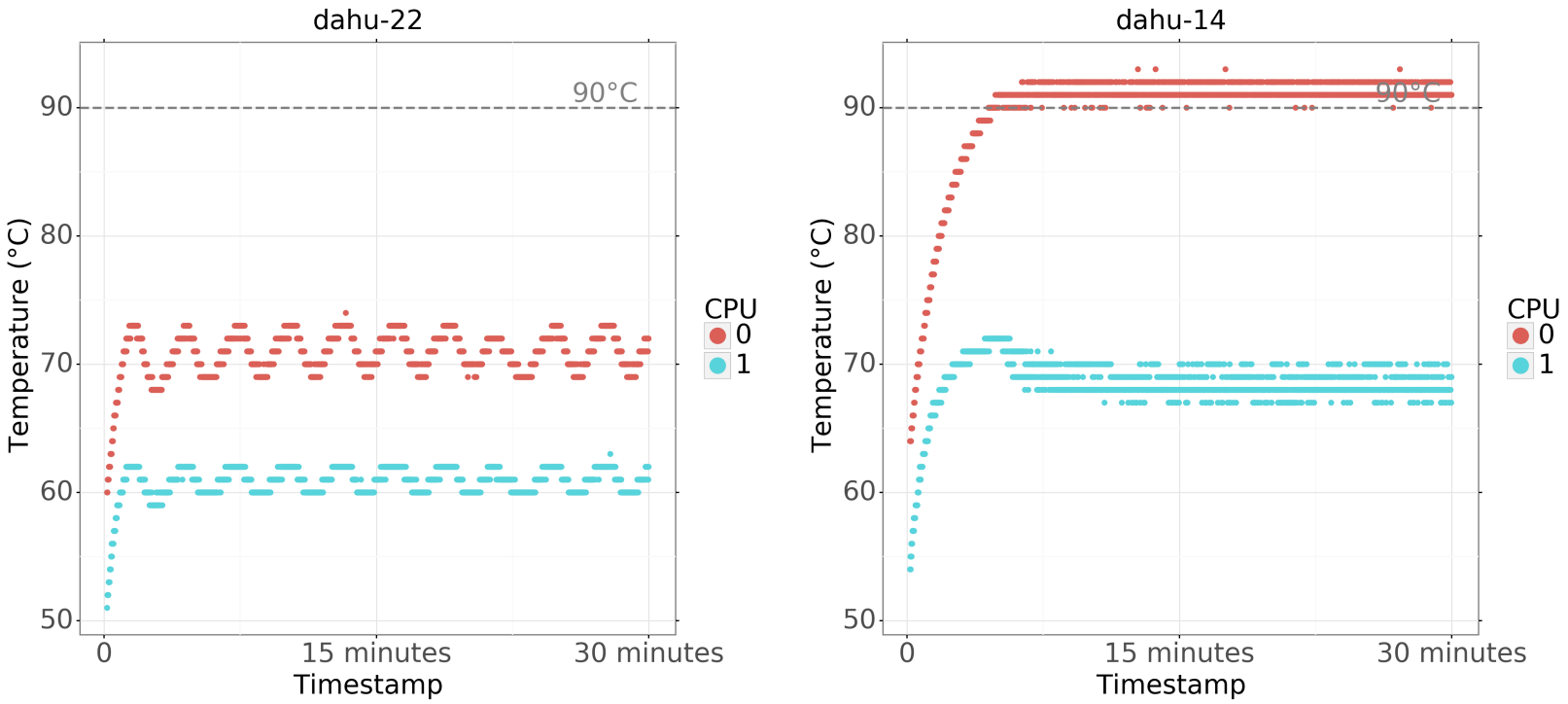
Performance drop, huge variability, CPU n°0 is worse

The Good & the Bad: Frequency evolution



Frequency drop, huge variability, CPU n°0 is worse

The Good & the Bad: Temperature evolution



Very high temperature ⇒ probably a cooling issue

Fixed by changing the node frames

The Ugly

Several other problems encountered on this cluster:

- Connectivity issue ($\Rightarrow$ replug the Omnipath cable, then reboot)

The Ugly

Several other problems encountered on this cluster:

- Connectivity issue ($\Rightarrow$ replug the Omnipath cable, then reboot)
- Memory bandwidth issue ($\Rightarrow$ change the faulty memory stick)

The Ugly

Several other problems encountered on this cluster:

- Connectivity issue ($\Rightarrow$ replug the Omnipath cable, then reboot)
- Memory bandwidth issue ($\Rightarrow$ change the faulty memory stick)
- Important heterogeneity between the nodes (10% difference between the slowest and the fastest, even without cooling problems)

Perspectives

Stepping back

Objective: ~~finding bugs~~ building models

Side effect: use these models for statistical tests, to automagically detect performance problems

Stepping back

Objective: ~~finding bugs~~ building models

Side effect: use these models for statistical tests, to automagically detect performance problems

Regular and semi-automated performance measures on Grid'5000: https://gitlab.in2p3.fr/tom.cornebize/g5k_data

Several problems detected, some very severe, others more subtile

Stepping back

Objective: ~~finding bugs~~ building models

Side effect: use these models for statistical tests, to automagically detect performance problems

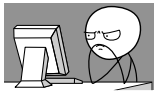
Regular and semi-automated performance measures on Grid'5000: https://gitlab.in2p3.fr/tom.cornebize/g5k_data

Several problems detected, some very severe, others more subtile

Dahu@Grid'5000 had troubles. What about Dahu@Ciment?

Context

Typical Performance Evaluation Questions (Given my application and a supercomputer)



- **Before** running
  - How many nodes?
  - For how long?
  - Which parameters?
- **After** running
  - Performance as "expected"?
  - Problem in the app or the platform?

So many large-scale runs, solely to tune performance?!?

Holy Grail: Predictive Simulation on a "Laptop"

Performance measures

Building a predictive model of the durations:

- Computations (`dgemm`, ...)
- Communications (`MPI_send`, ...)

A lot of measures, with different input sizes

Some troubles, wrong predictions
⇒ Needed to investigate.

Down the rabbit hole (2)

Automating the setup

- Job submission, deployment, software stack installation, experiment execution...
- Tools: OAR, Kadeploy, Peanut

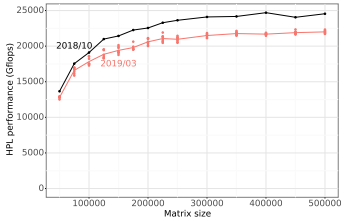
Automating the metadata collection

- Date, kernel and library versions, output of every command, CPU temperature, core frequencies...
- Tools: Peanut, custom scripts

Data analysis

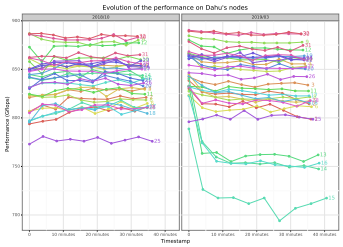
- Data visualization (correlations, temporal patterns, distributions)
- Statistics (linear regressions, ANOVA)
- Tools: Python & R with Jupyter, ggplot...

Performance of the whole cluster: parallel application



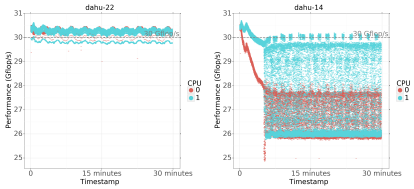
Same software, same hardware, 10% performance drop

Single-node performance



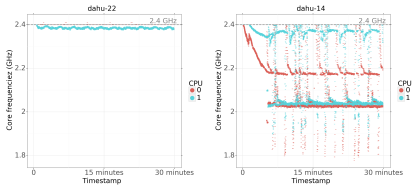
Performance drop for dahu-{13...16} after a few minutes

The Good & the Bad: Performance evolution



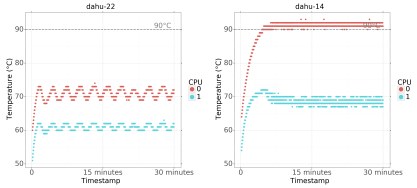
Performance drop, huge variability, CPU n°0 is worse

The Good & the Bad: Frequency evolution



Frequency drop, huge variability, CPU n°0 is worse

The Good & the Bad: Temperature evolution



Very high temperature ⇒ probably a cooling issue

Fixed by changing the node frames

Stepping back

Objective: ~~finding bugs~~ building models

Side effect: use these models for statistical tests, to automagically detect performance problems

Regular and semi-automated performance measures on Grid'5000: https://gitlab.in2p3.fr/tom.cornebize/g5k_data

Several problems detected, some very severe, others more subtile

Dahu@Grid'5000 had troubles. What about Dahu@Ciment?

tom.cornebize@univ-grenoble-alpes.fr

git https://gitlab.in2p3.fr/tom.cornebize/g5k_data