Intégration continue sur PlaFRIM

Un pipeline reproductible pour la non régression de performances

Sommaire

01.. Contexte

02.. Runner PlaFRIM

03.. Pipeline gitlab-ci

04.. Traitement des données

05.. Conclusion

01

Contexte

Thématique de recherche

Equipe Inria HiePACS

High-End Parallel Algorithms for Challenging Numerical Simulations

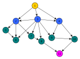
Linear algebra

$$\mathbf{AX} = \mathbf{B}$$

Sequential-Task-Flow

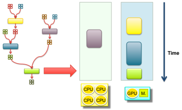
```
for (j = 0; j < N; j++)
Task (A[j]);
```

Direct Acyclic Graph



→

Runtime systems



→

Heterogeneous platforms

Objectifs : performances, passage à l'échelle

Parallèlisme : Threads, CUDA, MPI

Chameleon: matrices dense

- BLAS: opérations scalaires, vectoriel, matrix simple operations

$$\alpha\begin{pmatrix}\cdot\\\cdot\end{pmatrix},\quad \begin{pmatrix}\cdot\\\cdot\end{pmatrix}+\begin{pmatrix}\cdot\\\cdot\end{pmatrix},\quad \begin{pmatrix}\cdot & \cdot\\\cdot & \cdot\end{pmatrix}\begin{pmatrix}\cdot\\\cdot\end{pmatrix},\quad \begin{pmatrix}\cdot & \cdot\\\cdot & \cdot\end{pmatrix}\begin{pmatrix}\cdot & \cdot\\\cdot & \cdot\end{pmatrix}$$

- LAPACK: systèmes linéaires $\mathbf{AX} = \mathbf{B}$, moindres carrés, val. pr.

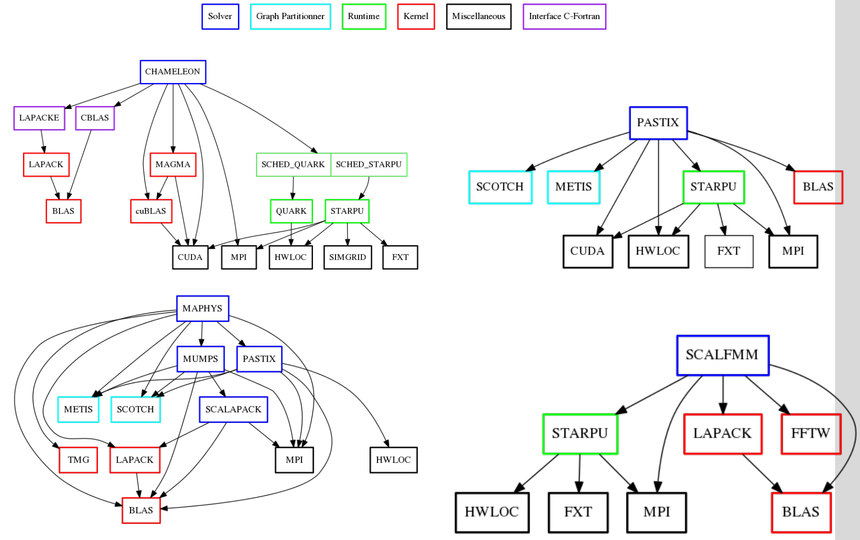
PaStiX: solveur direct matrices creuses

- systèmes linéaires $\mathbf{AX} = \mathbf{B}$, par factorisation $\mathbf{LL^T}, \mathbf{LDL^T}, \mathbf{LU}$

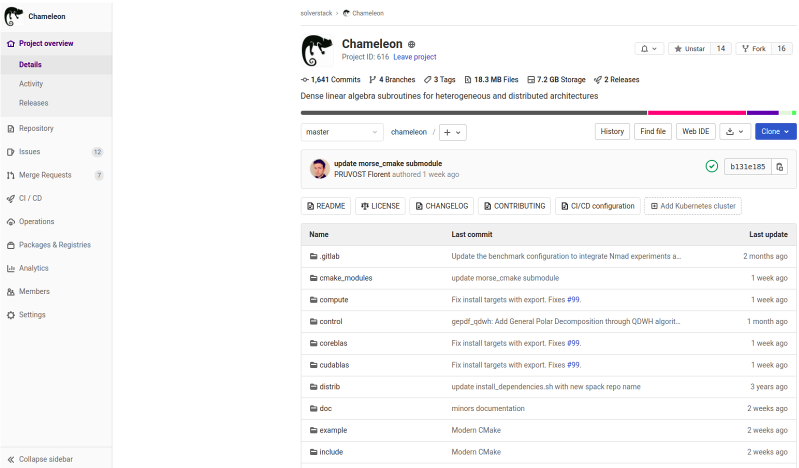
MaPHyS: solveur hybride direct/iteratif creux

- systèmes linéaires $\mathbf{AX} = \mathbf{B}$, méthodes CG/GMRES + precond.
- solveurs directs: MUMPS, PaStiX

Un environnement logiciel très modulaire



Intégration continue Gitlab



Chameleon

Project ID: 616 Leave project

1,641 Commits 4 Branches 3 Tags 18.3 MB Files 7.2 GB Storage 2 Releases

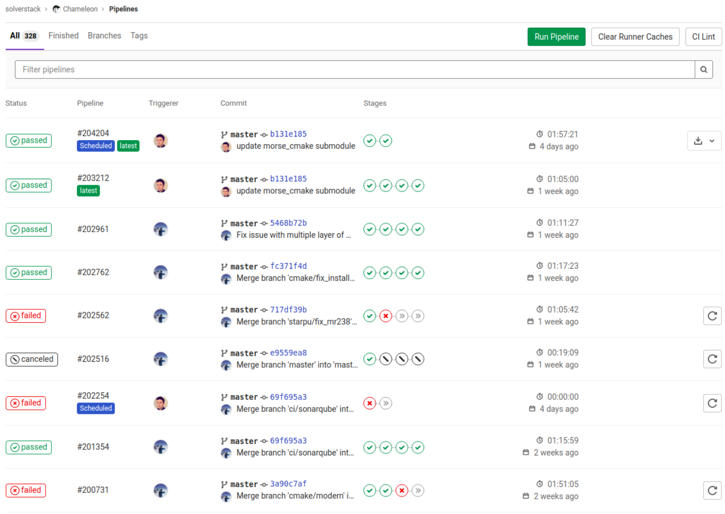
Dense linear algebra subroutines for heterogeneous and distributed architectures

update morse_cmake submodule
PRUVOST Florent authored 1 week ago ✓ b131e185

Name	Last commit	Last update
.gitlab	Update the benchmark configuration to integrate Nmad experiments a...	2 months ago
cmake_modules	update morse_cmake submodule	1 week ago
compute	Fix install targets with export. Fixes #99.	1 week ago
control	gepdf_qdwh: Add General Polar Decomposition through QDWH algorit...	1 month ago
coreblas	Fix install targets with export. Fixes #99.	1 week ago
cudablas	Fix install targets with export. Fixes #99.	1 week ago
distrib	update install_dependencies.sh with new spack repo name	3 years ago
doc	minors documentation	2 weeks ago
example	Modern CMake	2 weeks ago
include	Modern CMake	2 weeks ago

Hébergé sur https://gitlab.inria.fr

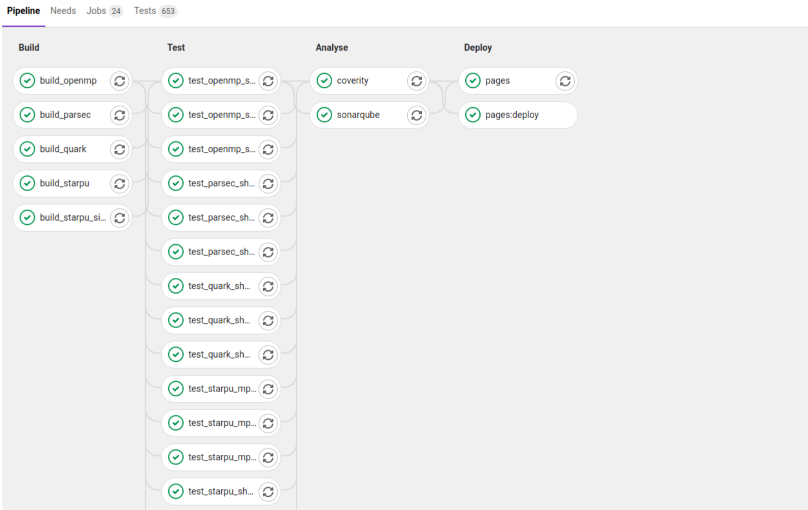
Intégration continue Gitlab



Status	Pipeline	Triggerer	Commit	Stages	
✓ passed	#204204 Scheduled latest		master b131e185 update morse_cmake submodule	✓ ✓	01:57:21 4 days ago
✓ passed	#203212 latest		master b131e185 update morse_cmake submodule	✓ ✓ ✓ ✓	01:05:00 1 week ago
✓ passed	#202961		master 5468b72b Fix issue with multiple layer of ...	✓ ✓ ✓ ✓	01:11:27 1 week ago
✓ passed	#202762		master fc371f4d Merge branch 'cmake/fix_install...	✓ ✓ ✓ ✓	01:17:23 1 week ago
✗ failed	#202562		master 717df39b Merge branch 'starpu/fix_mr238'...	✓ ✗ » »	01:05:42 1 week ago
⊘ canceled	#202516		master e9559ea8 Merge branch 'master' into 'mast...	✓ ⊘ ⊘ ⊘	00:19:09 1 week ago
✗ failed	#202254 Scheduled		master 69f695a3 Merge branch 'ci/sonarqube' int...	✗ »	00:00:00 4 days ago
✓ passed	#201354		master 69f695a3 Merge branch 'ci/sonarqube' int...	✓ ✓ ✓ ✓	01:15:59 2 weeks ago
✗ failed	#200731		master 3a90c7af Merge branch 'cmake/modern' i...	✓ ✓ ✗ »	01:51:05 2 weeks ago

Pipelines sur la branche *master*

Intégration continue Gitlab



Pipeline Needs Jobs 24 Tests 653

Build
- ✓ build_openmp
- ✓ build_parsec
- ✓ build_quark
- ✓ build_starpu
- ✓ build_starpu_si...

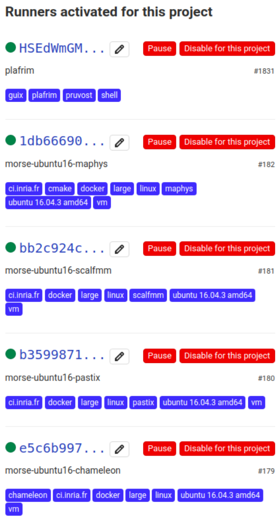
Test
- ✓ test_openmp_s...
- ✓ test_openmp_s...
- ✓ test_openmp_s...
- ✓ test_parsec_sh...
- ✓ test_parsec_sh...
- ✓ test_parsec_sh...
- ✓ test_quark_sh...
- ✓ test_quark_sh...
- ✓ test_quark_sh...
- ✓ test_starpu_mp...
- ✓ test_starpu_mp...
- ✓ test_starpu_mp...
- ✓ test_starpu_sh...

Analyse
- ✓ coverity
- ✓ sonarqube

Deploy
- ✓ pages
- ✓ pages:deploy

Chaque push sur master déclenche des jobs de build, tests, etc

Intégration continue Gitlab



Runners activated for this project

● HSEdWmGM... Pause Disable for this project

plafrim #1831

guix plafrim pruvost shell

● 1db66690... Pause Disable for this project

morse-ubuntu16-maphys #182

ci.inria.fr cmake docker large linux maphys ubuntu 16.04.3 amd64 vm

● bb2c924c... Pause Disable for this project

morse-ubuntu16-scalfmm #181

ci.inria.fr docker large linux scalfmm ubuntu 16.04.3 amd64 vm

● b3599871... Pause Disable for this project

morse-ubuntu16-pastix #180

ci.inria.fr docker large linux pastix ubuntu 16.04.3 amd64 vm

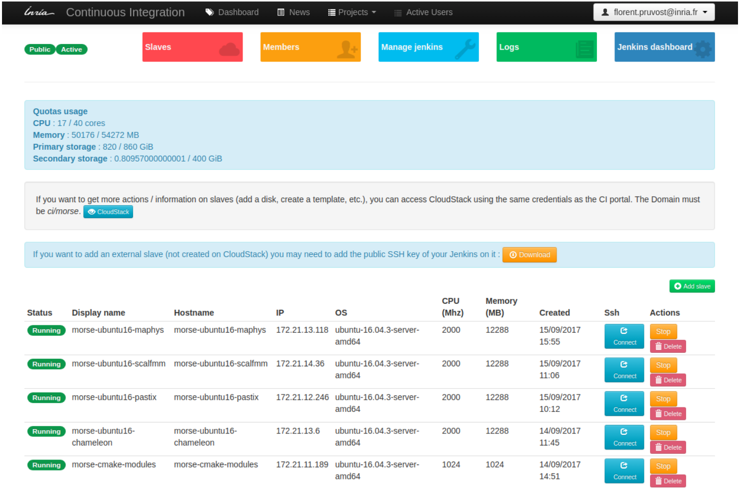
● e5c6b997... Pause Disable for this project

morse-ubuntu16-chameleon #179

chameleon ci.inria.fr docker large linux ubuntu 16.04.3 amd64 vm

Les runners sont des machines virtuelles avec docker

Intégration continue Gitlab



Quotas usage
CPU : 17 / 40 cores
Memory : 50176 / 54272 MB
Primary storage : 820 / 860 GiB
Secondary storage : 0.80957000000001 / 400 GiB

If you want to get more actions / information on slaves (add a disk, create a template, etc.), you can access CloudStack using the same credentials as the CI portal. The Domain must be *ci/morse*. CloudStack

If you want to add an external slave (not created on CloudStack) you may need to add the public SSH key of your Jenkins on it : Download

Add slave

Status	Display name	Hostname	IP	OS	CPU (Mhz)	Memory (MB)	Created	Ssh	Actions
Running	morse-ubuntu16-maphys	morse-ubuntu16-maphys	172.21.13.118	ubuntu-16.04.3-server-amd64	2000	12288	15/09/2017 15:55	Connect	Stop Delete
Running	morse-ubuntu16-scalfmm	morse-ubuntu16-scalfmm	172.21.14.36	ubuntu-16.04.3-server-amd64	2000	12288	15/09/2017 11:06	Connect	Stop Delete
Running	morse-ubuntu16-pastix	morse-ubuntu16-pastix	172.21.12.246	ubuntu-16.04.3-server-amd64	2000	12288	15/09/2017 10:12	Connect	Stop Delete
Running	morse-ubuntu16-chameleon	morse-ubuntu16-chameleon	172.21.13.6	ubuntu-16.04.3-server-amd64	2000	12288	14/09/2017 11:45	Connect	Stop Delete
Running	morse-cmake-modules	morse-cmake-modules	172.21.11.189	ubuntu-16.04.3-server-amd64	1024	1024	14/09/2017 14:51	Connect	Stop Delete

Gestion des VMs sur `https://ci.inria.fr`

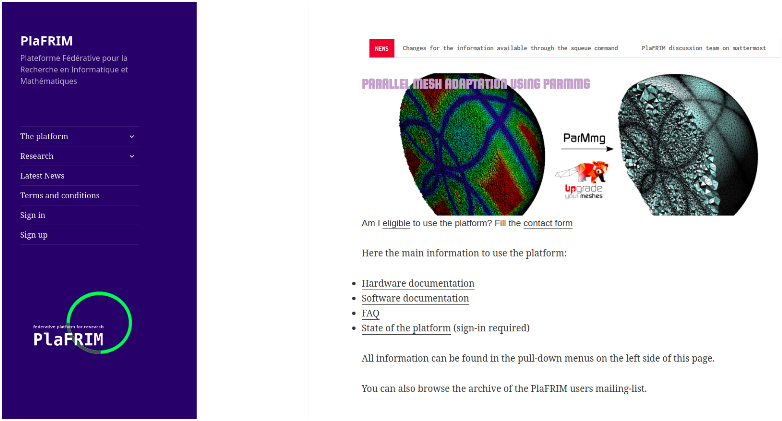
VMs : pas adaptées au HPC

- peu puissantes
- choix en architectures limité
- pas de cartes GPUs
- pas de réseau d'interconnexion rapide (tests MPI)

02

Runner PlaFRIM

Calculateur Bordelais PlaFRIM



PlaFRIM

Plateforme Fédérative pour la Recherche en Informatique et Mathématiques

- The platform
- Research
- Latest News
- Terms and conditions
- Sign in
- Sign up

federative platform for research
PlaFRIM

NEWS Changes for the information available through the squeue command PlaFRIM discussion team on mattermost

Am I eligible to use the platform? Fill the contact form

Here the main information to use the platform:

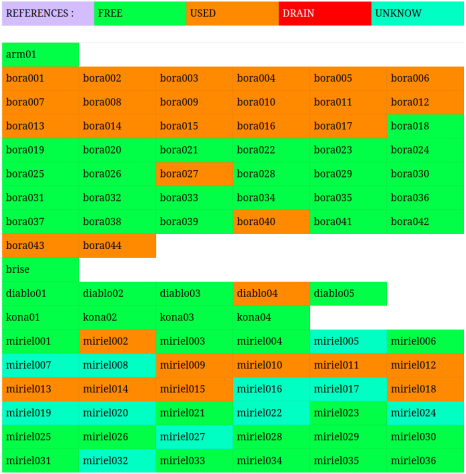
- Hardware documentation
- Software documentation
- FAQ
- State of the platform (sign-in required)

All information can be found in the pull-down menus on the left side of this page.

You can also browse the archive of the PlaFRIM users mailing-list.

Grappe de calcul expérimentale www.plafrim.fr

Calculateur Bordelais PlaFRIM



REFERENCES :	🟩 FREE	🟧 USED	🟥 DRAIN	🟦 UNKNOW

🟩 arm01					
🟧 bora001	🟧 bora002	🟧 bora003	🟧 bora004	🟧 bora005	🟧 bora006
🟧 bora007	🟧 bora008	🟧 bora009	🟧 bora010	🟧 bora011	🟧 bora012
🟧 bora013	🟧 bora014	🟧 bora015	🟧 bora016	🟧 bora017	🟩 bora018
🟩 bora019	🟩 bora020	🟩 bora021	🟩 bora022	🟩 bora023	🟩 bora024
🟩 bora025	🟩 bora026	🟧 bora027	🟩 bora028	🟩 bora029	🟩 bora030
🟩 bora031	🟩 bora032	🟩 bora033	🟩 bora034	🟩 bora035	🟩 bora036
🟩 bora037	🟩 bora038	🟩 bora039	🟧 bora040	🟩 bora041	🟩 bora042
🟧 bora043	🟧 bora044				
🟩 brise					
🟩 diablo01	🟩 diablo02	🟩 diablo03	🟧 diablo04	🟩 diablo05	
🟩 kona01	🟩 kona02	🟩 kona03	🟩 kona04		
🟩 miriel001	🟧 miriel002	🟩 miriel003	🟩 miriel004	🟦 miriel005	🟩 miriel006
🟦 miriel007	🟦 miriel008	🟧 miriel009	🟧 miriel010	🟧 miriel011	🟧 miriel012
🟧 miriel013	🟧 miriel014	🟧 miriel015	🟦 miriel016	🟦 miriel017	🟧 miriel018
🟦 miriel019	🟦 miriel020	🟩 miriel021	🟦 miriel022	🟩 miriel023	🟦 miriel024
🟩 miriel025	🟩 miriel026	🟦 miriel027	🟩 miriel028	🟩 miriel029	🟩 miriel030
🟩 miriel031	🟦 miriel032	🟩 miriel033	🟩 miriel034	🟩 miriel035	🟩 miriel036

Large éventails de machines: Intel, AMD, Arm, réseaux IB, GPUs Nvidia, Bigmem

Calculateur Bordelais PlaFRIM

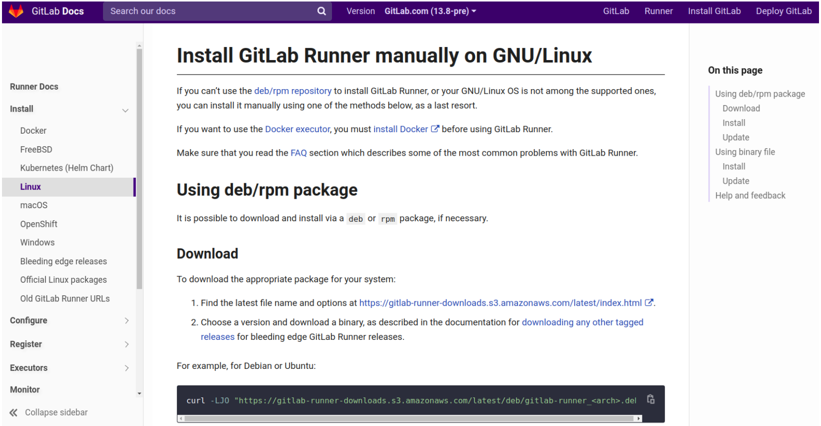
Accès illimité ! Pas de quotas

(Inria, la recherche locale et des PME régionales)

Accès internet pour certains noms de domaines

GNU Guix installé

Installation gitlab-runner sur PlaFRIM



Install GitLab Runner manually on GNU/Linux

If you can't use the deb/rpm repository to install GitLab Runner, or your GNU/Linux OS is not among the supported ones, you can install it manually using one of the methods below, as a last resort.

If you want to use the Docker executor, you must install Docker before using GitLab Runner.

Make sure that you read the FAQ section which describes some of the most common problems with GitLab Runner.

Using deb/rpm package

It is possible to download and install via a `deb` or `rpm` package, if necessary.

Download

To download the appropriate package for your system:

1. Find the latest file name and options at https://gitlab-runner-downloads.s3.amazonaws.com/latest/index.html.
2. Choose a version and download a binary, as described in the documentation for downloading any other tagged releases for bleeding edge GitLab Runner releases.

For example, for Debian or Ubuntu:

```
curl -LJO "https://gitlab-runner-downloads.s3.amazonaws.com/latest/deb/gitlab-runner_<arch>.del
```

gitlab-runner : programme exécutable permettant d'exposer la machine an tant que “runner” disponible pour les jobs de gitlab-ci

Installation gitlab-runner sur PlaFRIM

1. Téléchargement de l'éxécutable dans le HOME de sa session PlaFRIM

```
$ curl -L --output ~/gitlab-runner \
  "https://gitlab-runner-downloads.s3.amazonaws.com/ \
   latest/binaries/gitlab-runner-linux-amd64"
$ chmod +x ~/gitlab-runner
```

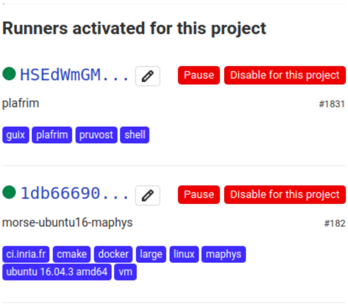
2. Donner l'accès au projet gitlab, droits pour télécharger le git, téléverser des logs/artéfacts vers gitlab (shell executor)

```
$ ~/gitlab-runner register
```

3. Lancement du programme qui communique en continue avec gitlab (dans un screen ou tmux pour qu'il reste en vie hors connexion)

```
$ ~/gitlab-runner run
```

Installation gitlab-runner sur PlaFRIM



Runners activated for this project

HSEdWmGM... Pause | Disable for this project

plafrim #1831

guix plafrim pruvost shell

1db66690... Pause | Disable for this project

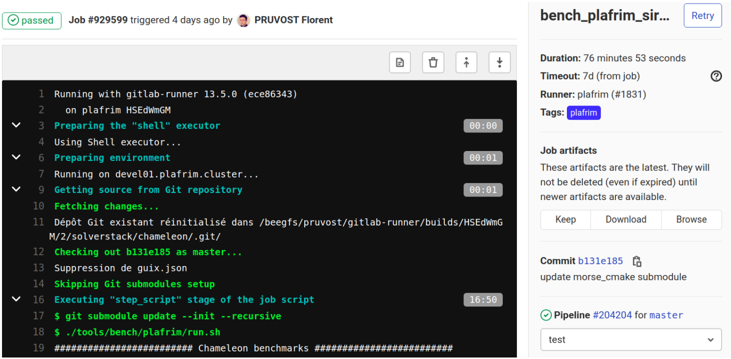
morse-ubuntu16-maphys #182

ci.inria.fr cmake docker large linux maphys ubuntu 16.04.3 amd64 vm

Runner plafrim à l'écoute prêt à en découdre

tag 'plafrim' pour affecter nos jobs sur ce runner

Installation gitlab-runner sur PlaFRIM



Job #929599 triggered 4 days ago by PRUVOST Florent

```
Running with gitlab-runner 13.5.0 (ece86343)
  on plafrim HSEdWmGM
Preparing the "shell" executor                                          00:00
Using Shell executor...
Preparing environment                                                   00:01
Running on devel01.plafrim.cluster...
Getting source from Git repository                                      00:01
Fetching changes...
Dépôt Git existant réinitialisé dans /beegfs/pruvost/gitlab-runner/builds/HSEdWmG
M/2/solverstack/chameleon/.git/
Checking out b131e185 as master...
Suppression de guix.json
Skipping Git submodules setup
Executing "step_script" stage of the job script                         16:50
$ git submodule update --init --recursive
$ ./tools/bench/plafrim/run.sh
######################## Chameleon benchmarks ########################
```

bench_plafrim_sir...

Duration: 76 minutes 53 seconds
Timeout: 7d (from job)
Runner: plafrim (#1831)
Tags: plafrim

Job artifacts

These artifacts are the latest. They will not be deleted (even if expired) until newer artifacts are available.

Commit b131e185
update morse_cmake submodule

Pipeline #204204 for master

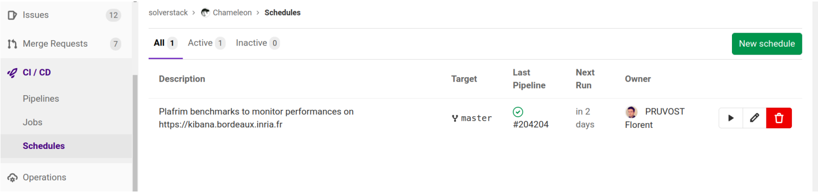
test

Exemple de log lorsqu'un job s'exécute

03

Pipeline gitlab-ci

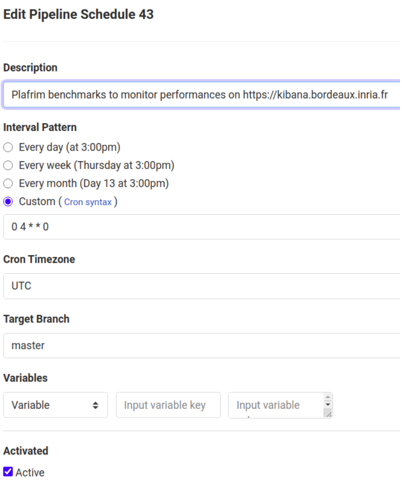
Le schedule du dimanche



Description	Target	Last Pipeline	Next Run	Owner
Plafrim benchmarks to monitor performances on https://kibana.bordeaux.inria.fr	master	✓ #204204	in 2 days	PRUVOST Florent

Pipeline spécial de type "schedule" (cron)

Le schedule du dimanche



Edit Pipeline Schedule 43

Description

Plafrim benchmarks to monitor performances on https://kibana.bordeaux.inria.fr

Interval Pattern

○ Every day (at 3:00pm)
○ Every week (Thursday at 3:00pm)
○ Every month (Day 13 at 3:00pm)
● Custom (Cron syntax)

0 4 * * 0

Cron Timezone

UTC

Target Branch

master

Variables

Activated

☑ Active

Lancé une fois par semaine le dimanche

Jobs gitlab-ci

```
.bench_plafrim_common:
  only:
    - schedules
  stage: test
  tags: [’plafrim’]
  timeout: 1 week
  before_script:
    - git submodule update --init --recursive
  script:
    - ./tools/bench/plafrim/run.sh
  artifacts:
    name: "$CI_JOB_NAME"
    expire_in: 1 week
    paths:
      - "chameleon-$NODE-$MPI-$SLURM_NP.err"
      - "chameleon-$NODE-$MPI-$SLURM_NP.out"
      - "tools/bench/plafrim/chameleon.csv"
      - "tools/bench/plafrim/results/$JUBE_ID"
  variables:
  PLATFORM: plafrim
```

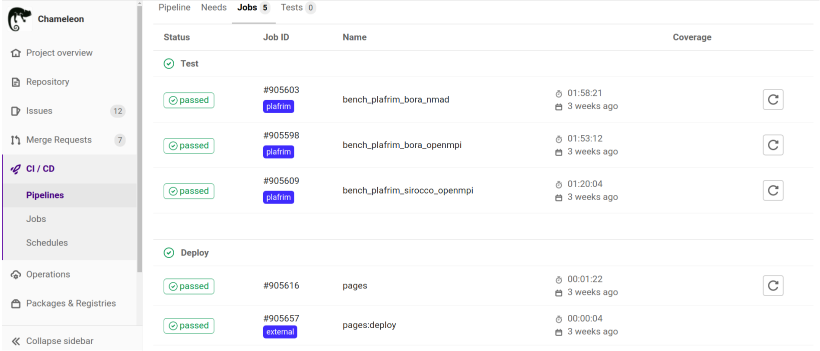
Jobs gitlab-ci

```
bench_plafrim_bora_openmpi:
  variables:
    NODE: bora
    MPI: openmpi
    SLURM_NP: 9
    JUBE_ID: "000001"
  extends: .bench_plafrim_common

bench_plafrim_bora_nmad:
  variables:
    NODE: bora
    MPI: nmad
    SLURM_NP: 9
    JUBE_ID: "000002"
  extends: .bench_plafrim_common

bench_plafrim_sirocco_openmpi:
  variables:
    NODE: sirocco
    MPI: openmpi
    SLURM_NP: 1
    JUBE_ID: "000003"
  extends: .bench_plafrim_common
```

Jobs gitlab-ci



Status	Job ID	Name		Coverage
Test				
passed	#905603 plafrim	bench_plafrim_bora_nmad	01:58:21 3 weeks ago	
passed	#905598 plafrim	bench_plafrim_bora_openmpi	01:53:12 3 weeks ago	
passed	#905609 plafrim	bench_plafrim_sirocco_openmpi	01:20:04 3 weeks ago	
Deploy				
passed	#905616	pages	00:01:22 3 weeks ago	
passed	#905657 external	pages:deploy	00:00:04 3 weeks ago	

Statut des jobs du pipeline

Guix environment

Dans le script principal run.sh on construit l'environnement logiciel avec GNU Guix

```
exec guix environment --pure \
                      $GUIX_RULE \
                      -- /bin/bash --norc \
                      ./tools/bench/plafrim/slurm.sh
```

- Guix_RULE contient ce qu'il faut installer, *i.e.* les dépendances du solveur plus quelques outils de post-traitement et slurm
- Le script slurm.sh est exécuté à l'intérieur de l'environnement construit
- Cet environnement est isolé de celui du système
- Il est reproductible, on contrôle toute la chaîne logicielle : compilateurs, cmake, blas/lapack, mpi, python, slurm ...

Job Slurm

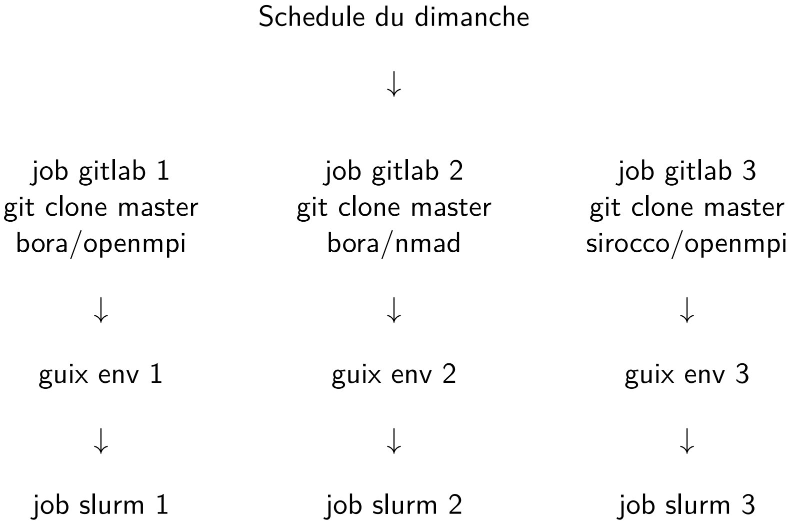
Dans le script slurm.sh on soumet notre job slurm

```
# Submit jobs
NJOB=0
JOB_ID=‘sbatch \
        --nodes=$NP --time=$TIME --partition=$PART \
        --constraint=$CONS --exclude=$EXCL --exclusive \
        --ntasks-per-node=1 --threads-per-core=1 \
        $CI_PROJECT_DIR/tools/bench/chameleon_guix.sh \
        | sed "s#Submitted batch job ##"‘
if [[ -n "$JOB_ID" ]]
then
    JOB_LIST="$JOB_LIST $JOB_ID"
    NJOB=$[NJOB+1]
fi

# Wait for completion of jobs
wait_completion
```

On attend la fin du job slurm pour terminer le job gitlab

Bilan workflow



Schedule du dimanche

↓

job gitlab 1
git clone master
bora/openmpi

↓

guix env 1

↓

job slurm 1

job gitlab 2
git clone master
bora/nmad

↓

guix env 2

↓

job slurm 2

job gitlab 3
git clone master
sirocco/openmpi

↓

guix env 3

↓

job slurm 3

1 job slurm = 1 benchmark

- 1 benchmark = série d'exécutions, différents paramètres
- Algorithme, taille des tuiles
- Précision arithmétique
- Nombre de noeuds (et CPUs/GPUs)
- Taille de matrice

Fonctionnalités Jube (python)

- Définition des plans d'expériences (xml)
- Pilotage des exécutions des différentes configs simplifiées
- Mode de reprise, si jobs en échecs
- Collecte des résultats (csv), parsing des logs
- `https://apps.fz-juelich.de/jsc/jube/jube2/docu`

Matrice de configuration paramétrée

algo=GEMM, nb=280

- prec = simple, double
- mpi = 1, 4, 9
- n = mpi*nb, 2*mpi*nb, 8*mpi*nb, ...

algo=POTRF, nb=320

- prec = simple, simple complexe
- mpi = 1, 3, 6
- n = mpi*nb, 2*mpi*nb, 8*mpi*nb, ...

algo=GEQRF, nb=480

- prec = double, double complexe
- mpi = 1, 2, 5
- n = mpi*nb, 2*mpi*nb, 8*mpi*nb, ...

Collecte des résultats

```
# Collecte des resultats (cputime, gflops)
jube result results/ --id $JUBE_ID > chameleon.csv

# Contenu de chameleon.csv
cat tools/bench/plafrim/chameleon.csv
hostname,algorithm,precision,nmpi,nthr,ngpu,m,n,k,cputime,gflops
bora,gemm,s,1,34,0,280,280,280,0.001915225,22.92368
bora,gemm,s,1,34,0,560,560,560,0.006108616,57.4978
bora,gemm,s,1,34,0,1120,1120,1120,0.002718295,1033.683
bora,gemm,s,1,34,0,2240,2240,2240,0.009067092,2479.168
bora,gemm,s,1,34,0,4480,4480,4480,0.05502595,3268.109
bora,gemm,s,1,34,0,8960,8960,8960,0.4372806,3289.984
bora,gemm,s,1,34,0,13440,13440,13440,1.551888,3128.725
bora,gemm,s,4,34,0,1120,1120,1120,0.01039234,270.3777
bora,gemm,s,4,34,0,2240,2240,2240,0.01441079,1559.862
bora,gemm,s,4,34,0,4480,4480,4480,0.03859214,4659.777
bora,gemm,s,4,34,0,8960,8960,8960,0.1565628,9188.94
bora,gemm,s,4,34,0,17920,17920,17920,1.206481,9539.452
bora,gemm,s,4,34,0,35840,35840,35840,7.808004,11792.18
bora,gemm,s,4,34,0,53760,53760,53760,25.56138,12156.92
bora,gemm,s,9,34,0,2520,2520,2520,0.01760951,1817.542
```

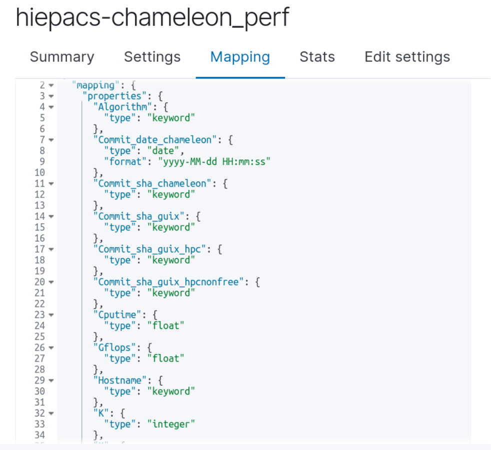
Mise à jour base de données Elasticsearch

- sauvegarde dans une base de donnée Elasticsearch
- https://www.elastic.co/fr/elastic-stack
- serveur Elastic commun Inria Bordeaux : https://elasticsearch.bordeaux.inria.fr
- script python, module existant *elasticsearch*

```
python3 $CI_PROJECT_DIR/tools/bench/jube/add_result.py \
-e https://elasticsearch.bordeaux.inria.fr \
-t hiepacs -p "chameleon" -h $NODE -m $MPI chameleon.csv
```

- ajout de paramètres qui influent sur les résultats
  - commit de guix, commit du solveur = environnement logiciel

Mise à jour base de données Elasticsearch



hiepacs-chameleon_perf

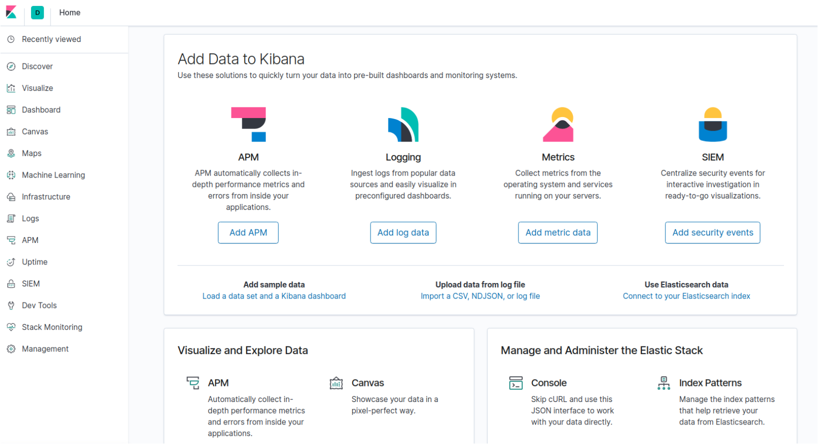
```
  "mapping": {
    "properties": {
      "Algorithm": {
        "type": "keyword"
      },
      "Commit_date_chameleon": {
        "type": "date",
        "format": "yyyy-MM-dd HH:mm:ss"
      },
      "Commit_sha_chameleon": {
        "type": "keyword"
      },
      "Commit_sha_guix": {
        "type": "keyword"
      },
      "Commit_sha_guix_hpc": {
        "type": "keyword"
      },
      "Commit_sha_guix_hpcnonfree": {
        "type": "keyword"
      },
      "Cputime": {
        "type": "float"
      },
      "Gflops": {
        "type": "float"
      },
      "Hostname": {
        "type": "keyword"
      },
      "K": {
        "type": "integer"
      },
```

Vue de l’index Elastic pour Chameleon

04

Traitement des données

Kibana+Elasticsearch



Add Data to Kibana

Use these solutions to quickly turn your data into pre-built dashboards and monitoring systems.

APM

APM automatically collects in-depth performance metrics and errors from inside your applications.

Add APM

Logging

Ingest logs from popular data sources and easily visualize in preconfigured dashboards.

Add log data

Metrics

Collect metrics from the operating system and services running on your servers.

Add metric data

SIEM

Centralize security events for interactive investigation in ready-to-go visualizations.

Add security events

Add sample data
Load a data set and a Kibana dashboard

Upload data from log file
Import a CSV, NDJSON, or log file

Use Elasticsearch data
Connect to your Elasticsearch index

Visualize and Explore Data

APM
Automatically collect in-depth performance metrics and errors from inside your applications.

Canvas
Showcase your data in a pixel-perfect way.

Manage and Administer the Elastic Stack

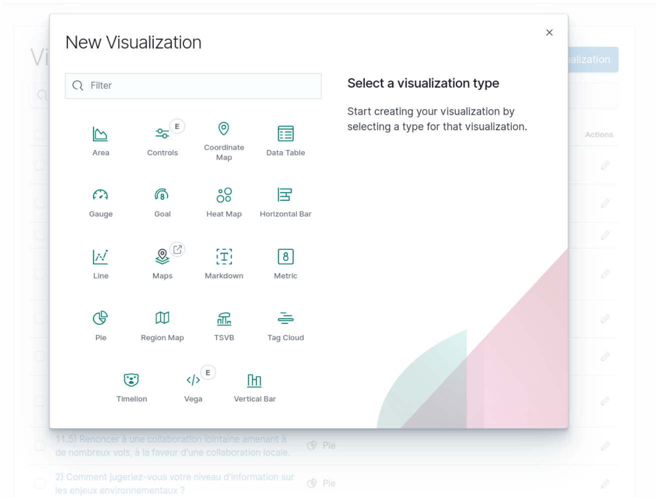
Console
Skip cURL and use this JSON interface to work with your data directly.

Index Patterns
Manage the index patterns that help retrieve your data from Elasticsearch.

Visualisation Kibana, dialogue avec Elastic

https://kibana.bordeaux.inria.fr

Kibana+Elasticsearch



New Visualization

Select a visualization type

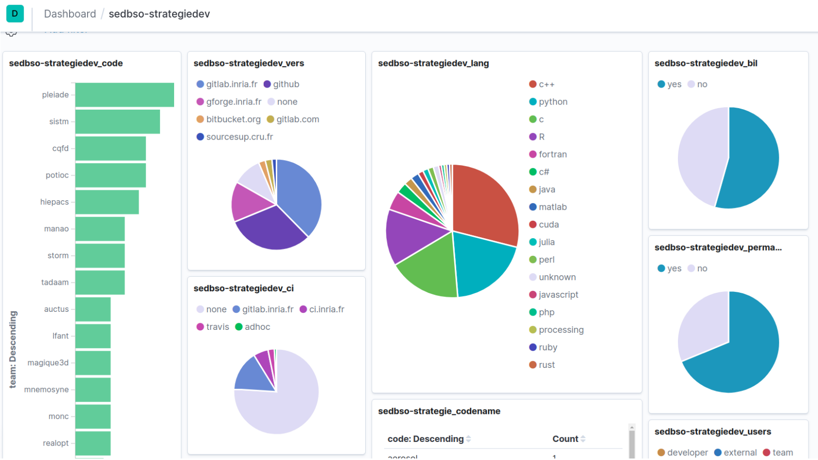
Start creating your visualization by selecting a type for that visualization.

11.5) Renoncer à une collaboration lointaine amenant à de nombreux vols, à la faveur d'une collaboration locale. Pie

2) Comment jugeriez-vous votre niveau d'information sur les enjeux environnementaux ? Pie

Large choix de représentations

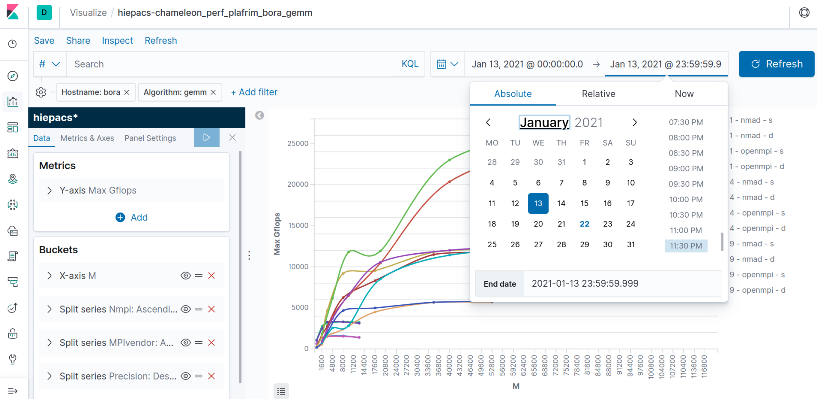
Kibana+Elasticsearch



code: Descending	Count
aerosol	1

Tableau de bord

Filtrage et opérations



Visualize / hiepacs-chameleon_perf_plafrim_bora_gemm

Search · KQL

Jan 13, 2021 @ 00:00:00.0 → Jan 13, 2021 @ 23:59:59.9

Hostname: bora ×
Algorithm: gemm ×

hiepacs*

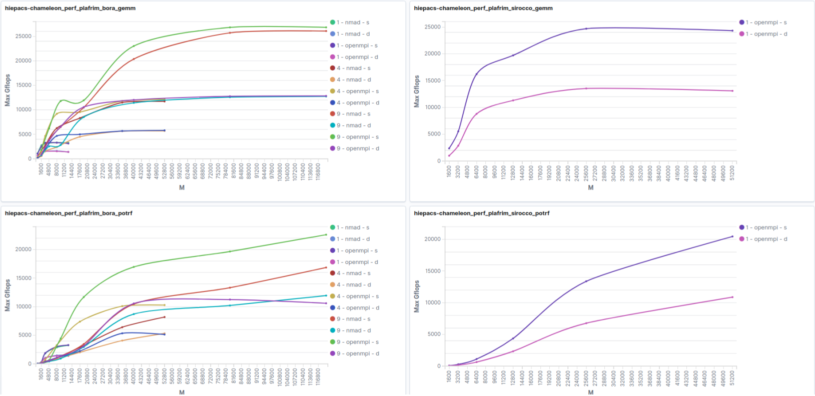
Metrics
- Y-axis Max Gflops

Buckets
- X-axis M
- Split series Nmpi: Ascendi...
- Split series MPIvendor: A...
- Split series Precision: Des...

End date: 2021-01-13 23:59:59.999

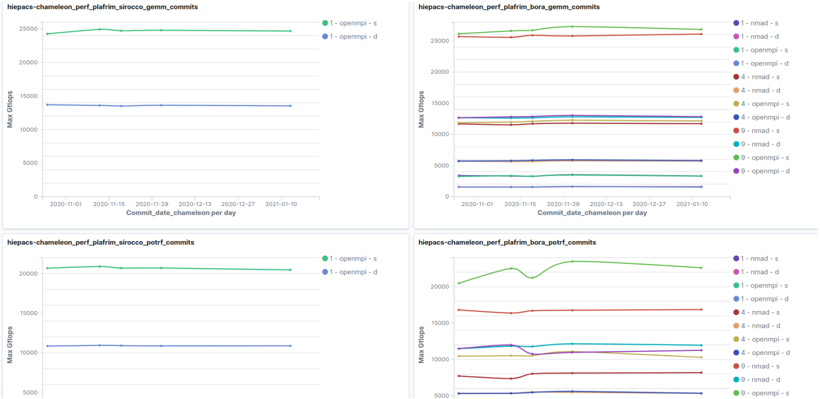
Intervalle de dates, fixer certains paramètres, méthodes d'agrégation (Max, Min, Average, etc)

Performances solveurs



Performances à commit fixé

Performances solveurs



Performances en fonction de la date de commit

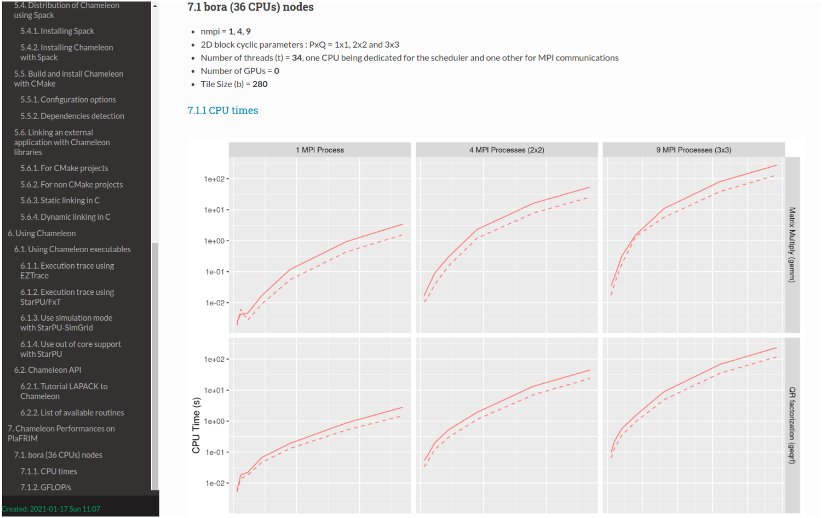
Affichage ailleurs plus tard

- récupération des données Elastic, par exemple pour la documentation du solveur
- ex. paramètre d'entrée : dernier commit
- via script python

```
# generate the csv file from elasticsearch
# for the given chameleon commit
python3 tools/bench/jube/get_result.py \
-e https://elasticsearch.bordeaux.inria.fr \
-t hiepacs -p chameleon -c $commit_sha

# generate the figures
Rscript tools/bench/jube/GenFigures.R
```

Affichage ailleurs plus tard



Created: 2021-01-17 Sun 11:07

7.1 bora (36 CPUs) nodes

- nmpi = **1, 4, 9**
- 2D block cyclic parameters : PxQ = 1x1, 2x2 and 3x3
- Number of threads (t) = **34**, one CPU being dedicated for the scheduler and one other for MPI communications
- Number of GPUs = **0**
- Tile Size (b) = **280**

7.1.1 CPU times

Performances affichées dans la documentation

05

Conclusion

Ce qu’on a

- des tests de performances automatisés → gitlab-ci
- sur une “vraie” machine de calcul → PlaFRIM
- un environnement logiciel isolé et reproductible → guix
- un suivi des performances dans le temps → Kibana+Elastic
- une analyse multiparamétrique des performances

Peu de codage, outils prêts sur l’étagère !

Perspectives

Détection automatique des régressions

- fonctionnalité **Watcher email action** existe bien dans la suite Elastic mais dans la version payante !
- voir outil Airbug https://github.com/jm-cc/gcvb

gitlab-runner partagé sur PlaFRIM ?

- factoriser la gestion des gitlab runner
- possible pour projets open-source (question des droits)
- questions d'administrateur système :
  - quel compte utilisateur ?
  - espace disque partagé entre les jobs → taille ?
  - combien de jobs possibles en parallèles ?

GitLab

elastic

federative platform for research
PlaFRIM

python™

Guix

JUBE
BENCHMARKING
ENVIRONMENT

slurm
workload manager

Merci pour votre attention !