
JINR
MLiT

Интеграция вычислительных ресурсов

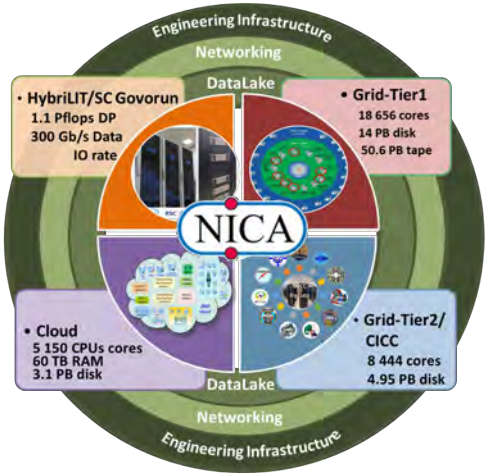
Научный сотрудник

Игорь Пелеванюк

Лаборатория
информационных технологий
им. М.Г. Мещерякова

Вычислительные ресурсы

MICC



NICA cluster

JINR Member States

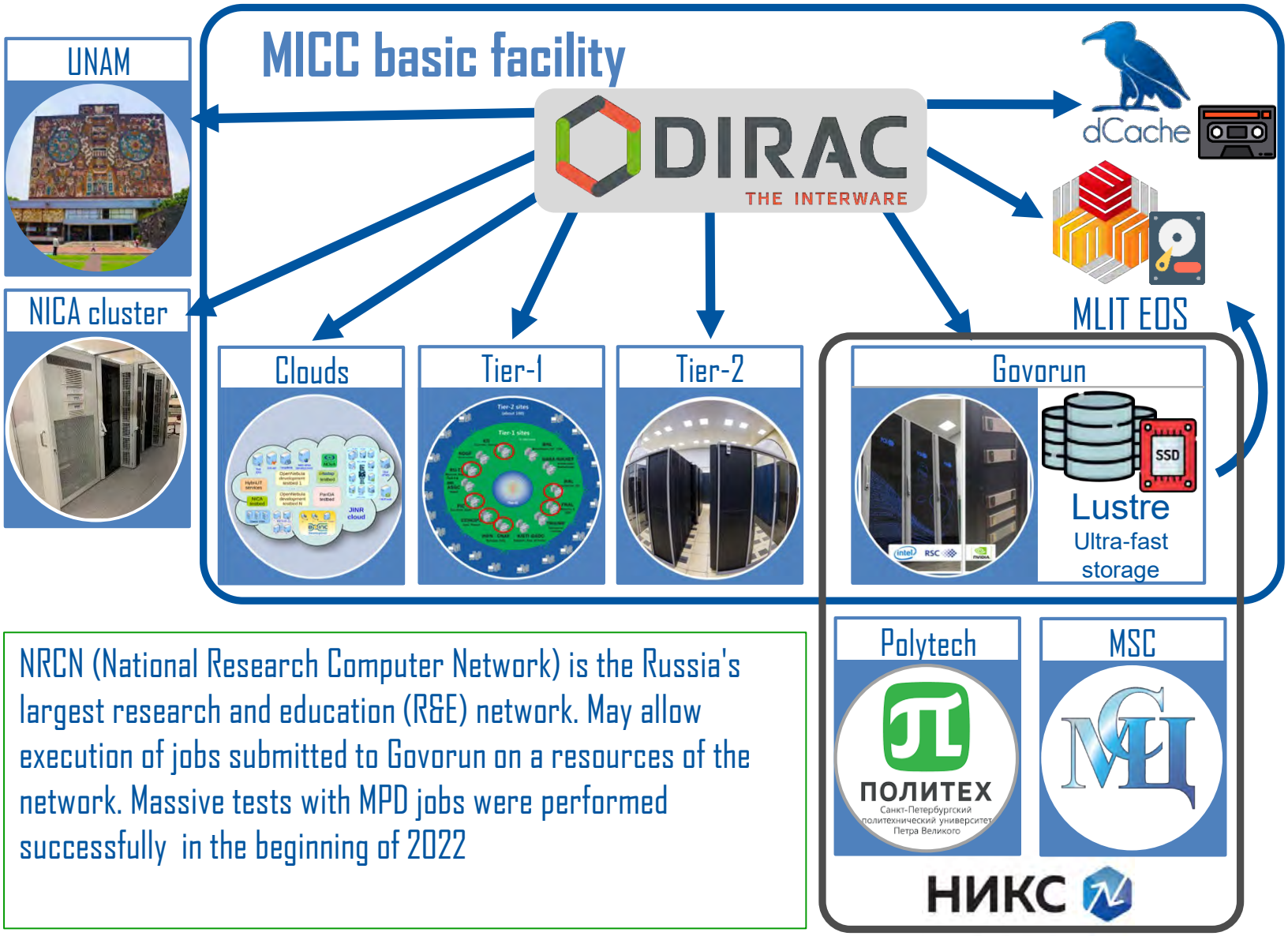
Others...

Зачем интегрировать ресурсы

Большое количество однотипных вычислительных задач - сотни тысяч задач, сотни CPU-лет в год.

1. Запуск таких объёмов на любом из доступных ресурсов приведёт к одному из двух исходов:
   - Задачи будут считаться слишком долго
   - Вычислительный ресурс будет недоступен для других пользователей
2. Ручной запуск на разных ресурсах превратится в большую проблему для ответственного:
   - Придётся эмпирически делить нагрузку между ресурсами
   - Входные данные придётся вручную распределять по разным ресурсам. Сбор результатов так же потребует дополнительных усилий.
   - Вопросы использования новых появившихся ресурсов – это на 100% проблема конечного пользователя. Ему придётся быть в курсе изменений на всех используемых ресурсах.
   - В случае появления проблем на ресурсах или обнаружения ошибок в запускаемых задачах пользователю потребуется перезапускать или перенаправлять нагрузку в «ручном режиме»

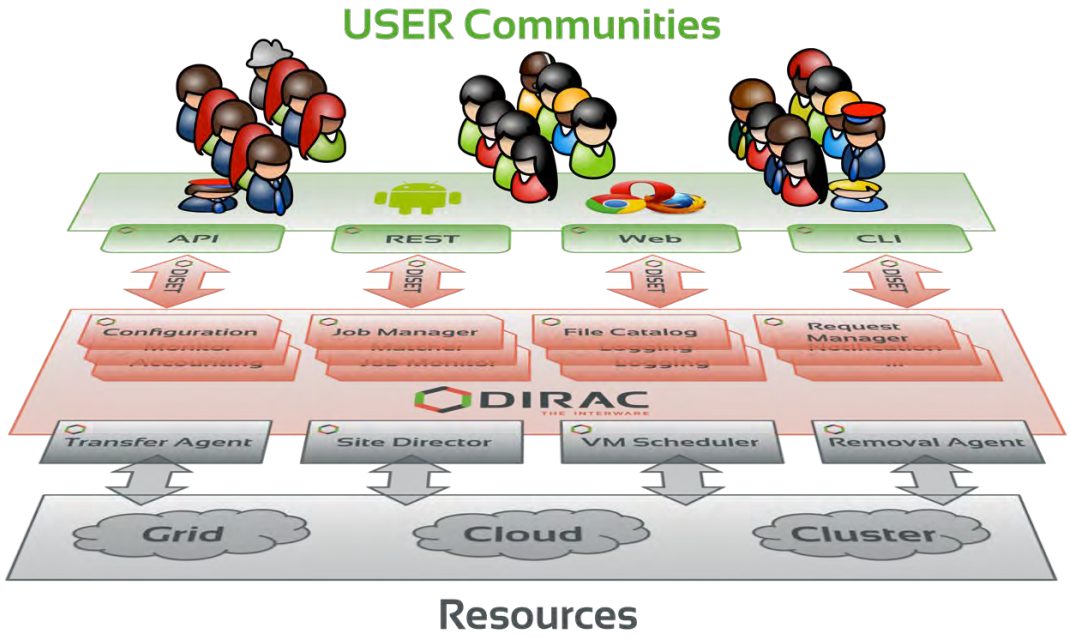
Что было сделано



NRCN (National Research Computer Network) is the Russia's largest research and education (R&E) network. May allow execution of jobs submitted to Govorun on a resources of the network. Massive tests with MPD jobs were performed successfully in the beginning of 2022

What is DIRAC?

DIRAC provides all the necessary components to build ad-hoc grid infrastructures **interconnecting** computing resources of different types, allowing **interoperability** and simplifying **interfaces**. This allows to speak about the DIRAC *interware*.



- Web
- CLI
- API
- REST

Birth of DIRAC

- LHC experiments, all developed their own middleware
  - PanDA, AliEn, glideIn WMS, PhEDEx, …
- DIRAC is developed originally for the LHCb experiment
- The experience collected with a production grid system of a large HEP experiment is very valuable
  - Several new experiments expressed interest in using this software relying on its proven in practice utility
- In 2009 the core DIRAC development team decided to generalize the software to make it suitable for any user community.
- The results of this work allow to offer DIRAC as a general purpose distributed computing framework

Why DIRAC?

1. Single system for all aspects of computing

User Interface

API

Central configuration

Workload management

Data management

Integration tools

File Catalog

Workflow management

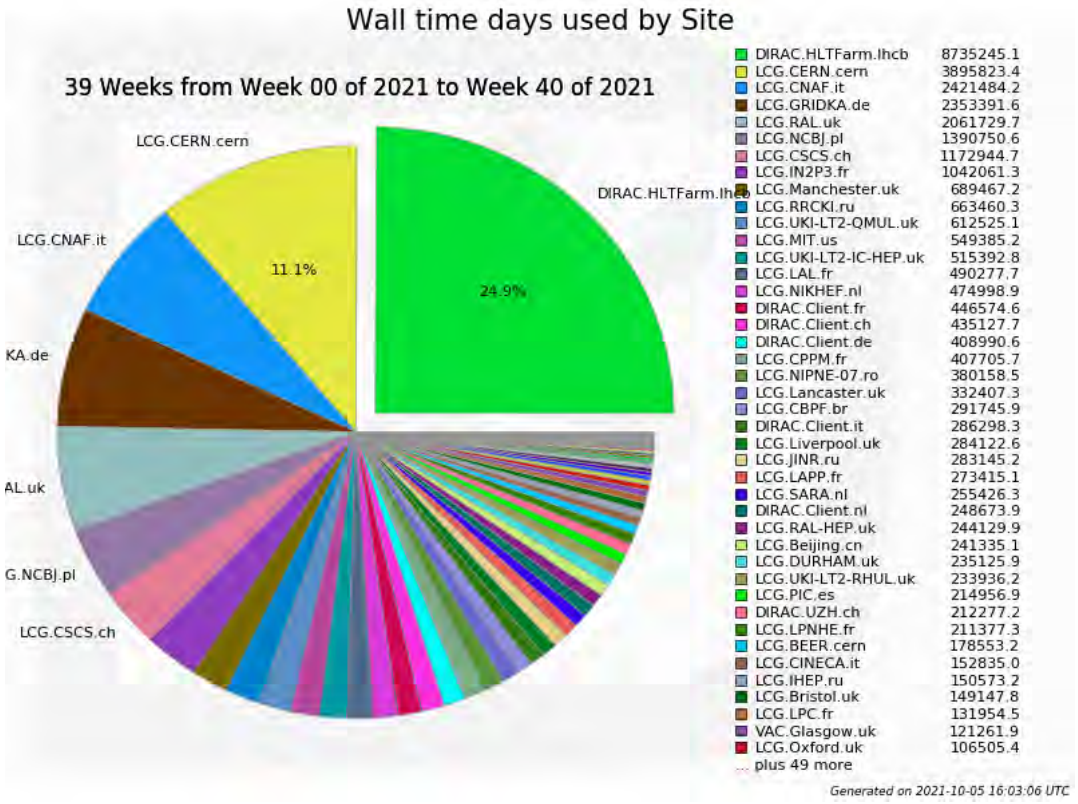
Metadata management

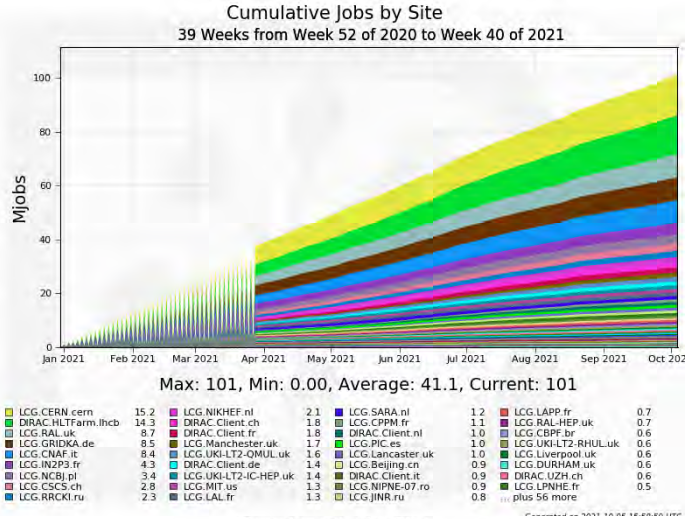
Accounting

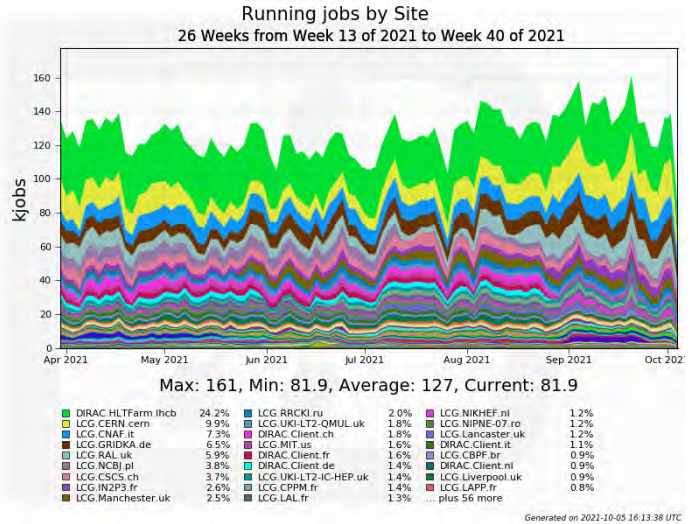
Management

Why DIRAC?

2. Good performance







Why DIRAC?

3. Active users and developers community

- Dedicated installations
  - LHCb, Belle II, CTA
- Multi-community services
  - ILC, CALICE
  - IHEP: BES III, Juno, CEPC
  - FG-DIRAC
  - GridPP
  - DIRAC4EGI
  - PNNL
  - DIRAC@JINR
  - DIRAC@CNAF
- Several DIRAC evaluations are ongoing
  - Auger, ELI, NICA, Virgo, LSST, …

DIRAC standard job workflow

1. Initial configuration
2. Input data download
3. Processing
4. Output data upload
5. Finalization

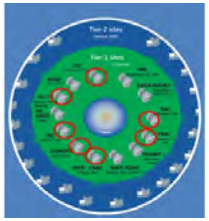
Job example

```
dirac-configure config.cfg

dirac-dms-get-file /bmn/raw/AuAu/data1002.raw

root –l –q –b reco.MC(“./data1002.raw”)

dirac-dms-put-file /bmn/reco/AuAu/data1002.root \\
                                    data1002.root \\
                                    JINR-EOS-BMN

rm –f data1002.raw data1002.root
```

Workload management

Submit thousand of jobs to DIRAC Job Queue

DIRAC
THE INTERWARE


Tier-1


CICC/Tier-2


Clouds

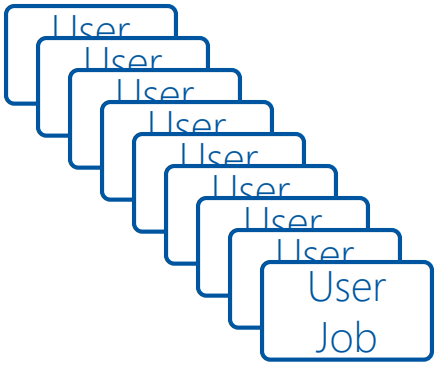

Govorun


NICA Cluster

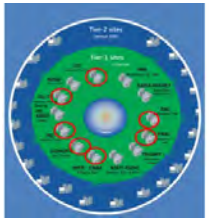

UNAM

Workload management

Submit thousand of jobs to DIRAC Job Queue



DIRAC
THE INTERWARE


Tier-1


CICC/Tier-2


Clouds

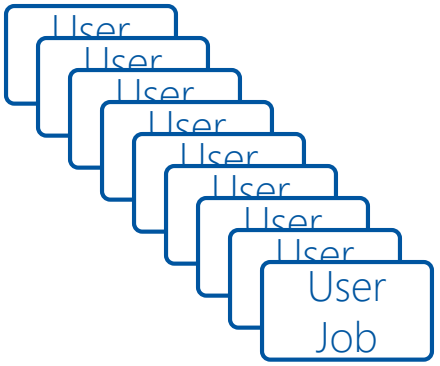

Govorun


NICA Cluster


UNAM

Workload management

Submit thousand of jobs to DIRAC Job Queue

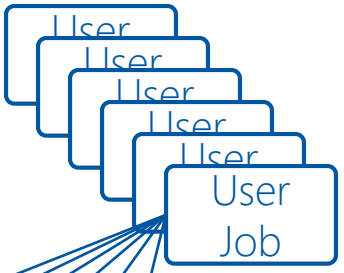


Pilot job | Pilot job | Pilot job | Pilot job | Pilot job | Pilot job

Tier-1 | CICC/Tier-2 | Clouds | Govorun | NICA Cluster | UNAM

Workload management

Submit thousand of jobs to DIRAC Job Queue



DIRAC
THE INTERWARE

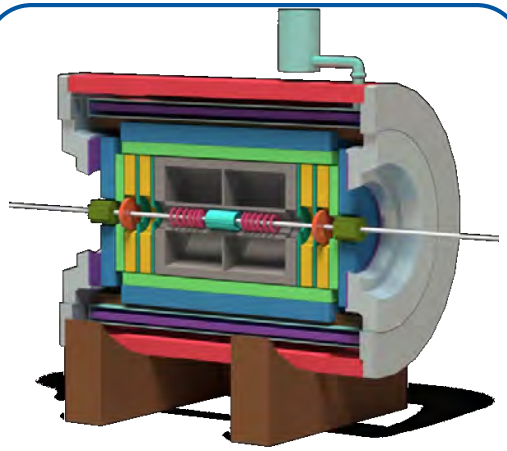
Tier-1 | CICC/Tier-2 | Clouds | Govorun | NICA Cluster | UNAM

History of DIRAC at JINR

2013 – Development of monitoring system for BES-III installation. First tries to setup and configure DIRAC infrastructure.

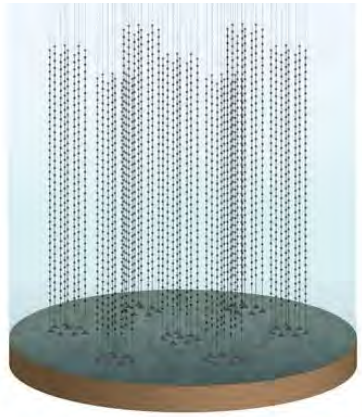
2017 – DIRAC Interware installed; basic configuration done. Used for educational purposes. dCache storage integrated, Tier2 integrated.
2018 – HybriLIT integrated. JINR cloud integrated using OCCI protocol. Tests of full cycle of Monte-Carlo for BM@N were performed.
2019 – Clouds of JINR Member-States integrated by module developed in JINR. MPD starts using DIRAC for massive Monte-Carlo production. Tier1, Govorun and EOS integrated in DIRAC.
2020 – Folding@Home jobs submitted to clouds via DIRAC. Baikal-GVD jobs submitted to JINR and PRUE clouds.
2021 – First tests for SPD Monte-Carlo successfully done. First million jobs done!
2022– Total walltime exceeds 1000 years. DIRAC in JINR updated to use Python 3.

What do we use DIRAC for?



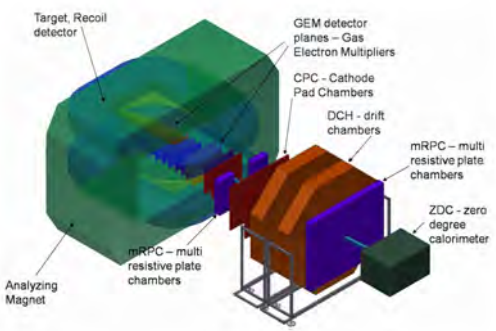
MPD@NICA

Monte-Carlo – Real
Analysis – Maybe



Baikal-GVD

Monte-Carlo – Real



BM@N

Monte-Carlo – Real
Reconstruction – Real

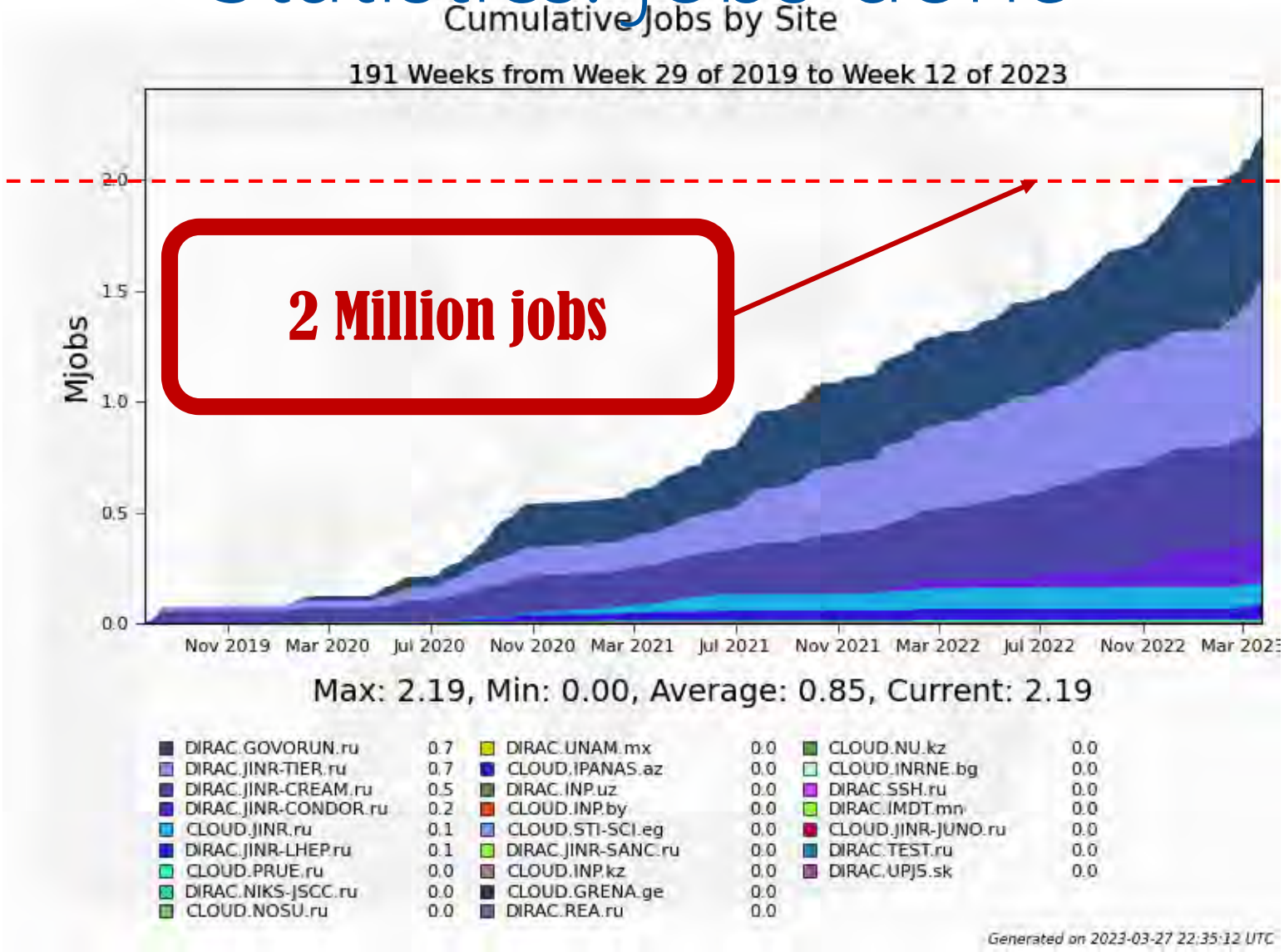
SPD@NICA

Monte-Carlo – Real

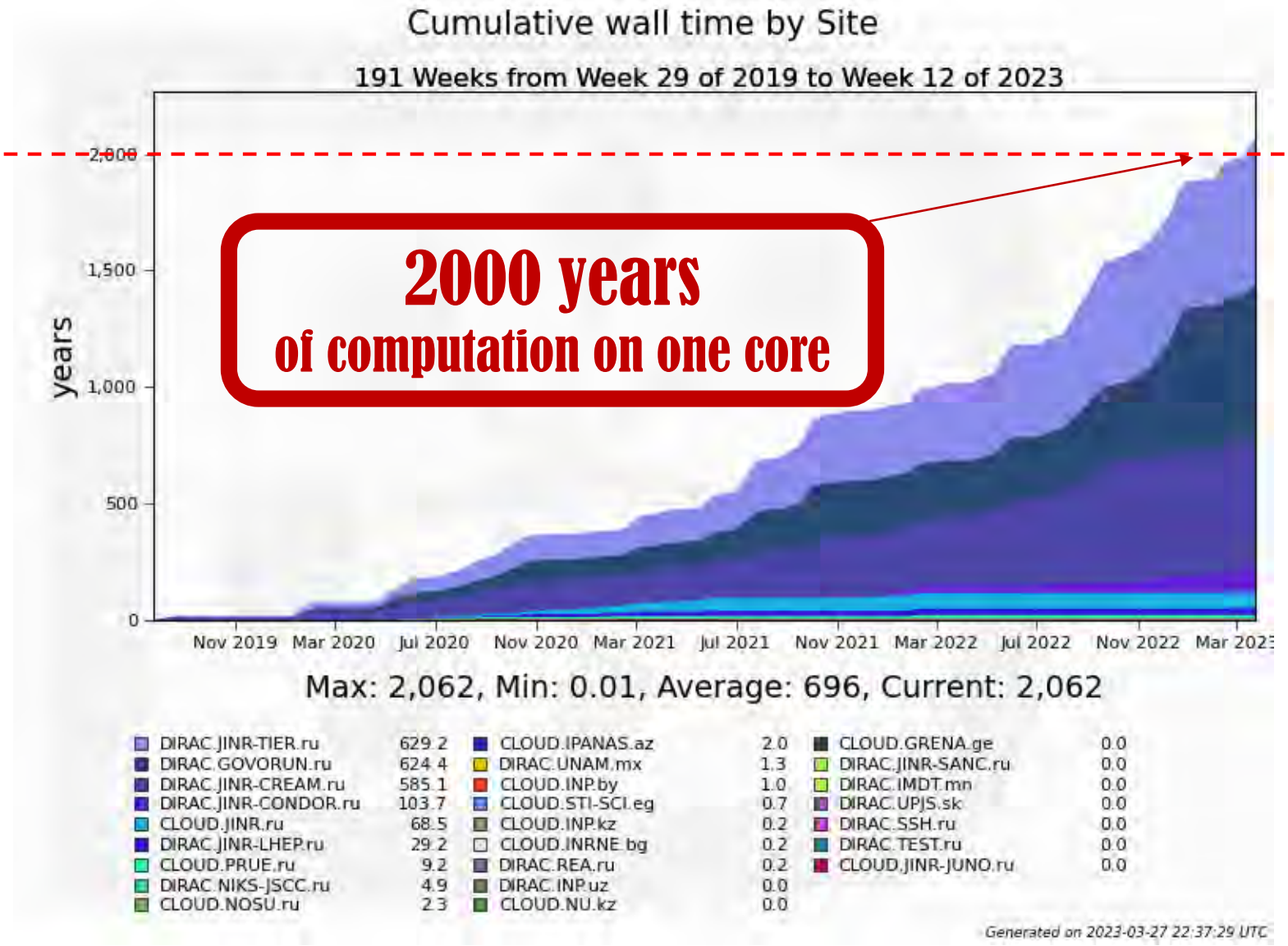
Folding@HOME

Teaching

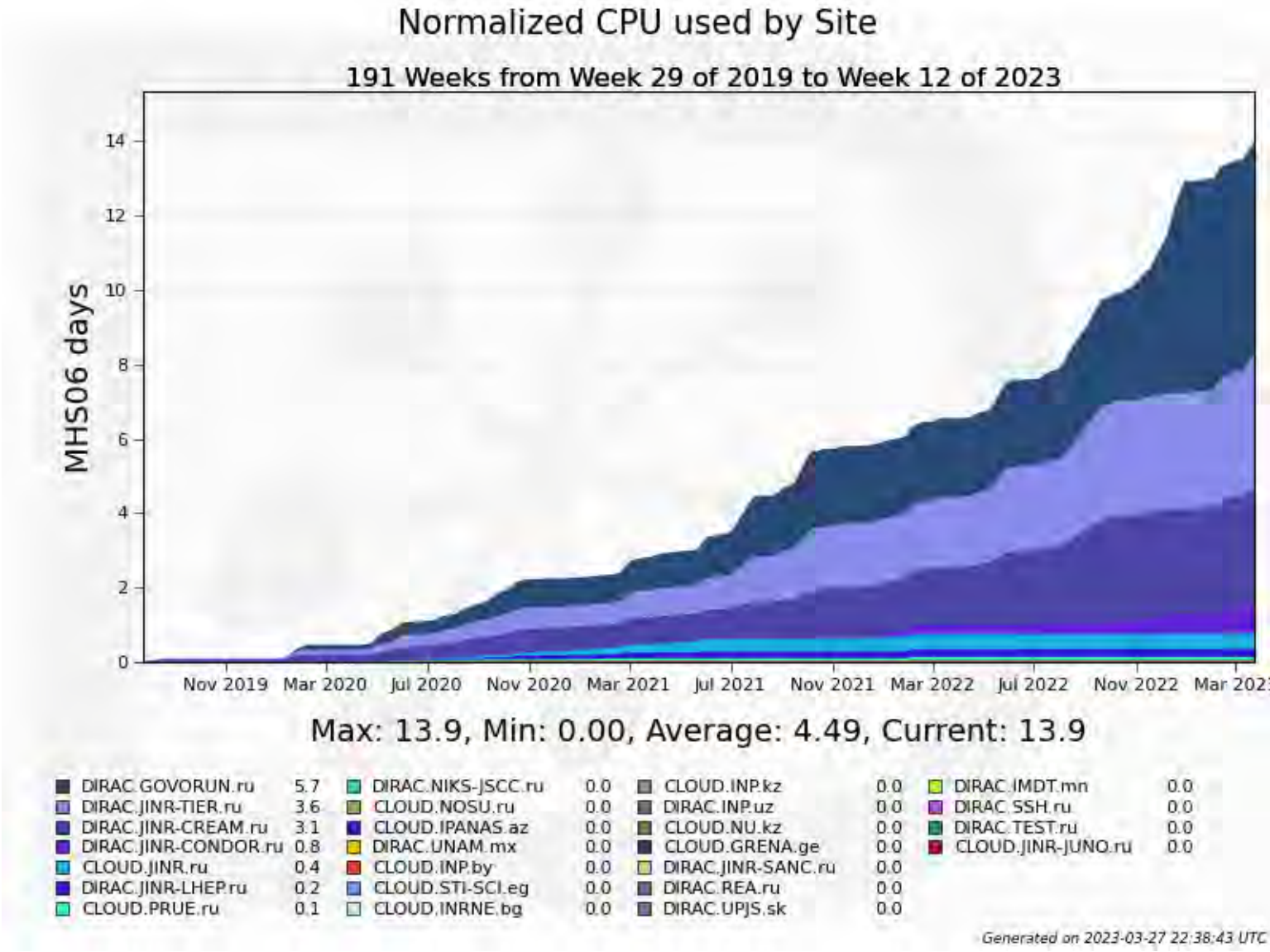
Statistics: jobs done



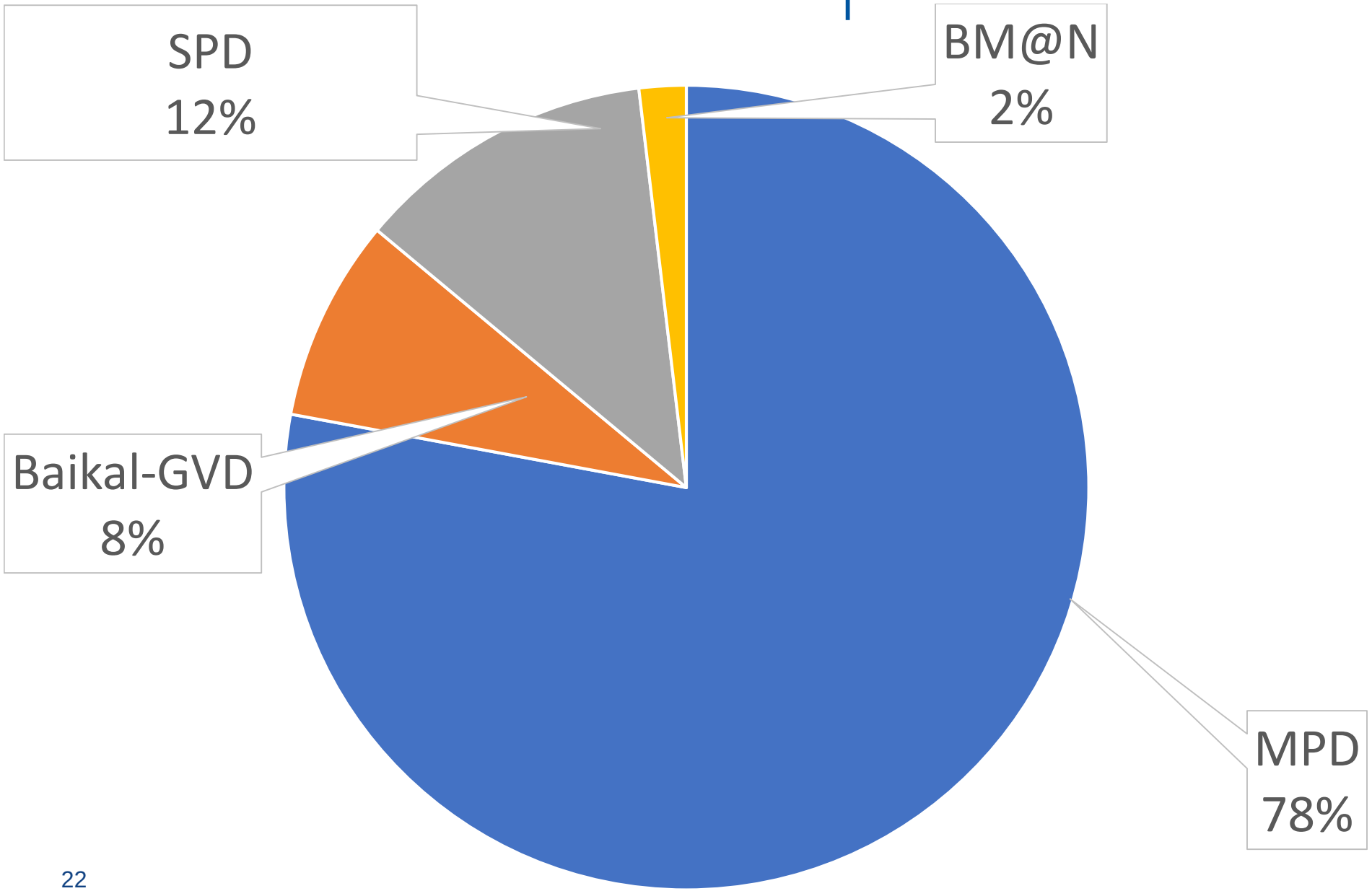
Statistics: walltime



Statistics: normalized time



Ratio: between experiments

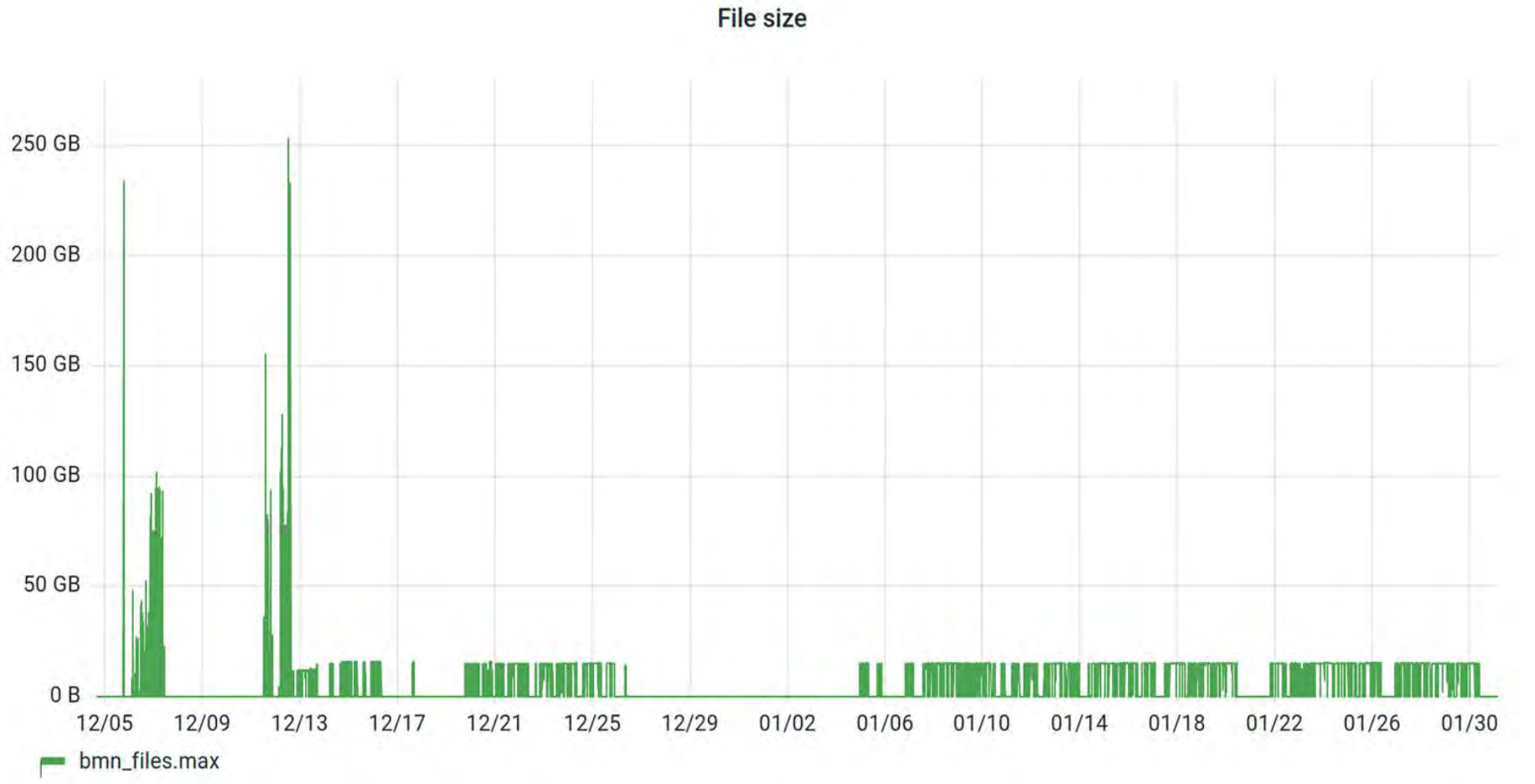


Example:

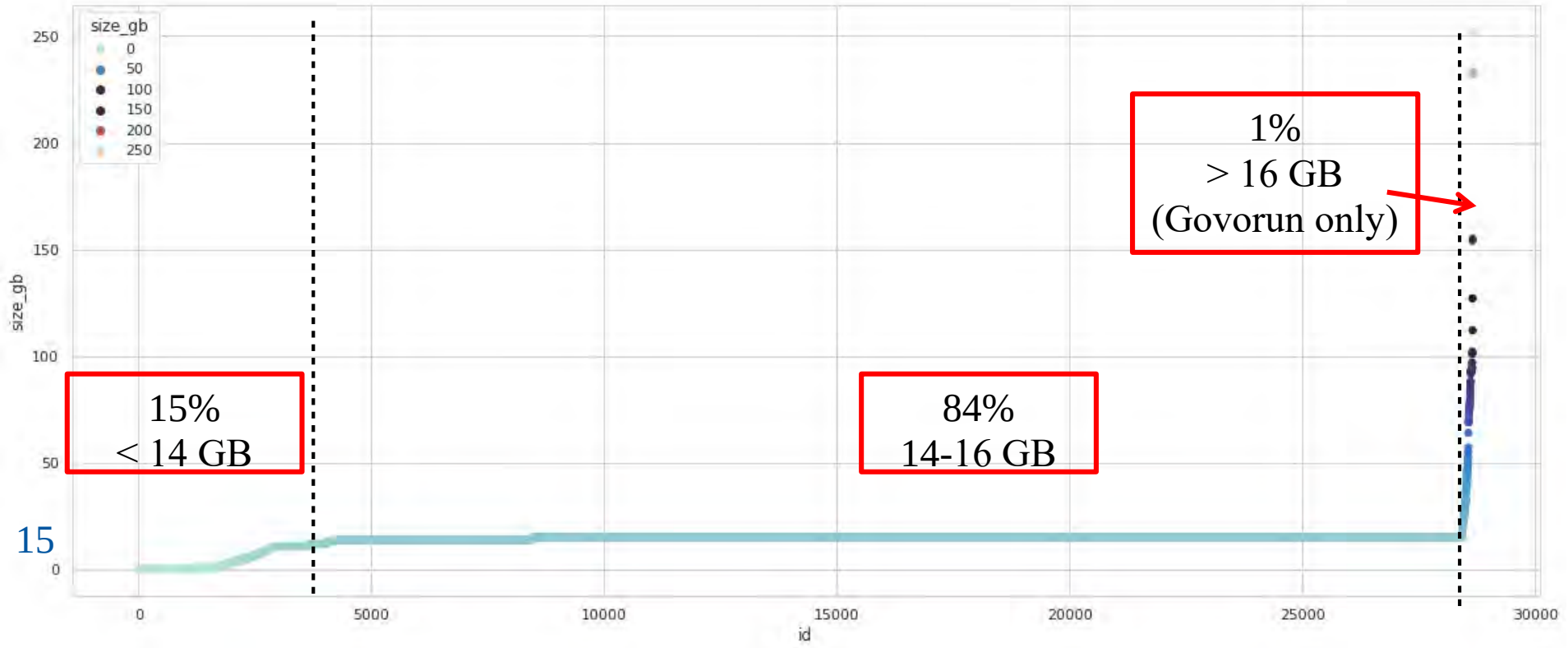
BM@N run8 raw data processing (with Konstantin Gertsenberger)

Total files ~ **30000**

Total raw size ~ **400TB**

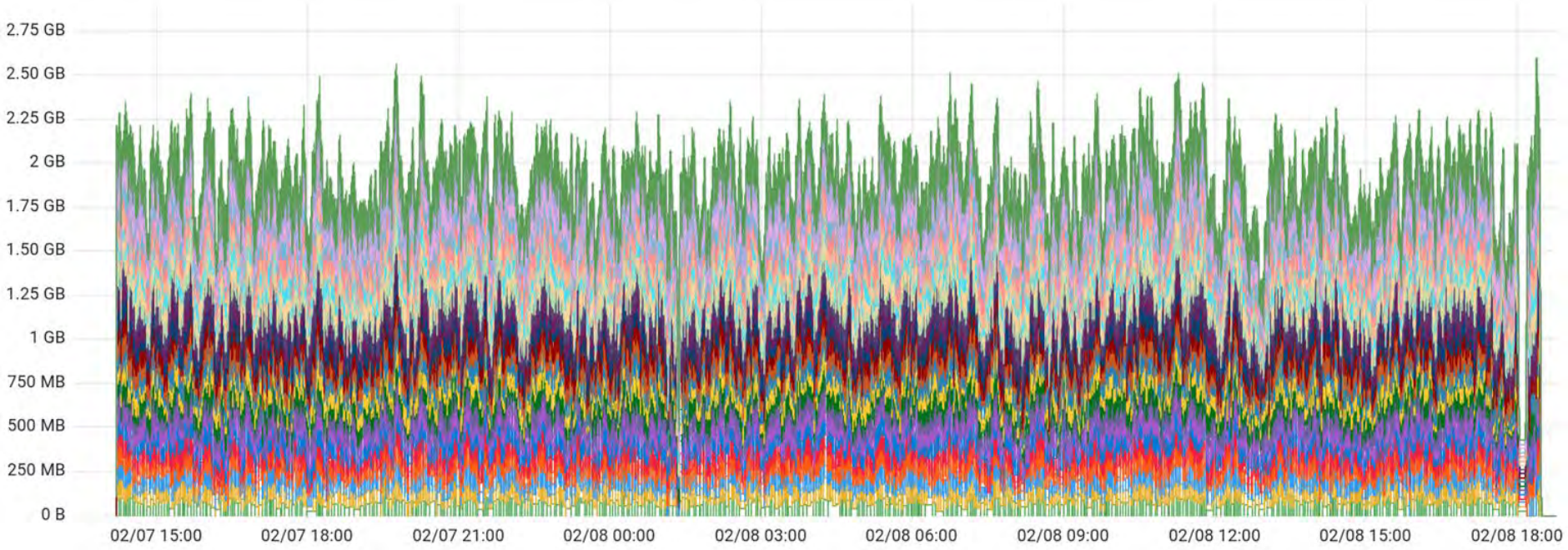


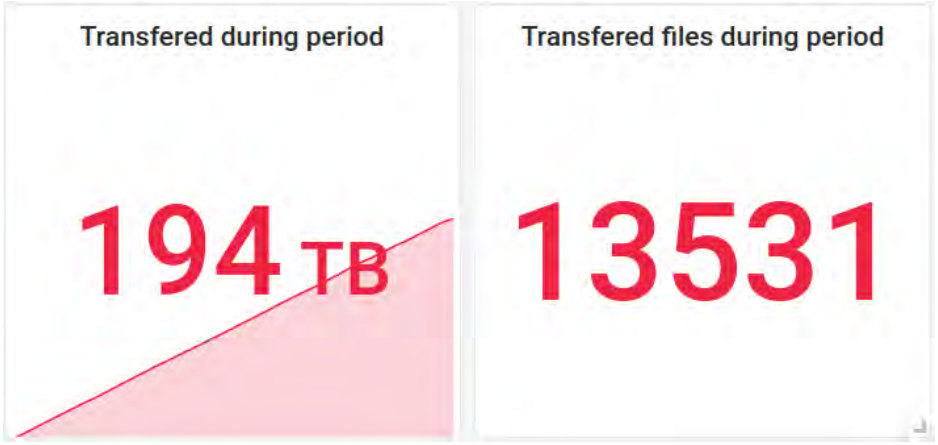
BM@N run8 raw data size



15

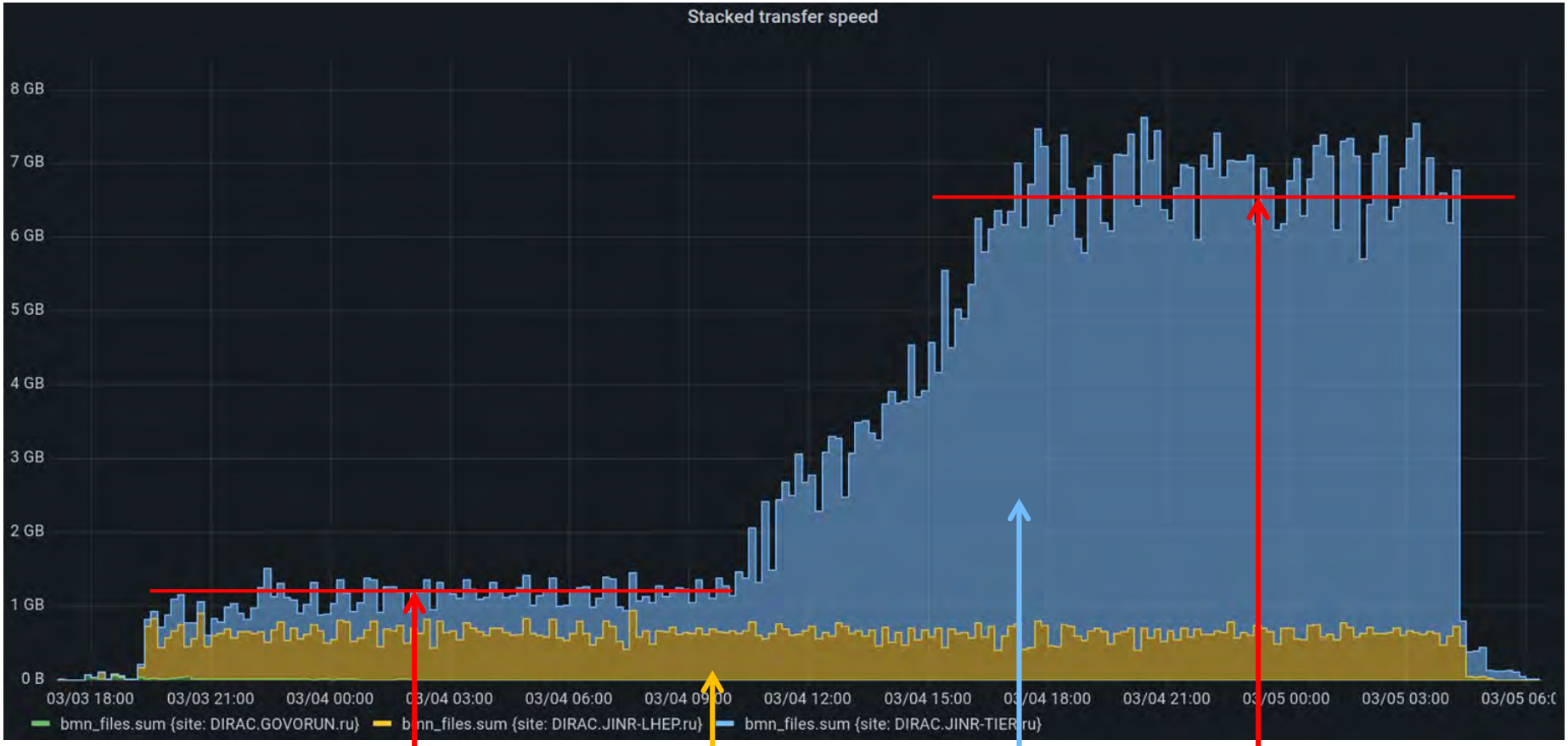
EOS LHEP -> EOS LIT





Average transfer speed on 20 streams
1.92 GB/s

BM@N Digi->Dst



300 jobs running
4 MB/s per job

NICA Cluster

Tier1

1580 jobs running
4.1 MB/s per job

List of participants

DIRAC: Igor Pelevanyk, Andrey Tsaregorodtzev
Baikal-GVD: Dmitry Zaborov
BM@N: Konstantin Gertsenberger
MPD: Oleg Rogachevskiy, Andrey Moshkin
SPD: Alexey Zhemchugov, Katherin Shtejer
Responsible for resources:
Govorun: Dmitry Podgainy, Dmitry Belyakov, Aleksandr Kokorev, Maxim Zuev,
NICA cluster: Boris Schinov, Ivan Slepov
Tier-1,Tier-2, EOS: Valery Mitsyn
Cloud: Nikolay Kutovskiy, Nikita Balashov
dCache: Vladimir Trofimov

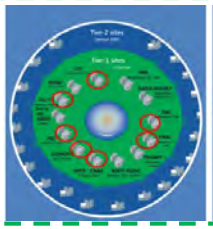
Detailed articles

1. Gergel, V., V. Korenkov, I. Pelevanyuk, M. Sapunov, A. Tsaregorodtsev, and P. Zrelov. 2017. **Hybrid Distributed Computing Service Based on the DIRAC Interware**.
2. Korenkov, V., Pelevanyuk, I. & Tsaregorodtsev, A. 2019, "**Dirac system as a mediator between hybrid resources and data intensive domains**", CEUR Workshop Proceedings, pp. 73.
3. Balashov, N.A., Kuchumov, R.I., Kutovskiy, N.A., Pelevanyuk, I.S., Petrunin, V.N. & Tsaregorodtsev, A.Y. 2019, "**Cloud integration within the DIRAC Interware**", CEUR Workshop Proceedings, pp. 256.
4. Korenkov, V., Pelevanyuk, I. & Tsaregorodtsev, A. 2020, **Integration of the JINR hybrid computing resources with the DIRAC interware for data intensive applications**.
5. Kutovskiy, N., Mitsyn, V., Moshkin, A., Pelevanyuk, I., Podgayny, D., Rogachevsky, O., Shchinov, B., Trofimov, V. & Tsaregorodtsev, A. 2021, "**Integration of Distributed Heterogeneous Computing Resources for the MPD Experiment with DIRAC Interware**", Physics of Particles and Nuclei, vol. 52, no. 4, pp. 835-841.
6. Pelevanyuk, I., "**Performance evaluation of computing resources with DIRAC interware**", AIP Conference Proceedings 2377, 040006 (2021)


JINR
MLiT

Computing Resources

JINR


Tier-1
920 slots


CICC/Tier-2
1000 slots


Cloud
80 slots


Govorun
184-3000
slots


NICA Cluster
250 slots


UNAM
100 slots

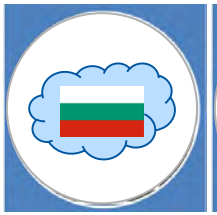
MPD
collaboration

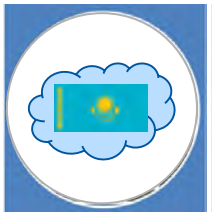
Clouds


IPANAS
16 slots


INP
132 slots


SU
48 slots


INRNE
20 slots


INP
50 slots


REA Plehanova
132 slots


NOSU
84 slots


STI-SCI
98 slots

Total amount of cores exceeds 5000

Individual CPU core performance study

- Centralized job management gives possibility for centralized and unified performance study of different computing resources.
- Before running user jobs DIRAC Pilots execute benchmark for CPU core they are running on.
- Benchmark is DiracBenchmark2012 or DB12. It evaluate just CPU core performance. Disk I/O, RAM speed, Network, CPU caches and other highly important aspects of performance are **neglected by DB12.**

DB12 benchmark study

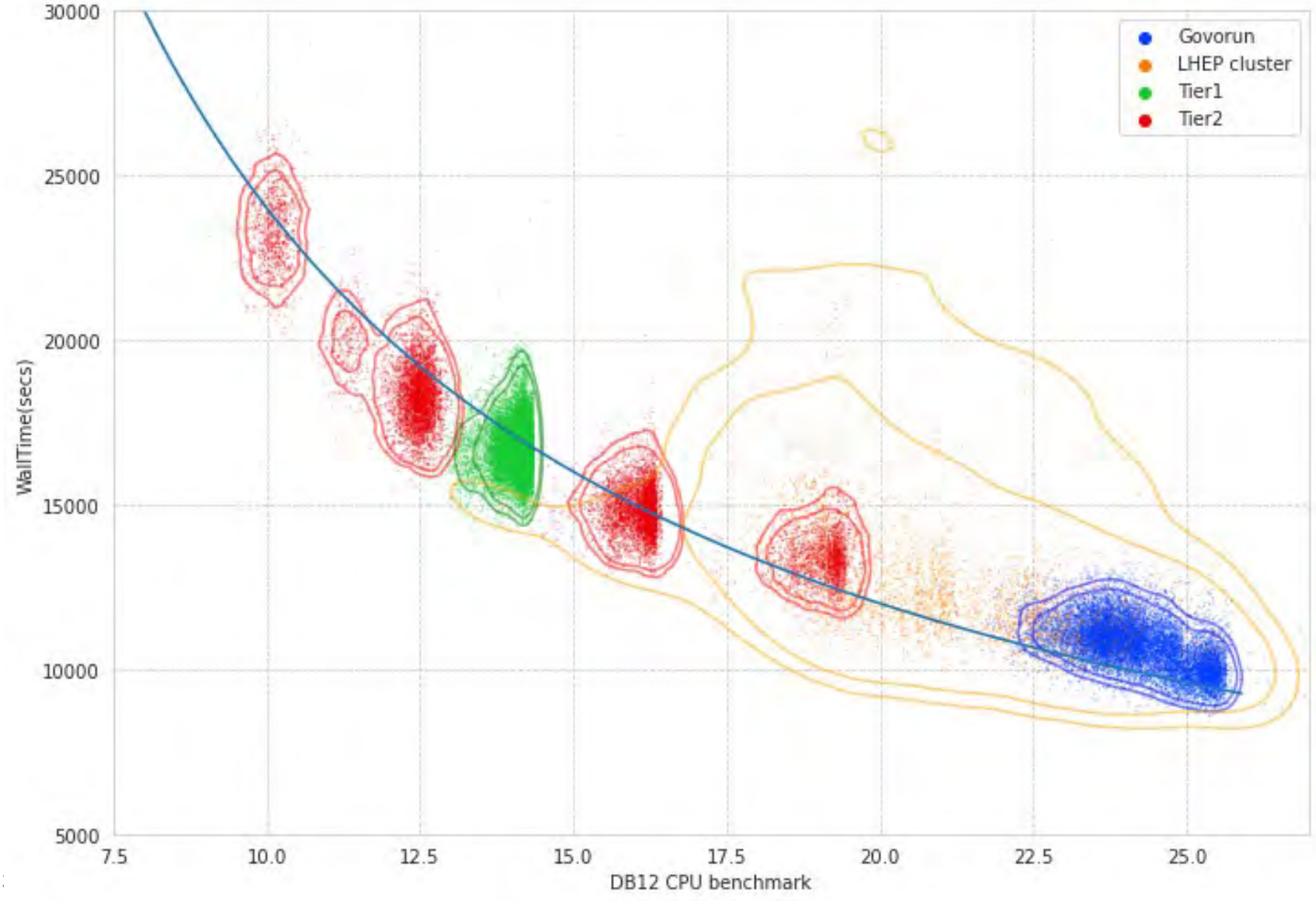
Piece of road from point A to B	←	A Monte-Carlo task
Speed of the car	←	Performance of the computer
Time to complete	←	Time to complete

$$Time = \frac{Amount\ of\ work}{Speed\ of\ computer}$$

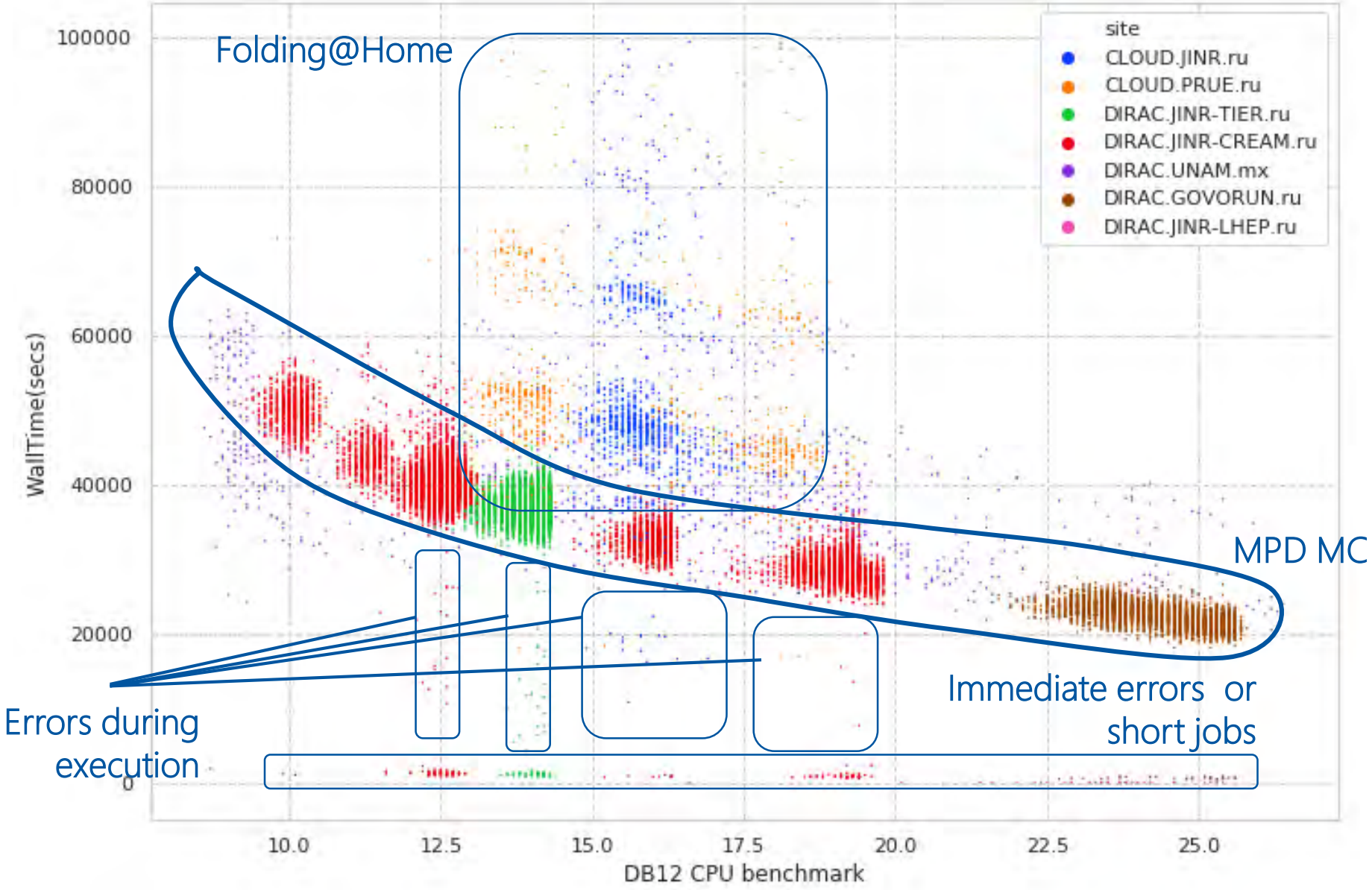
DB12 gives results like: 10(old slow core), 17 (standard server core), 27 (high performance core)

What if we build a plot, where X is DB12 result, Y is time in seconds. Then, every point on the plot represent one job.
It would be mostly useless if all jobs were unique and different.
But, in the real life there are usually many similar jobs.

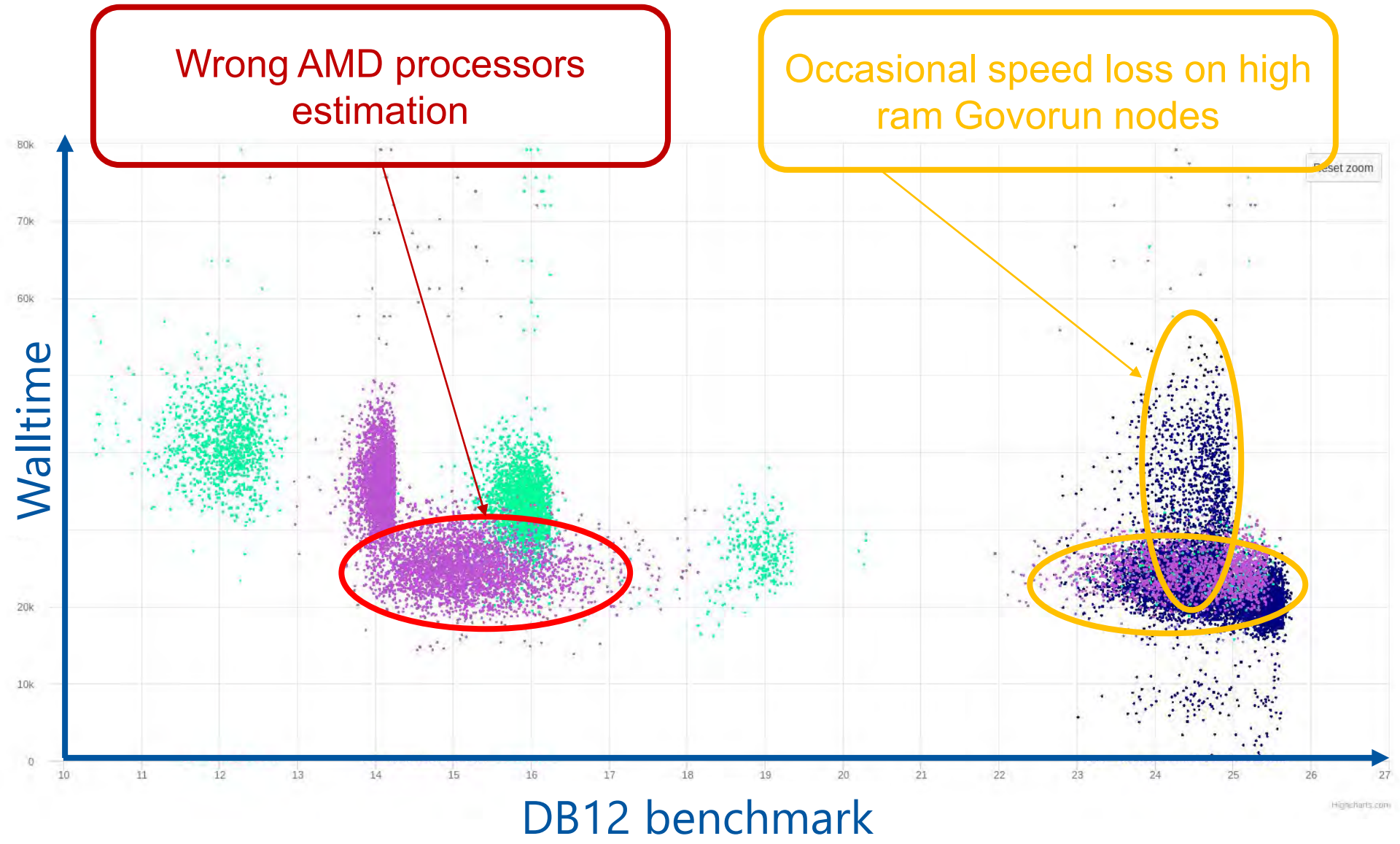
Performance analysis



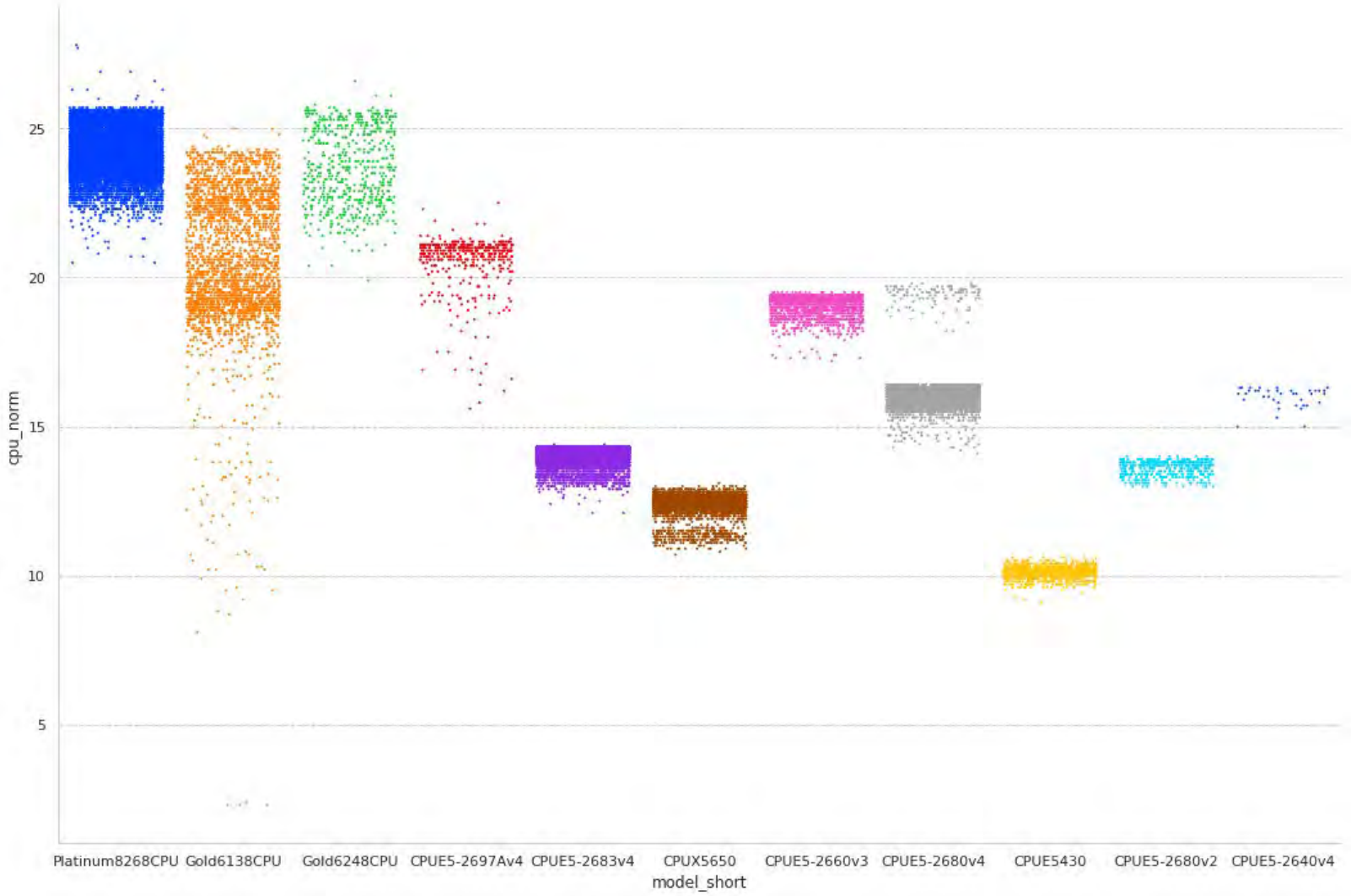
Performance analysis



Discoveries



CPU core performance

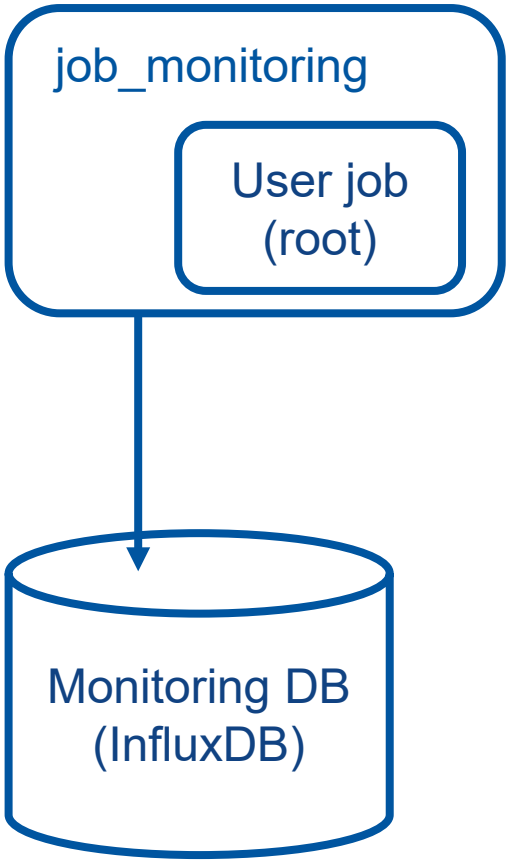


Total CPU performance

User job monitoring

```
$ root macro.c(input)
```


```
$ job_monitoring root macro.c(input)
```



User job monitoring

GenToDst job on Govorun