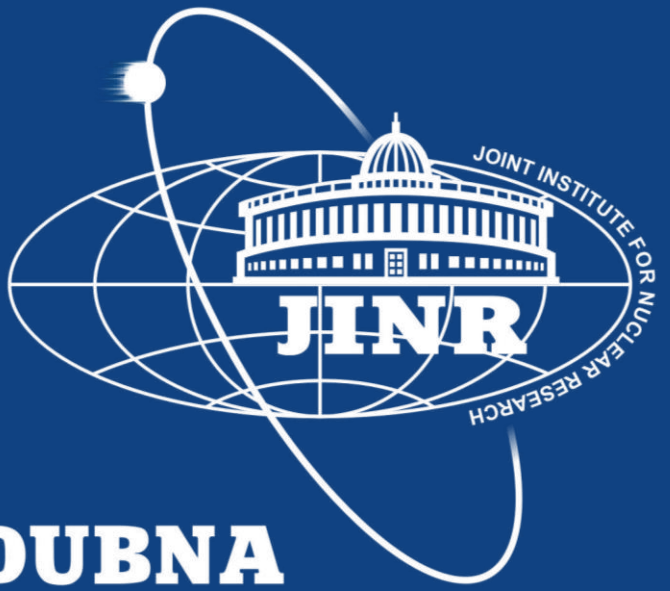

JOINT INSTITUTE FOR NUCLEAR RESEARCH
JINR
DUBNA

# Simulation of job execution in distributed heterogeneous computing infrastructures

Igor Pelevanyuk, Daniel Campis



27 October 2022

# What was done in JINR

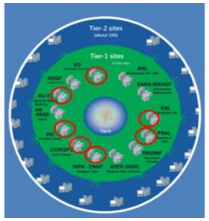
| Tier-1 | CICC/Tier-2 | Clouds | Govorun | NICA Cluster | UNAM |
|---|---|---|---|---|---|
| **Running** | **Running** | **Running** | **Running** | **Running** | **Running** |

The computing resources of the JINR Multifunctional Information and Computing Complex, clouds in JINR Member-States, cluster from Mexico University were combined using the DIRAC Interware.

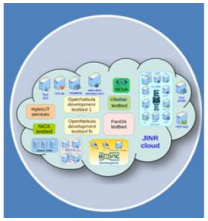
# Workload management

Submit thousand of jobs to DIRAC Job Queue

DIRAC
THE INTERWARE


Tier-1

CICC/Tier-2


Clouds

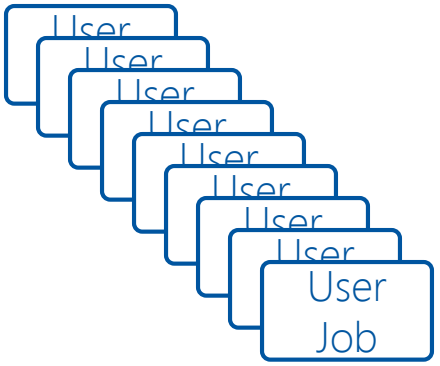
Govorun

NICA Cluster

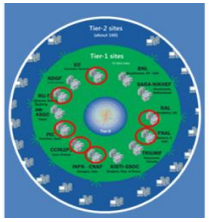
UNAM

# Workload management

Submit thousand of jobs to DIRAC Job Queue




DIRAC
THE INTERWARE


Tier-1

CICC/Tier-2

Clouds

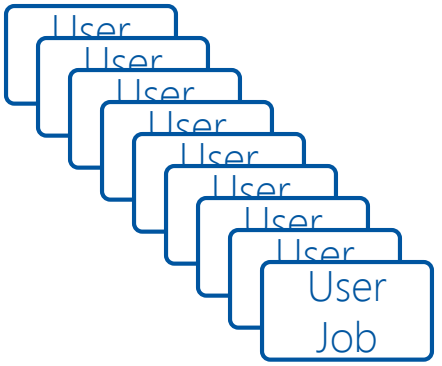
Govorun

NICA Cluster

UNAM

# Workload management

Submit thousand of jobs to DIRAC Job Queue



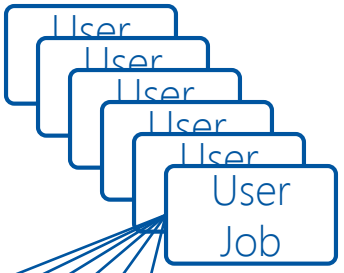

DIRAC
THE INTERWARE

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



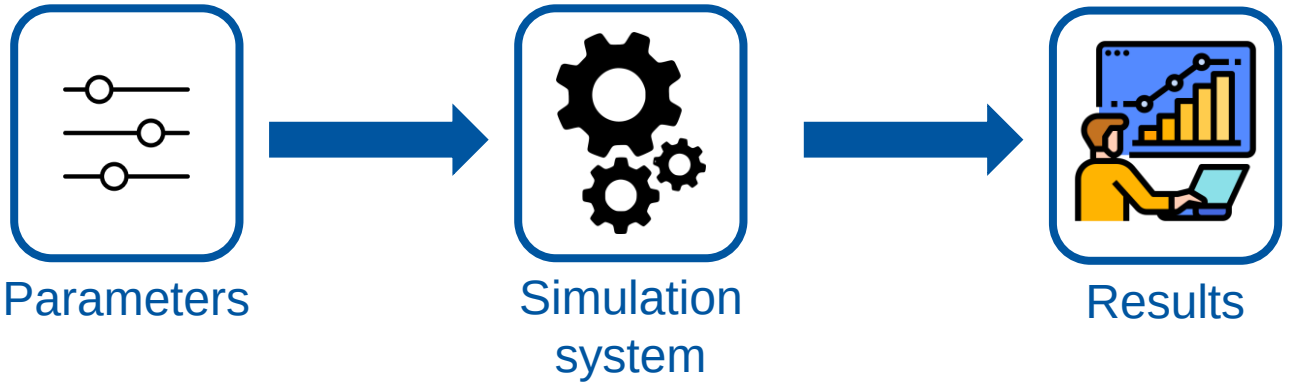

DIRAC
THE INTERWARE

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

# Questions that we ask

- Are we efficient in our job execution?
  - We have no access to monitoring information on a remote computing resources
  - Even if we have monitoring, it may be spoiled by other jobs
- Can we utilize more computing resources if we would have them in our disposal?
  - It is not so difficult to submit jobs on another thousand of cores, but will network handle them efficiently?

# Simulation is a solution



# Parameters

## Infrastructure description

- Storage elements: network speed
- Computing elements: network speed, list of servers
- Servers: cpu cores amount, cpu core performance, RAM, network

## Workload description

- Jobs: amount, rate of incoming, list of tasks
- Tasks: computing or transfer tasks with cpu work or transfer amount correspondingly

# Where do we get them

## Infrastructure description

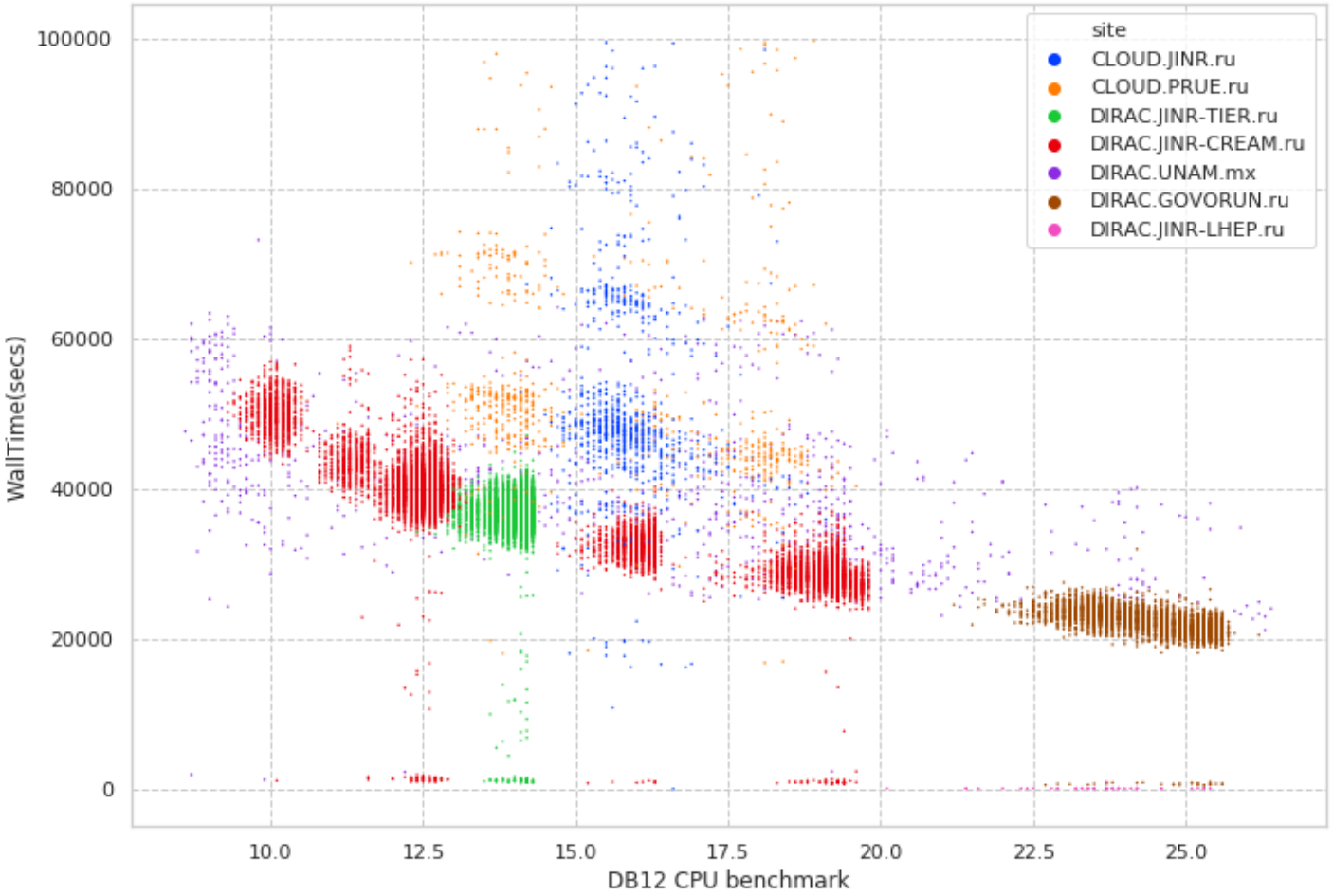
- Storage elements: network speed – transfer experiments
- Computing elements: network speed, list of servers - transfer experiments + accounting
- Servers: cpu cores amount, cpu core performance, RAM, network – network configuration + accounting

## Workload description

- Jobs: amount, rate of incoming, list of tasks
- Tasks: computing or transfer tasks with cpu work(in DB12 second) or transfer amount(in bytes) correspondingly

These values may be received either from specialist or from user job monitoring.

# Cluster configuration




Igor Pelevanyuk, **"Performance evaluation of computing resources with DIRAC interware",** AIP Conference Proceedings 2377, 040006 (2021) https://doi.org/10.1063/5.0064778

# CPU performance

Igor Pelevanyuk, **"Performance evaluation of computing resources with DIRAC interware",** AIP Conference Proceedings 2377, 040006 (2021) https://doi.org/10.1063/5.0064778

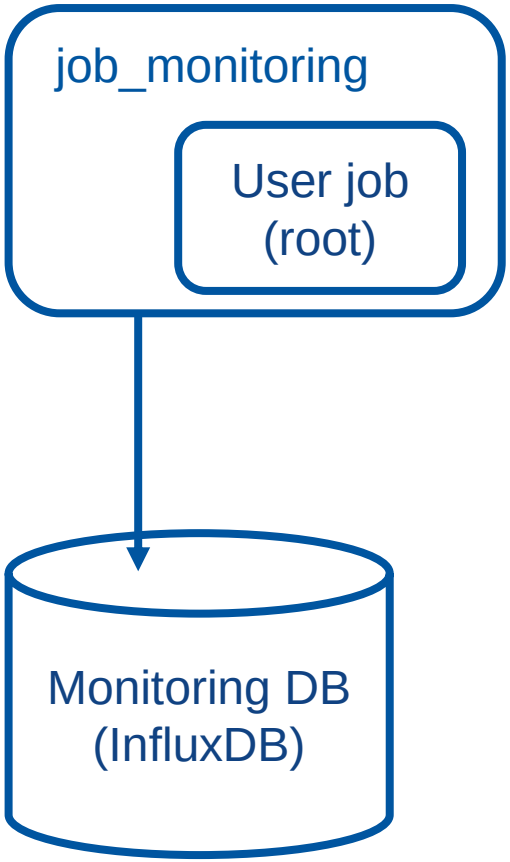
# Parameters example

- If we have **20** computing worknodes, **40** cores available on each worknode.
- If **100 MB/s** maximum disk writing speed on each worknode.
- If new **40 GB** RAW file appears every **90** seconds. 105000 events in each RAW file.
- If each event processing time is 0.5 sec – one file processing will last for **14.5** hours.

# User job monitoring

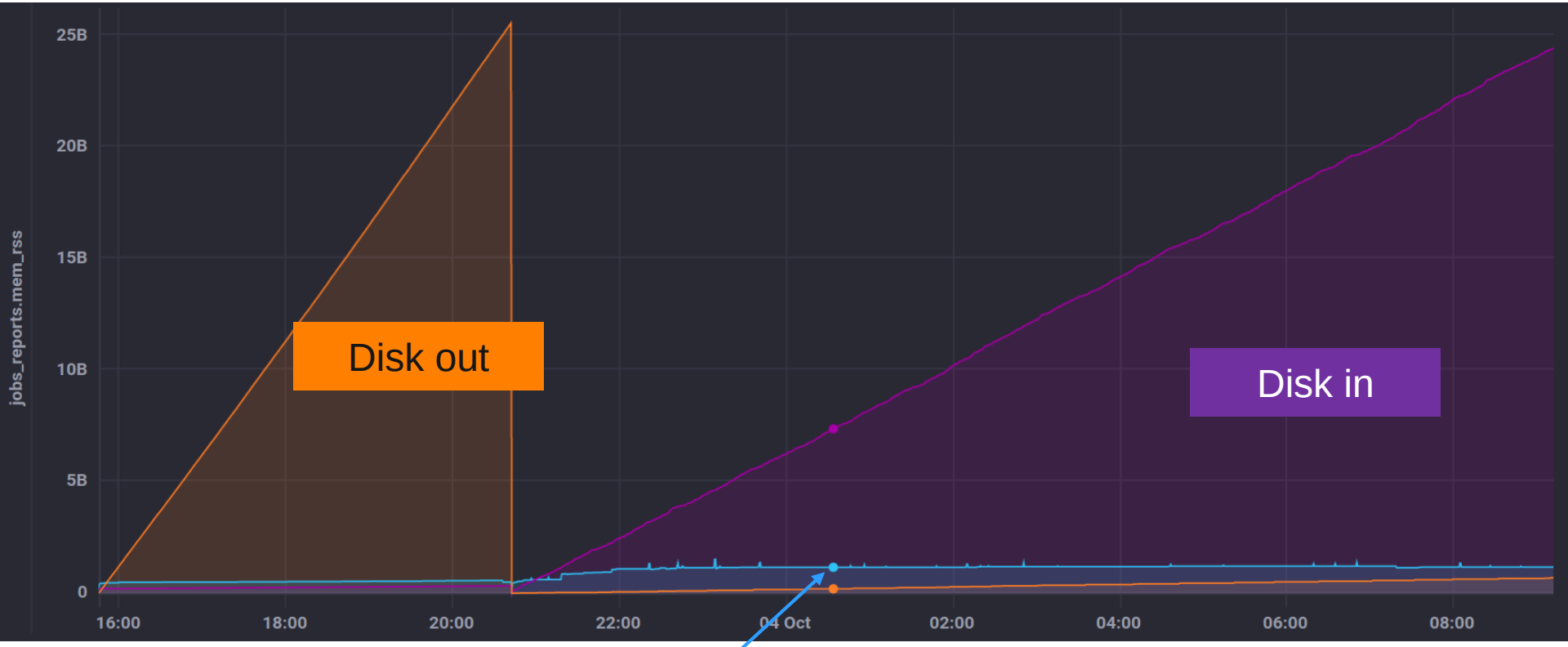
```
$ root macro.c(input)
```

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



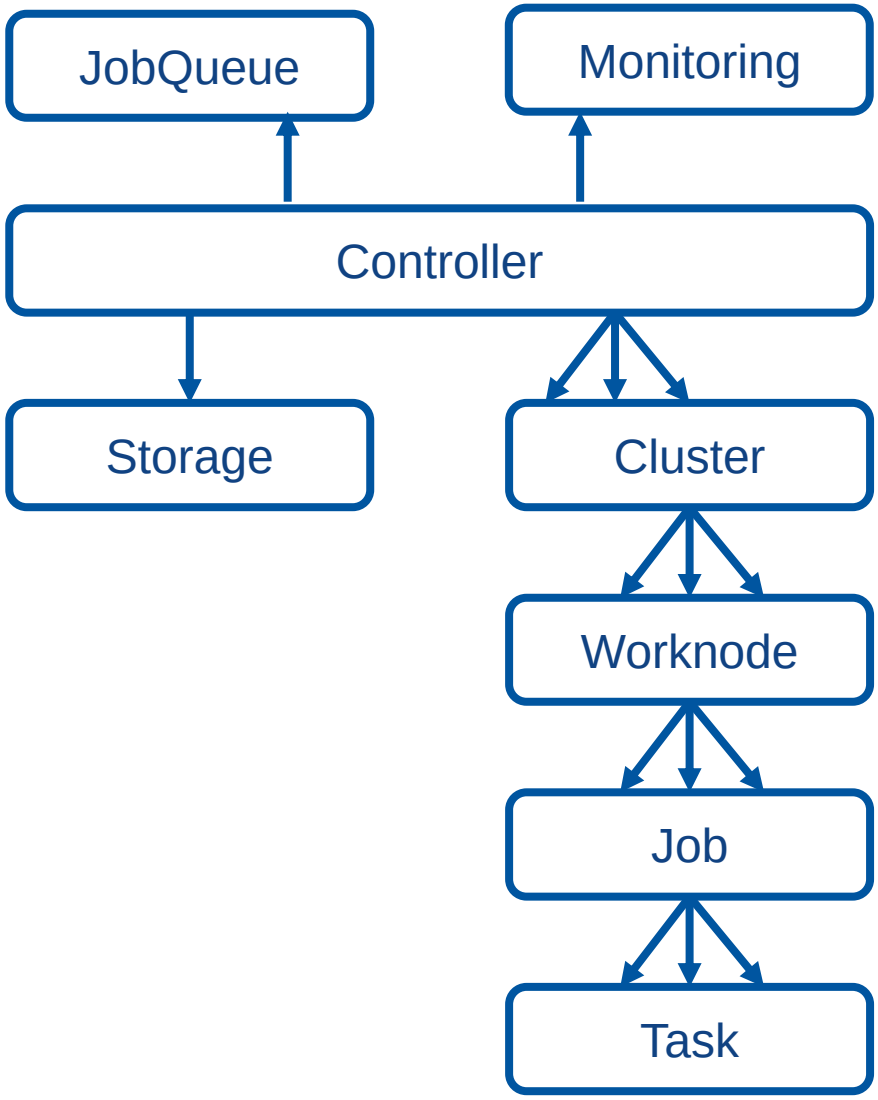
# User job monitoring

## GenToDst job on Govorun

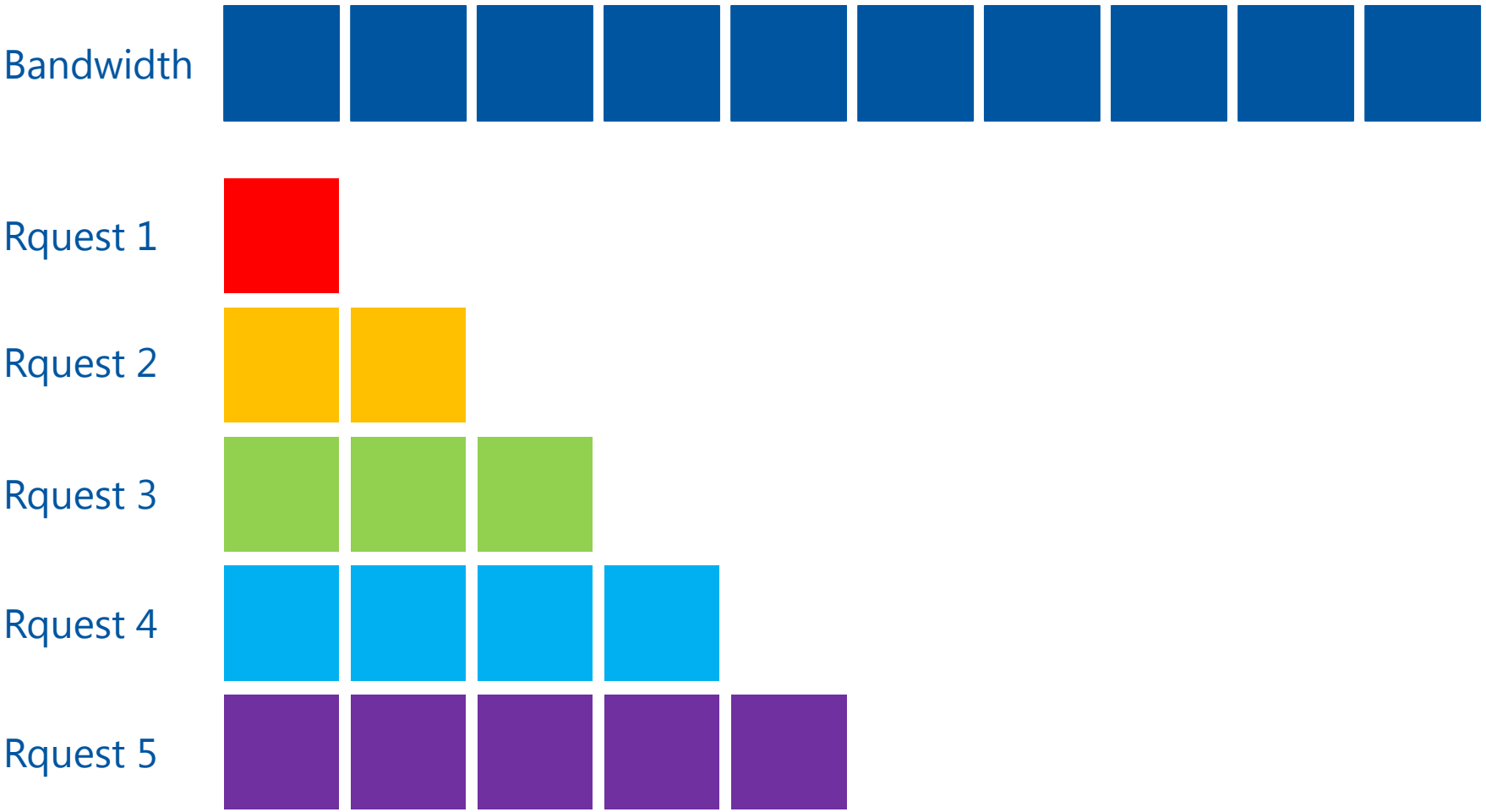


# Simulation system

- Written in python to predict CPU, RAM, network and disk load
- Uses data about performance of resources integrated in DIRAC
- It is used to check the behavior of DIRAC jobs in real infrastructure.
- Simulation is done every second, but period may be increased for speeding up simulation.
- InfluxDB is used for results storage and visualization



# Most important part
## transfer simulation



# Simple approach

Bandwidth

Rquest 1

Rquest 2

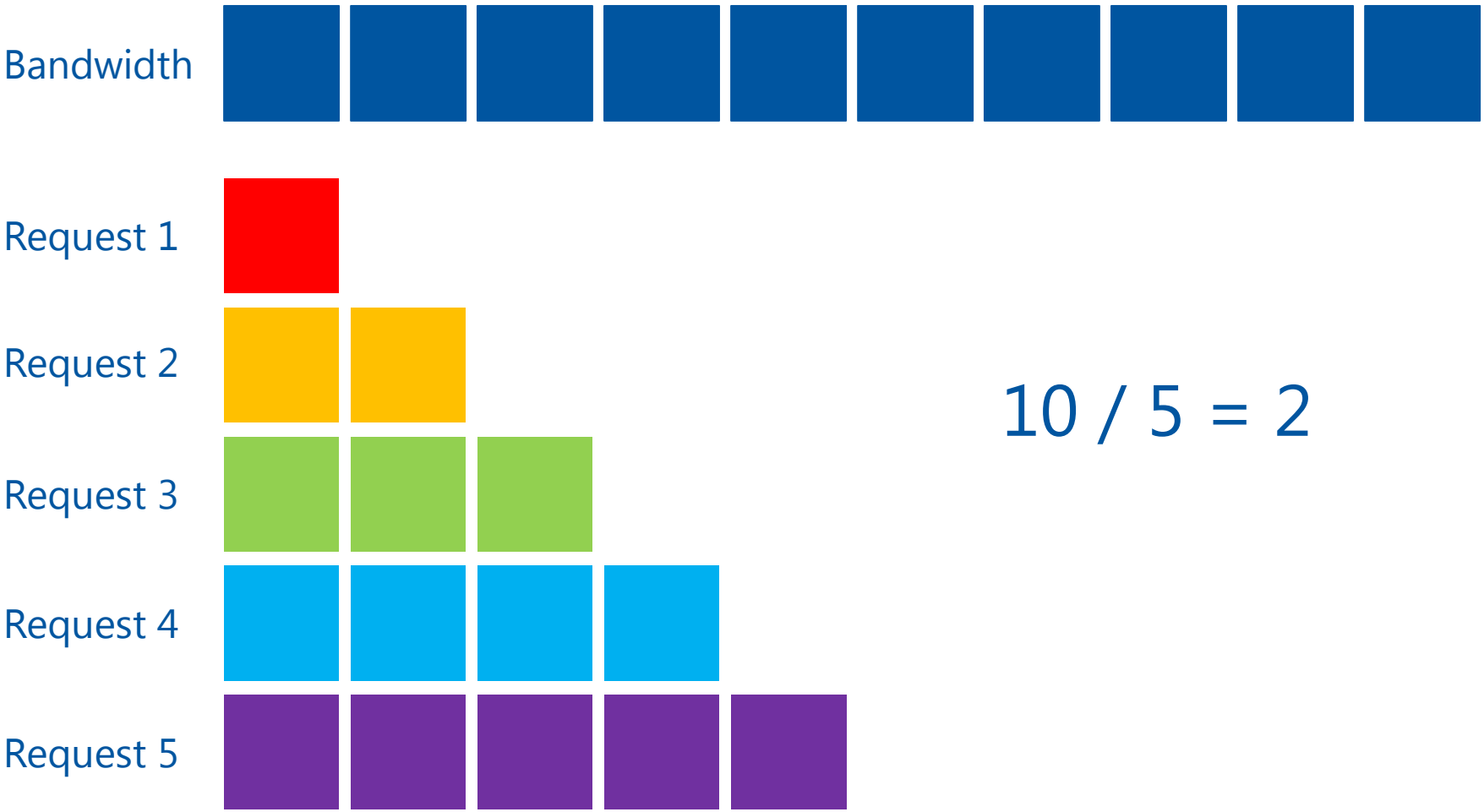
Rquest 3

Rquest 4

Rquest 5

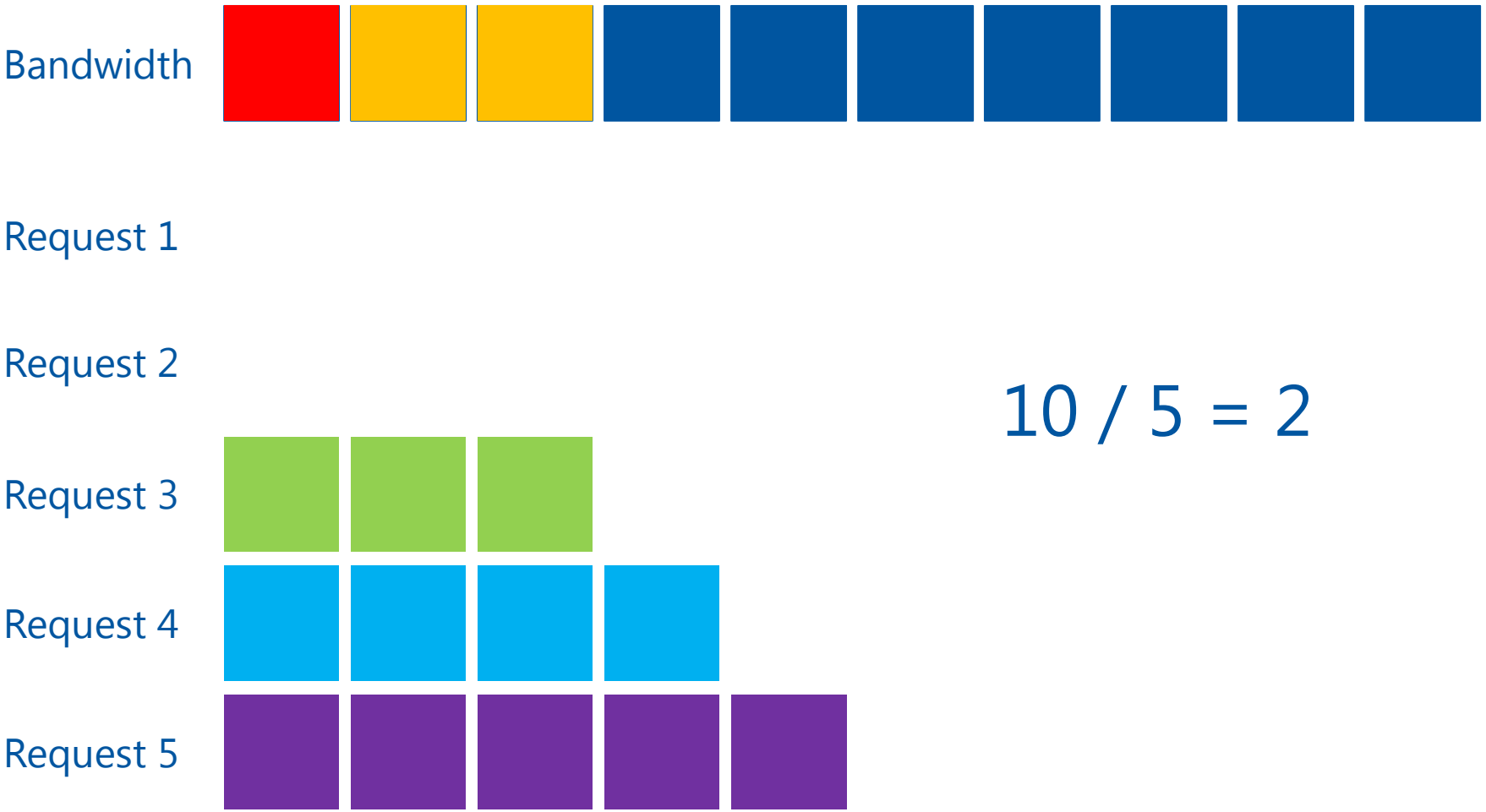
10 / 5 = 2

# Used approach

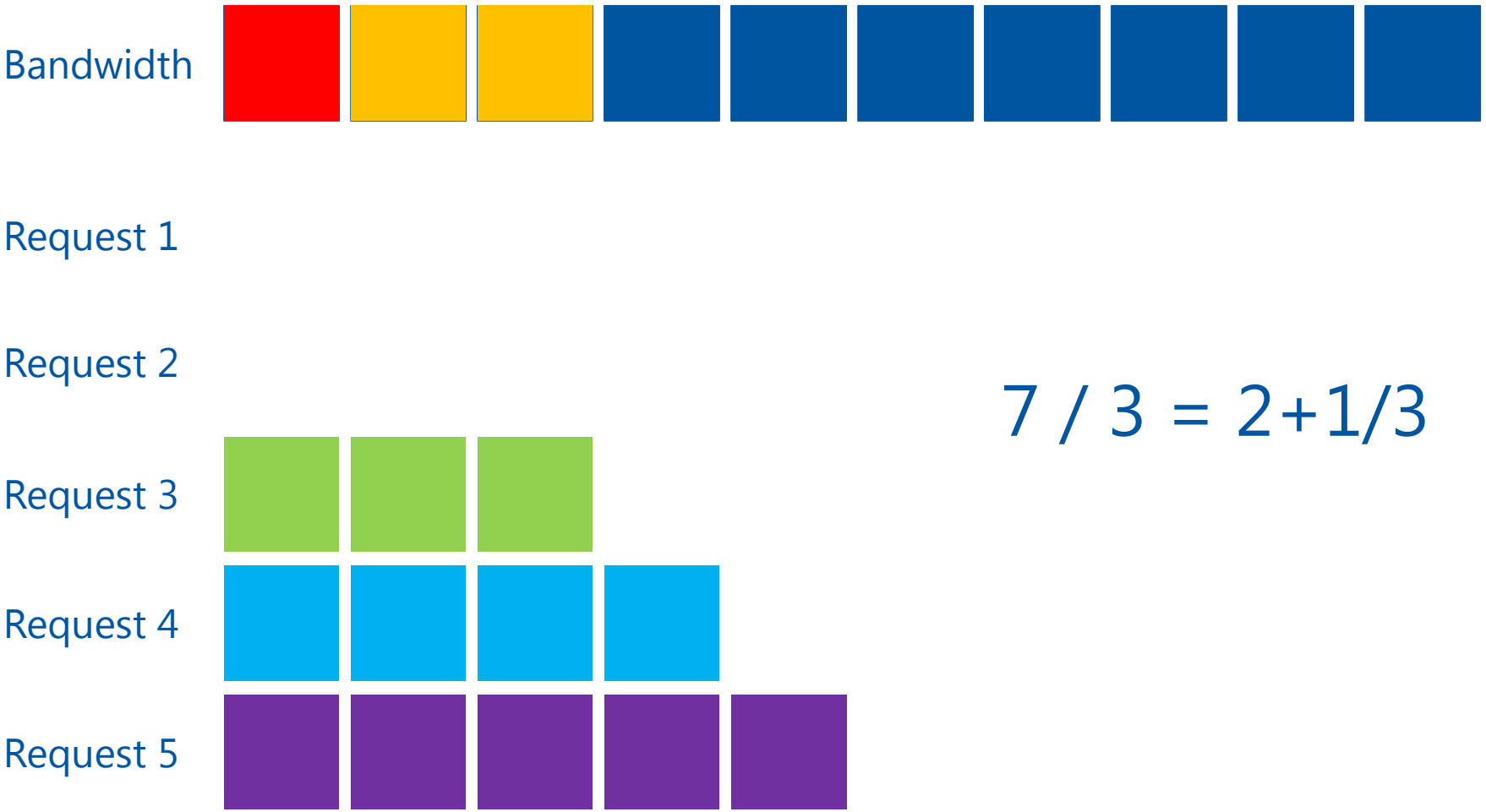



10 / 5 = 2

# Try to apply share



# Recalculate share

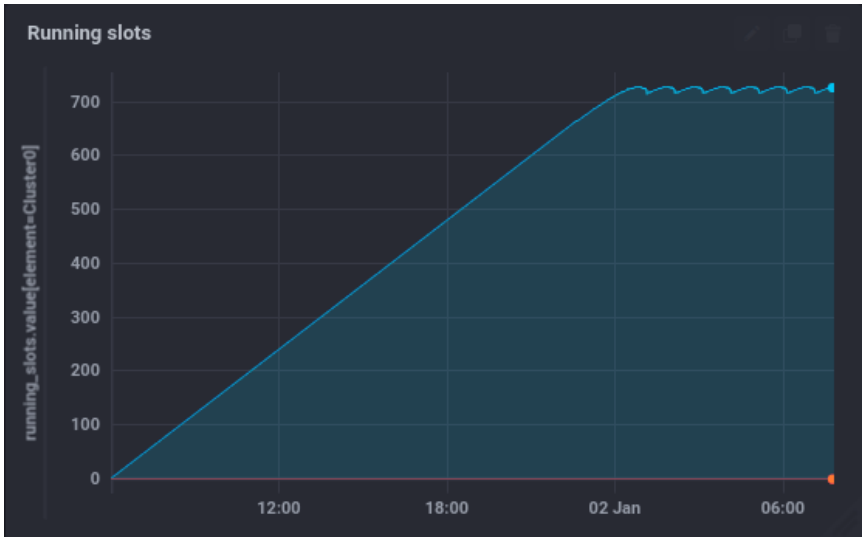


# Apply share

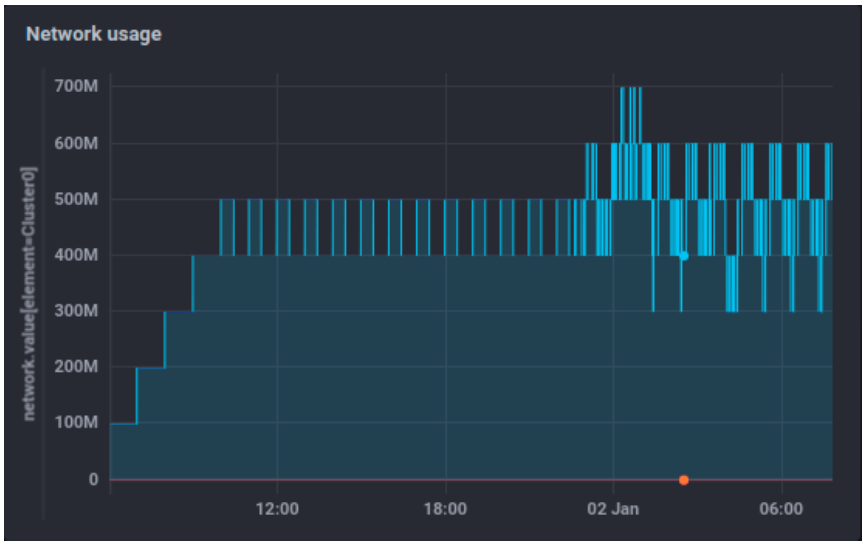
Bandwidth
Request 1
Request 2
Request 3
Request 4
Request 5
7 / 3 = 2+1/3
These are left for the next simulation period

# Results -1

Jobs are submited sequentially on each worknode until it is full

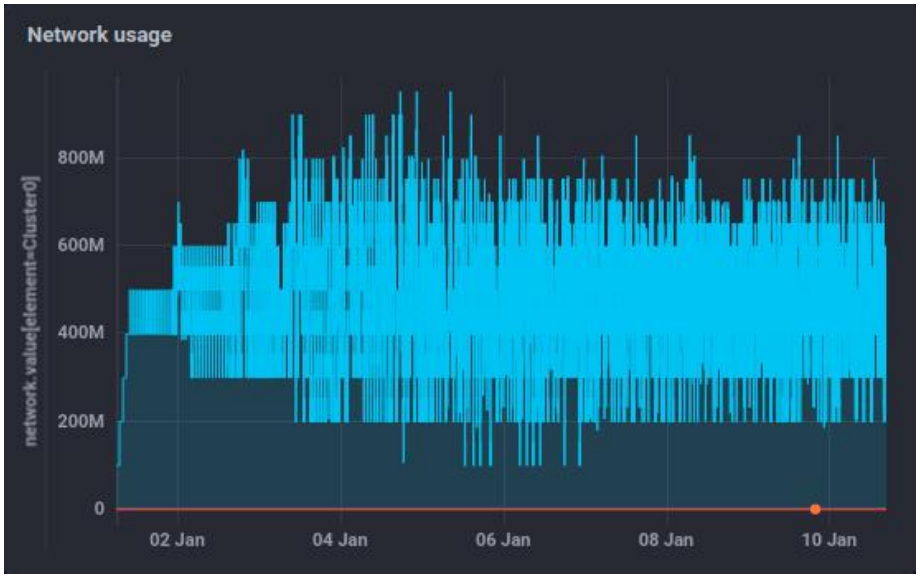


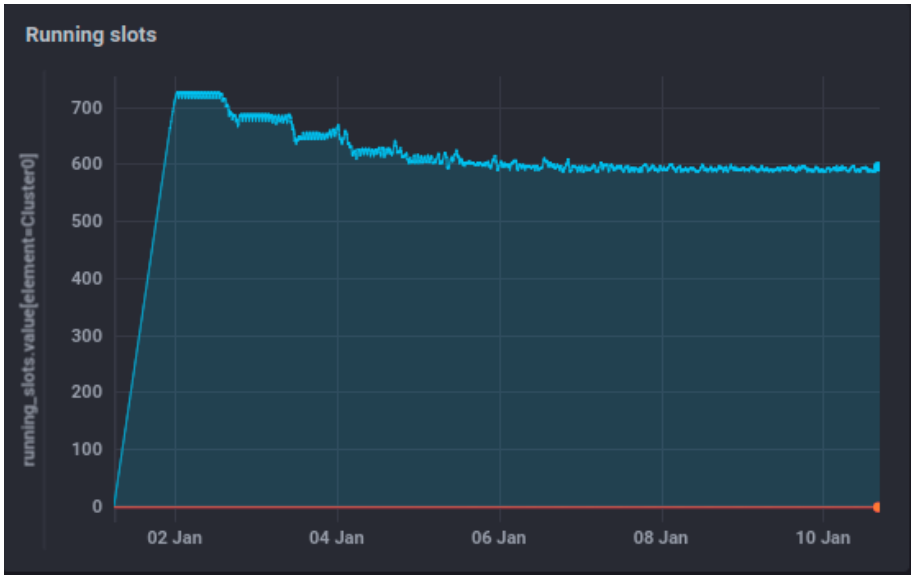

- Stable state of the system after ~19 hours
- Maxmim amount of running jobs ~730 (91% of slots)
- Eventually ~592 slots required
- Initial CPU load of available resources ~ 80%.

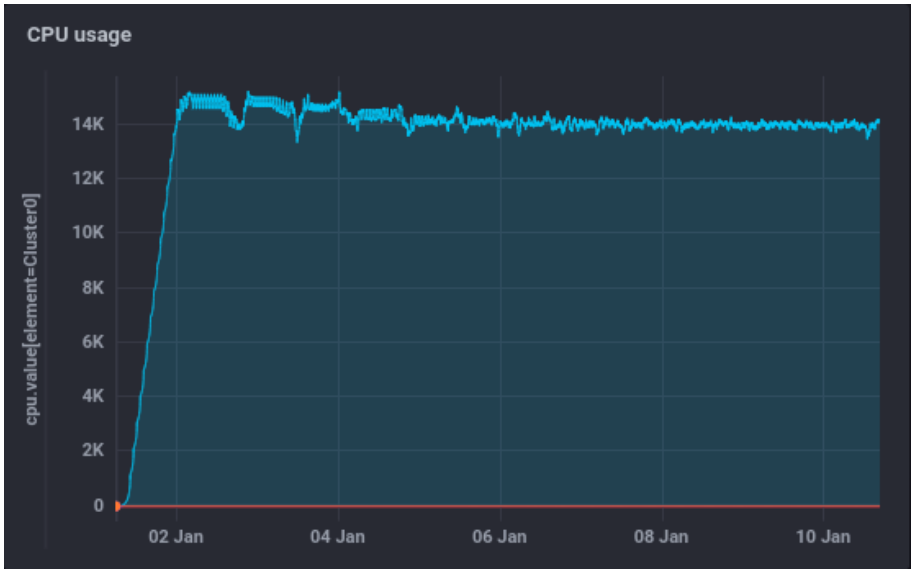



- Network usage not more than 700 MB/s
- Average usage between 400 and 500 MB/s

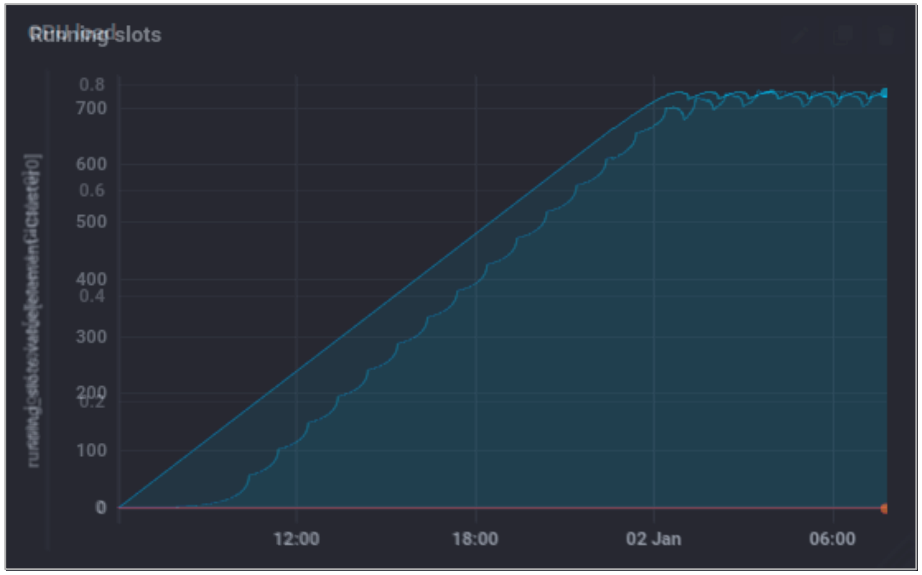
# Results - 1


Network usage
network.value[element=Cluster0]
800M
600M
400M
200M
0
02 Jan
04 Jan
06 Jan
08 Jan
10 Jan


Running slots
running_slots.value[element=Cluster0]
700
600
500
400
300
200
100
0
02 Jan
04 Jan
06 Jan
08 Jan
10 Jan


CPU usage
cpu.value[element=Cluster0]
14K
12K
10K
8K
6K
4K
2K
0
02 Jan
04 Jan
06 Jan
08 Jan
10 Jan

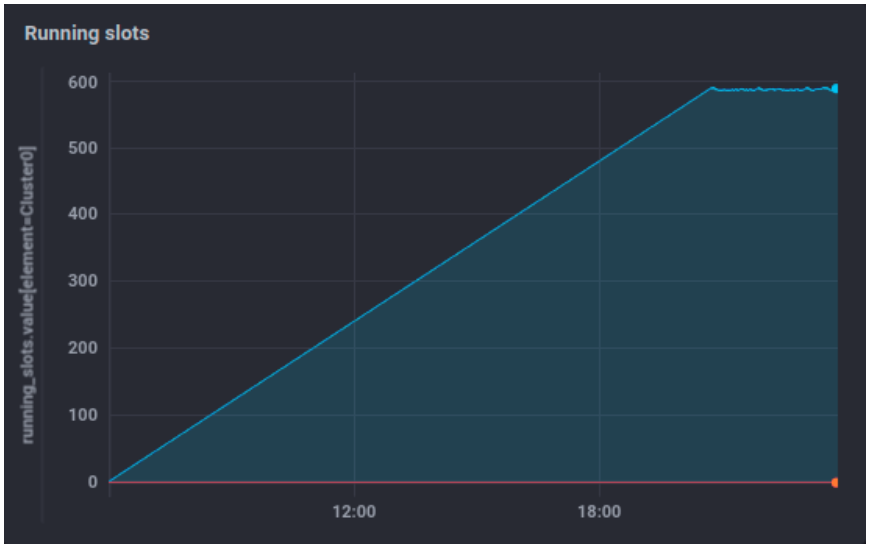
# Strange results



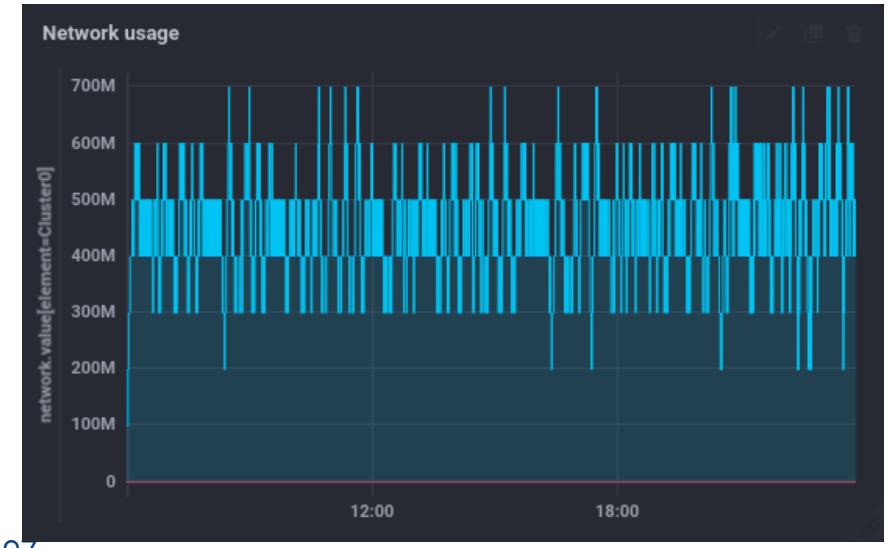

- These “waves” are on CPU load graph is definitely an issue.
- If we place Running slots graph on top of CPU load graph we see that in the beginning many slots occupied by jobs which struggle to download data.
- What if we will distribute jobs among worknodes randomly?

# Results -2

Jobs are submited on a random worknode



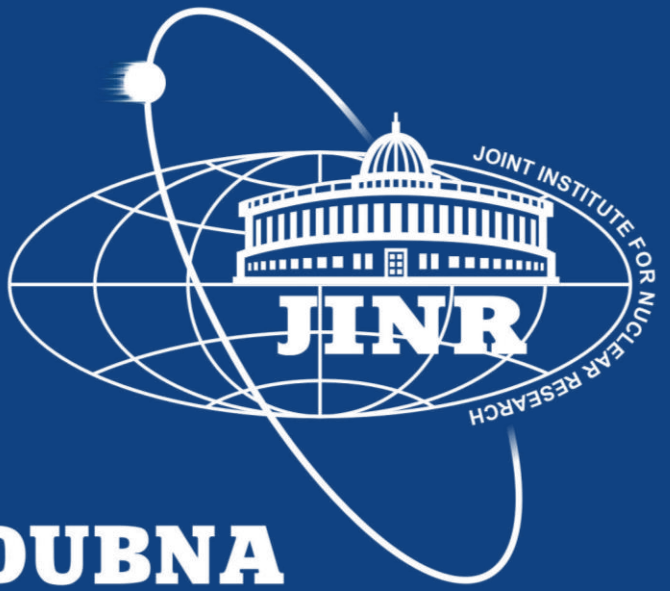

- No disaster happening
- Stable state of the system after ~15 hours
- Maxmim amount of running jobs ~591 (74% of slots)
- Average CPU load of available Govorun resources ~ 73%.




- Network usage not more than 700 MB/s
- Average usage ~450MB/s

# Conclusion

- Simulation of job execution can give accurate predictions of a load
- Accuratnes of prediction highly depends on initial parameters
- With simulation it is possible to estimate network load which is the biggest limiting factor in real job execution
- Special use-case simulation was performed in order to validate simulation results. **Thank to Daria Pryakhina and the team from *Simulation results of BM@N computing infrastructure* talk**


JOINT INSTITUTE FOR NUCLEAR RESEARCH
JINR
DUBNA