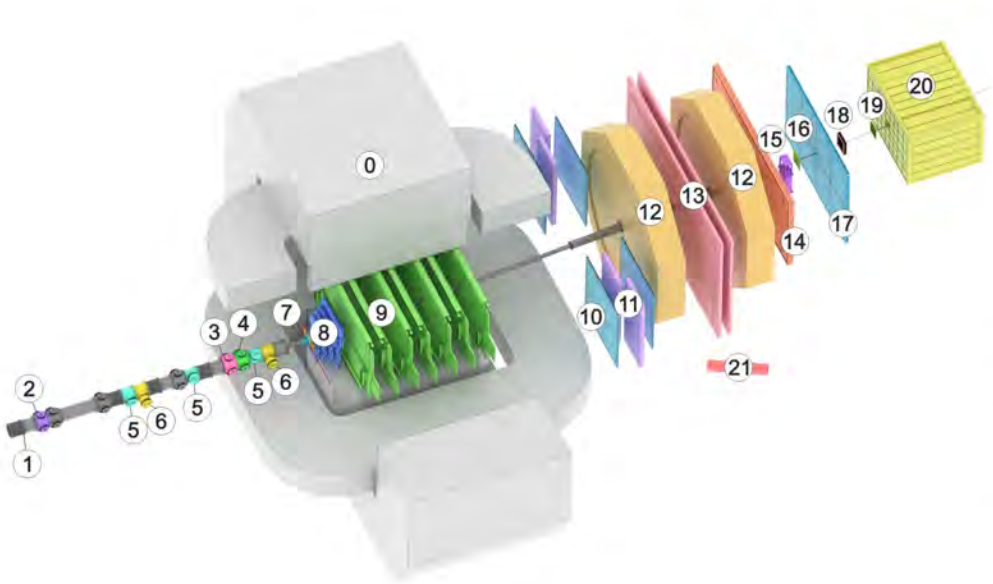
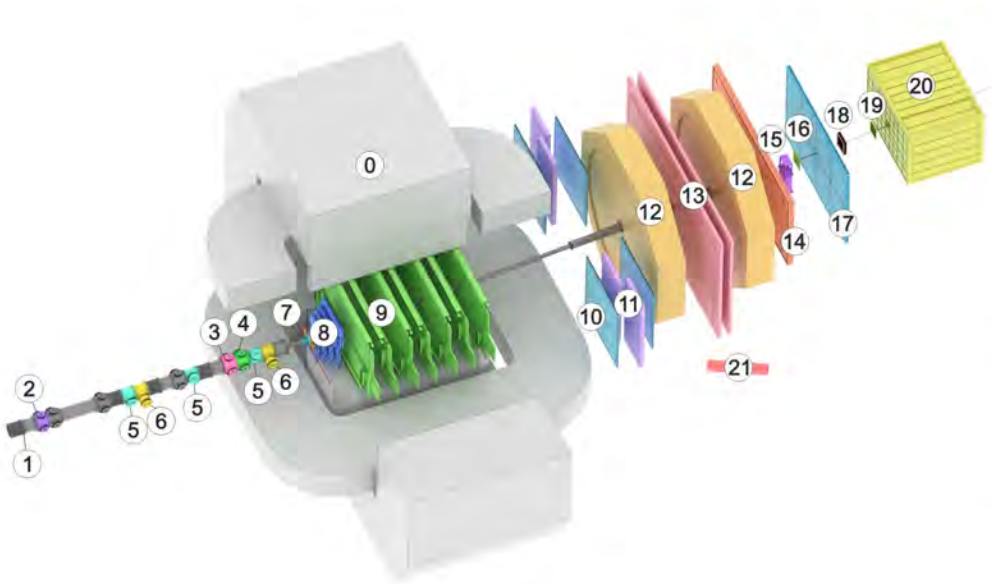
BM@N Distributed computing readiness for Run 9

Konstantin Gertsenberger, Igor Pelevanyuk
LHEP MLIT



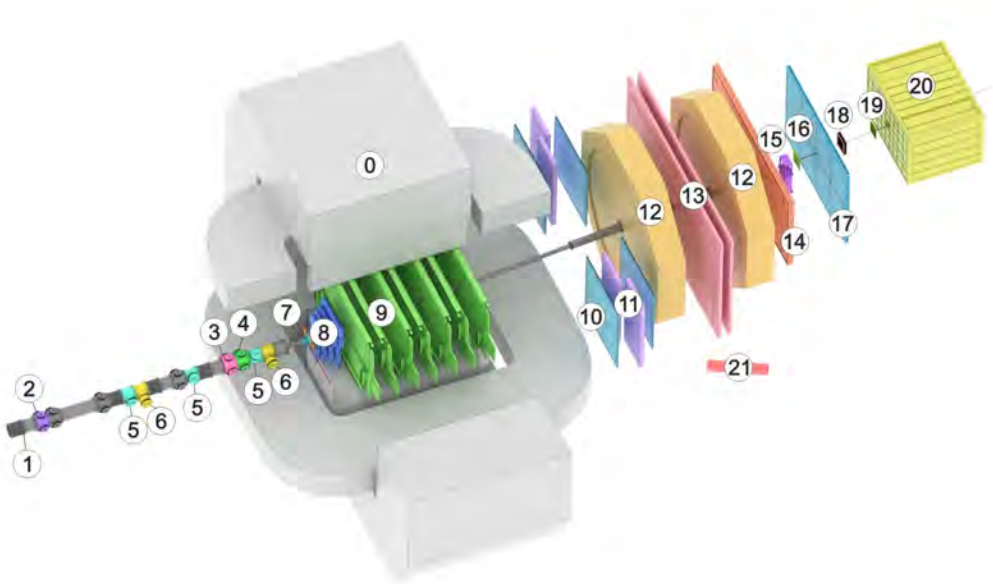
BM@N Distributed computing ~~readiness for Run 9~~

Konstantin Gertsenberger, Igor Pelevanyuk
LHEP MLIT

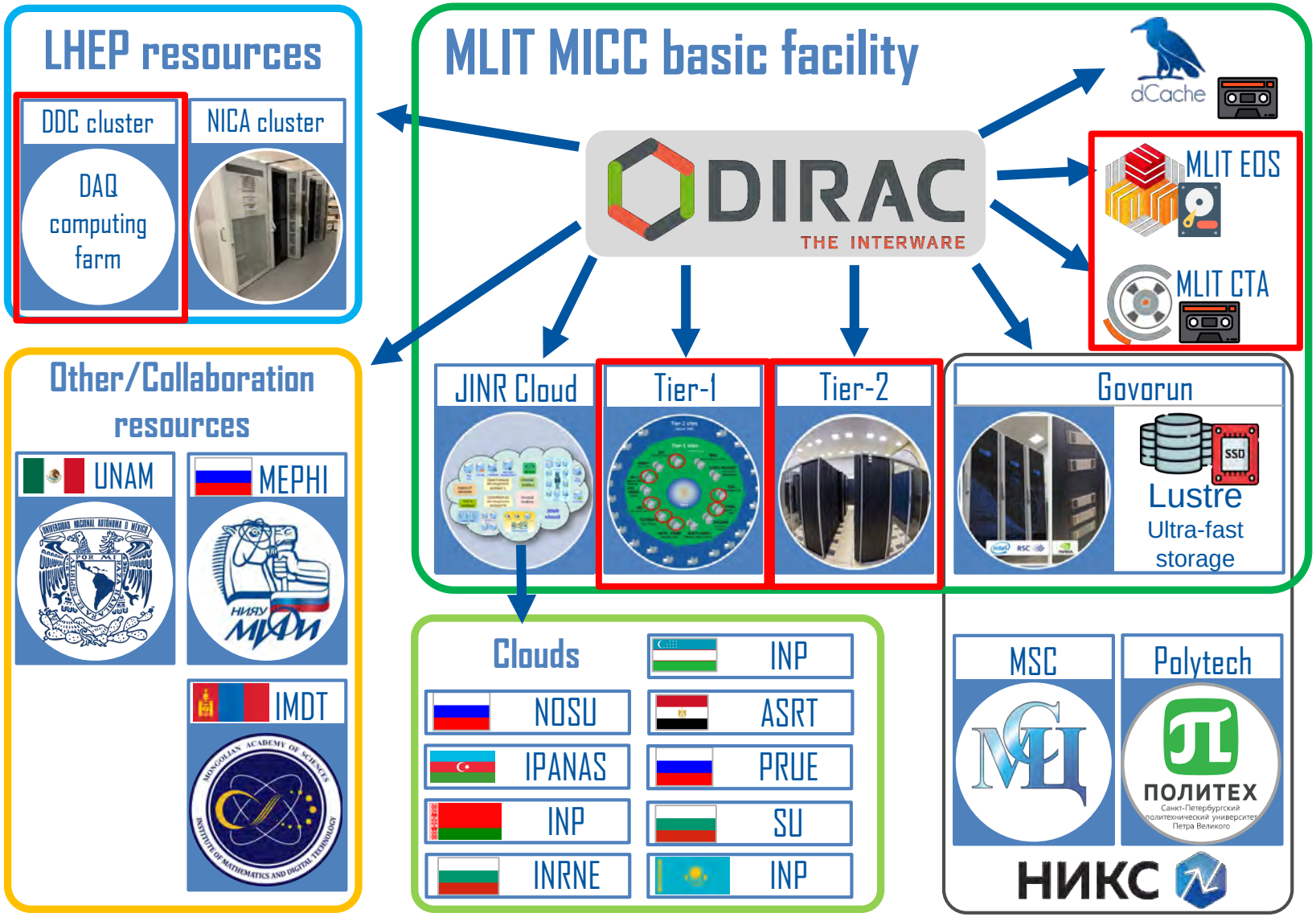


BM@N Distributed computing
~~readiness for Run 9~~
in the middle of Run 9

Konstantin Gertsenberger, Igor Pelevanyuk
LHEP MLIT



DIRAC in JINR



Requests

December 2025: **Request (Analyzis):** Convert to tree.root fr
Alexander Demanov
Done: ~28000 jobs, less than 1h per job. Results on NCX

3 days

January 2026: Request #12: DST Production for Run 9 (26.01.0) from
Sergey Merz
Done: 671 / 1732 files done.
A wake-up call – jobs failed lacking data from DB**.**
Issue with "Macro finished successfully".

11 days

February 2026: **Request # 13** DST Production for Run 9 (26.02.0) from
Sergey Merz
Done: … mostly. 35 files failed due to lacking data fr 4
runs.

~ 20 hours

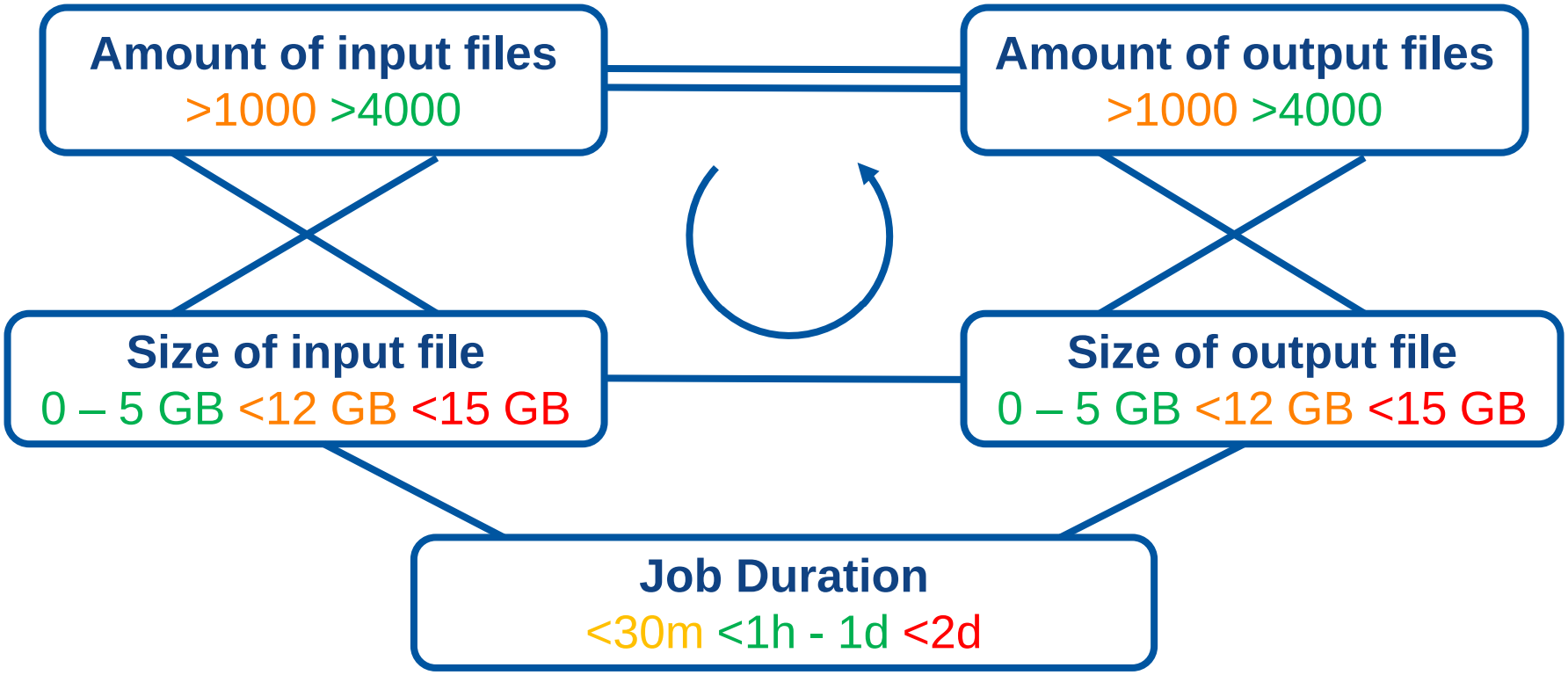
Requests

February 2026: **Request # 14 DST Production for Run 9 (2**
from Sergey Merz
Done: had to write the script to identify files that are lac data
from DB and avoid processing them.

~24 hours

And before preparation of production report I understood that we are right in the middle of Run 9! Yay!

For distributed processing



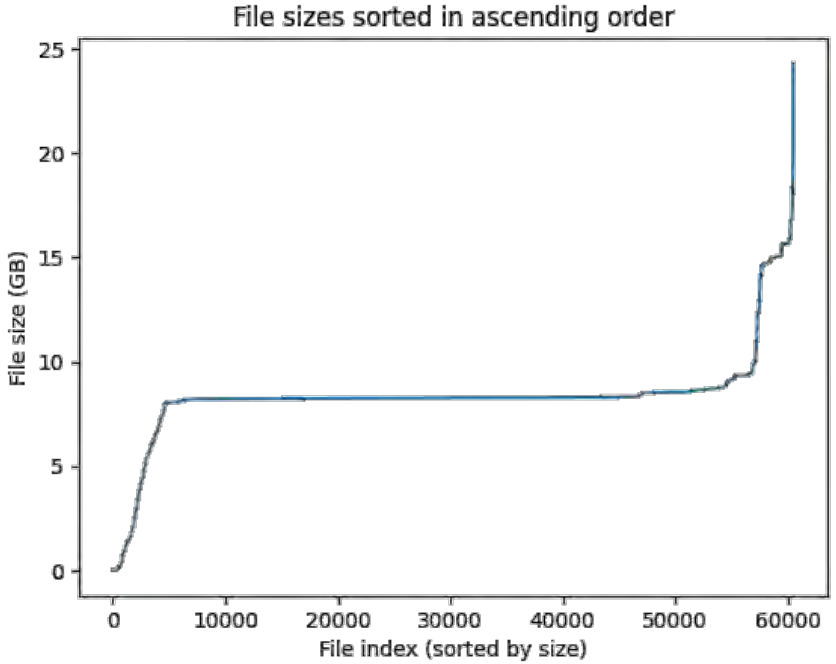
Comfortable size of RAW files is in range 7-12 GB.

Ideal size considering RAW **Job Duration** is 10 GB.

Run 9 in distributed infrastructure

some statistics at the moment

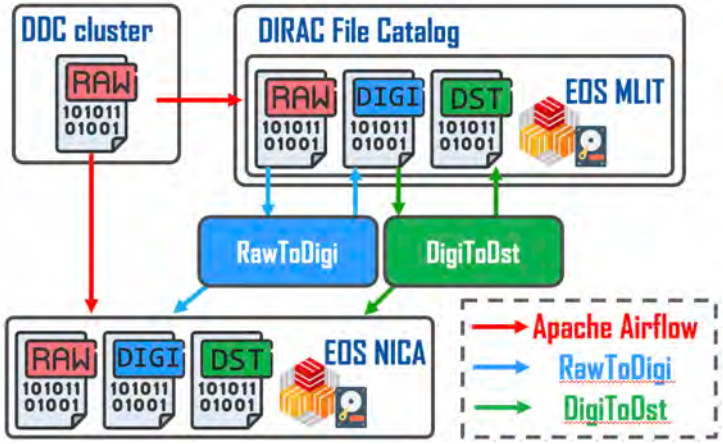
Files: **60476** (x2 comparing with Run8)
Total size: **505 TB** (x1.15 comparing with Run8)
RAW file size is varying between **0 and 24.29 GB**
80% of RAW files are between: **8.14 and 8.84 GB**
Avg RAW file size: **~8.36 GB**



Thanks to DAQ team and all responsible for almost twice smaller RAW files!

Automation (“Triggers+Observer”)

It was planned to make an automation using:
1. **Apache Airflow** for uploading data in DIRAC
2. **DIRAC Transformation System** mechanism (**DIRAC TS**).



- ✓ Tinker with **DIRAC TS** to allow transfer and job monitoring.
- ✓ Use DIRAC FileCatalog MetaData schema to apply to RAW, DIGI, DST
- ✓ Run artificial tests

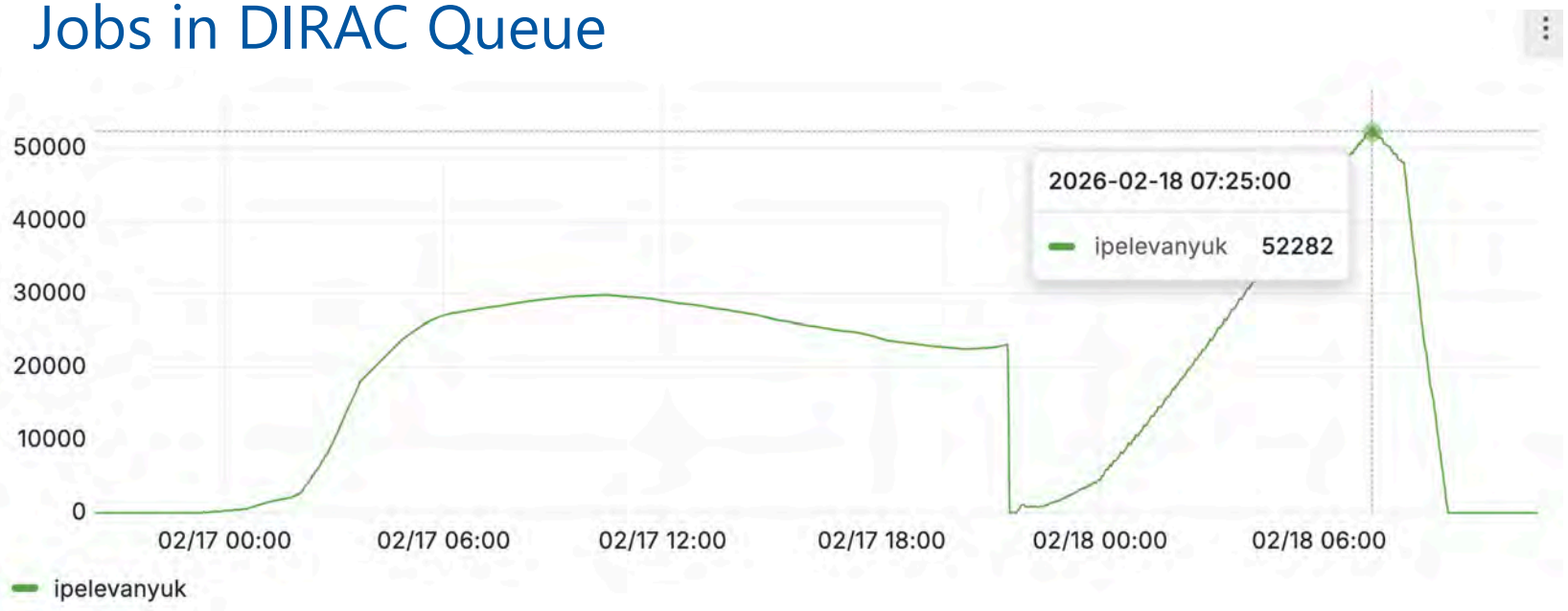
And since Apache Airflow was not available for data upload

- ✓ Perform manual data upload from DDC to DIRAC by DIRAC jobs.
- ✓ Apply metadata to RAW files in DIRAC.

And …

Result

Jobs in DIRAC Queue

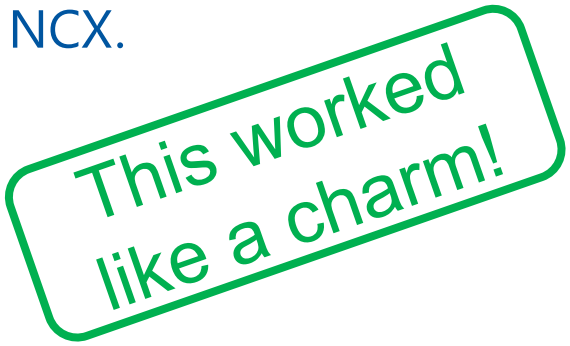


For some reason (mostly because of tinkering with DIRAC TS), once a lot of jobs are waiting in the queue, the system start submitting same jobs for the same files that are waiting to be processed, but only for waiting jobs.

So, plan B (”Ты водишь”)

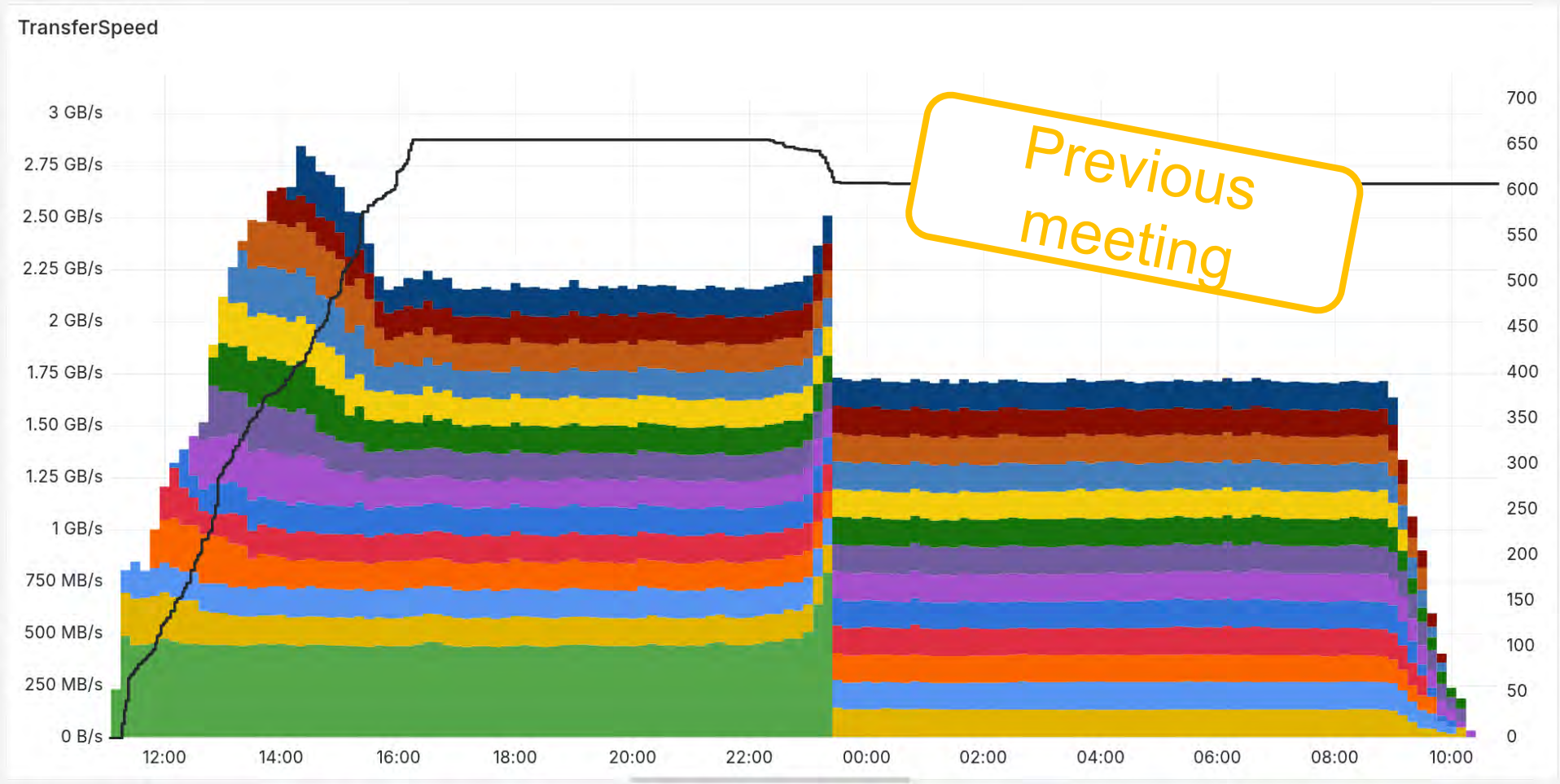
1. Turn of Transformations RawToDigi and DigiToDst.
2. Change data transfer tool to launch RawToDigi job schema within Transfer jobs for each transferred file.
3. Change RawToDigi job schema to behave in the following way in case of ”Macro finished successfully”:
   - Transfer DIGI file both to EOS MLIT and EOS NCX.
   - Initiate DigiToDst job.

4. Change DigiToDst to transfer DST file both to EOS MLIT and EOS NCX.



Once Airflow is working we can proceed with this approach and avoid delay caused that we would not like to spam DDC Ceph file scan too much.

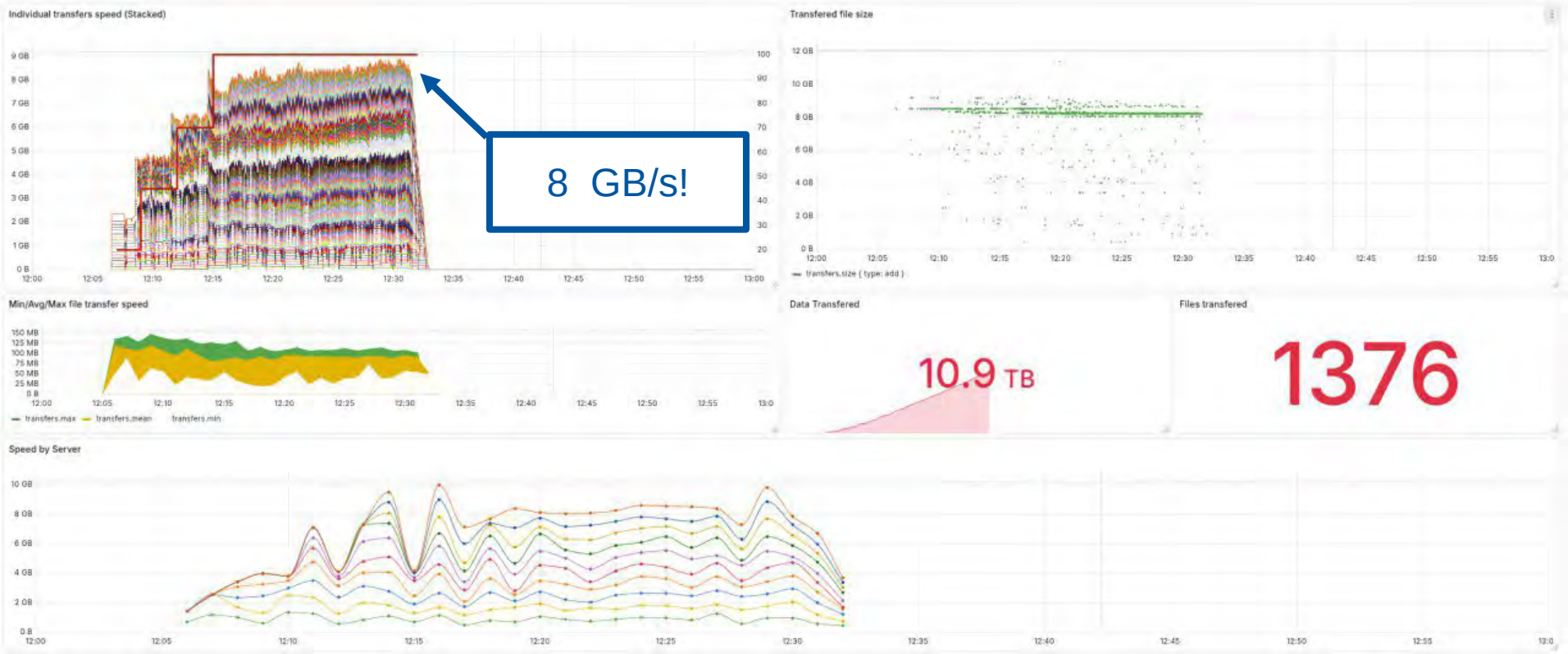
EOS -> DDC speed test



Black line – jobs running on DDC

Colored bars – transfer speed by whole node

Exceptional performance of DDC!



8 GB/s achieved between DDC Ceph and EOS MLIT.
100 transfer job were submitted to DDC with memory request that ensures 10 jobs per node.

Thanks to Ilia Slepnev for cooperation!

Directory structure

On DDC

`…/run9/**9623**/run_Top_9623_ev2_p20.root`

Better performance in run9 directory on Ceph and EOS.

Looks like it is critically important.
But to get all list of files users still have to use:

```
ls -la ./run9/*
```

or

```
find ./run9 -type f -depth 2
```

On DIRAC

`…/run9/run_Top_9623_ev2_p20.root`

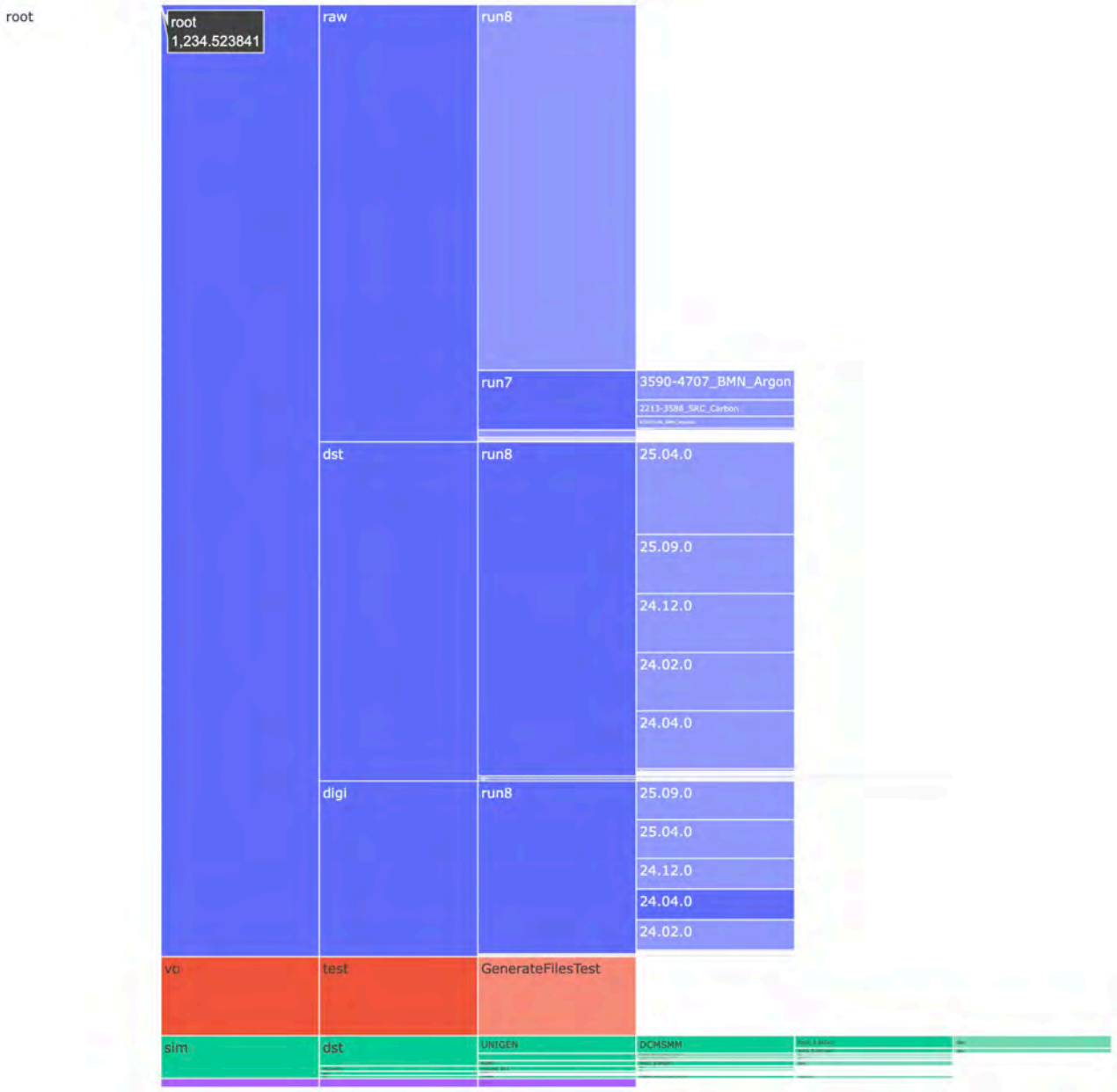
Optimized to keep hundred thousands file records in a single directory.

DIRAC FileCatalog can handle files in additional directories.

But **within DIRAC FileCatalog it is impossible to "move" files**. Only copy and remove.

Makes it incredibly more complex to build and visualize directories size and structure.

BM@N directories structure



Issues

lacking data from DB:
It is important to have data in uni-db as soon as possible.
Now the issue is mostly resolved from the last tests: only 19 runs lack information from 726.

Space for EOS in MLIT is over:
Waiting to be resolved. All requests were sent.

List of participants

DIRAC: Igor Pelevanyuk, Igor Zhironkin
BM@N: Konstantin Gertsenberger, Ilia Romanov

Responsible for resources:
Tier-1,Tier-2, EOS: Valery Mitsyn
CTA Tape library: Vladimir Trofimov
Govorun: Dmitry Podgainy, Dmitry Belyakov, Aleksandr Kokorev, Maxim Zuev
NICA cluster: Ivan Slepov
DDC cluster: Ilia Slepnev
Network: Andrey Dolbilov

Conclusion for BM@N ADM 2026

- We were successfully processing data from Run9. Up until the moment when EOS quota for BM@N was over.
- Initial plan for automation worked, but mess statistics with a lot of jobs that should appear in the first place.
- Change in automation schema allowed to provide almost immediate data processing.
- But there is no sense in this type of automation if jobs fail anyway because there are not data about run number or electromagnetic field in database.
- And we hope for soon solution with EOS quota in MLIT


JINR
MLiT

DIRAC directory structure

```
bmn.nica.jinr
├── exp
│   ├── raw
│   │   └── run[number]
│   ├── digi
│   │   └── run[number]
│   │       └── [version]
│   └── dst
│       └── run[number]
│           └── [version]
├── sim
│   ├── gen
│   │   └── [generator]
│   │       └── [campaign]
│   └── dst
│       └── [generator]
│           └── [campaign]
│               └── [version]
└── backup (on CTA tape library)
    ├── exp (same structure as in exp dir)
    └── sim (same structure as in sim dir)
```

Directory structure in DIRAC is now similar to NCX directory structure.

Still requires some work to do, to make it absolutely the same.

DIRAC directory structure

Special metadata applied for run9 data directories. This is crucial for further automatic processing!

```
bmn.nica.jinr
└── exp
    ├── raw {type: raw}
    │   └── run9 {run_period: 9}
    ├── digi {type: digi}
    │   └── run9 {run_period: 9}
    │       └── 26.02.0
    └── dst {type: dst}
        └── run9 {run_period: 9}
            └── 26.02.0
```

With that schema it is possible to select files by metadata:

```
$ dirac-dms-find-lfns type=raw run_period=9
/bmn.nica.jinr/exp/raw/run9/file_1_v0_p0.data
/bmn.nica.jinr/exp/raw/run9/file_2_v0_p0.data
/bmn.nica.jinr/exp/raw/run9/file_3_v0_p0.data
...
```