Queue-based job monitoring

ATLAS Belle II Computing Meeting

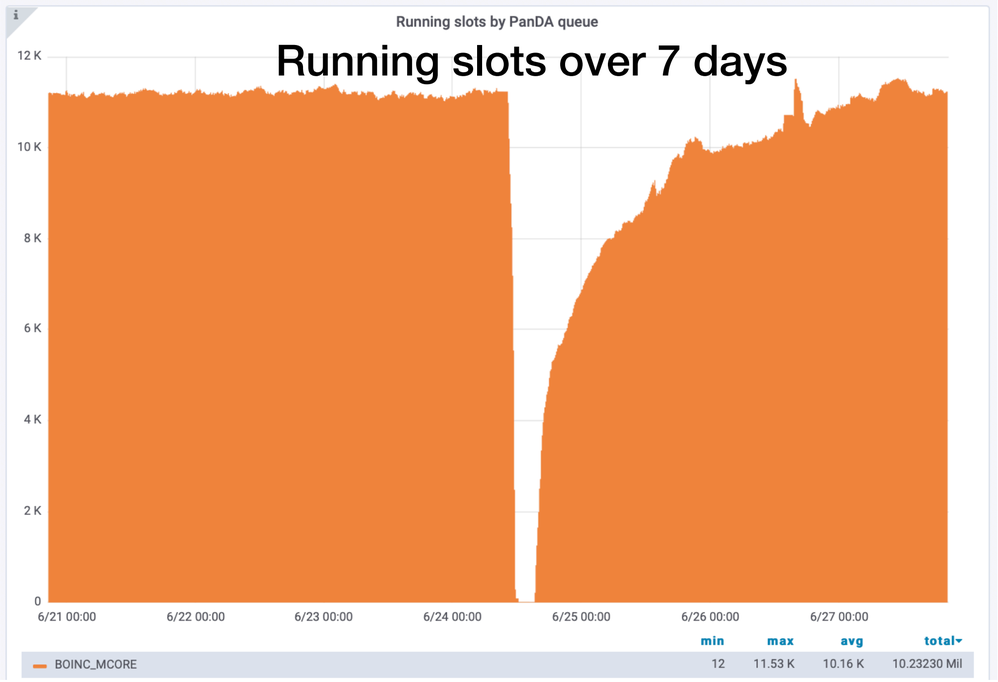
eric.schanet@cern.ch

Eric Schanet | LMU Munich | 12.08.2019

Introduction

- **Central problem is one of the long-standing problems in ADC**
  - Why **specific variations in the numbers of running slots**?
  - Want to make sure that we are not wasting resources due to e.g. one FTS server not working as expected.
  - Need to make sure we detect problems as fast as possible.

BOINC_MCORE



- **Complex and multi-dimensional problem**
  - Need to correlate things like e.g. pilot submissions, downtimes, failures, transfers, …, with e.g. activated and running jobs.
  - Many attempts made in the past, still no satisfying answer.

- **Slightly different approach this time**
  - **General idea:** start from something simple and then, bit-by-bit, add more information and correlations.
  - **Keep it lightweight:** only add feature if significant benefit for operational monitoring.
  - **In principle all the information is already available** (somewhere), only need to combine and correlate it, but also display in smart and easy-to-digest way.

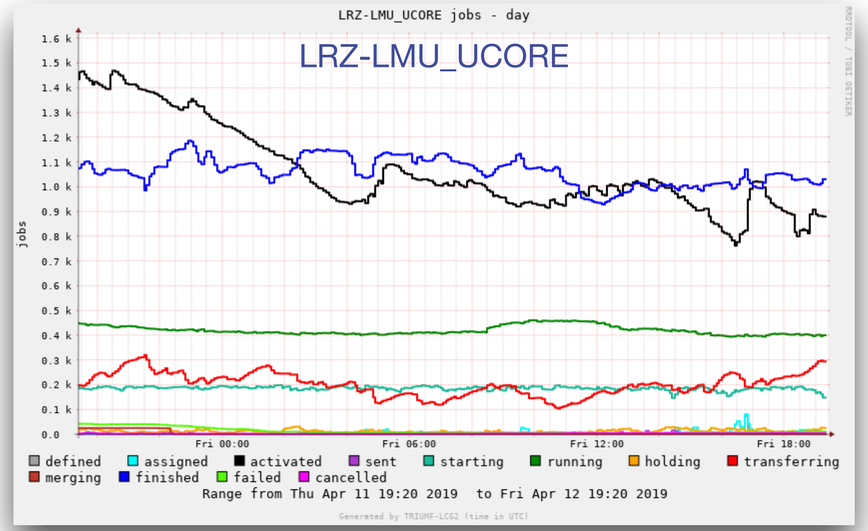
Reproducing Panglia at CERN

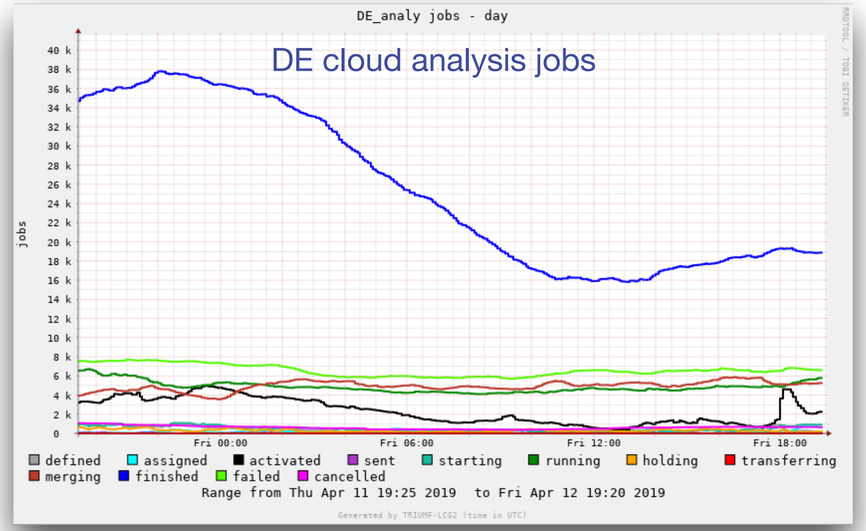
- **Existing services are not ideal in this case**
  - **Panglia:** low-latency but limited functionality
    - ➡ Need more features in the data structure.
  - **MONIT:** complex data structure but aggregated job states and limited flexibility (for good reason)
    - ➡ Need low latency and more flexibility when adding data features.

➡ Build **lightweight** and **low-latency** service using Grafana and fully controlled by ADC.

- **First step**
  - Reproduce Panglia at CERN.
  - Panglia plots: jobs in a queue by job state (low-latency).

Typical Panglia plots

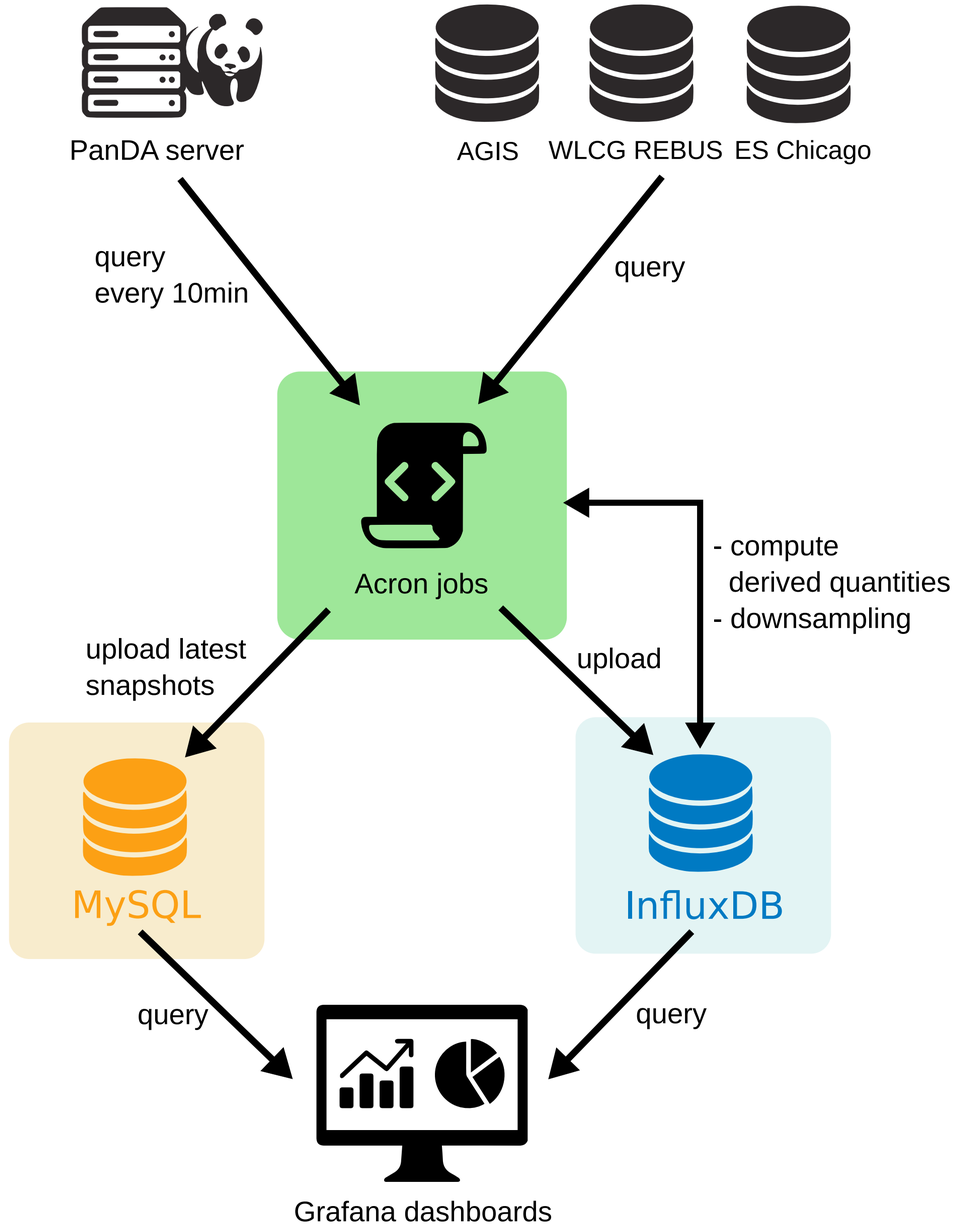




Technical setup

- **Main database: InfluxDB, 300GB size, hosted through DBoD**
  - Filled through Panda Client, every 10min.
  - Only queue-level data, no job-level data (low latency!).
  - Information from AGIS and WLCG REBUS, and ES Chicago.
  - Downsampling (handled offline):
    - 10min granularity for 7 days,
    - 1h granularity for 2 months,
    - 1d granularity for 1 year.
  - Derived quantities mostly computed offline (e.g. moving average).
    ➡ Prevents need for expensive on-the-fly computation.

- **Second database: MySQL, 300GB size, hosted through DBoD**
  - Used for non time-series plots or tables.
  - Contains latest snapshots of InfluxDB data.
  - Keeps some more load off InfluxDB.



Data structure in InfluxDB

- **From Panda:**
  - Number of jobs per Panda queue (no job-level information!)
  - For each Panda queue: One data point per job state and per resource type (SCORE, MCORE, etc.).

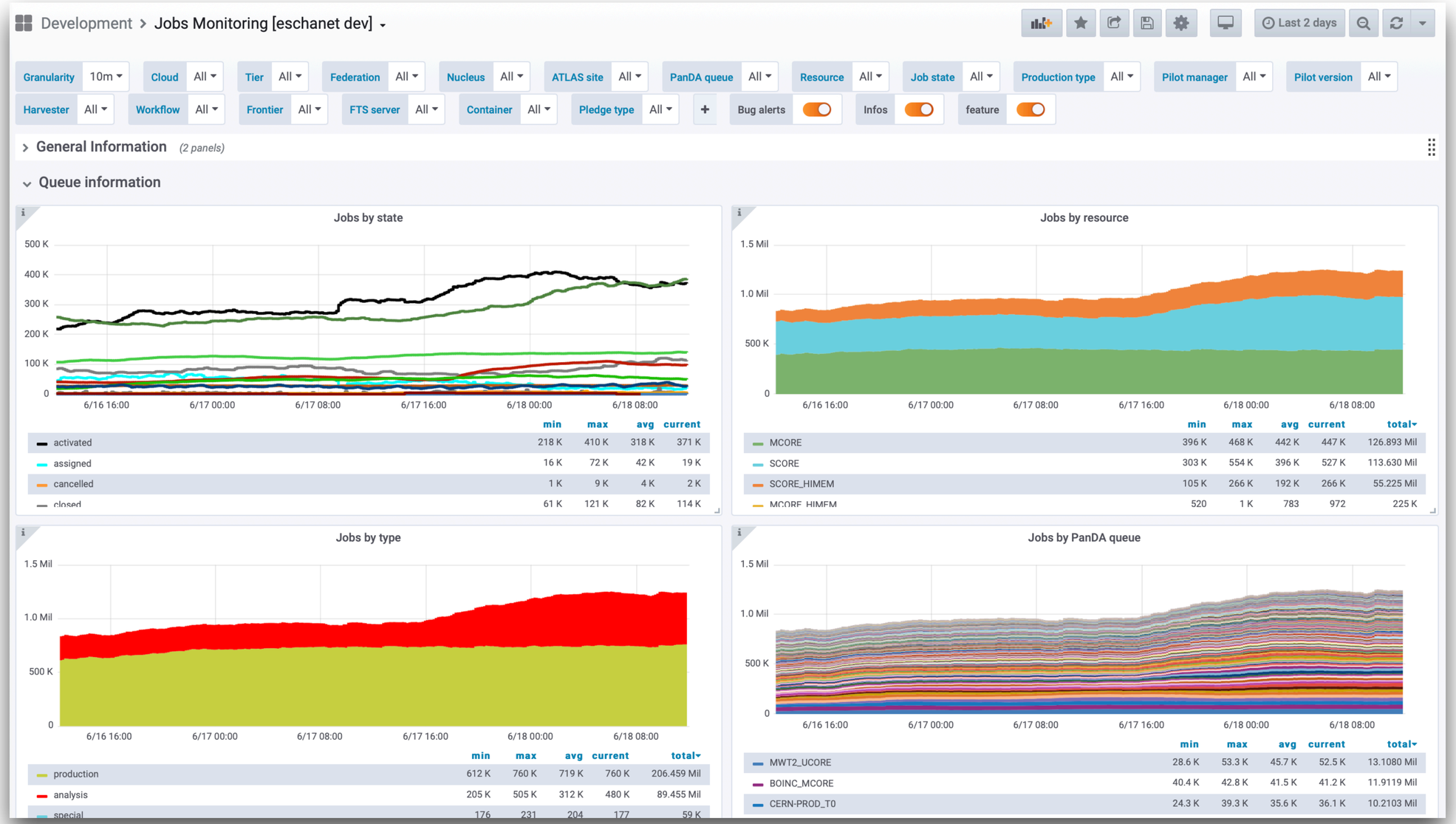
- **From AGIS:**
  - For each Panda queue: Cloud, Tier, Nucleus, ATLAS site, production type, pilot manager, pilot type, harvester instance, harvester workflow, frontier, FTS server, container mode.

- **From REBUS**
  - For each Panda queue: Federation, pledge and pledge type of federation.

- **From ES Chicago**
  - Benchmark job results (measured HS06) per PQ.

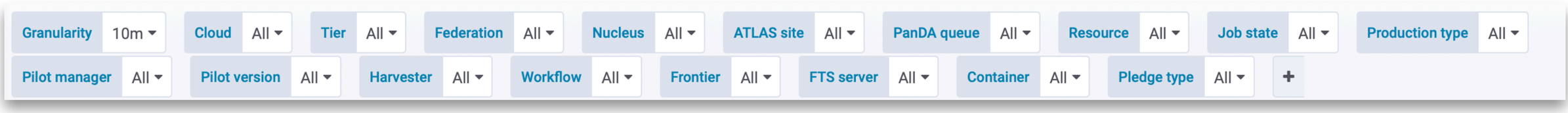
Jobs: Overview

- Queue-based low-latency job monitoring dashboard built with InfluxDB and MONIT grafana.

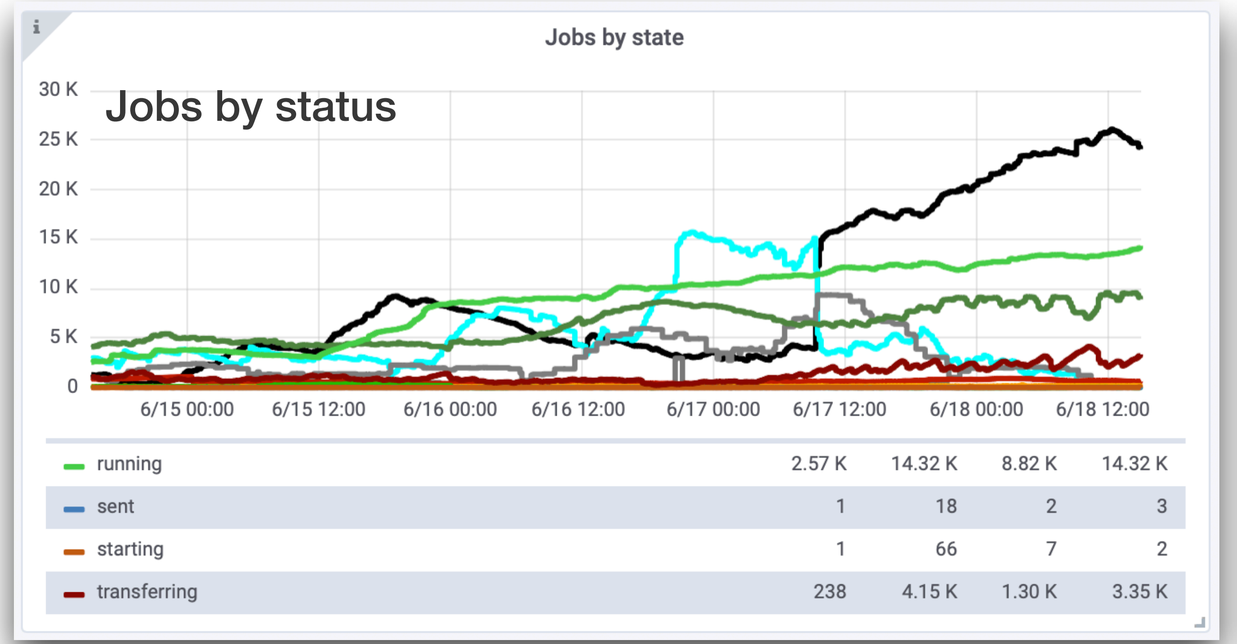


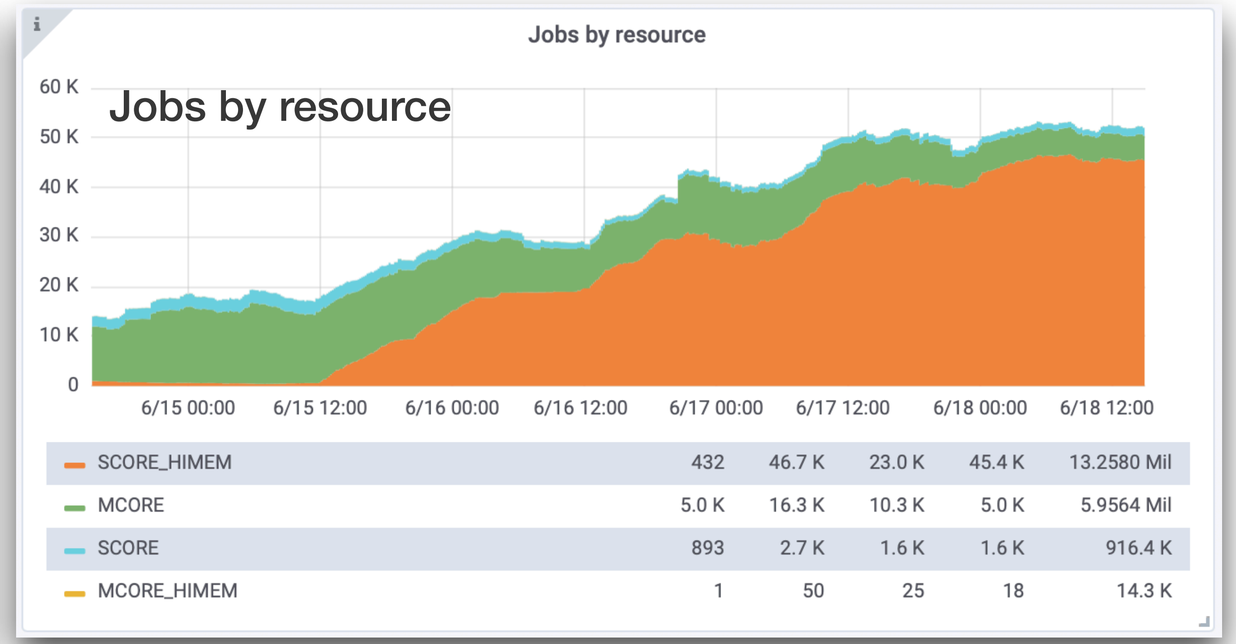
Jobs: Only Panglia?

- **Interactive version of the Panglia plots**
  - Freely select time range.
  - Large number of parameters can be selected.



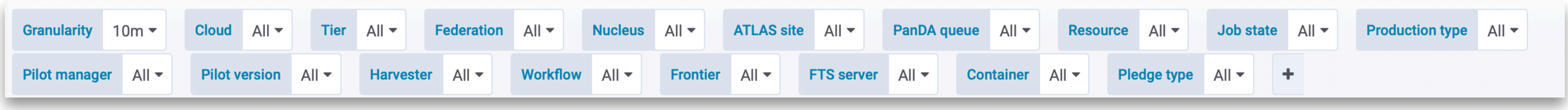
- **Typical Panglia-style plots are possible, all in low latency and high granularity.**



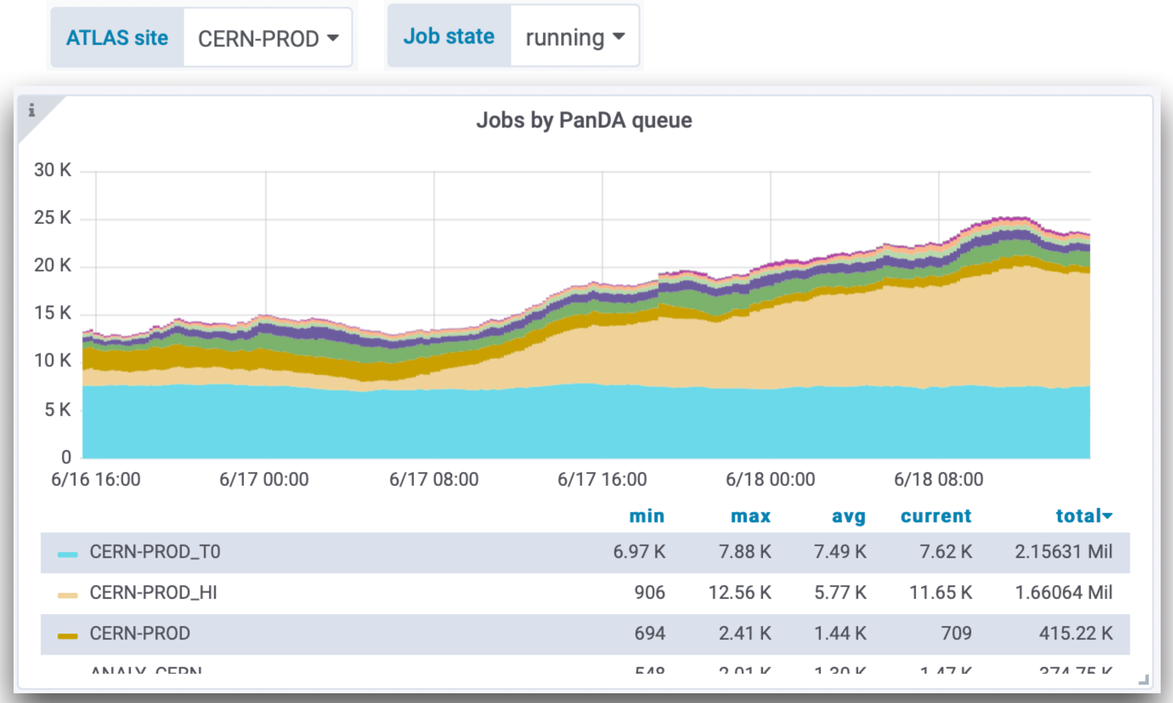


Jobs: More than Panglia!

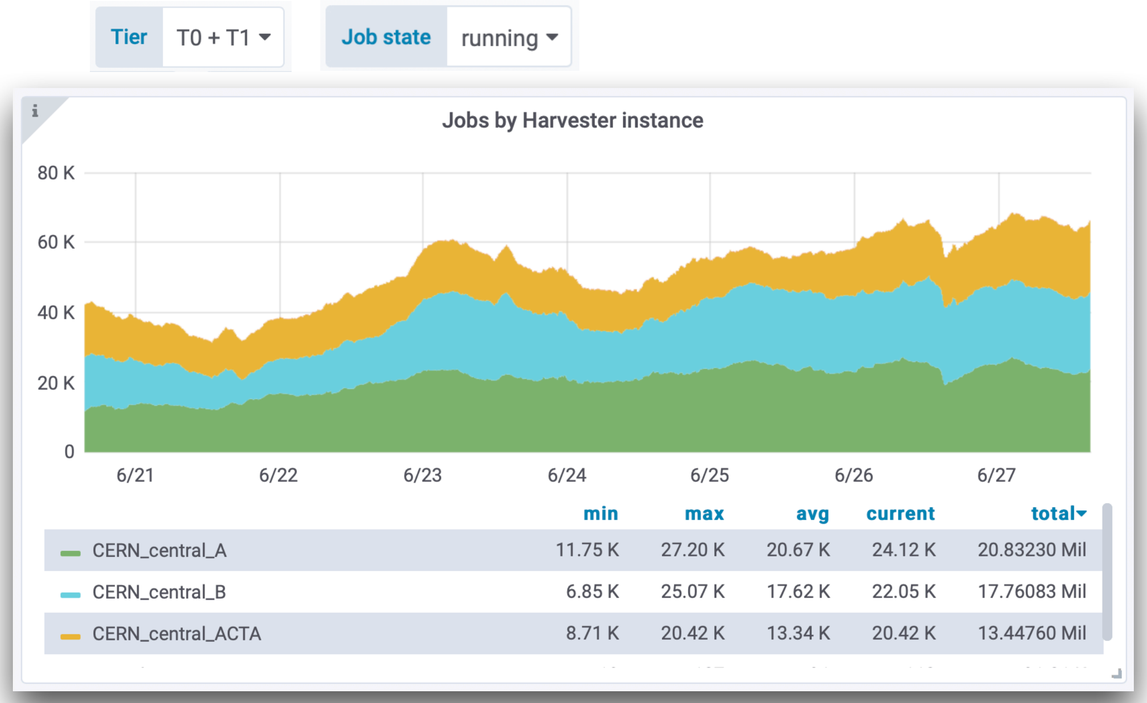
- **Lots of possibilities**
  - One plot per parameter (jobs grouped by that parameter).
  - Grafana: user can select multiple values per parameter → huge number of selections can be visualised.



Running jobs on CERN-PROD site grouped by Panda queue



Running jobs in T0+T1 grouped by Harvester instance



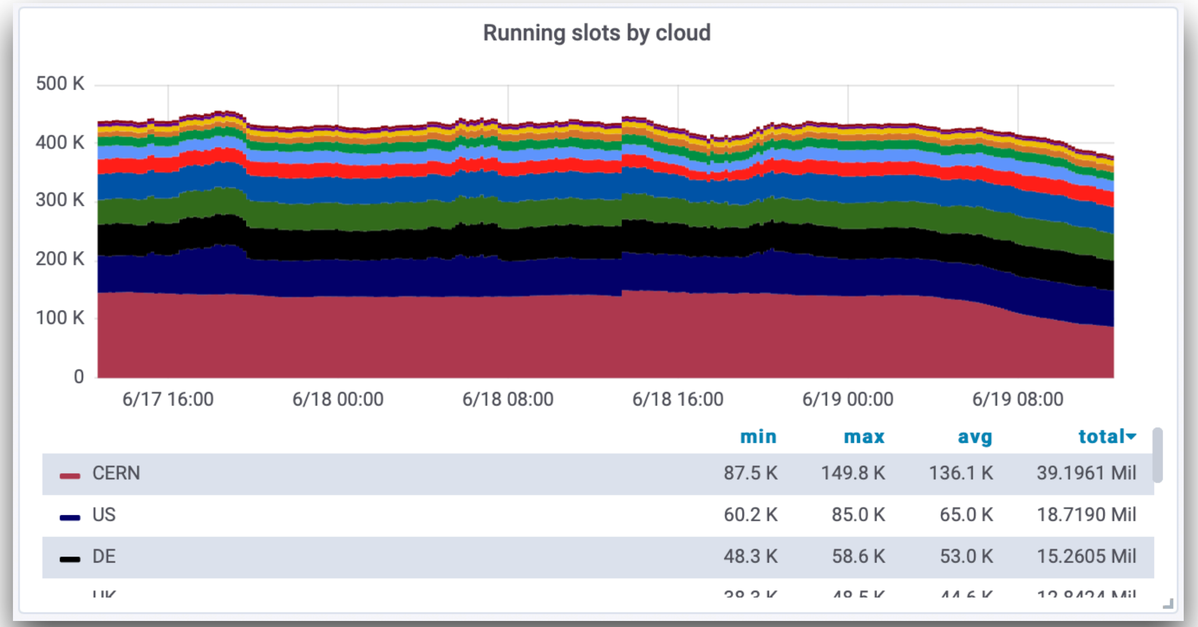
Slots: Overview

- **Also plots showing actual slots of jobs (including number of cores a job runs on)**
  - Same low-latency and high-granularity as for jobs.
  - Also grouped by different parameters in different plots.
  - Number of slots is calculated using number of cores per queue published on AGIS (respecting the resource type).

- **Examples:**

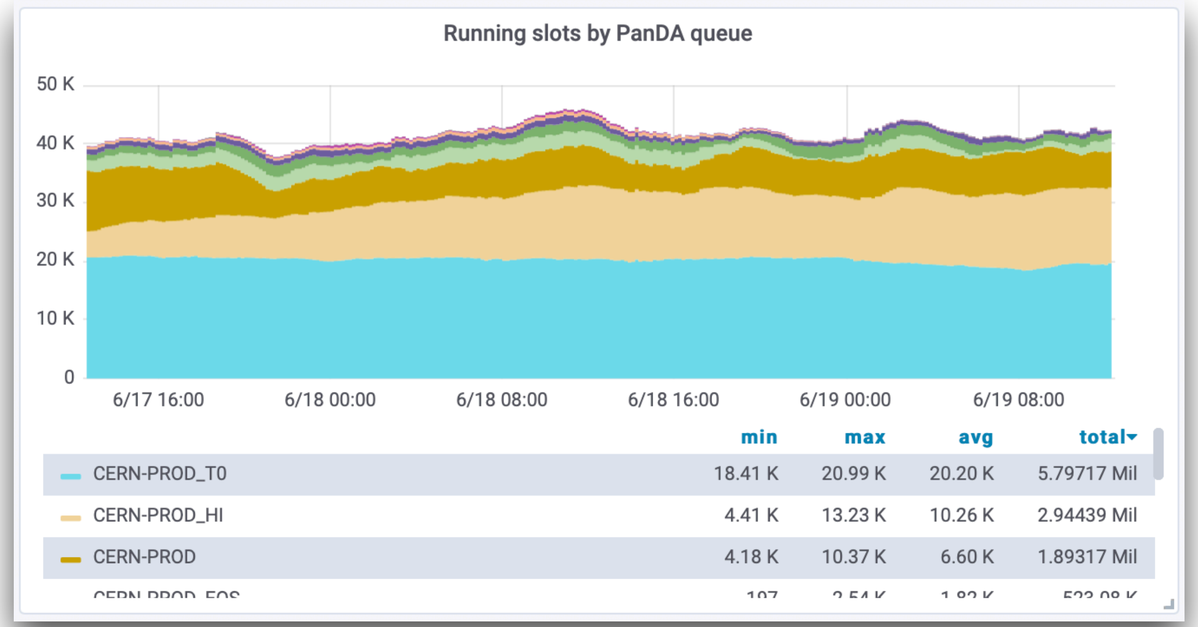
Cloud All

Slots of running jobs grouped by all clouds



ATLAS site CERN-PROD

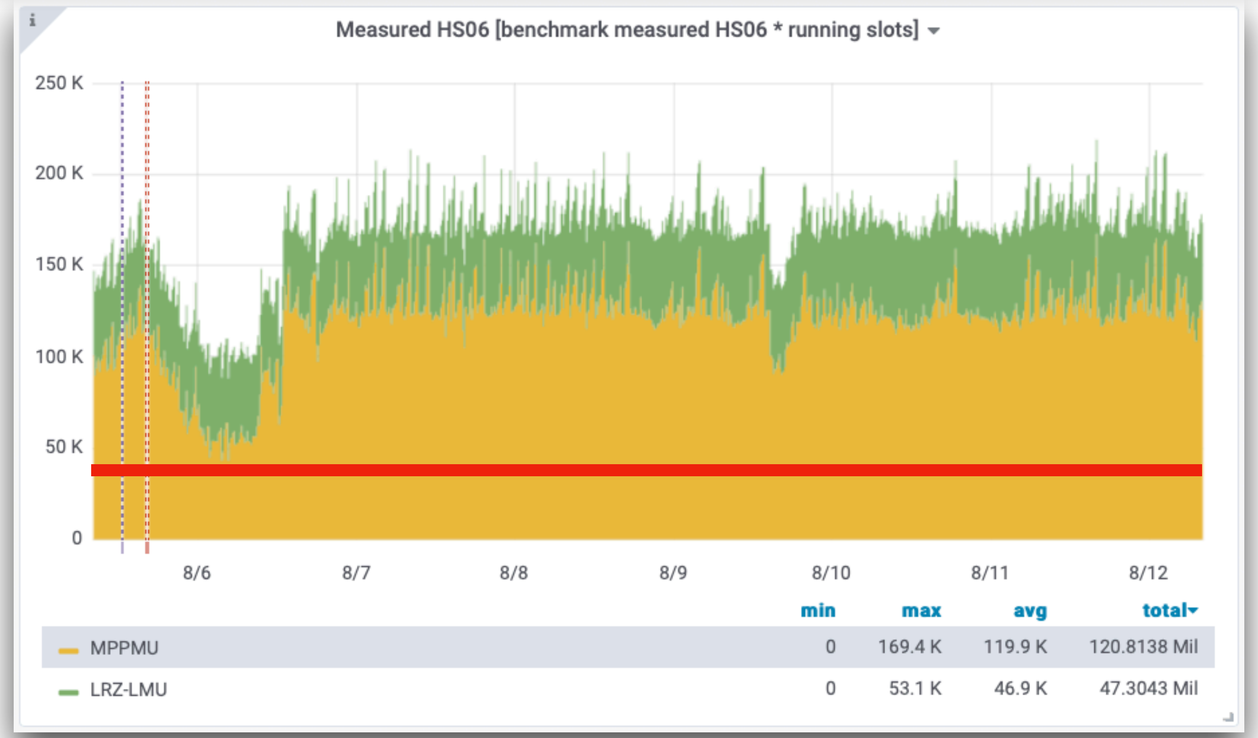
Slots of running jobs grouped by all queues in CERN-PROD



HS06: Do we get what we were promised?

- **Corepower exists already in AGIS**
  - However, often not very exact, sometimes even off by an order of magnitude
- **Use results from benchmark jobs that randomly run on each PQ**
  - Stored in ES Chicago.
- **Compare with pledges reported in WLCG REBUS per federation.**

Tier 2	Germany	ATLAS Federation, Munich	CPU (HEP-SPEC06)		43,066

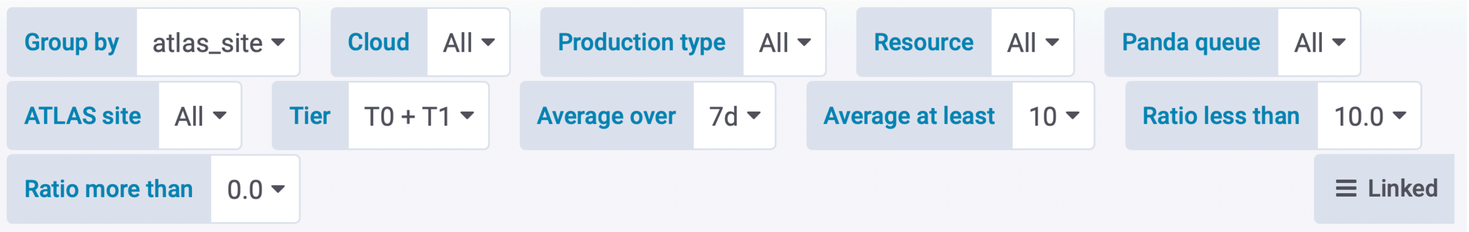


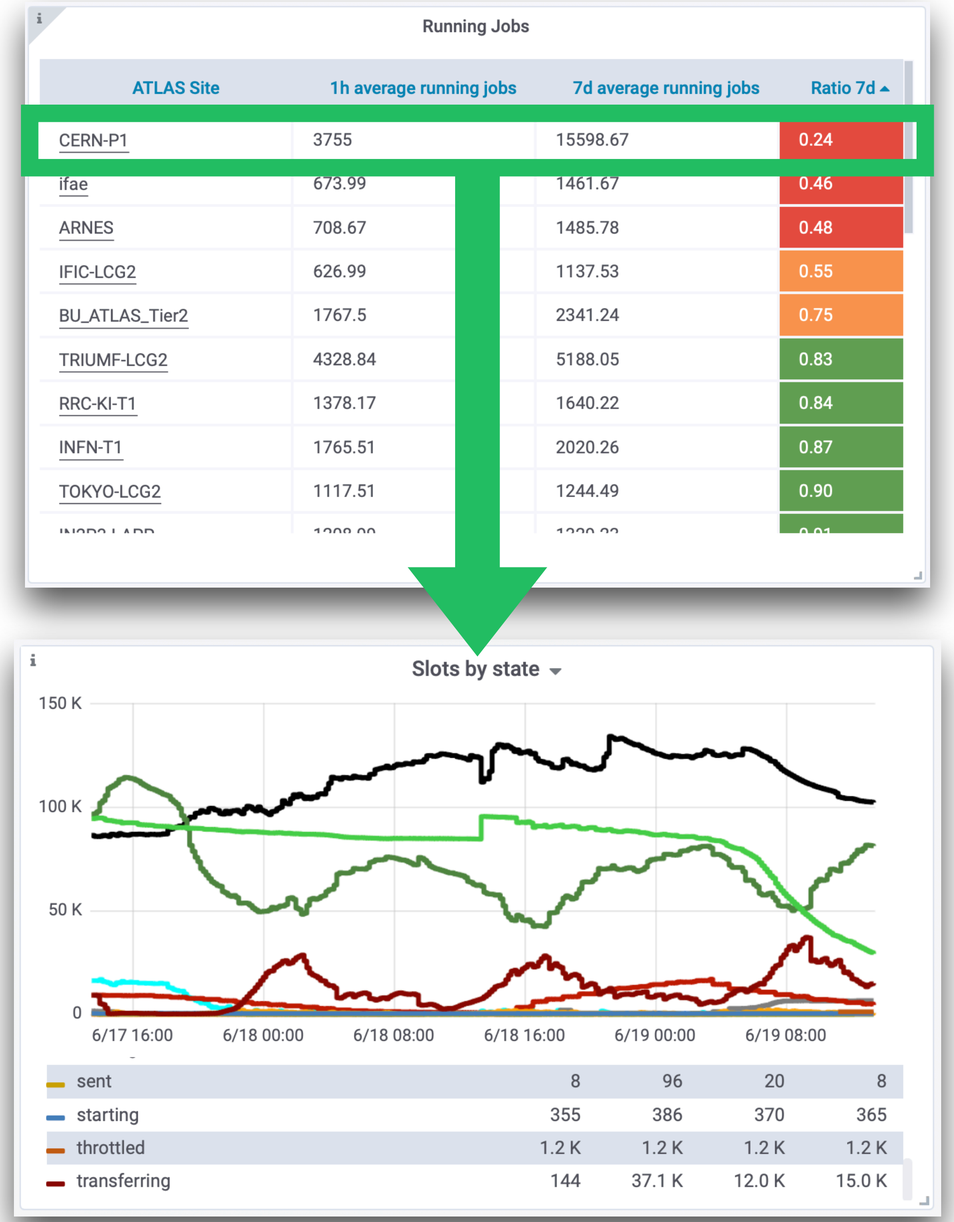
Detecting sudden changes

- **Low-latency perfect for detecting sudden changes, empty queues, etc.**
  - Suspicious sites dashboard compares ratios between, e.g.:
    - Moving 1h average of running slots (other averaging periods possible).
    - Moving 7d average of running slots (other averaging periods possible).
    - (Many more panels …)

- **Example on the right:**
  - Top table: showing ratios between running jobs of 1h and 7d.
  - CERN-P1 shows relatively low ratio (only 25% of running jobs currently compared to last 7d).
  - Click on CERN-P1: job monitoring page for site is opened.

- Selections are possible, e.g. looking only at T0+T1 sites.





Summary and outlook

- **Queue-based monitoring using MONIT Grafana and an InfluxDB hosted at CERN:** job monitoring dashboard
  - Interactive Panglia with lots of additional data features.
  - Based on queue-level data from Panda (through Panda Client Interface).
  - Integrates information from AGIS, REBUS and ES Chicago.

- **First attempt of creating something more automatic:** suspicious sites dashboard
  - Using simple metrics like e.g. ratios of average numbers of jobs in order to detect sudden changes.

- **Dashboard are already available in the development area of MONIT grafana.**
  - Production versions will become available soon.

BACKUP

General details

- **Git repository for InfluxDB code**
  All the code related to the data uploaded to the InfluxDB instance is being tracked in this repository.

- **Tracking of general progress**
  Trying to track the progress of the qualitask in a Google Doc.

- **Grafana dashboards**
  Dashboards are being built with Grafana, available through central monit-grafana.
  Jobs monitoring: https://monit-grafana.cern.ch/d/IGWcOe8iz/jobs-monitoring-eschanet-dev?orgId=17
  Suspicious sites: https://monit-grafana.cern.ch/d/nEL8aDumk/suspicious-sites-eschanet-dev?orgId=17
  DAOD distribution: https://monit-grafana.cern.ch/d/HAN2MQeiz/daod-distribution-eschanet-dev?orgId=17

- **Bi-weekly meetings**
  Typically on Fridays, 2 p.m., last one here, they usually pop up in this list

- **Supervisors**
  Technical supervisors: Ivan Glushkov (primary), Frank Berghaus (secondary)
  Local supervisor: Günter Duckeck

Side project: DAOD distribution for analysis jobs

- **Problem: Are we placing DAODs optimally in the Grid?**
  - Are there sites with a lot of available analysis slots but not enough DAODs? → Want a balanced situation!
  - Are there sites with a lot of DAODs but no available analysis slots?

- **Dashboard attempting to analyse and visualise this:** DAOD distribution dashboard
  - Showing size and number of files (only DAODs) on datadisks.
  - Summing up all running slots (averaged over last 24h) that are reading from each datadisk.
  - Computing ratio between running slots and size of datadisk (only DAODs) as a metric for how balanced it is.
  - Also showing a plot of the distribution of the ratios for all datadisks.

Datadisk	Running slots (average 24h)	Size (TB)	Files	Running slots / Size ▲
DESY-ZN_DATADISK	81.77	372.13	302.99 K	0.22
GOEGRID_DATADISK	196.69	854.50	824.87 K	0.23
FZK-LCG2_DATADISK	820.82	3.35 K	2.42 Mil	0.24
FMPHI-UNIBA_DATADISK	110.66	432.54	437.19 K	0.26
MPPMU_DATADISK	162.40	437.16	315.67 K	0.37
CYFRONET-LCG2_DATADISK	115.11	309.02	330.98 K	0.37