SwissFEL IT Infrastructure

Overview 12/12/2022

Alvise Dorigo / AWI Dept.

SwissFEL Hardware Recap

- ## Experimental Storage w/ GPFS
  - Aramis: Lenovo DSS-G240 – 2200 TiB online/cache net space (after parity)
  - Athos: Lenovo DSS-G220 – 1400 TiB online/cache net space (after parity)
- ## Camera Storage w/ GPFS
  - Lenovo DSS-G220 – 1800 TiB net space (after parity)
- ## Compute
  - 20 nodes, 800 cores, 5.6TiB RAM, IB connections for online fast data analysis
  - 2 more nodes coming (1 TiB RAM, AMD CPU, 112 cores, IB)
- ## DAQ (detector+camera)
  - 18 nodes in prod (more for tests), 816 cores, 7.9 TiB RAM, IB EDR, 100GbE
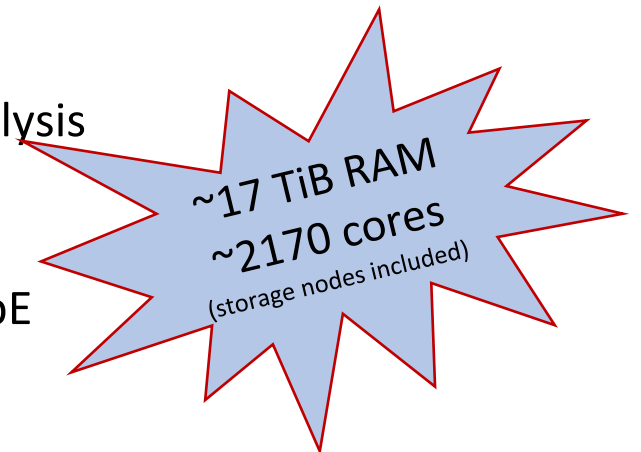- ## Export
  - 6 nodes, 2(NFS) + 4(AFM)
    - *AFM is the GPFS software component (redundant, 2 nodes per exp.) that keeps online data synchronized to RA*
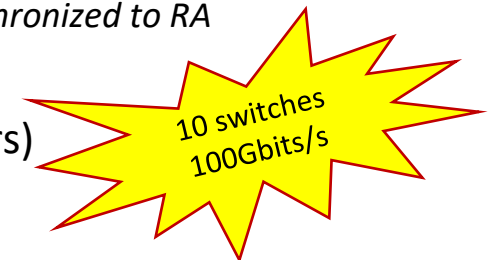- ## Interconnect
  - 7 Ethernet 100GbE switches (interconnection between exp. hutches and DAQ servers)
  - 3 InfiniBand EDR (100G) switches (fast access to storage from Compute and DAQ)

Tot 5.4PiB net

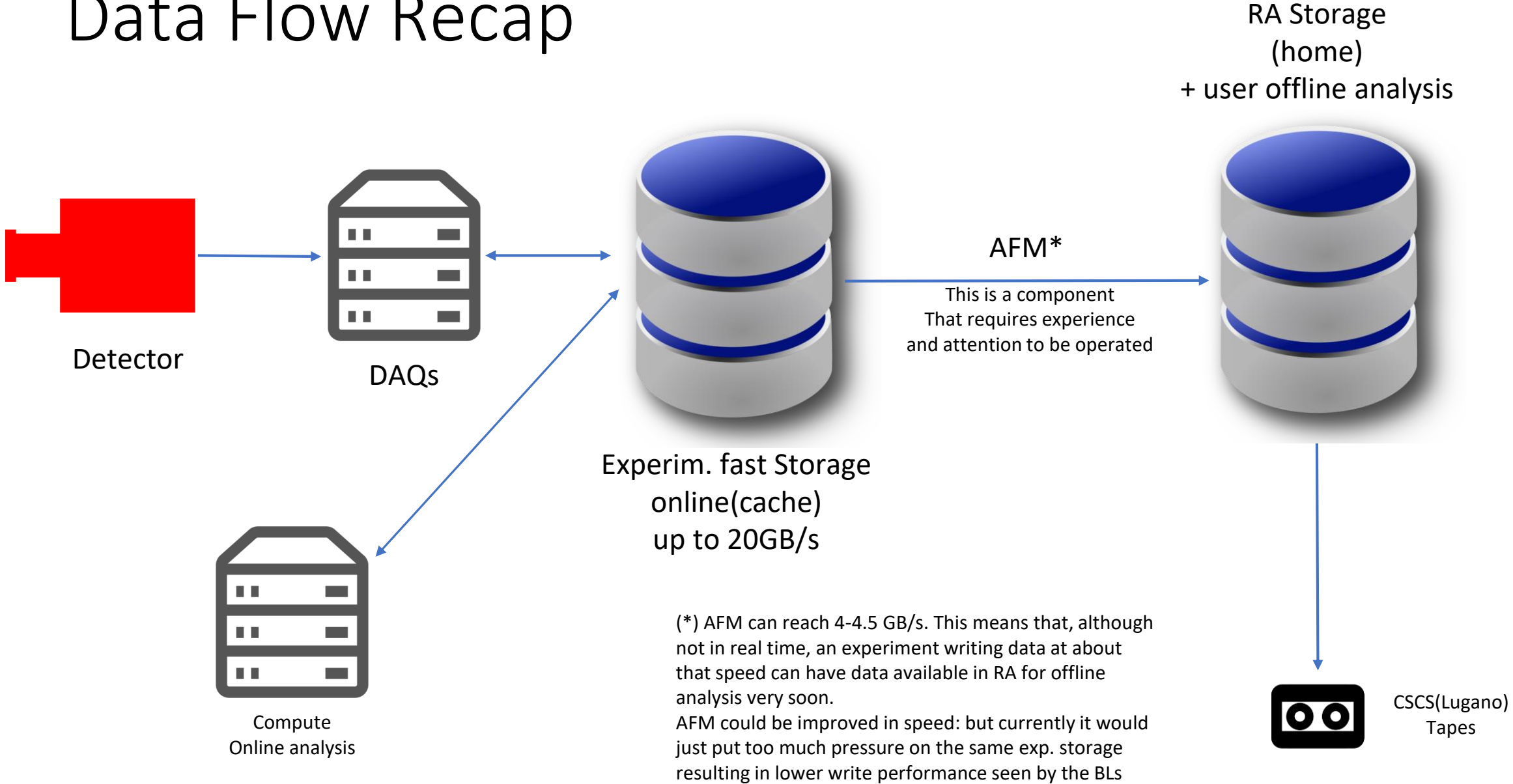~17 TiB RAM
~2170 cores
(storage nodes included)

10 switches
100Gbits/s

Hardware Procured in 2022

- 3 DAQ servers (2 repl. + 1 for Furka beamline)
- 2 Camera Writers (repl.)
- 1 DAQSync (for Cristallina beamline)
- 2 compute nodes (half Apollo HPE enclosure)
- Much various hardware (for interconnections and local storages)
- Lots of warranties to renew

Data Flow Recap

Detector

DAQs

Experim. fast Storage
online(cache)
up to 20GB/s

AFM*

This is a component
That requires experience
and attention to be operated

RA Storage
(home)
+ user offline analysis

Compute
Online analysis

(*) AFM can reach 4-4.5 GB/s. This means that, although
not in real time, an experiment writing data at about
that speed can have data available in RA for offline
analysis very soon.
AFM could be improved in speed: but currently it would
just put too much pressure on the same exp. storage
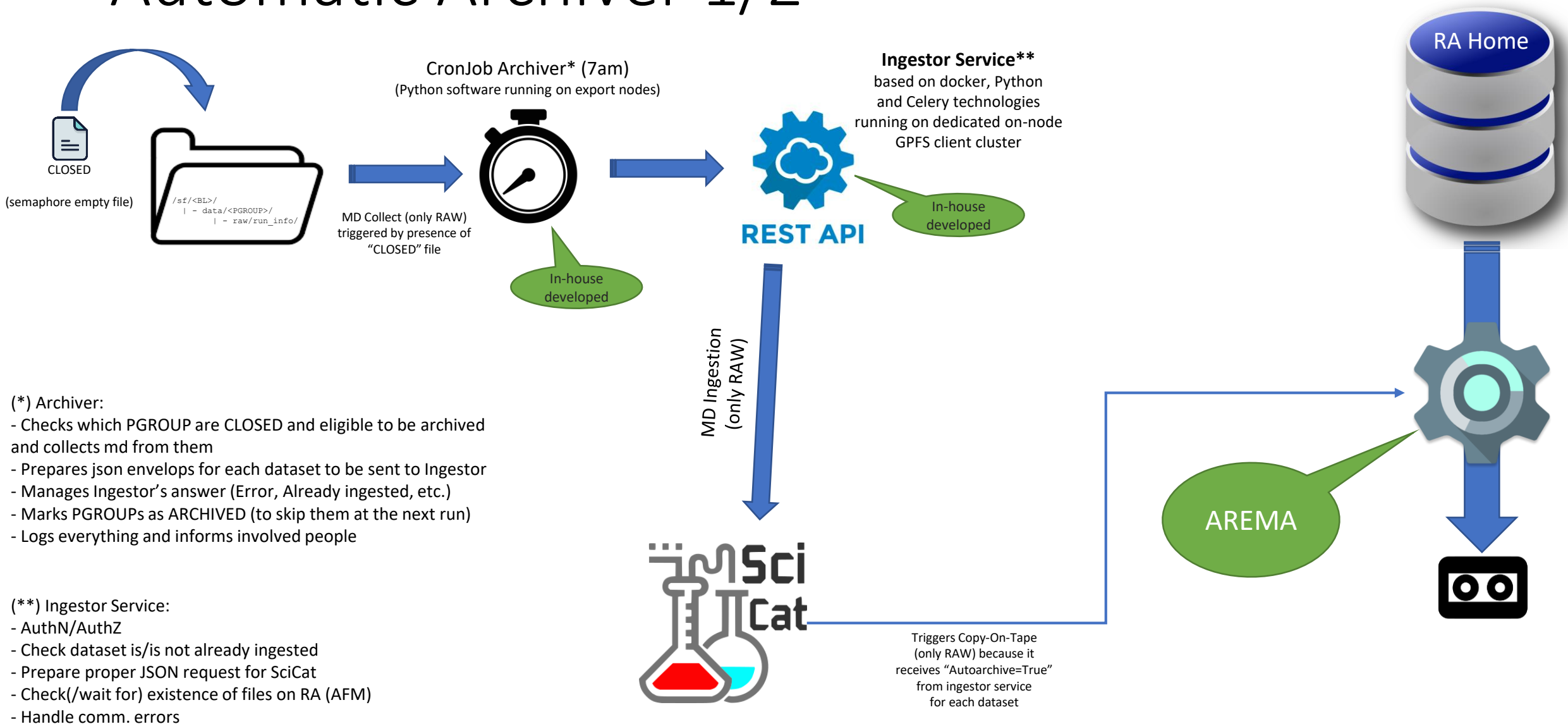resulting in lower write performance seen by the BLs

CSCS(Lugano)
Tapes

Link Physics Experiment to IT

- "Detector Switches"
  - Every exp. hutch has the detector sending data to DAQ through a dedicated and isolated 100GbE switch procured, installed and managed by us
- We have to deal with the interaction with SwissFEL's **electrical engineering** team and the **network** team to organise the optical cabling between exp. Hutches and the server rooms and the connection of the switches to the management network to manage them remotely:
  - In fact hutches are not supposed to be visited freely after "production" phase has started

Data Management

- User's data (RAW and RES) are quickly synchronized to RA (4-4.5 GB/s through optical fibers that cross the Aare river)
- RA quickly becomes the only storage of data
  - AFM "evicts" blocks (only, not metadata) from the small online storage as soon as certain thresholds are reached, in order to keep some free cache space to accommodate new data
  - AFM can fetch back (from RA) evicted blocks if users want to analyze old data using online resources (un-usual; but possible if RA is really overloaded)
  - This means that ARCHIVING ON TAPE must be done very early on, since RA blocks are often the only available instance of data
- Therefore, we have developed an **automatic archiving** system that can solve problems related to our absence due to vacation, illness, slowness, human error…

Automatic Archiver 1/2

CLOSED

(semaphore empty file)

```
/sf/<BL>/
    | - data/<PGROUP>/
            | - raw/run_info/
```

MD Collect (only RAW) triggered by presence of "CLOSED" file

CronJob Archiver* (7am)
(Python software running on export nodes)

In-house developed

Ingestor Service**
based on docker, Python and Celery technologies running on dedicated on-node GPFS client cluster

REST API

In-house developed

RA Home

MD Ingestion (only RAW)

Sci Cat

AREMA

Triggers Copy-On-Tape (only RAW) because it receives "Autoarchive=True" from ingestor service for each dataset

(*) Archiver:
- Checks which PGROUP are CLOSED and eligible to be archived and collects md from them
- Prepares json envelops for each dataset to be sent to Ingestor
- Manages Ingestor's answer (Error, Already ingested, etc.)
- Marks PGROUPs as ARCHIVED (to skip them at the next run)
- Logs everything and informs involved people

(**) Ingestor Service:
- AuthN/AuthZ
- Check dataset is/is not already ingested
- Prepare proper JSON request for SciCat
- Check(/wait for) existence of files on RA (AFM)
- Handle comm. errors

Automatic Archiver 2/2

- Auto-Archived **7.6 PiB** since March/April 2022
  - Aramis: 6.2 PiB
  - Athos: 1.4 PiB

- PI receives email about fail/success

- Admins also receive email…

- Per-datasets ingestion (to comply with SciCat data structure)
  - Only RAW datasets are ingested (atm, agreed with Users)
  - For every datasets, SciCat's REST API is called with "Autoarchive=True"

- **Fault-tolerant**: semaphore file "ARCHIVED" is created in PGROUP folder only if all ingestions are successful for that PGROUP
  - *Otherwise, the next morning the archiver will take care of the same PGROUP (skipping the datasets already ingested).*

- Tape-Archive is a black-box (AREMA software)
  - *Tape-Archive team showed to promptly take care of any error related with tape-ing, they usually fix themselves and just inform us. **Very reliable service from them !***

- **Users look "happy"** 👍

Automatic Clean up

- Agreed with BL that it must be implemented
- Software written and more or less stable (not much tested yet)
- Agreed with BL that they can delay the deletion of archived stuff
  - *because users could still need them*
- For the moment, as the operation is really delicate, we prefer to have a quite a long transition period
  - we delete only manually
  - we use the automatic tool to get the list of datasets eligible for deletion (it check that ingestion && ONTAPE status are both True)
  - We send this list to BL
  - They tell us to selectively delete them

Monitoring & Alarming

- The IT infrastructure is monitored in all aspects that are of concern (or that may pose a serious problem and block data acquisition).
- Technology used:
    - *iCinga*
    - *InfluxDB (time series database)*
    - *Grafana (full in-house made dashboards)*
    - *In-house made scripts (bash/python)*
- Alarms are triggered by
    - *iCinga (email)*
    - *Grafana (chat messages with "warning" flag: **Slack connector**)*
- Users (are supposed to) take care of quota monitoring:
    - we provided them simple Grafana dashboards
    - they are supposed to warn admins in case they need (temporary) quota increase
        - See Leo's presentation: we will develop a self-service system for quota and resource allocation
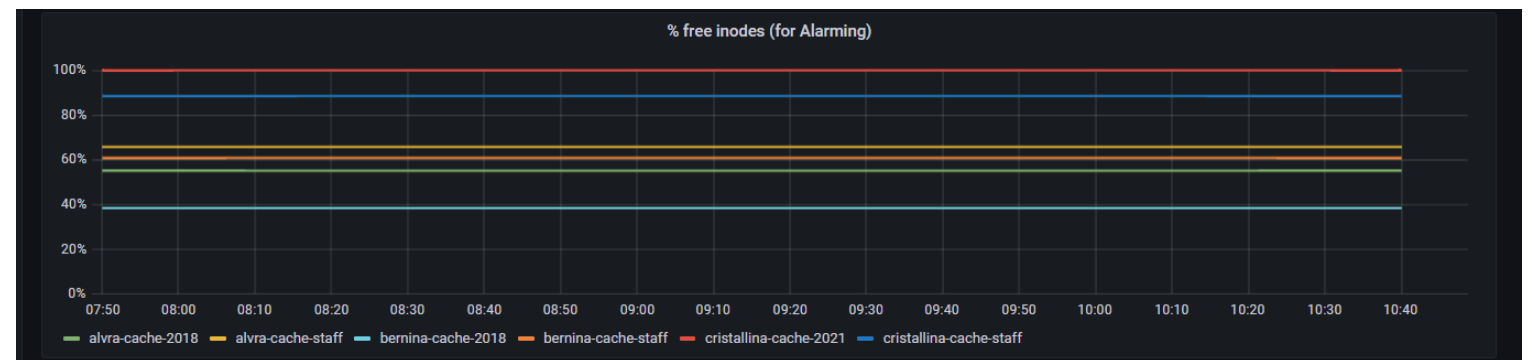
Grafana examples
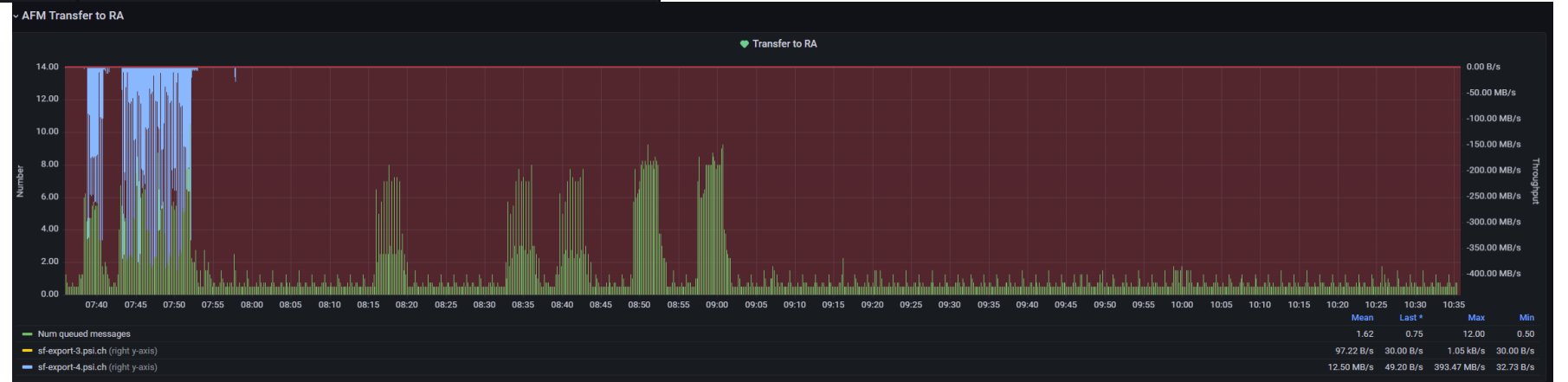
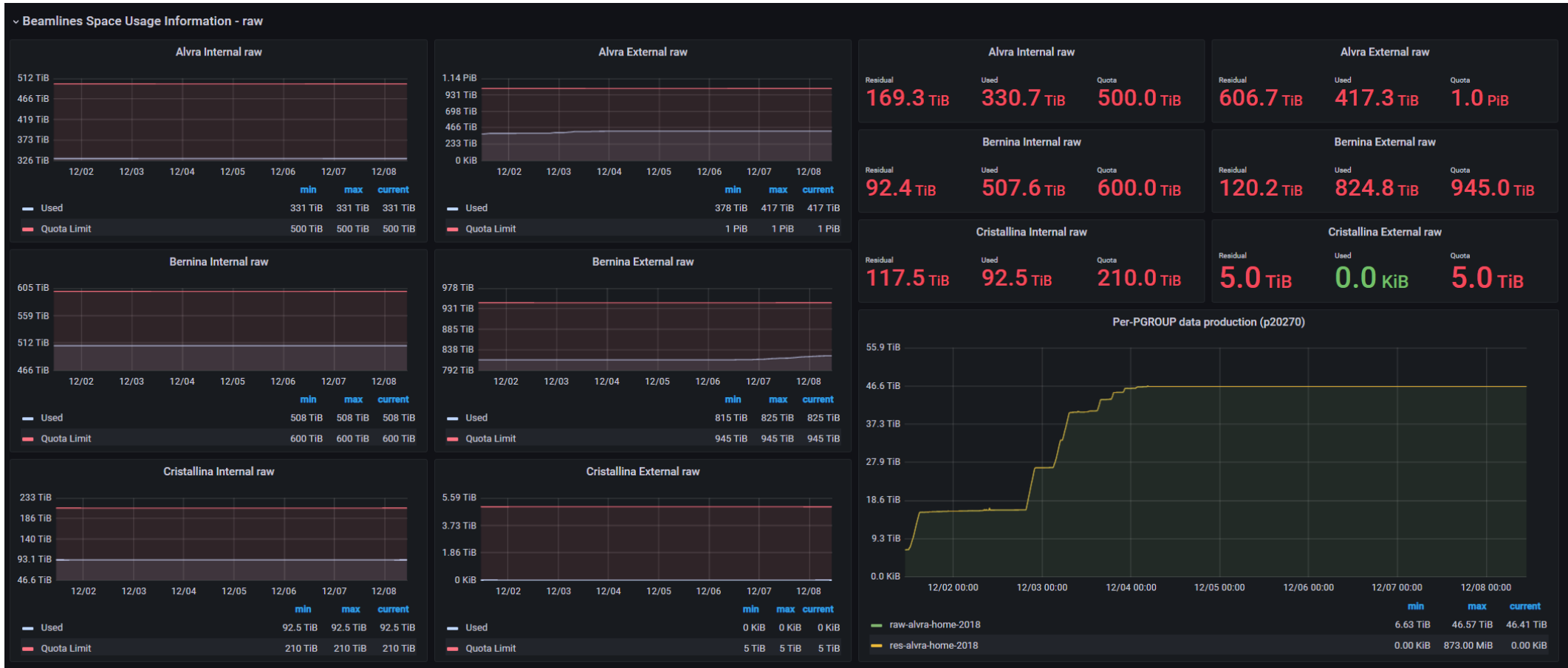# Grafana examples (alarms)

Storage 'ambient' temperature

INODES

Grafana examples (dedicated to end users)

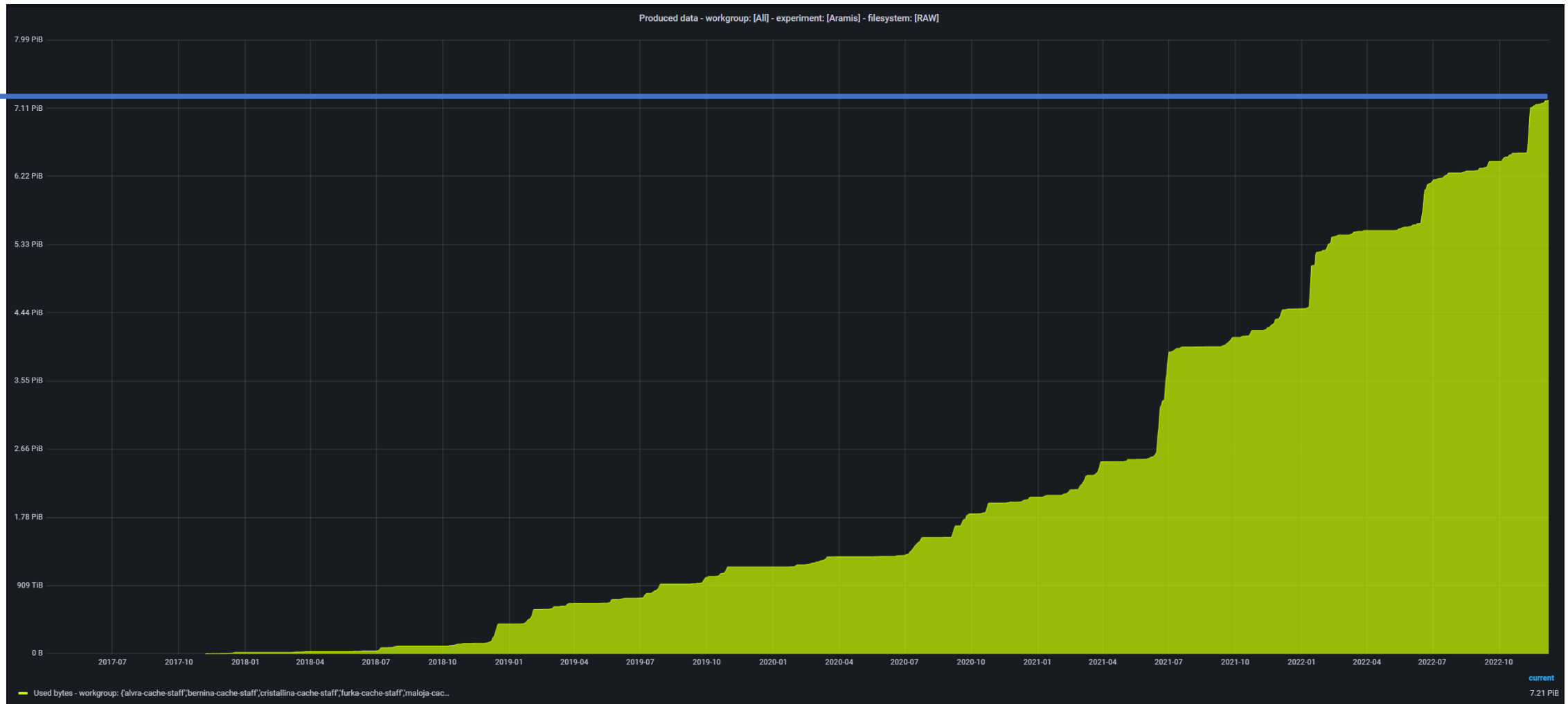# We also take an eye on Power – immagine più grande

Can select rack

Can select IT room

Overall Data Production (Aramis) – mostra fondo scale

7.21 PiB

2023 Challenge 1/3

- Aramis Storage replacement (2.2 PiB online cache)
- Accommodate much more GB/s than now
  - 20GB/s with large block files (i.e. streaming of large files)
  - smaller blocks or md operations have a lot of I/O penalty
- Explore hybrid architecture (SSD + spindle) to give a super-fast frontal cache (30-40 GB/s) and a regular longer term cache storage based on spindle
- Already collected BL's requirements; they need:
  - Best case: 40 or 25 GB/s (depending on the beamline)
  - Worst case: 45 or 50 GB/s (depending on the beamline)

2023 Challenge 2/3

- There is still room to improve the Grafana dashboard to relieve the users from focusing/knowing some technical IT details (work is in progress).

- Optimization of online computational resources usage

- Automatic clean up (better 'fine-tuning' to combine the need to free space and the users' needs)

2023 Challenge 3/3

- Better workflow for hardware management
  - Automatic database insertion of newly procured HW (done, but still some work in progress)
  - Automatic check of warranty coverage period (still some work in progress as the Vendors keep improving/changing WEB interfaces)
  - RH8 and very recent HPC hardware is facing us to new challenges (still work to do in our puppet system