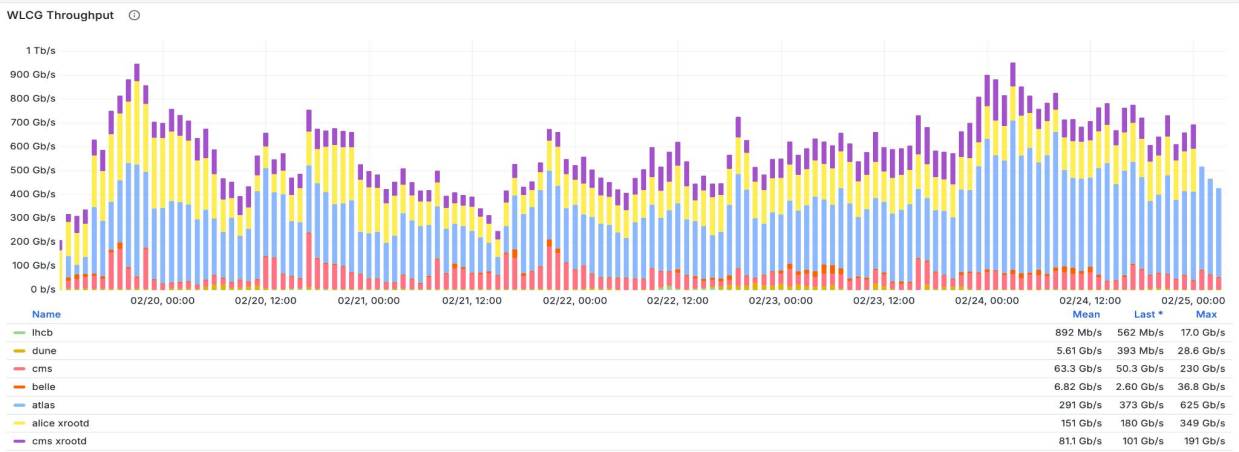
Towards WLCG-wide XRootD monitoring: deployment plans and the road to DC27

September 2026 XRootD and FTS workshop

Borja Garrido Bear (CERN), Panos Paparrigopoulos (CERN),
Karanjot Singh (CERN)

Why this matters

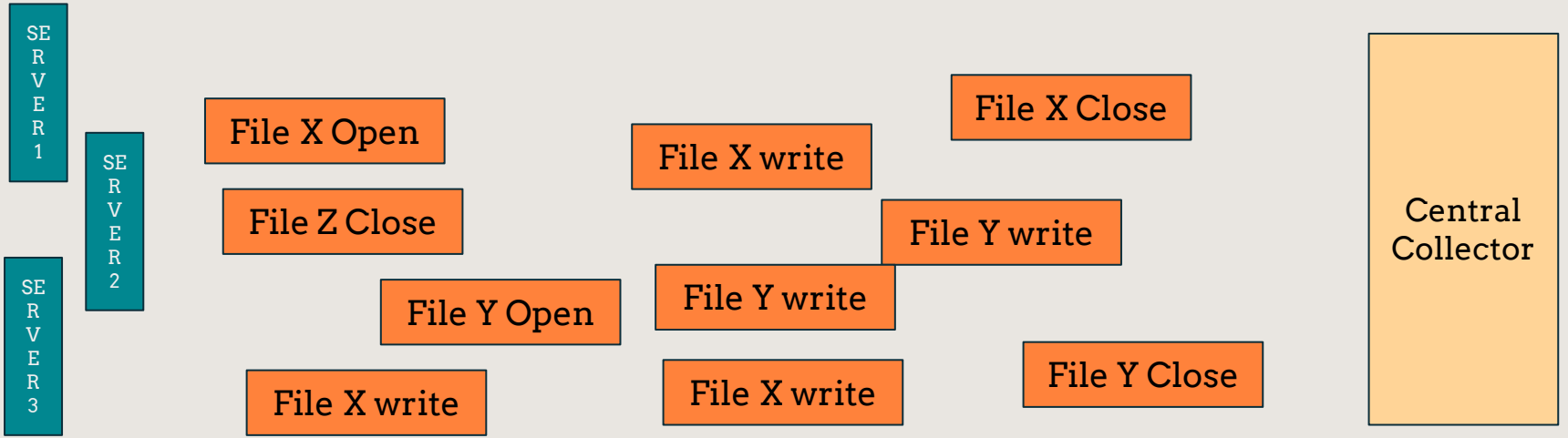
- Transfer monitoring is essential for WLCG Operations like capacity planning and network usage validation
- Two main players
  - FTS: Traffic is well-monitored and trusted
  - XRootD:
    - Represents a large and growing fraction of WLCG data transfers
    - WLCG currently does not monitor XRootD traffic properly!



Data Challenge 2024 Monitoring: Plot showing partial XRootD monitoring during Data Challenges 2024

Why is XRootD monitoring hard?

- XRootD is complex by nature: federated design, heterogeneous clients, wide range of usage patterns
- In XRootD native Monitoring, data comes fragmented into UDP packets:
  - May be lost in transit and generate multiple events per file transfer!
- Another completely different flow exists for sites that run dCache with XRootD



Reminders

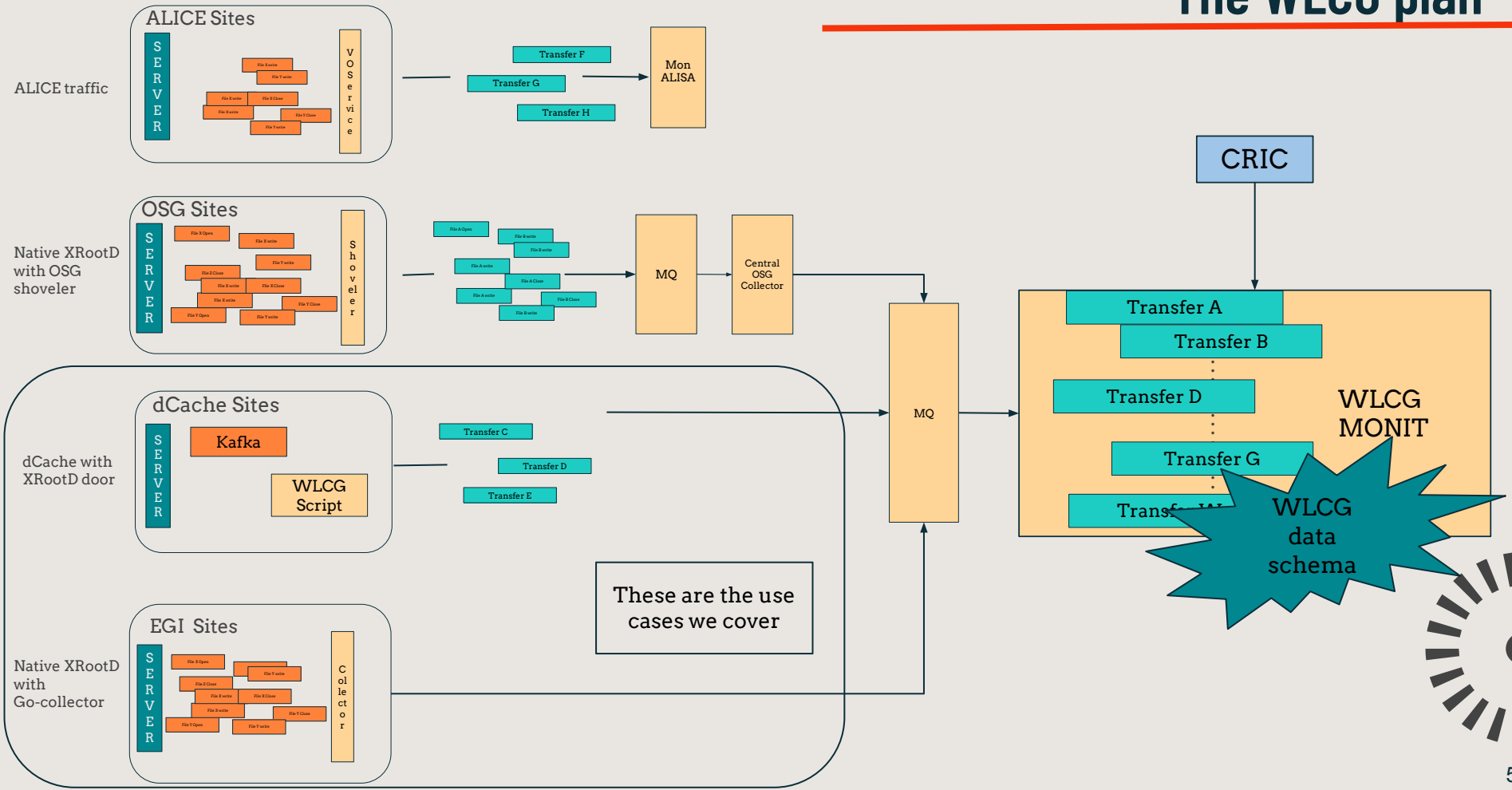
- **Shoveler**: A component that sits on a site, receives UDP packets and forwards them through some reliable protocol to minimise packet loss.

- **Collector**: A component that receives packets (either from UDP, locally, or from HTTP remotely) and contains all the necessary logic to stitch them together, correlate them and produce a final record, which contains all the relevant information.

- **MonALISA**: The ALICE monitoring system which for years now implements an ALICE-specific collector, parsing UDP packets at each site, making this the only reliable source to date for XRootD traffic monitoring.

- **WLCG goal/plan**: Provide global throughput monitoring for XRootD transfers
  - The WLCG scale does not allow for a model of central collectors.
  - **One collector at each site** is the preferred deployment model
  - To be enabled at several major sites ahead of the **2027 WLCG Data Challenge**

The WLCG plan



Collector Development

- The only VO-agnostic collector when we started was the "**OSG collector**", which was recently ported to the Go language, making it more scalable but also rendering it unvalidated.
  - A **native XRootD collector**, would be the technically prefered solution, provided it comes with **long-term support** strategy for it.
  - Our **DC27 goal** means that unfortunately there is no time to validate any new collectors

- A dedicated effort (1.5 FTE) at CERN went into **improving, validating and deploying** the OSG collector.

- 5 main areas of improvement:
  - Observability
  - Configurability
  - Record enrichment
  - Topology resolution
  - Bug fixes

ALL improvements are contributed upstream to the OSG repo!

Configurability + Record enrichment

See more details in the next talk!

- Most WLCG related **configurations** were hardcoded. We have implemented a range of possible configurations to accommodate the differences in the various site deployments.

- Some crucial info that was missing has been added:
  - **Activity** type (production, analysis, etc.) is being properly resolved
  - **Authentication method** (it's still all certificates but we are ready to record token-based traffic)
  - Identification of **LAN traffic** (both WNs traffic,and XRootD internal traffic)

- **Topology resolution** was lacking completely (we only get IPs)
  - We improved the collector to resolve source and destination based on **CRIC information**

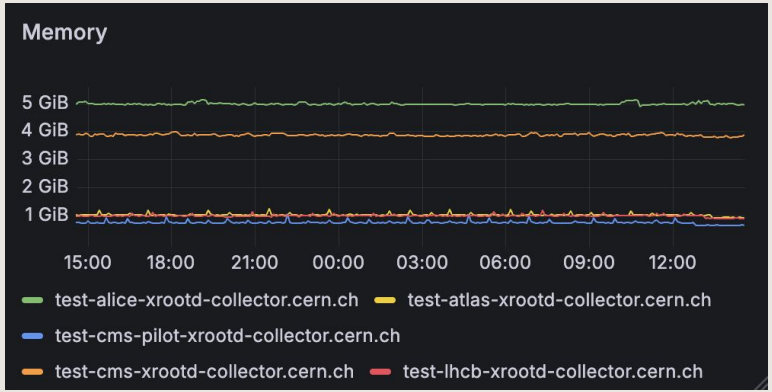
- **IP Version** is there but does not always match the actual IPs

Observability

See more details in the next talk!

- We added detailed metrics on the collector's own behaviour (packets, parsing, site resolution, dropped records)

ALICE collector metrics over 15 hours



LHC VOs Collectors Memory Usage

Validation

See more details in the next talk!

- The only verified source that we have is ALICE.
  - Initially we had huge discrepancies -> not good!

1. **Which stream is being parsed**
   The EOS nodes emit both t-stream and f-stream, a combination that is rarely enabled, and the collector sent both to the f-stream parser. Trace entries became file-close records with absurd byte counts, and sometimes with plausible ones, which was much harder to spot. The two are now dispatched separately.

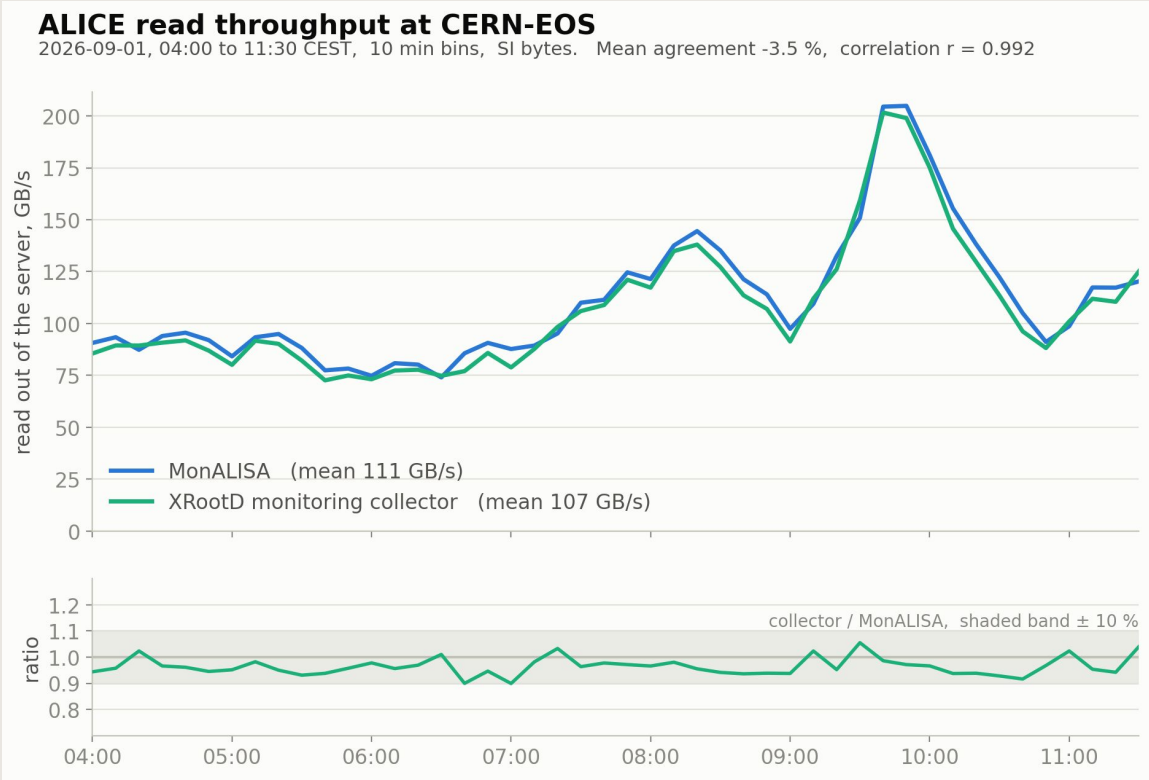
2. **Which bytes count as a read**
   Total read is ordinary plus vector reads. ALICE clients do most of their I/O through the ROOT TTreeCache, which issues vector reads, so counting ordinary reads alone undercounts the volume by roughly half.

3. **Which records the reference keeps**
   We assumed MonALISA excluded traffic internal to a site. Reproducing that made agreement worse. Its source code shows the selection is on the account, not the endpoints: system accounts, root and daemon replication are dropped, and worker-node traffic stays in.

Validation

Once both sides count the same thing, **they agree**

- **3.5% mean difference** in recorded throughput
- ALICE is using the outdated t-stream and makes a few different decisions on how to resolve internal traffic.
- Some packet loss or time shifting based on the bins we select may contribute marginally to this 3.5%
- 3.5% discrepancy is good enough for us **to accept the collector as validated**.



Current status

4

collectors in production at CERN,
one per LHC experiment

98 M

enriched records published per day

3.5 %

Mean difference with the ALICE reference

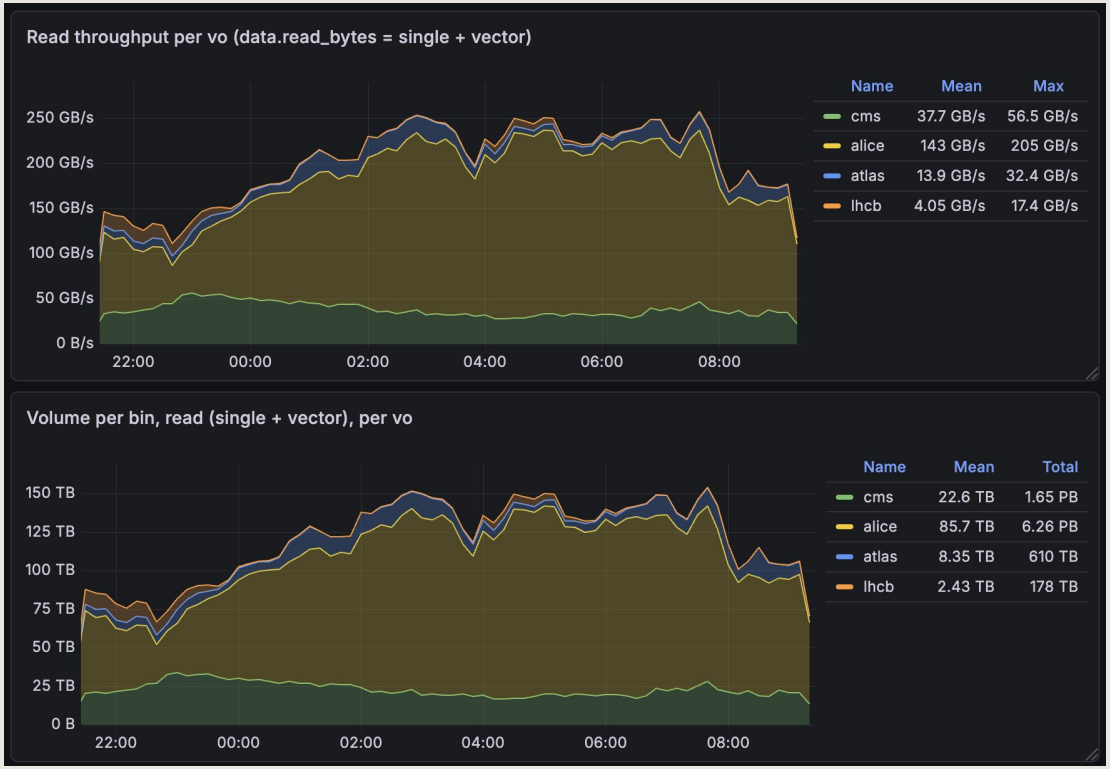
- The validated collectors run for **all 4 experiments at CERN**
  - Thanks to Guilherme for setting all the flows and helping us debug!

- We decided on one collector per VO instance to be able to split the load and also properly monitor each one
  - Different **deployment models** are also supported

- We can finally **reliably monitor XRootD traffic @ CERN**!

Some plots

Total recorded CERN XRootD traffic over 12 hours - **excluding internal server traffic**



Deployment plans

- All changes have been **committed upstream**
  - Once merged we will produce a **golden release**
- Working on site-facing **documentation** and the creation of RPMs and Docker images
- We want to have this **deployed at the first few sites** by the end of the month
  - This step will definitely produce more work :)
  - RAL is first (Thanks Jyothish and Katy!). Any other volunteers?

- All the infrastructure to push, store and display the data is provided by the **CERN-IT MONIT** team.
- We will also be implementing **central alerts**
  - At least for this first deployment phase
- Deployment campaign to follow: production XRootD monitoring at several major sites in time for the **2027 WLCG Data Challenge**.

Monitoring of dCache traffic over XRootD protocol

Deployment plan (dCache sites)

- Monitoring of **XRootD over dCache** traffic does not come out of the box
- Sites must enable the **Kafka** bookkeeping module coming in the dCache package
- **dCache v11.2** will bring a new way of advertising these records, directly sending them out, without the need of deploying Kafka at a site
- **FNAL** and **PIC** have already enabled the Kafka based flow

- We are **working with DESY** to have an improved version of the collector for v11.2
  - This will also include file-names, which are currently missing.
    - This is not strictly needed for WLCG, but after a CMS request we are working to make our data useful for **data-popularity studies**
  - A deployment campaign will follow to enable this at all sites that move to v11.2

Open Questions and XRootD

wishlist

Missing items and open questions

- We have to manually resolve **topology information** using the best approximation we can get
  - Most of the traffic is centrally orchestrated, can we have the XRootD Clients advertising things like the site they run and propagate this to a collector via some stream?

- **IP Version** is there but does not always match the actual IPs
  - We see traffic with an IPV4 side marked as IPv6 and vice-versa

- Flagging **internal traffic** right now requires mapping of specific accounts and protocols
  - Getting this directly from XRootD would be great

- There is no complete **error reporting**
  - HTTP errors have been implemented, but they lack granularity
  - xroot errors are not even reported yet (are they?)

Conclusions

- We put a lot of work into **improving** and validating the **OSG collector**
- The collector is now matching the **WLCG needs** and can be considered **validated**!
- A **golden release** is being prepared, and the first site deployments start in the next few weeks
- **dCache**: sites on current versions can already publish records via Kafka. The dCache team is developing an improved flow for v11.2, with a deployment campaign to follow
- There is **room for improvements**, so we will keep bugging the XRootD Team :)
- Huge **thanks** to the XRootD and dCache dev teams!

Thank you