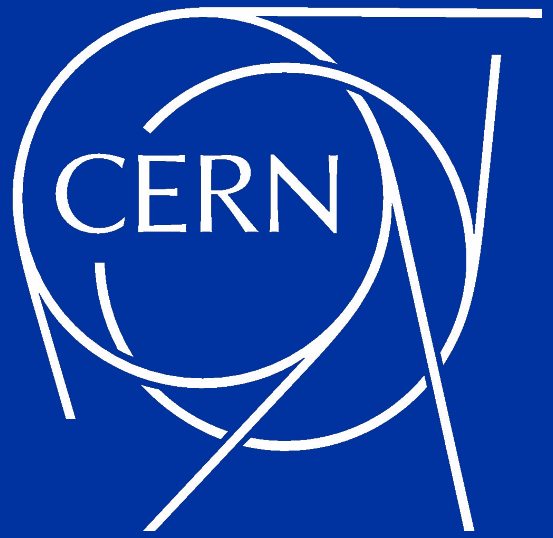

CERN

Applied Multi-Disciplinary AI on High-Performance Computing

a.k.a. Optimizing Storage and Data Access for ML Projects

Radomir Michal Wasowski
PRINCIPAL INVESTIGATOR
CERN IT, Storage

Openlab Technical Workshop
4 March 2025, CERN

Disclaimer

- **This project started ~4 weeks ago**
- **No concrete results to show today**
  - …but we will have many for Openlab Technical Workshop 2026!

- **What follows is our short-term project plan in the context of Storage Optimizations for AI/ML workloads**

- **We are eager to hear about your AI/ML use cases and workloads**
  - Storage requirements
  - Access patterns to data stores
  - Bottlenecks encountered in your setups

The Project

Applied Multi-Disciplinary AI on High-Performance Computing

- Phase I: Machine-Learned Particle Flow Reconstruction
- Phase II: Storage and Data Access for ML Projects ← **Our focus**
  - Nature of storage access by ML workloads
  - Optimizing bottlenecks
  - Focused on Ceph

SIMONS FOUNDATION

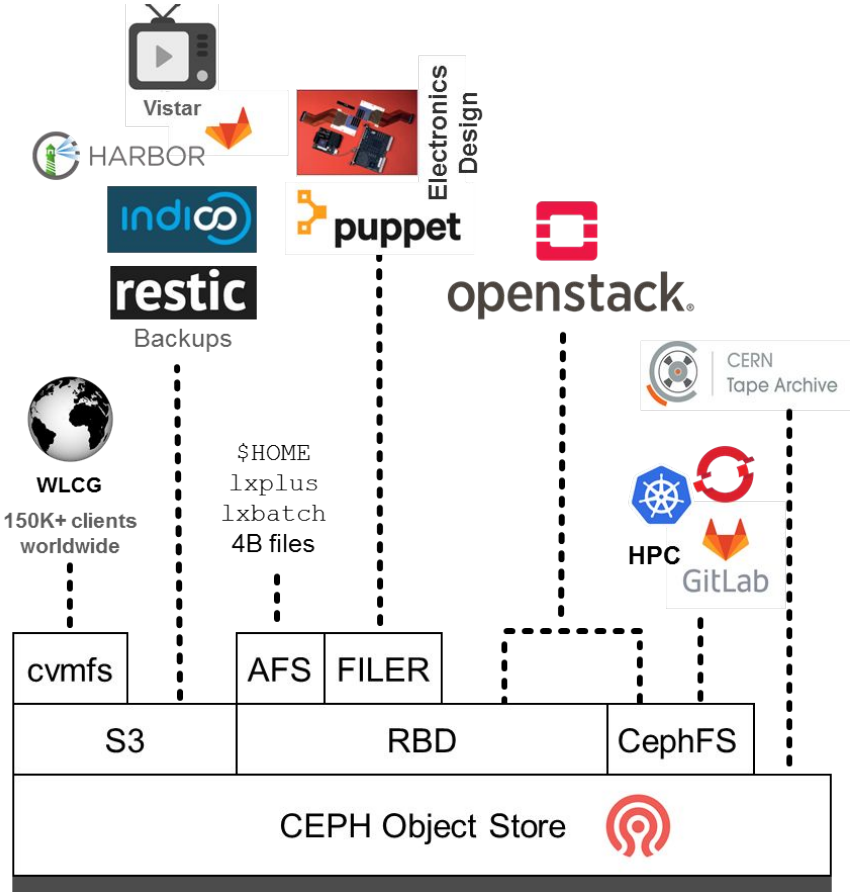
Ceph at CERN

Open source software-defined storage

- Proof-of-concept in 2013, and growing ever since
- Now, 24 production clusters spanning 2 data centres

- Backs the entirety of CERN's OpenStack cloud
- Virtualizes other storages – e.g., AFS, NFS
- Supports HEP analysis and Accelerator complex

Service		Capacity
RBD (OpenStack Cinder/Glance, `krbd`)	*HDD, Flash || Replica 3, EC 4+2*	34.9 PB
RGW: S3, Swift (Cloud Native Applications, Backups)	*HDD EC 4+2*	41.2 PB
CephFS (OpenStack Manila, K8s/okd PVs, HPC)	*HDD, Flash || Stretch (Replica 2+2)*	24.5 PB
RADOS Objects for CERN Tape Archive (CTA)	*Tape DB and Disk Buffer*	235 TB

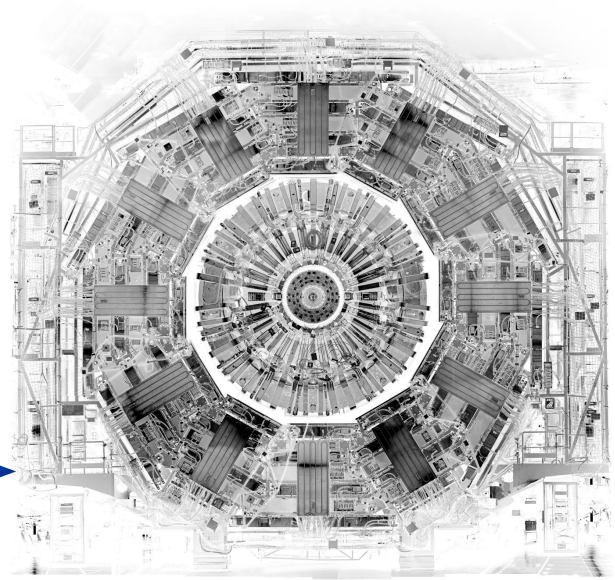


Ceph: Supporting High Energy Physics

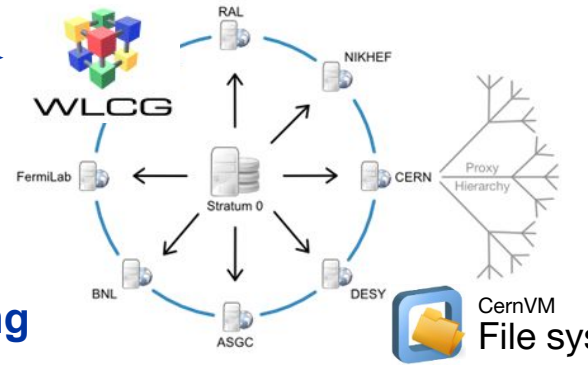
CERN Tape Archive

RAW Data to Tape

Rados Object Store



Events Reconstruction & Analysis

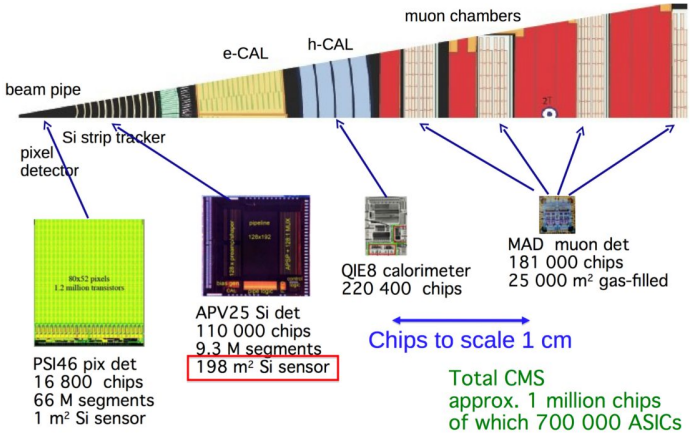


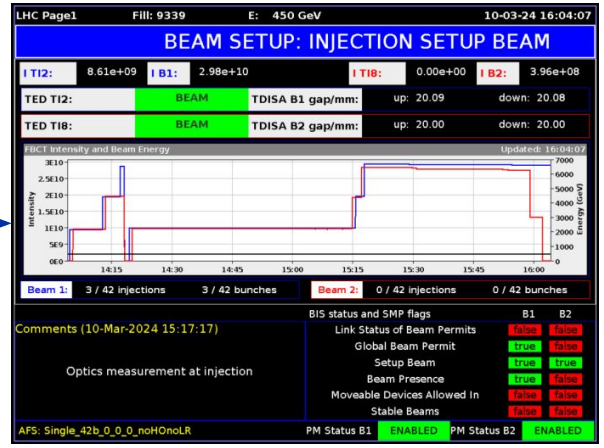
CernVM File system

Ceph RadosGW + Global HTTP CDN

Volunteer Computing
CephFS

ASIC Design
HPC Clusters + CephFS





Accelerator Monitoring
Ceph RadosGW

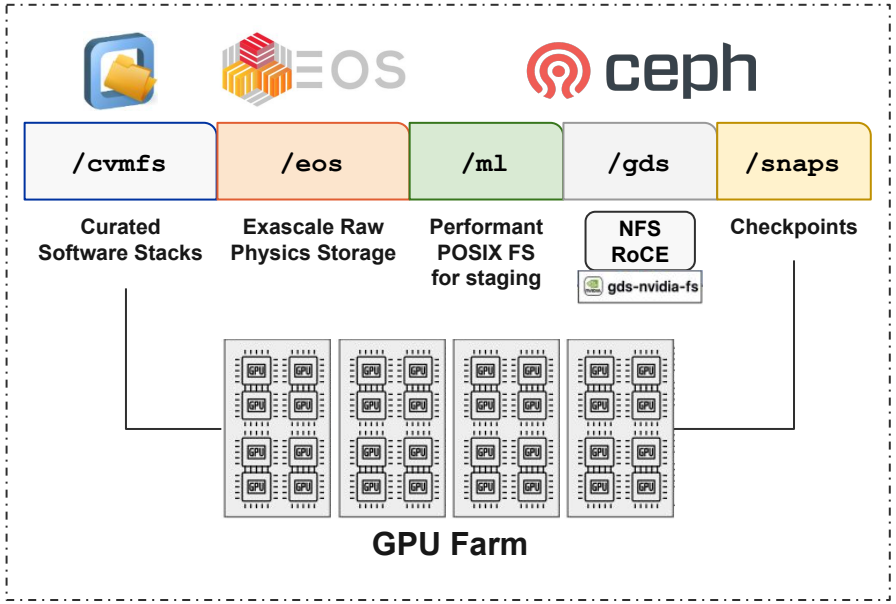
Relevance of Ceph for AI/ML workflows

Ceph is 3 storage types in 1 platform

- Virtualized blocks via kernel-RBD or librdb/QEMU
- Object storage via S3 ⇒ Widely-adopted API
- Files & Directories ⇒ Filesystem-level access

…potentially hosting

- Massive data stores ⇒ Training || Validation Datasets
- Shared scratch spaces ⇒ Preprocessed Data || Checkpoints
- High-performance storage ⇒ Massively Distributed Training



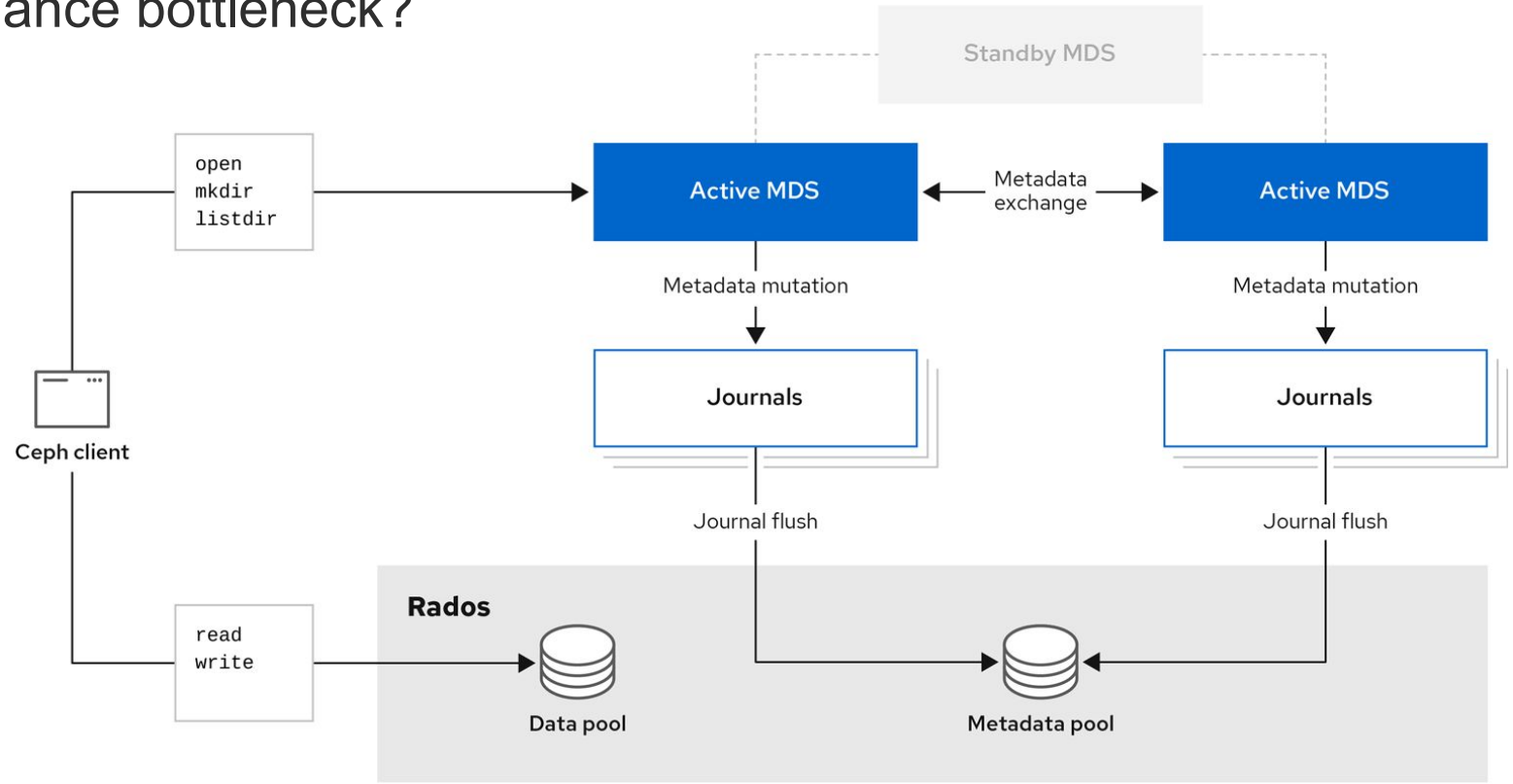
Hypothetical data stores for AI/ML facility

Challenges ahead

- Would performance be satisfactory out of the box?
- Can it be optimized for AI/ML storage access?

Interacting with CephFS

- First performance bottleneck?



https://docs.redhat.com/en/documentation/red_hat_ceph_storage/7/html/file_system_guide/introduction-to-the-ceph-file-system#ceph-file-system-components_fs

The Problem Areas

1. Metadata Scaling

1. CephFS Observability

1. AI/ML Workloads Characterization

The Problem Areas

1. Metadata Scaling
   - How to scale metadata operations effectively?
     - Use multiple-active MDS daemons?
     - How to balance load across daemons?

1. CephFS Observability

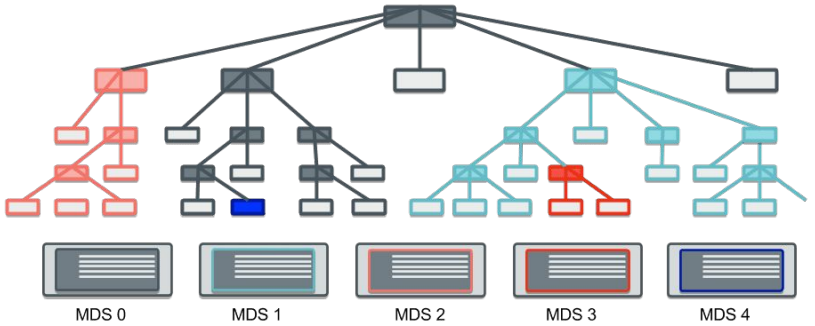
1. AI/ML Workloads Characterization

Metadata Scaling

- CephFS allows for multiple MDS daemons
  - Increase metadata throughput with highly-parallel access
  - More on I/O patterns later…

- With a multi-MDS approach,
  how to evenly spread metadata operations?
  - Several strategies available:
    i. Manual static pinning
    ii. Dynamic balancer
    iii. Ephemeral pinning

  - Which would be the best fit for AI/ML workloads?
  - What is the overhead caused by balancing?



The Problem Areas

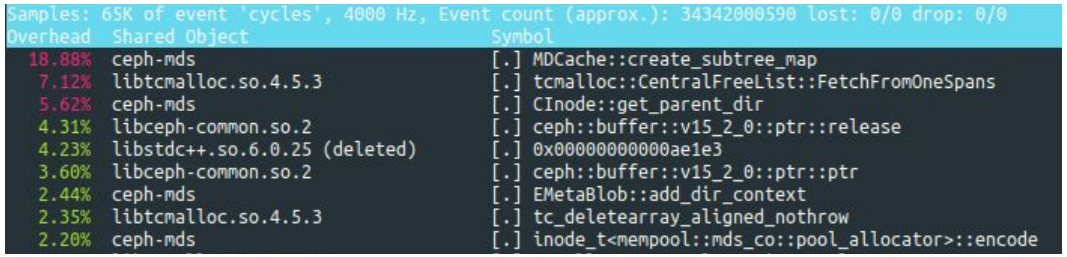
1. Metadata Scaling

1. CephFS Observability
   - Are performance counters exposed by MDS and Clients sufficient?
   - Can one accurately track performance-relevant information?
     - Client ⇐⇒ FS interactions
     - Time spent to service one request by MDS
     - Effect of concurrency by parallel access from multiple clients

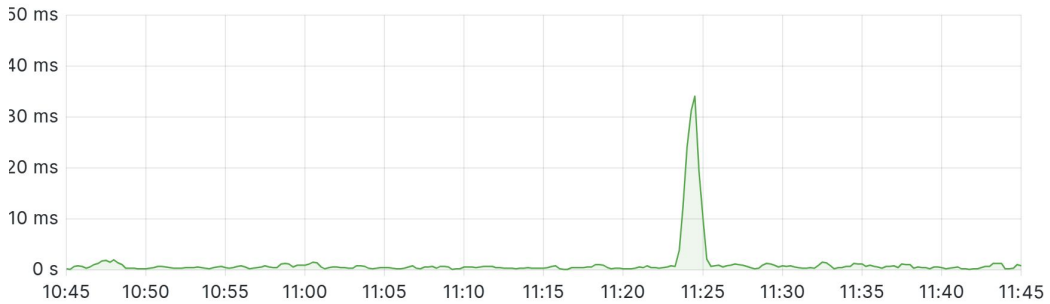
1. AI/ML Workloads Characterization

Observability: What to look into

1. File Operation ⇐⇒ MDS Events
   - MDS Performance Counters, `CephFS Top`
   - CephFS Client Metrics (MDS scraping, kernel dynamic debug)
2. OS Level Tracking
   - `inotify`, `strace`, `sgi_fam`
3. Wall-clock profiling
   - `perf`, `strace`, `libunwind`
4. And more (not always available)
   - Darshan
   - FSEvent, `kqueue`, `kevent`



`perf-trace` of MDS process



Spike in MDS `create_latency` not clearly correlating with other perf counters

The Problem Areas

1. Metadata Scaling

1. CephFS Observability

1. AI/ML Workloads Characterization
   - How to capture characteristics of I/O patterns during ML?
   - What strategies are employed in existing literature?
     - Are they representative of ML workloads @ CERN?
     - Can existing benchmarking tools be leveraged?

ML I/O Patterns

- Patterns depend on many factors:
  - Format of Inputs
  - Data Preparation Pipeline
  - ML Algorithmic Details
  - **Storage Schema & Backing Store**
- Randomness vs. Noise
  - Caching
  - Resource-Sharing

- Requires significant GPU resources
- Replicability is poor
  - Scheduling on shared GPUs
  - Non-deterministic
- Scalability is poor
  - Need many trials
  - Need many experiments
- **Minimize downsides:**
  Synthetic Data

Workload Producers

1. Ceph Cluster Benchmarking
   - `rados bench, fio`
2. Filesystem Benchmarking
   - `fio, mdtest, io500`
   - `iozone, filebench`
3. AI/ML specific?
   - DLIO Benchmark
   - Nvidia MLPerf

- End-to-end performance evaluators (e.g. Nvidia MLPerf) suffer “Real Data” problems
  - Unless recorded logs can be used to “replay” I/O requests realistically
  - What other opportunities are there to create synthetic I/O request data - at scale?

Experimental Setup

Controlled Variables

- Hardware
  - Storage nodes with NVMe devices
    - AMD EPYC 7402P, 256 GB memory
    - 10x 7.68 TB || 15.36 TB, 1DWPD, PCI v4.0 || v5.0
    - 25 Gbps || 100 Gbps Ethernet
- Ceph Versions
  - Reef (`v18`) or Squid (`v19`)
  - Custom patches and/or backports
- Tuning Parameters
  - `ceph config set…`
- Testing Methodology

Manipulated Variables

- ML Workload Parameters
  - Dataset/Model Sizes
  - ML Task

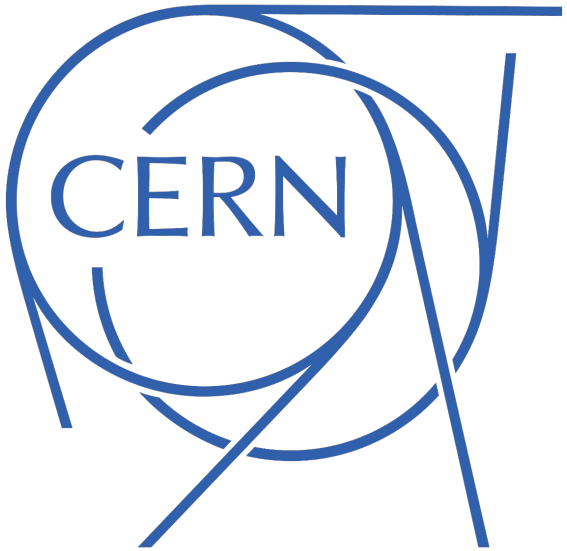
- **A Moving Target**

We’d Love to Hear from You

We are happy to discuss about:

- Your AI/ML use cases and workloads
- Storage requirements
  - Access patterns to data stores
  - Bottlenecks encountered in your setups
- …

THANK YOU!


CERN