EPJ Web of Conferences **337**, 01169 (2025) https://doi.org/10.1051/epjconf/202533701169
CHEP 2024

Reading Tea Leaves - Understanding internal events and addressing performance issues within a CephFS/XRootD Storage Element

Matt Doidge[1,*], *Gerard* Hand[1,**], *Peter* Love[1,***], and *Steven* Simpson[1,****]

[1]Lancaster University, Lancaster, UK

Abstract. Erasure-coded storage systems based on Ceph have become a mainstay within UK Grid sites as a means of providing bulk data storage whilst maintaining a good balance between data safety and space efficiency. These storage systems are complex and self-correcting, but despite access to a myriad of metrics, the inner workings of the storage tend to be opaque to the storage admin. One of the common problems seen within Ceph based systems is *slow ops*—instances of operations that take longer than expected, that are also often blocking in nature, impacting the overall performance and reliability of the system. Identifying the causes of slow ops can help to prevent or reduce the impact of future occurrences, leading to an increase in performance and reliability.
We detail the efforts of the Lancaster Grid Site to understand the causes of and mitigate against these slow ops and other performance bottlenecks within our storage system. We endeavour to bring together a holistic monitoring model, utilising Ceph metrics, detailed XRootD monitoring streams, and client-side logging, in order to understand how data-management events impact the health of the storage.

1 Framing the Issue

The WLCG Grid Site at Lancaster uses XRootD [1] to front a CephFS [2] volume as the site Storage Element. This is a complex system with many moving parts, and problems can cause a cascade of knock-on issues. One family of commonly experienced issues is what are colloquially referred to as "slow ops".

A *slow op* is an operation that has taken longer than a predetermined length of time, and can arise on the various Ceph daemons: OSD, MON, MDS and MGR. We see a predominance of slow OSD ops as the cause of loss or degradation of service at Lancaster.

The details of the Lancaster storage system have been discussed elsewhere [3], but are described in Figure 1. In summary, a Ceph cluster of 32 nodes each containing 24 hard discs provides 11 PB of storage, presented as POSIX-like CephFS through two metadata servers to two groups of clients: Slurm worker nodes (WNs) and XRootD gateways. Gateways have read-write access to (de-)populate the filesystem, and WNs have read-only access for job data stage-in.

*e-mail: m.doidge@lancaster.ac.uk
**e-mail: g.hand@lancaster.ac.uk
***e-mail: p.love@lancaster.ac.uk
****e-mail: s.simpson@lancaster.ac.uk

 https://doi.org/10.1051/epjconf/202533701169

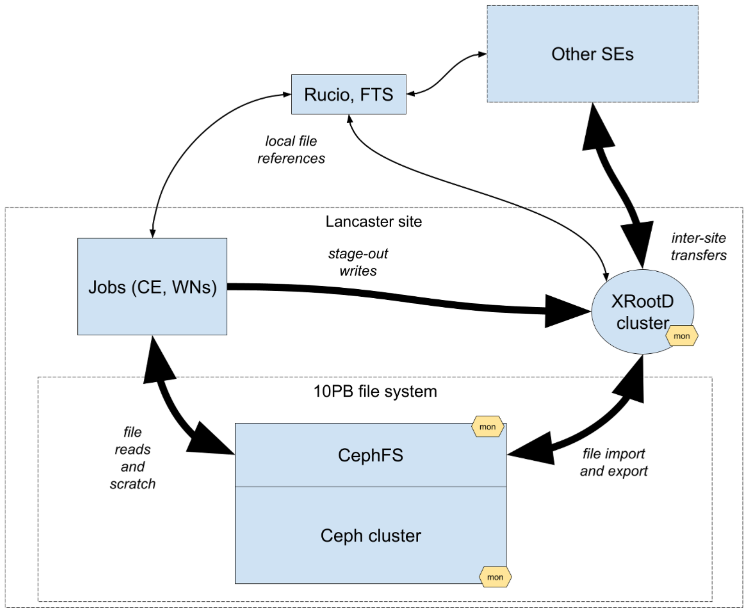

Figure 1. Components and interactions of the Lancaster Site Storage.

This paper recounts our experiences with slow ops, completing this section with summaries of some specific slow-ops incidents, followed by discussions on impact in Section 2, detection and monitoring in Section 3, causes and triggers in Sections 4 and 5, and our attempts at mitigation in Section 6. Section 7 concludes the paper.

1.1 Slow Ops Example Events

Throughout this paper, we refer to data gained during two separate incidents as examples:

Incident 2024-12-20 Three periods of slow ops were observed on and around this day. Figure 9 shows bursts of recovery-operation I/O preceding each event (indicated by the blue annotations). Figure 4 shows disc latency for one specific OSD increasing just before each event. These demonstrate how certain activities are candidates for causes of slow ops, yet these high disc latencies and recovery operations are not consistently present in all instances of slow ops.

Incident 2025-01-15 16:35–00:30 An evening of slow ops disrupted file transfers, as XRootD gateways got into largely unproductive states. Figure 2 shows XRootD gateways experiencing slow CephFS operations, and with CPU spending unusually long times in `iowait`. Figure 3 shows slow OSD ops impacting CephFS operations. These demonstrate how slow ops degrade high-level operations, which often lead to poor user experience.

In our investigations, we observed Ceph process latency building up prior to the incident (Figure 7), a phenomenon almost always present during slow ops, and likely an almost direct expression of them. For potential causes, we saw an initial drop in process latency coinciding with a prior spike of recovery activity (Figure 8), as well as read IOPS building up prior to the incident (Figure 6). Figure 5 shows a surge in ATLAS jobs prior to and during the incident, warranting suspicion that the higher (but not unusual) user load combined with spikes of background Ceph maintenance might be the cause of the slow ops. In other incidents, LSST job surges are often observed, and can have a greater impact despite being much smaller than ATLAS load, due to their types of job accessing many smaller files.

2 Consequences of Slow Ops

As the Lancaster site accesses the Ceph storage exclusively through CephFS, we observe slow ops impacting the function of the two main systems that use CephFS mounts:

- XRootD Gateways: Slow ops in Ceph cause file access delays in the gateway which then impact the XRootD operations, including transfers, stages-in and stages-out, which can eventually time out.
- Compute Node Data Mounts: Each compute node includes at least one CephFS mount, usually read-only, and data stage-in is often realised via cheap symbolic links into the mount. Attempts to read files directly amount to CephFS operations, which are directly impacted by slow Ceph ops, and many consequently time out, leading to job failures. Compute job stages-out are impacted less directly, as they are invocations on the local XRootD gateways.

In either case, Grid activities are negatively impacted, leading to internal alerts, ATLAS HammerCloud test exclusions, and tickets from VOs. Figure 2 demonstrates such impact on an XRootD gateway.

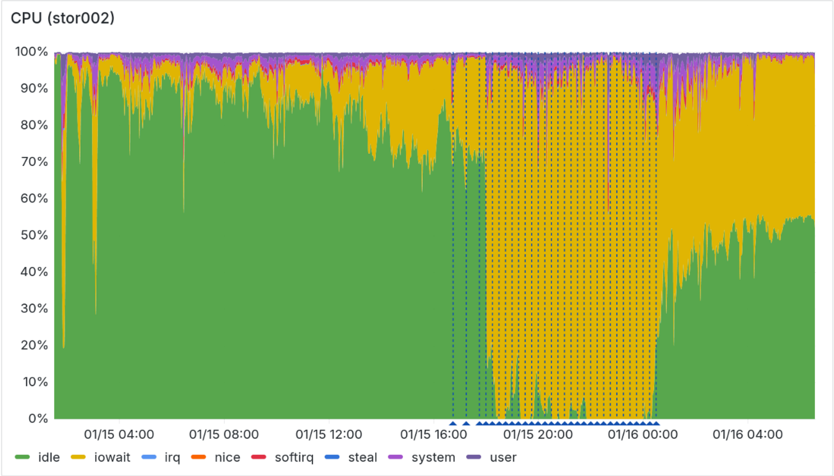

Figure 2. Each XRootD gateway has a CephFS mount, which can become sluggish as slow ops emerge. As XRootD attempts to fulfill requests by accessing the mount, it spends increasing time in `iowait` on local file operations. This manifests on all gateways except the redirector, which performs no bulk I/O itself, only directory operations and `stats`.

Slow OSD ops have a varying degree of impact on the overall system. At the very least, they slow CephFS file I/O operations (as seen in Figure 3), reducing transfer and data access efficiency. At their worst, slow OSD ops can cause a cascade of failures through the component services used to provide CephFS, and such failures can require manual intervention to clear.

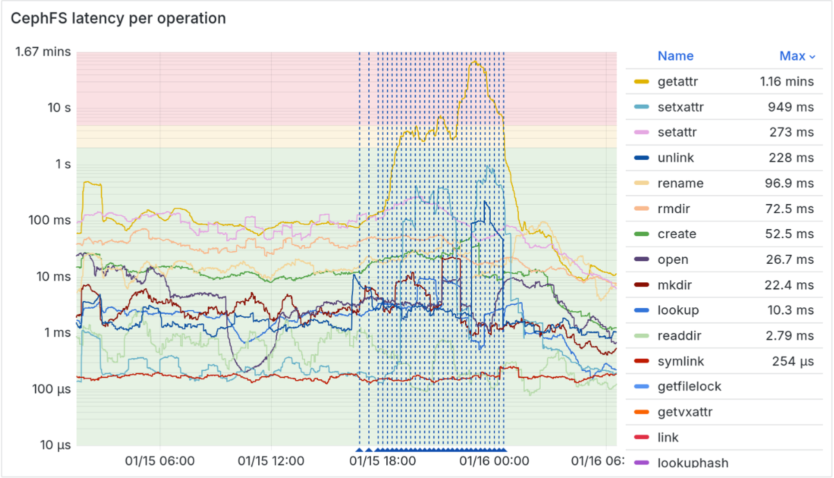

Figure 3. Slow ops on Ceph OSDs lead to slow CephFS operations. `getattr` operations, which constitute 10 % to 20 % of CephFS operations and around 80 % of CephFS load, take up to 1.16 min instead of a more typical 100 ms.

In CephFS, a file's contents and metadata are stored in distinct pools, and access to either can be affected by slow OSD ops. Slow ops in a metadata pool affect access to all files. With prolonged periods of slow OSD ops, we see the MDS being impacted with slow metadata IO and slow MDS ops. If this continues, we then see XRootD hosts failing to release capabilities and an increasing proportion of failed file transfers.

3 Detecting Slow Ops

Slow ops can be detected and monitored in various ways. First, we use metrics produced by Ceph to generate email alerts when slow ops are detected:

- Ceph metrics include `ceph_health_detail{name="SLOW_OPS"}`, giving an overall health warning for a Ceph cluster.
- Ceph metrics also include `ceph_daemon_health_metrics{type="SLOW_OPS"}` which reports specific daemons using the `ceph_daemon` label.
- CephFS may manifest slow filesystem operations, and these can be observed using metrics of the form `ceph_mds_server_req_(.+)_latency_{sum,count}`. These can be due to underlying Ceph issues or to resource problems on the CephFS meta-data servers. The metrics for `getattr` operations are particularly indicative of slowness.

Second, the `ceph status` and `ceph health detail` commands give an overview of the current state of the cluster, showing how many slow ops are currently active and which OSDs are involved:

```
# ceph status
  cluster:
    id:     12345678-1234-5678-90ab-cdef12345678
    health: HEALTH_WARN
            1 MDSs report slow metadata IOs
            1 MDSs report slow requests
            7 slow ops, oldest one blocked for 345 sec, daemons [osd.167,osd.201,osd.637,osd.653] have slow ops.
```

Slow ops can be identified in the cluster log channel by using the command `ceph -W cluster | grep -i slow`, giving output such as (omitting timestamps):

```
osd.455 [WRN] 1 slow requests (by type [ 'started' : 1 ] most affected pool [ 'atlas.ec.data' : 1 ])
mon.cephadmin1 [WRN] Health check failed: 1 slow ops, oldest one blocked for 31 sec,
osd.455 has slow ops (SLOW_OPS)
```

4 Identifying Causes of Slow Ops

Ceph is a complex system, and identifying the exact cause of slow ops can be difficult. Accessing data utilises multiple subsystems, each of which could introduce a delay. For example:

- Network infrastructure — Is there a bottleneck in the networking (overloaded NIC/switch/rack uplink) or a fault in the hardware?
- OSD node:
  - CPU — Is the CPU overloaded?
  - RAM — Is there sufficient RAM or is swap space being used?
  - BlueStore, OSD WAL — Is there a delay accessing the OSD's BlueStore (the Ceph Storage Backend and Database), or the write-ahead log?
  - Storage device — Is there a fault on the device or is the device working at its limits?

At Lancaster, we store data using erasure code 8+3 which means that any object written into a pool is spread across 11 OSDs. So when a slow OSD op occurs (e.g. Figure 4), which of the 11 OSDs involved is causing the problem?

The current operations and previous (historic) operations for an OSD can be shown using the commands `ceph daemon osd.`*`id`*` dump_ops_in_flight` and `ceph daemon osd.`*`id`*` dump_historic_ops`.

An instance of slow OSD ops can involve more than one OSD. Looking at which OSDs have slow ops may give an indication of where a problem lies. Do they have anything in common, such as being on the same host? Do they go through the same switch?

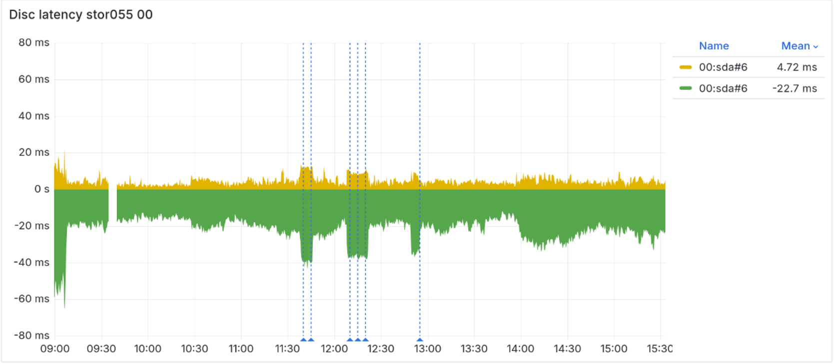

Figure 4. Bursts of disc latency on OSD 6 (slot 00 on `stor055`) appear to precede these slow ops on the same OSD, but are not present in all cases.

5 Triggers of Slow Ops

We have observed events of the following kinds immediately preceding and during slow ops. Within normal operational parameters, no single type of event is guaranteed to result in slow ops.

High storage utilization

We have seen an increase in slow OSD op instances when the cluster utilization reaches approximately 87 %. As the cluster continues to fill, the frequency of slow ops increases.

High CephFS load

Determining the I/O levels that lead to slow OSD operations has proven challenging. At Lancaster, exceeding 15 000 io/s or 12 GB s^{-1} in data transfers frequently results in slow OSD operations. Below these thresholds, predicting slow OSD ops becomes more difficult. High levels of read, write, or delete activities may appear to cause slow OSD ops one moment and then perform normally the next, as those triggered by bursts of user jobs (Figures 5, 6, 7).

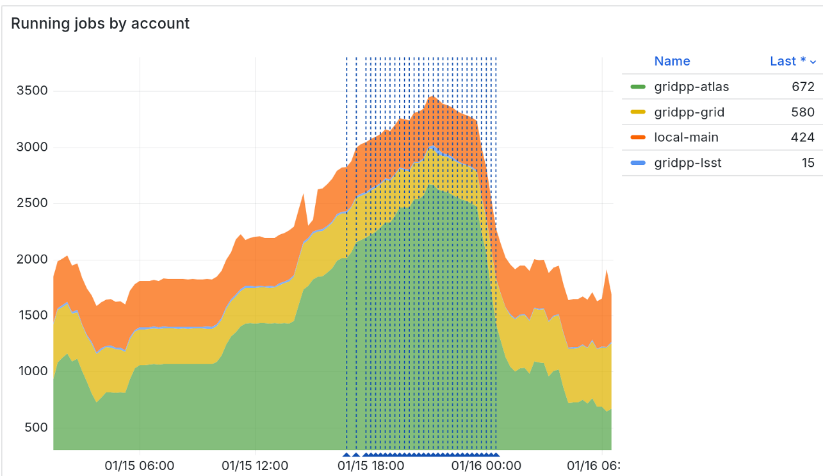

Figure 5. A surge of ATLAS jobs might have created a significant load on the CephFS mount to contribute to slow ops at this time. However, direct access is read-only; writes (stage-outs) must go through the local XRootD gateways, which have read-write mounts.

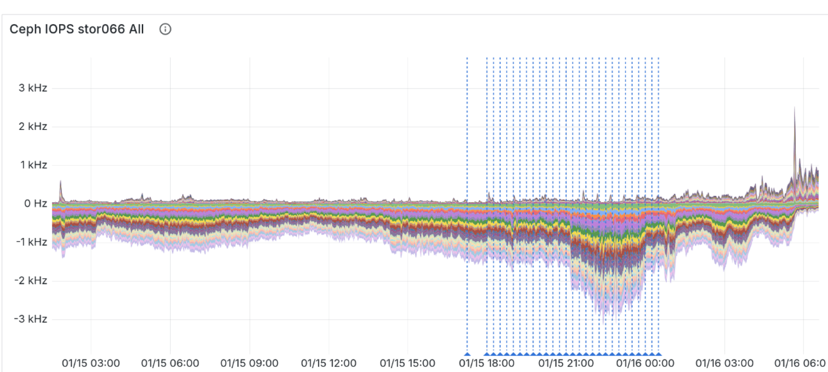

Figure 6. High read IOPS are observed during and building up to the slow-ops event. Interestingly, OSD 276's IOPS is only 15th highest out of 24 on `stor066`. Indeed, it drops to zero briefly near the start of the event. This could be indicative of IOPs starvation.

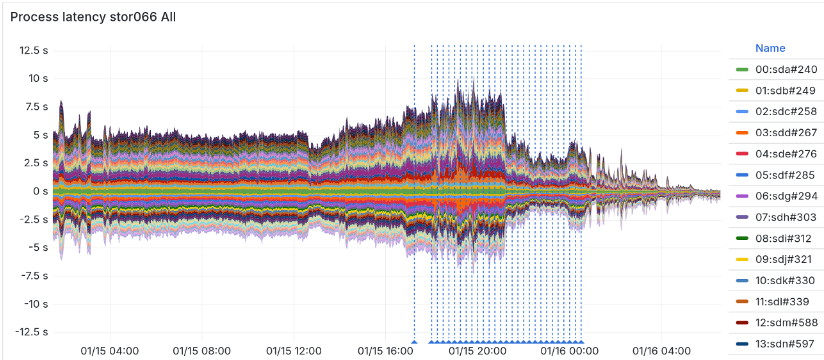

Figure 7. OSD 276 (in drive slot 04) suffers slow ops continuously during this event. Process latency across its node (`stor066`) is observed to increase before slow ops, but as this is also a Ceph metric, it does not directly point to an external influence. (Values in this graph are stacked, so the peak of about 8 s is the aggregate of 24 OSDs' latencies.)

High levels of backfilling

Upgrading our Ceph installation from Pacific to Reef has introduced mClock functionality to schedule and tune tasks such as backfilling and scrubbing according to the system's needs and current load. This proved to be problematic, so as a result, we have set `osd_mclock_override_recovery_settings` to allow manually setting the following configuration options: `osd_max_backfills`, `osd_recovery_max_active`, `osd_recovery_max_active_hdd`, `osd_recovery_max_active_ssd`. Slow ops need monitoring carefully when recovery/backfilling is going on as setting the values too high with high data I/O will cause slow OSD ops (Figure 9). As the I/O rate increases, these recovery/backfilling values need reducing, and in some cases it is necessary to stop backfilling temporarily.

High levels of scrubbing

Both scrubbing and deep scrubbing are also seen to trigger slow ops, as these operations cause additional load on the OSDs.

Figure 8. The drop of process latency (Figure 7) from a steady level followed by the increase at 12:38, which precedes and runs into the slow ops, coincides with a cluster of recovery operation frequency spikes on a few OSDs of `stor066`.

6 Mitigating Slow Ops

We have adopted several practices in responding to slow ops:

- When the load on Ceph is high, HammerCloud exclusions result in relief from the load from the abatement of user jobs. Consequently, the slow ops abate, and the incident goes away on its own. This behaviour can be mimicked by placing harder caps on user workloads during periods of slow ops when Ceph operations must be prioritised.
- Pause Ceph scrubbing operations to reduce OSD load.
- Reduce the number of OSDs concurrently backfilling, or even pause completely until the current issues are resolved. See Figure 9.
- Identify if a particular OSD (hard disc) or Ceph host is suffering from faults, and replace or repair as needed. Monitoring can aid in the identification of these pressure points.

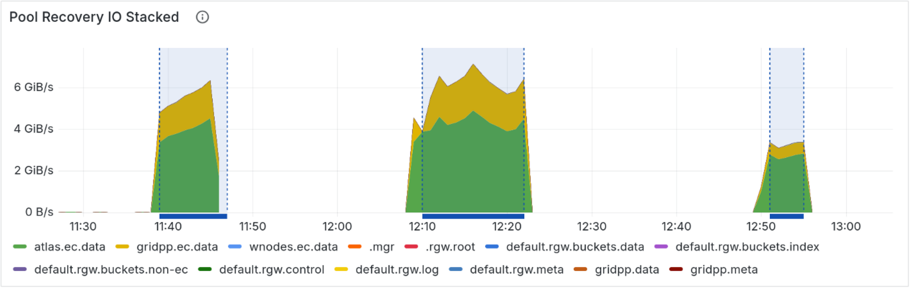

Figure 9. The blue bars show the periods where slow ops were active. Starting recovery seems to instigate slow ops and once recovery has been turned off the slow ops resolve.

7 Conclusion

A storage system with erasure-encoding redundancy across a cluster of local storage nodes will inevitably experience high internal maintenance loads that create excess stress on the system, yielding slow ops. We have recounted some typical incidents, and shown how they can be monitored, prevented and mitigated, with varying degrees of success.

References

[1] (SLAC/CERN, 2025) The XRootD Project. Available at: https://xrootd.slac.stanford.edu (Accessed March 2025)

[2] (Ceph Foundation, 2025) Ceph, The future of storage. Available at: https://ceph.io/en/discover/ (Accessed March 2025)

[3] R. Jones, M. Doidge, G. Hand, P. Love and S. Simpson, “A Blueprint for a Contemporary Storage Element, building a new WLCG storage system with widely available hardware and software components: Ceph, XRootD, and Prometheus”, EPJ Web of Conferences Volume 295 (2024) 01002 https://doi.org/10.1051/epjconf/202429501002