GridPP Storage Ops

In the Twilight of GridPP6

GridPP49

Matt Doidge, for the GridPP Storage Group.

Many thanks to everyone for their contributions.

Intro

This has turned into a “State of the Union” talk, attempting to describe the recent, current and near-term plans and storage activities of the GridPP sites, as well as mention some of the other activities of the GridPP Storage Group.

My apologies - it’s a lot of text.

Disclaimer: Matt tried to find out or remember what the storage plans for sites are, but some assumptions were made and information may be stale.
Feel free to shout at Matt and tell him he’s wrong.

Going Through Changes

- DPM EOL is June this year
  - Just in a migration holding pattern, and has been for a while.
- For UK DPM sites this has been the prompt to jump to either a new solution or take the plunge into the Post-Storage Age.
  - Although TBH this has been a long time coming.
    - Many sites experienced performance issues.
    - Feature light and wasn’t getting any better.
    - Although to be fair on DPM, when it worked it worked quite well.
  - Bristol already had to make the jump after HDFS support got pulled.
  - Birmingham have EOS for ALICE
- For sites maintaining storage in the long term XRootD on top of something (e.g. CEPHFS) seems to be the favoured option.
- For “Post-Storage” sites there is the option of using some kind of (X)cache or a “direct connection” to another site for their data needs.
  - Or a mixture of the two.

Holding Course

It’s worth noting the sites that aren’t engaging in any migration or seismic storage shifts:

- Imperial: dCache (CMS)
- QMUL: STORM* (ATLAS)
- RALPP: dCache (CMS)

Outside looking in, dCache and STORM have been sturdy (I think the only thing from recent memory I can remember dCache being iffy with is the SRR, and I think that’s working now).

- Also Birmingham will maintain running EOS for ALICE.

*Lustre backend.

DPM->dCache Migration

3 sites have or are migrating their DPM to DCache

- ECDF
  - Rob C pioneered the migration, with only minor peril.
- Lancaster
  - Should have migrated, but got stuck with by not preempting the time required for the pre-migration database tidy up.
  - Plan to go before Easter.
- Liverpool
  - Plan to migrate shortly after Lancaster. Already learned from the former’s mistakes.

It’s worth noting that all these are “temporary” solutions (~1 year), aimed at riding out current hardware life and experiment obligations.

The site not migrating (AFAIK) are Manchester, Durham, Brunel and RHUL - all going straight to their next step, decommissioning the DPM along the way.

Decommissioning and Clearing Up

Over the last few years we’ve had some experience with decommissioning Storage Elements, and I don’t think it’s ever gone as smoothly as we’d like - best case scenario is often deafening silence from the VOs.

Alessandra has created a script that checks for unused, unread data at LOCALGROUPDISK, which will be a useful tool in dealing with the data-hoarding users.

Recommended Storage Solutions

Sites that aren’t holding their course have two broadly recommended storage methods going forward

- Going Storageless, getting their data from elsewhere with the option of deploying XCache to smooth things along.
- Remaining Storage-y, with the recommended deployment being XRootD layered on top of some block of Storage.
    - CEPH/CEPHFS has become dominate here, but it important to note that any workable storage solution is viable.
    - In the event of “free” storage being made available this would be the advised way to leverage and expose that resource.

XRootD on top of: Object Store (CEPH)

Glasgow and the TIER 1 have well documented experiences with their XRootD frontends to CEPH object storage.

It’s important to note the front-loading of effort put in at RAL and Glasgow to make this work smoothly. Many (many, many) dev-hours went into improving xrootd-ceph performance, and still many more are and will be put in to add new improvements or features.

The choice to go with an object store is largely due to performance (no filesystem metadata to slow your down), but there was also a component of timing, with CEPHS not being as mature when these systems were being designed and commissioned.

A note on CEPH

In a statement that might not surprise anyone, it’s important to note that CEPH is not something you can just install and mount.

- It is a system all of its own.
  - A reasonably steep learning curve.
  - Need to design your infrastructure.
- It works best with purposely designed hardware (making repurposing the existing storages troublesome).
  - Most notably all disk nodes need some fast disk for peak performance.
  - And it works best with in-warranty hardware.
  - Inter-CEPH network traffic is non-trivial.
- It requires monitoring to maintain health.
- Major version upgrades are non-trivial.
- But once it gets working it is fairly solid, requiring tending instead of interventions.
  - The “killer feature”, at least for Lancaster, was it mitigates the risk of data loss for a single server.

XRootD on top of POSIX (mostly CEPHFS)

CEPHFS has become dominant in the UK, but it’s important to again note it’s not the only one.

The Lancaster experience of XRootD fronting a CEPHFS backend is well documented:

- Similar motivations to Glasgow for choosing CEPH, but choose the more familiar environment of a filesystem in CEPHFS rather then an object store.
    - Still quite a lot of complexity to make work at scale.
        - Gotchas almost all on the XRootD side.
        - Despite 18 months of operation the last “piece of the puzzle” (ATLAS jobs reading straight from the mount) only went online this month.
            - Thanks to James and Alessandra for pushing that forward.
- Also gone/going down this road:
    - Glasgow (for non-atlas groups)
    - Bristol (currently HDFS, eventually migrating to CEPHFS)
    - Brunel
    - Manchester
    - Durham

XRootD support group

Between the Storage group, Mattermost channels and a weekly Tier-1 focussed XRootD/CEPH dev meeting chaired by James a strong UK XRootD support network has developed.

- Shared experiences and best practices
- Comparing bugs and workarounds.
- Future: focussing dev effort?

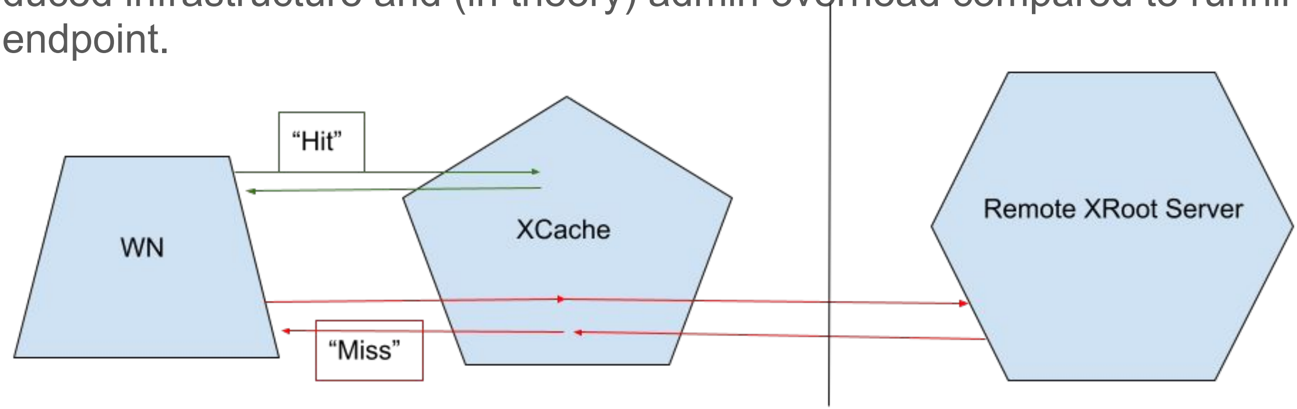
Choose your adage: “A Problem Shared is a Problem Halved” or “Misery Loves Company”.

(there’s also a lot of shared CEPH knowledge too, but that’s not as traumatic).

“Storageless” Option - XCache

An option for sites wanting to drop their storage but still wanting to not be completely at the mercy of the WAN.

- No expectation of data retention, no custodial duties (implied or otherwise).
- Can operate in passthrough mode, or even just set to be ignored by jobs.
- File “reuse” can reduce total bytes transferred, reducing pressure on network and the “other end”.
- Much reduced infrastructure and (in theory) admin overhead compared to running a full storage endpoint.

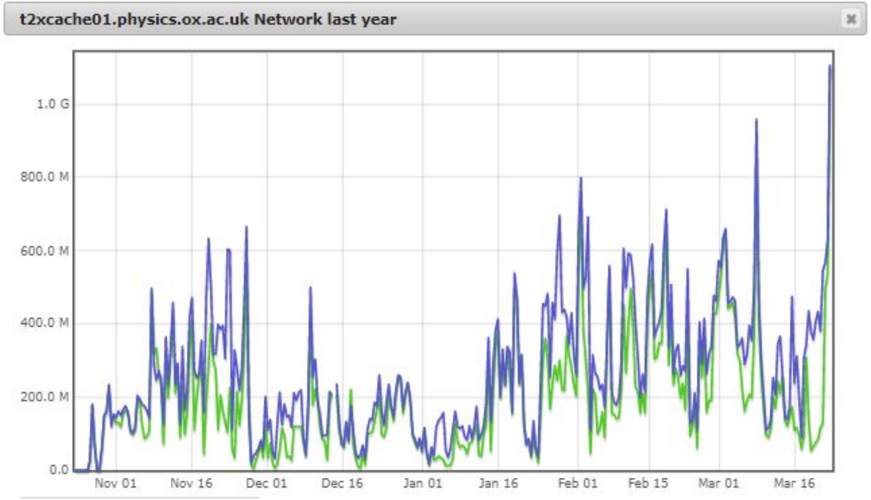


There are many good XCache diagrams in the world, but it was quicker for me to throw this one together.

XCache only

Oxford and Birmingham have been running XCache's for atlas jobs for a while, pulling data from RAL or Manchester respectively. RHUL is a more recent addition to this work, whos cache is able to pull from anywhere.

- Intention to smooth and reduce load on the "linked" site's storage.
  - Particularly important for Oxford and the RAL gateways.
- A lot of effort into investigating and tuning.
  - "Real" results were a little disappointing for Oxford - cached file reuse tended to be poor and job efficiencies often seemed independent of cache use.
  - Have "borrowed" a high-performance SSD-filled server from RAL to see how that affects cache efficiency.
  - RHUL however have had good performance with decade old repurposed disk servers.


t2xcache01.physics.ox.ac.uk Network last year

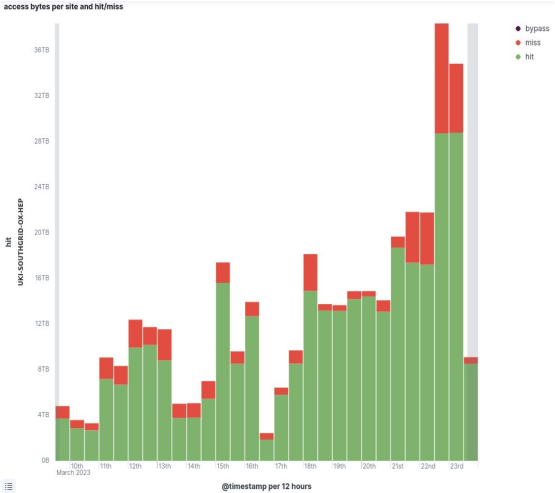
Green "In", Blue "Out" from the XCache.

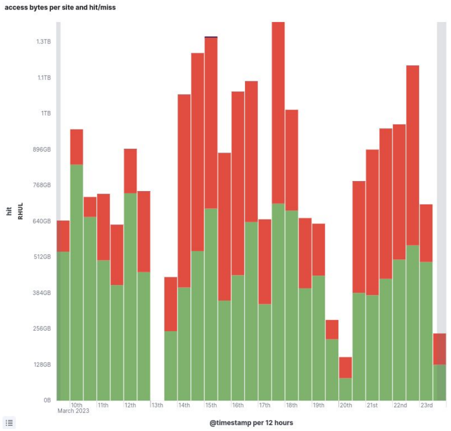
Virtual Placement

The short explanation: VP is pre-emptive pre-caching for atlas jobs, performed using rucio (and hence could be shipped to other rucio using groups).

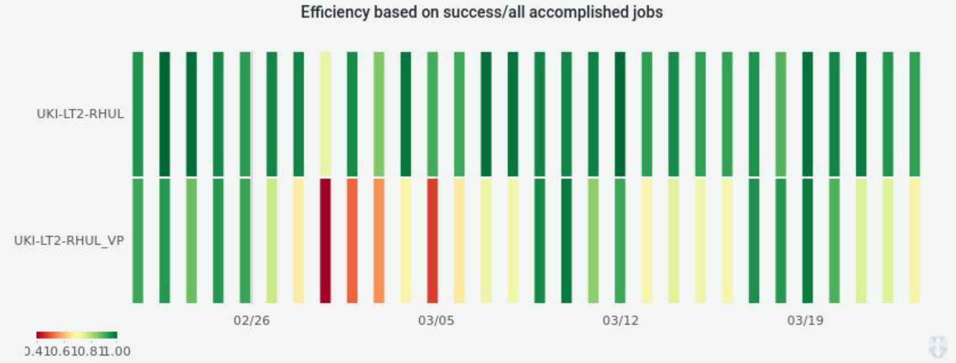
- Being rolled out and tested at Birmingham, RHUL and Oxford.
- Uses the same hardware as the XCache.
- As with XCache this is designed as a Containerised Install, with the official preferred roll out method being Kubernetes (and I think remotely, over SLATE or similar).
    - Although all the UK sites have deployed “by hand”, either manually or deploying the docker images themselves.
- Seems to work okay when you get get the config right.

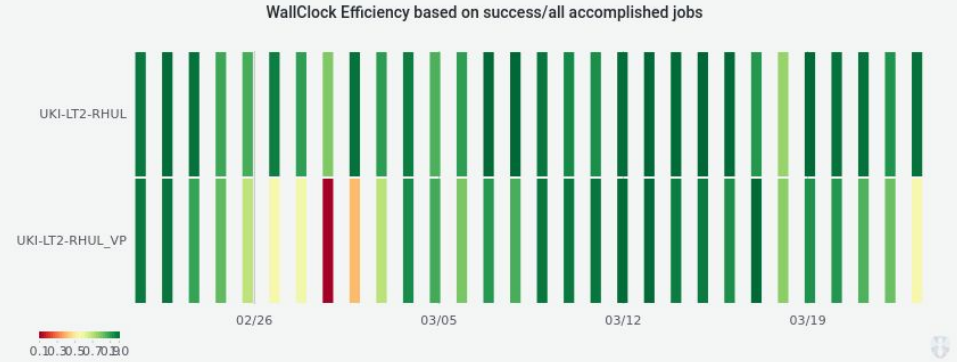
XCache/VP Performance Plots (from these web sites)





Left: Hit and Miss plots for Oxford and RHUL
Below: Atlas Job efficiencies for RHUL's "regular" and VP queue (thanks Simon for these).





Truely Storageless - No Cache

Two sites don’t run with any local storage or cache: Sussex and Sheffield.

- Pull (and push) their data from QMUL and RAL respectively.
- Largely works with some tailoring of job loads.

A future possible scenario for Liverpool is to use Lancaster’s storage.

- Upcoming improvements to Liverpool’s networking.
- Sites physically close, seems sensible.
- But is it optimal? We have the Perfsonar mesh to provide network rates, and there are efforts to build a “Data Transfer” mesh.
- Possible future procedure could be to audit and review site “twinings”.

Some of the other work on the Storage Group’s list

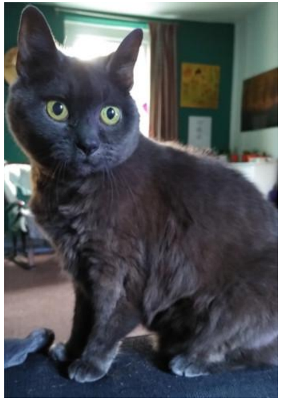
- Stashcache
- XRootD Monitoring, Shoveller
    - Debugging the XRootD monitoring requires a lot of R&D
- Volatile Storage
- Rucio (largely via Tim).
- Tokens (the next big crusade)
- “Onboarding” New User Groups - w.r.t. Storage, Data Management
- “Education” (e.g. the XRootD Installation 101)

In Conclusion

- A core set of sites are staying their course. These are notably the sites that never touched DPM.
- The official EOL of DPM hasn’t shaped operations as much as you’d have thought, and the migration path to DCache only has a few takers.
  - Sites plans have more been shaped by the volatility of the middleware and their aging hardware.
  - Many already jumped ship due to local pressures.
- Strong support within the GridPP Storage Community
  - A lot of experience and expertise gained the hard way with CEPH, XRootD and now XCache
  - Grand engagement and support from ATLAS (particularly James and Illija), without which there wouldn’t be any progress.
  - Thanks to everyone involved with the efforts mentioned here, and for the help producing these slides.
  - And special thanks to Sam for steering the Storage Group.

Landscape at a Glance: Status and near-term plans

- **TIER-1:** ECHO
- **GLASGOW:** XRootD/CEPH/CEPHFS
- **ECDF:** dCache'd DPM -> VP eventually
- **DURHAM:** DPM -> XRootD/CEPH
- **LANCASTER:** XRootD/CEPHFS, temp dCache'd DPM
- **LIVERPOOL:** temp dCache'd DPM -> ?? (possible satellite site to Lancaster).
- **MANCHESTER:** DPM -> XRootD/CEPHFS
- **SHEFFIELD:** Satellite Site (RAL)
- **BIRMINGHAM:** EOS (for ALICE), VP'd XCache (Manchester)
- **OXFORD:** XCache -> VP'd XCache (RAL)
- **BRISTOL:** XRootD/HDFS -> XRootD/CEPHFS
- **RALPP:** DCache
- **SUSSEX:** Satellite Site (QMUL)
- **RHUL:** DPM -> VP'd XCache (QMUL)
- **BRUNEL:** DPM -> XRootD/CEPHFS
- **IMPERIAL:** DCache and proud.
- **QMUL:** STORM



Obligatory Cat Picture

Omni-Slide

- Time of Transition, fall of DPM
- DPM is dead
  - dCache Migration: ECDF, Liverpool, Lancs. Note all “temporary”.
  - Moving to something else- Manchester, Brunel, Durham
- Long live dCache (sticking with what they have) - Imperial, RALPP
  - Or STORM - QMUL
  - Or EOS - Birmingham (ALICE only)
- Or just using someone else’s storage (Sheffield, Sussex, QMUL/IC duality, pre-VP Birmingham and Oxford with XCaches).
- XRoot on top of a filesystem - running: Bristol (with HDFS bells on), Lancs
- XRoot on top of a filesystem - planned: Brunel, Durham, Manchester
  - What filesystems? (mainly CEPHFS)
- XRoot on top of an object store - Glasgow, RAL
- XCache VP - RHUL, Oxford, Birmingham.
- XRoot plans (testbed, monitoring, shoveller, nod to James and Katy at the workshop).