Caching technologies for Tier-2 sites: A UK perspective

Samuel Cadellin Skipsey[1], *Chris* Brew[2], *Alessandra* Forti[3], *Dan* Traynor[4], *Teng* Li[5], *Adam* Boutcher[6], *Gareth* Roy[1], *Gordon* Stewart[1], and *David* Britton[1]

[1]University of Glasgow, Glasgow, G12 8QQ, UK
[2]Rutherford Appleton Laboratory, Harwell Campus, Oxon, OX11 0QX, UK
[3]The University of Manchester, Oxford Road, Manchester, M13 9PL, UK
[4]Queen Mary University of London, Mile End Rd, London E1 4NS, UK
[5]University of Edinburgh, King's Buildings, Edinburgh EH9 3FD, UK
[6]Durham University, Stockton Road, Durham, DH1 3LE, UK

Abstract. Pressures from both WLCG VOs and externalities have led to a desire to "simplify" data access and handling for Tier-2 resources across the Grid. This has mostly been imagined in terms of reducing book-keeping for VOs, and total replicas needed across sites. One common direction of motion is to increasing the amount of remote-access to data for jobs, which is also seen as enabling the development of administratively-cheaper Tier-2 subcategories, reducing manpower and equipment costs. Caching technologies are often seen as a "cheap" way to ameliorate the increased latency (and decreased bandwidth) introduced by ubiquitous remote-access approaches, but the usefulness of caches is strongly dependant on the reuse of the data thus cached. We report on work done in the UK at four GridPP Tier-2 sites - ECDF, Glasgow, RALPP and Durham - to investigate the suitability of transparent caching via the recently-rebranded XCache (Xrootd Proxy Cache) for both ATLAS and CMS workloads, and to support workloads by other caching approaches (such as the ARC CE Cache).

1 Introduction

The provision of computing and storage resources across the WLCG, and especially in the UK under GridPP, is subject to increasing requirements to improve efficiency, in terms of both cost and staffing. In the UK, the effect of this pressure on storage resource provision has been to drive a move towards "simplifying" storage[1], in accordance with survey results suggesting that Grid storage services were responsible for a significant fraction of the FTEs needed to run a site.

This impetus is in alignment with a general desire from the WLCG Experiments, and in particular ATLAS, to have a simpler, more streamlined view of the resource available to them. The UK is host to a particularly large number of Tier-2 facilities, for historical reasons, which are grouped into "Regional Tier-2" units as represented in figure 1. This grouping is loosely administrative, but all Tier-2s expose separate compute and storage resources, making GridPP the most "endpoint-rich" region in WLCG. This plethora of endpoints could be made more tractable by removing or consolidating resources between sites, whilst still retaining all, or most, of the Tier-2s themselves.

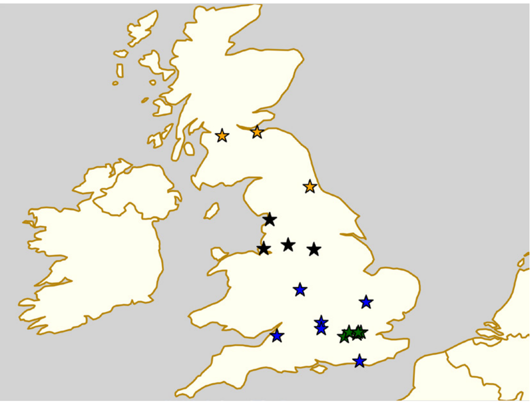

Figure 1. Map of GridPP site locations, coloured by Regional Tier-2. ScotGrid: yellow; NorthGrid: black; SouthGrid: blue; LondonGrid: green.

Some steps have already been taken on this path. For CMS, all their "Tier-3" resources at Tier-2 sites (comprising services at Glasgow, Oxford, Queen Margaret University of London (QMUL), Royal Holloway University of London (RHUL) and Bristol) have operated without local storage for more than a year, using Imperial College, RAL PPD and Brunel's SEs via an Xrootd redirector. For ATLAS, University College London uses QMUL's storage as its “close SE", as does Imperial College. More recently, the sites at Sussex and Birmingham have both moved to using remote storage to support some or all of the experiments running at them. In the case of Birmingham, where the transition happened in October 2018, after the work in this paper was completed, the resulting increase of network IO against the remote storage host (at Manchester) was significant enough to cause issues for both sites. This retrospectively justifies the work presented in the rest of this paper, summarising various tests of ways to limit or reduce the impact of remote access on storage elements.

In the limit, we might expect a case where a small number of Tier-2 sites host all of the storage for GridPP, with all other Tier-2s pulling data from them remotely. (For some experiments, such as CMS, this could be configured as a general pool; for others, such as ATLAS, at present only "pairing" relationships (matching a given site with a single remote SE) are available.) This change of access presents obvious potential issues: both in terms of increased latency (and decreased maximum bandwidth) for workflows at storageless sites; and in terms of increased load on the consolidated "data Tier-2s" hosting the storage itself.

One approach for mitigating all of these issues has been the implementation of various "caches" at storageless sites, in order to prevent excess transfer of data and to allow that data which has been transferred to potentially be accessed at lower cost.

2 Caching Approaches

In general, the word "cache" is used fairly loosely in the WLCG context, as any kind of temporary storage used to improve access to data that is expected to be frequently accessed. In fact, some implementations of "caches" in the context of WLCG data policy blur the boundaries between "buffers" and "caches" (the distinction being one of intent). It is also notable that almost references to "caches" in these contexts are read caches; we will follow this tradition and not discuss the caching of output from jobs or workflows.

For the purpose of this paper, we will break down the aspects of caching policy in terms of the amount of prior knowledge involved in decisions to place particular items into the caching system: "pre-emptive" caches, which are filled with data that is known to be useful to workloads before those workloads begin; versus "opportunistic" caches, which retain local copies of data the first time it is accessed remotely, such that future accesses are made more efficient. (In the latter case, the first transfer which fills the cache is also a buffering operation, and may benefit from, or be harmed by, the resulting performance difference to a direct access, depending on the cache design.)

We will also distinguish between caches for "remote" accesses; that is, caches which sit between the WAN and the LAN, and reduce latency by increasing network locality of the data; and "local" caches, which sit between already network-local storage and the compute resource, and reduce latency by increasing the available IOPs of the host storage. (In this sense, jobs which operate in the "copy to local WN (worker node)" mode for data access are a simple version of the latter case, as the local WN's storage is often less contended, and lower latency, than the site SE infrastructure; "copy to WN" also has the advantage of implicit parallelism with respect to a "dedicated cache" version of the same approach, as the "cached" data is distributed across a large number of WN disks, rather than requiring a single, high performance system capable of serving IO to many WNs at once.)

In the context of this light taxonomy, we present three small case-studies from the UK's testing over the previous year. One of these is a "pre-emptive" cache, whilst the latter two are variants of opportunistic cache derived from XrootD proxies.

2.1 ARC Cache at UKI-SCOTGRID-DURHAM

The ARC[2] Compute Element, developed by NorduGrid, was designed from the start to support distributed Tier-2s, and thus has mechanisms to minimise data movement for jobs available. In particular, any ARC CE can be configured to prefetch data dependencies for a submitted workload in a site-local filesystem, only submitting the dependant job to the local batch system when the data is fully locally available. In the semantics of our simple taxonomy, ARC Caches are an example of a pre-emptive cache; the first time a workload accesses data in an ARC Cache, that data must already exist in the cache (having been transferred before the workload even begins). There is no mechanism for filling the ARC Cache from within a workload, once that workload has begun execution.

This "ARC Cache” is not widely used by WLCG experiments other than ATLAS. One reason for this is the dependence of the larger Virtual Organisations on "pilot jobs" for workload submission - the jobs seen by the site ARC will be the pilots (which have no data dependencies), and the actual workloads (which are pulled from remote sources by instantiated pilot jobs) are not visible to the ARC. The ATLAS experiment, however, has performed significant work on their "ARC Control Tower” (ACT)[3], essentially a conversion shim which interfaces with the ATLAS pilot system to pull real workloads and then submit them directly to ARC sites which wish to use caching. This mechanism has been successfully used to submit to the NorduGrid project resources for many years, and with great success.

The Durham Tier-2 site, which has a particularly large CPU count in comparison to its storage, has, since July 2016, configured their ARC CE to enable caching, against a shared filesystem (exported over NFS to the worker nodes), and has been added to the ATLAS ACT list so that they can receive jobs from that service.

Results (presented in the ATLAS Sites Jamboree, January 2017 [4]), show that workloads from the ACT are at least as efficient as conventional ATLAS pilots (which the site continues to receive). However, the total storage used by the cache is comparatively small, as figure 2 shows.

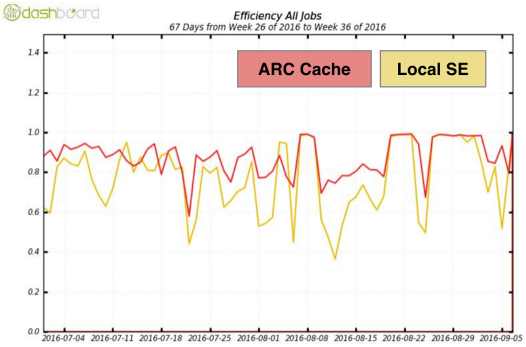

Figure 2. Reproduction of measured efficiencies for ATLAS jobs at the Durham site, running via conventional ATLAS pilots (no cache) and via the ACT/ARC Cache, from ATLAS Jamboree presentations.

Despite the success of our trial, and the simplicity of implementing the caching in an existing ARC CE, this approach is limited by the lack of equivalents to the ACT for other pilot-dependant VOs. That said, the significant number of ATLAS-supporting sites in the UK means that this option is still of use for reducing the impact of ATLAS-related workloads.

An additional issue, which became evident later in 2017, was the distribution of expertise needed to manage "specialist" approaches such as ACT-backed job submission. As the Durham site is unique in the UK in having this configuration, resolution of issues concerning ACT jobs has been much slower than for general cases of issues on their non-ACT queue. Despite efforts to encourage majority-ATLAS GridPP sites to adopt ARC Cache as a simplifying solution for their storage, Durham remains the only such site to do so.

2.2 Xrootd Proxy "XCache" at UKI-SOUTHGRID-RALPP

With a significant amount of internal experience in Xrootd proxy configuration at the RAL Tier-1, we decided to extend our test on remote access for CMS workloads by testing the effectiveness of Xrootd Proxy Caches[5] (since rebranded to "XCache") in improving performance against the "AAA" federated Xrootd storage. This work required little additional effort, as Xrootd Proxies were already being set up to help manage network flows in the RAL LAN/WAN itself. Caching is an additional feature which any Xrootd Proxy can enable independently, so we simply activated this for a period on relevant Proxies.

The RAL Tier-2 ("RALPP") was configured within the CMS job management system to accept jobs with no local data (accessing data entirely over the "AAA" federation), to ensure that we could measure the efficiency of jobs without contamination from local, low-latency, accesses outside the proxy.

Our initial configuration for the cache component of the proxy was for it to perform full-file caching on all data requested through it, using a moderately-performant local filesystem as the backing store. This was a deliberately trivial configuration, and was not expected to be optimal for the site; the aim was to attain initial data on the patterns of file re-use for jobs running at the site.

Measurement of the cache efficiency showed that cache hit rates were exceptionally low, resulting in an exceptionally low caching factor figure 3. As can be seen, on average, much more data - one to two orders of magnitude more - was transferred into (and through) the cache from the remote AAA sources than was ever returned from local hits on cached files.

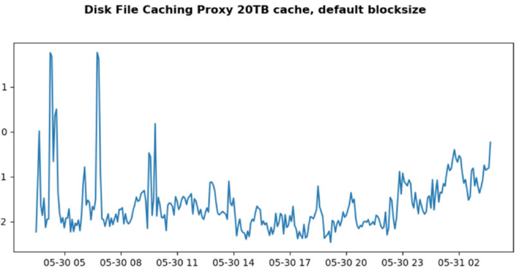

Figure 3. Calculated caching factor (ratio of bytes read into the cache to bytes written out by the cache) for the RALPP Xrootd Proxy Cache test. Smaller numbers are worse; a value of zero is a “neutral" proxy.

In addition, the IO load on the proxy cache became exceptionally high, as the underlying disk layer spent almost all its time writing new files to disk. The resulting load became high enough to saturate the storage subsystem entirely and increased latency on accesses through the proxy as a result. Analysis of the cache contents itself showed that files were almost exclusively accessed only once, corresponding to the point at which they entered the cache; hence the cache itself could have no positive effect other than as a buffer. (CMS AAA jobs tend to access only byte ranges in remote files, so fetching a whole copy of a file could reduce latency on subsequent accesses to other ranges in the file.)

Turning off the on-disk cache, and using the Xrootd Proxy as a simple proxy with a small in-memory cache, significantly improved performance compared to the previous case, and the proxy maintained performance at near-network-rates for all requests against it.

2.3 Xrootd Proxy "XCache" at UKI-SCOTGRID-ECDF

Work at the Edinburgh site, UKI-SCOTGRID-ECDF, involving Xrootd proxy caches was driven partly by the interests of the ATLAS experiment, after discussion as to their needs. Originally, it was intended that work at at least one site would replicate the RALPP testing for CMS, with ATLAS workflows. However, the current ATLAS workflow management system at the time of testing was unable to "naturally" implement a Site which could take data from anywhere, so this was impossible.

The work done at Edinburgh, then, resulted in tests of Xrootd Proxy Caches in a local configuration, with the proxy interposed between the local storage system and a small subset of the sites worker node pool. The full discussion of this work, and its applicability to modelling performance of caching for ATLAS workflows in the general case, is left for the separate paper dedicated to this in this journal[6].

However, we will bring up one result in this paper in order to compare it with the other cases. Monitoring of cache hits and misses within the configured system showed that there were two populations of files accessed by ATLAS workloads, distinguished by how well they could be cached. "Data" files (large root files) were very badly cached, with a median access frequency of only 1, whilst "library files" (detector conditions databases etc) had much higher median access frequencies. This is to be expected from *a priori* analysis of the way in which ATLAS workflows parallelise: a set of jobs in a given workflow will share the same "library files", as they define the parameters of the run itself; but each job will run over a disjoint subset of the total "data files" allocated to the run, as runs tend to be event-parallel. In one sense, this result is unsurprising, and did not need a cache or proxy to discover - indeed, our previous

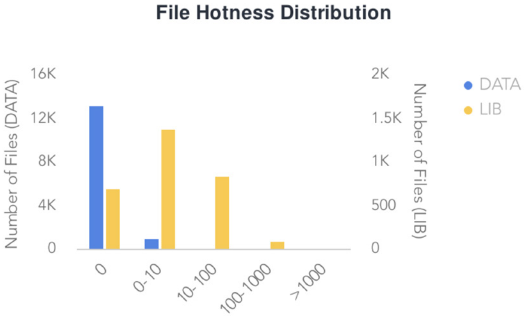

Figure 4. Graph of relative distributions of "File Hotness", corresponding to the number of times each file was accessed via the cache (equals number of total accesses -1) for the two populations of ATLAS data identified in the ECDF work.

paper on this topic from CHEP 2016[1] showed similar access patterns from a simple analysis of the internal logs of the UKI-SCOTGRID-GLASGOW SE. Advantageously, this approach also naturally revealed other, non-ATLAS, access patterns, as reported. It would have been trivial for any experiment to extend this analysis to other sites for increased statistical validity. Indeed, collecting the same results via analysis of the proxy cache logs involved considerably more work, including the effort in setting the proxy cache itself, and required development effort beyond the simple SQL query needed to access local SE records.

We note that, in this special case of ATLAS data flows tending towards accessing each file only once, that the ideal "caching" model is, in fact, a WN local buffer; something similar to the "copy to WN" approach already available as a data access option. This exploits the natural parallelism of the workflows as designed, and avoids the need for additional expense on highly performant caches. Readers may compare this to the similar architectural decision made by the RAL Tier-1 in providing node-local Xrootd proxy caches to mitigate access to their Ceph-backed object storage[7].

3 Future Work and Recommendations

In addition to the presented statistics, we have other projects in development to investigate other modes of simplification. With the advent of the WLCG Data Organisation Management Access (DOMA)[8] working group, we anticipate these projects becoming subsumed with or additionally driven by the interests and effort available as a part of that group.

In general, we note that much of the benefit of caches is hard to realise in any of the cases that we have examined: data is rarely read more than once, so the access used to fill the cache is most often the only access which occurs. In cases where caches are useful, they're more akin to classical buffers than true caches, suggesting that managed data flows might be a more useful way to limit the effect of network latency on jobs than laissez-faire "opportunistic" approaches.

Important additional data on ATLAS workflows in particular is being gathered as part of the current (temporary) configuration of the Birmingham site to use the Manchester site storage as a remote repository. We hope that this can feed back into both ATLAS policy for remote storage paired sites, and into better models for GridPP-local provision.

One potential use case of caching approaches (albeit still effectively as a managed buffer) is to hold reconstructed stripes of objects from geographically distributed object stores (often referred to as "Data Lakes" in CERN presentations). As this architecture would provide significant simplification for our client experiments, it is a technology which we have committed to investigate. We hope to provide updates from this work at future conferences, and as part of the DOMA Quality-of-Service taskforce.

References

[1] Samuel Cadellin Skipsey et al, J. Phys.: Conf. Ser. **898** 062047 (2017)

[2] Ellert M et al, Future Gener. Comput. Syst. **23** (1):219240 (2007)

[3] Nilsen J K et al, J. Phys.: Conf. Ser. **664** 062042 (2015)

[4] ATLAS Sites Jamboree, Indico https://indico.cern.ch/event/440821/

[5] Dorigo A Elmer P Furano F and Hanushevsky A , Proceedings of the 4th WSEAS International Conference on Telecommunications and Informatics **TELE-INFO'05** 46:1–46:6 (2005)

[6] Teng Li et al, EPJ Web of Conferences **CHEP 2018** These Proceedings (2019)

[7] Rob Appleyard et al, EPJ Web of Conferences **CHEP 2018** These Proceedings (2019)

[8] S. Campana, EPJ Web of Conferences **CHEP 2018**, These Proceedings (2019)