XRootD caching for Belle II

Moritz Bauer, Max Fischer, Manuel Giffels, Günter Quast, Matthias Schnepf

KIT – Karlsruhe Institute of Technology, Germany

E-mail: moritz.bauer@kit.edu

Abstract. The Belle II experiment at the second generation e^+/e^- B-factory SuperKEKB has been collecting data since 2019 and aims to accumulate a 50PB data set. To efficiently process these steadily growing data sets of recorded and simulated data as well as support Grid-based analysis workflows using the DIRAC Workload Management System, an XRootD-based caching architecture is presented. The presented mechanism decreases job waiting time for often-used data sets by transparently adding copies of these files at smaller sites without managed storage. The described architecture seamlessly integrates local storage services and supports the use of dynamic computing resources with minimal deployment effort. This is especially useful in environments with many institutions providing comparatively small numbers of cores and limited personpower.

1. Introduction

The Belle II experiment [1], at the SuperKEKB accelerator [2] at the High Energy Accelerator Research Organization (KEK) is dedicated to research at the intensity frontier. For centralized MC production as well as user analysis, Belle II uses a computing grid spanning approximately 30 compute centers. This decouples workload requirements from individual computing sites, thereby increasing reliability and accessibility for all members of the collaboration. To manage this diversity of resources, Belle II has selected the DIRAC middleware [3] for workload management and the data management system Rucio [4] to distribute data sets among grid storage endpoints — so-called "Storage Elements" — worldwide.

Grid-based approaches in general average out the computing demands of large collaborations by distributing workloads between the resources. This presents challenges for workloads requiring specific data sets, workflow management systems only submit jobs to sites which have local copies of these data sets ("data locality"). This requirement for data locality, which decreases network strain caused by transferring data over wide-area networks, significantly reduces the number of eligible sites for specific jobs. While systems such as Rucio can be used to increase the amount of replicas of popular data sets and thereby the number of sites at which jobs requiring them can run, this method requires identification of popular files first. Additionally, this approach also assumes static availability of CPU resources at computing sites which is not always guaranteed at e.g. resources explicitly dedicated to HEP analysis "opportunistic resources" [5] A consequence of imperfect distribution of data sets and thus jobs are high numbers of waiting jobs at individual sites as depicted in Figure 1.

Transparent, dynamic caches at individual sites present an alternative solution for this problem. They can be used to dynamically add copies of popular data sets to storage

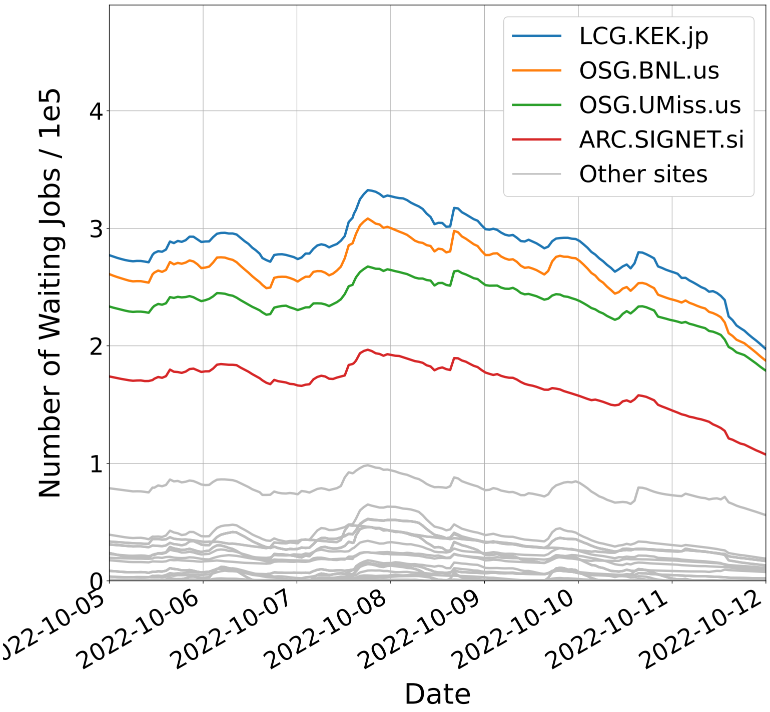

Figure 1. Number of waiting jobs at different Belle II grid computing sites, captured during one week. Unequal resource demands lead to more than 100,000 jobs at four sites with a maximum of roughly 300,000 waiting jobs at a single site.

systems which do not have to fulfill the requirements on redundancy, reliability, and long-term commitment usually imposed on Storage Elements. In addition to increasing the utilization of resources already integrated into the grid, lightweight caching solutions could also allow sites which currently do not meet the reliability requirements for grid sites to contribute storage and CPU resources to the Belle II collaboration.

2. Caching Files with XRootD

The XRootD project [6, 7] provides a data transfer protocol as well as server and client implementations to efficiently transfer files between computing sites connected via wide-area networks. The XRootD protocol is particularly suited to transfer large files typically found in HEP environments and allows partial transfer of supported file types (“streaming”). The XRootD protocol has been most notably adopted by the ALICE collaboration as the primary data transfer protocol. Via plugin interfaces in both the XRootD client and server, additional protocols such as HTTP can be supported. This is desirable for application in the Belle II grid: While XRootD protocol support exists at multiple computing sites, all sites must support HTTP.

Also supported via the XRootD plugin interface are disk-based proxy caches (“XCache”). As shown in [8] and [9], XCaches provide opportunities to reduce network bottlenecks and increase the CPU utilization of jobs. Both the XRootD server and the associated XCache plugin are relatively lightweight and require little configuration. Deployment on a single server

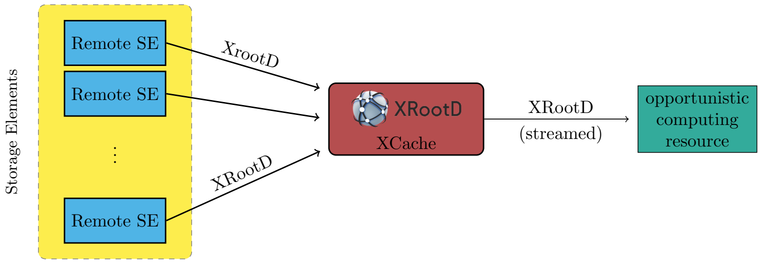

Figure 2. Schematic depiction of file access through an XRootD XCache.

is straightforward, with the option to expand to multiple machines via XRootD's regular cluster manager mechanism if the demand can not be satisfied by a single machine anymore. Via the XRootD streaming mechanism, the XCache can provide files already once they are partially downloaded, introducing almost no delay compared to access without the cache.

To redirect selected file access to the XCache, a XRootD client-side plugin is available as part of the XRootD software package. The redirection is completely transparent to the user as well as the DIRAC and Rucio systems.

3. Implemented Setup

An XCache instance is deployed to a node at the WLCG Tier1 / Belle II Raw Data Center "GridKa" which allows access to several opportunistic computing resources [5]. These resources provide up to 2200 CPU cores to Belle II and are referred to as "LCG.KIT-TARDIS.de". On these opportunistic computing resources, all read access via the XRootD client is redirected by the redirection plugin mentioned above. No caching is performed for write access via XRootD or other protocols.

The 40-core server node running the XCache is equipped with 256 GB of memory and a 100 Gbit/s network interface. Via this network interface, the machine has access to a `ceph` distributed storage cluster [10] of which 500 TB is available as storage space for the cache.

To test the functionality of the XCache instance and the associated performance monitoring, all read accesses to the `GridKa` storage back-end by Belle II jobs on `LCG.KIT-TARDIS.de` is redirected to the XCache instance at GridKa. This does not add any performance or job load distribution advantages as no network bottlenecks are removed between the opportunistic resources and the storage. It does, however, allow evaluating the file reuse rate of typical jobs in a Belle II production environment, both for centrally launched MC productions as well as user analysis workloads.

In a second step, the XCache also caches data located at the "Institut Jožef Stefan" (IJS) in Ljubljana, Slovenija. Here, actual performance benefits are to be expected, however not enough data on this has been collected yet as caching has only been enabled for roughly two weeks at the time of writing.

While caching the `GridKa` storage back-end is completely transparent to `DIRAC`, caching data located at remote sites requires changes in the central configuration. To match jobs to the individual sites, `DIRAC` uses information on the Storage Elements connected to each computing site. Amending the configuration allows jobs at the `LCG.KIT-TARDIS.de` computing site to access data located at IJS.

To evaluate the performance of the XCache, the network monitoring streams provided by XRootD are captured and unpacked using the `xrootdlib` python library [11].

The unpacked information is forwarded to an `elasticsearch` database [12] where it is

Table 1. Cache performance metrics, collected over 14 months and separated into types of jobs.

Job type	Written into cache (TiB)	Read from cache (TiB)	Average access count	Cache hit rate	Contribution to overall cache hits
MC production	18.72	573.10	30.5	96.7%	98.89%
Data analysis	2.37	5.97	2.4	57.5%	0.36%
MC analysis	11.37	15.47	1.4	27.6%	0.76%

collected for analysis.

4. Results

Over a period of 14 months, the XCache is operated in a Belle II production environment to demonstrate the utility of the technology. In the last two weeks of operation, the XCache also started caching from a remote site, however these data points have been excluded from the following results as they do not allow independent statistical analysis yet. Key performance metrics are collected in Table 1. Here the computing jobs are divided into three categories: Data from simulation (MC) production jobs, analysis jobs using MC and analysis jobs using recorded data. The first category makes up the bulk of jobs in the Belle II grid, an effect which is increased on the `LCG.KIT-TARDIS.de` site due to the specifics of its configuration. MC production jobs use only a limited set of input files, necessary to accurately describe beam-induced backgrounds. Due to the high number of jobs and a comparatively small number of input files, the cache hit rate for these file requests is high. The second category, analysis jobs using real data, shows a lower cache hit rate. The total contribution to cache hits is small but the cache hit rate is above 50%, most likely due to the limited and universal nature of all recorded data so far. The last category, analysis jobs on simulated data, contains a varied set of simulated input files. While data sets are centrally produced, they can be highly analysis specific. This explains the low cache hit rate.

For MC production jobs, clear benefits can be observed by employing XCache as average access counts above 30 indicate high re-use of data sets. For user analysis jobs on both recorded and simulated data, no final conclusion on the usefulness of caching data for user analysis jobs can be drawn yet. As the XCache storage is far from being filled with approximately 6.5% utilization and the data set sizes for Belle II are expected to grow rapidly with the amount of collected data, higher cache hit rates and average access counts are to be expected.

5. Conclusion

The deployment and evaluation of an XRootD XCache in the production environment used by the Belle II collaboration have been presented, both in a local demonstration context with no direct tangible benefits and, on a limited timescale, caching of a remote site with tangible advantages. The evaluated caching technology is completely transparent for both the workload management system DIRAC and the data management system Rucio and shows clear advantages for files used in the production of simulated data. Moving on, the project aims to study the effects of caching in greater detail to establish whether benefits can be found when caching much bigger amounts of files as needed for data analysis jobs.

References

[1] Belle II Collaboration Belle II Technical Design Report (*Preprint* `1011.0352`) URL `http://arxiv.org/abs/1011.0352`

[2] Akai K, Furukawa K and Koiso H 2018 *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* **907** 188–199 ISSN 0168-9002 URL `https://www.sciencedirect.com/science/article/pii/S0168900218309616`

[3] Tsaregorodtsev A *et al.* 2022 DIRACGrid/DIRAC: V7.2.50 Zenodo URL `https://zenodo.org/record/7071472`

[4] Barisits M *et al.* 2019 *Computing and Software for Big Science* **3** 11 ISSN 2510-2044 URL `https://doi.org/10.1007/s41781-019-0026-3`

[5] Böhler M *et al.* 2021 *EPJ Web of Conferences* **251** 02039 ISSN 2100-014X URL `https://doi.org/10.1051/epjconf/202125102039`

[6] Dorigo A, Elmer P, Furano F and Hanushevsky A 2005 *Proceedings of the 4th WSEAS International Conference on Telecommunications and Informatics* TELE-INFO'05 (Stevens Point, Wisconsin, USA: World Scientific and Engineering Academy and Society (WSEAS)) pp 1–6 ISBN 978-960-8457-11-9

[7] XRootD URL `https://xrootd.slac.stanford.edu/index.html`

[8] Bauerdick L A T *et al.* 2014 *Journal of Physics: Conference Series* **513** 042044 ISSN 1742-6596 URL `https://dx.doi.org/10.1088/1742-6596/513/4/042044`

[9] Li T, Currie R and Washbrook A 2019 *EPJ Web of Conferences* **214** 04047 ISSN 2100-014X URL `https://doi.org/10.1051/epjconf/201921404047`

[10] Weil S A, Brandt S A, Miller E L, Long D D E and Maltzahn C 2006 *Proceedings of the 7th Symposium on Operating Systems Design and Implementation* OSDI '06 (USA: USENIX Association) pp 307–320 ISBN 978-1-931971-47-8

[11] Fischer M 2022 Xrootdlib - Tools for working with the XRootD middleware URL `https://github.com/maxfischer2781/xrootdlib`

[12] Elastic 2023 Elasticsearch elastic URL `https://github.com/elastic/elasticsearch`