A case study of content delivery networks for the CMS experiment

C. Acosta-Silva[1,2], *J.* Casals[2,3], *A.* Delgado Peris[3], *J.* Flix Molina[2,3], *J.M.* Hernández[3], *C.* Morcillo Pérez[3], *C.* Pérez Dengra[2,3,*], *A.* Pérez-Calero Yzquierdo[2,3], *F.J.* Rodríguez Calonge[3], and *A.* Sikora[4], on behalf of the CMS Collaboration.

[1]IFAE, The Barcelona Institute of Science and Technology, 08193 Bellaterra (Barcelona), Spain
[2]PIC, 08193 Bellaterra (Barcelona), Spain
[3]CIEMAT, Scientific Computing Unit, 28040 Madrid, Spain
[4]Autonomous University of Barcelona, 08193 Bellaterra (Barcelona), Spain

Abstract. In 2029 the LHC will start the high-luminosity LHC program, with a boost in the integrated luminosity resulting in an unprecedented amount of experimental and simulated data samples to be transferred, processed and stored in disk and tape systems across the worldwide LHC computing Grid. Content delivery network solutions are being explored with the purposes of improving the performance of the compute tasks reading input data via the wide area network, and also to provide a mechanism for cost-effective deployment of lightweight storage systems supporting traditional or opportunistic compute resources. In this contribution we study the benefits of applying cache solutions for the CMS experiment, in particular the configuration and deployment of XCache serving data to two Spanish WLCG sites supporting CMS: the Tier-1 site at PIC and the Tier-2 site at CIEMAT. The deployment and configuration of the system and the developed monitoring tools will be shown, as well as data popularity studies in relation to the optimization of the cache configuration, the effects on CPU efficiency improvements for analysis tasks, and the cost benefits and impact of including this solution in the region.

1 Introduction

The Large Hadron Collider (LHC) [1] experiments have generated since the start of operations in 2009 1 exabyte of data from simulated and experimental proton an ion collisions. This amount of data is stored in the Worldwide LHC Computing Grid (WLCG) [2], spanning 170 centers in 35 countries, classified as Tier-0, Tier-1 (13 centers), and Tier-2 (about 150 centers) facilities. The High-Luminosity Large Hadron Collider (HL-LHC) program presents a major challenge to the field of high-energy physics. Scheduled to become operational by 2029, the HL-LHC is designed to provide an integrated luminosity of experimental data 10 times larger than in the initial LHC phase, hence it will increase proton-proton collision rates to unprecedented scale. Anticipating increased data production and computational demands during the HL-LHC phase, the LHC computing experts are actively innovating to address these unprecedented compute resource needs, introducing techniques that curb resource growth while

*Corresponding author: cperez@pic.es

staying within the experiments' budgetary limits. To reduce future resource requirements and enhance large-scale data delivery, the LHC experiments have launched a common R&D program [3], aimed to reduce the overall cost of future compute and storage resources, both in terms of hardware and operations.

The LHC experiments collaborate finding innovative ways to manage experiment data [4]. In particular, to optimize cost-efficient operations and storage, a new approach is being considered: consolidating storage resources into fewer sites within the WLCG to reduce data replication and take advantage of economical large-scale deployments. Content Delivery Network (CDN) solutions, like caching systems, are considered to bring data closer to CPU-focused centers. Successful cases [5] [6] involve data federations using XCache [7], the preferred caching system for XRootD [8], used in High-Energy Physics (HEP) experiments. This novel technology has been integrated into the data management of the experiments after several efforts made by the community. Data caches expect to decrease data transfer over the wide-area network, reduce data access latency (improving CPU efficiencies), and can even allow to downsize the storage deployed in the regions.

By default, the Compact Muon Solenoid (CMS) [9] computational jobs are submitted to the sites hosting the input data, but occasionally must resort to read data remotely using the CMS XRootD federation [10]. The objective of the study presented in this report is to evaluate the CPU efficiency gains achieved in jobs executed in the Spanish CMS sites while reading input data remotely, after the introduction of an XCache service. An XCache service was deployed in 2021 at the PIC Tier-1 in Barcelona, which serves data to both PIC and the CIEMAT Tier-2 site in Madrid. Both sites are separated by ~620 km (~9 ms network latency). Dedicated studies and performance measurements have been carried out to demonstrate the usefulness of the service and to reach the best possible configuration. Additionally, these studies aim to assess if the deployment of a single cache in PIC Tier-1 serving data to the whole region is a viable solution.

The CMS experiment uses an infrastructure of XRootD redirector servers that spans all of its computing sites worldwide. The XRootD redirectors subscribe to each other in a hierarchical and redundant fashion. If the requested input data is not available at the site where the CMS job is being executed, the client will be automatically redirected to the remote storage server that can provide the data using the mentioned redirector hierarchy, and data will be served from that remote site at task execution time. Despite the enhanced data accessibility of this system, the increased read latency can have negative effects on tasks' CPU efficiency. The CMS jobs can be as well configured to use the XCache service. When data is not found in the local disk storage, the XCache fetches the data from remote sites using the XRootD federation. The jobs can be configured to use the XCache service only for specific types of data and bypass it for other data types. The XCache service allows a site or regional network to cache files frequently used by the CMS experiment, reducing data transfer over the Wide-Area Network (WAN) and decreasing access latency, hence potentially minimizing the aforementioned negative effect on CPU efficiency.

2 Deployment of XCache services in Spain

The XCache service has been deployed at PIC site, in a single storage server with 175 TiB capacity. The PIC XCache service serves data to all of the PIC compute nodes, and to half of the CIEMAT compute nodes (so we can compare the benefits of using a remote XCache for the site compared to direct remote data reads). PIC CMS jobs running at PIC or at CIEMAT are configured to use the XCache for all types of CMS data (except intermediate data products or test data). Although this is not the optimal running mode, since we want to cache popular datasets only, this configuration proved to be useful to better tune the systems at scale. In

particular, running at cache saturation allowed us to set the proper low and high usage "water marks" at the cache, set to 90% and 95%, respectively. When usage goes above the high "water mark", the XCache service deletes cached files until usage goes below the low "water mark". These levels ensure that popular data is kept at the cache, maximizing the hit rates (data re-reads). A Least Recently Used (LRU) algorithm is used to identify files that are suitable for deletion. Figure 1 shows the PIC XCache usage, and how the water marks act to keep the cached data below the maximum allowed usage.

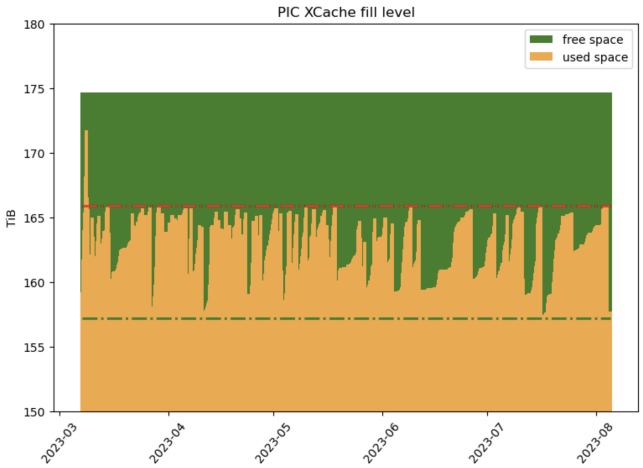

Figure 1. PIC XCache fill level over five months. The used and free space is seen, as well as the low and high water marks (green and red dashdot lines, respectively).

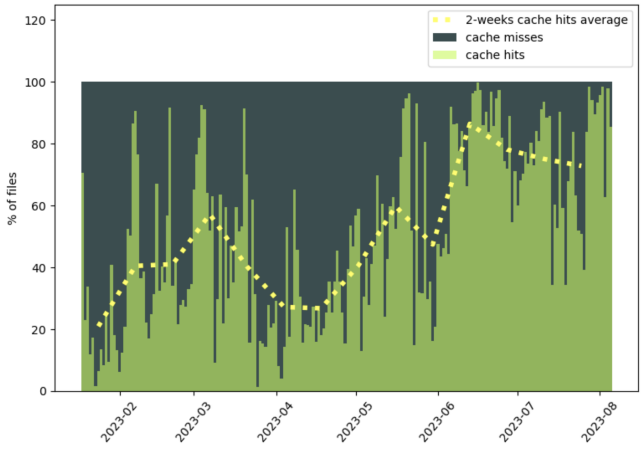

Figure 2. Percentage of cache misses and cache hits for files being accessed through the PIC XCache service, as well as the two-weeks average cache hits evolution.

The XCache at PIC serves data to 4500 CPU-cores in use by CMS at both sites. About ~5000 files are accessed daily through the XCache service (serving about 15 TB of data per day, in average). Figure 2 shows the percentage of cache misses (i.e. the file was not present

at the cache, and had to be downloaded) and cache hits (the file was already at the cache) for a several months period. Maximizing the cache hit rate is one of the subjects of these studies, which is seen to improve in time.

Monitoring tools have been developed for the XCache service, retrieving relevant information from the XCache *cinfo* files, that keep historical records of cached files access, and feeding the data to an ElasticSearch server. This data is then used to build dedicated Kibana graphs. During the period of time shown in both Figure 1 and Figure 2, the XCache was populated with files coming from several remote CMS sites, with ~65.1% of files from sites in central Europe (excluding Spain), ~15.6% from America, ~8.1% from Spain, ~7.7% from North-Europe (UK and Finland), ~2% from Russia and ~1.5% from Asia.

3 Performance of analysis jobs executed at CIEMAT

For these studies, half of the compute nodes at CIEMAT were configured to access to PIC's XCache, while the other half was kept to directly use the CMS XRootD redirectors system. In both cases, if input data files are present locally, they are directly read from the local storage. This configuration allowed us to study the effects of the caching techniques, as compared to the standard way in which the system is currently operated.

Typically users analyze popular datasets, hence we expect that the XCache system works well for analysis jobs, since the datasets reuse is expected to be much higher than the observed for datasets used for CMS centrally organized processing campaigns. We focus on jobs submitted through the CMS Remote Analysis Builder (CRAB [11]), a tool that is widely used for distributed data analysis in CMS. In the period spanning 100 days (from January to April 2023) we measured and compared the average CPU efficiency for CRAB jobs executed at CIEMAT compute nodes where XCache was enabled or disabled. The average CPU efficiency for the jobs with the XCache enabled was 77.2 ± 0.9%, while it was 70.4 ± 1.0% for jobs with the XCache disabled. The fact that the XCache serves the popular files with reduced latency improves the overall efficiency for analysis tasks at CIEMAT compute nodes with the XCache enabled. In this test, ~20% of the files were re-accessed from the PIC XCache server.

Figure 3 shows the breakdown of the average CPU efficiency of the CRAB jobs executed at CIEMAT when reading data from local, XCache at PIC or remote CMS sites. When data is read locally (T2_ES_CIEMAT) the average CPU efficiency is very high, of ~ 95%. When data is already in PIC XCache and served to CIEMAT, the observed CPU efficiency is very close to the one observed when reading from the local storage, a good measurement that indicates that a single XCache server placed in PIC Tier-1 could serve data to all of the CMS Spanish sites. The figure clearly shows how the CPU efficiency degrades when reading from remote sites. Additional to network latency, there are other factors that can affect the CPU efficiency for executed tasks, such as the load on the remote storage system, or the WAN configuration and/or load.

Taking into account the HS06·hours[1] and CPU efficiency of these jobs, we have estimated that if the access to PIC XCache would have been enabled to the whole CIEMAT compute nodes in this period, we could have saved around 13% of the total HS06·hours spent by these jobs executed at CIEMAT in this period, which indicates a potential margin of performance improvement and resources savings in the region when using the XCache service.

4 Controlled submission of analysis jobs accessing MINIAOD files

Analysis jobs submitted by users of the CRAB service are very different from each other since users might perform various types of analyses on different types of datasets. Our previous re-

[1] HEP-SPEC06 (HS06) [12] is a widely employed benchmarking CPU metric in WLCG.

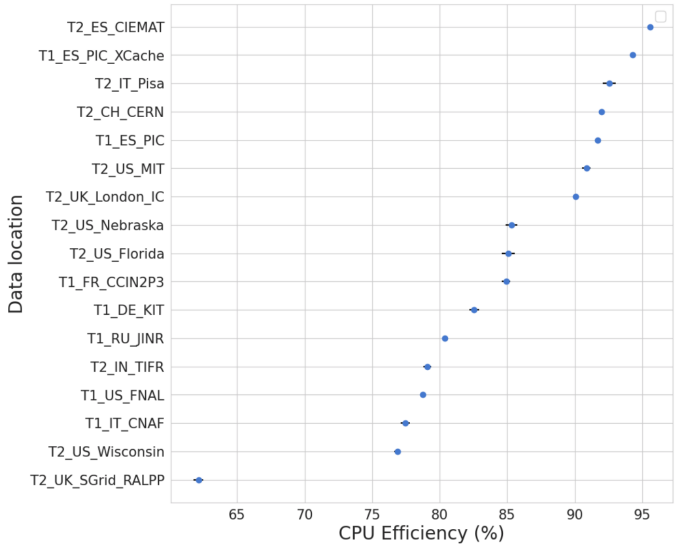

Figure 3. Average CPU efficiency of CRAB jobs executed at CIEMAT when reading data from local, XCache or remote sites, in the period covering 100 days (from January to April 2023).

search [13] demonstrated that MINIAOD files accessed by users' analysis jobs are identified as the most appropriate files to be stored in a cache. The MINIAOD data tier analysis format contains only relevant reconstructed physics object information for faster processing and quick analysis, being the most frequently used data format when performing final analyses for publications.

We conducted a similar study as the one described in the previous section, but this time executing at PIC controlled jobs that access MINIAOD files from local and remote sites. For this purpose, a benchmark analysis [14] was selected, a tag and probe analysis from the muon physics object group. For these tests, we selected similar MINIAOD input files from either local (PIC) or remote sites worldwide.

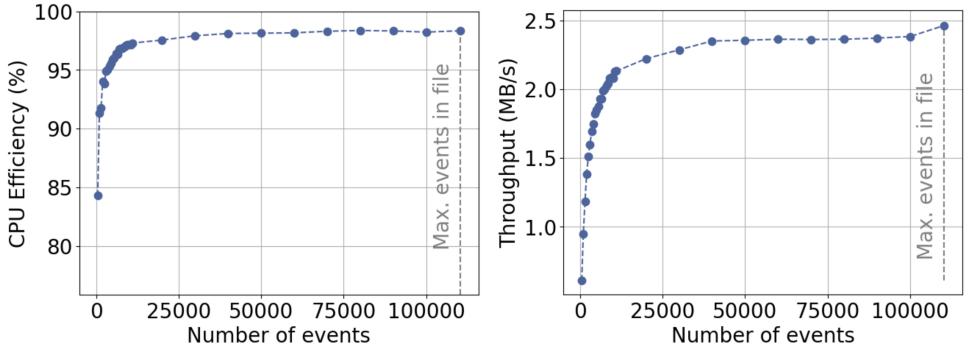

Figure 4. CPU efficiency (%) and input data throughput (MB/s) as a function of the total number of processed MINIAOD events, from tasks executed at a PIC compute node reading from local XCache.

When running the analysis benchmark jobs locally at PIC, we observed that the CPU efficiency stabilizes at around ~98% after reading around 50% of the file events, and also the file read throughput stabilizes at around 2.4 MB/s. Figure 4 shows the performance of the benchmark analysis jobs when reading the input MINIAOD file from the PIC XCache.

Hence, around 25 sites were selected to run this analysis benchmark job. An isolated PIC compute node was set up for the study, to avoid any interference with any other job present at the node. Our application took 1 CPU-core and made use of the LHCOPN [15] and LHCONE [16] networks. We executed approximately 100 jobs reading events for similar MINIAOD files from each site, starting from the first up to the 25th site, and repeating the same process 100 times. In this way, we make sure to run the same number of tests spaced in time for all of the selected sites. We then evaluated the average CPU efficiency of the analysis benchmark jobs when reading data from these remote sites. The test was executed spanning 25 days, and a total of 6.5k HS06·hours were accounted.

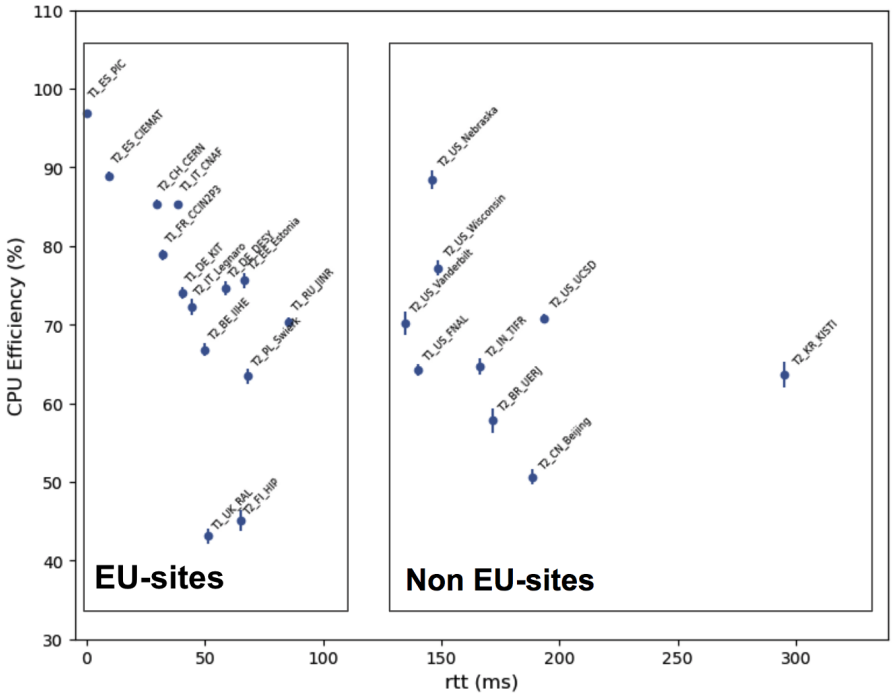

Figure 5. Average CPU efficiency of the benchmark analysis jobs executed at PIC when reading data from local, EU and non-EU sites, as a function of the site's latencies (round-trip-time [rtt], in ms).

Figure 5 shows the average CPU efficiency of the benchmark analysis jobs executed at PIC when reading data from local (labelled as T1_ES_PIC, with data stored at PIC XCache) or from remote EU and non-EU sites, as a function of latency (rtt, in ms). The graph displays a few sites that showed particularly poor or good performance despite their large rtt (T1_UK_RAL, T2_FI_HIP, and T2_US_Nebraska), but in general there is a clear trend of CPU efficiency degradation for tasks reading from remote sites. It is remarkable that these measurements were stable during the month in which the test was realized.

Reading data from France, Italy or CERN when the job is executed at PIC, results in a significant degradation on the CPU efficiency, that drops from 98% to $80-85\%$ levels. These

sites are placed at 650 km, 1,100 km and 1,000 km, respectively, and show similar rtt figures. The FNAL Tier-1 site in Chicago, USA, which is approximately 7,000 km away from PIC center (with a latency of 150 ms) is far enough to result in a significant degradation of the mean CPU efficiency of these jobs, which drops down to approximately 65%. The farthest tested site in this test was a site in South Korea (KISTI), at a distance of approximately 10,000 km from PIC. While remote access to data across the transatlantic or pacific networks is not a conventional CMS practice, we conducted the study to assess the advantages of bringing data from very distant locations closer to compute nodes.

Following the same approach as done in section 3, if all of these benchmark analysis jobs were run accessing data locally at PIC, 1.8k HS06·hours (or 28% of walltime) could have been saved out of the total 6.5k HS06·hours spent in the test. Moreover, the results obtained with the benchmark analysis jobs are quite well correlated to those results obtained for CRAB jobs executed at CIEMAT, which indicates that the selected benchmark is very representative as to the overall results obtained when averaging the performance of many user tasks, which access a wider variety of dataset types.

5 Conclusions

The results of our study provide valuable insights into the potential for improving CPU efficiency for CMS tasks using an XCache service. We have demonstrated that CMS CRAB jobs executed at CIEMAT compute nodes with enabled remote reads from PIC XCache show better performance than similar jobs executed at CIEMAT using the CMS global XRootD re-director infrastructure. The results show that the CPU efficiency for tasks executed at CIEMAT reading from their local storage, or reading from PIC's XCache (if data was already in the cache), is very similar. This indicates that a single cache placed in PIC Tier-1 could effectively serve data to all Spanish CMS Tier-2 sites without a significant impact on the application performance.

We have also made a detailed study on the CPU efficiency degradation when using an analysis benchmark job reading data from local or diverse remote sites. The results obtained are very stable in time and allow us to understand how much degradation is present when streaming data from distant sites. Moreover, the selected analysis benchmark job correlates quite well with the results seen at CIEMAT compute nodes that use the XRootD re-director infrastructure, so the job we selected is confirmed to be very representative to the degradation that real jobs might experience.

The XCache service can be used to cache and store popular data files, improving the efficiency of the executed tasks in the Spanish region. Our studies did not include CMS Tier-2 site in Santander (IFCA), and the overall benefit for the whole region is subject to a deepest evaluation. Using XCache would also allow to reduce the amount of storage resources deployed in the region. Currently, the CMS analysis datasets stored at PIC and CIEMAT account for 65% of the total storage usage of the experiment (4.6 PB). We plan to understand the scale of the XCache service in the region which would be needed, using real data access patterns, to alleviate part of the storage that holds analysis files. These ongoing efforts contribute to the understanding of the use and the benefits of the XCache service in the Spanish region, and ultimately within the CMS computing infrastructure.

Acknowledgements

This project is partially financed by the Spanish Ministry of Science and Innovation (MINECO) through grants FPA2016-80994-C2-1-R, PID2019-110942RB-C22, DATA-2020-1-0039, and BES-2017-082665.

It has also been supported by the Ministerio de Ciencia e Innovación (MCIN) AEI/10.13039/501100011033 under contract PID2020-113614RB-C21, and the Catalan government under contract 2021 SGR 00574. The deployment of the XCache service is financed by the Red Española de Supercomputación (RES) through the grant DATA-2020-1-0039.

References

[1] *The large hadron collider:*, https://home.cern/science/accelerators/large-hadron-collider (2023), accessed: 2023-07-31

[2] *Worldwide LHC Computing Grid*, https://wlcg-public.web.cern.ch/ (2023), accessed: 2023-07-31

[3] J. Albrecht, A.A. Alves, G. Amadio, G. Andronico, N. Anh-Ky, L. Aphecetche, J. Apostolakis, M. Asai, L. Atzori, M. Babik et al., Computing and software for big science **3**, 1 (2019)

[4] X. Espinal, S. Jezequel, M. Schulz, A. Sciabà, I. Vukotic, F. Wuerthwein, *The Quest to solve the HL-LHC data access puzzle*, in *EPJ Web of Conferences* (EDP Sciences, 2020), Vol. 245, p. 04027

[5] D. Ciangottini, G. Bagliesi, M. Biasotto, T. Boccali, D. Cesini, G. Donvito, A. Falabella, E. Mazzoni, S. D., M. Tracolli, *Integration of the Italian cache federation within the CMS computing model*, in *Proceedings of Science* (PoS, 2020), Vol. ISGC2019, p. 014

[6] E. Fajardo, M. Tadel, J. Balcas, A. Tadel, F. Würthwein, D. Davila, J. Guiang, I. Sfiligoi, *Moving the California distributed CMS XCache from bare metal into containers using Kubernetes*, in *EPJ Web of Conferences* (EDP Sciences, 2020), Vol. 245, p. 04042

[7] L. Bauerdick, K. Bloom, B. Bockelman, D. Bradley, S. Dasu, J. Dost, I. Sfiligoi, A. Tadel, M. Tadel, F. Wuerthwein et al., *XRootd, disk-based, caching proxy for optimization of data access, data placement and data replication*, in *Journal of Physics: Conference Series* (IOP Publishing, 2014), Vol. 513, p. 042044

[8] A. Dorigo, P. Elmer, F. Furano, A. Hanushevsky, WSEAS Transactions on Computers **1**, 348 (2005)

[9] *Cms experiment*, https://home.cern/science/experiments/cms (2023), accessed: 2023-07-31

[10] L. Bauerdick, D. Benjamin, K. Bloom, B. Bockelman, D. Bradley, S. Dasu, M. Ernst, R. Gardner, A. Hanushevsky, H. Ito et al., Journal of Physics: Conference Series **396**, 042009 (2012)

[11] M. Mascheroni, et al, *CMS distributed data analysis with CRAB3*, in *Journal of Physics: Conference Series* (IOP Publishing, 2015), Vol. 664, p. 062038

[12] A. Valassi, M. Alef, J. Barbet, O. Datskova, R. De Maria, M. Fontes Medeiros, D. Giordano, C. Grigoras, C. Hollowell, M. Javurkova et al., *Using HEP experiment workflows for the benchmarking and accounting of WLCG computing resources*, in *EPJ Web of Conferences* (EDP Sciences, 2020), Vol. 245, p. 07035

[13] C. Pérez Dengra, J. Flix Molina, A. Sikora, on behalf of the CMS Collaboration, *New storage and data access solution for CMS experiment in Spain towards HL-LHC era*, in *Journal of Physics: Conference Series* (IOP Publishing, 2023), Vol. 2438, p. 012053

[14] *Ramírez sánchez, gabriel. "muonanalysis-muonanalyzer", gitlab, version 1.0, 2023*, https://gitlab.cern.ch/garamire/muonanalysis-muonanalyzer/-/tree/master/ (2023), accessed: 2023-07-31

[15] *The lhcopn*, https://lhcopn.web.cern.ch/ (2023), accessed: 2023-07-31

[16] *The lhcone*, https://lhcone.web.cern.ch/ (2023), accessed: 2023-07-31