EPJ Web of Conferences **245**, 09012 (2020) https://doi.org/10.1051/epjconf/202024509012
CHEP 2019

CMS strategy for HPC resource exploitation

Antonio Pérez-Calero Yzquierdo[1,2,*] on behalf of the CMS Collaboration

[1]Centro de Investigaciones Energéticas, Tecnológicas y Medioambientales (CIEMAT), Madrid, Spain
[2]Port d'Informació Científica (PIC), Barcelona, Spain

Abstract. High Energy Physics (HEP) experiments will enter a new era with the start of the HL-LHC program, with computing needs surpassing by large factors the current capacities. Anticipating such scenario, funding agencies from participating countries are encouraging the experimental collaborations to consider the rapidly developing High Performance Computing (HPC) international infrastructures to satisfy at least a fraction of the foreseen HEP processing demands. These HPC systems are highly non-standard facilities, custom-built for use cases largely different from HEP demands, namely the processing of particle collisions (real or simulated) which can be analyzed individually without correlation. The access and utilization of these systems by HEP experiments will not be trivial, given the diversity of configuration and requirements for access among HPC centers, increasing the level of complexity from the HEP experiment integration and operations perspectives. Additionally, while HEP data is residing on a distributed highly-interconnected storage infrastructure, HPC systems are in general not meant for accessing large data volumes residing outside the facility. Finally, the allocation policies to these resources are generally different from the current usage of pledged resources deployed at supporting Grid sites. This report covers the CMS strategy developed to make effective use of HPC resources, involving a closer collaboration between CMS and HPC centers in order to further understand and subsequently overcome the present obstacles. Progress in the necessary technical and operational adaptations being made in CMS computing is described.

1 Motivation for the use of HPC resources

High Energy Physics (HEP) experiments, such as CMS, are aiming towards increasing the usage of High Performance Computing (HPC) resources to help cover the expected increase in computing resources needs in the mid to long term future (Run3 and HL-LHC)[1], while coping with the projected continuation of current levels of funding.

In the current international landscape for ever larger scientific projects and bigger scientific computing installations, growing funds are being committed to HPC centers, whose managers are looking onwards to reaching the Exascale for their infrastructures (see, for example [2] and [3]). As a consequence of this strategic policy, funding agencies (FAs) from countries participating in the LHC program are encouraging their national communities to make use of such resources in order to cover, at least partially, their needs for computing

*e-mail: aperez@pic.es

power demands. In addition to the funding implications, policymakers also see the participation of HEP experiments in HPC infrastructures as an opportunity to get HPC managers experienced in providing support in the near future for other data-intensive scientific studies.

LHC experiments have taken notice of such trends, which present the opportunity to help cover their growing computing demands while also gaining access to the best technologies available in the market, usually employed at HPC sites. The current, and specially future, contribution from HPC sites is already regarded as an integral part of WLCG [4] strategy towards HL-LHC (see, for example, figure 1, from [5]). However, CMS, while committed to make the best posible use of non-Grid opportunistic CPUs, including the HPC ones, is at this point not ready to transition form traditional computing resources at WLCG sites into HPCs. The following sections describe the necessary technical and operational adaptations in CMS computing for optimal HPC resource exploitation.

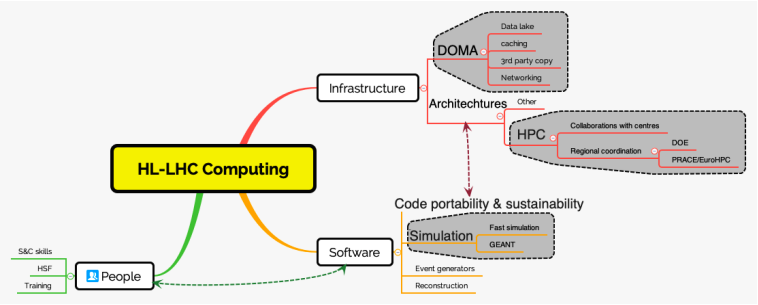

Figure 1. HPC resources as part of the WLCG strategic plans towards HL-LHC computing model, from [5]

2 Integration efforts and access to HPC resources by CMS

CMS strategy to approach HPC centers is based on the national and local CMS teams doing the handshaking with HPC representatives from their respective nations or regions. In order to support this plan, a technical document addressing the CMS requirements to successfully exploit HPC systems has been produced [6], along with an executive summary mainly addressed to FA officials [7], in line with similar recommendations common to LHC experiments as compiled by WLCG [8].

The key element to consider when discussing HPC integration is that each HPC is different, which generates the need to reach agreements on technical and policy questions in each case. This is perceived as a potential cause for big integration efforts by CMS. Technically, the deployment of required services such as singularity, CVMFS [9], data cache services, job gateway (computing elements, or CEs) and storage access for input/output data files has to be negotiated. When some of the technical constraints are not met, ad hoc solutions need to be explored. For example, if no CVMFS or run-time access to conditions data by network is available, CMS would need to prepare appropriate images in order to run a containerized version of the CMS applications.

In terms of policy, the major concern is the need of a consensus in terms of resource availability. In many cases, the standard procedure for HPC resource access is to participate in open calls for resource allocation grants at every specific HPC center. Moreover, often, allocations are granted for specific physics studies, not as generic resource for all experiment's needs. These conditions are in general not suitable for HEP experiments, including CMS. Instead, a stable multi-year term plan specifying a certain share of resources at HPCs would be needed to properly plan CMS computing operations in advance.

When discussing the use of HPCs as *pledged* resources with FAs, CMS and the rest of LHC experiments should also be explicit in protecting the traditional agreements on High Throughput Computing (HTC) resources at sites integrated into the WLCG infrastructure. WLCG sites are key contributors to the LHC computing success, beyond just CPU power. For example, staff from WLCG sites dedicate efforts to the deployment and operations of data storage and management capacities (which in turn generates local technical expertise), and participate in multiple projects of middleware development as well as continuous improvement to WLCG metrics monitoring and resource utilization accounting.

2.1 Software development

A sizable fraction of the total computing power that HPCs will amount to is expected to be provided by processor types other than the standard Intel x86_64 architecture CPUs, which CMS has successfully exploited from the Grid sites supporting the experiment. Indeed, CMS software is mostly written and compiled for such type of CPUs, as these have been the best choice for affordable computing in the last decade. Changes at the software level are therefore required if CMS is to profit from this share of the HPC computing power, which would not accessible to CMS otherwise.

Efforts within CMS are already being organized to address this challenge, with CMS software evolving towards being capable to profit from heterogeneous resources. Increasing work is being done on enabling support for heterogeneity in the CMSSW framework [10]. along with a revision of the reconstruction algorithms and data structures [11][12][13]. The use of performance portability libraries aims at running single implementations of algorithms on diverse types of accelerators, as well as on CPUs, therefore several candidates are being explored [14][15]. Moreover, CMS is investigating the opportunity to test a heterogeneous solution at the HLT farm for Run3 [16].

2.2 Operational challenges

The CMS team in charge of operations and workload planning aims at achieving a transparent integration of HPC resources, with respect to the assignment of tasks to either HTC or HPC sites. An automatized selection algorithm that identifies workflows suitable to be executed in a given HPC is required in order to achieve a sustainable and stable exploitation of these centers. In order to accomplish that goal, accurate characterization of both resources properties and workflows requirements becomes critical. The capabilities of the resources must be known to our workload management system (WMS), for example in relation to the network connectivity of the execution nodes, the availability of input data at (or near) the computing facility with local or any remote I/O throughput constraints. Considering the general situation of HPC resource capabilities in relation to input data accessibility, Monte-Carlo (MC) simulation jobs are considered the easiest workflow to run. However, a more ambitious approach including all stages of a simulated data sample could be pursued as well, moving to a single workflow type that includes all stages GEN-SIM, then followed by DIGI-RECO.

Another difficult aspect operationally is that of deciding a workload splitting into jobs that can satisfy HPC requirements, for example maximum execution time per job. Measuring average time per processed event, in order to estimate total job running time, is already a challenge, considering the diversity of CPU types in the Grid. This difficulty in determining job execution time will only grow with the enlarged range of processor types, different than the standard found on Grid sites, in use in present and future HPCs (e.g. Xeon Phi, coprocessors, etc).

2.3 Data Management

The main concern regarding data management when trying to run CMS jobs on HPC systems is that those are generally not meant for accessing massive amounts of data residing outside the facility. In general HPC centers are not designed prioritizing bulk storage size and high bandwidth WAN connectivity so that they can join a distributed infrastructure and optimize overall processing throughput. In order to make access to storage available for input and output datasets, a twofold approach has been conceived. A first strategy will involve the use of non-local I/O via xrootd data transfers from and to "nearby" WLCG sites, in cases that sufficiently performant WAN capacity is available. Alternatively, when this strategy is not suitable but the HPC site provides a storage endpoint, accessible via any of the standard protocols employed in the Grid (http, gridFTP, xrootd) and with enough local storage, a mechanism for data prestaging could be implemented using the data management CMS tools (presently PhEDEx, soon to be replaced by Rucio [17]). Output datasets would be written to local disks, from which they would require to be transferred to a final CMS-managed storage endpoint, from which it can be inserted into the CMS data catalogues.

2.4 Resource provisioning

The CMS Submission Infrastructure [18] (SI) is in charge of allocating compute resources as well as scheduling tasks. Current CMS SI comprises multiple HTCondor [19] pools operating as a federation, including multiple workload submission nodes that can overflow tasks from one pool to the others. The integration of HPC resources into the SI can be achieved in a number of ways (see figure 2). For instance, following the main route that allocates resources at the grid sites, some HPC centers have setup grid middleware such as CEs, so their CPUs are made accessible via the submission of GlideinWMS [20] pilot jobs that spawn HTCondor execution nodes that join the CMS SI global HTCondor pool (the case of the CSCS Swiss site, for example). Alternatively, the HTCondor execution nodes can be launched at the HPC independently of the CMS SI route, by employing a *vacuum* model [21], which can be implemented for example, employing a DODAS [22] approach. The access for CMS workloads to HPC resources can also be granted by means of incorporating the HPC slots into externally-managed HTCondor pools, which are then federated with the centralized CMS infrastructure. This is the case of the HPC fraction of resources included in the HEPCloud maintained from FNAL [23], and that of the prototype being worked on in order to enable job flocking from CMS workload schedulers on to the BSC nodes via PIC [24].

2.5 Network requirements

This section summarizes the crucial requirements of network connectivity by CPU resources so that they can be employed by CMS, as stressed in [6] and [8]. Firstly, CMS SI relies mainly on the submission of pilot jobs to sites, which require WAN connectivity in order to interact with the SI nodes from which they retrieve their configuration and validation scripts, used to setup and spawn HTCondor execution nodes (*startd*) for CMS. A late-binding model is employed for workload scheduling, which implies, as described in figure 2, that the startds need to connect to the central node of the pool in order to be assigned workloads to execute (payload jobs). The execution of these payload tasks requires software packages (such as CMS simulation, reconstruction and analysis software, along with appropriate container images) to be distributed at the job slots, which are synchronized from a central repository via CVMFS. Moreover, payload jobs need access to event processing conditions, provided by distributed database replicas that are accessed from squid cache servers (using port 80).

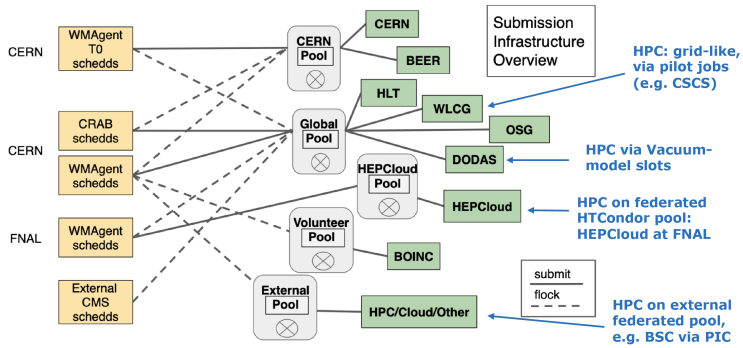

Figure 2. Schema of the CMS SI, modified from [25], indicating the mechanisms that allow the allocation and exploitation of resources at HPC sites. On the left-hand side, a number of redundant and geographically distributed WM submission nodes (HTCondor *schedds*) manage the diverse CMS workload types. These nodes, can be assigned resources from a number of HTCondor resource pools, each managed by a central node running *collector* and *negotiator* services that produce the matchmaking to resources belonging to the same pool, or by allowing overflow of workload to a neighboring federated pool (*flocking*). Diverse CPU resources, summarized on the right-hand side of the diagram, include WLCG and CERN job execution slots, but also opportunistic offline usage of the CMS HLT farm, among others. The integration of HPC resources into this flexible infrastructure can be achieved by a number of technical ways, as described in section 2.4

Finally, input and output datasets need to be accessed and transferred to storage, in many cases via WAN, as described in 2.3.

3 Ongoing HPC integration examples

A first example of the successful exploitation of HPC by CMS is the use of NERSC [26] resources in the USA. Provisioning of slots at NERSC clusters into the HEPCloud pool is performed by means of the Decision Engine [27], a key element in the HEPCloud design which triggers resource allocation on diverse cluster, grid or HPC, as well as commercial Cloud providers, based on the queued job properties, the instantaneous state of the HEPCloud pool and a set of policies defined from FNAL as resource managers. Figure 3 describes the HEPCloud architecture, including the role of the Decision Engine, while also providing an example time period with 55,000 CPU cores at NERSC running CMS production workflows.

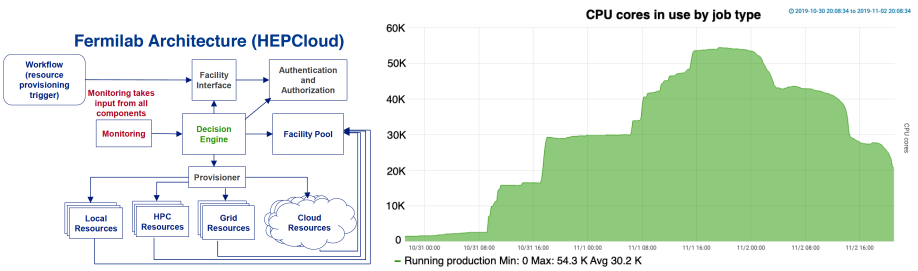

Figure 3. HPC resources integrated in the HEPCloud infrastructure schema (left) and NERSC CPUs recently in use running CMS simulation jobs (right)

Secondly, it is worth mentioning the integration of the Marconi cluster, at CINECA [28], in Italy. In this case, the efforts of the local CMS community, together with CNAF and

CINECA teams, lead to the adoption of standard grid technologies in order to enable the use of CMS pilots to access the HPC cluster [29]. A new feature has been introduced in these pilots which makes them configurable at the destination endpoint, in order to allow the local team to filter unsuited workloads (based for example of requested execution time) from the matchmaking process for these resources.

Finally, the efforts towards the integration of the Barcelona Supercomputing Center (BSC), the main HPC site in Spain, can be summarized in this report to highlight the difficulties present in a network-restrictive scenario such as the BSC, comparable to other cases in discussion in other countries. Their biggest general-purpose cluster at BSC, the MareNostrum 4 (MN4, with over 150.000 processors, with a peak power rated at 11.15 Petaflops [30]), has been selected in 2019 for the deployment of a new near-Exascale cluster (MN5, designed for 200 Petaflops by 2021), making it a resource to be considered in future LHC computing planning in Spain. The BSC presents however a particularly hard environment for CMS processing integration, as the compute nodes do not have open ports that allow essential connectivity. The login machines only allow ssh incoming connection, and there is no support for edge services at the site (e.g. Squids and CVMFS). Disk is instead accessible, with a shared GPFS area mounted on all of the BSC compute nodes, as well as on the login machines and externally by sshfs. Considering these strict circumstances, the Spanish CMS team at PIC and CIEMAT have joined the HTCondor development team on a project to enable access to BSC network-isolated nodes for CMS, using the file system (FS) for the HTCondor control signal path [24]. The main elements of the working model under discussion are described in figure 4.

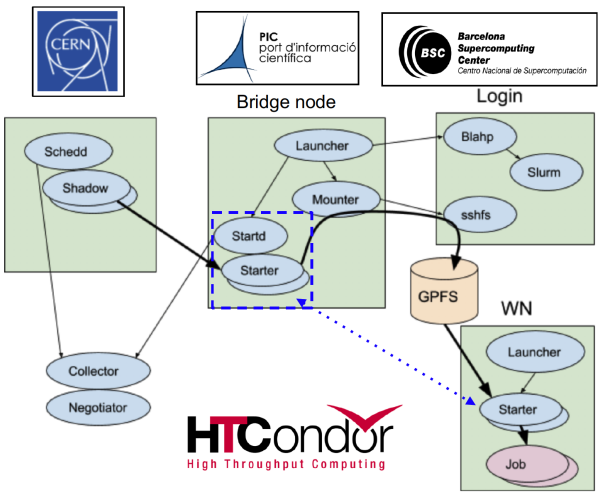

Figure 4. HPC resources at BSC integration model into CMS computing using communication via FS to replace WAN. A *bridge* node installed at PIC submits Slurm jobs to BSC queues, that, upon execution, instantiate HTCondor starters on the BSC WN. These starter processes update their status to the mirror starter processes running at PIC, at the other side of the network barrier. In order to enable this mirroring, job input sandboxes and status signals are regularly synchronized between the bridge to the WN at BSC, passing .tar files via gpfs. The startd process at PIC opens to matchmaking by CMS job schedds at CERN, passing payload jobs to the starter, and via FS synchronization, to the BSC WN. From a functional perspective, in this model, the node at BSC has joined the HTCondor pool at PIC, federated to CMS Global Pool.

4 Conclusions

CMS is dedicating increasing efforts to the progressive integration of HPC resources, given the important role they could play in the computing budget for future stages of the LHC program. CMS strategy relies on local CMS teams establishing agreements with their national HPC centers, while providing useful guidelines on the technical details for a successful integration. Software development would be needed to fully exploit HPCs employing GPUs, a trend many future HPC clusters are expected to follow. In order for CMS to achieving a sustainable way of exploiting HPCs, integration work must be done to ensure transparent operations, compared to the current model dominated by WLCG resources. A key element required for this will be an enhanced description of resources and workflows that allows for further automatization of the matchmaking, minimizing human intervention in workload scheduling. In terms of access to data, a dual strategy could be followed, with input datasets being pre-placed at the HPCs or alternatively streamed from nearby storage via WAN connectivity. Finally, HPC resource allocation into CMS SI can be achieved via a number of technical routes, some of which are already working, others being in testing or prototyping phases. Even with connectivity restrictions on the HPC side, technical solutions are being worked on to overcome the network barrier.

This work was partially supported by the Spain's Ministry of Economy and Competitiveness grant FPA2016-80994. CMS thanks our partners in the GlideinWMS and HTCondor development teams, the OSG, and our colleagues at CERN and Fermilab, all of whom make the shared computing infrastructure a success.

References

[1] HEP Software Foundation. "A Roadmap for HEP Software and Computing R&D for the 2020s", HSF-CWP-2017-01, arXiv:1712.06982 physics.comp-ph (2017).

[2] Parnertship for Advanced Computing in Europe, http://www.prace-ri.eu.

[3] Exascale Computing Project, https://exascaleproject.org.

[4] The Worldwide LHC Computing Grid http://wlcg.web.cern.ch.

[5] I. Bird. "WLCG preparations for Run 3 and beyond", 7th Scientific Computing Forum (2019) https://indico.cern.ch/event/851050/contributions/3578170/.

[6] CMS Offline, Software and Computing, HPC resources integration at CMS, CMS-NOTE-2020-002 ; CERN-CMS-NOTE-2020-002.

[7] CMS Offline, Software and Computing, A closer collaboration between HEP Experiments and HPC centers, CMS-NOTE-2020-003 ; CERN-CMS-NOTE-2020-003.

[8] M. Girone, "Common challenges for HPC integration into LHC computing", WLCG-MB-2019-01, http://wlcg-docs.web.cern.ch/wlcg-docs/technical_documents/HPC-WLCG-V2-2.pdf (2019).

[9] The CernVM File System, https://cernvm.cern.ch/portal/filesystem.

[10] O. Gutsche et al. "Bringing heterogeneity to the CMS software framework", to be published in these proceedings.

[11] A. Bocci et al. "Heterogeneous reconstruction: combining an ARM processor with a GPU", to be published in these proceedings.

[12] Z. Chen et al. "GPU-based Offline Clustering Algorithm for the CMS High Granularity Calorimeter", to be published in these proceedings.

[13] A. Bocci et al. The CMS Patatrack Project. United States: N. p., 2019. Web. doi:10.2172/1570206, FERMILAB-SLIDES-19-010-CD.

[14] H. Carter Edwards et al. “Kokkos: Enabling manycore performance portability through polymorphic memory access patterns”, Journal of Parallel and Distributed Computing, Volume 74, Issue 12 (2014).

[15] E. Zenker et al. “Alpaka – An Abstraction Library for Parallel Kernel Acceleration”, 2016 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW), Chicago, IL, (2016).

[16] A. Bocci et al, “Heterogeneous online reconstruction at CMS”, to be published in these proceedings.

[17] RUCIO project, https://rucio.cern.ch.

[18] J. Balcas et al. “Using the glideinWMS System as a Common Resource Provisioning Layer in CMS”, J. Phys.: Conf. Ser. **664** 062031 (2015).

[19] HTCondor public web site, https://research.cs.wisc.edu/htcondor/index.html.

[20] The Glidein-based Workflow Management System, https://glideinwms.fnal.gov/doc.prd/index.html.

[21] A. McNab et al. “Running Jobs in the Vacuum”, J. Phys.: Conf. Ser. 513 032065 (2014).

[22] D. Spiga et al. “Exploiting private and commercial clouds to generate on-demand CMS computing facilities with DODAS”, EPJ Web of Conferences. **214**. 07027 (2019).

[23] S. Timm et al. “Virtual machine provisioning, code management, and data movement design for the Fermilab HEPCloud Facility”, J. Phys.: Conf. Ser. **898** 052041 (2017).

[24] J. Flix et al, “Exploiting network restricted compute resources with HTCondor: a CMS experiment experience”, to be published in these proceedings.

[25] A. Pérez-Calero Yzquierdo et al. “Evolution of the CMS Global Submission Infrastructure for the HL-LHC Era”, to be published in these proceedings.

[26] National Energy Research Scientific Computing Center (NERSC), https://www.nersc.gov/about/

[27] A. Tiradani et al. “Fermilab HEPCloud Facility Decision Engine Design”, FERMILAB-TM-2654-CD, CS-doc-6000 (2017).

[28] CINECA consortium, https://www.cineca.it/en/hpc.

[29] T. Boccali, et al. “ Extension of the INFN Tier-1 on a HPC system”, to be published in these proceedings.

[30] MareNostrum 4 system architecture, https://www.bsc.es/marenostrum/marenostrum/technical-information.