Pushing HTCondor and glideinWMS to 200K+ Jobs in a Global Pool for CMS before Run 2

FERMILAB-CONF-15-604-CD

Pushing HTCondor and glideinWMS to 200K+ Jobs in a Global Pool for CMS before Run 2

J Balcas[1], S Belforte[2], B Bockelman[3], O Gütsche[4], F Khan[5], K Larson[4], J Letts[6], M Mascheroni[7], D Mason[4], A McCrea[6], M Saiz-Santos[6], I Sfiligoi[6], on behalf of the CMS Collaboration

[1]Vilnius University, Vilnius, Lithuania
[2]Università di Trieste, INFN Sezione di Trieste, Trieste, Italy
[3]University of Nebraska - Lincoln, USA
[4]Fermi National Accelerator Laboratory, Batavia, USA
[5]National Centre for Physics, Quaid-I-Azam University, Islamabad, Pakistan
[6]University of California, San Diego, La Jolla, USA
[7]INFN Sezione di Milano-Bicocca, Università di Milano-Bicocca, Milano, Italy

E-mail: jletts@ucsd.edu

Abstract. The CMS experiment at the LHC relies on HTCondor and glideinWMS as its primary batch and pilot-based Grid provisioning system. So far we have been running several independent resource pools, but we are working on unifying them all to reduce the operational load and more effectively share resources between various activities in CMS. The major challenge of this unification activity is scale. The combined pool size is expected to reach 200K job slots, which is significantly bigger than any other multi-user HTCondor based system currently in production. To get there we have studied scaling limitations in our existing pools, the biggest of which tops out at about 70K slots, providing valuable feedback to the development communities, who have responded by delivering improvements which have helped us reach higher and higher scales with more stability. We have also worked on improving the organization and support model for this critical service during Run 2 of the LHC. This contribution will present the results of the scale testing and experiences from the first months of running the Global Pool.

1. Introduction to the CMS Global Pool

GlideinWMS [1] together with HTCondor [2] form basis of the main resource provisioning system of the CMS experiment [3] at the LHC [4]. As shown in figure 1, the main components are a HTCondor central manager, various submission nodes which hold the batch queues (HTCondor schedd), and execute nodes (HTCondor startd) which run on various Grid resources. These execute nodes are submitted by glideinWMS factories upon request by a CMS glideinWMS frontend, which examines the job queues and asks for the appropriate matching resources.

For the past several years we have operated independent HTCondor pools, one for analysis and another for central data processing and Monte Carlo production [5]. The initial motivation [6] for unifying the various pools in CMS into a single “Global Pool” [7] was to be able to rapidly prioritize between different kinds of workflows, e.g. high vs. low priority Monte Carlo production, or to boost reprocessing or a high-stakes analysis. An example of this is shown in figure 2, in which a high-priority Monte Carlo workflow was demonstrated to take over quickly a large share of the total

Published under licence by IOP Publishing Ltd

Operated by Fermi Research Alliance, LLC under Contract No. De-AC02-07CH11359 with the United States Department of Energy.

resources available to the Global Pool, crowding out lower-priority Monte Carlo workflows. The share dedicated to physics analysis was largely unaffected, by design, since analysis and non-analysis activities generally each have a 50% share of the resources dedicated to the Global Pool. An exception is at the Tier-1 sites, 95% of whose resources are dedicated to non-analysis activities. This different fair share is enforced at the site level still, since we have not yet attempted to configure the Global Pool to have resource-dependent fair share.

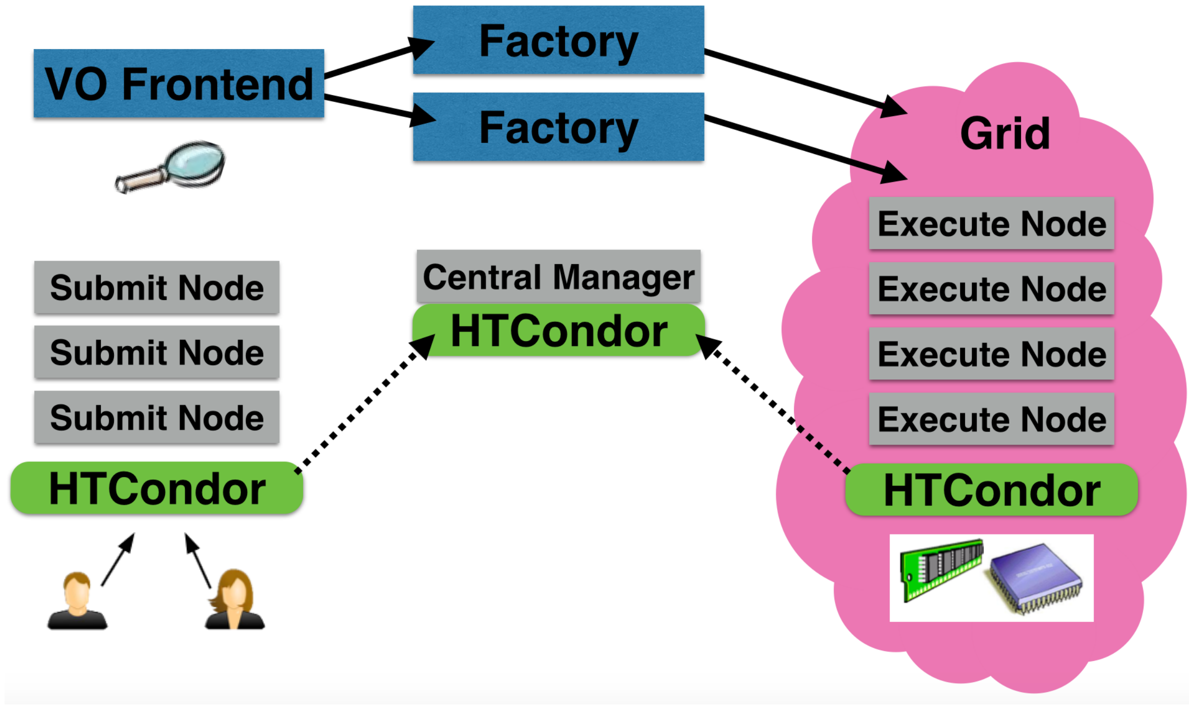

Figure 1. Architecture of a glideinWMS pilot submission system to a HTCondor pool. The main elements of glideinWMS are factories which submit light-weight pilots to grid, and now cloud, sites, and a glideinWMS frontend which requests the pilots based on need for resources in the underlying HTCondor pool. The HTCondor pool itself consists of scheduler (submit) nodes, daemons which run on execute nodes, and a central manager which negotiates matches between queued jobs and resources.

Additional motivations for establishing a unified submission infrastructure were to reduce operational load and the ability to bring new and different types of resources into a Global Pool. The main challenge, however, is that a glideinWMS or HTCondor pool on the scale of the resources available to CMS had never been attempted before.

2. The Scale Challenge

Currently the WLCG resources pledged to CMS by the sites accessible to the Global Pool is about 108,000 batch slots. Using the Global Pool infrastructure, however, we can discover over time the totality of the resources available to the pool, including opportunistic resources reachable through the regular Grid architecture, which we currently estimate to be about 200,000 batch cores, as shown in figure 3. Every day one can examine the HTCondor log files and record the unique machine names for each glidein pilot, and number of CPU cores each machine has. Over time, as seen in figure 3, the Global Pool will hit more and more of the resources, reaching finally a maximum of about 200,000 CPU cores.

On some of these resources CMS must compete for access opportunisticly, but this estimate gives an idea of the scale necessary for the Global Pool to reach during Run 2, also taking into account that the resources requested (and CMS needs) will grow from year to year. This is significantly bigger than any other multi-user HTCondor based system currently in production.

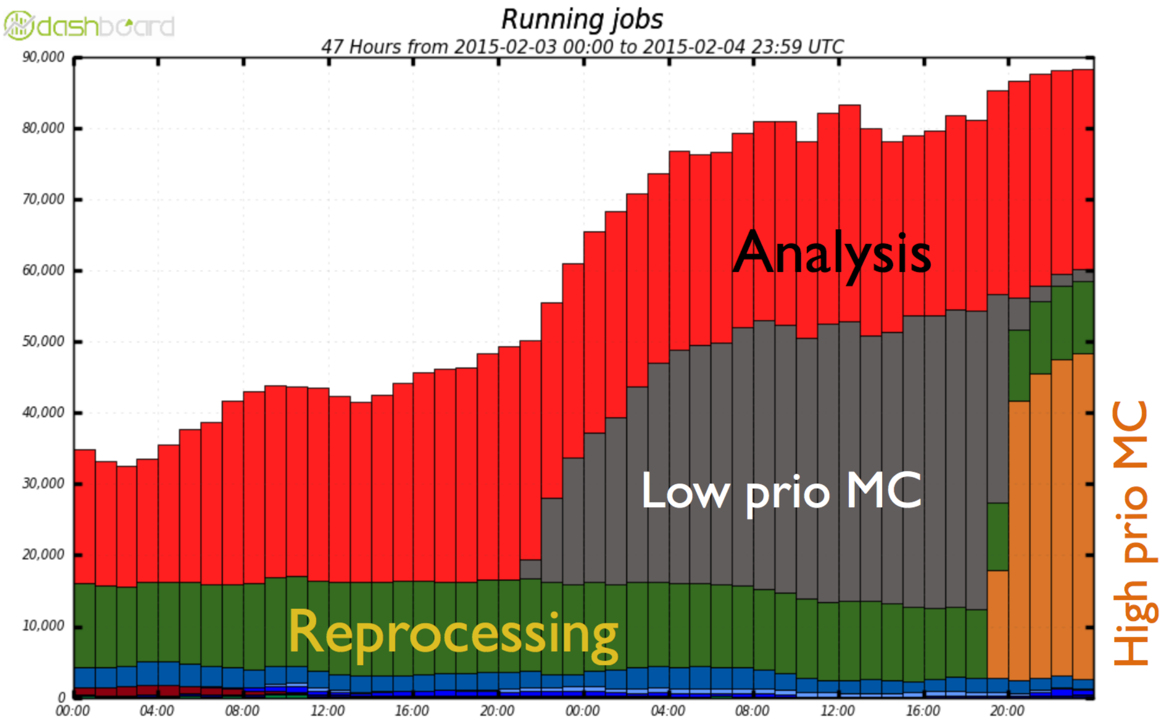

Figure 2. Demonstration that a high-priority Monte Carlo workflow (orange) can quickly take over a large share of the total resources available to the Global Pool, taking share away from lower-priority Monte Carlo workflows (grey). The share dedicated to physics analysis (red) is largely unaffected by design, since analysis and non-analysis activities generally each have a 50% share of the resources dedicated to the Global Pool.

3. Inclusion of New Cloud and Opportunistic Resources

CMS has further unified the resource provisioning system by including new types and combinations of facilities and workflows that we did not have during Run 1, such as using the HLT (High Level Trigger) farm during LHC inter-fills [8], and running the Tier-0 on Cloud resources as part of the glideinWMS system [9].

This expansion, however, further increases the scale at which the Global Pool must operate. In order to mitigate the risk during Run 2 that any scaling limitations we might encounter do not impact data taking, we opted to run the Tier-0 as an independent yet highly similar pool which can "flock" extra jobs to the Global Pool when needed. CMS also has won some initial allocations on HPC clusters such as Gordon at the SDSC, to which we can submit workflows with glideinWMS and which we would like to include in the Global Pool as well [10].

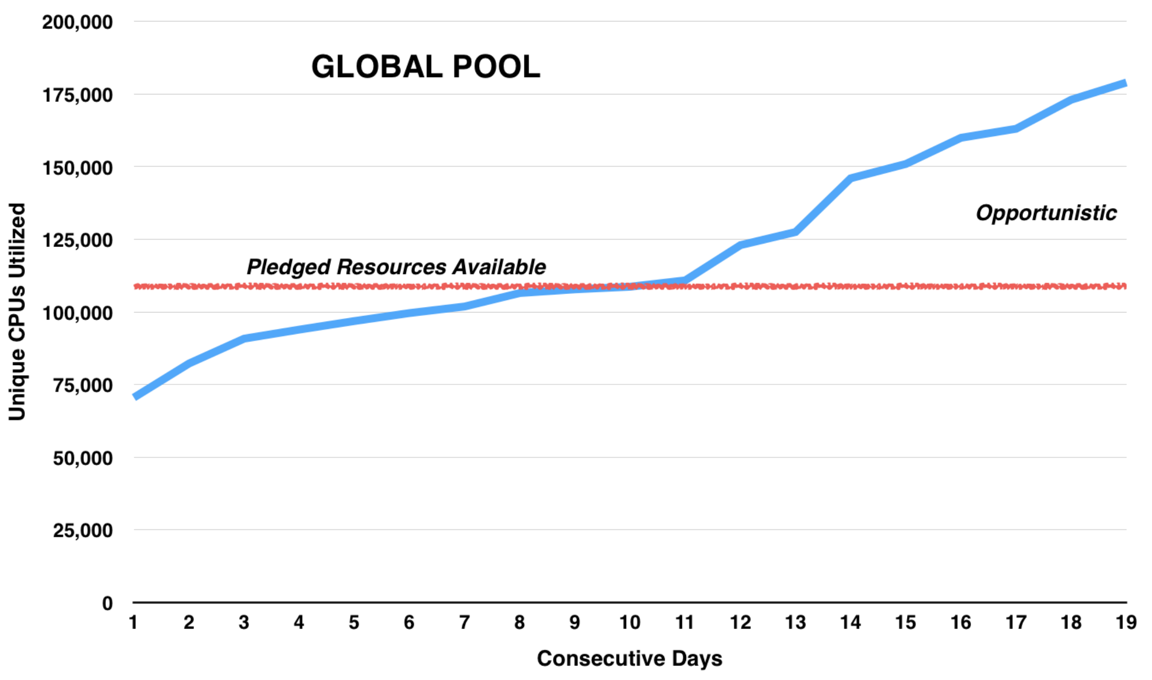

Figure 3. Batch cores discovered by the Global Pool over nearly three weeks, as compared to the WLCG pledged resources available to the pool.

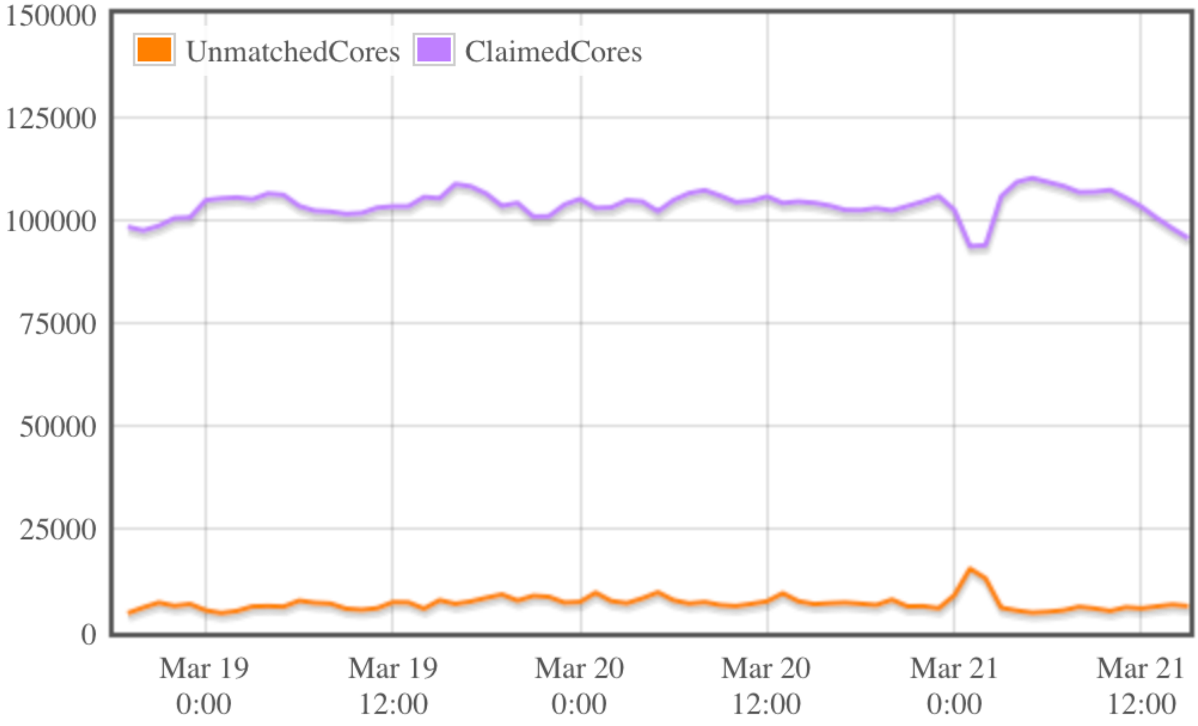

Figure 4. Demonstration of stable running of the Global Pool at scales of the WLCG pledged resources available to the pool, approximately 108,000 CPU cores.

4. Scale Tests and Feedback

During 2014 we worked closely with both the HTCondor and glideinWMS development teams and the OSG to find and fix problems that might limit the scalability of the system. In particular, the scale testing performed by the OSG [11] using CMS resources and CMS's own scale testing were both invaluable to identifying improvements that could be made in the communication between the various HTCondor components, in the Negotiator cycle, scheduler stability, etc.

In particular, the OSG scale tests have demonstrated that stable operation of a HTCondor pool is possible at a scale of 200,000 parallel running jobs. However, there are several specialized tunings of the HTCondor system that need to be made, which we have adopted in CMS, which are detailed in [11]. In addition, a component of the pool called the Condor Connection Broker (CCB) needs to be separated out onto its own hardware in the glideinWMS system. We have communicated this need as a high priority for CMS to the glideinWMS developers. The close cooperation we have with both the HTCondor and glideinWMS development communities is invaluable.

5. Support Model

The consolidation of glideinWMS operations in CMS into a single Global Pool has achieved significant economies of effort. To support the unified submission infrastructure, CMS has a written support model document. The key elements of the support model are redundancy, testing and integration, and close cooperation with the middleware developers.

Furthermore, we take full advantage of the "High Availability" (HA) mode of glideinWMS to locate critical services in multiple availability zones. When one critical service (such as the central manager) goes down, another machine can take over the functionality in a seamless way. As shown in figure 6, most glideinWMS and HTCondor services, such as the Central Maanger, and soon the glideinWMS frontend, are run in "high-availability" (HA) mode. Schedulers and glideinWMS factories are run in different availability zones, so that if one fails, others can take its place. glideinWMS frontend operations are performed by a team at CERN with support from Fermilab, where much of the HA backup services are run.

One component that was not available in HA mode was the glideinWMS frontend. Having this functionality was also communicated as a high priority of CMS to the glideinWMS development community. In general this close cooperation between CMS, the developers, and the OSG forms one of the backbones of our support model.

For testing and integration, we have established a glideinWMS Integration Testbed (ITB) at CERN to test and major configuration or software changes to either glideinWMS or HTCondor. Through our close cooperations with the HTCondor and glideinWMS development teams, we also can test pre-releases of the middleware on the ITB and provide valuable feedback and bug reports to the developers. CMS holds regular meetings with the developers to communicate this feedback as well as to prioritize feature and development requests.

6. Conclusions

We are currently running a Global Pool for glideinWMS in CMS which serves physics analysis, central data production and reconstruction, overflow from the Tier-0, and opportunistic and special allocations at HPC centers. We are confident that it can scale to our needs during Run 2, at least to 200,000 parallel running jobs and beyond, based on the testing that the OSG and CMS have made during the past year and the improvements made to HTCondor and glideinWMS partially as a result of feedback from those tests.

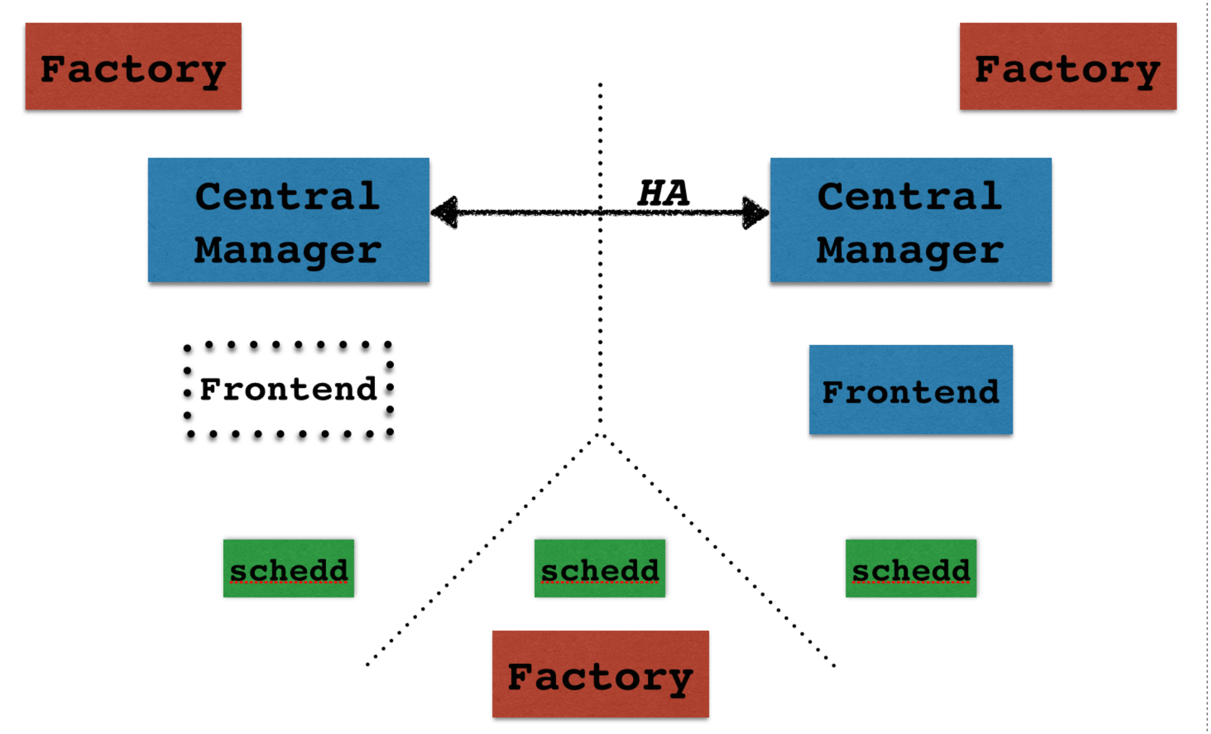

Figure 5. Most glideinWMS and HTCondor services are run in "high-availability" (HA) mode. Schedulers and glideinWMS factories are run in different availability zones, and critical central services such as the Central Manager have automatic failover to redundant machines.

References

[1] Sfiligoi I, Bradley D C, Holzman B, Mhashilkar P, Padhi S and Würthwein F 2010 The pilot way to grid resources using glideinwms *WRI World Congress* **2** 428-432

[2] Thain D, Tannenbaum T and Livny M 2004 Distributed computing in practice: the condor experience *Concurrency: Pract. Exper.* **17** 323-356

[3] S. Chatrchyan *et al.* CMS Collaboration 2008 The CMS experiment at the CERN LHC *J. Inst.* **3** S08004

[4] Evans L and Bryant P 2008 LHC machine *J. Inst.* **3** S08001

[5] Gutsche O *et al.* 2014 CMS computing operations during Run 1 *J. Phys. Conf. Ser.* **513** 032040

[6] Belforte S *et al.* 2014 Evolution of the pilot infrastructure of CMS: towards a single glideinWMS pool *J. Phys. Conf. Ser.* **513** 032041

[7] Balcas J et al. 2015 Using the glideinWMS system as a common resource provisioning layer in CMS *J. Phys. Conf. Ser.* published in these proceedings

[8] Colling D *et al.* 2015, The diverse use of clouds by CMS *J. Phys. Conf. Ser.* published in these proceedings

[9] Hufnagel D *et al.* 2015 The CMS Tier-0 goes Cloud and Grid for LHC Run 2 *J. Phys. Conf. Ser.* published in these proceedings

[10] Hufnagel D *et al.* 2015 Enabling opportunistic resources for CMS computing operations *J. Phys. Conf. Ser.* published in these proceedings

[11] Fajardo E *et al.* 2015 How much higher can HTCondor fly *J. Phys. Conf. Ser.* published in these proceedings

Acknowledgements
The present work is partially funded under program PRIN 20108T4XTM, and grants from the U.S. Department of Energy and National Science Foundation.