FERMILAB-CONF-15-603-CD

Using the glideinWMS System as a Common Resource Provisioning Layer in CMS

J Balcas[1], S Belforte[2], B Bockelman[3], D Colling[4], O Gütsche[5], D Hufnagel[5], F Khan[6], K Larson[5], J Letts[7], M Mascheroni[8], D Mason[5], A McCrea[7], S Piperov[9], M Saiz-Santos[7], I Sfiligoi[7], A Tanasijczuk[7], and C Wissing[10] 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]Imperial College London, United Kingdom
[5]Fermi National Accelerator Laboratory, Batavia, USA
[6]National Centre for Physics, Quaid-I-Azam University, Islamabad, Pakistan
[7]University of California, San Diego, La Jolla, USA
[8]INFN Sezione di Milano-Bicocca, Università di Milano-Bicocca, Milano, Italy
[9]Brown University, Providence, USA
[10]Deutsches Elektronen-Synchrotron, Hamburg, Germany

E-mail: jletts@ucsd.edu

Abstract. CMS will require access to more than 125k processor cores for the beginning of Run 2 in 2015 to carry out its ambitious physics program with more and higher complexity events. During Run1 these resources were predominantly provided by a mix of grid sites and local batch resources. During the long shut down cloud infrastructures, diverse opportunistic resources and HPC supercomputing centers were made available to CMS, which further complicated the operations of the submission infrastructure. In this presentation we will discuss the CMS effort to adopt and deploy the glideinWMS system as a common resource provisioning layer to grid, cloud, local batch, and opportunistic resources and sites. We will address the challenges associated with integrating the various types of resources, the efficiency gains and simplifications associated with using a common resource provisioning layer, and discuss the solutions found. We will finish with an outlook of future plans for how CMS is moving forward on resource provisioning for more heterogenous architectures and services.

1. Introduction

CMS [1], one of the four experiments at the Large Hadron Collider (LHC) [2], a proton-proton accelerator at CERN in Geneva, Switzerland, was designed from the beginning as a global experiment with a distributed computing infrastructure, as described in [3]. In Run1 of the LHC these resources were predominantly provided by a mix of grid sites and local batch resources. During the long shut down of the LHC, cloud infrastructures, diverse opportunistic resources, and HPC supercomputing centers were made available to CMS, further complicating the operations of the submission infrastructure.

The challenge we faced was to simplify the submission infrastructure to cover flexibly all the different resource types and the varied types of workflows we wanted to run on them. For example, as

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

shown in figure 1, the old computing model where CMS ran one type of workflow predominantly on a particular resource type has been broken down. Previously, the Tier-2 resources ran Monte Carlo generation and analysis, while the Tier-1 sites ran data and Monte Carlo reconstruction, for example.

With the introduction of cloud resources attached to the Tier-0, and the desire to utilize the high level trigger (HLT) farm during LHC shutdowns and even periods between fills, the mapping of workflows to resource types quickly evolved from roughly one-to-one mapping to many-to-many. This new model of working necessitates a unified submission infrastructure that can prioritize different types of workflows on different types of resources as requirements change.

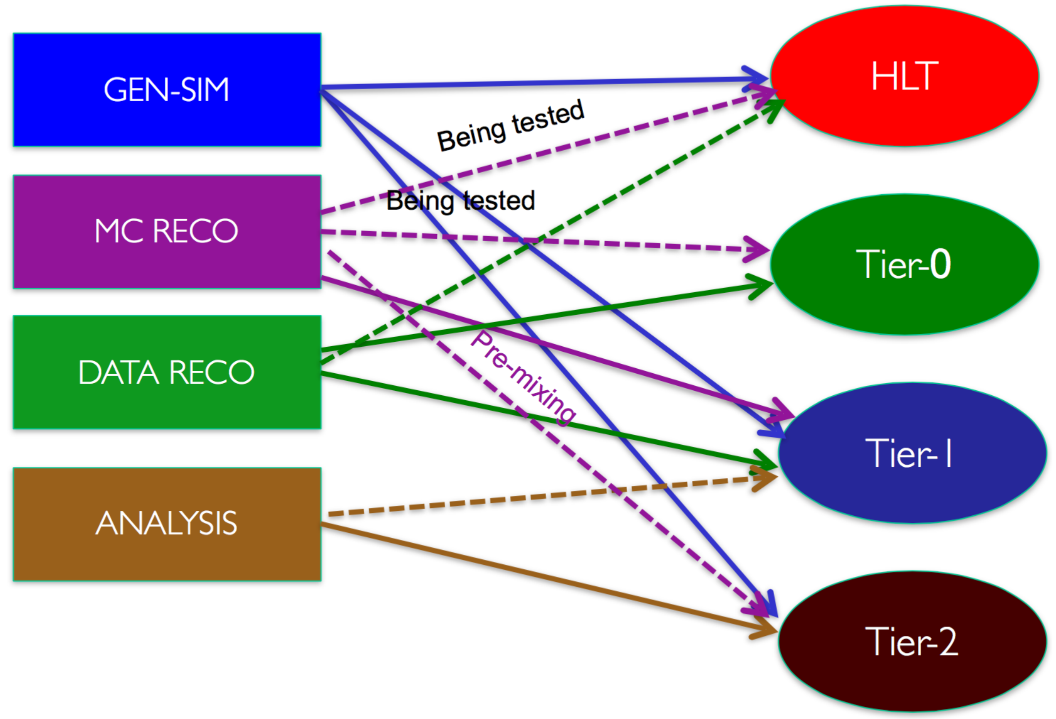


Figure 1. The evolution of the computing model such that different types of workflows can be directed to different sets of resource types, including cloud and opportunistic resources.

2. glideinWMS and HTCondor Pools in CMS

CMS transitioned to pilot-based submission systems based on glideinWMS and HTCondor during Run 1 of the LHC, completing the transition by late 2013 [4]. Inefficiencies in direct submission architectures due to networking and site issues drove the transition to a light-weight pilot submission system.

As shown in figure 2, 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 nodes which hold the job queues, HTCondor startd daemons which run on execute nodes, and a central manager (collector and negotiator) which negotiates matches between queued jobs and resources. GSI authentication based on grid certificate proxies is used to establish trust between the various elements of the pool. At job run time, CMS uses glexec [5] where implemented to switch context to the proxy used to submit the job.

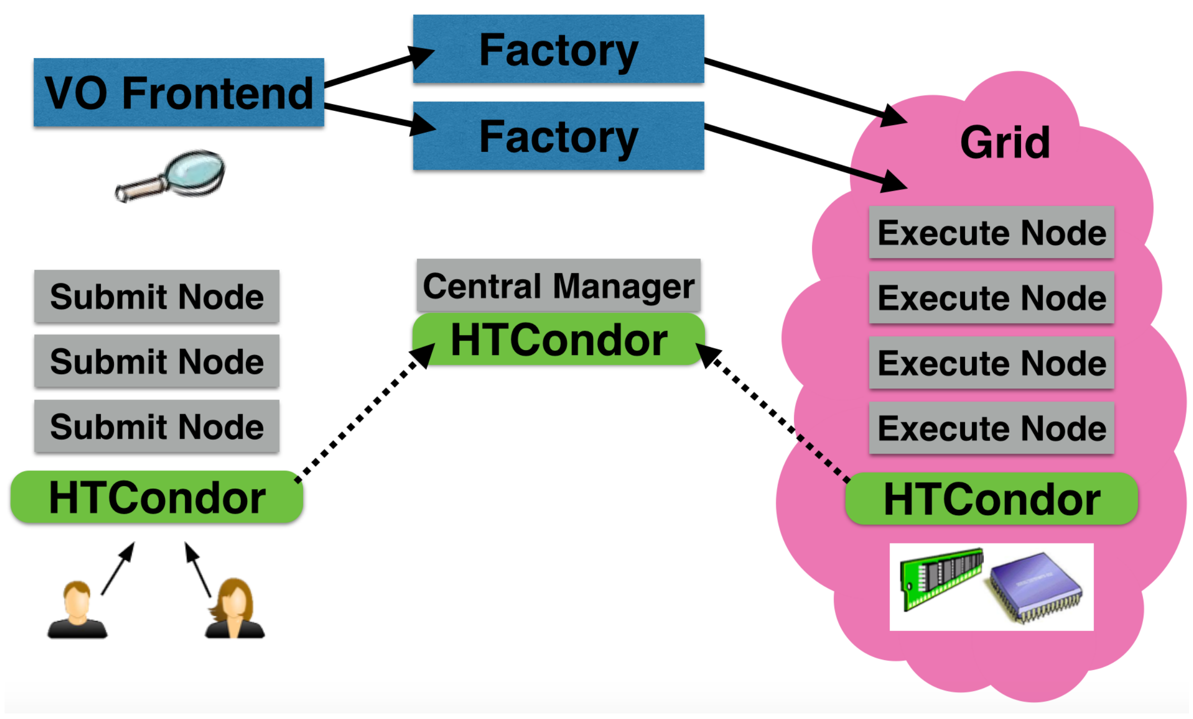


Figure 2. Architecture of a glideinWMS pilot submission system to a HTCondor pool. The main elements of glideinWMS are factories which submit light-weight pilots to grid, and now cloud, sites, and a glideinWMS frontend which requests the pilots based on need for resources in the underlying HTCondor pool. 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.

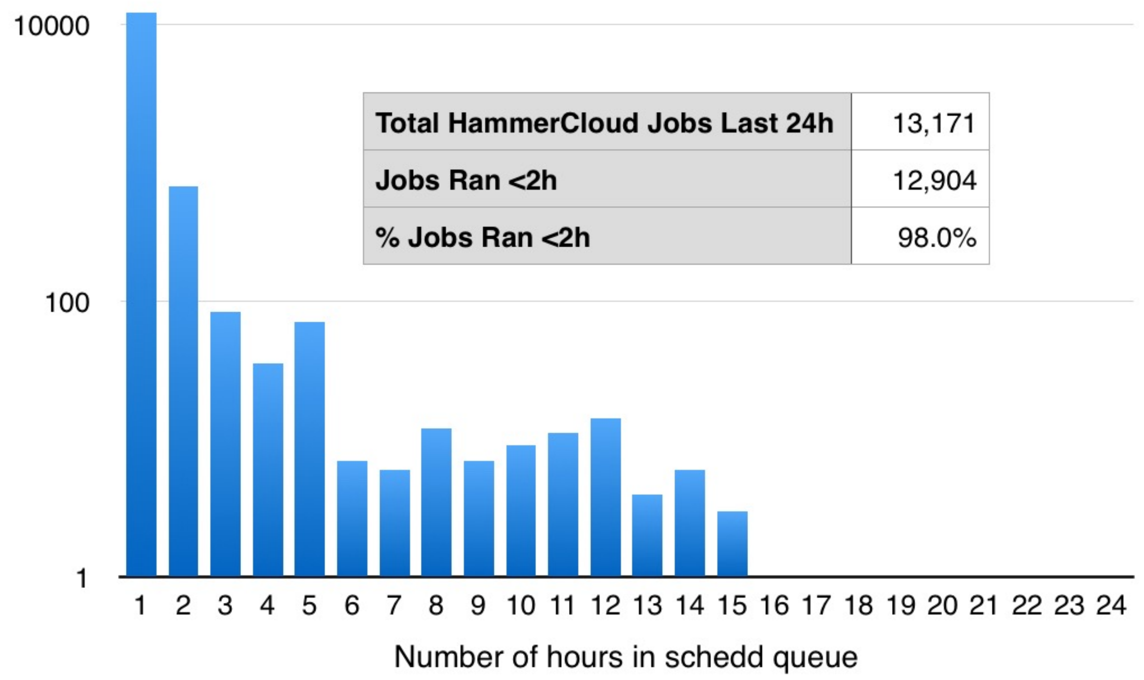
3. Clients and Use Cases

Historically, however, different type of workflows were submitted to separate glideinWMS infrastructures. In early 2014 there were separate large glideinWMS set-ups for physics data analysis, mainly directed to CMS Tier-2 sites, and another for data operations, comprising everything from Monte Carlo production to data reprocessing, mainly focused on Tier-1 sites.

With the migration of the Tier-0 to cloud resources at CERN [6,7] during LS1, the desire to use the HLT farm as a production resource [8], and the need to use spare cycles at the Tier-1 sites also for physics analysis, it became clear that a unified submission infrastructure was urgently needed. Only in this way could CMS centrally prioritize between different tasks to make most efficient use of the resources. Therefore during 2014 we migrated all of the submission nodes for production and physics analysis to a single Global Pool.

One example of the resiliency of the functionality of the Global Pool can be seen in the prioritization of jobs. In figure 3, it is shown how CMS can boost the priority and fair-share of a single user as needed. HammerCloud jobs are like CMS analysis jobs but used to test the sites for submission readiness, and as such it is imperative that these few jobs run (or fail to run) quickly. By boosting the priority factor of the user submitting the HammerCloud jobs, it was shown that 98% of the jobs submitted to sites world-wide could be made to start within 2 hours.

Another example of flexibility of prioritization, this time between different types of workflows, is shown in figure 4, in which it is demonstrated that a high-priority Monte Carlo workflow can quickly take over a large share of the resources available to the Global Pool. In the Global Pool analysis and other activities have a roughly 50%/50% fair share, except at Tier-1 sites, where the share for analysis is 5%. In figure 4 it is also seen that the high-priority Monte Carlo production takes share away from other non-analysis activities only. More about the analysis middleware CRAB3 and its interaction with the Global Pool can be found in related work [9].



Total HammerCloud Jobs Last 24h	13,171
Jobs Ran <2h	12,904
% Jobs Ran <2h	98.0%

Figure 3. It was demonstrated by boosting the HTCondor priority factor that 98% HammerCloud test jobs submitted to Grid sites world-wide could be made to start execution within two hours.

The Tier-0 was ported to Cloud resources at CERN during LS1 [6,7] and integrated into the Global Pool. However, in order to handle any risk to data taking from potential scalability or other issues with the Global Pool itself, it was decided to put the Tier-0 in its own glideinWMS set-up for the beginning of Run 2 of the LHC. The Tier-0 resources are somewhat dedicated to data processing activities only, unlike the other Tiers, whose resources are becoming more generic over time. As shown in figure 5, jobs can use HTCondor “flocking” to move between the Tier-0 and the Global Pool on demand in either direction. CMS has tested flocking Tier-0 workflows to the Tier-1 sites in the Global Pool at a scale of 50% of the Tier-1 resources. In principle, when the Tier-0 is idle, the Cloud resources could be used for other activities, as are the resources of the HLT farm.

4. Future Use Cases

While CMS has expanded into Clouds, there are many more opportunities for introducing other types of resources into a unified submission infrastructure, including special allocations and opportunistic resources. Two special allocations, one at SDSC in San Diego, and another at NERSC, are in the process of being integrated into the Global Pool, using BOSCO for pilot submission, and also deploying Parrot if CVMFS is not available at the site. CMS has established a glideinWMS development testbed infrastructure at CERN to organize and propagate knowledge of how to integrate new types of resources into a unified submission infrastructure, complementing other such testbeds at other sites in the U.S. and Europe.

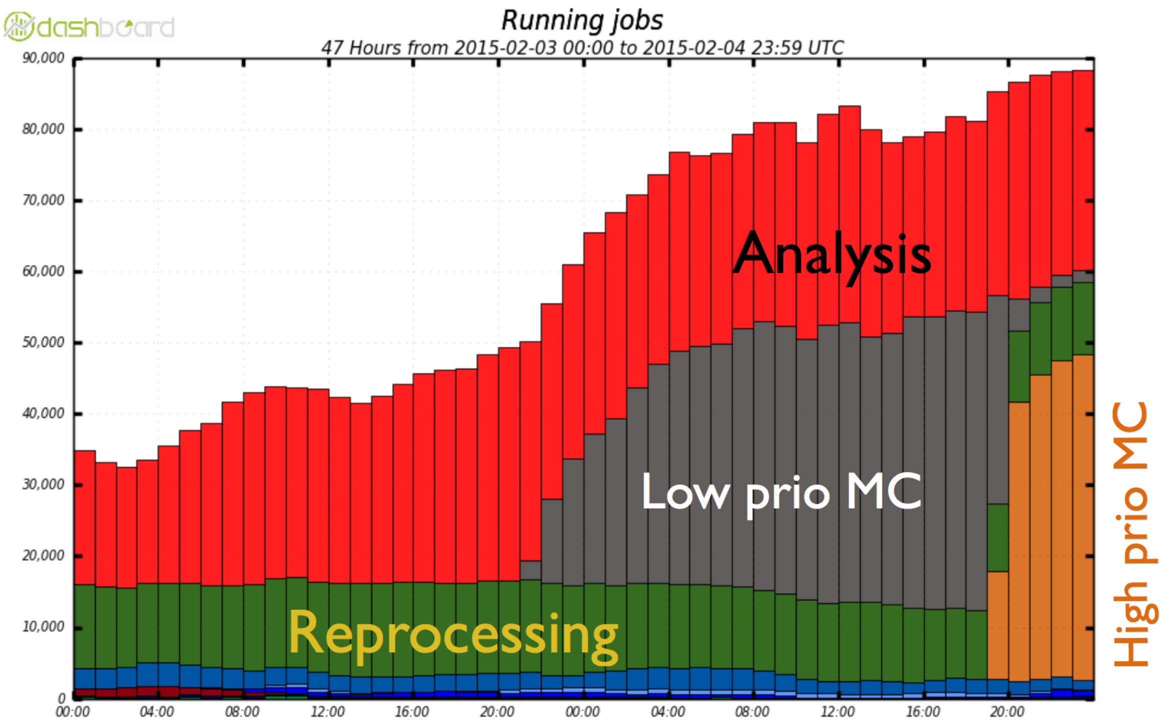


Figure 4. Demonstration that a high-priority Monte Carlo workflow (orange) can quickly take over a large share of the total resources available to the Global Pool, taking share away from lower-priority Monte Carlo workflows (grey). The share dedicated to physics analysis (red) is largely unaffected by design, since analysis and non-analysis activities generally each have a 50% share of the resources dedicated to the Global Pool.

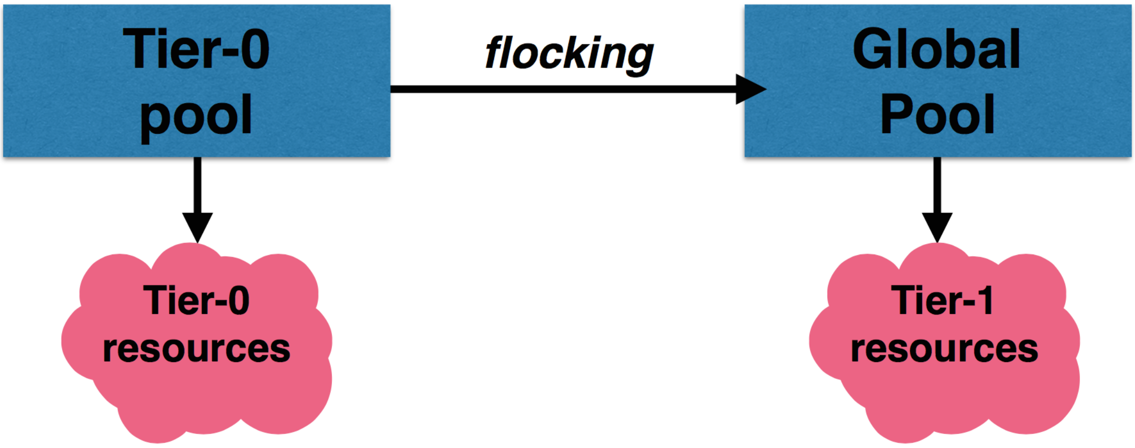


Figure 5. To mitigate any risk to data taking, the Tier-0 resources and submission infrastructure were placed in a dedicated glideinWMS set-up. However, work can use HTCondor flocking to move between the pools as needed. This has been tested at scales of 50% of the Tier-1 resources.

In the future CMS will work towards submission techniques for sites without Grid computing elements, through site-launched or user-launched pilots, including local control over user fair-share and prioritization.

5. Support Model and Scalability

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.

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 with minimal loss of data.

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 middleware developers to communicate this feedback as well as to prioritize feature and development requests.

glideinWMS frontend operations are performed by a team at CERN with support from Fermilab, where much of the HA backup services are run. CMS also cooperates closely with the Open Science Grid (OSG), who perform the glideinWMS factory operations, as well as driving much of the scale testing [10] that will allow us to reach the levels needed during Run 2.

In early 2015, the number of CPU cores pledged to CMS in the context of the WLCG and also available to the Global Pool was approximately 108,000. While CMS has demonstrated that the Global Pool can be stably run at this scale, as shown in figure 7, we expect that with the addition of Cloud and opportunistic resources, we will be able to need to reach scales of 200,000 parallel running jobs, assuming one CPU per job, during 2015. This topic is explored in more detail in related work [11]. We will make increasing use of multi-CPU applications and multi-core pilots in the near future [12].

6. Conclusions

We have deployed a single Global Pool based on HTCondor and glideinWMS which provides stable, flexible, scalable, and diverse resource provisioning to CMS for Run 2 of the LHC, backed by a strong operations team working under a written support model document, with close cooperation with the various software development teams. We plan to expand this model to cover new and different types of resources in the future, reaching ever higher scales.

References

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

[2] Evans L and Bryant P 2008 LHC Machine *J. Inst.* **3** S08001

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

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

[5] Sfiligoi I *et al*. 2012, glideinWMS experience with glexec *J. Phys. Conf. Ser.* **396** 032101

[6] 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

[7] Hufnagel D *et al*. 2015 Enabling opportunistic resources for CMS computing operations *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] M. Mascheroni *et al*. CMS Distributed Data Analysis with CRAB3 *J. Phys. Conf. Ser.* published in these proceedings
[10] Fajardo E *et al*. 2015 How much higher can HTCondor fly *J. Phys. Conf. Ser.* published in these proceedings
[11] Balcas J *et al*. 2015 Pushing HTCondor and glideinWMS to 200K+ Jobs in a Global Pool for CMS before LHC Run 2 *J. Phys. Conf. Ser.* published in these proceedings
[12] A. Perez *et al*. Evolution of CMS workload management towards multicore job support *J. Phys. Conf. Ser.* published in these proceedings

Acknowledgments

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

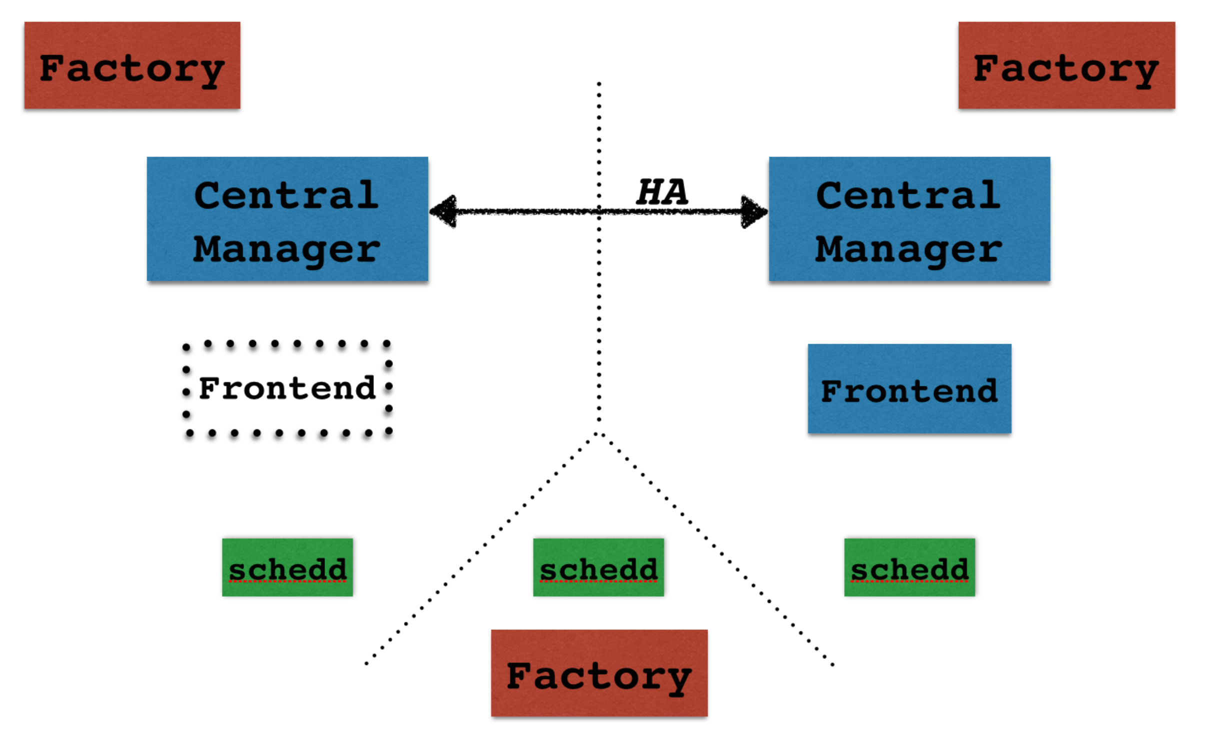


Figure 6. Most glideinWMS and HTCondor services are run in "high-availability" (HA) mode. Schedulers and glideinWMS factories are run in different availability zones, and critical central services such as the Central Manager have automatic failover to redundant machines.

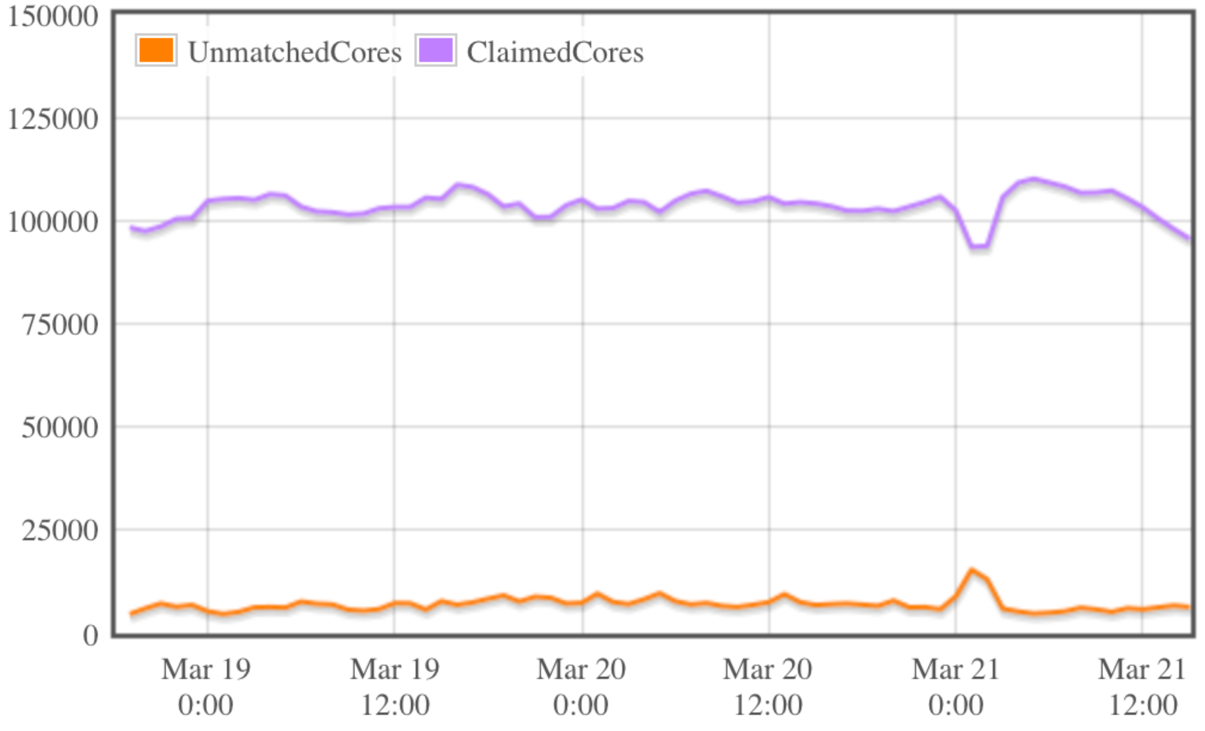


Figure 7. Demonstration of stable running of the Global Pool at scales of the WLCG pledged resources available to the pool.