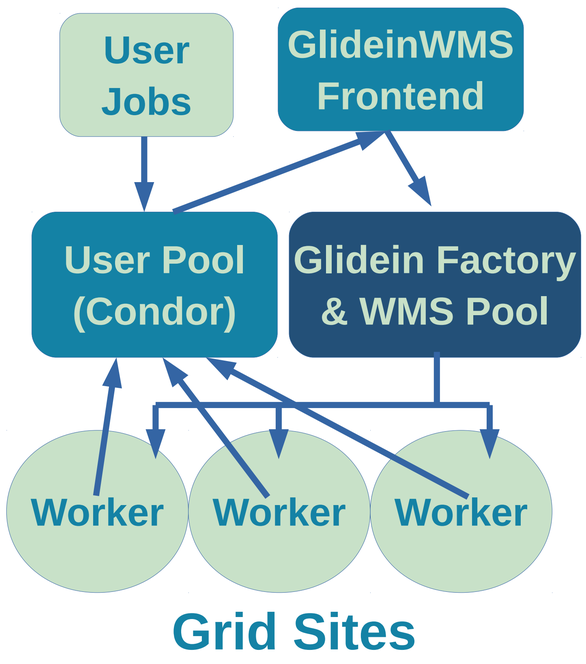
Recent Developments in GlideinWMS: Minimizing Resource Wastages

D. Box[1], J. M. Dost[2], J. Letts[2], M. Mambelli[1], M. Mascheroni[2], P. Mhashilkar[1], A. Perez-Calero Yzquierdo[3]

[1]FNAL (US), [2]UCSD(US), [3]CIEMAT & PIC (ES)

Introduction



GlideinWMS[1] is a workload management and provisioning system that lets you share computing resources distributed over independent sites. A dynamically sized pool of resources is created by GlideinWMS pilot **Factories**, based on the requests made by GlideinWMS **Frontends**. The CMS VO employs a pool that ranges from 150k to over 260k cores, worldwide distributed over 100 sites[2]. Calculating the proper pilot pressure is essential to efficiently provision resources, and being able to effectively drain resources during site downtimes also helps minimize the wastages in the system. However, the often spiky nature of the demand, and the topology of certain sites makes it difficult to tightly couple pilot submission to the actual demand.

Metasites

Entries in the GlideinWMS Factories are CE, clusters, gateways, e.g.:

CMS_T2_US_SITE**A**_CE**1**
CMS_T2_US_SITE**A**_CE**2**
CMS_T2_US_SITE**A**_CE**3**
CMS_T2_US_SITE**A**_CE**4**
CMS_T2_US_SITE**B**_CE**1**
CMS_T2_US_SITE**B**_CE**2**

The Frontend shares the load across entries. Assume 600 jobs can run on both SITE**A** and SITE**B**, pilots request will be:

CMS_T2_US_SITE**A**_CE**1 = 100**
CMS_T2_US_SITE**A**_CE**2 = 100**
CMS_T2_US_SITE**A**_CE**3 = 100**
CMS_T2_US_SITE**A**_CE**4 = 100**
CMS_T2_US_SITE**B**_CE**1 = 100**
CMS_T2_US_SITE**B**_CE**2 =100**

Pilot submission was uneven: 400 for SITEA and 200 for SITEB

Metasites have been introduced to group Factory Entries in one single logical element, the only item seen by the Frontend. Pilot pressure is hence calculated on per-site basis therefore decoupling it from site's internal configuration.

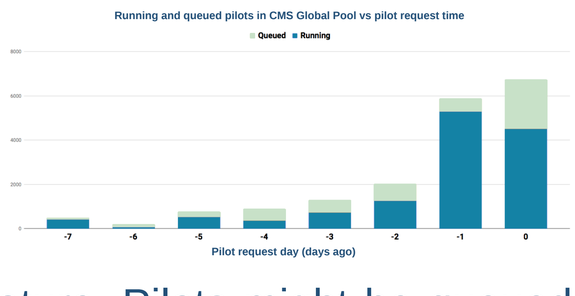
SITE**A**
CMS_T2_US_SITE**A**_CE1
CMS_T2_US_SITE**A**_CE2
CMS_T2_US_SITE**A**_CE3
CMS_T2_US_SITE**A**_CE4
SITE**B**
CMS_T2_US_SITE**B**_CE1
CMS_T2_US_SITE**B**_CE2

SITE**A = 300** (75 per 4 CE)
SITE**B = 300** (150 per 2 CE)

Even submission.
Additional benefit: compact configuration!

Other Features

Idle Removal



User's job demand is spiky by nature. Pilots might be queued to a site, but the pressure could go away while they they wait to start.

Remove idle excess to better match pilot submission to actual demand. Either automatic removal after a timeout, or Frontend triggered removal.

Draining Sites

Sites that want to vacate some or all resources in a cluster can now use the **Machine Job Feature**[3] mechanism to tell the pilot on a node to stop accepting jobs (gentle draining) before eventually kill the job and exit (not so gentle draining).

Conclusions

Optimizing the usage of resources is essential in a context where the computing demand is constantly increasing. Grouping different entries into the same metasites, improving the way a spiky job pressure is handled by removing pilots in excess of the current demand, and giving the sites the possibility of cleanly drain using native GlideinWMS mechanisms, are all recently introduced features that contribute to minimize resource's wastages.

References

[1] Igor Sfiligoi et al., "The Pilot Way to Grid Resources Using glideinWMS", Computer Science and Information Engineering, 2009.
[2] A. Perez-Calero Yzquierdo et al., "Exploring GlideinWMS and HTCondor scalability frontiers for an expanding CMS Global Pool", CHEP18 Oral Presentation #438.
[3] M. Alef et al., "The machine/job features mechanism", Journal of Physics: Conference Series, Volume 898