Journal of Physics: Conference Series **664** (2015) 052013 doi:10.1088/1742-6596/664/5/052013

Running and testing GRID services with Puppet at GRIF-IRFU

S Ferry[1], F Schaer[1], JP Meyer[1]

[1]CEA Saclay,
91191 Gif-Sur-Yvette, France

E-mail: sophie.ferry@cea.fr

Abstract. GRIF is a distributed Tiers 2 centre, made of 6 different centres in the Paris region, and serving many VOs. The sub-sites are connected with 10 Gbps private network and share tools for central management. One of the sub-sites, GRIF-IRFU held and maintained in the CEA-Saclay centre, moved a year ago, to a configuration management using Puppet. Thanks to the versatility of Puppet/Foreman automation, the GRIF-IRFU site maintains usual grid services, with, among them: a CREAM-CE with a TORQUE+Maui (running a batch with more than 5000 jobs slots), a DPM storage of more than 2 PB, a Nagios monitoring essentially based on check_mk, as well as centralized services for the French NGI, like the accounting, or the argus central suspension system. We report on the actual functionalities of Puppet and present the last tests and evolutions including a monitoring with Graphite, a HT-condor multicore batch accessed with an ARC-CE and a CEPH storage file system.

1. Introduction

Grid computing has become a fundamental tool for scientific communities such as high energy physics or human sciences as well as for planetary or life sciences.

Thanks to high speed networks, the grid is able to share computing and storage resources all over the world. In 2010 the EGI project was created for developing and maintaining the operations over the European Grid [1]. At the national level, the National Grid Initiatives (NGIs) relays the operational management to the sites.

GRIF is a distributed Tiers 2 in France, and the sub-sites composing it are working closely together. One of the sites, GRIF-IRFU, has developed its own central configuration system based on *Puppet* [2] and tests several solutions for batch or storage systems.

In this document, GRIF and IRFU are presented in section 2 and section 3. Sections 4 and 5 present the current status of the GRIF-IRFU site describing respectively its batch system and monitoring. The ongoing tests are given in section 6. Section 7 ends this document with the conclusion.

2. GRIF

The project "GRIF" (*Grille de production pour la Recherche en Ile-de- France*) [3] is a combined effort of 6 high energy physics laboratories aiming at providing a single resource of storage and computing using grid technologies. The 6 sub-sites are located in the Region Ile-De-France as shown in figure 1.

The total resource is more than 10000 jobs slots and 6.6 PB, and is accessible to all users working at public companies or institutes. The goal is to be part of the WLCG project as a Tiers 2 site for the LHC experiments, as well as to provide computing and storage resources for non LHC experiments included in EGI project.

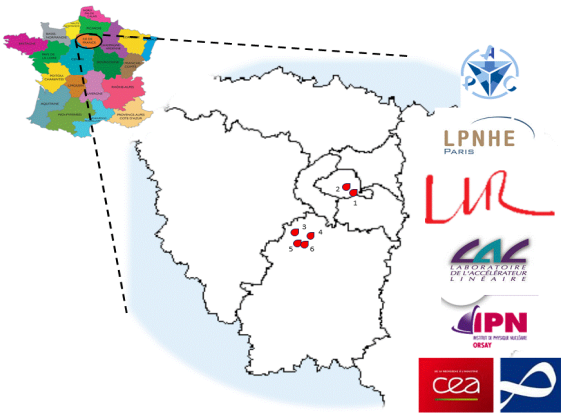

Figure 1. Name and location of the 6 sub-sites of the GRIF T2.

GRIF started in 2005, and represents now 30% of the total normalized CPU time provided by the French NGI (France-Grilles).

The network infrastructure is an important part of GRIF success. The LHCONE network has a dedicated 20 Gbps fiber which links GRIF to the French Tier-1 Computing Center (Centre de Calcul de l'IN2P3 in Lyon) through Orsay PoP. Orsay is connected with one 10 Gbps link to the two inner Paris sites and with another 10 Gbps link to IRFU. The 3 sites LAL, IPNO, LLR share a 20 Gbps link. The network interconnection is presented in figure 2.

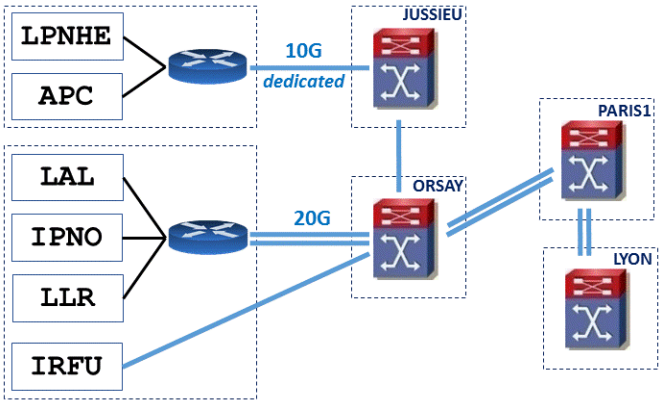

Figure 2. GRIF network interconnection

The GRIF organisation is based on a scientific council which discuss project goals and funds, and on a technical committee which deals with the day-to-day operations. The technical committee is composed of the engineers at each sites. They work together for a cooperative administration, following:

- Monthly face to face meetings for long/medium term projects,
- Weekly remote meeting for short term actions,
- Daily interaction by email.

Amongst them, some technical aspects include:

- A shared configuration tool (*Quattor*), which was used to share configurations between the 6 sites. At the IRFU site, it was decided to move out for a fully enabled puppet configuration with its own central server.
- Possibility to intervene in other sites with inter-site logins and *sudo* commands.
- A distributed monitoring infrastructure: the IRFU server collects all monitoring information from each sub-site slave server.

A "Tour de garde" was established on a volunteering basis, where one checks for the 6 sub-sites about errors and malfunctions. A MoU has been in preparation for a year, but the volunteering principle still works.

3. GRIF IRFU

One of the sub-sites, "IRFU", is located inside the secured CEA institute [4] (*Commissariat à l'énergie atomique et aux énergies alternatives*) and have been part of GRIF since the very beginning. The computing resources are open to ALTAS, ALICE and CMS as well as to a tenth of non-LHC VOs.

In terms of computing and storage resources, it represents a large part of the full T2: 40% of GRIF storage and 45% of GRIF computing. The batch system is TORQUE+Maui with a CREAM-CE. In total, 340 servers are providing more than 5000 job slots. DPM is used as the storage system, with 40 file system servers supplying 2.6 PB.

Most of the Grid services are running at GRIF-IRFU, including: *squid*, *perfsonar*, *wms*, *vobox*, *UI*, *BDII*… etc. Some central services for the whole French NGI are also provided: *argus-ngi* (central suspension system), *accounting-ngi*.

4. Puppet at GRIF IRFU

Originally GRIF had a shared configuration and a central server sharing configuration files, and Quattor [5] was the configuration system. It is still true except for the IRFU sub-site which has moved to *Puppet* configuration management.

4.1. Transition to Puppet

With the CEA policies and requirements, the GRIF-IRFU configuration often had to be different from the rest of GRIF. Benefits of having a central configuration for the whole GRIF were lost and the central server was considered as a single point of failure (SPOF). Quattor documentation was scarce and learning it was uneasy for a new system administrator. In our experience, at that time, many elements coming with Quattor were time consuming (compiling times close to 10 min, manual package dependency checks, failing *ncm-accounts* …etc.) which had its share of responsibility in failing the upgrade from *gLite* to *EMI* on time.

Puppet was chosen mainly because CERN chose it, and a very complete documentation was already available together with a large responsive community. Professional trainings existed (and still exists), and learning Puppet was award winning for our temporary contract sysadmin.

The basic installation can be limited to a puppetmaster server and a puppet agent. Our first tests included Foreman [7] as a standalone application, using the foreman-installer (a collection of puppet modules that installs everything required for a full working Foreman setup).

These tests were deploying simple configurations services, with our own modules. Later, even Foreman was configured by Puppet itself, and modules from GitHub and the Puppet Forge were used as well. As an ENC, Foreman can detect all the environments and manifests contained on a puppetmaster, and import them automatically. The Git [6] repository (version control system for the modules), *hiera* (externalisation of parameters to keep site-specific data out of manifests), other tools

like *librarian-puppet* (modules version dependence checking tool) were added as they became necessary, as more complex services were deployed with Puppet.

We started with Puppet release 2.7 and updated our infrastructure to each releases with no major issues. It is now in Puppet 3.8.

Our puppet configuration deployment follows a workflow which has been considered as a best practice (but might be subject of debate now):

- One Git repository contains all puppet modules.
- One Git repository contains hiera configuration files.

The Git repositories contain "post commit hooks", which on successful updates ("git push") synchronize all manifests in each of environments in the puppetmaster.

Next sections present the elements included in our system management. The schema figure 3 represents the configuration management at IRFU.

4.2. Git repository

The repository contains all the puppet modules which define the services to deploy. The services are divided according to the environment: dev (to develop new services), pre-production (to test before deployment), and production (running services). Each environment is dynamically mapped to a branch in Git. If necessary a new branch can be created to correspond to a new environment.

4.3. Foreman

Foreman is a management tool for servers. It takes care of the provisioning (bare-metal and virtual hosts), the initial configuration and the configuration monitoring. The services (*i.e.* puppet modules) to be deployed on a specific server are defined using the web frontend. Configuration monitoring is followed with the dashboard.

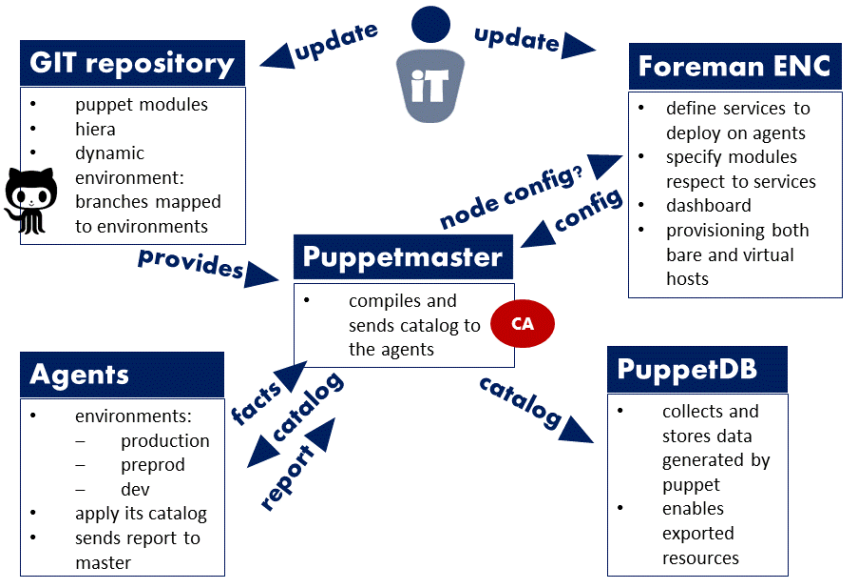

Figure 3. Puppet at GRIF-IRFU. One puppetmaster manages all the nodes.

4.4. Puppet

Each agent (*i.e.* puppet client) sends its facts to the puppetmaster. The latter asks Foreman which configuration is to be applied. Foreman sends the configuration which is compiled into a catalog by the puppetmaster and sent to the agent. After applying the catalog, the agent sends its report. It can be checked with the Foreman dashboard. The *puppetDB* collects the reports, catalogs and configurations and enables exported resources.

One puppetmaster manages 380 servers (bare-metal or virtual) in production, pre-production or dev. The Git repository, the puppetmaster, foreman and the puppetDB all are hosted on a single server.

5. Monitoring

In order to monitor the site, two systems are set up: Nagios to check services and server health, and Graphite to report the state of running and waiting jobs.

5.1 Open Monitoring Distribution

The convenient way to set up a monitoring with *Nagios* and *check_mk* is using *OMD* [8]. Check_mk agents on each server to monitor execute their defined tests, caching the results. A monitoring server pulls the results to be published on the check_mk web monitoring. Tests include integrated ones like server health, or user defined ones like number of threads or DIMM memory checks.

In order to dynamically include all check_mk agents to the monitoring server, we use exported resources from the puppetDB. Tags are included in the puppet configuration of the servers and are exported to the puppetDB. The monitoring server pulls the exported resources out of the puppetDB in order to list the check_mk agents and their tests. Results of the tests are published via the check_mk web server.

Each of the 6 GRIF sub-sites have their own monitoring server. All of them report to the IRFU one so all servers can be monitored at once.

5.2 Grafana

Grafana is an open source graph editor for *Graphite* [9]. The Graphite API is used to query the BDII or the schedulers about various metrics. It pushes the results to the Grafana application which parses the results to display. An example of graphic is shown figure 4.

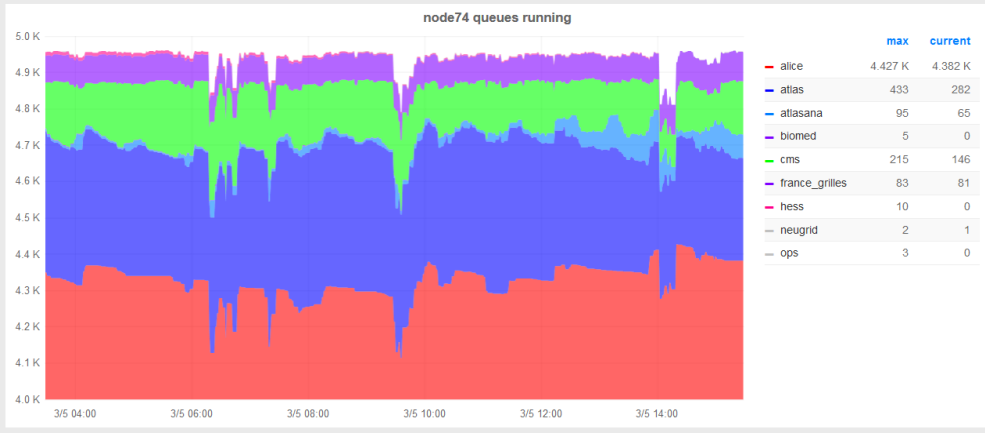

Figure 4. Graphite graph showing the number of running jobs for the CREAM-CE queues. Each colour represents a different VO.

6. Testing batch and storage systems

6.1. HT Condor

As MAUI is not supported, and since the growing demand for multicore job slots, HT Condor was chosen to be tested with an ARC-CE as computing element. The system is in the pre-production environment, nevertheless the 3 LHC VOs are accessing it. A mix of single core and 8-core queues are defined, and defragging liberates slots for the 8-core queue. The demand for 8-core jobs is still rather small, so no problem of concurrence between single core and 8-cores appears. The queues provide 216 cores over 12 servers. Currently, more workernodes in the CREAM-CE production farm is migrated to the ARC-CE batch. Eventually, all the production will move to the ARC-CE+HTCondor system.

6.2. CEPH

CEPH [10] appears to be a storage solution usable in many cases of high availability file servers. We used decommissioned hardware for testing reading and writing performance in accessing data. First tests were performed on Dell 2950 with MD1000 bays, where new PERC card to enable *jbod* access to disks were installed. They showed unsatisfactory results.

Second tests were carried out on 5 Dell R510 with MD1200 bays containing 11x2 TB HDD disks plus 1 SSD for the journals. Six types of pool were created, 3 using replication (replicated pools) and 3 using erasure coding (erasure coded pools). The detail of the pool are shown in table 1. For each of the pool reading and writing *rados* tests were performed with 3 different block size and 4 different parallel I/O values. This represents 24 tests on each pool.

Table 1. Definition of the CEPH POOLS for testing purpose.

ERASURE CODED POOLS						**REPLICATED POOLS**		
ID	**k**	**m**	**failure**	**PG_num**		**ID**	**Replic#**	**PG_num**
0	4	1	Host	1024		10	2	2048
1	4	1	OSD	1024		11	4	1024
2	12	3	OSD	256		12	10	512

Performance results were as low as a few MB/s up to 1200 MB/s. Some trends were seen:

- with smaller block size, the bandwidth is broader (better performance) with the number of parallel IO
- with larger block size, the bandwidth slightly shrinks (worse performance) with the number of parallel IO

This was seen in the writing performance as well as for the reading performance. Worse and best results numbers are presented figure 5.

In overall, these performance are not enough for the purpose of WLCG storage. After discussion with the manufacturer, it appeared that the SSD were sold with a firmware bridling the speed of the controller. Further tests are ongoing.

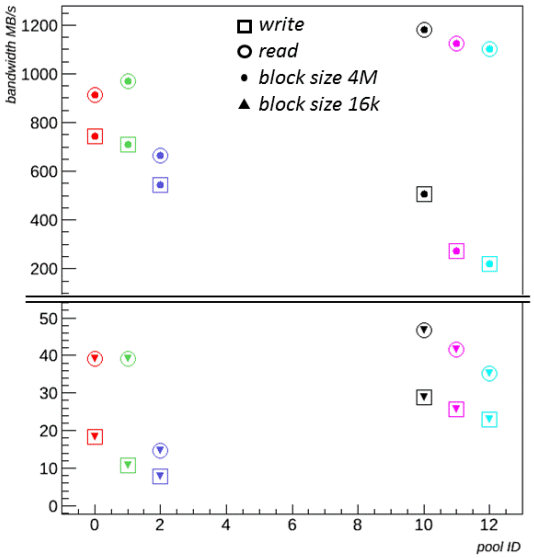

Figure 5. Best and worst results reading and writing performance for the 6 different tested pools. Pool IDs correspond to the pools described in table 1.

7. Conclusion

The choice of a distributed site for GRIF is a success, as it has become one of the major T2 for the French NGI. On key of this achievement is the sharing of manpower competences and expertise.

Thanks to Puppet versatility, GRIF-IRFU site has been able to grow its services and resources as well as testing new products. HT-Condor and ARC-CE are currently in pre-production state, and CEPH is examined.

In the close future, *puppet server* will take over the puppetmaster, and an *Openstack* instance will be tested as well.

References

[1] www.egi.eu
[2] www.puppetlabs.com
[3] www.grif.fr
[4] www.cea.fr
[5] www.quattor.org
[6] www.git-scm.com
[7] www.theforeman.org
[8] www.omdistro.org
[9] www.grafana.org
www.gaphite.wikidot.com
[10] www.ceph.com