

JINR Infrastructure for Tier3 Monitoring Development

N.A. Kutovskiy^{1,2}, I.S. Kadochnikov¹, L. Valova^{1,3}, A. Yakshov

email: Nikolay.Kutovskiy@jinr.ru, ¹Laboratory of Information Technologies, JINR, Dubna

² National Scientific and Educational Centre of Particle and High Energy Physics of the Belarusian State University, 220040, Minsk, Belarus

³Technical University of Kosice, 040 11, Kosice, Slovakia

According to ATLAS Tier3 survey [1], a several types of local resources management systems (LRMS) and mass storage systems (MSS) are used on ATLAS tier3 sites:

- LRMS: PROOF, PBS, Condor, Oracle Grid Engine, LSF.
- MSS: XRootD, dCache, DPM, NFS, GPFS, Lustre.

The development of a software suite for the local site monitoring assumes the following activities:

- validation of existing monitoring tools for each of the used components,
- development and debugging new monitoring tools.

The activities listed above imply the following:

- deployment of a separate testbed for each of the LRMS and MSS reported as being used at ATLAS Tier3 sites,
- Ganglia and Nagios servers deployment,
- Ganglia or/and Nagios agents installation and configuration for a specific testbed,
- Installation and validation of the additional ganglia plug-ins for monitoring metrics collection as well as non-related to ganglia monitoring tools.

Testbeds with different LRMS and MSS have to be run in parallel and must be available at any time. It causes sufficient hardware requirements if each component of each testbed is run on a dedicated physical server.

Monitoring tools deployment and development as well as testbeds operation may require re-deployment of a certain testbed or its parts. Hence, there should be the ability to easily manage the testbed parts.

Testbeds performance is not a critical issue for such kind of task as development.

Taking all mentioned above into account, testbeds can be run on virtual machines.

Virtualization allows one more effectively to utilize hardware resources and provides abilities to perform the following operations quickly and easily:

- create VM from existing image/template,
- backup VM before sufficient changes and quickly restore VM from backup if needed.

Since most of the components of each testbed can be run on linux (inside VM) and physical servers are running on linux as well, the virtualization on the OS-level can be used which is more lightweight and faster than full (hardware emulation) or paravirtualization approaches. The only exception is MSS Lustre since it is installed as a kernel module what can be done only in VMs with a full hardware emulation.

Among possible candidates the OpenVZ [2] as a solution for virtualization on OS-level and Xen [3] as a product providing a full hardware emulation was chosen by the following reasons:

- stable and actively developing software with a sufficient set of tools for VM management and monitoring,
- strong and helpful community,
- good documentation,
- freeware (GNU GPL v2 license).

Apart from that the services of training and testing grid infrastructure (see [4] for more details) have been run OpenVZ-based VMs successfully for about 5 years.

For the time being the following LRMS and MSS are deployed on the JINR tier3 infrastructure (see figure 1 and table 1): PBS, Condor, XRootD, Lustre, PROOF, OGE. Apart from that the Ganglia, Nagios and development servers are deployed. All the testbeds excluding Lustre are running on a single OpenVZ-enabled server. Lustre services are deployed on Xen-based VMs on a separate server.

A report on the performed work described above (see [5]) was delivered at ATLAS Computing Technical Interchange Meeting (31.05.2011–02.06.2011, JINR, Dubna).

There is ongoing work to enable OpenVZ support in OpenNebula [6] software aimed to build private,

Table 1: The List of running services of the LRMS and MSS testbeds

testbed name	services
PBS	torque headnode (HN) + worker node (WN) + ganglia (gmond, gmetad, webfronted) + jobmonarch, 2 torque WNs + gmond
PROOF	HN + 2 WNs
Condor	HN+WN, 2WNs
OGE	HN + ganglia (gmond, gmetad, webfrontend), 2 WNs + gmond
XRootD testbed 1	manager + gmond + reportxrd.py, server + gmond, server + ganglia (gmond, gmetad, webfrontend)
XRootD testbed 2	manager + gmond + reportxrd.py, server + gmond, server + ganglia (gmond, gmetad, webfrontend)
Lustre	MDS + gmond + gmetad + gweb + t3mon-site-lustre, OSS + gmond, client + gmond

public and hybrid clouds. OpenVZ-enabled OpenNebula is planned to be used for building a private cloud at JINR based on OpenVZ (for OS-level virtualization) and KVM (for full hardware emulation) hypervisors (both hypervisors can co-exist on the same node). Then the infrastructure for the tier3 monitoring development is planned to be migrated into the JINR private cloud.

References

- [1] D. Benjamin ATLAS tier3 survey // report at ATLAS Software and computing workshop (29.11.2010 03.12.2010), CERN.
- [2] <http://openvz.org>
- [3] <http://xen.org>
- [4] <http://gridedu.jinr.ru>
- [5] I. Kadochnikov et al. JINR infrastructure for Tier3 simulation // report at ATLAS Computing Technical Interchange Meeting (31.05.2011 02.06.2011), JINR, Dubna.
- [6] <http://opennebula.org>

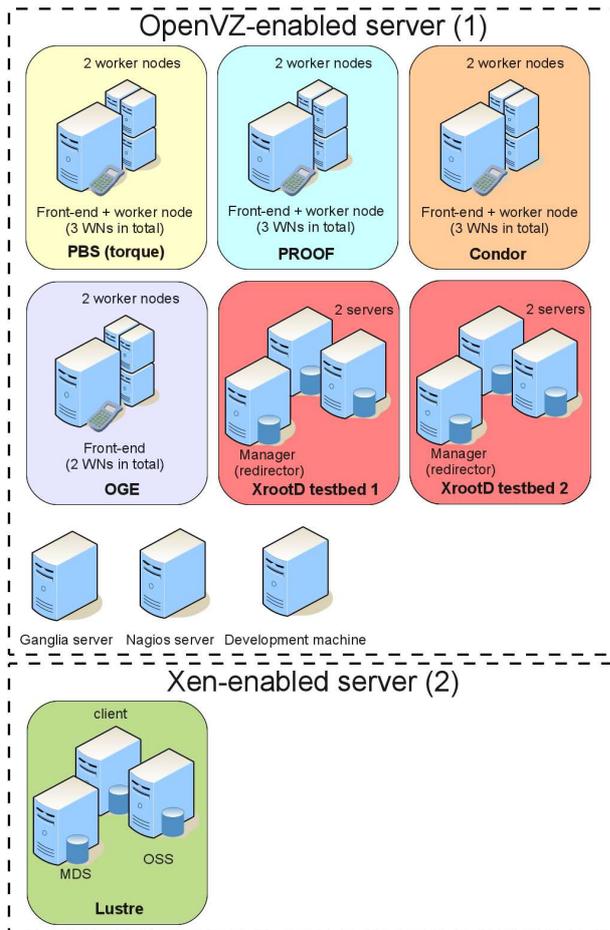


Figure 1: A schema of LRMS and MSS testbeds distributions over the servers