INFIN Tier-1 Testbed Facility

Daniele Gregori\textsuperscript{1,2}, Alessandro Cavalli\textsuperscript{1}, Luca dell’Agnello\textsuperscript{1}, Stefano Dal Pra\textsuperscript{1}, Andrea Prosperini\textsuperscript{1}, Pierpaolo Ricci\textsuperscript{1}, Elisabetta Ronchieri\textsuperscript{1}, Vladimir Sapunenko\textsuperscript{1}

\textsuperscript{1} INFN-CNAF, viale Berti-Pichat 6/2, 40127 Bologna, Italy
\textsuperscript{2} Corresponding Author
E-mail: daniele.gregori@cnaf.infn.it

Abstract. INFN-CNAF, located in Bologna, is the Information Technology Center of National Institute of Nuclear Physics (INFN). In the framework of the Worldwide LHC Computing Grid, INFN-CNAF is one of the eleven worldwide Tier-1 centers to store and reprocessing Large Hadron Collider (LHC) data. The Italian Tier-1 provides the resources of storage (i.e., disk space for short term needs and tapes for long term needs) and computing power that are needed for data processing and analysis to the LHC scientific community. Furthermore, INFN Tier-1 houses computing resources for other particle physics experiments, like CDF at Fermilab, SuperB at Frascati, as well as for astro particle and spatial physics experiments. The computing center is a very complex infrastructure, the hardware layer include the network, storage and farming area, while the software layer includes open source and proprietary software. Software updating and new hardware adding can unexpectedly deteriorate the production activity of the center: therefore a testbed facility has been set up in order to reproduce and certify the various layers of the Tier-1. In this article we describe the testbed and the checks performed.

1. Introduction
LHC produces data that are distributed from the CERN, identified as Tier0, to the Tier-1 centers that make data available to about two hundreds Tier-2 centers all over the world. The Italian Tier-1 \cite{1} houses 14 PB of tape space, 10 PB of disk space and more than 1000 worker nodes. INFN Tier-1 supports a wide range of experiments belonging to Worldwide LHC Computing Grid (WLCG) and to other physics communities for which a high level of reliability, scalability and performances needs to be guaranteed. Each of these experiments has specific requirements that the center satisfies in production with a given hardware and software configuration. Hence a dedicated controlled environment has been designed and set up through a testbed in order to verify and validate any change in the hardware and software layers without impacting the production.

This testbed has allowed the center to evaluate also Grid Enabled Mass Storage System (GEMSS) \cite{2}, a new mass storage solution. GEMSS is a combination of several software, such as General Parallel File System (GPFS) \cite{3}, Tivoli Storage Manager (TSM) \cite{4}, StoRM \cite{5} and a set of further scripts, that can be tested separately against different versions. Because of the complexity of GEMSS the testbed is the mean by which INFN Tier-1 can pre-verify any change in the solution and therefore assure a high quality service in production.

The rest of the paper is structured as follows. Section 2 details the procedure adopted at the INFN Tier-1 to put new software releases and hardware components into production. Section 3...
describe the elements involved in the GEMSS solutions and the adopted hardware. Section 4 describe the testbed installation and configuration too. Section 5 shows the main performed tests, and finally Section 6 summaries conclusions.

2. INFN Tier-1 Deployment Procedure
At INFN Tier-1 a procedure has been defined to put new hardware components and software releases into production. This procedure is part of the INFN Tier-1 production policy and is composed of phases and decisions: phases express the maturity of software as it advances from development to production (see Table 1), whilst decisions, commonly a Passed or Failed test, determine a change in the progressing phases such as installation and configuration testing when deployment administrator performs sanity tests, system testing when deployment testers performs functional and performance tests.

| Name         | Description                                                                 |
|--------------|-----------------------------------------------------------------------------|
| Beta         | It is when software begins to be feature complete. Software in beta phase is typically the first time that the software is available outside of the organization that developed it. |
| Ready for testing | It is when software is ready to be validated and verified to be a final product and be release unless fatal bugs emerge. At this time all product configuration features have been tested. |
| Ready for production | It is when software is ready to be installed and configured in production. Software in this phase has a sufficient quality for mass distribution. |
| Rejected Release | The rejected release phase is when wrong behaviours are found in software. |

Figure 1 shows how the deployment procedure is modeled in case of software and highlights what actors (i.e., release manager, customer service levels, deployment tester, deployment administrator, deployment manager) are involved in each decision. The workflow is followed by the INFN Tier-1 team every time an hardware and software upgrade is suggested from software and hardware providers. As soon as the deployment manager receives a notification by the release manager of an adopted software in beta phase, the deployment administrator starts the installation and configuration testing decision: if he or she evaluates that sanity tests are passed, administrator will communicate to the deployment tester that the software is ready for testing. At this point the deployment tester starts the system testing decision: if after this session he or she verifies that system tests are passed, tester will communicate to the deployment manager that the software is ready for production. System testing is based on functional and performance tests in relation with the production scenarios that are commonly set at INFN Tier-1.

3. Testbed Hardware Layout
The testbed is realized with dedicated resources and systems as shown in Figure 2. Specifically, the testbed reproduces a small production layout with 11 servers that are distributed among several services as detailed below:
Figure 1. Workflow for deploying new hardware and software at INFN Tier-1.

- 2 GPFS servers
- 1 TSM server
- 2 Tsm-hsm client and storage agent
- 2 GridFTP[7] server
- 4 StoRM server (3 Front-End and 1 Back-End)

All these machines are DELL 1950 with 2 processors Intel Xeon 5110 dual core, 8 GB of RAM, 1 NIC for 1 Gbit/s, two host bus adapter (HBA) Fibre Channel ISP2432 card with 4 Gbit/s connections each to the SAN or Tape Area Network (TAN). GPFS performs the data access in a shared mode between servers on 2 distinct file systems formed by 2 LUN each one for a total size of 10 TB. The disk hardware is EMC CX380-3 connected to the SAN with 4 Front-End link per controller for 4 Gbit/s each. Storage disk perform 230 MB/s bandwidth and 500 IOPS of
throughput per filesystem as shown in Figure 3, the SAN configuration is scalable and we can improve the I/O when needed. The SAN fabric is structured on a Fibre Channel Director switch Brocade 48000 equiped with 8 hot swap blade of 48 optical port each one for 4 Gbit/s. Finally, the tape backend is managed by the TSM software that thanks to its server and storage agent HSM client migrates and recalls files from disk to tape media and vice versa. Our tapes are T10000T1 and T10000T2 models with capacity of 512 GB and 5 TB respectively. These media are hosted in the Oracle SL8500 tape library and tape drives are T10kA and T10kC, which can achieve an I/O throughput respectively equal to 120 MB/s and 240 MB/s.

![Figure 3. Testbed file system performance in term of Bandwith and Throughput.](image)

### 3.1. Configuration Details

GEMSS is configured by using its own configuration file in each file system that needs to be interfaced with the tape. Through this file an administrator can define whether to use extended attributes in StoRM or exclude it. Can define the number of tape drives that each machine HSM-Storage Agent can be used for migration and for the recall, the timeouts and the percentage of filling of the file system to enable the garbage collector.

The testbed is composed of 2 GPFS file systems each of which is divided into 2 separate fileset in order to verify the functionality with more file systems. GPFS policies have been defined to collect the list of recent files written to disk to migrate them to tape. Side TSM storage pools are defined as tape, each representing a different target on which to migrate data from different file set following a correspondence fileset-storage pool. In addition, the storage pool can be configured to use different types of tape drives that are in our case T10kA and T10kC.

StoRM is configured by having multiple-frontends servers and one backend server. The Front-End exposes the SRM service interface and manages user authentication. The Back-End is the StoRM core component that executes all SRM requests, interacts with file systems via a driver mechanism, publishes a Web Service to be used by external interface for implementing full Hierarchical Storage Management (HSM) functionalities. A database component, called
Request DB, is configured with the backend and stores SRM request data and StoRM internal metadata.

The gsiFTP protocol is configured by having multiple-gridFTP servers that are used to access disk and tape data.

4. Testbed Installation
The testbed installation is characterized by two factors: the former is the SAN zoning by using a Brocade Web application, and the latter is the installation and configuration of each machine by using the Quattor fabric management tool 1.

The high performance is guaranteed by the SAN that leverages Fibre Channel connections type. The SAN zoning configuration was built with one zone per HBA, where one HBA is linked with 2 target per storage controller. The load balancing and fail over layers are managed by the power-path devices drive that is an EMC2 proprietary software.

Quattor allows the testbed to be easily installed and differently configured in order to cover several deployment scenarios. Specifically, configurations per LHC experiment can be set up to reproduce the production environment.

5. Performed Tests
This testbed is used to perform YAMSS-GPFS-TSM, SRM ans globus-url-copy tests.

5.1. YAMSS-GPFS-TSM
The HSM system management data disk and tape is based on GPFS for the disk, TSM for the tape and a set of scripts home-made (called YAMSS) for the interfacing of GPFS and TSM.

The tests are divided on several levels:

- Tests involving writing and access files via StoRM to ensure right extended attribute (EA) and right behaviour of the system to set the additional EA and manage the data.
- Test of the functionality GPFS, in particular the scanning of the file system to create lists of data to be migrated to tape.
- TSM test capabilities: migration and ordered recall of file from disk to tape, and vice versa with tsm commands.
- Test YAMSS scripts and configurations to move data from disk to tape (yamssStubbyfication and yamssRecall)
- Stress testing of the system where it performs many writes files to disk, the checksum calculation, migration to tape with erasure of the physical file from disk, subsequent recall and checksum verification again.
- Testing of sensitive operations before they are executed in the production system, such as moving data between storage pools with different types of tape drives (e.g. from T10kA to T10kC).

To facilitate these operations we have been created in bash provided scripts to perform these tests and verify the accuracy of test results.

5.2. SRM and globus-url-copy
The SRM implementation adopted at INFN-CNAF is provided by StoRM[5]. Latest StoRM releases have been checked and validated through extensive stress and functionality tests performed on the testbed. These are performed by an automated and configurable python

1 Quattor, a fabric management for grid and cloud, http://quattor.sourceforge.net/
tool, running on a User Interface and cyclically submitting sequences of filetransfer (globus-url-copy) or clientSRM commands and triggering delivery of email alerts upon failure. While a single sequence of commands is useful for functional validation, a stress-test is obtained by continuously repeating the commands sequence, from several instances of the python tool. This way we can simulate high load situations on the StoRM Instances and verify stability and health status of the components, particularly on the BackEnd. Comparative test are possible for the FrontEnd component and they have been performed after installing two different FE variants.

The tool is able to perform filetransfer with `globus-url-copy` and almost all clientSRM commands, some which are briefly summarized below:

- **ls**, to list files on the directory specified by the provided SURL
- **rm**, to delete a file specified by the SURL
- **mkdir**, to create a directory specified by the SURL
- **PrepareToPut**: An asynchronous request to create a new filename on the storage. A token id is returned to the client. Upon completion, a TURL is available to the client. This is used by `globus-url-copy` at client side to actually transfer the file to the storage.
- **PutDone**: To close a PrepareToPut operation, thus declaring to StoRM that a filetransfer is done.
- **PrepareToGet**: An asynchronous request to copy the specified file from the storage.
- **ReleaseFile** to release a file previously subject of a PrepareToGet request.

To initialize a test session, a desired sequence of the above operations can be written in the configuration file. When starting the test, the sequence will be consecutively executed a number of times, with a sleep time (0 or more seconds) after each iteration. Null sleep time is used for stressests; positive sleep time is used for periodic health checks.

The `globus-url-copy` command is a GridFTP client available with the Globus Toolkit middleware distribution. Examples of GridFTP and GPFS performances are shown in [8].

### 6. Conclusions

The testbed has been realized with hardware out of production but it has proved very important to validate new versions of StoRM and YAMSS, ensuring the operation of new hardware and avoiding unexpected malfunction in the event of an upgrade of TSM and GPFS. It was also used to test procedures for moving data between different storage pools, avoiding unexpected behavior hardware or software in production. In the last use case, thanks to the testbed, we were able to move from technology T10kA to T10kC, making the upgrade tsm server in production with few working hours having already experiment with how to proceed. Besides the use of new drive later we moved the data contained in the old media in new ones. The testbed has become an essential tool for the administration of production systems.

### References

[1] Gregori D. et Al. *INFN-CNAF Activity in the TIER-1 and GRID for LHC Experiments*, Proceedings of IEEE International on Parallel & Distributed Processing Symposium (IPDPS), Rome, Italy, May 2009.
[2] Gregori D. et Al. *INFN-CNAF Tier-1 Storage and Data Management System for the LHC Experiments*, Proceedings of Computing in High Eenergy and Nuclear Physics (CHEP), 2010.
[3] http://www-03.ibm.com/systems/software/gpfs/
[4] http://www-01.ibm.com/software/tivoli/products/storage-mgr/
[5] Ronchieri E. et Al. *An efficient Grid data access with StoRM*, S.C. Lin and E. Yen (eds.), Data Driven e-Science: Use Cases and Successful Applications of Distributed Computing Infrastructure (ISGC 2010), Springer Science + Business Media, LLC 2011.
[6] Shoshani A. et Al. *Storage Resource Managers: Middleware Components for Grid Storage*, Proceedings of NASA CONFERENCE PUBLICATION, pp. 209-224, 2002.
[7] W. Allock, *GridFTP: Protocol Extensions to FTP for the Grid*, Global Grid Forum, 2003.
[8] Ronchieri E. et Al. *On Enhancing GridFTP and GPFS performances*, Proceedings of Computing in High Energy and Nuclear Physics (CHEP), 2009.