PROCEEDINGS OF SCIENCE

First sights on a non-grid end-user analysis model on Grid Infrastructure

Roberto Santinelli
CERN
E-mail: roberto.santinelli@cern.ch

Fabrizio Furano
CERN
E-mail: fabrzio.furano@cern.ch

Andrew Maier
CERN
E-mail: andrew.maier@cern.ch

Vinicius Bucard
CBPF
E-mail: bucard@cbpf.br

Renato Santana
CERN
E-mail: rsantana@cbpf.br

Unprecedented amount of data has started to come out of CERN's Large Hadron Collider (LHC). Large user communities demanding to access this data will arise in order to perform analysis. Despite the existence of Grid and distributed infrastructure, which allows a geographically distributed data mining and analysis, there will be, certainly, an important user analysis concentration activity, where data resides. That would nullify, in some extent, the grid paradigm itself. LHCb (Large Hadron Collider beauty) experiment computing model envisages data distribution only to selected centres, known as Tiers-1. Due to the storage capability, which is not infinite, none of the LHC experiment computing models predicts the accelerator's data distribution across all sites. The present work proposes a model which intends to copy data, on demand, from the main grid computing centres to storage facilities, at non Tier-1 centres, allowing to perform local analysis. This solution allows local Physics Institutes' communities to define their own priorities by running on their owned resources. It also allows reducing the risk of having crowded batch queues on remote systems (e.g. the LSF at CERN). In order to keep a consistent interface for the end-user analysis, in both LHCb and ATLAS user communities, some work have been done aiming to integrate it and customize Ganga interface, allowing one to submit local non-grid jobs. This paper, finally presents a first working prototype, proof of a new model concept for end-user analysis. It aims to take advantage of the potential storage in the Grid and allows local communities for an immediate end-user analysis over the LHC data.

13th International Workshop on Advanced Computing and Analysis Techniques in Physics Research - ACAT 2010
Jaipur, India
February 22–27 2010

1.Introduction

This T3 Analysis Facility project aims to exploit Tier 3 (non-grid) computing and storage facilities at Tier 2 sites, in order to run local analysis (see Figure 1).

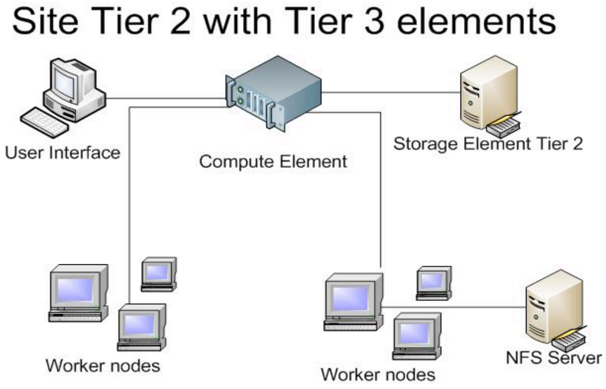


Figure 1: Overview of the T3 Analysis Facility

This concept will **guarantee** to local communities the desired precedence in accessing data and **to be better exploit** local resources. LHC experiments Computing Models did not envisage to run distributed analysis at small sites. These sites are not committed to provide an agreed level of Quality of Services (QoS). A preliminary experience with the first data taken in 2010, confirmed that an unpredictable load, coming from many users looking for their analysis to run, has stills to be properly addressed within the LHC collaborations.

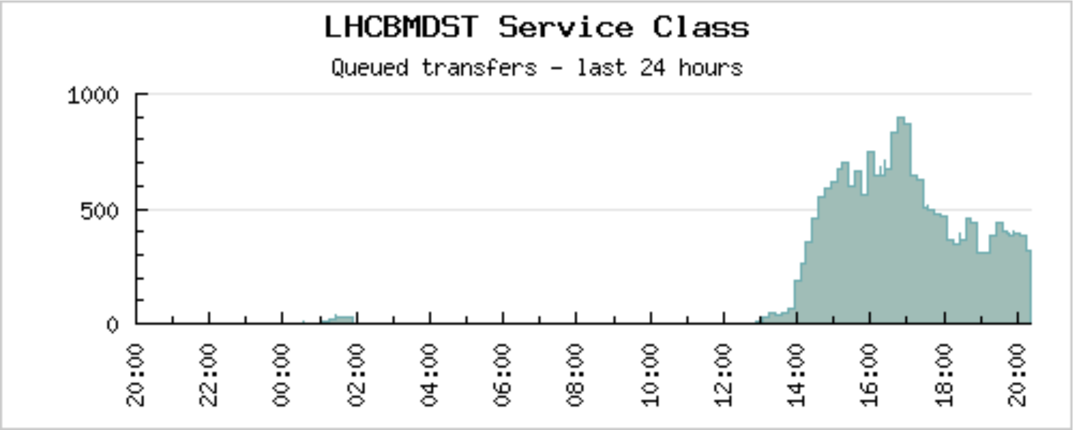


Figure 2: CERN service class load following first data available

Figure 2 shows the queued transfers number, on real data disk pool at CERN, for LHCb, in May 2010, when new data was taken. The number of disk servers available was not enough to sustain the load spike. The raise of many concurrent requests to access newly available data caused a

huge backlog of requests. That prevented further users to access data and, as a consequence, to run their analysis. The unresponsiveness of the overloaded storage was also affecting the production activities running in parallel.
It has been estimated that the total amount of spare storage available on the Grid, via non Tier-1 grid sites, might be comparable with the total T1 sites space available. The idea of this work is to exploit this potentiality offered by "other" sites. Grid site services and facilities are used to download data needed from the grid, by local physics groups. Once available on the site, the data is then used by the institute's scientific community to perform local analysis, using local resources. Ganga[4], the distributed analysis facility adopted by LHCb and ATLAS, allows users to choose from where they submit the job. It is possible to submit jobs from local cluster or from the Grid, without the need of changing job description. This brings the possibility of running jobs locally, on the site, without any major change at the user interface level. This is the main reason that it has been considered in this work.

2. LHCb Computing Model and the bench site

LHCb Computing Model distinguishes collaboration computing centers in three main categories:

- **Tier0** center: (part of CERN infrastructure), targeted for real data acquisition, first data reconstruction, data archive and distribution.
- **Tier1** (CERN+6) centers: used for data reconstruction/processing/reprocessing and collaboration analysis. There, Computing Elements and Storage Elements are used.
- **Tier2** centers: used exclusively for MC(Monte Carlo) generation. The is no use of Storage Service. Under some strict conditions, some T2 centers are entitled to be LHCb Analysis Center (LAC) to host the collaboration distributed analysis.

From this model it is clear that the services number and the availability required at T1's is much higher than at T2s. This LHCb approach was driven by the size of the collaboration and the consequent expected load on the infrastructure. Besides, there was the need of reducing services numbers to be run by sites, minimizing the load in infrastructure operations.
Recent experiences, also several years of data and service challenges, may induce that services stability, offered by large T1 centers, represents the main plague for a smooth system operation.

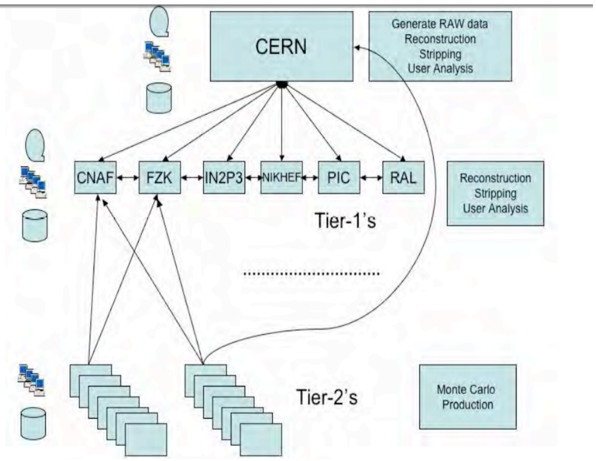


Figure 3: Use case: the LHCb Computing Model.

CBPF site is a medium (~400 cores) T2 Center, located at Rio de Janeiro (Brazil), supports both LHCb and CMS; it is used for MC production and has been also used to successfully host a first LHCb T3 analysis facility prototype.

The site Computing Element (CE) offers a grid interface to submit user payloads to the farm computing nodes via the local batch scheduler (LRMS). Both CREAM-CE and LCG-CE are currently supported and interfaced to PBS job scheduler.

The site Storage Element (SE) offers a uniform grid data access, in this case, based on the DPM (Disk Pool Manager) technology. GSIFTP protocol (a GSI-secure FTP) has been used to transfer files and the secure RFIO protocol to handle local and remote file access.

3. Ganga

Submitting jobs to the Grid is a non-trivial task for the average non-technical user. It requires the user to:

- Write a wrapper script to set up his job.
- Manage jobs inputs and outputs.
- Write a JDL to steer the jobs.
- Monitor jobs progress, success or failure.

Concurrently, Grid may not be the only computational resource available. Users may have access to a large local batch farm, or he might simply want to run his job locally, on his desktop machine for testing purposes. In all these cases, the way to run a job is slightly different and requires users to learn a multitude of commands and techniques, instead of being able to concentrate on the analysis he wants to perform. Ganga[4] provides a solution for the user to address all these issues. Starting off as common project of the ATLAS and LHCb collaborations and written in Python [5] it is freely available under the GNU[6] Public License. It is an easy to use front-end for the submission of computing jobs to a variety of computing resources such as the local machine, a variety of batch systems and various computing Grid flavours. The Ganga philosophy (see Figure [4]) is to try to make the job submission to different submission back-ends as transparent as possible. Ganga includes a plug-in system, which allows it to be extended both for submission backends and application plug-ins. While submission backends extend the possibility of where to send a job, application plug-ins allow simplified job submission of applications commonly used within a user community. Currently Ganga includes submission backends for batch systems, such as LSF, PBS, SGE and Condor, Grid systems such as LCG[7] and NorduGrid[8] and workload management systems (WMS), typically sitting on top of Grid middleware such as the PanDA[9] system by ATLAS and the DIRAC[10] system by LHCb. In addition Ganga is shipped with customised application handlers, such as the Root[11] handler and the experiment specific application handlers for Athena (ATLAS) and Gaudi (LHCb). A general catchall executable application handler exists, capable of handling any application. However, specialised application handlers can offload a lot of the work needed to write special wrapper scripts and to configure the application separately. This simplification is achieved by knowing how the application has to be run, how to set up the necessary environment and which inputs and outputs are to be expected.

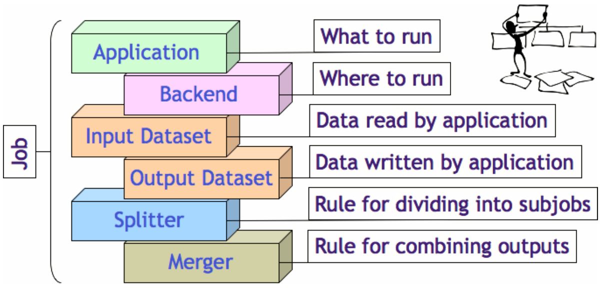


Figure 4. The concept of a Ganga job: users are required to specify at least the application to run and the backend, where to run.

4. Workflow

Physicists want to access files dataset, for analysis. Dataset usually consists of a file list, specified as LFNs (Logical Filenames). The LFC [3] (LCG File Catalog) is used to retrieve the physical replicas list of files and their location (data resolution). Once the LFC gives the replicas available to the Grid, the file is either copied down to the to storage element or replicated (Replicated means that the file is also registered in the file catalog and then potentially available to the whole community.**)**. Once the download to the local Grid SE finishes successfully, files are cached into the local NFS area. From this point, data is available on the local site. Local user jobs can be scheduled to use data, either locally, or through the grid. This is best achieved using Ganga, which hides the differences between local and grid jobs. The use of local NFS area as cache is just an option to bypass issues related with grid proxies management in non-grid resources. Indeed, the secure RFIO protocol used by DPM requires the user jobs running locally to have a valid X509 certificate available; ultimately this would have implied to modify the LRMS plug-in to also upload the certificate at local job submission time.

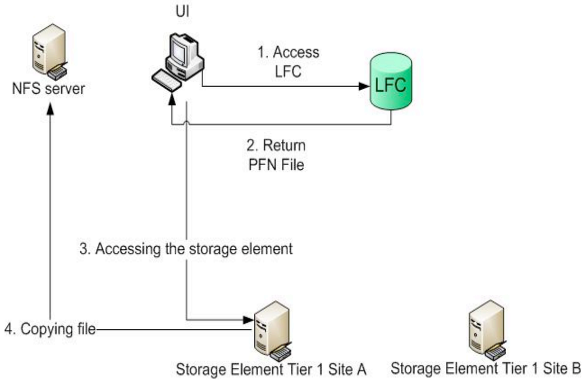
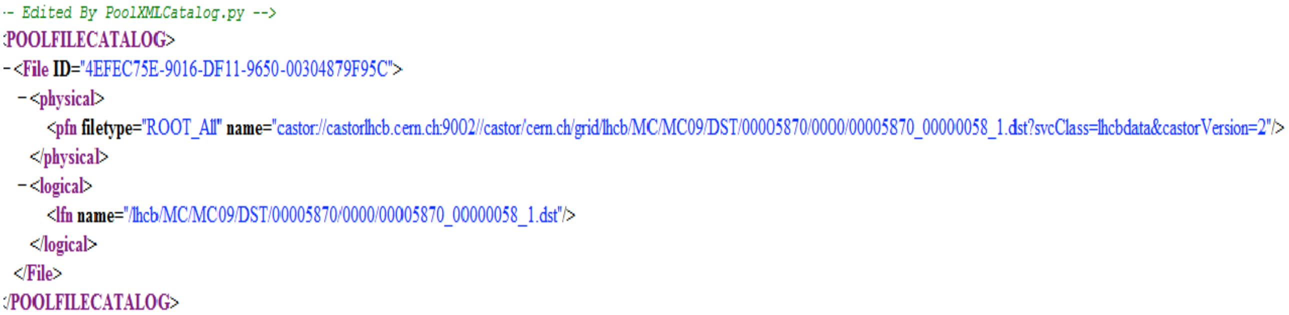

UI
1. Access LFC
LFC
NFS server
2. Return PFN File
3. Accessing the storage element
4. Copying file
Storage Element Tier 1 Site A
Storage Element Tier 1 Site B

Figure 5: The workflow

5. The XML Slice

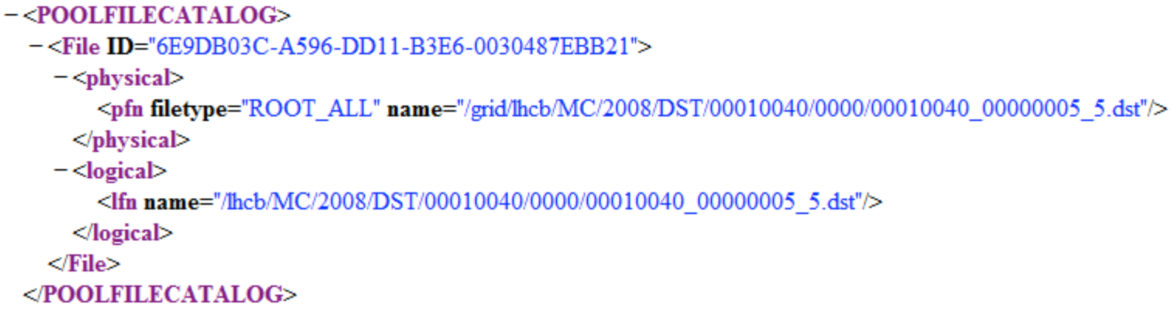
Ganga produces a XML slice to indicate file´s GUID, PFN and LFN. Therefore, through the PFN one can analyze grid files using their protocol to access data. The proposed project permits to use PFNs of the non-grid files using directly the path of their POSIX directories. This is already an exercised protocol being used in production through GPFS [6] at one of the LHCb T1 centers (CNAF).

This XML slice is generated at submission time, thus it was necessary to change it. Some routines (RTHUtils.py) - responsible to generate these XML files – have been overloaded for that reason and a working example of such XML file is available in figure 7.



```
-- Edited By PoolXMLCatalog.py -->
<POOLFILECATALOG>
- <File ID="4EFEC75E-9016-DF11-9650-00304879F95C">
  - <physical>
      <pfn filetype="ROOT_All" name="castor://castorlhcb.cern.ch:9002//castor/cern.ch/grid/lhcb/MC/MC09/DST/00005870/0000/00005870_00000058_1.dst?svcClass=lhcbdata&castorVersion=2"/>
    </physical>
  - <logical>
      <lfn name="/lhcb/MC/MC09/DST/00005870/0000/00005870_00000058_1.dst"/>
    </logical>
  </File>
</POOLFILECATALOG>
```

Figure 6. Example of XML slice for a Ganga job running at CERN via Grid



```
- <POOLFILECATALOG>
  - <File ID="6E9DB03C-A596-DD11-B3E6-0030487EBB21">
    - <physical>
        <pfn filetype="ROOT_ALL" name="/grid/lhcb/MC/2008/DST/00010040/0000/00010040_00000005_5.dst"/>
      </physical>
    - <logical>
        <lfn name="/lhcb/MC/2008/DST/00010040/0000/00010040_00000005_5.dst"/>
      </logical>
    </File>
  </POOLFILECATALOG>
```

Figure 7: Example of a modified XML for a job running on the local T3 facility at CBPF

6. Conclusion

This work is a proof of a concept to demonstrate that analysis at T3 centers is feasible just using tools and facilities currently in use for the typical distributed grid analysis. Changing few routines in the Ganga layer and writing a very tiny layer for data downloading and caching makes possible to run different users' analysis via resources, not visible through the Grid. Various teams, within the four LHC experiments, are now working on this subject, having in mind the importance that spare, otherwise unused, resources in the Grid may introduce a non negligible contribution to sustain the load that the analysis and the reconstruction of an unprecedented amount of data implies. On a more broad view, this paper allows grid space optimization, allowing unused data space to become useful.

References

[1] LHCb Computing Model, https://twiki.cern.ch/twiki/bin/view/LHCb/ComputingModel

[2] *EIS team*, Glite User Guide, https://edms.cern.ch/file/722398/1.3/gLite-3-UserGuide.html

[3] LFC service https://twiki.cern.ch/twiki/bin/view/LCG/LfcWlcg.

[4] *J.T. Moscicki, F. Brochu, J. Ebke, U. Egede, J. Elmsheuser, K. Harrison, R.W.L. Jones, H.C. Lee, D. Liko, A. Maier, A. Muraru, G.N. Patrick, K. Pajchel, W. Reece, B.H. Samset, M.W. Slater, A. Soroko, C.L. Tan, D.C. van der Ster, M. Williams.* , Ganga: A tool for computational-task management and easy access to Grid resources Computer Physics Communications, Volume 180, Issue 11, November 2009, Pages 2303-2316, ISSN 0010-4655, DOI: 10.1016/j.cpc.2009.06.016

[5] *The Python scripting language* http://www.python.org

[6] The GNU General Public License, http://www.gnu.org/licenses/gpl.html

[7] Worldwide LHC Computing Grid, http://www.cern.ch/LHCgrid

[8] *M. Ellert et al.,* "Advanced Resource Connector middleware for lightweight computational Grids", Future Generation Computer Systems (2007) 23.

[9] *T Maeno.* PanDA: distributed production and distributed analysis system for ATLAS. 2008 J. Phys.: Conf. Ser. 119 062036 (4pp) doi: 10.1088/1742-6596/119/6/062036

[10] *A.Tsaregorodtsev et al*., "DIRAC, The LHCb Data Production and Distributed Analysis System", Proceedings from CHEP06, 2006.

[11] *Rene Brun and Fons Rademakers*, "ROOT - An Object Oriented Data Analysis Framework", Proceeding AIHENP'96 Workshop, Lausanne, Sep. 1996, Nuclear Ins. Methods Phys. Res., A389