A Comparison of Data-Access Platforms for BaBar and ALICE analysis Computing Model at the Italian Tier1

A Fella¹, F Furano², L Li Gioi¹, F Noferini¹, M Steinke¹, D Andreotti³, A Cavalli¹, A Chierici¹, L dell’Agnello¹, D Gregori¹, A Italiano¹, E Luppi³, B Martelli¹, A Prosperini¹, P Ricci¹, E Ronchieri¹, D Salomoni¹, V Sapunenko¹ and D Vitlaci¹

¹ INFN-CNAF Bologna, Italy
² Conseil Europeen Recherche Nucl. (CERN)
³ INFN Ferrara Ferrara, Italy
⁴ RUHR-UNIVERSITÄT BOCHUM

E-mail: armando.fella@cnaf.infn.it, luigi.ligioi@cnaf.infn.it, francesco.noferini@bo.infn.it

Abstract. Performance, reliability and scalability in data access are key issues in the context of Grid computing and High Energy Physics (HEP) data analysis. We present the technical details and the results of a large scale validation and performance measurement achieved at the CNAF Tier1, the central computing facility of the Italian National Institute for Nuclear Research (INFN). The aim of this work is the evaluation of data access activity during analysis tasks within BaBar and ALICE computing models against two of the most used data handling systems in HEP scenario: GPFS and Scalla/Xrootd.

1. Introduction

Data handling and file access systems are key topics in distributed computing model as it is the Grid context in which the work presented is developed. The High Energy Physics (HEP) community, facing the challenges of the new scientific program starting at the Large Hadron Collider (LHC), needs to provide the best choice in terms of reliability, availability and performance for high throughput data services. The study of data handling systems in real analysis use case is one of the main work-in-progress at the moment we are writing.

The test session took place at INFN-CNAF Italian Tier-1 computing centre and it is designed to be representative of typical HEP experiment in production status. It consists of a comparative evaluation of two data-access solutions in use at Tier-1 and Tier-2 centres, namely General Parallel File System (GPFS) [1] and the Scalla/Xrootd system [2]. At the time of writing GPFS is an IBM product distributed under license, whereas Scalla/Xrootd is an open source platform free of license fees. Both BaBar [3] and ALICE [4] computing models are based on Scalla/Xrootd data access system. On August 2007 the BaBar choice at CNAF was to move from Scalla/Xrootd to GPFS data access solution: a comparison of the two data access models was performed via a complete test session on September 2008 [5]. We cite also the first work on data access platform comparison performed at the same centre [6].
The **ALICE** interest in adopting Scalla/Xrootd as data access at Tier1 scale brought us to develop a test design being able to describe in deep the data access systems performance. This will provide useful information for enabling **ALICE** at CNAF. We used real physics analysis running on real physics datasets in the **BaBar** case and on fully simulated dataset in the **ALICE** case.

Since 2004 **BaBar** collaboration includes INFN-CNAF Tier-1 in its distributed computing model moving a subset of physics data and the related analysis effort from SLAC to Italy. Moreover, the Italian Tier-1 represents an important node in the **ALICE** computing model in particular for the Italian component of the collaboration which is strongly involved in the physics program of the experiment.

Unlike the September 2008 test in which the measurements were performed in a controlled environment, the work we present shows the results of a test in a pure production scenario. This choice permits to study the data access systems behavior by measurements involving farm heterogeneity in terms of worker nodes architectures and jobs differentiation.

### 2. Test-Bed Layout

A sketch of the test-bed is depicted in Fig. 1. In our setup, 8 disk-servers were integrated into a Storage Area Network (SAN) via FC links, while the communication of the disk-servers with the computing nodes was provided via Gigabit LAN. As disk-storage hardware we used 2 EMC CX3-80 SAN systems, with a total of 160 TB of raw-disk space, assembled by aggregating 1-TB SATA disks. The EMC system comprised two storage controllers (with 8 GB of RAM each),

---

**Figure 1.** Schematic representation of the test-bed layout.
connected to a Brocade 48000 Fibre Channel fabric director provided with 8 links, for a total theoretical bandwidth of 32 Gb/s.

The 8 Dell PowerEdge M600 disk-servers were equipped with Xeon 2.33 GHz quad-core processors, 16 GB of RAM, dual-port Qlogic 2432 4Gb Fibre Channel to PCI Express Host Bus Adapter (HBA) and 1 Gigabit Ethernet link. As an Operating System (OS) they run Scientific Linux CERN (SLC) version 4.6, with a 2.6.9-67.0.15.EL.cernsmp kernel operating at 64 bits.

The whole farm services and infrastructure were shared with the usual Tier1 activity including worker nodes job slots, DNS, LDAP Authentication, Network and local batch system managed by Load Sharing Facility (LSF). The worker node job allocation is ruled by an overloading policy that permits to run 1.2 jobs per core. For these tests a specific LSF queue was configured to be able to exploit computational resources equal to the sum of the 80% of the ALICE and BaBar ones. In this scenario we can successfully manage synchronous job starts working into the CNAF production submission environment. The CNAF farm is composed by the following Worker Node (WN) hardware architectures with the associated occurrences: Woodcrest 200, Intel Xeon E5420 (70%), Woodcrest 266, Intel Xeon E5440 (20%), AMD Opteron 252 (10%). The Operating System of whole WNs was Scientific Linux CERN 4.5 with i386 kernel 2.6.9-67.0.15.EL.cernsmp. Each node was connected to a Gigabit switch, one switch per rack, and each switch had a 2 x 1 Gb/s trunked up-link to the core network switch.

2.1. Monitoring Tools
In order to monitor the actual throughput of the various components of the system during the tests, both the network and disk ones, we used several products, hence cross-checking in a coherent view all the information available from different sources: MRTG was used in order to monitor the network throughput as seen by the Gigabit network switches, Lemon [7] system was used to monitor the network interfaces of the disk-servers and the throughput to/from the back-end storage systems. Every 100 seconds, read/write throughput information were generated by LeMon system and stored in its data base component. A C++ utility based on ROOT [8] was developed to process these information and generate graphs.

3. Data-Access Platforms
GPFS is a general purpose distributed file-system developed by IBM. It provides file-system services to parallel and serial applications. GPFS allows parallel applications to simultaneously access the same files in a concurrent way, ensuring the global coherence, from any node which has the GPFS filesystem locally mounted. GPFS is particularly appropriate in an environment where the aggregate I/O peak exceeds the capability of a single file-system server. Differently from the other solution employed in our tests, from the point of view of the applications, the file-system is accessed by a standard POSIX interface, without the need of compiling the client code with ad-hoc libraries. In our tests we use the production BaBar GPFS cluster, specifically we work against 10 disk-servers: 8 data servers and 2 cluster admin node, aggregating all the SAN LUNs available in one global file-system. The GPFS version adopted was 3.2.1-4.

The Scalla/Xrootd platform is a pure disk data handling system developed as a collaboration between SLAC and INFN, with some other contributors in the HEP software community, in specific the last two releases was developed by ALICE experiment. Scalla/Xrootd is designed to provide fault tolerant location and access to files distributed throughout cluster and WAN environment by employing peer-to-peer-like mechanisms. The architecture provided for this test was based on the last Scalla/Xrootd production version: 20090206.1632, and notably included two admin nodes (also known as redirectors) and one Scalla/Xrootd server on each of the 8 disk-servers. The Scalla/Xrootd disk back-end is composed by SAN LUNs XFS formatted. A new BaBar software release was deployed ad-hoc to permit a well-advised and coherent comparison between the two experiments test results: the client side software for both the experiments
included the use of ROOT 5/21-06. A single test run was performed using the *BaBar* typical deployment based on ROOT 5/14-00e to obtain client side comparison information.

The hybrid solution Scalla/Rootd with GPFS backend (Scalla/Rootd over GPFS in the following) was tested too. In this case data files were stored all in one GPFS file system mounted on all the Scalla/Xrootd managed disk servers.

### 3.1. Scalla/Xrootd setup

The data used in the *ALICE* analysis were imported at CNAF via Scalla/Xrootd protocol in order to register them in the Alien catalog [9] and subsequently replicated on the GPFS filesystem. AliEn is a Grid framework developed and adopted by the *ALICE* collaboration that provides an interface to all Grid services including the data access authentication through Scalla/Xrootd protocol. The *ALICE* data handling model include a central Scalla/Xrootd server, named global redirector, working as an upper level file location broker. The Scalla/Xrootd package in the *BaBar* runs is the same as the *ALICE* one apart from the configuration that was modified to be as similar as possible to the *BaBar* computing model. Table 1 reports the detailed configuration parameters for both experiments. *ALICE* parameters have been assigned standard values obtained by previous optimizations.

| Experiments | Monitor system | Outbound service | Read-ahead [KB] | Cache [MB] |
|-------------|----------------|------------------|-----------------|------------|
| *ALICE*     | Monalisa       | Global redirector| 500             | 10         |
| *BaBar*     | No             | No               | 0               | 0          |

### 4. Data Analysis

We performed a real physics data analysis to evaluate the data access behavior in a real production scenario using the analysis framework of the *ALICE* and *BaBar* experiments [11, 3]. The test consisted in running several bunches of analysis jobs each one accessing different input files.

The *ALICE* analysis was performed on a set of about 55k of *PbPb* events from the *ALICE* Monte Carlo production 2007 corresponding to $\sim 2.6$ TB of disk space. These events were produced by detector response simulation in order to provide a data structure as possible close to the future real data. All the data set was analyzed through 1000 parallel jobs; so the data were split in 1000 homogenous collections of files. The analysis was also repeated with a smaller number of jobs (700). The analysis performed on the data was a typical two particles correlation study. The software used in the analysis is AliRoot [10] that is a ROOT extension developed by *ALICE* collaboration. The job authentication phase is managed by AliEn during the data access process via Scalla/Xrootd protocol.

The *BaBar* jobs run on files containing data collected by the *BaBar* detector. Each job analyze 170k physics events, searching the occurrence of a $D^{*+} \rightarrow [K_S\pi^-\pi^+]|D\pi^+$ decay in each event, with a procedure consisting in the initial event reading followed by the execution of the proper decay selection algorithm. In deep the process parses portions of file related to the event information the specific analysis it is interested in. We started a run consisting of 1000 synchronized jobs analyzing $\sim 1.2$ TB of data.

Data files managed by the Scalla/Xrootd system were spread uniformly over all the disk partitions served by the related disk-servers.
4.1. Execution Time

One of the most useful results to compare the different data access solutions is the job duration. Fig. 2 and Fig. 3 show the execution time distribution for 700 and 1000 ALICE parallel jobs. Since run 700 reads 87% amount of data then run 1000, the job duration is different within the two runs: run 700 is 20% longer then run 1000. Since we run the tests in a real production scenario no difference in the performance between Scalla/Xrootd and GPFS can be estimated. We conclude that the behavior of the two data access solutions was the same during the two ALICE runs. The run with Scalla/Xrootd over GPFS instead appears to suffer of a kind of data access system overlapping effect that is the cause of the raising of two peaks shape. Fig. 4 shows the execution time distribution for 1000 BaBar parallel jobs. The slightly better performance of GPFS is compatible with the one observed during the September 2008 test session. The two peaks shape in the Scalla/Xrootd case is due to an unlucky distribution of the jobs over the different worker nodes architectures. No data access system overlapping effect is present in the case of Scalla/Xrootd over GPFS. Fig. 5 shows the same distributions for the BaBar runs with 2 different versions of the Scalla/Xrootd client: root 5/21-06 and root 5/14-00e. We saw no difference in the behavior of the two Scalla/Xrootd clients except for the different distribution within the worker nodes architectures.

**Figure 2.** Execution time distribution for 700 ALICE parallel jobs.

**Figure 3.** Execution time distribution for 1000 ALICE parallel jobs.

**Figure 4.** Execution time distribution for 1000 BaBar parallel jobs.

**Figure 5.** Execution time distribution for 1000 BaBar parallel jobs with two different Scalla/Xrootd client version: root 5/21-06 and root 5/14-00e.
4.2. Worker nodes hardware architecture

In order to understand better the effect of the production environment, a detailed study has been performed separately for the different worker nodes architecture types. In Fig. 6 and Fig. 9 the fraction of completed jobs as a function of the execution time are shown for ALICE and BaBar runs respectively.

**Figure 6.** Fastest WN architecture (ALICE 700 jobs).

**Figure 7.** Main WN architecture (ALICE 700 jobs).

**Figure 8.** Slowest WN architecture (ALICE 700 jobs).

During the ALICE runs, the three tested solutions perform in a very similar way for all the hardware architectures. The Scalla/Xrootd system seems to take advance of most recent hardware architecture.

**Figure 9.** Fastest WN architecture (BaBar 1000 jobs).

**Figure 10.** Main WN architecture (BaBar 1000 jobs).

**Figure 11.** Slowest WN architecture (BaBar 1000 jobs).

The difference between the ALICE and BaBar computing models and in particular the client side prefetch parameters (Tab. 1) impacts Scalla/Xrootd performances: in fact in the BaBar case, as shown in Fig. 9 GPFS performs better then Scalla/Xrootd for all the architectures. These results are compatible with the ones showed in Fig. 12 and Fig. 13 where the same issue is treated considering the WallClock time. In these plots we consider only the case in which the same job, insisting to the same data collections, runs on the same worker node architecture. Fig. 13 shows a performance worsening of Scalla/Xrootd over GPFS with respect the Scalla/Xrootd pure run. The corresponding BaBar case showed in Fig. 14 and Fig. 15 confirm the GPFS overcame in performance with respect Scalla/Xrootd. The Scalla/Xrootd run suffers of a suspect production environment interaction (Fig. 10). In Fig. 15 the case of Scalla/Xrootd over GPFS is 10% slower then the GPFS one.

The results just commented are confirmed by the CPU time analysis presented in Fig. 16 and Fig. 17 for the ALICE runs. The focusing on CPU time reveals the Scalla/Xrootd and GPFS
client side comparison: the Scalla/Xrootd client side appears to be more CPU time consuming then the GPFS one, the hybrid case Scalla/Xrootd over GPFS is instead clearly more CPU time consuming then the other two.

4.3. Throughput
The second useful result to compare the different data access solutions is the aggregated throughput recorded at the Ethernet card of the disk-servers. Since we configured differently the Scalla/Xrootd client we expect to have different results between ALICE and BaBar.

4.3.1. ALICE results  The ALICE aggregated throughput as a function of time is shown in Fig. 18 for the 1000 jobs run and in Fig. 19 for the 700 jobs run. As it is shown in Fig. 22, the integral over the time of the aggregated throughput is the same for all data access solutions in the 700 jobs run. In the case of 1000 parallel jobs it is a little bit smaller for the Scalla/Xrootd
4.3.2. BaBar results

The plateau and the integral values over the time of the aggregated throughput using Scala/Xrootd, as it is shown in Fig. 22, appears to be less than 50% of the GPFS ones, performing exactly the same jobs in the two scenarios. This behavior is due to the distinct prefetch configurations for the two systems. This explains also the different shape of the aggregated throughput as a function of time shown in Fig. 20. At the beginning of the run an initial peak is visible in both cases but with different magnitude. The Scala/Xrootd peak is due to job accessing in a contemporary manner to the detector condition database at starting time. This effect appears also in the GPFS case, but here a large initial read-ahead is also present. After the peak the graph enters in a stationary region. The different tails in the last region of the throughput graph can be explained with the different distribution of the jobs over GPFS run. This behavior is affected by a larger number of job failure in this run: 6% respect to the null jobs failure of the other ones. The different jobs failure is also one of the cause of the lower plateau throughput of the Scala/Xrootd over GPFS run. The other reason is the higher fraction of jobs running on the slowest worker node architecture (20%).

Figure 16. The job Cpu Time duration difference distribution, per Worker Node architecture, between GPFS and Scala/Xrootd runs of 1000 ALICE jobs.

Figure 17. The job Cpu Time duration difference distribution, per Worker Node architecture, between Scala/Xrootd over GPFS and Scala/Xrootd runs of 1000 ALICE jobs.
within the worker node architectures. This reflects also the different job duration distribution shape. During the Scalla/Xrootd over GPFS run the Scalla/Xrootd prefetch policy deal the throughput, so the shape is very close to the Scalla/Xrootd one.

Fig. 21 shows the aggregated throughput as a function of time in the case of 1000 BaBar parallel analysis jobs using two Scalla/Xrootd client versions: root 5/21-06 and root 5/14-00e. We saw no difference in the behavior of the two Scalla/Xrootd clients except from the different distribution within the worker nodes architectures.

4.3.3. Throughput summary The summary of integrated throughput and plateau values for all runs is shown in Fig. 22. All values are normalized to the Scalla/Xrootd runs. This plot shows clearly the effect of the different Scalla/Xrootd prefetch algorithms between ALICE and BaBar. Setting to zero the read-ahead value, BaBar uses about one half of the bandwidth respect to GPFS. The throughput curve of the Scalla/Xrootd over GPFS run results very close to the Scalla/Xrootd one (the measures are based on data recorded disk-server side). We saw no difference in the throughput behavior due to the updated Scalla/Xrootd client version except from the different distributions within the worker nodes architectures. Otherwise, with the default prefetch settings, ALICE bandwidth usage is quite the same in all cases except for the Scalla/Xrootd over GPFS, 1000 ALICE jobs which is affected by a 6% of failure rate.

5. Conclusions We performed a comparison between two of the most used data access systems in HEP: GPFS and Scalla/Xrootd, addressing relevant questions such as performance and reliability. We used large scale test-bed, composed of one EMC CX3-80 storage system providing 160 TB of raw-disk space via a SAN, 12 Gigabit disk-servers. We tested three types of access patterns, for HEP data analysis workloads, giving this way a useful view of the actual capabilities of the systems in a real production scenario.

The test was performed in a production environment so we faced with difficulties proper of this scenario: heterogeneous worker node architectures, production and test jobs contemporary running on the same client node and sharing base services (eg. network and authentication). However this permitted to collect information about Tier1 and Tier2 representative working scale. Both appeared to be a valid data access system for HEP environment especially looking at the results obtained with new generation worker nodes hardware architecture.
The behavior of the two data access platforms concerning the job duration was the same during the two ALICE runs. The small observed differences were due to the production scenario characteristics. Otherwise the Scalla/Xrootd over GPFS run appears to suffer of a kind of data access systems overlapping effect. During the BaBar runs we observed a slightly better performance running on GPFS that is compatible with the results we obtained during the September 2008 test session and we saw no overlapping effects in the Scalla/Xrootd over GPFS run.

The different Scalla/Xrootd setting of the prefetch algorithms between ALICE and BaBar results in different bandwidth usage respect to GPFS: setting to zero the read-ahead value, BaBar uses about one half of the bandwidth. ALICE bandwidth usage instead is quite the same in all cases.

We found that Scalla/Xrood is more CPU consuming than GPFS in both client and server side. The latter presents I/O wait values of 20% higher the almost null GPFS ones.

BaBar computing model performance was not affected by the use of different client side ROOT version: 5/14-00e and 5/21-06 reported the same behavior during both job duration and throughput measurements.

References
[1] General Parallel File System Documentation [Online]. Available: http://publib.boulder.ibm.com/infocenter/clresctr/vxrx/topic/com.ibm.cluster.gpfs.doc/gpfsbooks.html
[2] A. Dorigo, P. Elmer, F. Furano, and A. Hanusevsky, in WSEAS Trans. Comput., Apr. 2005.
[3] Cosmo G, Nuclear Physics B - Proceedings Supplements, Volume 78, Number 1, August 1999 , pp. 732-737(6)
[4] P. Saiz, L. Aphecetche, P. Buncic, R. Piskac, J. E. Revsbech and V. Sego [ALICE Collaboration], Nucl. Instrum. Meth. A 502, 437 (2003).
[5] A. Fella, L. Li Gioi, D. Andreotti, E. Luppi, V. Sapunenko, 2008 IEEE Nuclear Science Symposium Conference Record N29-8 1984 (2008)
[6] M. Benevenni et al., IEEE Transactions on Nuclear Science (June 2008) Volume: 55, Issue: 3, Part 3, pp. 1621-1630
[7] G. Cancio et al., in Proc. Computing in High Energy Physics (CHEP04), Interlaken, Switzerland, 2004, CD-ROM.
[8] ROOT [Online]. Available: http://root.cern.ch
[9] P. Saiz et al., Nucl. Instrum. Meth. A 502 2003 437440
[10] ALICE ROOT [Online]. Available: http://aliceinfo.cern.ch/Offline/AliRoot/Manual.html
[11] The ALICE Offline Bible [Online]. Available: http://aliceinfo.cern.ch/export/sites/AlicePortal/Offline/galleries/Download/OfflineDownload/OfflineBible.pdf