Available on CMS information server

CMS NOTE 2006/087

The Compact Muon Solenoid Experiment

CMS Note

Mailing address: CMS CERN, CH-1211 GENEVA 23, Switzerland

LCG Service Challenge 3 at the Spanish Tier-1 and Tier-2 sites

J.M. Hernández, F.J. Rodríguez-Calonge

CIEMAT, Madrid, Spain

F. Martínez, G. Merino

PIC, Barcelona, Spain

Abstract

This note describes the participation of the Spanish centres PIC and CIEMAT in the LCG Service Challenge 3 as Tier-1 and Tier-2 sites respectively. Data transfer, job submission and data throughput from mass storage to the data processing jobs have been successfully exercised at the desired level. Very valuable experience has been gained running the complex computing system under realistic conditions at a significant scale.

1 Introduction

The LCG Service Challenge 3 (SC3) [1] took place during the last half of 2005. It was regarded by CMS as an integration test exercising the bulk data processing part of the CMS computing model under realistic conditions. It focused on validation of the data storage, transfer and data serving infrastructure together with the required workload components for job submission and tracking. In addition, full CMS software stack required by transfers, data publication and processing jobs was used. SC3 was divided in two phases: a first data throughput phase during July 2005 to demonstrate data transfer between sites at maximum rate followed by a service phase, that took place between September and November 2005, where computing services were exercised in parallel with data transfers at a reduced rate.

The original goals of SC3 were the following:

- Structured data flow executing the CMS computing model: data distribution from CERN Tier-0 to Tier-1 centres; skimmed data routed down to Tier-2 sites; data transfers from Tier-2 sites to the associated Tier-1 resulting from Monte Carlo production.

- Mass data processing concurrently with data transfers: automatic data publication at sites as data arrive making data immediately available for analysis; run skimming applications at the Tier-1 centres and Monte Carlo production at the Tier-2 sites; access data from storage demonstrating the desired aggregate read throughput.

These initial goals were revised in light of the experiences from the throughput phase. It was realized that some of the components were not sufficiently prepared to conduct meaningful tests in a challenge environment. It was considered important not to perform artificial tasks during the service phase just to meet the original goals. Monte Carlo production at the Tier-2 sites with the subsequent transfer of the produced data to the associated Tier-1 was de-scoped. In addition, skimming applications running at the Tier-1 sites were replaced by simple analysis jobs.

Tested, integrated and commissioned components during SC3 became production services. SC3 was considered an important step for subsequent computing challenges in 2006, LCG Service Challenge 4 and the CMS Computing, Software and Analysis challenge (CSA06) scheduled to start on September 2006.

The Spanish Tier-1 and Tier-2 centres participated in SC3. The Tier-1, PIC, is located in Barcelona while the Tier-2 is a federation of two sites, CIEMAT in Madrid and IFCA in Santander. In SC3 only the CIEMAT part of the federated Tier-2 was active.

2 Configuration

The setup deployed for SC3 was a combination of dedicated hardware and services with already installed production computing resources. A dedicated storage system was set up at PIC together with SC3 instances of the storage services. Network and compute resources were shared with production activities. The production Workload Management System was used. There was neither a dedicated Computing Element (CE) nor a dedicated group of worker nodes. SC3 jobs did not have higher priority. At CIEMAT only dedicated transfer agents and data catalogues were instantiated. All computing resources, including CPU, network and storage, were shared with production.

2.1 Hardware setup

Figure 1 shows the bandwidth and approximate latencies (half of the round trip time) of the different network sections between CERN and the Spanish sites. The traffic from CERN reaches Spain via France through the Geant network infrastructure which has a bandwidth of 10 Gbps. In Spain, Rediris, the Spanish academic network, transports the traffic with a bandwidth of 2.5 Gbps. In Catalonia, the Anella Cientifica network interconnects academic and research centres. The bandwidth available to PIC is currently limited to 1 Gbps. The total latency CERN-PIC is about 24 msec. PIC-CIEMAT latency is 9 msec.

10 Gbps links are now being deployed in Spain as part of the Geant-2 infrastructure. In particular there will be a new 10 Gbps dedicated link between Madrid and Geneva.

At PIC, a dedicated CASTOR [2] storage system was set up for SC3. It consisted of a dedicated CASTOR stager, name server and file servers together with disk and tape pools. Disk and tape pools had an aggregate capacity of 15 TB each. A single SRM [3] entry point, configured as a DNS round-robin of 4 SRM servers, was deployed. Two

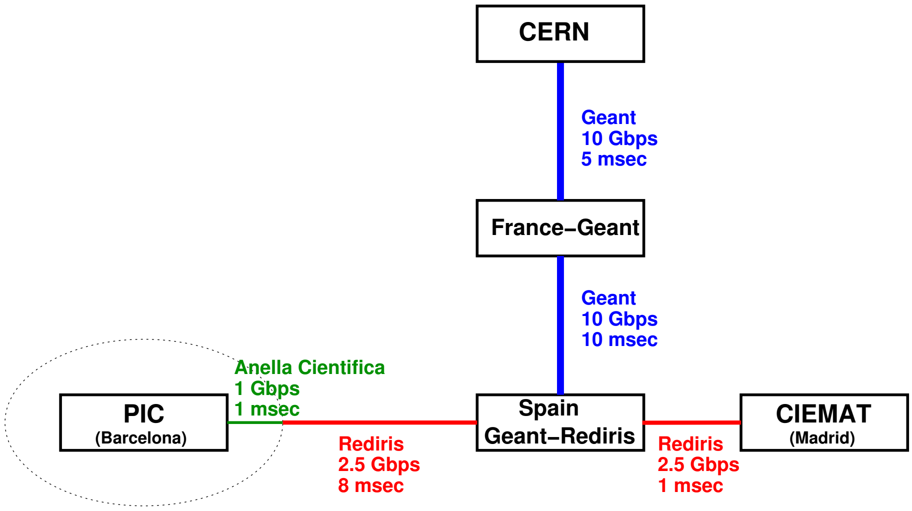

Figure 1: Network diagram showing the bandwidth and approximate latencies (half of the round trip time) of the different network sections between CERN and the Spanish sites.

different CASTOR disk pools were configured (see fig. 2). An import/export disk pool, built up with redundant and high-performance disks, was set up for accepting data from CERN, exporting data to CIEMAT and running data publication jobs which require access to the event data (see section 2.2). In addition, an analysis pool, highly distributed in about 50 WNs in order to achieve high aggregate read performance, was deployed for data analysis jobs. This pool is feeded with data from tape.

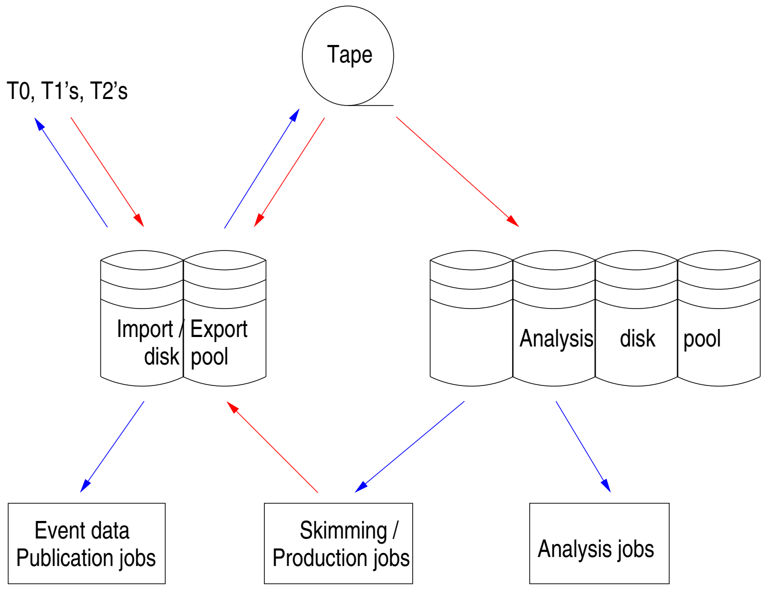

Figure 2: Configuration of storage resources at PIC for SC3

No CPU resources were reserved for SC3. SC3 jobs shared the 160 CPUs (about 200 kSI2k) available at PIC with CMS production jobs and other virtual organizations.

At CIEMAT, the CMS production CASTOR system was used. It consisted of a two-node DNS round-robin SRM entry point and a single disk pool of 5 TB. 120 worker nodes (about 120 kSI2k) were available for SC3, shared with CMS production computing activities.

2.2 Computing services and workflow

Figure 3 shows the CMS computing services used in SC3 together with an indication of the data- and workflows. The CMS transfer system, PhEDEx [4], consists of a central database, the Transfer Management Database (TMDB) a set of software agents running at each site and local site file catalogues. Local PhEDEx agents interact with the local mass storage system. In SC3 data were distributed from the Tier-0 at CERN to the Tier-1 centres and from each Tier-1 further down to the associated Tier-2 centres. Data were placed in Storage Elements (SE) and archived to tape at the Tier-1 sites. SEs are interfaced to the underlying storage technology through data storage managers like dCache [5], CASTOR or DPM [6], which provide a global file naming space and a transparent access to the data. At PIC and at CIEMAT we used CASTOR. Data were distributed as 2 GB zip file archives, containing several event data files, to minimize the number of files and improve performance of network and storage systems. Data were logically grouped in datasets each dataset consisting of three data tiers (hit, digi, DST) resulting from different Monte Carlo processing steps (simulation, digitization and reconstruction). Data tiers were distributed separately.

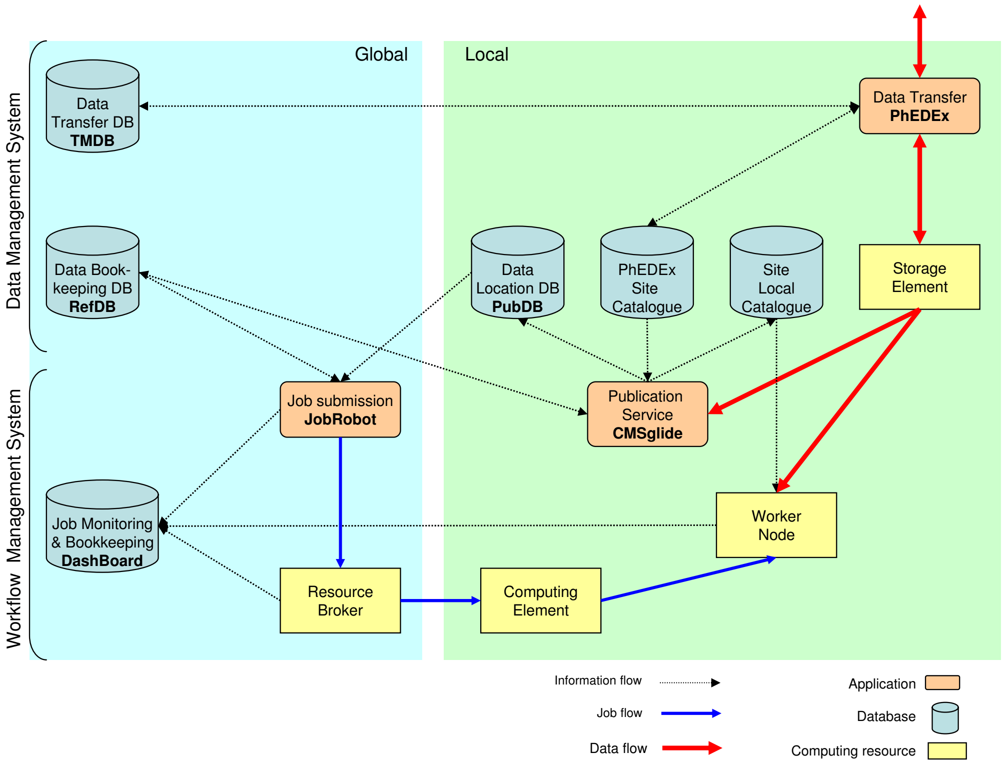

Figure 3: CMS services, data- and workflows during SC3. The picture indicates which services and computing resources belong to the Data Management and to the Workload Management System. It is also indicated whether these components run globally or locally at the sites.

Once a given data collection (all files of a data tier in a dataset) is completely downloaded, it has to be *published* so that it can be analyzed. The data publication procedure (its worflow is sketched in fig. 3) creates metadata files from the event data and generates XML POOL [7] file catalogues with local Physical File Names (PFN), Logical File Names (LFN), global unique file identifiers (GUID) and other POOL file attributes. The publication service CMSglide [8] carries out the data publication, getting from the central bookkeeping database (RefDB [9]) information about the collection being published, extracting from the local PhEDEx catalogue the PFNs of the files locally available for that collection, reading the event data from the local SE, creating local XML POOL file catalogues and finally flagging the published collection as available for analysis in the site data location database (PubDB [10]) and announcing its availability in the central RefDB.

Data publication is automatically triggered by a local PhEDEx agent. When a complete collection has been downloaded, the agent executes a local publication script which in turn invokes the publication service. Publication jobs were submitted to the local batch system in order to parallelize publication, which is an I/O and time expensive operation. After data publication, a data pre-stage job is issued by the publication agent to populate the analysis disk pool with the published event data from tape. Only when the whole event collection is staged on the analysis disk pool, the collection is announced in RefDB as available for analysis. This is done to avoid inefficient data pre-staging triggered by analysis jobs. Unorganized and atomic (file-by-file) data pre-stage is known to be quite inefficient. For SC3 the size of the analysis pool was big enough to hold on disk all samples. Triggering CASTOR garbage collection, trashing the data on the cache pool would result in an inefficient data pre-staging triggered by jobs trying to read data missing from the disk pool.

In SC3 a central job load generator (JobRobot) was run to continuously submit analysis jobs to the sites (see fig. 3).

Experiment software was preinstalled at the sites and software tags were published in the Grid information system. The software installation is centrally performed by the CMS software manager sending Grid installation jobs to the sites. JobRobot queries RefDB for available collections, interrogates the PubDB of the sites hosting a given collection, prepares the jobs to analyze the collection and submit them to the sites via a Resource Broker (RB). The RB performs matchmaking to identify the sites verifying certain conditions, like having installed the proper experiment software version, and submits the jobs to the Computer Element (CE) of the site. The CE is the gateway for incoming jobs and submits them to the local batch system. The local batch system runs the jobs in free Worker Nodes (WN) which fetch the appropriate local XML POOL file catalogue according to the job configuration. Jobs access the data from the local SE. Files are not copied from the SE to the WN but are remotely read using the posix-IO implementation of the local storage manager (rfio for CASTOR and DPM, dcap for dCache).

Job monitoring and bookkeeping information is sent to a central database, the Dashboard, using a connectionless UDP protocol. JobRobot sends information at submission and job output retrieval time while running jobs send real time information at job start and job end. In addition, the Dashboard gets information about the state of the jobs (queued, running, etc) from the Logging and Bookkeeping system running at the RB. Bookkeeping is also done through the job log files retrieved by JobRobot when jobs are completed.

3 The SC3 throughput phase

The first phase in SC3 took place during the first two weeks of July 2005. After this period, CMS transfers conducted with PhEDEx were switched off in order not to interfere with experiment-independent FTS [11] transfers centrally run by the SC3 LCG team.

The throughput phase consisted of continuous data transfers at maximum rate from disk at CERN to disk at the Tier-1 centres. Figure 4 shows the average daily throughputs in MB/s during this phase for different sites. Sustained rates above 40 MB/s during almost two weeks with peaks of 85 MB/s (see figure 5) were reached for the transfers CERN to PIC. These transfer rates refer to the successful CMS data volume transferred. The total network traffic, which includes the overhead of the network transfer protocols and failed transfers, was somewhat higher.

In total, about 33 TB were transferred from CERN to PIC in this phase. Plain gridFTP (through *globus-url-copy* client executable) transfers [12] were conducted running 10 parallel file transfers with 10 tcp streams each. Low transfer rates per tcp stream were observed (between 0.5 and 2 MB/s) despite the optimization of kernel tcp parameters in disk servers. At this stage, SRM transfers were not conducted since they suffered from low performance, high unreliability and were difficult to debug (see section 5 for details).

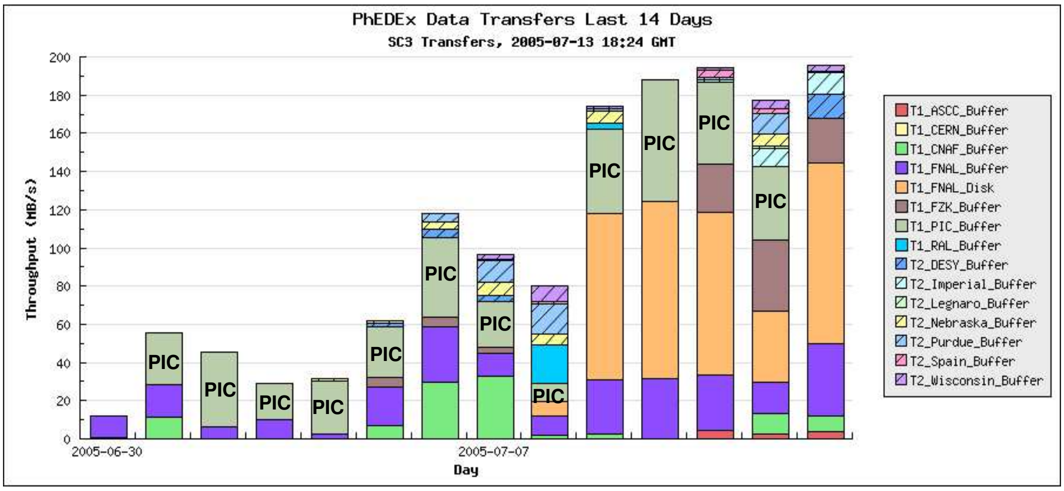

Figure 4: Transfer rates during the SC3 throughput phase between CERN and the Tier-1 centres.

Shortly after the throughput phase, data transfers between Tier-1 sites and their associated tier-2 centres were exercised. Sustained rates of about 18 MB/s from PIC to CIEMAT during 4 days were reached (see figure 6). The transfer rate was found to be limited by the single CASTOR/gridFTP head node at CIEMAT. Later, the transfer rate was doubled by setting up an additional head node, very close to the traffic limit imposed at CIEMAT by the firewall (320 Mbps).

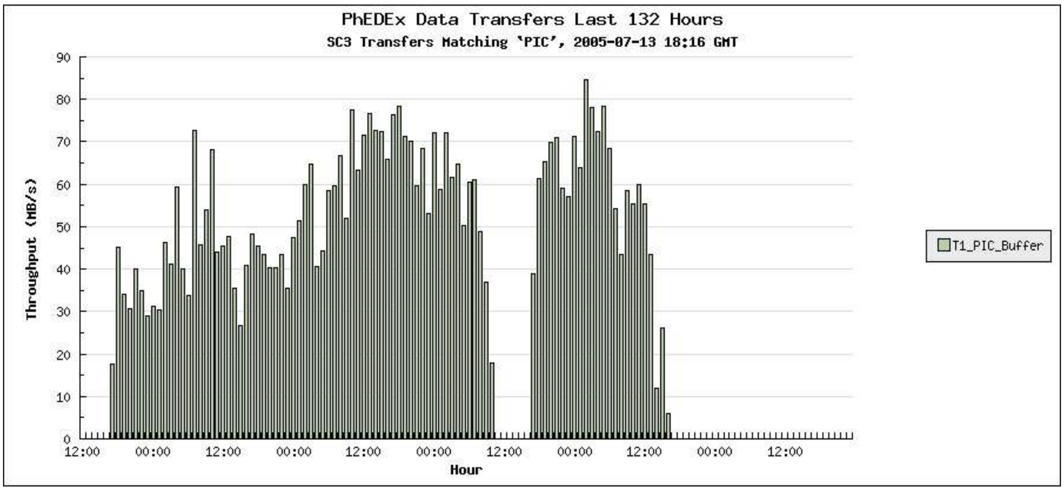

Figure 5: Detail of transfer rates between CERN and PIC during 5 days in the SC3 throughput phase.

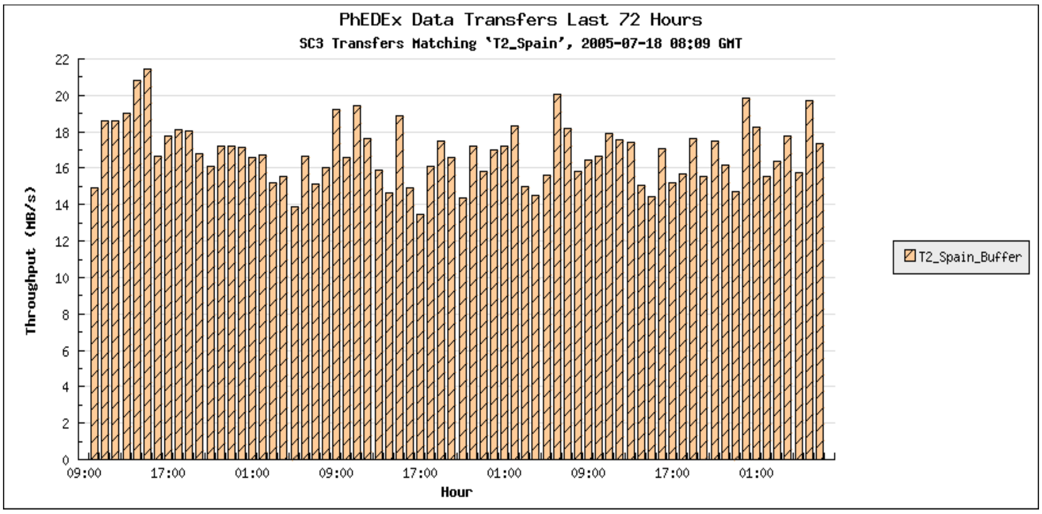

Figure 6: Transfer rates from PIC to CIEMAT during 72 hours.

4 The SC3 service phase

The SC3 service phase took place between September and November 2005. Data transfers at a reduced rate were run in parallel with automatic data publication at the sites and analysis job submission.

In this phase, SRM data transfers via the *srmcp* client executable were conducted, unlike in the throughput phase where plain gridFTP transfers via *globus-url-copy* were done. At the beginning of the service phase, about 3 TB of data were transferred to PIC in order to commission the automated data publication system and provide data to process for analysis jobs. During the last two weeks of the service phase, once the automatic data publication procedure was established, data transfers were gradually resumed increasing the transfer rate up to a stable rate of 2-3 TB/day (see figure 7). Several problems were found and fixed in the SRM implementation in CASTOR (see section 5 for details). In total, 13 TB of data were transferred from CERN to PIC in this phase, with an average throughput of 11.4 MB/s and an average transfer success rate of 62%. The transfer rate and quality were much higher during the last week of the service phase, after understanding and fixing the problems in SRM and CASTOR. An average throughput of 25 MB/s with a success rate of 76% were reached in the last week. 2 TB of data were transferred from PIC to CIEMAT during the service phase with an average throughput of 6.5 MB and a success rate of 63%.

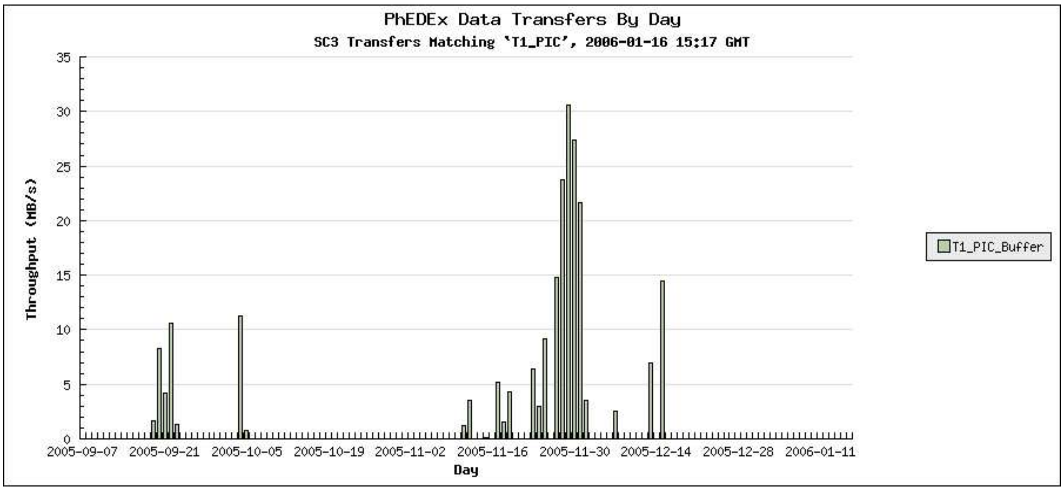

Figure 7: Daily data transfer throughputs from CERN to PIC during the SC3 service phase.

A total of 90 events collections amounting more than 5 million events were published at PIC and CIEMAT. See section 2.2 for details on the publication procedure. Analysis jobs were centrally submitted to the sites as data got published and became available for analysis. The preparation, submission and monitoring of analysis jobs has been already described in section 2.2. About 2000 analysis jobs were run at PIC and 6000 at CIEMAT. Figures 8 and 9 show the job return codes and the distribution of errors over time for PIC and CIEMAT respectively. Both sites reached a job success rate over 95%, being most of the job failures related to application errors not to the infrastructure.

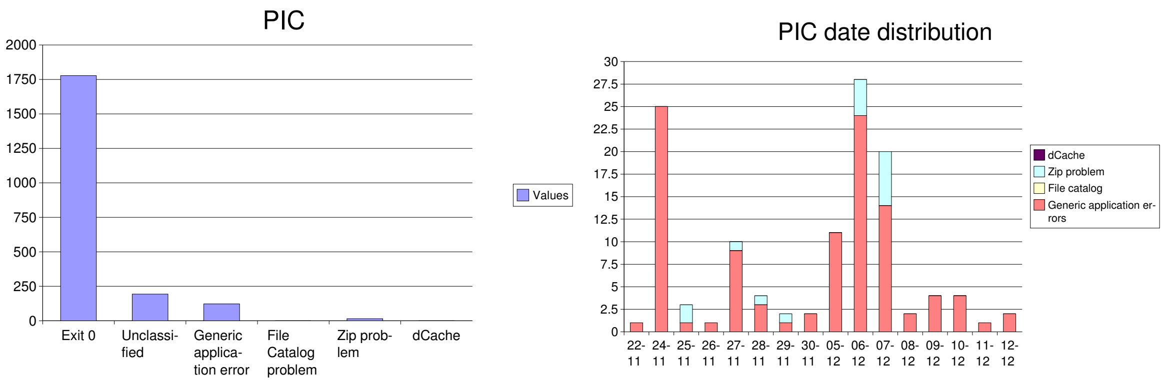

Figure 8: Exit code distribution (left) and evolution with time of the number and type of errors (right) for SC3 jobs run at PIC.

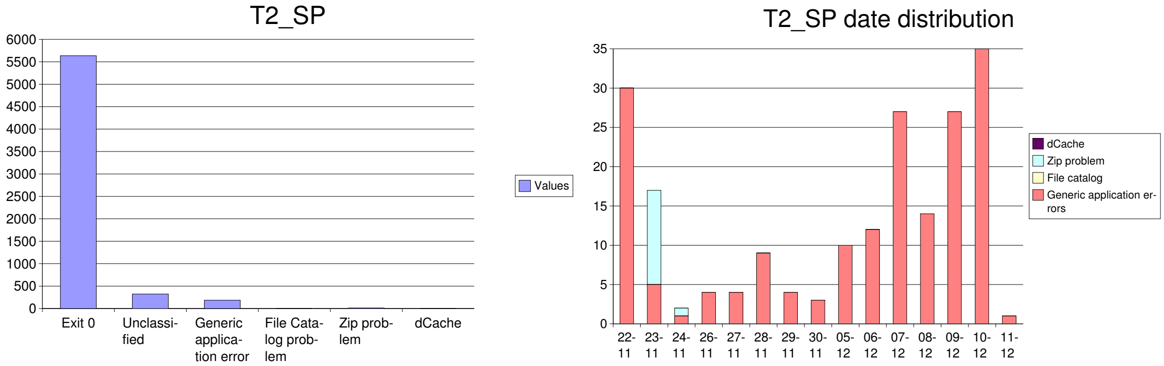

Figure 9: Exit code distribution (left) and evolution with time of the number and type of errors (right) for SC3 jobs run at CIEMAT.

Figures 10 and 11 show for PIC and CIEMAT the aggregate read throughput from disk to CPU for the analysis jobs. The shaded histogram (left scale in the plot) represents the total number of jobs concurrently running a given time. The red line shows the aggregate read throughput (right scale). CPU resources for SC3 were shared with production so that it was not possible to keep a large and stable number of SC3 jobs running concurrently. In addition, the read throughput per job was limited by the CMS analysis application used to a bit more than 1 MB/s. As a consequence, the total aggregate read throughput shown in the plots was limited by those two facts rather than by the storage system. A peak of 130 jobs with an aggregate read throughput of about 200 MB/s was reached at PIC. At CIEMAT, about 110 jobs running in parallel produced a read throughput of about 80 MB/s. These performance figures satisfied SC3 expectations for Tier-1 and Tier-2 sites.

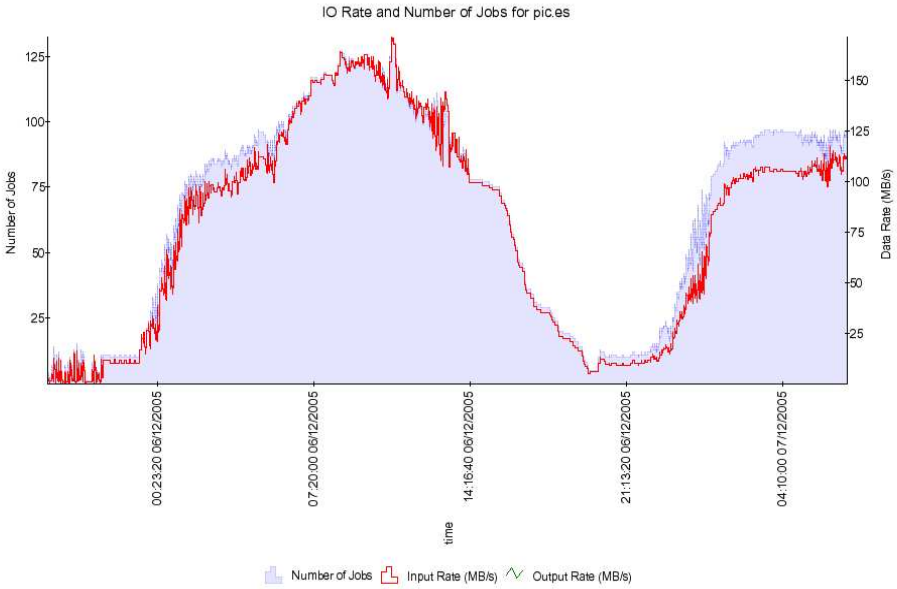

Figure 10: Number of jobs an aggregate read throughput for jobs run at PIC.

5 Experience

Very valuable experience has been gained running SC3. We operated a complex system exercising CMS bulk data processing under realistic conditions. We demonstrated that well-performing, large and complex storage systems are a reality at PIC and CIEMAT. Several parts of the system behaved really well whereas other suffered from several problems. We describe below in detail the main problems we encountered during the computing challenge.

Some of the components performed rather well. That was the case of the job submission system (CRAB/JobRobot), the data management catalogues (RefDB/PubDB) and the PhEDEx data transfer system. PhEDEx is a quite mature product, with good performance and reliability, good monitoring, easy deployment and configuration, reasonable documentation and good support. The increase in average file size (around 2 GB) and decrease of number of files with respect to previous data challenges by using zip file archives had positive effects in several parts of the

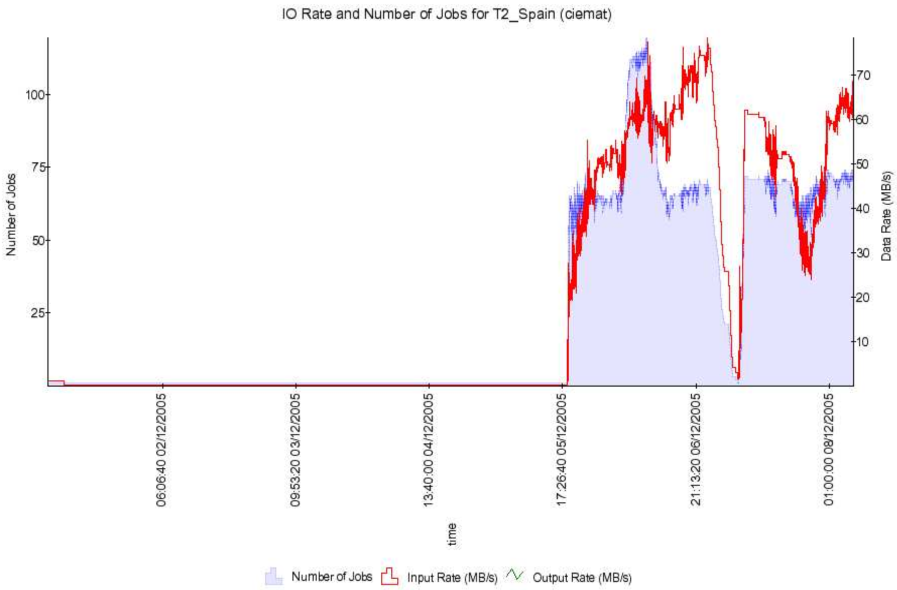

Figure 11: Number of jobs (left scale) an aggregate read throughput (right scale) for SC3 jobs run at CIEMAT

computing system such as catalogues, transfer and storage systems.

The main problem encountered was that some of the components were not sufficiently tested prior to the computing challenge. We had to run a colossal debugging challenge. Little of the software put into test by LCG was validated with CMS prior to SC3. We suffered from endless instabilities in CASTOR-2 at CERN. LCG products like the LCG File Catalogue (LFC), the Data Pool Manager (DPM) and the File Transfer Service (FTS), intended to be tested in SC3 were taken out as immaturity became evident. The Storage Management Interface (SRM) turned out to be less standard than anticipated suffering from interoperability issues between SRM servers at different sites. In addition, we found SRM very difficult to debug and tune. As it became clear that the infrastructure did not work as expected, it was necessary to de-scope initial goals of the challenge.

We found SRM implementation problems in CASTOR. We observed that SRM transfers through the srmcp copy command were returning before CASTOR had finished all internal operations related with the transfer. In particular, file size information in CASTOR was only updated some time after completion of the SRM transfers. This flaw generated a race condition in PhEDEx since, before declaring a transfer successful, as an additional check it makes sure that the destination file size is the one expected. Stability of the SRM servers suffered from the fact that SRM produces a huge number of temporary files in /tmp without cleaning them up. The fill up of the /tmp partition caused all subsequent transfers to fail. A cron job had to be installed at the SRM servers to periodically clean up these SRM temporary files. We also observed that transfers via the srmcp client copy command were ignoring the number of parallel tcp streams, specified as an argument, to be used for file transfers. Transfer were using the default value as specified in the /etc/srm.conf configuration file. We found srmcp error messages not very informative making it difficult to trace problems. SRM transfer via srmcp were run in pull mode. The destination SRM server pulls the data from the source site by initiating globus-url-copy processes at the SRM head node. We observed high CPU load at the destination SRM server caused by srmcp polling at high frequency asking for the status of the transfer.

In data transfers we observed poor network throughput per tcp stream, between 1 and 2 MB/s depending on the number of parallel tcp streams. This translated into a poor individual file transfer throughput. We found a saturation in the file transfer throughput at 5 MB/s running 5 tcp streams in parallel. A higher number of tcp streams did not increase the throughput. We tuned kernel tcp parameters in the SRM servers according to the available bandwidth and latencies. Contrary to what we expected, the tuning did not have any effect on the file transfer throughput. In order to reach a decent aggregate transfer throughput, we were forced to run many transfers in parallel (between 10 and 15) with 5 tcp streams each. This running mode produced a high fragmentation level on the disks of the import pool. This badly affected migration to tape and garbage collection. We observed the CASTOR stager becoming unresponsive while running garbage collection causing all ongoing transfers to fail. In order to alleviate this problem, at the end of the challenge we configured the import disk pool with many partitions/file systems, much higher than the number of parallel transfers, and run srmcp transfers with a single tcp stream. With this configuration disk fragmentation was dramatically reduced greatly improving transfer reliability during garbabe collection. However, we observed a slow increase in the fragmentation that we did not understand. Having separate disk pools for data import/export and for analysis helps to minimize the effects of disk fragmentation in the read throughput of the analysis jobs. Files in the highly fragmented import/export pool were written to tape. From tape they were staged to the highly distributed analysis disk pool resulting in very low disk fragmentation.

CMS data model constraints made it quite difficult to automate parts of the workflow, in particular, due to the necessity of a complicated data publication process. The metadata generation involved in the publication procedure was a resource (CPU, memory, network and time) consuming operation. It was almost impossible to automate becoming the main problem for smooth execution of the SC3 workflows. Data produced with different versions of the CMS processing framework required different versions of the publication software, making automation quite difficult. Metadata generation from the event data requires event data files to be on disk. In order to run data publication as data are downloaded, avoiding to recall data from tape, a large import disk pool was required to fit large event collections on disk. During SC3 there was no automatic way to perform efficient bulk data pre-staging from tape to disk, therefore data pre-stage triggered by access to files not residing on disk any more was avoided as much as possible. Given the large amount of time required to publish big collections, publication had to be parallelized by sending publication jobs to the local batch system making error handling difficult. Automatic error handling was also complicated by the fact that the exit code of the publication application was unreliable. The only way to assess publication success or failure was to look inside publication log files. Lack of transfer priorities in PhEDEx based on event collections resulted in parallel transfer of many collections. Since publication is only run on complete collections, this fact made difficult to have complete collections on disk before data was garbage-collected by CASTOR to free disk space on full disk pools. Publication of zip data files required a catalogue mapping the event data file members inside each zip file. Since a global catalogue containing that information did not exist, PhEDEx, after downloading every zip file had to use COBRA tools to open the zip file, extract the mapping and publish it into the local PhEDEx catalogue. This operation, quite often unreliable, had adverse side effects in the data transfer. In the new CMS event data model, metadata generation will not be necessary any more [13].

Lack of priorities, both in the data transfer and in the workload management system, introduced inefficiencies in making data available for analysis and in running SC3 analysis jobs. Event collections were only published after being downloaded completely. Analysis jobs shared production resources and had to compete with production Monte Carlo and analysis jobs. In these conditions stress tests of the SC3 storage system were difficult to carry out. Priorities will be handled in future introducing roles and groups through the VOMS [14] system.

The job monitoring and bookkeeping system (Dashboard) suffered from unreliability getting job status information via the LCG monitoring system (R-GMA [15]). Performance in the monitoring web interface was quite low making it difficult to follow jobs in real time.

The reliability of the Grid Workload Management System has substantially increased since previous computing challenges. Grid-related errors in job submission were limited to few percent. Most of the errors in analysis jobs were due to local data access problems (wrong data publication resulting in badly generated metadata or local file catalogues, problems accessing data from mass storage, etc) or application crashes. The lack of a proper error handling system made it difficult to track errors and react on them. The new CMS data processing framework [13] will better handle errors and will provide a persistent job report to facilitate error tracking.

We learnt that it takes quite some time to configure and tune sites, especially when the infrastructure is unreliable. It is essential to work with local system administrators at sites during the commissioning phase. SC3 was very costly in terms of manpower, mainly due to the unreliability of the infrastructure.

6 Summary and Conclusions

SC3 has been a substantial test of the CMS end-to-end computing system providing very valuable feedback to the computing programme. It has been an integration test for the next production system focusing on testing the data transfer and data serving infrastructure under realistic conditions.

Important achievements have been attained. A separation of site and global CMS responsibilities has been established. Sites are responsible for providing a functional and performant mass storage system and mass storage interfaces, reliable Grid infrastructure and Grid interfaces as well as publishing and making data available for analysis. Data transfer and workflow management as well as organized job submission and monitoring is a global CMS responsibility. The user support model where Tier-1 centres support associated Tier-2 sites has been exercised. Data replication from Tier-0 to all Tier-1 centres down to associated Tier-2 sites has been successfully achieved. Automatic data preparation and publication for analysis has been conducted at the participating Tier-1 and Tier-2 sites. The data placement policy has been demonstrated and applications have been driven to sites based on data location. CMS workflow for analysis job has been carried out and the transfer, data serving and job submission systems have been run at a significant scale.

However, since some of the components were not sufficiently prepared to conduct meaningful tests in a challenge environment, some tests on functionality and performance levels had to be replaced by infrastructure debugging and tuning. Some original goals of the computing challenge had to be de-scoped: running simulation jobs at the Tier-2 sites and sending the produced simulated data to the associated Tier-1; running organized data processing applications like data skimming, calibration reprocessing which involve output data, registration and storage; reaching higher scale in transfers, data serving and job submission. These goals will be taken up again in the upcoming SC4 computing challenge in June 2006.

We were dealing with very complex systems where frequent errors caused by the unreliability and instability of the underlying infrastructure occurred. The lack of a proper error handling and diagnostic tools made it quite difficult to automate some of the pieces of the workflow. Fragile elements like data publication will be removed in the new Event Data Model and Data processing framework currently under development.

Overall, despite the huge effort invested, the results are positive. We have demonstrated that large, complex and well-performing Grid storage and workflow management systems are a reality at the Spanish Tier-1 and Tier-2 centres.

Acknowledgments

Without the collaboration of many people at PIC and CIEMAT, SC3 at the Spanish sites would not have been a success. We are indebted to them. We wish to thank the CMS SC3 team, in particular Lassi Tuura, Jens Rehn, Andrea Sciabà and Marco Corvo.

References

[1] https://twiki.cern.ch/twiki/bin/view/LCG/LCGServiceChallenges

[2] http://castor.web.cern.ch/castor/

[3] http://sdm.lbl.gov/srm-wg/

[4] J. Rehns et al., PhEDEx high-throughput data transfer management system. Proceedings of the CHEP06 Conference. Mumbai, India, February 2006.

[5] http://www.dcache.org

[6] https://twiki.cern.ch/twiki/bin/view/LCG/DpmAdminGuide

[7] http://pool.cern.ch

[8] http://lynx.fnal.gov/runjob/Setup_20Publish_20Service

[9] http://cmsdoc.cern.ch/cms/cpt/Computing/Technical/subproj/RefDB.html.

[10] http://cmsdoc.cern.ch/cms/cpt/Computing/Technical/subproj/PubDB.html

[11] https://twiki.cern.ch/twiki/bin/view/LCG/GliteFTSInformation

[12] http://www.globus.org/toolkit/data/gridftp/

[13] C. Jones et al., The new CMS event data model and framework. Proceedings of the CHEP06 Conference. Mumbai, India, February 2006.

[14] http://hep-project-grid-scg.web.cern.ch/hep-project-grid-scg/voms.html

[15] http://www.r-gma.org/