Distributed Computing Grid Experiences in CMS

J. Andreeva, A. Anjum, T. Barrass, D. Bonacorsi, J. Bunn, P. Capiluppi, M. Corvo, N. Darmenov, N. De Filippis, F. Donno, G. Donvito, G. Eulisse, A. Fanfani, F. Fanzago, A. Filine, C. Grandi, J. M. Hernández, V. Innocente, A. Jan, S. Lacaprara, I. Legrand, S. Metson, H. Newman, D. Newbold, A. Pierro, L. Silvestris, C. Steenberg, H. Stockinger, L. Taylor, M. Thomas, L. Tuura, T. Wildish, and F. Van Lingen

***Abstract*—The CMS experiment is currently developing a computing system capable of serving, processing and archiving the large number of events that will be generated when the CMS detector starts taking data. During 2004 CMS undertook a large scale data challenge to demonstrate the ability of the CMS computing system to cope with a sustained data-taking rate equivalent to 25% of startup rate. Its goals were: to run CMS event reconstruction at CERN for a sustained period at 25 Hz input rate; to distribute the data to several regional centers; and enable data access at those centers for analysis. Grid middleware was utilized to help complete all aspects of the challenge. To continue to provide scalable access from anywhere in the world to the data, CMS is developing a layer of software that uses Grid tools to gain access to data and resources, and that aims to provide physicists with a user friendly interface for submitting their analysis jobs. This paper describes the data challenge experience with Grid infrastructure and the current development of the CMS analysis system.**

***Index Terms*—Data flow analysis, data management, data processing, distributed computing, distributed information systems, high energy physics.**

I. Introduction

THE Compact Muon Solenoid (CMS) is one of four particle physics experiments associated with the Large Hadron Collider (LHC) currently being built at CERN. Even though the detector will not take data until 2007 there is a large-scale data simulation and analysis effort underway for CMS: the hundreds of physicists that contribute to the CMS collaboration are currently taking part in computationally-intensive Monte Carlo simulation studies of the detector, and its potential for discovering new physics. The CMS collaboration has a long-term need to perform large-scale simulations in which physics events are generated and their manifestations in the CMS detector are simulated. These simulation efforts support detector design and the design of real-time event filtering algorithms that will be used when CMS is running. Furthermore, running large-scale simulations develops the collaboration's working environment. Experience gained in developing this environment helps to refine the design of reconstruction and analysis frameworks needed to process the large number of events that will be generated when the detector starts collecting data.

The challenge for CMS computing is therefore to cope with large-scale computational and data access requirements. The size of the required resources in terms of processing power and storage capacity, the complexity of the software and the geographical distribution of the CMS collaboration have led to an underlying distributed computing and data access system. Grid technology is one of the most promising infrastructures with the potential to manage such a system in a scalable way. CMS is collaborating with many Grid projects around the world in order to explore the maturity and availability of middleware implementations and architectures. CMS decided to actively participate in the Grid projects at their outset, with the aim of understanding how Grids might be useful for CMS and how CMS software needs to be adapted to maximize the benefit of using Grid functionalities and tools.

CMS requires that the design and construction of a computing system capable of managing CMS' data pass through a series of planned test-steps of increasing complexity, named D*ata* and P*hysics challenges*. The Data Challenge for CMS during 2004 (CMS DC04) aimed to reach a complexity equivalent to about 25% of that foreseen for LHC initial running. Its goal was to run CMS reconstruction at CERN for a sustained period of time at 25 Hz input rate, distribute the data to the CMS regional centers and then enable analysis at those centers.

To meet this challenge a large simulated event generation, named the prechallenge production (PCP), of about 50 million events was undertaken during the preceding months. During the

Manuscript received November 15, 2004; revised March 24, 2005.

J. Andreeva, N. Darmenov, F. Donno, A. Filine, V. Innocente, and A. Jan are with CERN, Geneva CH-1211, Switzerland (e-mail: Julia.Andreeva@cern.ch; Nikolay.Darmenov@cern.ch; Flavia.Donno@cern.ch; Alexei.Filine@cern.ch; Vincenzo.Innocente@cern.ch; Asif.Muhammad@cern.ch).

A. Anjum is with National University of Science and Technology, Pakistan (e-mail: Ashiq.Anjum@cern.ch).

T. Barrass, S. Metson, and D. Newbold are with Bristol University, Bristol BS8 1UQ, U.K. (e-mail: tim.barrass@bristol.ac.uk; Simon.Metson@cern.ch; Dave.Newbold@cern.ch).

D. Bonacorsi is with INFN-CNAF, 40127 Bologna, Italy (e-mail: bonacorsi@bo.infn.it).

J. Bunn, I. Legrand, H. Newman, C. Steenberg, M. Thomas, and F. Van Lingen are with California Institute of Technology, Pasadena, CA 91125 USA (e-mail: julian@cacr.caltech.edu; Iosif.Legrand@cern.ch; Harvey.Newman@cern.ch; conrad@hep.caltech.edu; thomas@hep.caltech.edu; fvlingen@caltech.edu).

P. Capiluppi, A. Fanfani, and C. Grandi are with University of Bologna, Bologna 40127, Italy, and INFN-Bologna, Bologna 40011, Italy (e-mail: Paolo.Capiluppi@bo.infn.it; fanfani@bo.infn.it; Claudio.Grandi@bo.infn.it).

M. Corvo, F. Fanzago, and S. Lacaprara are with University of Padova and INFN-Padova, Padova 35131, Italy (e-mail: Marco.Corvo@cern.ch; fanzago@pd.infn.it; Stefano.Lacaprara@pd.infn.it).

N. De Filippis, G. Donvito, and A. Pierro are with University and Politecnico of Bari and INFN-Bari, Bari 141980, Italy (e-mail: Nicola.Defilippis@ba.infn.it; Giacinto.Donvito@ba.infn.it; Antonio.Pierro@ba.infn.it).

G. Eulisse, L. Taylor, and L. Tuura are with Northeastern University, Evanston, IL 60625 USA (e-mail: Giulio.Eulisse@cern.ch; Lucas.Taylor@cern.ch; Lassi.Tuura@cern.ch).

J. M. Hernández is with CIEMAT, Madrid 28043, Spain (e-mail: jose.hernandez@ ciemat.es).

L. Silvestris is with INFN-Bari, Bari 141980, Italy (e-mail: Lucia.Silvestris@cern.ch).

H. Stockinger is with CERN, Geneva CH-1211, Switzerland and INFN-Padova, Padova 35131, Italy (e-mail: Heinz.Stockinger@cern.ch).

T. Wildish is with Princeton University, Princeton, NJ 08544 USA (e-mail: Tony.Wildish@cern.ch).

Digital Object Identifier 10.1109/TNS.2005.852755

PCP, prototype CMS production infrastructures based on Grid middleware were deployed. The prototypes were based on early LHC Computing Grid [1] (LCG-0 and LCG-1) systems, where most of the features used were provided by EU implemented middleware, and on Grid environments like Grid3 [2], as used by the USMOP system [3] in the USA. Large scale productions were performed using these prototypes, demonstrating that it is possible to use them for real data production tasks [4].

During the CMS DC04, data distribution and data analysis at distributed sites ran in a prototype Grid environment using several LCG-2 tools. Automatic procedures were implemented to submit analysis jobs on the arrival of new data at a given site They were integrated with the Grid services, and exhibited good performance, enabling CMS to undertake Grid-enabled data analysis and to identify potential bottlenecks.

The next challenge, due by the end of 2005, will be the preparation of the CMS Physics Technical Design Report that will require analysis of hundreds of Terabytes of data. CMS is developing a layer of software that uses Grid tools to provide access to the whole data sample and analysis services to CMS physicists worldwide, via a user-friendly interface for submitting analysis jobs.

The paper is organized as follows. Section II describes the CMS computing environment and the software used for simulating and processing CMS event data. Experience of data distribution and analysis during CMS data challenge 2004 is reported in Section III. Current activities to provide an user analysis system born of CMS' experience during DC04 are described in Section IV. Section V summarizes the results and gives a brief outlook.

II. CMS Computing Environment

LHC will produce 40 million collisions (or events) per second in the CMS detector, which correspond to a data rate of about 1000 TB/s. The on-line system will reduce the rate to 100 events per second, equivalent to an estimated data rate of 100 MB/s, streamed to permanent store and used as input for off-line processing. The on-line system selects interesting events in the following two steps.

- the *Level-1 Trigger:* which is implemented in custom designed hardware.
- the *High Level Trigger:* which is implemented as software running on a computing (on-line) farm.

Detector data that passes the High Level Trigger selection are called *raw* data. The raw data event size will be approximately 1 MB and will be archived on persistent storage (∼1 PB/year).

Raw data will be reconstructed at CERN to create new higher-level physics objects (Reconstructed objects). The raw and reconstructed data will be distributed to regional computing centers of collaborating institutes, from where it will be made available to the collaboration for analysis.

Software has been produced both for simulating and processing CMS event data. Fig. 1 describes the programs and data formats used.

- *Event Generation:* Pythia [5] and other generators that produce *N-tuple* files in the HEPEVT format [6];

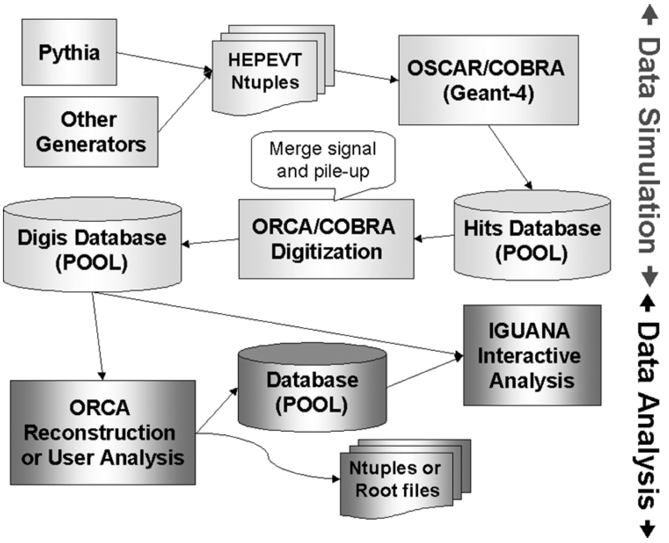

Fig. 1. CMS data formats and software used for data simulation and analysis. Arrows indicate data-flow.

- *Detector simulation:* OSCAR [7] is the package simulating the particle passage through the CMS detector, based on Geant-4 [8] toolkit and the CMS object-oriented framework COBRA [9]. The persistency layer used by the CMS framework is POOL [10]. Reading the generated N-tuples OSCAR produces simulated particle positions in all sub-detectors, producing data structures named *Hits*.
- *Digitization:* This is the simulation of the data acquisition (DAQ) process and its output is the simulated detector response, in data structures named *Digis*. This data is produced by software named ORCA (Object-oriented Reconstruction for CMS Analysis) [9] that uses the CMS COBRA framework. It produces POOL files containing *Digis*, taking files of Hits as input.
- *Trigger simulation:* ORCA simulates the Level-1 trigger and High Level Trigger, taking the Digi POOL files as input. Trigger simulation is normally run as part of the reconstruction phase.
- *Reconstruction:* ORCA reconstructs particle tracks in the detector using Digi data, and outputs data structures representing reconstructed physics objects in POOL files.
- *Analysis:* Both Physics group and end-user analysis is done using ORCA. It is possible to read any kind of POOL file produced in one of the processing steps. Visual analysis is undertaken using IGUANACMS [11], a program that uses ORCA and OSCAR as back-ends. It provides the functionality needed for displaying events, and undertaking statistical analysis.

A multi-Tier hierarchical distributed model is adopted in CMS to rationalize resource use. The detector is associated with a Tier-0 site. Tier-1 sites are typically large regional or national computing centers with significant tape and disk capacity, and network resources with high bandwidth and availability. Tier-2 sites are institutes with a sizeable computing capacity that are able to contribute to the institutional needs of the experiments as well as to the support of local users. Tier-3 centers are

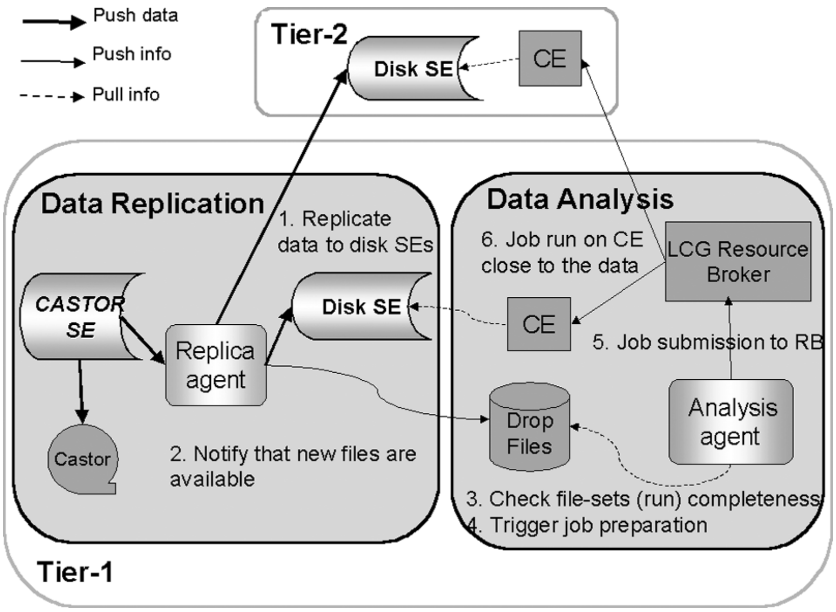

Fig. 2. Description of the real-time analysis architecture during CMS data challenge 2004. Detailed description is in the text.

institutes with a more restricted availability of resources and/or services that basically provide support only to local users.

A core set of Tier-1 sites will store raw and reconstructed data to safeguard against data loss at CERN. These sites will also provide facilities for analysis, some re-reconstruction, and handle regional data distribution. Smaller sites, associated with certain Physics analysis groups or Universities, will contribute substantially to the analysis activities.

III. CMS Data Challenge 2004 Experience

The CMS Data Challenge in 2004 (DC04) comprised the following phases:

- reconstruction of data on the CERN Tier-0 farm for a sustained period at 25 Hz;
- data distribution to Tier-1 and Tier-2 sites;
- prompt data analysis at remote sites on arrival of data;
- the monitoring and archiving of resource and process information;

- reconstruction of data on the CERN Tier-0 farm for a sustained period at 25 Hz;
- data distribution to Tier-1 and Tier-2 sites;
- prompt data analysis at remote sites on arrival of data;
- the monitoring and archiving of resource and process information.

The aim of the challenge was to demonstrate the feasibility of operating this full chain of processes.

A. Reconstruction

Digitized data were stored on CASTOR [12] Mass storage system at CERN. A fake on-line process made these data available as input for the reconstruction with a rate of 40 MB/s.

Reconstruction jobs were submitted to a computer farm of about 500 CPUs at the CERN Tier-0. The produced data (4 MB/s) were stored on a CASTOR stage area, so files were automatically archived to tape. Some limitations concerning the use of CASTOR at CERN due to the overload of the central tape stager were found during DC04 operations.

B. Data Distribution

For DC04 CMS developed a data distribution system over available Grid point-to-point file transfer tools, to form a scheduled large-scale replica management system [13]. The distribution system was based on a structure of semi-autonomous software agents collaborating by sharing state information through a transfer management database. A distribution network with a star topology was used to propagate replicas from CERN to 6 Tier-1s and multiple associated Tier-2s in the USA, France, U.K., Germany, Spain, and Italy. Several data transfer tools were supported: the LCG Replica Manager tools [1], storage resource manager [14], specific transfer tools, and the storage resource broker [15]. A series of "export buffers" at CERN were used as staging posts to inject data into the domain of each transfer tool. Software agents at Tier-1 sites replicated files, migrated them to tape, and also made them available to associated Tier-2s. The final number of file-replicas at the end of the two months of DC04 was ~3.5 million. The data transfer (~6 TB of data) to Tier-1s was able to keep up with the rate of data coming from the reconstruction at Tier-0. The total network throughput was limited by the small size of the files being pushed through the system [16].

A single local replica catalog (LRC) instance of the LCG replica location service (RLS) [17] was deployed at CERN to locate all the replicas. Transfer tools relied on the LRC component of the RLS as a global file catalogue to store physical file locations.

The replica metadata catalog (RMC) component of the RLS was used as global metadata catalogue, registering the files attributes of the reconstructed data; typically the metadata stored in the RMC was the primary source of information used to identify logical file collections. Roughly 570k files were registered in the RLS during DC04, each with 5 to 10 replicas, and 9 metadata attributes per file (up to ~1 kB metadata per file). Some performance issues were found when inserting and querying information; the RMC was identified as the main source of these issues. The time to insert files with their attributes in the RLS—about 3s/file in optimal conditions—was at the limit of acceptability; however, service quality degraded significantly with extended periods of constant load at the required data rate. Metadata queries were generally too slow, sometimes requiring several hours to find all the files belonging to a given "dataset" collection. Several workarounds were provided to speed up the access to data in the RLS during DC04. However, serious performance issues and missing functionality, like a robust transaction model, still need to be addressed.

C. Data Analysis

Prompt analysis of reconstructed data on arrival at a site was performed in quasi-real-time at the Italian and Spanish Tier-1 and Tier-2 centers using a combination of CMS-specific triggering scripts coupled to the data distribution system, and the LCG infrastructure.

A set of software agents and automatic procedures were developed to allow analysis-job preparation and submission as data files were replicated to Tier-1s [18]. The quasi-real-time analysis architecture is shown in Fig. 2. The data arriving at the Tier-1 CASTOR data server (Storage Element) were replicated by a dedicated agent (Replica agent) to disk storage elements at Tier-1 and Tier-2 sites. Whenever new files were available on disk the Replica agent was also responsible for notifying an Analysis agent, which in turn triggered job preparation when all files of a given file set (run) were available. The jobs were submitted to an LCG-2 RB, which selected the appropriate site to run the jobs.

The official release of the CMS software required for analysis (ORCA) was preinstalled on LCG-2 sites by the CMS software manager by running installation Grid jobs. The ORCA analysis executable and libraries for specific analyses were sent with the job.

The workflow of an analysis job in the LCG environment is shown in Fig. 3. The job was submitted from the user interface (UI) to the RB that interpreted the user requirements specified using the job description language. The RB queried the RLS to discover the location of the input files needed by the job and selected the computing element (CE) hosting those data. The LCG information system is used by the RB to find out the information about the available grid resources (CEs and storage elements). A RB and an Information System reserved for CMS were set-up at CERN. CMS could dynamically add or remove resources as needed.

The job ran on a worker node, performing the following operations: establish a CMS environment, including access to the preinstalled ORCA; read the input data from a storage element using the RFIO protocol [12] whenever possible—otherwise via LCG Replica Manager commands; execute the user-provided executable; store the job output on a data server; and register it to the RLS to make it available to the whole collaboration.

The automated analysis ran quasi-continuously for two weeks, submitting a total of more than 15 000 jobs, with a job completion efficiency of 90%–95%. An example plot of the number of jobs run per hour in a day is shown in Fig. 4. Taking into account that the number of events per job varied from 250 to 1000, the maximum rate of jobs, ~260 jobs/hour, translated into rate of analyzed events of about 40 Hz. The LCG submission system could cope very well with this maximum rate of data coming from CERN. The Grid overhead for each job, defined as the difference between the job submission time and the time of start execution, was on average around 2 min. An average latency of 20 min between the appearance of the file at CERN and the start of the analysis job at the remote sites was measured during the last days of DC04 running, as reported in Fig. 5.

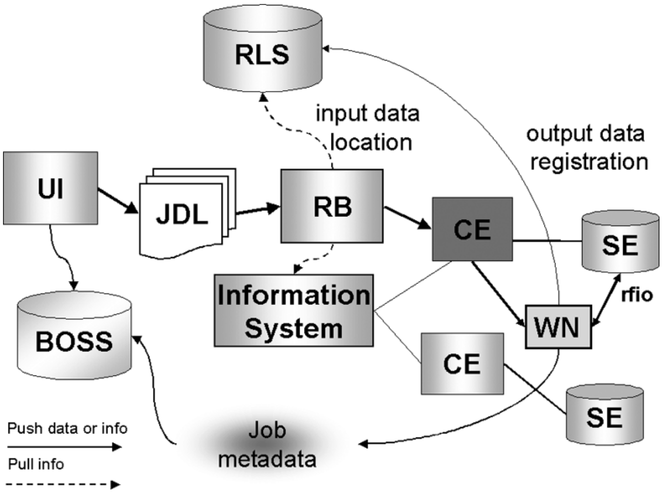

Fig. 3. Description of the analysis job operations in an LCG-2 environment during CMS data challenge 2004. Detailed description is in the text.

D. Monitoring

MonaLisa [19] and GridICE [20] were used to monitor the distributed analysis infrastructure, collecting detailed information about nodes and service machines (the RB, and Computing and Storage Elements), and were able to notify the operators in the event of problems. CMS-specific job monitoring was managed using BOSS [21]. BOSS extracts the specific job information to be monitored from the standard output and error of the job itself and stores it in a dedicated MySQL database. The job submission time, the time of start and end execution, the executing host are monitored by default. The user can also provide to BOSS the description of the parameters to be monitored and the way to access them by registering a job-type. An analysis specific job-type was defined to collect information like the number of analyzed events, the datasets being analyzed.

IV. User Analysis

CMS distributed analysis activities are now in a research-and-design phase, and are focused on providing an end-to-end analysis system. In general user analysis is a chaotic, nonorganized task, carried out concurrently by many independent users that do not have a deep knowledge of the distributed computing environment they are working on. CMS is testing several prototypes of tools that act as an interface to such environments.

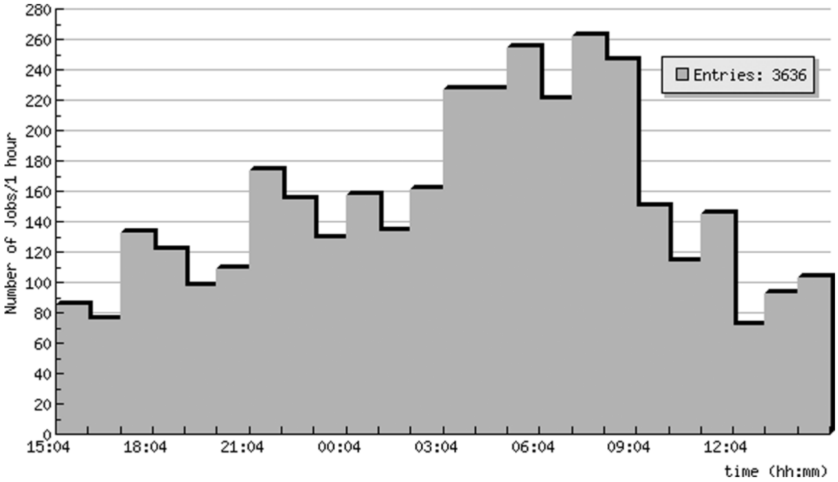

Fig. 4. Number of jobs per hour analyzed in a day.

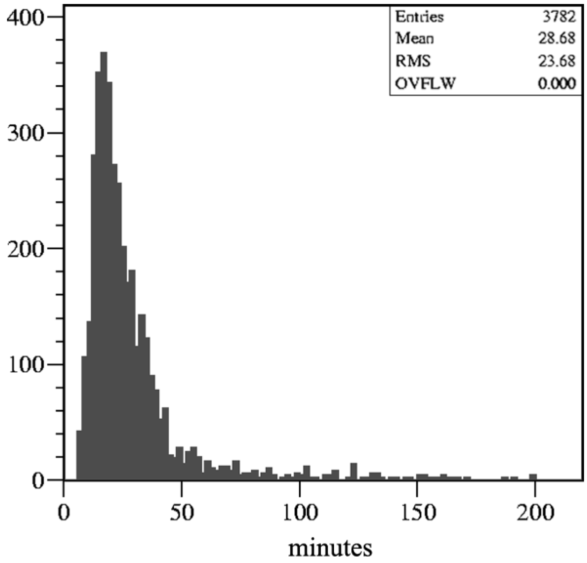

Fig. 5. Time elapsed from the file being available for distribution at CERN and its analysis at remote site.

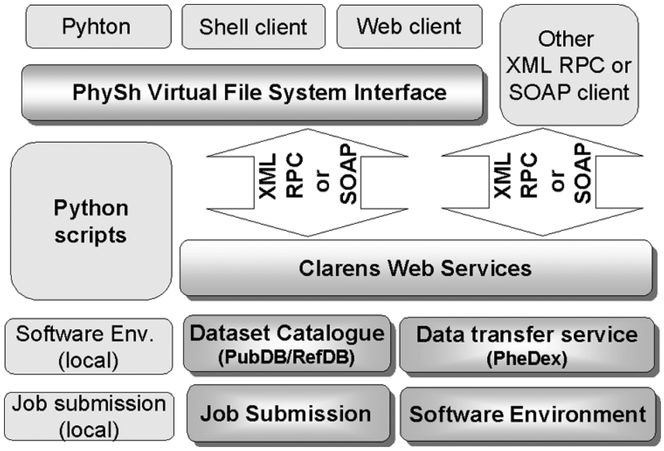

Fig. 6. Description of the PhySh architecture.

PhySh (Physics Shell) is an application that aims to reduce the number of different tools and environments that the CMS physicist must learn to interact with so that they can to use distributed data and computing services. In essence PhySh is an extensible "glue" interface among different services already present or to be coded, like locating physics data of interest, copying/moving event data to new locations, accessing software releases/repositories, and so on. The PhySh interface is modeled as a virtual file-system, since file-system interfaces are what most people are familiar with when dealing with their data. PhySh is based on the Clarens [22] Grid-enabled web service infrastructure. Clarens was developed as part of the Grid-enabled Analysis environment (GAE) [23]. Clarens servers leverage the Apache web server to provide a scalable framework for clients to communicate with services using the SOAP and XML-RPC protocols. The architecture of PhySh is shown in Fig. 6.

A. Data Access

Users require a *Data Location* service, to discover what data exists and where, and a *Data Transfer* service to accommodate the needs of data distribution between multiple sites. For CMS data location is currently provided by services named RefDB and PubDB, while bulk data transfer is handled by a system named PhEDEx.

Users typically want to access large collections of data spanning many files; thus, file-based data access, as provided by many existing Grid tools is not satisfactory. CMS-specific dataset catalogues that describe dataset characteristics and enable the location of replicas that comprise a dataset are under development. The reference database (RefDB) [24] is a Dataset Metadata Catalogue; it maintains audit trails of data production, information that associates files with datasets, and other bookkeeping metadata. The Publication Database (PubDB) manages information about "local" dataset catalogues, allowing the users to locate the data of a dataset and determine access methods.

The information in PubDB can be used by users or by the Workload Management System [25] to decide where to submit a job analysing a given dataset. The natural design is to have distributed PubDBs, one per site that serves data, allowing the site data-manager to manage their local catalogues coherently and in a manner consistent with other CMS sites. The user can discover the available datasets querying the RefDB. A global map of all dataset catalogues is held in the RefDB through the links to the various PubDBs.

The CMS bulk data transfer management system is named PhEDEx (PHysics Experiment Data EXport) [12]. PhEDEx is a project born of CMS' experience during DC04. It retains the same architecture as that used during DC04, relying on a central "blackboard" to enable the exchange of information among a series of distributed agents. The principal current aim of PhEDEx is to incorporate the CMS distribution use cases of subscription pull of data where a site subscribes to all data in a given set and data are transferred as they are produced, and of random pull where a site or individual physicist just wishes to replicate an existent dataset in a one-off transfer.

Data is distributed for CMS through a hierarchy of sites, with smaller sites associating with larger, and subscribing to some subset of the data stored at the larger. The system also has multiple sources of data—the detector, various simulation sites, and analysis sites.

To enable the addition of multiple data sources PhEDEx borrows from established internet technology: where routes from single source to multiple destinations were hard coded into agents during DC04, in PhEDEx nodes in the distribution chain act as routers which share route information using an implementation of the RIP2 [26] algorithm. Any node in the network can act as a source of data, and a route from any node to any other node in the chain can be determined.

B. Analysis Strategy

The current approach to analysis in CMS is to concentrate on simple analysis scenarios and learn from the implementation of simple use cases. The adopted analysis strategy using the Grid is to submit analysis jobs close to the data. The user typically wants to access a dataset in order to analyze it with his private code. The user provides, in the specification of a Grid job, the dataset and the private code. On submission to the Grid a combination of specialized CMS tools and Grid components should take care of resource matching and submission to distributed sites.

The Workload Management System finds suitable computing resources (i.e., CEs) to execute the job. Data discovery is one of the most important aspects in the match-making process, and a Data Location Interface is being developed to allow a uniform "query interface" for data location. This interface will provide the Workload Management System with the functionality to query several catalogues: the LCG RLS to perform file-based data location; the CMS specific dataset catalogues to perform dataset-based data location; or any upcoming data catalogue.

Several user-friendly tools dealing with job preparation, job splitting and job submission are under development. These tools are being integrated with middleware and tools already available and new ones being developed in several Grid projects: LCG, EGEE [27], Grid3, OSG [28].

V. Conclusions

CMS is exploring the maturity and availability of middleware implementations and architectures of many Grid projects to provide access to the data, to process and distribute the data to a large number of globally dispersed CMS physicists.

In CMS Data Challenge 2004 the LCG environment provided the functionality required for distributed computing: global file and metadata catalogues, Grid point-to-point file transfer tools, workload infrastructure for data analysis and Grid monitoring service.

The CERN RLS provided the replica catalogue functionality for all the data distribution chains. Major performance issues were found with data at the scale of the challenge.

A data distribution management layer was developed by CMS, by loosely integrating available Grid components to manage wide area transfers. The system allowed the management of large data flows automating a succession of Grid point-to-point transfers. Over 6 TB of data were distributed to Tier-1 sites, reaching sustained transfer rates of 30 MB/s. The total network throughput was limited by the small size of the files being pushed through the system. Dealing with too many small files also increased the load in updating/querying the catalogue and also affected the scalability of CASTOR MSS.

Prompt data analysis occurring as soon as files arrived at Tier-1 and Tier-2 sites was demonstrated using CMS software and the LCG infrastructure. During the last days of the data challenge a median latency of $\sim$20 min was measured between the appearance of the file at CERN and the start of the analysis job at remote sites.

The limitations identified during the CMS Data Challenge 2004 are being addressed by the LCG and EGEE project.

Current work is focused on developing a layer of software that uses the Grid tools to gain access to data and resources, and that aims to provide physicists with a user-friendly interface for submitting analysis jobs. This activity includes components from several Grid projects such as LCG, EGEE, GRID-3, and Clarens.

References

[1] I. Bird *et al.*, "Operating the LCG and EGEE production Grids for HEP," presented at the Computing in High Energy and Nuclear Physics (CHEP) Conf., Interlaken, Switzerland, 2004.

[2] I. Foster *et al.*, "The Grid2003 production Grid: principles and practice," presented at the 13th IEEE Int. Symp. High Performance Distributed Computing, Honolulu, HI, 2004.

[3] The MOP Project. [Online]. Available: http://www.uscms.org/Software-Computing/Grid/MOP

[4] A. Fanfani *et al.*, "Distributed computing grid experiences in CMS DC04," presented at the Computing in High Energy and Nuclear Physics (CHEP) Conf., Interlaken, Switzerland, 2004.

[5] T. Sjöstrand *et al.*, "High energy physics event generation with PYTHIA 6.1," *Comput. Phys. Commun.*, vol. 135, p. 238, 2001.

[6] ——, *Z Physics at LEP1*, G. Altarelli, R. Kleiss, and C. Verzegnassi, Eds., 1989, vol. 3, p. 143.

[7] M. Stavrianakou *et al.*, "An object-oriented simulation program for CMS," presented at the Computing in High Energy and Nuclear Physics (CHEP) Conf., Interlaken, Switzerland, 2004.

[8] S. Agostinelli *et al.*, "Geant4: a simulation toolkit," *Nucl. Instrum. Meth.*, vol. A 506, pp. 250–303, 2003.

[9] V. Innocente *et al.*, "CMS software architecture: software framework, services and persistency in high level trigger, reconstruction and analysis," *Comput. Phys. Commun.*, vol. 140, pp. 31–44, 2001.

[10] D. Düllmann *et al.*, "POOL development status and plans," presented at the Computing in High Energy and Nuclear Physics (CHEP) Conf., Interlaken, Switzerland, 2004.

[11] I. Osborne *et al.*, "Composite framework for CMS applications," presented at the Computing in High Energy and Nuclear Physics (CHEP) Conf., Interlaken, Switzerland, 2004.

[12] O. Bärring, B. Couturier, J.-D. Durand, and S. Ponce, "CASTOR: operational issues and new developments," presented at the Computing in High Enery and Nuclear Physics (CHEP) Conf., Interlaken, Switzerland, 2004.

[13] T. Barrass, "Software agents in data and workload management," presented at the Computing in High Energy and Nuclear Physics (CHEP) Conf., Interlaken, Switzerland, 2004.

[14] J. Gu *et al.*. (2004, Mar.) The Storage Resource Manager Interface Specification. [Online]. Available: http://sdm.lbl.gov/srm-wg/doc/SRM.spec.v2.1.1.doc

[15] A. Rajasekar *et al.*, "SRB, managing distributed data in a Grid," *Comput. Soc. India J. (Special Issue on SAN)*, vol. 33, no. 4, pp. 42–54, 2003.

[16] D. Bonacorsi, "Role of Tier-0, Tier-1 and Tier-2 regional centers during CMS DC04," presented at the Computing in High Energy and Nuclear Physics (CHEP) Conf., Interlaken, Switzerland, 2004.

[17] D. Cameron *et al.*, "Replica management in the european data grid project," *J. Grid Computing 2004*, vol. 2, no. 4, pp. 341–351, 2004.

[18] N. De Filippis *et al.*, "Real-time analysis at Tier-1 and Tier-2 in CMS DC04," presented at the Computing in High Energy and Nuclear Physics (CHEP) Conf., Interlaken, Switzerland, 2004.

[19] I. C. Legrand *et al.*, "Monalisa: a distribute monitoring service architecture," presented at the Computing in High Enery and Nuclear Physics (CHEP) Conf., La Jolla, CA, 2003.

[20] S. Andreozzi *et al.*, "GridICE: a monitoring service for the Grid," presented at the 3rd Cracow Grid Workshop, Cracow, Poland, Oct 2003.

[21] C. Grandi and A. Renzi. (2003, March) Object Based System for Batch Job Submission and Monitoring. [Online]http://cmsdoc.cern.ch/documents/03/note03_005.pdf

[22] C. Steenberg *et al.*, "The Clarens web service architecture," presented at the Computing in High Energy and Nuclear Physics (CHEP) Conf., La Jolla, CA, 2003.

[23] F. van Lingen *et al.*, "Grid enabled analysis: architecture, prototype and status," presented at the Computing in High Energy and Nuclear Physics (CHEP) Conf., Interlaken, Switzerland, 2004.

[24] V. Lefebure and J. Andreeva, "RefDB: the reference database for CMS Monte Carlo production," presented at the Computing in High Energy and Nuclear Physics (CHEP) Conf., La Jolla, CA, 2003.

[25] Workload Management System (WMS). (2002, Sep.) Definition of the architecture, technical plan and evaluation criteria for the resource co-allocation framework and mechanisms for parallel job partitioning. [Online]http://server11.infn.it/workload-grid/docs/DataGrid-01-D1.4-0127-1_0.pdf

[26] G. Malkin, "The Routing Internet Protocol Version 2," Xylogics, Inc., Burlington, MA, 1995.

[27] EGEE (Enabling Grid for E-science in Europe) Middleware Architecture. [Online]. Available: https://edms.cern.ch/document/476 451

[28] R. Pordes *et al.*, "The open science grid," presented at the Computing in High Energy and Nuclear Physics (CHEP) Conf., Interlaken, Switzerland, 2004.