FERMILAB-PUB-06-437-E

CDF II Production Farm Project

A. Baranovski [a], D. Benjamin [b], G. Cooper [a], S. Farrington [c], K. Genser [a], S. Hou [d], T. Hsieh [d], A. Kotwal [b], E. Lipeles [e], P. Murat [a], M. Norman [e], A. Robson [f], I. Sfiligoi [g], R. Snider [a], B. Stelzer [h], J. Syu [a], S. Timm [a], E. Vataga [i,*], S. Wolbers [a], D. Zhang [a]

[a] *Fermi National Accelerator Laboratory, Batavia, IL 60510, USA*

[b] *Duke University, Durham, NC 27708, USA*

[c] *Liverpool L69 7ZE, United Kingdom*

[d] *Institute of Physics, Academia Sinica, Taipei, Taiwan 11529, Republic of China*

[e] *University of California at San Diego, La Jolla, CA 92093, USA*

[f] *Glasgow University, Glasgow G12 8QQ, United Kingdom*

[g] *Istituto Nazionale di Fisica Nucleare, I-00044 Frascati, Italy*

[h] *University of California at Los Angeles, Los Angeles, CA 90095, USA*

[i] *University of New Mexico, Albuquerque, NM 87131, USA*

Abstract

We describe the architecture and discuss our operational experience in running the off-line reconstruction farm of the CDFII experiment. The Linux PC-based farm performs a wide set of tasks, ranging from producing calibrations and primary event reconstruction to large scale ntuple production. The farm control software uses a standard Condor toolkit [1] and the data handling part is based on SAM (Sequential Access via Metadata [2]) software. During its lifetime, the CDFII experiment will integrate a large amount of data (several petabytes) and the data processing chain is one of the key components of the successful physics program of the experiment.

Key words: Data management, Farms
PACS: 07.05.Fb, 07.05.Kf

1 Introduction

The CDFII experiment started collecting data in 2000. The peak rate for data recording currently reaches 40 MB/s and further increases are

* Corresponding author.
Email address: vataga@fnal.gov (E. Vataga).

expected after an upgrade of the Data Acquisition system in 2006. As of March 2006, the total volume of raw data collected by the experiment amounts to 0.5 Pbytes. This number will increase by a factor of 4 by the year 2009. That means that data management constitutes a challenging part of the experiment in terms of both computing and human resources. During the course of the experiment the CDF Production Farm was upgraded several times taking into account these growing requirements. We describe the main hardware and software components of the Production Farm, a framework for monitoring and recovery, a spectrum of Farm Projects and recent operational experience.

2 Main Components

Hardware and Software
High-throughput Linux clusters are now widely used in HEP for event reconstruction and analysis. The Production Farm consists of about 150 dual CPU PC's with a total computing power of the order of 800 GHz. It corresponds to about 15% of the experiment's computing resources inside Fermi National Accelerator Laboratory. The main software components of farm architecture are summarized in two acronyms: SAM and CAF.

SAM [2] stands for Sequential data Access via Metadata. It is a distributed data handling system managed by a database and cataloguing system based on CORBA, which provides a sophisticated set of tools for storing, cataloguing and delivering data both inside and outside of FNAL. The concept of metadata [3] has been adopted by many major collider experiments such as ATLAS, BaBar, CMS, D0, LHCb.

CAF (CDF Analysis Farm) is software and control system used by all farms inside the experiment. It is based on the Condor batch job system [1] with customized submission infrastructure and monitoring [4,5].

The important feature of Production Farm is its similarity to other farms in use by CDF experiment. The fact that Production Farm is "just another CAF", although for a single user, makes borders between farms flexible. In fact, a fraction of CPU from the Analysis Farms can be easily switched for reconstruction use. The Production Farm is based on network-distributed architecture; the main components of the system are shown on a schematic diagram in Fig.1. More details on the Farm Architecture and its evolution in time can be found in [8].

Data Flow
Raw data from the CDF detector proceed in the following way. Proton-antiproton collision information is filtered by a 3-level trigger system. Trigger output is managed by a data-logging sub-system. In eight separate streams the data are written to robotic tape storage [6]. A tape robot is used for mass storage of raw data, reconstructed events, Monte Carlo samples and ntuples. The mass storage system has three major com-

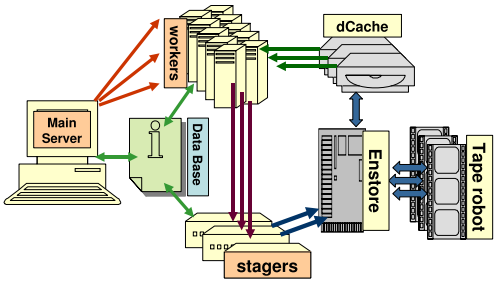

Fig. 1. Main components of CDFII Production Farm and their interface with Robotic Tape Storage and Oracle Data Base.

ponents: The Enstore system is used to access data on tape, dCache [7] serves as a front-end to mass storage, and PNFS software is used to map all files stored in the robot to a Unix-like name space. The Production Farm reads input raw data from dCache and writes reconstructed output directly to Enstore.

Production cycle

The main goal of the Production Farm is to make the recorded data available for physics analysis as soon as possible. After the time required to calculate and validate detector calibration and alignment parameters, the farm begins raw data processing. Starting from the input run range, we form a list of raw data files to be processed. That list is divided into units small enough to form independent jobs. This operation is done on the main farm server (see Fig.1). The main server provides multiple functions: SAM station, Condor head node, web server for monitoring. A prepared job enters the loop made of three schedulers: submitter, concatenator and uploader. The submitter goes through the list, selecting unprocessed or failed jobs. For a new job it performs two operations: it starts a SAM project, responsible for delivering raw data files, and a batch job in Condor, which divides the job into parallel processes and sends them to several worker nodes. Each worker receives a set of files, containing a binary executable and necessary libraries. SAM delivers a unique address of a raw data file. After that the file is copied locally and processed by the reconstruction program. As a result of reconstruction, the data are further split into physics stream according to a trigger mapping. There is an overlap of the order of 40% over 50 output streams. Before saving the results of reconstruction to Enstore, the farm collects all output in the intermediate buffers on dedicated stagers with a total of 24 TB of disk space. In order to optimize performance, files written to Enstore must have a size of 1-2 GB, hence some of the output data streams require concatenation. This operation is done on six stagers. The concatenator attempts to preserve the event order. Later the uploader copies the files to Enstore, and reconstructed data become available for analysis. All steps of processing heavily rely on information from Oracle Data Base (DB). Not only do worker nodes retrieve calibration and geometry constants from the DB, but also all reconstructed files, including intermediate output on worker nodes, are declared to the DB in order to avoid lost or duplicated events.

Control, Monitoring and Recovery

Different kind of failures are unavoidable in a system with the level of com-

plexity of Production Farm. Constant monitoring and recovery procedures have been the subject of much work in the farm maintenance. The emphasis was made on automatic recovery and remote web-based monitoring. Different schedulers control available resources, CPU load, data delivery, continuous concatenation and tape upload. Farm maintenance was significantly simplified by introducing wiki-based Farm Projects. Tikiwiki software[9] has proven to be an extremely efficient tool for web-based documentation. Moreover, its underlying database server keeps a history of all changes to the Farm Projects. Tiki pages with Project configurations give possibility to keep track of all existing projects, start or stop them, change resource sharing between projects, redirect output to another stager, forward execution to CAF instead of Farm without actually connecting to the main farm server.

3 Operational experience

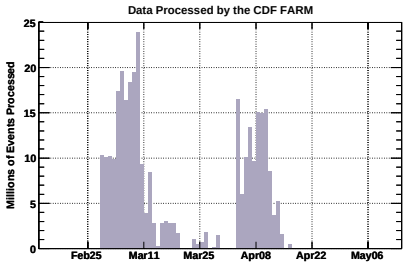

Fig. 2. Number of raw events processed daily by Production Farm in the period from mid-February to mid-May 2006.

Data Reconstruction
Fig.2 provides a summary of the Farm operation in 3 months preceding the conference. The peak raw data logging rate was of the order of 5M events per day. The rate of data processing was between 5M to 15M/day. During March processing Production Farm resources were augmented by additional CPU from the Analysis Farm, which allowed a production rate up to 24M/day (corresponding to output data volume about 3TByte/day). The operation was quite stable and data throughput capacity well exceeded raw data recording rate. The crash rate of reconstruction code has been of the order 10^{-8}. The Production Farm has sufficient resources to handle future increase of data volume and can accommodate eventual data reprocessing as well.

Ntuple Production and other usage of Farm
In order to take advantage of the efficient framework for data access, processing and monitoring and make full utilization of farm resources, we adapted the farm infrastructure to be usable for different kind of projects. This infrastructure includes 4 steps:

(1) creation of input blocks of data, which can be either raw or reconstructed data;
(2) job submission of unprocessed or failed sections
(3) merging of output files
(4) tape upload

In some cases, for example calibration or special run processing, last two steps are not necessary and can be skipped. Several physics groups in CDFII have chosen to use common

Standard ntuples (stntuples), which were thoroughly validated and developed to fit a broad range of physics analysises. Since March of 2006 ntuple production became a part of the Production Farm cycle, which significantly shortens the time between raw data recording and physics results. The number of Farm Projects grew from 10 in February to about 80 in May. Extended use of farm resources for ntuple production by a wider team of physicists and physics students provides more people able to operate the Farm.

4 Future plans

A major initiative in the Production Farm Project is to create more user-friendly interfaces and to overcome the barrier of an "expert-only" environment. In future we are planning to use a shift crew for Farm monitoring, which would guarantee greater stability and robustness in operations. By merging the Production Farm with CAFs we can increase CPU resources for reconstruction and centralized ntuple production and at the same time use the Farm Interface for Project handling and monitoring.

Acknowledgment

One of the authors (E.V.) wishes to thank the organizing committee of 10^{th} *Pisa Meeting* for their warm hospitality and Prof. Sally C. Seidel for support of this work.

References

[1] The Condor project page, http://www.cs.wisc.edu/condor

[2] Sequential data Access via Metadata http://d0db.fnal.gov/sam/

[3] "HEP Metadata Schema" GLAS-PPE/2005-04 http://ppewww.ph.gla.ac.uk/preprints

[4] "Computing for RunII at CDF", NIMA 502(2003) 386

[5] E. Lipeles, M. Neubauer, I. Sfiligoi, F.Wurthwein, "The Condor based CDF CAF", CHEP 2004 proceedings

[6] Fermilab Mass Storage System http://hppc.fnal.gov/enstore/

[7] dCache http://www-dcache.desy.de/

[8] J. Antos *et al.*, "Data production of a large Linux PC Farm for the CDF experiment", hep-ex/0603008 (2006).

[9] Tiki open-source Content Managment System (CMS) and Groupware http://doc.tikiwiki.org