The COMPASS Computing Farm Project

M. Lamanna

CERN, 1211 Geneva 23 (Switzerland) and INFN, Padriciano n.99 34012 Trieste (Italy)

Abstract. The COMPASS experiment at CERN is building a large facility for off-line computing in close collaboration with the CERN IT division: the COMPASS Computing Farm. The motivations, the experience, and the plans for deploying a 2,000 Spec INT95 computing farm are discussed. The main technical points are the use of Intel-based Linux PCs for both the I/O and the CPU intensive tasks, and the use of an object data base to store all the data.

INTRODUCTION

COMPASS is a fixed-target experiment with a wide physics programme at the CERN SPS. The apparatus will perform a number of different measurements, in different configurations, notably using both muon and hadron beams in the 100-300 GeV range at very high intensities(1).

The high data rate (over a few months per year) to be processed and the need for a flexible software environment to cope with the experiment's different measurements have pushed the COMPASS collaboration to design the off-line analysis software from scratch and to build a dedicated facility for the off-line computing, namely the COMPASS Computing Farm (CCF).

The COMPASS experiment has started commissioning the detector and taking data in the Summer 2000. All detector systems were present on the floor (although with a limited numbers of components) and a full chain from the front-end system, through the online data acquisition, to the off-line system was tested.

The off-line system should cope with a number of constraints, namely the high data acquisition rate (35 MB/s) and the very large data sample (10G events, 30 kB each, 300 TB/y) to be reconstructed almost online.

COMPASS decided to use the Central Data Recording (CDR) to record all data: the online system does not write events on tape at the experiment site, but sends them over a few km of dedicated network to the computer centre, where the CCF, the tape servers, and the corresponding high-speed tape drives are located.

The estimated computing power to reconstruct all the events at the speed of the data acquisition is 2,000 Spec INT95, which will be provided by some 100 PCs. The choice of network technology is Gigabit and Fast Ethernet. A disk pool of a few TB is being setup: presently most of the disk space is SCSI, but EIDE servers are being deployed.

The last requirement is the use of a data base layer integrated with a storage manager to handle the very large sample of data. The solutions for the data storage are still under evaluation, together with the development of the C++ reconstruction framework. During Summer 2000, the data have been kept in both native and Objectivity/DB format. CASTOR, a hierarchical storage manager being developed at CERN has been used to manage all data(3) (originally HPSS has been used).

THE DATA AND ANALYSIS MODELS

COMPASS decided to adopt a computing model such that all the data are reconstructed in a single reconstruction facility in parallel with the data taking.

COMPASS decided to write the bulk of the reconstruction programs in C++. The reconstruction and analysis program CORAL (COmpass Reconstruction and AnaLysis; *http://coral.cern.ch*) has been designed using object-oriented techniques. It uses an object data base called Objectivity/DB for both event and calibration data.

The large quantity of data (300 TB/year) means that all the data may not be disk resident at the same time. The limitation in total disk size can be made transparent with the use of a Hierarchical Storage Manager (HSM).

The data flow model should foresee different scenarios; the most relevant ones are the DAQ mode and the quasi-online reconstruction mode.

In the DAQ mode, events are written on the online farm event builder disks in raw format ($\approx$ 10 streams) to the CCF and then populate *in parallel* the data bases ("stage1"). The original files are not kept, and the corresponding data bases are moved to the HSM.

In the quasi-online reconstruction mode, the data follow the same path as in the DAQ mode, but they are read by the CPU clients when they are still on the disk pool. The events are dispatched to clients via the Objectivity/DB object server (AMS).

The model proposed for the CCF is a farm which uses as much as possible "commodity solutions", mainly PCs and mass-market network technologies.

At the beginning of 1998 we started with the idea of deploying a number of mid-range RISC servers for I/O plus a cluster of cheap PCs. In this document, *data servers* refer to the I/O intensive core of the farm, and *CPU clients* the number-crunching PCs (Fig.1).

This model is also well suited for the collaborating institutes; COMPASS institutes from Germany and Italy have shown interest and some prototypes already exist. These analysis farms will be fed with data samples from the CCF, to perform the final analysis of specific reactions; some of the data not present in these remote farms (RAW data samples, calibrations) can be accessed via wide-area network.

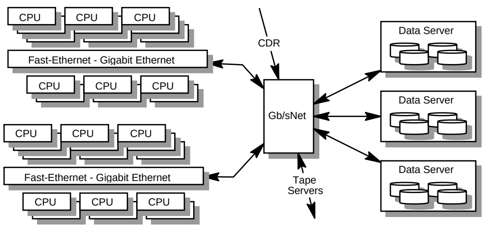

FIGURE 1. A schematic of the CCF. The CPU clients are on the left, on the right the data servers (holding the farm disk pool). The lines represent the main network connections and data flows: the CDR line represents the data flux from the experiment to the farm; the Tape Servers line represents the data exchange between the farm and the tape infrastructure.

THE COMPONENTS OF THE CCF

The decision to shape the CCF in a modular environment was taken very early, on the basis of both technical and economical considerations. The deployment of "independent" units in all CCF areas (instead of single high-end devices) best addresses the problem of the performance to price ratio for each area, gives the possibility to increase the configuration in small steps, to share hardware and resources within similar projects and, most importantly, to remove single points of failure.

The outcome is that all solutions should either be tested against others (e.g. Windows NT vs. Linux) or at least allow for possible future comparisons (i.e. use generic solutions or exploit only generic functionalities from a software component). Obviously, the tests should be performed if possible in a realistic environment to assess compatibility and scalability issues.

The CPU clients are Intel PCs. For the data server part, we tested DEC and Sun servers, and Intel PCs (2 Pentium II CPUs). Linux PCs (with Fast Ethernet interface) became eventually the solution also for data servers.

THE TESTS

An online farm prototype and the corresponding software to generate mock data with the appropriate characteristics (data rate, events size) have been developed to test the CCF. This farm has been set-up at the experiment site and uses the full network infrastructure which will be used in the experiment.

The CCF has been tested in the DAQ mode in a set-up with 11 data servers. The main result is that the behaviour scales with the number of data servers: the 11 data servers can sustain about 35 MB/s for many hours (Fig.2).

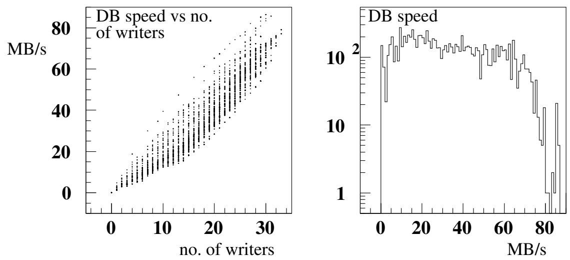

FIGURE 2. The "stage1" subsystem in the 35 MB DAQ test (Events conversion into objects in Objectivity/DB data bases). The rate of data in MB/s is shown as a function of the number of active parallel writers and as a histogram.

The sustained 35 MB/s figure means that at the same time the following activities coexist, all of them at 35 MB/s: the raw data are input into the CCF and written to the data servers disk; the same flux is input into the data bases; data bases are read from disk and copied to a remote system to mimic the HSM transfer. All quantities are measured over a test period of $\approx$ 8 hours.

The Quasi-Online mode is presently under test; configurations with 5 data servers and about 10 MB/s are run to test the behaviour of the multi-threaded version of the AMS (v.5.2.1). The system is stable but the AMS behaviour is not yet satisfactory because of time-out problems, which could be recovered at the expense of resubmitting some reconstruction jobs.

The calibration data base (CDB) is an important component of the CORAL, because it will be in charge to distribute calibration constants in a consistent way to all nodes processing data in parallel. The library is also in evolution (the CERN port of the original BaBar calibration data base is being completely rewritten)(5). To test the functionality, the performances, and the implementation, we perform dedicated tests (using CORAL clients).

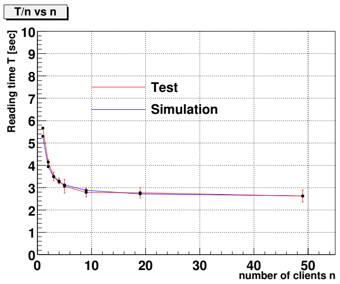

FIGURE 3. The plot shows the results of a set of tests of CDB access, compared with a simulation. The mean reading time per client vs. the number of clients (up to 50) is plotted.

In Fig. 3, the performances of the CDB are shown as measured versus the number of parallel clients. The data base access speed is $\approx$75% of the theoretical network speed (7-8 MB/s; for a Fast Ethernet network AMS server). The measurement is compared with some preliminary results of a simulation made using the MONARC toolkit(4).

PILOT RUN

All the data have been recorded successfully and made available to the collaboration using CASTOR (more than 1 TB of data for detector debugging).

The observed data base input rate is consistent with the previous tests. Since the actual data rate from the experiment was lower than the design value (due to the detector commissioning activities and lower number of channels), this was achieved accumulating data for hours and then processing them at the maximum speed (Fig. 4): the speed per server is consistent with the previous tests.

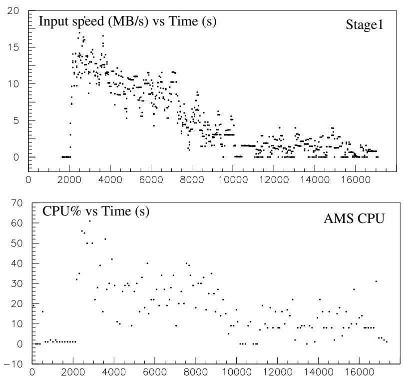

FIGURE 4. The data base input speed vs. time is shown (Stage1). The process is started after a period of accumulation of data; rapidly all the data are processed and the speed becomes equal to the input one. The corresponding AMS CPU consumption is shown.

All the chain down to a scan of the data (similar to quasi-online processing) has been demonstrated with real data. The performances are still somewhat poor as in the mock data tests (some bugs in the Objectivity/DB have been pinned down since then and our code and data organisation are also being improved).

CONCLUSIONS

The mock data tests gave us the possibility to select the appropriate hardware for every option, to shape the architecture and the control software in a modular way, and to test many prototypes in different realistic conditions.

The main message from the 2000 run is that the control software is adequate also for data acquisition. The mock data results have been cross-checked with the experience with the real data; no major new problem has been found.

The experience with the Objectivity/DB package is rather positive, but open problems remain. The usage of Objectivity/DB within the CORAL framework is well-understood and sufficiently isolated to consider it a separate interchangeable component.

The experience with CASTOR was very positive and the replacement of HPSS straightforward. The control software is still being simplified giving to CASTOR full responsibility of the various data pools present in the system. The CASTOR-Objectivity/DB interface was successfully used for the first time.

I would like to acknowledge the support and the help of many groups in the CERN IT division (especially the PDP and the DB groups), the COMPASS Collaboration, and the COMPASS Offline Group.

REFERENCES

1. COMPASS, COMPASS Proposal, CERN/SPSLC/96-14, SPSLC/P297, March 1, 1996; COMPASS, Addendum 1, CERN/SPSLC/96-30, SPSLC/P297 Add. 1, May 20, 1996.
2. M. Lamanna "The COMPASS Computing Farm project", Proceedings of the CHEP 2000 Conference, Padova, February 2000, edited by M. Mazzucato, p.576.
3. CASTOR, developed in the IT/PDP group, *http://wwwinfo.cern.ch/pdp/castor/* the HSM interface has been developed in the IT/DB group.
4. I. LeGrand "Simulating Distributed Systems", contribution to this conference.
5. Calibration DB, developed in the IT/DB group, *http://wwwinfo.cern.ch/db/objectivity/docs/conditionsdb.*