

The Level-3 Trigger at the CDF Experiment at Tevatron Run II

Y.S. Chung¹, G. De Lentdecker¹, S. Demers¹, B.Y. Han¹, B. Kilminster¹, J. Lee¹, K. McFarland¹, A. Vaiciulis¹, F. Azfar², T. Huffman², T. Akimoto³, H. Kobayashi³, H. Matsunaga³, M. Shimojima³, K. Tollefson⁴, D. Torretta⁵, D. Waters⁶

Abstract—We describe the filtering and analysis software running in the CDF Run II Level-3 trigger as well as the client and event driven data hub system. These systems constitute critical components of the data acquisition system of the CDF detector. The Level-3 trigger is responsible for reconstructing the events and forming the final trigger decision. The Consumer-Server Logger (CSL) system receives the accepted physics events from multiple connections and writes them to disk in multiple streams while distributing them to online analysis programs (consumers). Since 2001 the system has been running successfully at high throughput rates: the Level-3 trigger reduces the event rate from 350 Hz to about 100 Hz with an average event size of 150 kB while the CSL is able to process up to 22 MB/s of constant event logging. We describe these systems in detail and report on their current performance. We briefly outline upgrade plans for the remainder of Tevatron Run II.

Index Terms—Hadron collider, trigger, software, data hub

I. INTRODUCTION

THE Collider Detector at Fermilab (CDF) [1] is a general purpose particle detector at the 1.96 TeV $p\bar{p}$ Tevatron collider at the Fermi National Accelerator Laboratory. Since the end of Run I (1992-1995) the detector, as well as the accelerator complex, have undergone significant improvements in the pursuit of a better understanding of physics at the smallest scales. Run IIa of the Tevatron is planned to deliver 2 fb^{-1} , which is twenty times the integrated luminosity of Run I.

CDF began taking data for Tevatron Run II in July 2001 and has collected so far over 500 pb^{-1} of data. Based on Run I experience, the CDF Collaboration has chosen to preserve a three-level trigger hierarchy, where each succeeding level filters events on the basis of increasingly refined reconstructions of objects within the event. In this way the first two trigger levels reduce the event rate from 2.5 MHz, the maximum bunch crossing rate, to about 350 Hz which is the design input rate to the Event Builder and the Level-3 trigger systems.

While Level-1 (L1) and Level-2 (L2) triggers are based on either analog or fast digitized data of parts of the detector,

the Level-3 (L3) trigger makes use of fully digitized information from the whole detector. The Level-3 trigger, which is implemented as a PC farm, processes the complete event record and reconstructs the event following given algorithms to further reduce the event rate to roughly 100 Hz. The accepted events are collected and sent to the CSL, which logs the events to persistent storage for further offline analysis, while distributing a fraction of them to online consumers for real-time monitoring purposes.

The extended running plans which call for Tevatron operations until 2009 (Run IIb) do not now include the option for 132 ns bunch crossing operations but will keep the present bunch spacing of 396 ns instead. Therefore at the expected Run IIb peak luminosity of $3 \times 10^{32}\text{ cm}^{-2}\text{s}^{-1}$, the number of interactions per crossing will be ten, which is well beyond what was envisaged for Run II. This implies that the average data size will increase substantially, and the combinatorics will grow in processing multi-object triggers. This future running environment will be very challenging for the detectors as well as the Data Acquisition System (DAQ).

II. THE CDF TRIGGER SYSTEM

The CDF Run II trigger, shown in Fig. 1, has a three level architecture with each level providing a rate reduction sufficient to allow for processing in the next level with minimal dead-time.

The Level-1 system is a synchronous 40 stage pipeline. When an event is accepted by the Level-1 trigger, all data is moved to one of four Level-2 buffers in the front-end electronics, and trigger data is sent to the asynchronous Level-2 system. Here some limited event reconstruction is performed. The Level-1 and Level-2 output rates are $\sim 30\text{ kHz}$ and $\sim 350\text{ Hz}$ respectively. When an event is accepted by Level-2, its data become available to the Event Builder (EVB) which assembles the events from 16 different readout locations and sends them to the Level-3 trigger where the final event filtering is done, reducing the rate to $\sim 100\text{ Hz}$. Finally these events are written to tape for offline analysis. The system responsible for this final step is the CSL, described in section IV below.

III. THE LEVEL-3 TRIGGER

For Run II, the CDF Collaboration adopted a processor based filtering mechanism for the Level-3 trigger using the C++

¹ University of Rochester, USA

² Oxford University, UK

³ University of Tsukuba, Japan

⁴ Michigan State University, USA

⁵ Fermi National Accelerator Laboratory, USA

⁶ University College London, UK

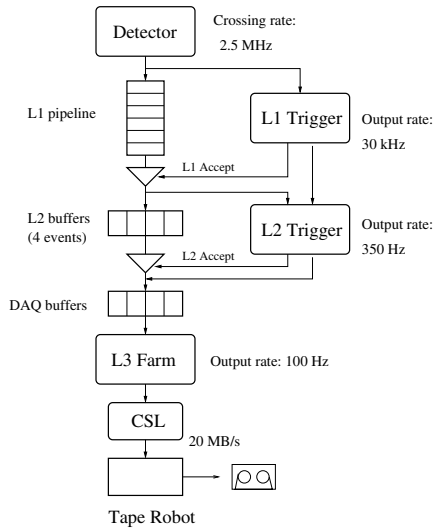


Fig. 1. The architecture of the three level trigger system of CDF at Tevatron Run II.

programming language. The event reconstruction and selection is performed on a Level-3 farm made up of approximately 256 commodity dual processor Linux PC's, called processor nodes, running at clock speeds of 1.0 - 2.6 GHz. The total CPU power of the Level-3 farm amounts to 850 GHz. Event data enter the Event Builder through Scanner CPU's and are sent through the event network, which is an ATM switch, to the Level-3 farm. A node receiving event data from the ATM switch is called a converter node which distributes them to the processor nodes. The processor nodes run the filter algorithms on the events and the accepted events are collected by the output nodes which direct them to the CSL. The Level-3 farm is subdivided into 16 sub-farms, each sub-farm consisting of one converter node and 16 processor nodes. Each pair of sub-farms is connected to an output node.

A. The Level-3 Trigger framework

To achieve a sufficiently small processing time, less than $\approx 1.5 \text{ GHz} \cdot \text{s}$ at an instantaneous luminosity of $1 \times 10^{32} \text{ cm}^{-2} \text{s}^{-1}$, the Level-3 algorithms use Level-3 specific reconstruction and filtering code. In addition two desirable goals in Level-3 are consistency with the offline reconstruction and accessible, reproducible code.

In order to ensure a high level of agreement between results in Level-3 and offline analysis, the Level-3 code is based on the offline production reconstruction code. Level-3-specific reconstruction and selection code resides in the offline CVS [2] repository, along with the offline production code. Level-3 code versioning is based on the offline code versioning scheme, with a list of specific patches also residing in the CVS repository. This scheme has the following advantages:

- Level-3 code versions can be updated on a shorter time-scale than offline releases simply by updating a patch

list, which is automatically applied in the building of new Level-3 executables;

- anyone with access to the CDF software can remake any current or previously used Level-3 executable;
- CVS provides a complete history of the code and Level-3 specific patches that have been applied.

For Run II, the CDF Collaboration has decided to have dedicated trigger paths defined by analysis datasets. For example, if an event passes a muon trigger at Level-1 then only Level-2 triggers which specify that particular Level-1 muon as prerequisite are considered. This implies that Level-3 trigger paths derive from Level-1 and Level-2 trigger paths. The idea of dedicated trigger paths is reflected in the design and implementation of the "trigger table", a description of the complete trigger configuration stored in a relational database schema. For each Level-3 trigger path the trigger table contains the Level-1 and Level-2 trigger prerequisites, the list of reconstruction and filters to be run, their sequence as well as the physics cuts associated with a given trigger path. The interface between the Level-3 executable and the trigger table consists of a TCL [3] script. It is important to note that all events accepted by the Level-2 trigger are fully reconstructed at Level-3 before applying the Level-3 cuts. Figure 2 shows a schematic view of the Level-3 framework.

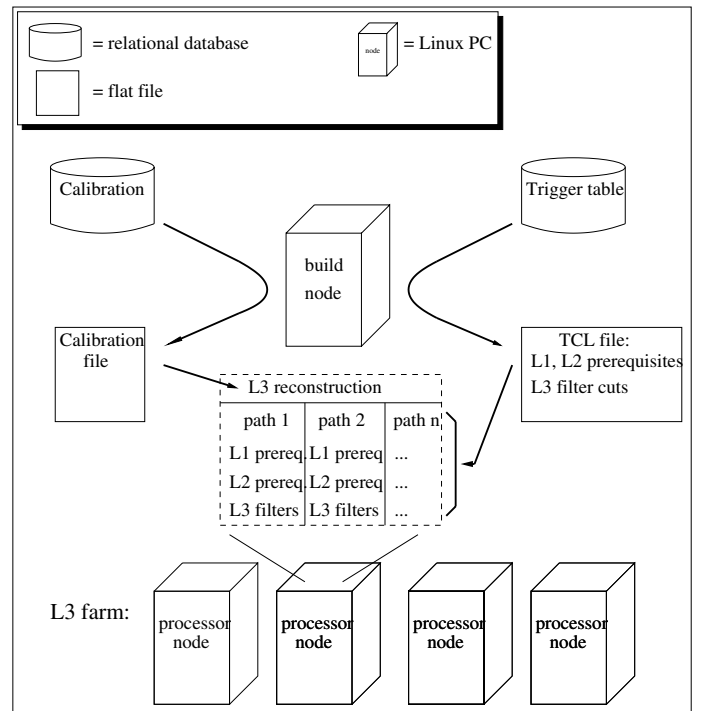


Fig. 2. Level-3 framework: at the beginning of data taking a dedicated PC (build node) sends a text version of the calibration database (newly generated in case new constants are available), to all the PC's of the Level-3 farm. The TCL [3] script containing the Level-1 and Level-2 prerequisites and the cuts to be applied by the Level-3 filters, and the Level-3 executable itself, are also distributed if not already cached on the processor nodes. See text for further details.

The Level-3 executable is built on a dedicated Linux PC,

through a largely automated procedure. Using a database and CVS repository we are able to create a unique and reproducible combination of executable, TCL and calibration files. The executable, the TCL file, the shared libraries and the calibrations are stored as tarred files and copied automatically from the build node to the Level-3 processor nodes at the beginning of data taking. Calibration tar files are automatically re-created at the start of a run if new calibrations are available for any sub-detector. The use of a text calibration database distributed to every processor node avoids latencies associated with multiple direct connections to a single relational database instance at the start of a run.

B. The Level-3 Trigger performance

As the instantaneous luminosity of the Tevatron continues to grow, the average number of interactions per bunch crossing increases. This implies that the average event size as well as the combinatorics will increase substantially, requiring longer processing time to accept or discard the events. Consequently the processing time and the event size are the main Level-3 issues for Run IIb.

1) *Processing time*: Figure 3 shows the average Level-3 processing time (in units of GHz – s) for each event as a function of the Tevatron instantaneous luminosity. At a luminosity of $1 \times 10^{32} \text{ cm}^{-2} \text{ s}^{-1}$ the processing time is $\approx 1.4 \text{ GHz} - \text{s}$, but rapidly increases at higher luminosity. We are currently reducing the processing time by optimizing the Level-3 executable. This optimization will reduce the processing time by $\sim 25\%$. This code optimization is also combined with the progressive

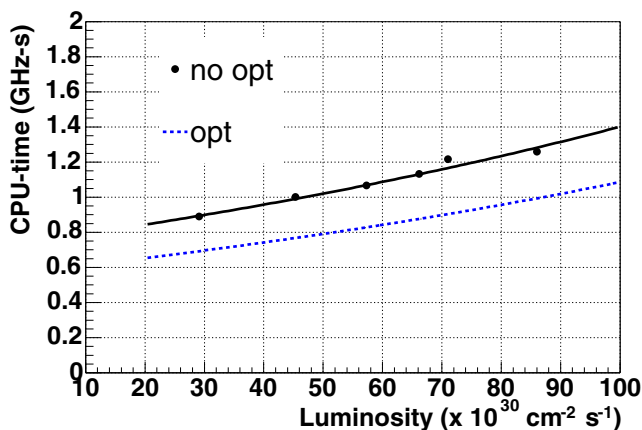


Fig. 3. Level-3 processing time on a 1.0 GHz PC as a function of the Tevatron instantaneous luminosity for the non-optimized (full line) and optimized executable (dashed line).

replacement of the oldest processors of the Level-3 farm by new 2.6 GHz PC's. This should also increase the CPU of the farm by $\sim 25\%$. Additional flexibility is available through the ability to adjust physics thresholds in the reconstruction (especially the track reconstruction), although the effects on trigger efficiencies for various physics processes need to be carefully monitored in this case.

2) *Event size*: The present bandwidth of the CSL is limited to 22 MB/s. With a Level-3 accept rate approaching 100 Hz at the highest luminosity of $1 \times 10^{32} \text{ cm}^{-2} \text{ s}^{-1}$, we have to keep the event size below $\sim 200 \text{ kB}$. Recently the effective bandwidth of the CDF data acquisition system was increased by keeping only the compressed forms of several large raw data banks, namely those coming from the silicon and central tracking detectors. This results in an event size decrease from $\sim 220 \text{ kB}$ to $\sim 150 \text{ kB}$ at a luminosity of $5.0 \times 10^{31} \text{ cm}^{-2} \text{ s}^{-1}$. Figure 4 shows the average event size as a function of the Tevatron instantaneous luminosity. To avoid reaching the CSL limit at luminosities higher than $1 \times 10^{32} \text{ cm}^{-2} \text{ s}^{-1}$, we will now compress the calorimeter data, which should reduce the rate by 1 MB/s. In addition the CSL will soon be improved before the long term CSL upgrade will occur in 2005. Both issues are discussed in the next section.

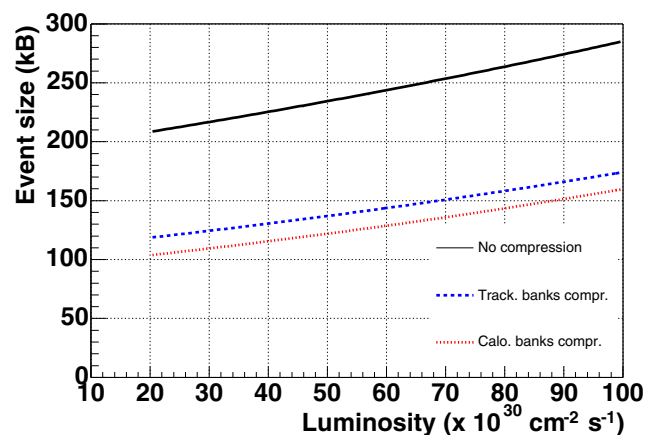


Fig. 4. Event size at Level-3, before compressing the tracking banks (full line), after compressing the tracking banks (dashed line) and adding the compression of the calorimeter banks (dotted line), as a function of the Tevatron instantaneous luminosity.

IV. THE CONSUMER-SERVER/LOGGER

The events accepted by the Level-3 trigger are passed through the CSL for future offline and online processing. The decision made at Level-3 is also used by the CSL to sort the data, placing accepted events into categories useful for prioritizing the data analysis, for offline monitoring or for immediate analysis in the case of rare events. The importance of the data splitting task of the online system, which results in early identification of small samples of events for further analysis, can be easily understood in the context of the multi-petabyte data sample that CDF will collect during Run II.

Events processed by the Level-3 farm are received by the CSL from a Fast-Ethernet switch via four Fast-Ethernet ports. The CSL runs on a single SGI 2200 Server with four 400 MHz R12000 processors. It logs data to FiberChannel-attached dual ported RAID arrays. The files are then read out from the RAID arrays and sent by single-mode fiber to a robot which

writes data to a 1 PByte tape archive. A 3 TByte online RAID system allows CDF to buffer data for at least eight hours if the communication with the tape robot is down.

The CSL consists of independently running C programs which communicate through shared memory segments and message queues. Typical operation is as follows (see Figure 5): A receiver process waits for connections from an Level-3 output node. After receiving an event, the receiver writes it to one of 150 shared memory buffers and puts a message on the logger queue representing the buffer. The logger, after finding a buffer message in its queue, logs the event to the appropriate data files and puts the message onto the distributor queue. The distributor either puts the buffer onto the consumer-send queue or returns the message to the receiver queue. The consumer-send process looks for a message on its queue and sends the corresponding event to the appropriate consumers. After this event has been fully processed the message is returned to the receiver queue so that another event may be received.

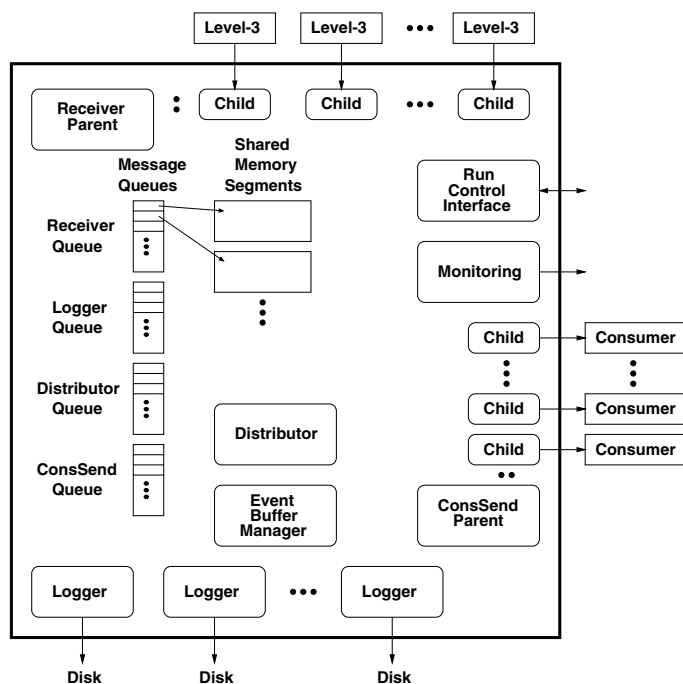


Fig. 5. Software design of the Consumer-Server/Logger

The architecture of the currently operating CSL with directly attached disk arrays is not scalable, in large part because of limits on the number of disks that can be attached to a single system. In addition there is only a single, dedicated logger process. This process is responsible for writing all data onto disk.

A. Parallel logger

The parallel logger is a software design which can be relatively quickly implemented which allows for multiple logging processes to run in parallel during physics data taking. There is one master logger to keep track of the begin/end of the run,

the run sections, etc. and several slave loggers. Tests of the new code have shown that the system can run at an overall throughput rate exceeding 35 MB/s. The bandwidth is now limited by the CPU, not by the network or disk. The parallel logger should be fully operational at the end of the present Tevatron shutdown and be used until summer 2005.

B. CSL upgrade

The CSL hardware upgrade will increase the CSL bandwidth to 60 MB/s, which is the CDF Run IIb bandwidth goal to write data onto tape. For the CSL upgrade we propose to move to an array of network-attached buffer disks. These disks arrays are purposely modeled on the network-attached arrays chosen for the CDF central analysis facility in order to share maintenance and upgrade hardware.

To test the feasibility of network-attached disk arrays we have set up a prototype system consisting of the following:

- two 3Ware 7500-8 IDE RAID controllers which are capable of forming RAID volumes from inexpensive IDE disks
- 3 TB of IDE disk space distributed over 200 GB Maxtor drives
- The system has been configured into RAID 10 arrays with the XFS filesystem

Both RAID level 5 (data and parity striped across multiple disks) and level 10 (two copies of data striped across disks) were tested. RAID level 10 can sustain multiple disk failures, at a cost of making available only 50% of the disk space. First the bit error rate was tested by repeatedly writing several 1 GB files, checking after each write if any bit was written incorrectly. With RAID level 10 we measured no bit errors while writing 8146 1 GB files. With RAID level 5 we measured two bit errors while writing 3804 1 GB files. Although RAID level 10 has been chosen for reason of write speed and fault tolerance, the RAID level 5 bit error rate of one bit per 10 million events is most likely sufficiently low.

We also performed extensive testing of the prototype with the goal of demonstrating that the network attached disk could provide the needed bandwidth. The load pattern on the device is complex during data-taking: the CSL logs data at roughly a constant rate to disk, using multiple 'streams' into which the raw data is stored. Each stream is written independently, although not necessarily to different disks. However when files are completed, they are typically read back by the tape writing system at a fixed rate much higher than the write rate for a single stream, because the number of tape drives allocated for data logging is smaller than the number of data streams. Therefore the new CSL design must be able to demonstrate sufficient write rate under a high instantaneous read load.

The prototype system was evaluated by measuring the disk reading and writing performance on the network attached disk performing simulated CSL event transitions. Figure 6 shows the total read and write bandwidth of the system under different configurations in this test which maps out a space of maximum

read rate versus maximum write rate. In all cases, performance is well in excess of requirements for the upgraded CSL.

bandwidth. A prototype has been successfully tested, meeting all specifications for the Run II CSL upgrade.

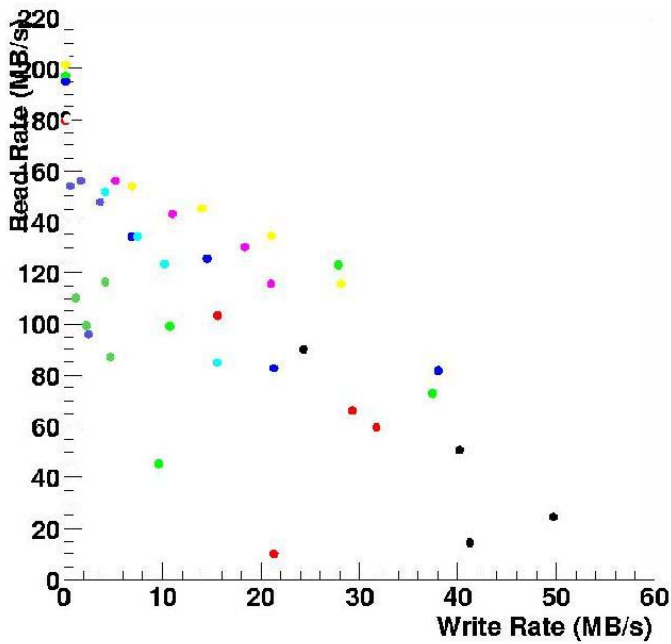


Fig. 6. Summed write rates and maximum sustained read rates for different configurations.

V. CONCLUSIONS

This paper presents the current status of the Level-3 Trigger and the Consumer-Server/Logger of the CDF experiment at the Tevatron Run II. The accelerator has nearly reached its Run II design luminosity and continues to improve its output. So far, the Level-3 and the Consumer-Server/Logger of CDF have successfully run at very high throughput rates and exceeded the Run IIa specifications. For Run IIb the CDF goal is to increase the bandwidth of data being written to tape to 60 MB/s. In addition the extended running plans with an average number of interactions per bunch crossing well beyond what was envisaged for Run II will be very challenging for the detector and the data acquisition system.

To cope with the increased complexity of the events the Level-3 executable is being optimized to reduce the processing time by 25%. This optimization is combined with the progressive replacement of the oldest PC's of the Level-3 farm, raising soon the total CPU of the farm by 25%.

Because the current bandwidth of the CSL is limited to 22 MB/s, we are working on the compression of several large raw banks. This effort to reduce the event size will be combined with the introduction of the CSL parallel logger which should increase the bandwidth limit to 35 MB/s. Finally in 2005, the CSL will be upgrade, using an architecture with network-attached buffer disks. This design is fully scalable and makes it more easy to accommodate further increase of the CSL

REFERENCES

- [1] CDF Collaboration, F. Abe *et al.*, Nuclear Instrum. Methods **271**, 387 (1988).
CDF Collaboration, F. Abe *et al.* Phys. Rev. D **50**, 2966 (1994).
- [2] P. Cedeqvist *et al.*, Version Management with CVS, cvs.texinfo released with CVS 1.8.1, see http://w4.lns.cornell.edu/public/COMP/info/cvs/cvs_toc.html
- [3] Brent B. Welch, Practical Programming in Tcl and Tk, Prentice Hall PTR, 1995