A Commodity Solution Based High Data Rate Asynchronous Trigger System for Hadron Collider Experiments

Michael Wang and Jin Yuan Wu
(for the BTeV collaboration)

***Abstract*— We show how commodity CPU, networking and memory components greatly simplify a sophisticated trigger system designed for the demanding environment of a high data rate hadron collider experiment operating at 2×10^{32} cm^{-2}s^{-1}. The trigger system is based on an 8-fold way, 3-level architecture capable of processing over 22 million data channels at a crossing rate of 2.5 MHz with a mean value of 6 interactions per crossing. Although originally designed for a B-physics experiment at the Tevatron, the flexible and highly scalable nature of the design should also be relevant to other next generation hadron collider experiments such as those at the LHC.**

I. Introduction

Modern day hadron collider experiments are faced with similar challenges of designing trigger systems that can achieve high efficiencies for physics signals of interest while maintaining high levels of background rejection within the context of a high data rate environment. In this paper, we describe a trigger system that was designed precisely for this purpose. We present the baseline design of a trigger system for the BTeV experiment which has been scrutinized carefully in major reviews conducted by the U.S. Department of Energy. Various components of this design have been prototyped and tested and are described in detail in the BTeV Technical Design Report and references therein [1]. Components for which prototypes exist will be indicated below and references provided whenever possible. In addition to the baseline, we also present a major proposed change to the architecture for which detailed specifications were being written. Unfortunately BTeV was abruptly cancelled earlier last year after successfully passing critical reviews [2]. Many of the lessons learned and ideas introduced in designing this trigger, however, should be useful to other experiments facing similar challenges.

II. The BTeV Experiment

BTeV is a collider experiment designed to acquire as many $B\overline{B}$ events as possible in order to probe the subtle differences between B and $\overline{B}$ mesons for a better understanding of the cosmic asymmetry between matter and anti-matter. It was meant to operate in the C0 interaction region of Fermilab's Tevatron at a luminosity of 2×10^{32} cm^{-2}s^{-1} corresponding to a mean value of 6 interactions per beam crossing at a crossing rate of 2.5 MHz. BTeV takes advantage of the large $b\overline{b}$ cross sections at hadron colliders ($\sigma(b\overline{b}) \sim 100$ μb) and the unique characteristics of hadronic b production in the forward region [3], [4].

The BTeV detector is a unique forward spectrometer consisting of a muon detector, a Ring Imaging Cherenkov detector, and an electromagnetic calorimeter for particle ID, and a combination of straw tubes and Si-microstrips for charged particle tracking [1]. The centerpiece of the BTeV detector is a 120 cm long 30 station Si-pixel vertex detector centered at the C0 collision point and immersed in a 1.5 Tesla dipole field. Each pixel station has over 7.6×10^5 rectangular pixels measuring 50×400 $\mu\mathrm{m}^2$ for a total of over 22×10^6 pixels in the full detector. As will be seen below, the pixel detector plays a crucial role in the BTeV trigger.

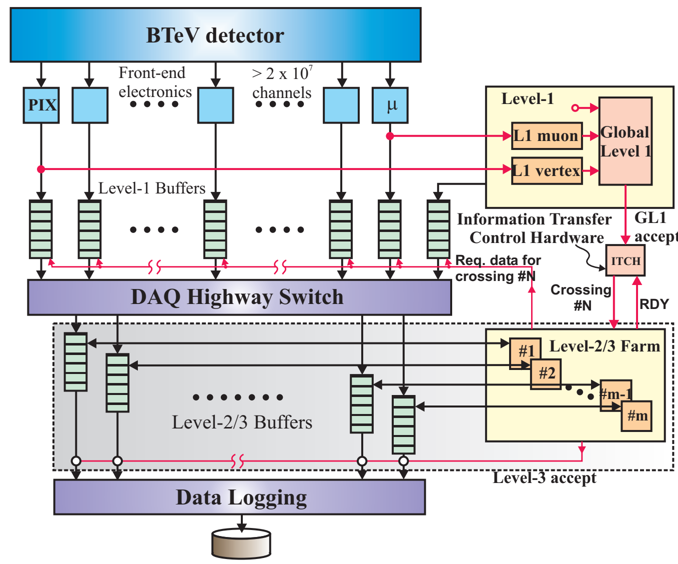

Fig. 1. BTeV's three-level trigger architecture.

III. Overview of the BTeV Trigger

A. Trigger Strategy

Unstable B mesons are identified by their decay products which form a V-shaped prong a short distance (a few mm at Tevatron energies) from the $p\overline{p}$ collision point where they are created. The problem facing the BTeV trigger is to sift through every single beam crossing in order to detect tracks from

Manuscript received June 17, 2005; revised April 14, 2006
M. Wang is with Fermilab, Batavia, IL 60510, U.S.A.
J. Y. Wu is with Fermilab, Batavia, IL 60510, U.S.A.

the decay products of rarely produced B mesons (~1/1000 $p\bar{p}$ collisions at the Tevatron) in the presence of high track multiplicities.

In order to do this, BTeV employs a three level hierarchical trigger architecture shown in Fig. 1 that is typical of many High Energy Physics (HEP) experiments [5]. In such architectures, processing at each stage reduces the input rate providing subsequent stages more time to perform a more detailed analysis of the data to separate the interesting from the background events.

In BTeV's case, the lowest level which is referred to as Level 1 (L1), examines every crossing to find interesting events while data from the full detector is temporarily stored in L1 buffers. L1 reduces the input rate by 50× resulting in 50KHz going into the Level 2/3 trigger (L2/3). L2/3, which is implemented on a farm of commodity PC's with each node's onboard memory playing the role of a L2/3 buffer, performs a more detailed analysis on each event using data from a larger subset of the detector. The processing performed online at this stage corresponds to the CPU-intensive reconstruction traditionally done offline in other HEP experiments. L2/3 further reduces the data rate by 20× resulting in an output rate of 2.5 KHz written into archival storage.

What sets the BTeV trigger apart is the amount of processing applied at L1 in which few sacrifices are made in the sophistication of the algorithms. This is unlike other HEP experiments which are forced to use fast but crude algorithms on dedicated hardware for the lowest level triggers due to severe time constraints at this stage. BTeV's L1 trigger is relatively decoupled from such constraints due to the asynchronous nature of its data acquisition (DAQ) system in which data from the entire detector is read out at the full crossing rate of 2.5MHz and stored in huge L1 buffers based on low cost commodity DRAM [6]. This makes it practical to achieve a large enough memory capacity corresponding to three orders of magnitude more than the average L1 latency allowing every single crossing to be processed by L1. The superior pattern recognition provided by the pixel detector is also a tremendous advantage since it greatly simplifies the L1 trigger algorithm.

In the next section, we focus our attention on a brief description of the L1 trigger and skip over L2/3 since this consists simply of a cluster of commodity Linux PC's.

B. Level 1 Trigger

A simplified block diagram of BTeV's L1 trigger is shown in Fig. 2. Pre-processed pixel data from three adjacent stations are sent to FPGA (Field Programmable Gate Arrays) based custom hardware which perform pattern recognition to find the beginning and ending segments of tracks referred to as triplets. Since these segment finders deal with only a portion of the full pixel detector, all triplets belonging to the same crossing are routed through an event building switch to one processor in the track/vertex farm consisting of programmable processors running C-code. This farm performs the second stage of the L1 trigger algorithm matching beginning and ending track segments to reconstruct complete tracks which are in turn used to locate the primary interaction vertices to see if any

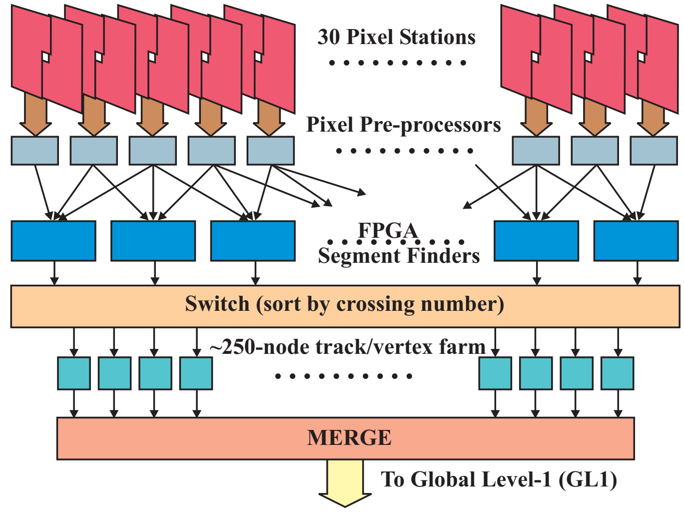

Fig. 2. BTeV Level 1 trigger.

remaining tracks are detached from these vertices by a given amount. The results are then sent to Global Level 1 and if there are at least two detached tracks meeting certain criteria, going in the same z direction, the event is considered a B physics candidate and sent to L2/3. A more detailed description of the algorithm can be found in Ref. [7], [8]

IV. BTeV Trigger Architecture

The three-level trigger system shown in Fig. 3 consists of eight parallel data pathways called "highways". Data from each beam crossing is distributed in a data-driven round-robin sequence to one of these eight highways, each of which forms a complete and independent three-level trigger system. This reduces the full data rate from the detector by $1/8$ into each highway allowing the use of cheap components such as commercially available ethernet switches.

Detector data are digitized and zero-suppressed by front-end electronics and sent via high-speed copper links to data-combiner boards (DCB) located in the collision hall which serve as a common interface for all detector subsystems to the DAQ [1], [6]. Each DCB multiplexes data packets from 24 inputs to 1 of 8 outputs destined for one of the 8 data highways. The smaller input packets are merged into larger ones allowing more efficient use of network switch bandwidth in the later stages. DCB's are arranged in groups of 12 to form a total of 48 DCB subsystems. The backplane in a subsystem routes data from the 12×8 output ports on the DCB's to optical transmitters that send the data via 8 12-channel optical links over 30m from the collision hall to the counting room where they are received by the L1 buffers in a highway.

For each highway, data from all detector components are sent from the optical receivers to L1 buffers for temporary storage as trigger decisions are made. An exception is the pixel detector whose data goes through the pixel pre-processors before being stored in L1 buffers. Data from the pixel pre-processors are also sent to an FPGA based segment finder[1]

[1] A prototype based on an Altera FPGA on a PCI card has been built and tested [1].

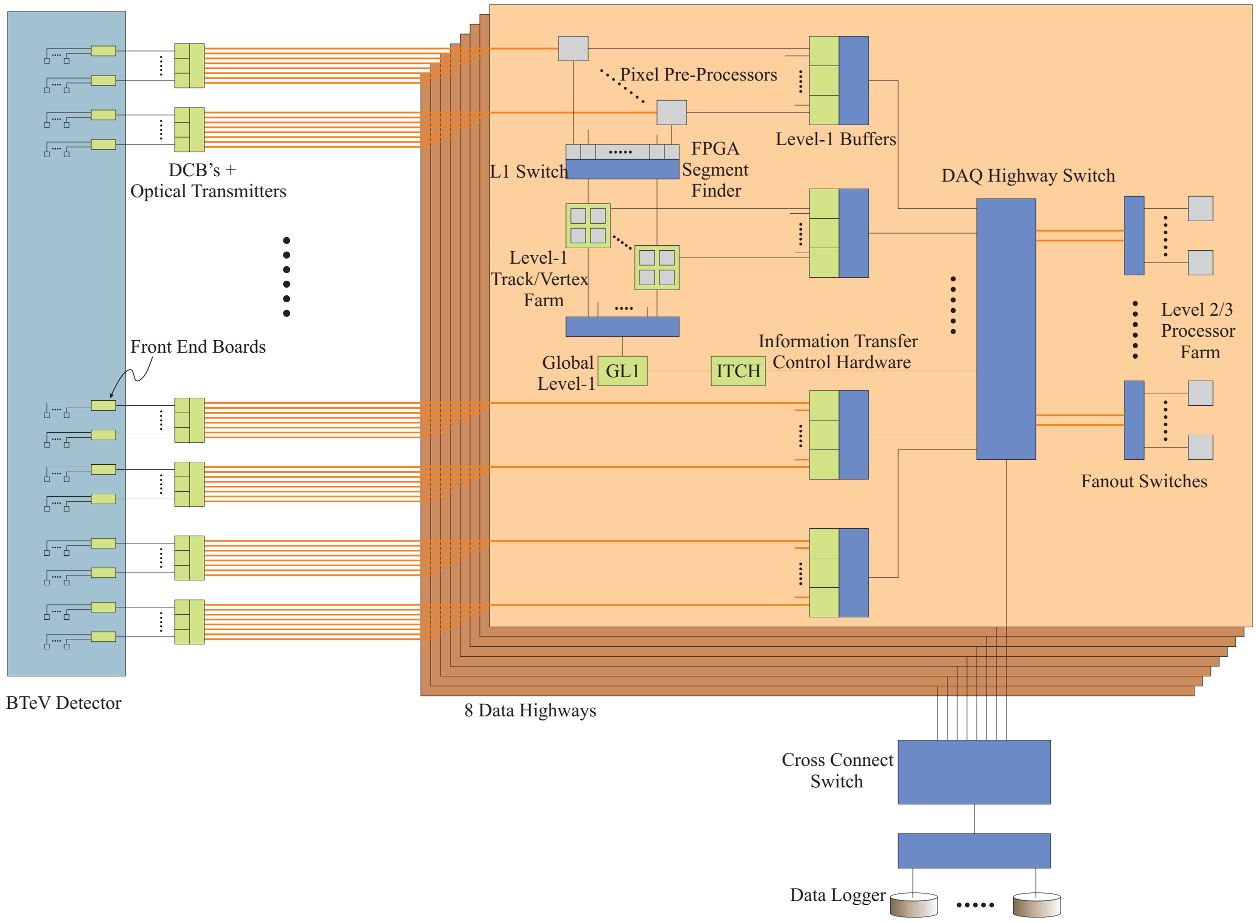

Fig. 3. BTeV's three-level eightfold-way trigger architecture.

that executes the segment finding stage of the algorithm. Inner and outer triplets belonging to the same crossing are then routed by an event building switch (L1 switch) from the segment finder to one node in the vertex farm. For each node, complete processed results are routed to the L1 buffers while summarized trigger results are sent to a Global Level-1 (GL1) processor responsible for the ultimate trigger decision. Crossings accepted by GL1 are then maintained as a list in the Information Transfer Control Hardware (ITCH) which also broadcasts accept messages to all L1 buffers indicating which crossings are to be saved.

A L1 buffer is built out of a module consisting of commodity DRAM configured as circular buffers paired with a commodity PC server motherboard acting as a controller [1]. Prototypes exist and are currently being used with an ionization profile monitor for Fermilab's Tevatron [9]. There are 24 L1 buffers in a highway each having 24 input buffers serving 2 DCB subsystems in the collision hall. Upon receiving a Level-1 accept, data from all 24 input buffers are concatenated and copied to an output buffer on the L1 buffer controller where it remains until transferred to a Level-2/3 (L2/3) node. The use of low-cost DRAM allows enough memory to buffer over 100 thousand crossings in each highway corresponding to $\sim$500 ms of L1 trigger decision time which is over three orders of magnitude more than the average L1 processing time. The use of large buffers with circular access is far more cost effective than smaller ones employing sophisticated memory management. It also allows the system to handle the long processing times of events in the tail of the L1 time distribution. The 24 L1 buffer subsystems feed ports in a DAQ highway switch consisting of a commercial 72-port gigabit ethernet switch. Output ports on this switch feed 32 8-port gigabit ethernet fanout switches. These fanouts, in turn, feed data to 96 dual CPU commodity Linux-PC's that make up the L2/3 processor farm[2] in each highway where the DRAM in each of the nodes functions as a Level-2/3 buffer.

After receiving a request from an idle node in the L2/3 processor farm for an event, the ITCH responds by assigning an accepted crossing number to that node. Once it receives its assignment, the L2/3 node sends a request to a subset of the L1 buffers which respond by sending their data to that node. All requests and data transfers between the L2/3 farm and the L1 buffers and the ITCH are routed through the highway and fanout switches. Upon receiving the data, the L2/3 node executes the L2 trigger algorithm which now has the option of using additional information from the first few stations of the straw and Si-microstrip trackers in doing a more refined analysis of the event. If the event passes L2, data from the rest of the L1 buffers is transferred to the same node to execute L3.

[2]A prototype L2/3 farm based on retired Fermilab farm nodes has been set up and is being used to conduct tests.

At this stage, the processing node has, at its disposal, particle ID information in addition to that from the rest of the forward tracking stations to further improve upon the L2 results.

In practice, each L2/3 node issues multiple event requests, storing data for $\sim$16-32 events in the L2/3 buffers at any given instant. When a Level-2/3 processor completes executing the L2 trigger algorithm on an event, it performs a context switch to process one of the other events in the buffer while new data is requested to fill the buffers. Processing of events that pass L2 is temporarily suspended while the L2/3 processor switches to another event, resuming L3 processing on the L2-accepted event after additional data has been transferred. This way, no dead-time is incurred between events due to data transfers between the L1 and L2/3 buffers.

If the event passes L3, the processed results are propagated back up the fanout and DAQ highway switches to an external cross-connect switch that routes accepted events from all 8 highways to a small cluster of data-logging nodes for archival.

V. Data Rates

The data rates in this section are determined assuming certain data formats and network protocols. The format of the data sent from the front-ends to the DCB's are detector dependent and are described in detail in [1]. Communication between the DCB's and L1 is based on a custom low-overhead protocol with 8B/10B encoding on optical fibers. Data traffic within L1 is also based on a custom low-overhead protocol via differential copper links. For L2/L3, standards like TCP/IP and ethernet are used.

Each of the 48 DCB subsystems serving the detector front-end uses 8×12 2.5 Gb/s optical links to bring the data from the collision hall to the L1 buffers in the counting room [1]. This makes it possible to achieve a peak design rate of over 1 TB/s providing sufficient headroom for the estimated data rate of 500 GB/s (200KB/event) from the full detector. The data rate going into the L1 buffers in each highway is reduced $8\times$ to 62.5 GB/s through the use of the parallel highway architecture described above. Assuming an occupancy of 0.1 hit/interaction for each of the 8100 FPIX2 pixel readout chips [10], a mean value of 9 interactions per crossing, 3 bytes/hit, and including a safety factor of 2, the data coming out of the pixel detector front-ends at a crossing rate of 2.5 MHz is on the order of 110 GB/s. This is increased to about 220 GB/s after additional information, such as the ID's of the pixel read-out chips, is inserted into the data stream by the pixel DCB's. This means the data rate going into the pixel pre-processors in each highway is 27 GB/s. If the average size for the total number of triplets found by the L1 segment finder for each crossing is $\sim$8 KB, the data rate going into the L1 track/vertex processor farm in each highway is 2.5 GB/s. Assuming 50 bytes of summarized trigger results per crossing, the total data rate going from the track/vertex farm to the GL1 processor in each highway is 16 MB/s.

The L1 trigger rejects 98% of the input to the L1 buffers reducing the output from the buffers by $50\times$ to 1.56 GB/s per highway for an average event size of 250KB at this stage(for simplification, we will treat L2 and L3 as a single trigger stage in this discussion). This means the data rate coming out of each of the 24 L1 buffer subsystems is 65 MB/s and that going into the fanout switches is 49 MB/s, both of which can be handled by the gigabit ethernet ports on the DAQ highway switch. Data is distributed to each node in the L2/3 processor farm at 16 MB/s using the gigabit ethernet ports on the fanout switches. The L2/3 trigger rejects 95% of the incoming data reducing this by $20\times$ to 78 MB/s. A $3.125\times$ data compression further reduces this to 25 MB/s. The resulting data rate from all 8 highways into the cross-connect switch and the data-logging cluster is a mere 200MB/s which can easily be handled by commercially available storage technology.

VI. Baseline Changes to the Trigger Architecture

A. Commodity Based Level 1 Trigger

The original baseline design of BTeV's L1 trigger consisted of a custom designed event building switch and a vertex farm of several thousand 150MHz TI TMS320C6711 floating point Digital Signal Processors (DSP). A pre-prototype version of the vertex farm hardware based on these DSP's has been developed and tested [11]. Extensive benchmarks of the L1 trigger code were ran on these DSP's and a host of other more general purpose processors ranging from high-performance System-On-a-Chip (SOC) designs to the ubiquitous x86 based architectures found on desktops [1], [12]. These tests clearly demonstrated the superiority of the general purpose designs over the DSP's–in many cases executing the L1 trigger code at least an order of magnitude faster. Based on these results and additional tests conducted on commercially available, high-bandwidth networking solutions with extremely low latencies [1], [13], the original baseline was replaced with one consisting of a vertex farm of 264 dual CPU (IBM970) Apple Xserve's and Infiniband-based event building switches. In addition to performance issues, this decision was also made in order to reduce labor and costs and to minimize scheduling risks. A prototype version of this commodity based vertex farm consisting of 16 Apple Xserve's and a 24-port Infiniband switch has been assembled and undergoing tests.

Although the move from custom switches and DSP-based vertex farms to a commodity based solution represented a significant change to the L1 trigger components, it did not represent a fundamental change to the baseline architecture of the L1 trigger. In the next two sections, we briefly describe a proposed second baseline change BTeV was in the process of making before its cancellation earlier this year that represented a fundamental change to the L1 architecture [14], [15].

B. Integrated Upstream Event Builder

This second baseline change is depicted by the two diagrams shown in Fig. 4. The diagram on the left shows the original baseline architecture where the segment finders deal with data fragments from a small portion of the full pixel detector requiring a downstream event building switch to route all fragments to the same processor in the vertex farm. The new baseline architecture is depicted on the right where the event building switch has been moved upstream of the segment finders and

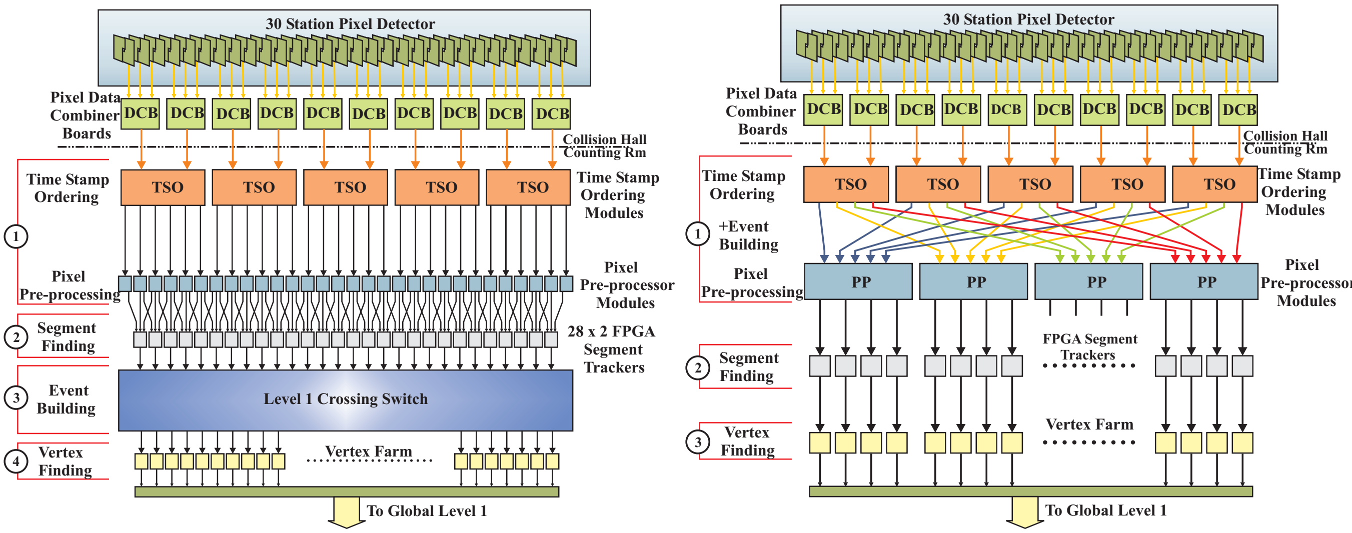

Fig. 4. The original baseline Level 1 trigger architecture shown next to the new integrated upstream event builder architecture.

integrated with the pixel pre-processing stages. This change is technically feasible and practical because the time stamp ordering function performed in the pixel pre-processing stage is fundamentally a switching operation making it natural to incorporate event building functions into this stage.

Moving to an upstream event builder introduces significant advantages. First of all, since the segment finders now receive data from the full pixel detector, their number is no longer tied to the geometry of the detector making it much easier to scale this number in response to changes in running conditions. An upstream switch also makes it easier to route around failures in the segment finding hardware increasing the fault tolerance of the system. Furthermore, the fact that the segment finders now see the entire pixel detector also makes it possible to offload CPU-intensive operations like the track finding done in the vertex farm processors to the FPGA hardware. This significantly reduces processing time making it possible to move portions of the higher level trigger algorithms into Level 1. The single large event building switch downstream of the segment finders can now be replaced with a number of smaller and simpler distributed switches for the purpose of load balancing and maintaining fault tolerance in the vertex farm. Needless to say, the larger packet sizes of fully built events allows more efficient use of the network bandwidth at this stage of the L1 trigger.

C. Commodity Blade Server Platform

Aside from the advantages enumerated above, the move to an upstream event building architecture also makes it possible to package the L1 trigger hardware in creative new ways. One such solution being adopted by BTeV is based on the Intel/IBM blade server platform whose specifications have been made open to the public [16]. It consists of a 19” 7U chassis that holds up to 14 commodity high-performance dual CPU x86 or PowerPC blade servers mating with a modular high-speed serial midplane that provides redundant connections for each blade to 4 modular switches in the rear of the chassis. A managment module serves the role of a crate controller and provides remote manangement capability for each blade. Reliable operation is further enhanced by redundant modular power supplies and blowers.

The design of this platform was flexible enough that the plan was to house a combination of custom designed segment finder boards with off-the-shelf CPU blades serving as vertex farm nodes in a single chassis. A block diagram of the configuration for one highway is shown in Figure 5 in which the main box at the top represents the pixel pre-processors and the upstream event builder. Fully built events from this stage are sent to the L1 segment finder, which together with the vertex farm hardware, is housed in four blade server chassis. Each chassis consists of 4 custom blades (ST), with 4 FPGA segment finders each, feeding 8 dual CPU processor blades (TV) serving as vertex farm nodes through simple custom designed switches plugging into modular switch bays in the rear of the chassis. The on-board memory on each CPU blade also serves as the L1 buffer to hold the processed output from this stage. Gigabit ethernet connections to the DAQ highway switch provide a data path from these buffers to a L2/3 node upon a L1 accept. Additional ethernet connections are used for monitoring and control.

With this high-density solution, it is possible to house all of the L1 trigger hardware for one highway in a single 42U rack reducing the L1 rack count by a factor of 2-3. The use of a commodity based solution such as this is also far more cost effective than solutions based on VME or ATCA standards used in the telecom industry. Furthermore, it assures the wide availability of processor blades with the highest performance CPU’s on the market which is often not the case with VME or ATCA. This approach allows one to mix and match relatively low cost but very high performance off-the-shelf CPU blades with custom hardware based boards in a flexible package that can be tailored to a wide variety of trigger and DAQ problems.

VII. Conclusion

We have designed a trigger system for the demanding environment of a hadron collider B-physics experiment using

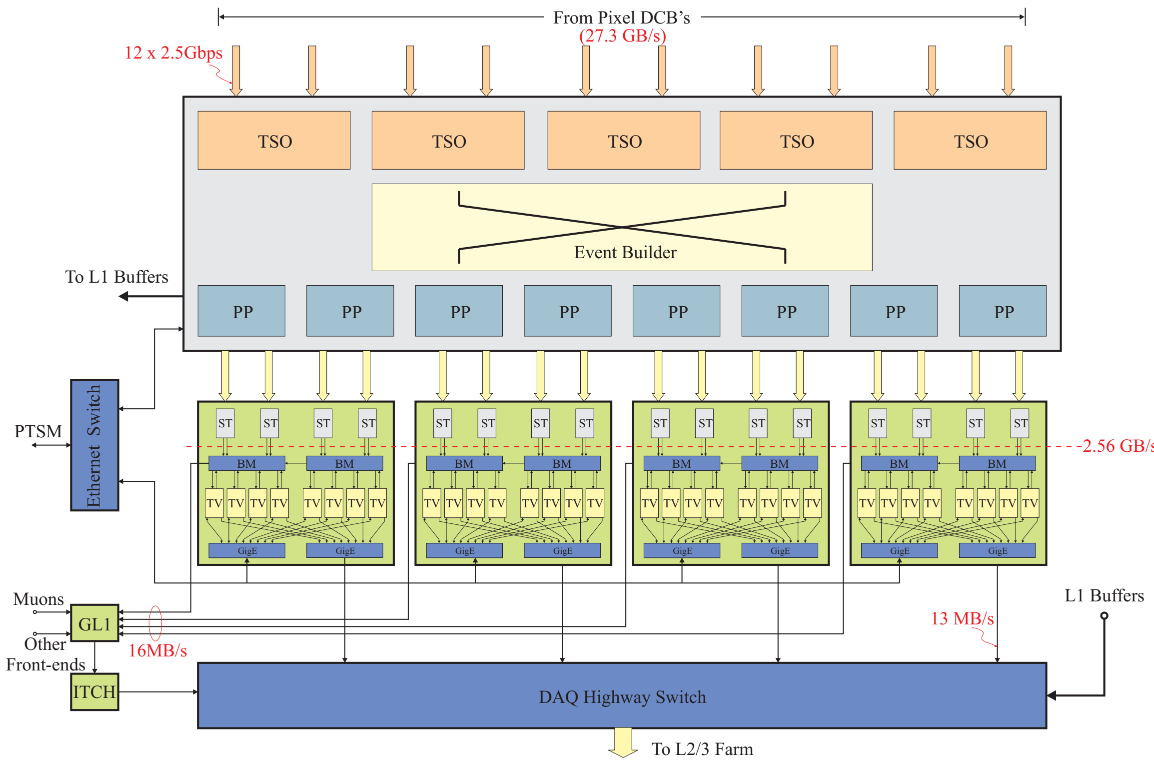

Fig. 5. L1 trigger with an upstream event builder architecture using commodity blade server components.

commodity PC, memory, and networking components for a substantial portion of the system. This trigger is unique in the amount and quality of processing applied at the lowest level which is, to a large extent, possible due to the asynchronous readout system based on large memory buffers built from commodity DRAM. Aside from improving performance and ease of programming, moving from a DSP-based to a COTS L1 track/vertex farm resulted in substantial savings in cost and labor and in reducing scheduling risks. Replacing the baseline architecture with that of an integrated upstream event builder greatly enhances the scalability and fault tolerance of the system. It will allow us to move even more of the processing done by the higher level triggers into Level 1. And it allows packaging of the L1 trigger hardware using a flexible, high-density commodity blade server platform. We believe the ideas presented here will be useful to other HEP experiments faced with similar challenges.

References

[1] G. Y. Drobychev *et al.*, "The BTeV detector technical design report," Fermilab, Batavia, IL, BTeV-doc-2115, Dec. 2004. [Online]. Available: https://www-btev.fnal.gov/cgi-bin/DocDB/ShowDocument?docid=2115

[2] (2005) More information about btev's cancellation is available on its website. [Online]. Available: http://www-btev.fnal.gov

[3] A. Kulyavtsev *et al.*, "BTeV proposal," Fermilab, Batavia, IL, BTeV-doc-66, May 2000. [Online]. Available: https://www-btev.fnal.gov/cgi-bin/DocDB/ShowDocument?docid=66

[4] G. Y. Drobychev *et al.*, "Update to BTeV proposal," Fermilab, Batavia, IL, BTeV-doc-316, Mar. 2002. [Online]. Available: https://www-btev.fnal.gov/cgi-bin/DocDB/ShowDocument?docid=316

[5] J. N. Butler, "The BTeV trigger and data acquisition system," presented at the 10th Workshop on Electronics for LHC and Future Experiments, Boston MA, USA, Sept. 13–17, 2004.

[6] M. Votava on behalf of the BTeV DAQ and Trigger Groups, "BTeV trigger/DAQ innovations," presented at the 14th IEEE-NPSS Real Time Conference 2005, Stockholm, Sweden, June 4–10, 2005.

[7] M. Wang, "BTeV Level 1 vertex trigger algorithm," Fermilab, Batavia, IL, BTeV-doc-1179, 2002. [Online]. Available: https://www-btev.fnal.gov/cgi-bin/DocDB/ShowDocument?docid=1179

[8] E. E. Gottschalk, "BTeV detached vertex trigger," *Nucl. Instrum. Meth.*, vol. A473, pp. 167–173, 2001.

[9] A. Jansson *et al.*, "An ionization profile monitor for the Tevatron," Fermilab, Batavia, IL, FERMILAB-CONF-05-170-AD-CD-E, May 2005. [Online]. Available: http://lss.fnal.gov/archive/test-preprint/fermilab-conf-05-170-ad-cd-e.shtml

[10] D. C. Christian *et al.*, "FPIX2, the BTeV pixel readout chip," *Nucl. Instrum. Meth.*, vol. A549, pp. 165–170, 2005.

[11] G. Cancelo *et al.*, "The DSP-based Level 1 trigger track and vertex pre-prototype hardware," Fermilab, Batavia, IL, BTeV-doc-3360, 2004. [Online]. Available: https://www-btev.fnal.gov/cgi-bin/DocDB/ShowDocument?docid=3360

[12] M. Wang and J. Y. Wu, "Level 1 trigger DSP timing studies and the hardware hash sorter," Fermilab, Batavia, IL, BTeV-doc-3361, 2004. [Online]. Available: https://www-btev.fnal.gov/cgi-bin/DocDB/ShowDocument?docid=3361

[13] J. Kowalkowski *et al.*, "COTS-based Level 1 trigger architecture proposal," Fermilab, Batavia, IL, BTeV-doc-3262, 2005. [Online]. Available: https://www-btev.fnal.gov/cgi-bin/DocDB/ShowDocument?docid=3262

[14] J. Y. Wu *et al.*, "Integrated upstream parasitic event building architecture for the BTeV Level 1 pixel trigger system," presented at the 14th IEEE-NPSS Real Time Conference 2005, Stockholm, Sweden, June 4–10, 2005.

[15] M. Wang and J. Y. Wu, "Proposal for a new Level 1 trigger architecture for BTeV," Fermilab, Batavia, IL, BTeV-doc-3342, 2004. [Online]. Available: https://www-btev.fnal.gov/cgi-bin/DocDB/ShowDocument?docid=3342

[16] More information regarding the Intel/IBM BladeCenter Platform Design Specifications are availale at the Intel and IBM websites. [Online]. Available: http://developer.intel.com/design/servers/blades/designspecs.htm, http://www-1.ibm.com/servers/eserver/bladecenter/open_specs.html