An event builder network for LHCb upgrade and the simulations on its performance

Guoming Liu, Niko Neufeld
CERN, CH-1211 Geneva 23, Switzerland
E-mail: guoming.liu@cern.ch

Abstract.
The LHCb experiment is considering an upgrade towards a trigger-free Data Acquisition (DAQ) system. All detector information will be readout at full collision rate (40 MHz) and sent to a large processing farm which performs event building and filtering. According to the physics simulation result, the DAQ system requires a large event builder network (also called DAQ network) with an aggregate bandwidth more than 34 Tb/s.

In this paper we present a new architecture of the event builder network, which is a multi-stage network using low-cost 10-Gigabit Ethernet switches. To evaluate the performance of the full scale system, several simulations have been done. This paper describes different DAQ protocols and the simulation developed for the event builder network based on the framework OMNeT++. The buffer occupancy of the switches has been studied.

1. Introduction to the LHCb upgrade

The LHCb experiment is dedicated to the study of CP violation and other rare phenomena in the B-meson decays with very high precision. LHCb is designed to operate at a luminosity $\sim 2 \times 10^{32} cm^{-2} \cdot s^{-1}$. It has already started taking physics data from 2009 and is expected to accumulate a data sample of 6 to 10 fb^{-1} during the following five years. At that time, the statistical precision on measurements increases very slowly if the LHCb experiment continues operating at current peak luminosity without an upgrade. LHCb has planned an upgrade in 2016 [1] to enable operation at 10 times of the current design luminosity, i.e. at about $2 \times 10^{33} cm^{-2} \cdot s^{-1}$, to collect an integrated luminosity of about 100 fb^{-1}.

One of the main aims of the LHCb upgrade is to improve the trigger efficiency for hadronic decays. The current LHCb trigger comprises a first level trigger (L0) implemented in hardware and a Higher Level Trigger (HLT) running on a large CPU farm. The maximum L0 trigger rate is ∼1 MHz, which is limited by the hardware. The LHCb upgrade has opted for a new readout architecture which requires all data readout at the full collision rate of LHC (40 MHz) and sent to a large CPU farm. The L0 trigger will be removed and the event selection becomes entirely based on software running in the large CPU farm.

The new architecture of the upgrade readout system will be discussed in Section 2, followed by the description of a readout network in Section 3. The readout protocols will be discussed and studied by simulation in Section 4. The conclusions will be presented in Section 5.

2. Architecture of the upgrade DAQ system

The architectures of the current and the upgrade DAQ system are shown in Fig. 1 [1, 2].

When increasing the event rate, the limit on the readout rate leads to a considerable drop of efficiency in particular for the hadron trigger in current system. To read out data at a higher rate, the L0 trigger will be much more complicated. And since the general purpose CPU has become more and more powerful, LHCb has decided to increase the readout rate to 40 MHz and transmit all data to the high-performance CPU farm. The trigger algorithm will be implemented in software running in the large CPU farm, which allows more flexibility and efficiency.

In the Front-End Electronics (FEE), the L0 buffer for 4 μs latency will be removed. The front-end electronics will digitize the signals from the detectors, then zero-suppress the data in order to reduce the number of links to the back-end electronics called TELL40. The front-end electronics and TELL40 are connected through high-speed optical links Gigabit Transceiver (GBT). TELL40 are common to all sub-detectors. TELL40 will receive the fragments arriving on the different data links from the detector and reassemble them, then send the data to the HLT farm through the event builder network which will be described in the next section. All the functions will be implemented in field-programmable-gate-arrays (FPGAs).

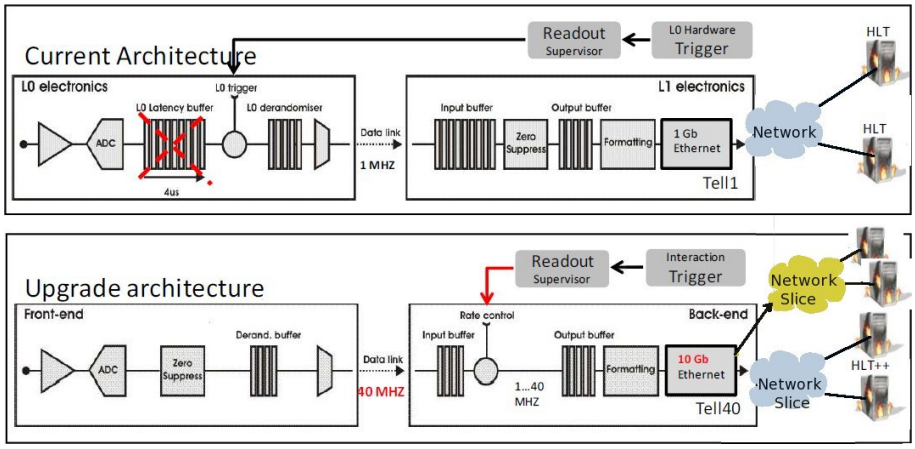


Figure 1: Current and upgrade architecture of DAQ

A throttling mechanics will be implemented in the interaction trigger, which will control the data flow in case of saturation of the network or the CPU farm. The interaction trigger rejects uninteresting events, this will bring down the requirements to the event builder network and the CPU farm.

3. Event builder networks for the LHCb upgrade

According to the simulation, the event size of the upgrade LHCb will be about 100 kB. The effective event rate will be about 30 MHz in LHCb, because 10 MHz out of 40 MHz are empty. The data rate pass through the event builder network will be about 3 TB/second. Assuming the link load is 70%, the event builder network will have to provide an unidirectional bandwidth of $\sim$ 34 Tb/s.

10 Gigabit Ethernet (10GbE) and InfiniBand are the two promising technologies for high speed interconnection today and for the near future. 10GbE is more popular and familiar by users, and it is easy to build a system using Ethernet technology. InfiniBand has a higher bandwidth and lower latency over 10GbE, however our application is not latency sensitive. The data rate each CPU can process will be less than 10 Gb/s according to Moore's law at the time of the deployment, so a 10 Gb/s link should be enough to saturate the CPU, even if it is not enough, more links can be added to form a aggregation link to increase the bandwidth. 10GbE is the baseline solution for the LHCb upgrade, we will discuss 10GbE in this paper.

According to Moore's law, we assume about 5000 servers are needed for the event filter and about 4800 10GbE links are needed to connect TELL40s [3], so the event builder network need to provide about 9800 10GbE ports. This is a large network. Since all the CPUs are identical and run their jobs independently, the event filter farm nodes do not communicate with each other, so the network can be split into several independent small networks (slices). Each slice has its independent CPU farm attached, while TELL40 has multiple output links and is connected to each independent slice through one or a group of links, so each TELL40 can reach any farm node. This solution reduces the network complexity and allows to stage the deployment of the network and the CPU farm. Several readout-systems will be operated in parallel providing high redundancy, which has been demonstrated by the current CMS DAQ system. The architecture of the network with eight slices is shown in Fig. 2

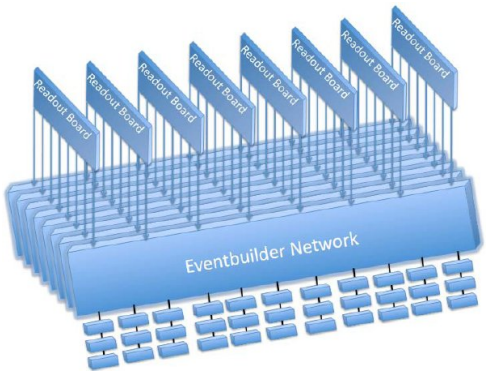


Figure 2: Upgrade readout-network with eight slices

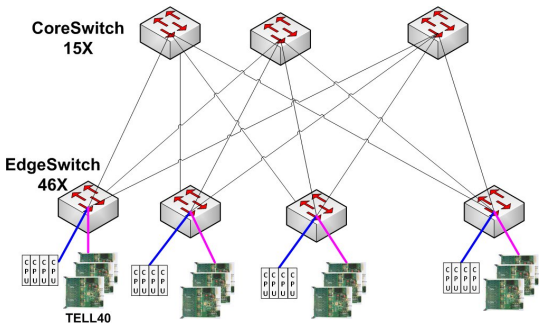


Figure 3: Network architecture based on TOR switches

A single slice can be implemented either using a large monolithic core-router as the current LHCb DAQ system today or using an aggregate of smaller and cheaper Top-Of-Rack (TOR) switches. The choice between these two solutions is subject to the development of both kinds of devices from industries. For the solution based on a large core-router is rather simple, we will wait for the latest products in the market at the time of deployment, evaluate and compare the devices from different suppliers. The solution based on TOR switches will need more R&D efforts.

In this paper we focus on the solution based on TOR switches. Fig. 3 shows the architecture of one slice. Assuming the network will be split into 8 slices, each slice will need to provide more than 1200 10GbE ports. The slice consists of two layers, the core layer and the aggregation layer. The aggregation layer consists of 46 edge switches, while the core layer consists of 15 core switches. We will use the same 48-port switches for both layers. Since the traffic is unidirectional from TELL40s to the HLT farm, we mix the TELL40s and HLT farm nodes into the edge switches in order to make use of the duplex switching capacity in the switches. Each edge switch has 15 ports for the TELL40s, 18 ports for the HLT farm nodes and 15 links for the connection to the core layer. The number of ports for TELL40s and HLT farm nodes can be adjustable. The core layer provides interconnection between the edge switches, this is a non-blocking network.

4. Data readout schemes and the simulation studies

In the current DAQ system, LHCb uses a UDP-like protocol called Multi-Event Packet (MEP) for the transport layer. In this protocol, several events are packed into one packet, the number of events in one packet is called packing factor. In the upgrade system, we will keep the transport

protocol MEP. There are several schemes for the data readout with different traffic shaping methods. we will discuss the push and pull schemes in the following sections.

4.1. Push scheme

The push scheme is a simple solution for data readout, and is used by the LHCb DAQ system today. In this scheme, the available farm nodes declare their availability to the readout supervisor, then the readout supervisor assigns the farm nodes for the specific collision events based on a round-robin scheme and communicate the result to the readout boards. The readout boards send packets to the assigned farm node as soon as they receive the data fragments from the front-end electronics. Because of the nature of the particle physics experiment, all data fragments belong to the same event should go to the same server for event filtering, they arrive at the event builder network almost at the same time. The data from multiple input links contest one output link in the switches. This requires a deep buffer and a powerful switching capacity in the switches, it will be a challenge in a high data rate and big event size. On the other hand, the readout boards only need very limited buffer. The schematic of the push DAQ scheme is shown in Fig. 4.

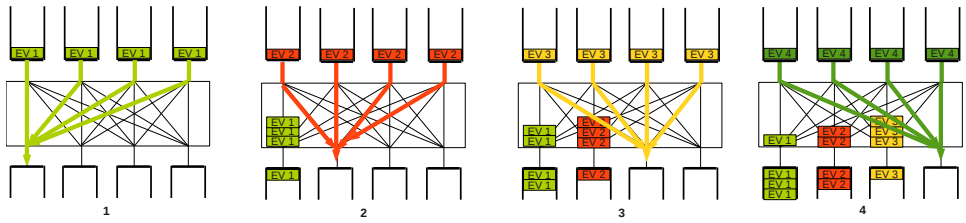


Figure 4: Schematic of the push DAQ scheme

4.2. Pull scheme

The pull scheme is another traffic-shaping method which is similar to the barrel-shift scheme [4]. As schematically shown in Fig. 5, the readout supervisor assigns the events to the farm node when the events are ready in the readout boards. During the first time slot, node 1 sends a data request to source 1 for event 1, then source 1 responds to node 1 with event 1. During time slot 2, node 1 sends a data request to source 2 for event 1, while node 2 sends a data request to source 1 for event 2, then the sources respond to the corresponding nodes with events. After N time slots (where N is the number of sources) all sources are sending data to mutually exclusive destinations. The pull scheme shapes the traffic in the network and makes it less bursty and reduces the requirements to the network devices, however it requires significant buffering on the readout boards.

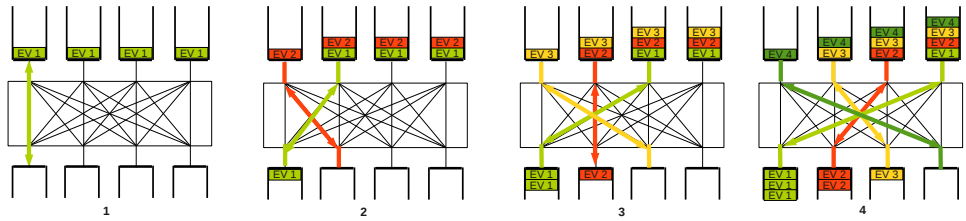
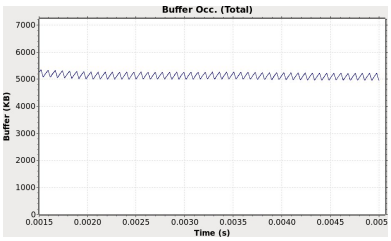


Figure 5: Schematic of the pull DAQ scheme

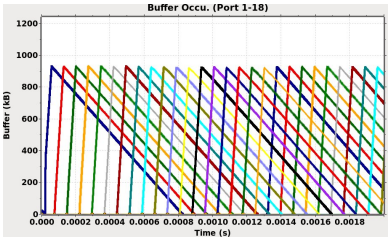
4.3. Simulations on different schemes

In the network devices, buffers are required to temporarily store packets in queue. Buffer overflow is the main cause of packet drop in the network. We have done some simulations on the network proposed in Fig. 3 for different schemes. The simulation is based on the framework OMNET++ which is an extensible, modular, component-based C++ simulation library and framework (see [5] for more detail). In the simulation, we assume the switches use a shared-memory output queuing strategy [6], the traffic is buffered in output ports. The event size of 100 kB and packing factor of 10 are used for the simulation.

The simulation result for the buffer occupancy in the push scheme is shown in Fig. 6 (edge switches) and Fig. 7 (core switches). The buffer occupancy in the edge switches is quite high, 970 kB for a single port and 5.4 MB for the switch in total, while the buffer occupancy in the core switches is quite moderate, only 48 kB for a single port and 648 kB for the whole switch.

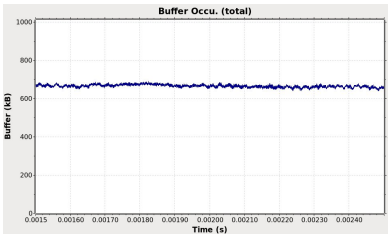


(a) Total

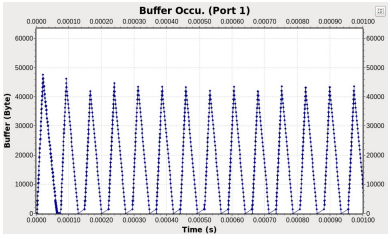


(b) Individual ports

Figure 6: Buffer occupancy of edge switches in push scheme



(a) Total



(b) Single port

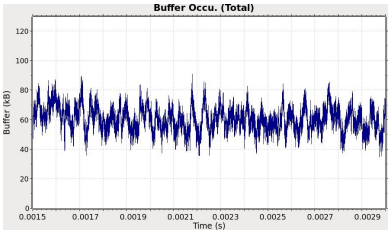
Figure 7: Buffer occupancy of core switches in push scheme

The simulation result for the buffer occupancy in the pull scheme is shown in Fig. 8 (edge switches) and Fig. 9 (core switches). This scheme needs very limited buffer in the switches, only 15 kB for a single port in both edge and core switches. However, it needs a deep buffer in the readout boards, about 483 kB for each port.

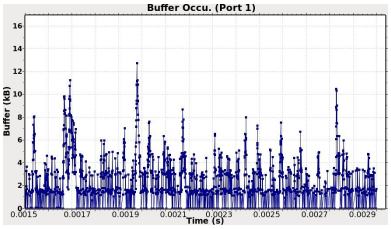
Instead of buffering data in either the network devices or the readout boards, we can also combine the pull and push methods. In the pull-push combined scheme, the readout boards are divided into N groups. In each time slot, the farm node pull data from one group of readout boards. The simulation result for the buffer occupancy in the pull-push scheme with 2 groups is shown in Fig. 10 (edge switches) and Fig. 11 (core switches). The buffer occupancy of a single port in the edge switches is about 480 kB which is half of the push mode, the total buffer occupancy is about 3.6 MB. The buffer occupancy in the core switches is still quite moderate. In the readout boards, each port need a buffer of 240 kB.

5. Conclusions

In this paper, we have introduced the LHCb upgrade and the new architecture for DAQ. The upgrade DAQ system requires a large event builder network, which will be split into several independent slices. We propose a solution based on 10GbE TOR switches for the slice. We have studied different data readout schemes (push, pull and pull-push) and done some simulations

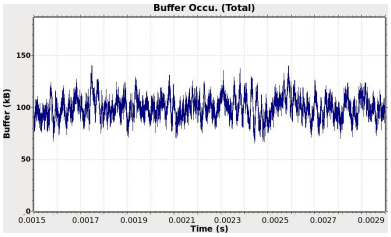


(a) Total

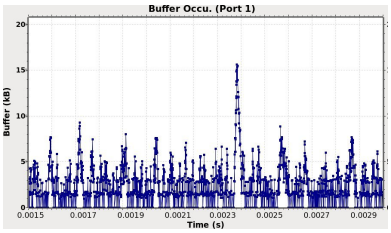


(b) Single port

Figure 8: Buffer occupancy of edge switches in pull scheme

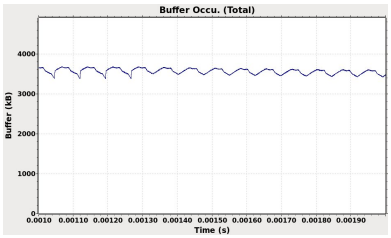


(a) Total

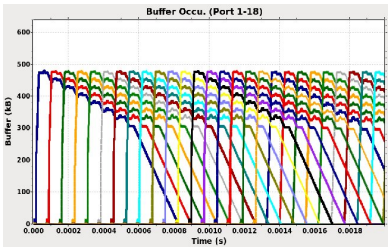


(b) Single port

Figure 9: Buffer occupancy of core switches in pull scheme

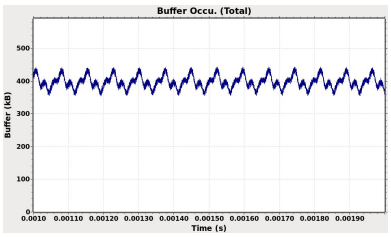


(a) Total

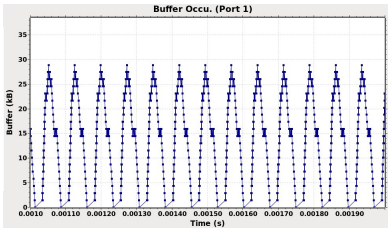


(b) Individual ports

Figure 10: Buffer occupancy of edge switches in pull-push scheme



(a) Total



(b) Single port

Figure 11: Buffer occupancy of core switches in pull-push scheme

using OMNET++. The simulation results show that the push scheme require a deep buffer in the edge switch, the pull scheme needs a deep buffer in the readout boards, while the requirement to the network switches and the readout boards are compromised in the pull-push combined mode.

References

[1] LHCb collaboration 2010 Letter of Intent for the LHCb Upgrade URL `https://twiki.cern.ch/twiki/pub/LHCb/LHCbUpgrade/LOI.pdf`

[2] Buytaert J 2010 *JINST* **5** C12035 URL `http://stacks.iop.org/1748-0221/5/i=12/a=C12035`

[3] Cachemiche J P, Duval P Y, Hachon F, Gac R L and Marin F 2010 *JINST* **5** C12036 URL `http://stacks.iop.org/1748-0221/5/i=12/a=C12036`

[4] Barsotti E, Booth A and Bowden M 1990 *AIP Conference Proceedings* **209** 82–101 URL `http://link.aip.org/link/?APC/209/82/1`

[5] OMNET++ `http://www.omnetpp.org`

[6] CHAO J and LIU B 2007 *High performance switches and routers*