DAQ Architecture for the LHCb Upgrade

Guoming Liu[1] **and Niko Neufeld**
CERN, Geneva, Switzerland

E-mail: `niko.neufeld@cern.ch`

Abstract.
LHCb will have an upgrade of its detector in 2018. After the upgrade, the LHCb experiment will run at a high luminosity of 2×10^{33} $\mathrm{cm}^{-2}\mathrm{s}^{-1}$. The upgraded detector will be read out at 40 MHz with a highly flexible software-based triggering strategy. The Data Acquisition (DAQ) system of LHCb reads out the data fragments from the Front-End Electronics and transports them to the High-Lever Trigger farm at an aggregate throughput of ~ 32 Tbit/s. The DAQ system will be based on high speed network technologies such as InfiniBand and/or 10/40/100 Gigabit Ethernet. Independent of the network technology, there are different possible architectures for the DAQ system.

In this paper, we present our studies on the DAQ architecture, where we analyze size, complexity and relative cost. We evaluate and compare several data-flow schemes for a network-based DAQ: push, pull and push with barrel-shifter traffic shaping. We also discuss the requirements and overall implications of the data-flow schemes on the DAQ system.

1. Introduction

The LHCb upgrade during the Long Shutdown 2 (LS2) of the Large Hadron Collider (LHC) has two principle goals: to improve detectors and electronics such that the experiment can run at an instantaneous luminosity of 2×10^{33} $\mathrm{cm}^{-2}\mathrm{s}^{-1}$ and to read out every detector element for every bunch-crossing, thus creating essentially a trigger-less experiment [1]. The first of these goals is expected to increase the event-size from between 50 and 60 kB during the LHC runs 1 and 2 to 100 kB. The second increases the event-rate from 1 MHz to 40 MHz[2]. Multiplying the two figures yields the first key parameter of the future data acquisition (DAQ), the aggregated bandwidth of 32 Tbit/s. The input data will be distributed in about 500 data-sources, connected to the detector front-end electronics. The DAQ must ensure that data from all these 500 sources can reach any of up to 5000 filter units in a future high-level-trigger (HLT)[3] farm. The key parameters are compared with the situation in Runs 1 and 2 in Table 1. The system should be scalable, but it is understood that depending on the technology there can be significant non-linearities in the cost, e.g. if a new router needs to be added to the system. The system should degrade gracefully under hardware failures, however redundancy is not required. Data losses should be as small as achievable at reasonable cost and they must be random in nature, i. e. they must not depend in any way on the size or content of the data, to avoid the introduction of any bias into the event-sample seen by the HLT.

[1] Now with ITER, France

[2] Strictly speaking only 30 out of the 40 MHz events will have collision data, because of the empty bunch-positions in the LHC.

[3] The high-level trigger is actually an *event-filter*, not a trigger.

Table 1. Key paramaters of the LHCb DAQ

	Runs 1 & 2	Run 3
event-size [kB]	50 - 60	100
event-rate [MHz]	1	40
# data sources	313	500
# data sinks	up to 2000	up to 5000

Such a system can only be implemented at reasonable cost using a commercial local area network technology, such as Ethernet. In the following we describe various ways of building a network, which fulfills the requirements and we will discuss their relative merits in particular also in view of the prospective cost. Other important aspects such as run-control, monitoring and data-storage will not be discussed here, they are expected to at least conceptually resemble very closely the respective systems of Runs 1 and 2 [2], [3], [4].

2. DAQ architectures

In the following discussion of possible DAQ architectures we distinguish three functions

(i) The Readout Unit (RU) acquires data from the detector and sends them to an assigned builder-unit

(ii) The Builder Unit (BU) collects data from all Reaout Units which belong to the same collection of bunch-crossings and assembles them into a complete event. It sends complete events to the Filter Unit.

(iii) The Filter Unit (FU) receives complete events and selects events for permanent storage

RUs and BUs together with the network which connects them are called the event-builder.

It is important to understand that the same physical device can implement one or more functions[4].

We will now discuss three ways to distribute the functions within a local area network. It should be noted that all solutions permit several ways of actually distributing the data. Data can be pushed from the RUs to the BUs, or pulled by the BUs from the RUs. Mixtures of these two basic network traffic organizations are also possible. We are not going to discuss these so-called event-building protocols here, because all presented architectures can work with a wide range of them.

In the network we distinguish between the core network and the dis-aggregation layer. The core connects the RUs, which require high-speed links and high-end networking equipment. The RUs will most of the time send data at nominal speed.

The FUs are normally connected to the dis-aggregation (distribution) layer, because the amount of data they receive is limited by their processing power and hence they can typically connect with a much lower-speed link. The dis-aggregation switches are normally rather cost-effective compact “top of the rack” (TOR) devices. They are connected to the core-network via a small number of high-speed up-links.

We attempt to achieve overall cost efficiency by minimising the number of core ports in the system.

2.1. *Unidirectional data-flow*

In the unidirectional data-flow the RUs are distinct from the FUs and BUs. The same computer implements both the BU and FU, i.e. the BU is simply a software process at the “entry” of the

[4] In the most extreme form of such systems, every device implements all functions including the network itself (e.g. torus-like interconnects such as used by the BlueGene architecture)

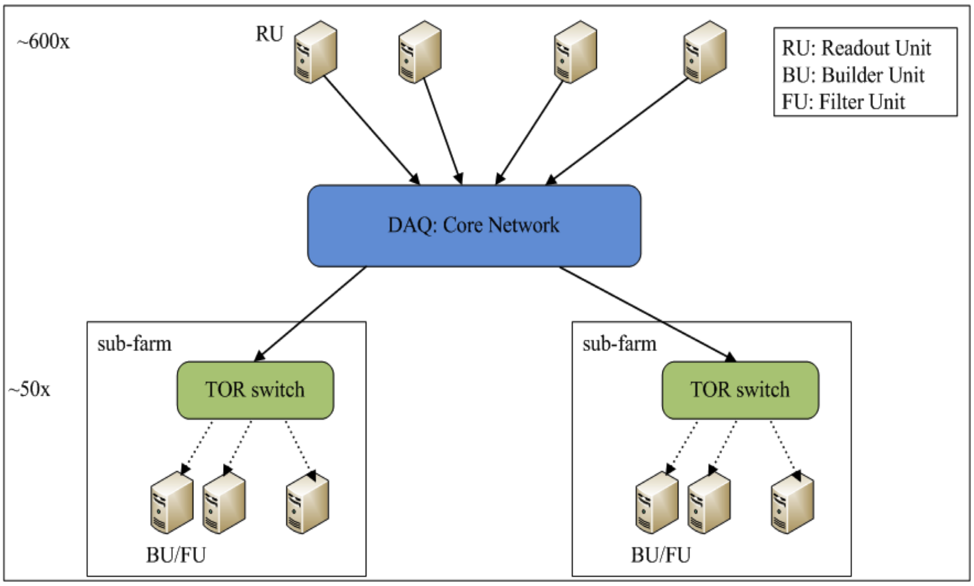

Figure 1. Unidirectional dataflow. Data flows from RUs (top) to BU/FUs (bottom) only.

host. This is the architecture adopted for LHCb in Run 1 and 2.

The advantage of such an architecture lies in its inherent simplicity. The RUs are only sending data. This architecture is therefore particularly well suited to a push-protocol.

The disadvantage is that only half of the bisectional band-width of the network is used and hence the number of core-network ports is high. A minor disadvantage is that in such a system one cannot easily choose a different technology for the core and the distribution layer, one can normally only use a lower speed-grade of the same network technology.

The unidirectional dataflow is illustrated in Figure 1.

2.2. Bi-directional data-flow

In the bi-directional data-flow model one tries to better use the expensive core network, by exploiting its full bisectional band-width. To this end RUs and BU/FUs are mixed in the distribution layer, i.e. RUs and BU/FUs are connected to the same TOR switch. The TOR switches in turn are connected to the core.

The advantage of such a system is the reduction in the number of core network ports by up to 50%.

The disadvantage is the bad speed match between RUs and BU/FUs, which now share the same TOR switch. This will likely lead to under-use of the TORs, thus loosing some of the gains by the reduction of the core. Moreover since the TORs typically have shallow buffers, care must be taken in the shaping of the traffic between RUs and BUs, the event-building protocol will be more complex than int he uni-directional case. The bi-directional data-flow is illustrated in Figure 2.

2.3. Bi-directional data-flow with combined RU and BU

This idea takes the bi-directional data-flow a bit further. Here RU and BU function are combined in the same device. The bandwidth required becomes automatically symmetric (provided all RU receive roughly the same amount of data). These combined entities are connected directly to a relatively small core. The combined unit will be a PC server, where the custom-links are interfaced via a PCIe card [5].

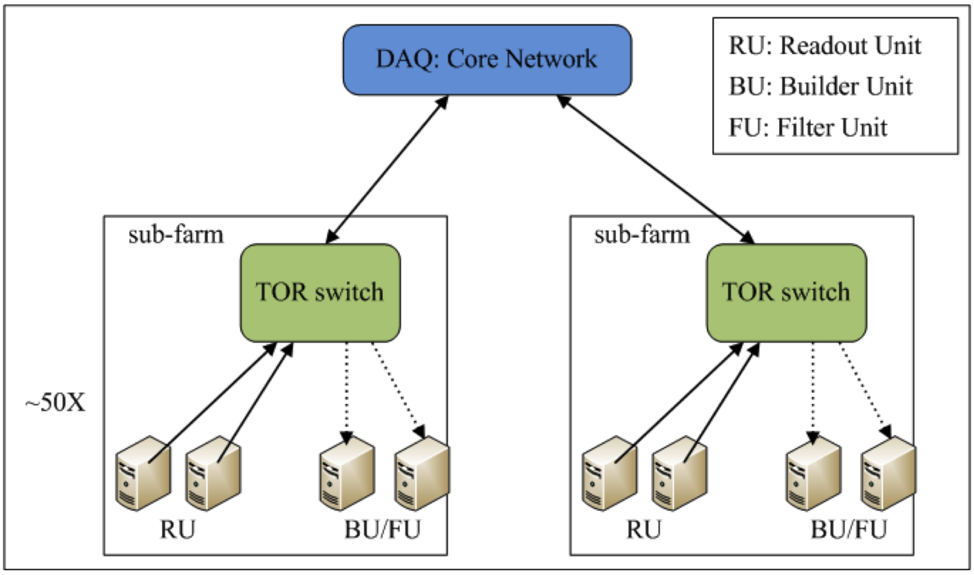

Figure 2. Bi-directional dataflow. RUs and BUs/FUs are connected to the same TOR switches, which are in turn connect to the core network.

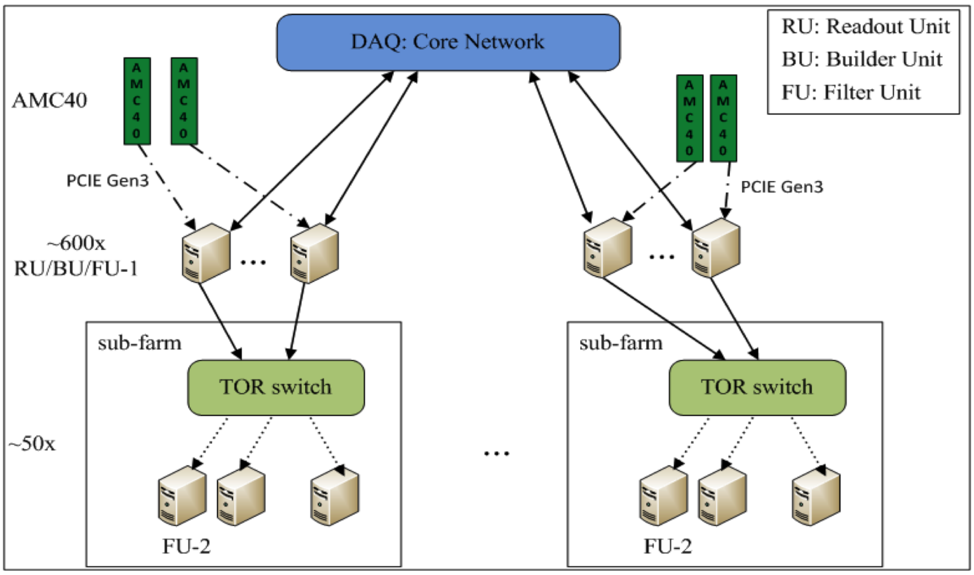

Figure 3. Bi-directional dataflow with combined RU/FU. The AMC40 units convert the propriatary detector links to a LAN technology.

For each eventr (or rather group of events) a node is selected to be the BU. It will have one block from its own RU and get or request the data from all the other RUs to build a complete event. The complete event is then passed on to the FU. The FU functionality can run on dedicated nodes as well as on the server which runs the RU/BU. This architecture is illustrated in Figure 3. In addition to the advantages of the bi-directional dataflow discussed above, this adds a very powerful separation between the technologies for the fast building network and the event-distribution network. The FUs are typically CPU-limited and do not need a high-speed network. They will be very well served with a 10 Gbit/s connection. Moreover the server, which serves the RU/BU function provides a large amount of buffering, which is very convenient

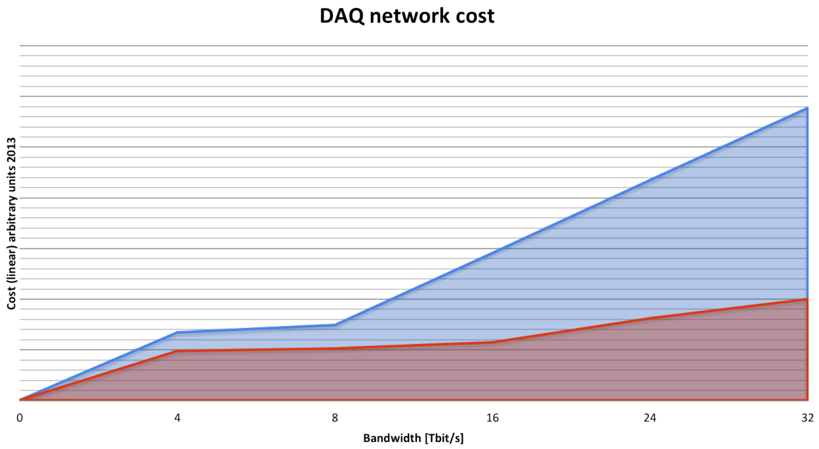

Figure 4. Network cost in arbitrary units(y axis), comparing the unidirectional (blue) and bidirectional with combined BU/RU (red) models

for event-building and a push data-flow from the detector. The combination of these features allows to build the cheapest system, where we compare building networks based on the three discussed architectures. The cost-analysis has been done using known prices (2013) for network technologies and using Ethernet and InfiniBand. The networks provide the same number of usable ports and the same capacity for event-building. The result can be seen in Figure 4. The disadvantage is essentially the increased load and the complexity of the software in the combined RU/BUs. Tests elsewhere [5] show that the resource-loads are perfectly managable already today, so there will be no problem in 2018 either. The complexity is higher than in a unidirectional network but not more complex than some other past and future event-building protocols.

3. Conclusions

Building a multi-terabit data acquisition network is now feasible at an affordable cost. In this paper we have discussed various possible network architectures to achieve this. All have advantages and disadvantages, usually trading in cost for complexity. However we are convinced that in particular for the third option the cost and technological advantages outweigh by far the disadvantages and it is this solution which we are now actively pursuing, leaving the unidirectional network as a proven fall-back solution.

References

[1] Bediaga I *et al.* (LHCb collaboration) 2012 Framework TDR for the LHCb Upgrade: Technical Design Report Tech. Rep. CERN-LHCC-2012-007. LHCb-TDR-12 CERN Geneva

[2] C Gaspar, F Alessio, L Cardoso, M Frank, J C Garnier, E v Herwijnen, R Jacobsson, B Jost, N Neufeld, R Schwemmer, O Callot, B Franek 2011 The LHCb Experiment ControlL System: on the path to full automation *ICALEPCS2011 Contributions to the Proceedings* available: [Online] http://icalepcs2011.esrf.eu/proceedings.htm

[3] Frank M, Gaspar C, Neufeld N and Jost B 2014 Deferred High Level Trigger in LHCb: A Boost to CPU Resource Utilization *Proceedings of the 20th International Conference on Computing in High Energy and Nuclear Physics (CHEP2013)* ed Groep D

[4] Cherukuwada S S and Neufeld N 2008 *IEEE Trans. Nucl. Sci.* **55** 278–283

[5] Marconi U, Schwemmer R, Galli D, Vagnoni V, Jost B, Neufeld N, Lax I and Durante P 2014 A PCIe GEn3 based readout for the LHCb upgrade *Proceedings of the 20th International Conference on Computing in High Energy and Nuclear Physics (CHEP2013)* ed Groep D