April 8, 2003

L1 and Subsequent Triggers

Abstract

During the last year the scope of the L1 trigger has changed rather drastically compared to the TP. This note aims at summarising the changes, both in trigger philosophy and the implementation of the L1 trigger.

1 Introduction

The nomenclature of the trigger levels is the following throughout this note:

L1 this refers to all triggers which run at the output rate of L0, i.e. at a maximum rate of 1.11 MHz. While during TP times only the VELO was considered to provide data for this trigger, now the aim is to allow all the data which could be needed to be used at this rate.

HLT The Higher Level Triggers used to be called L2 and L3, but now it refers to the algorithms which are used on the on-line farm after the L1 decision, and which have access to the full event data.

The main lesson we learned from the evolution of the triggers is that however well we will simulate our data now, we will not be able to predict the distribution of our CPU power over the different trigger levels because of the generator, the noise, the background and the material distribution in our simulation compared to the real experiment.

The optimisation of LHCb, so-called LHCb-light, allowed the L1 trigger to evolve to a more robust trigger by including the information from TT, and hence the L1 implementation has to cope with more data. In contrast the total event size is reduced by about a factor two compared to what was assumed for the DAQ-TDR, due to a combination of less background in the detectors, and a reduction of the number of tracking planes.

The way to make L1 more robust against the unexpected is to allow L1 to in principle use all data, which will give us the flexibility to adapt the algorithms to achieve an optimal efficiency. It is clear that the CPU power in the L1 trigger will constitute a much larger fraction of our total online CPU power, say 1200-1500 nodes in 2007, compared to what was estimated at the time of the TP. The way the total CPU power which we can afford for both L1 and HLT will be distributed over the L1 and HLT triggers has to be flexible, based on what the real data will look like, or what the physics priorities will be at that time.

Furthermore, given that LHCb will not be able to increase the luminosity, or rather the B-yield of the L0 trigger, in any significant way after the LHC reaches its start-up luminosity of $10^{33}\text{cm}^{-2}\text{s}^{-1}$, the only way to increase our yield is to increase our trigger efficiency. The final aim is to approach an “off-line” like selection of the data at the L0-output rate, thus minimising losses.

Given the new requirements for L1, also the implementation has changed. The pursued option is to use the same implementation as used for the DAQ, and include the L1 implementation in the DAQ project. An overview of the combined L1&DAQ is given in the next section.

Given the above general considerations we have to define concrete targets for the start up phase of LHCb, i.e. what we will budget in the Trigger-TDR, and what will be the upgrade path after that. We assume the following to be installed in 2007:

- a CPU farm with at least 1200 nodes.
- L1 will access the data from the L0DU, VELO and TT.
- the FE of the following subsytems will be prepared to allow inclusion in L1: OT, IT, muons and the data from the L0-CALO-Selection boards.

Then after the start-up, and given that we then have real data, we should define our upgrade path. The most likely scenario will be that we will eventually want to do off-line like reconstruction at the highest possible rate, hence we want to include the muons, OT, IT and calorimeter data in L1, and increase the CPU power of the farm. This will roughly double the size of the L1 data.

2 Constructing the LHCb software triggers

After a short review of the impact of the new requirements on the DAQ, we give an overview of the infrastructure for a combined L1&HLT DAQ system. By “software triggers” we refer for short to all trigger levels after L0, when talking about the common infrastructure aspect of the trigger. The system described here is largely based on the architecture devised originally for the DAQ system, which was designed to provide the complete event to the HLT. Since the details of this system are described extensively in the Online TDR [1], we give here only an overview which attempts to put the various components in place.

LHCb light has changed the requirements and base parameters for the DAQ drastically. There are now many more data-sources (i. e. ultimately front-end boards) used or at least potentially used in the L1 trigger and the corresponding event size has grown (to some six kByte in the initial version of the system. On the other hand the reduced number of detector stations has reduced the even size for the complete readout required for the HLT to 40 kByte[1]. As described in the introduction it also became clear that it is very desirable to be able to

1. add parts of the detector to the L1 trigger, if resources permit and physics performance requires it
2. redistribute the total available CPU power seamlessly between the various levels of the software triggers.

The first item calls for a highly scalable system, where cost is a monotonous (albeit not always linear) function of the size, facilitating staging. The second point on the other hand requires that all CPUs are reachable from both readout systems, which quickly leads to the conclusion that this is most easily achieved by having only one readout system. The architecture described in the Online TDR, based on Ethernet, is believed to fulfil both requirements.

[1] The aggregated data traffic is now dominated to a large extent by the L1 data!

3 The Gigabit Ethernet Implementation of the Software Triggers

Figure 1 shows a block diagram of the system. The main features of this system are the following:

- Functionality of the hardware and software components is well defined and attempts to be simple, so that the components avoid strong coupling.

- Data flows from the top to the bottom in a write through manner. Each source sends data as soon as they are available. Buffer control is centralized via the Timing and Fast Control (TFC) system, which throttles (i.e. disables temporarily) the trigger to avoid overflows.

- Both data flows, the L1 and the HLT, use the same network infrastructure[2].

- All data travel as Ethernet packets. The connectivity between the data sources (ultimately the front-end electronics boards) and the CPU farm (ultimately a single compute node for each L1 or HLT event) is provided by a large (high end but commercial) Ethernet switching network (a Local Area Network or LAN).

- Fast Ethernet packet processing, which is necessary upstream (before the switch in the Multiplexing/Readout Unit layer) and downstream (before the Subfarm Controller) to optimise the utilisation of the network and support commodity hardware is done at all places using a single dedicated Network Processor (NP) based module, whose principles are described in the Online TDR [1].

- The CPU farm is sub-divided into sub-farms, consisting of an entry point (the Subfarm Controller) and several worker CPU nodes. The subfarm controller distributes the events among its nodes, balancing the load while doing so.

- The sub-farm nodes execute the algorithms for the L1 and the HLT triggers in parallel, the L1 runs in a high priority thread and is pre-empting the HLT processes, which do not suffer from latency restrictions due to limited front-end buffers.

- The decisions from the L1 processes travel back through the system as Ethernet packets. They are sorted by the L1 decision sorter, which is then transmitting them to the TFC system[3].

- All components mentioned above, including the actual physics trigger algorithms, are controlled, monitored and configured by the Experiment Control System (ECS).

From the above list we can already see a few building blocks of the complete system (numbers in brackets correspond to the projects listed below):

[2] In a similar manner as DECNET and IP traffic share for example the same Ethernet infrastructure.

[3] The decisions must be sorted because of the FIFO organization of the L1-buffers in the front-end electronics boards.

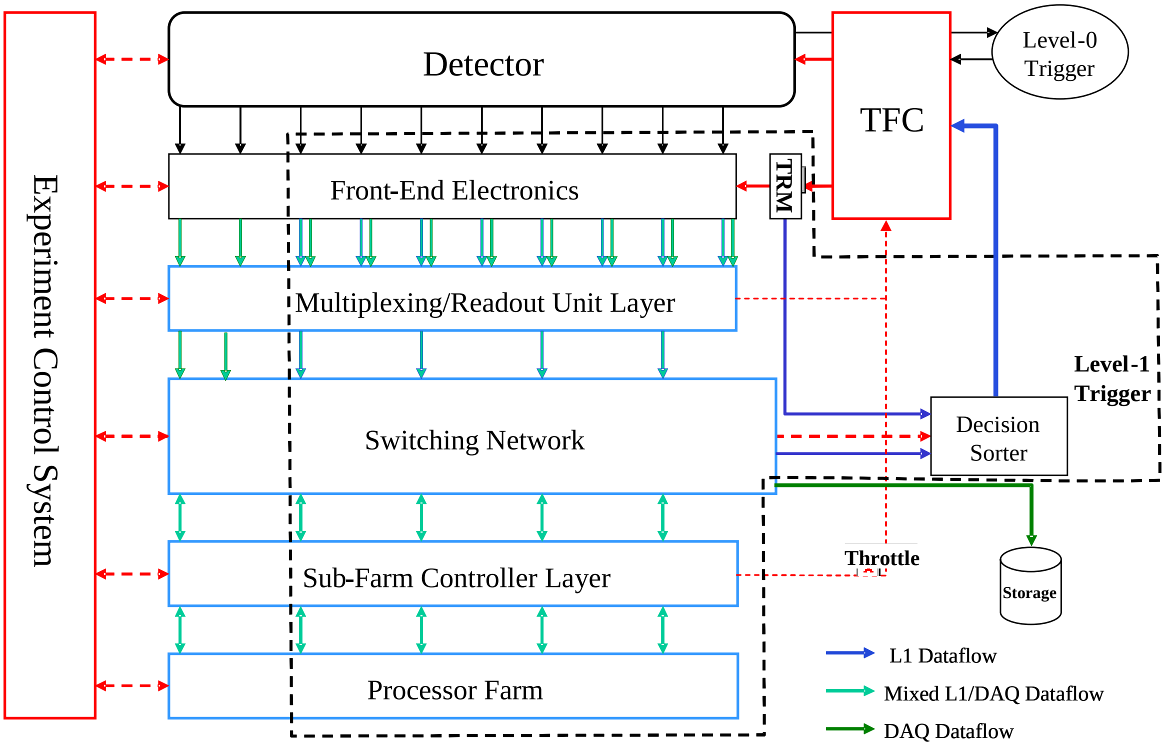

Figure 1: The combined system for the L1 trigger and the HLT based on an Ethernet switching network

The data acquisition (DAQ) in the narrower sense is taking the data from the output of the Level 1 front-end electronics boards through the main event builder switch[4] to the input of one of the subfarm controllers (8).

The whole DAQ system needs to be configured (e.g. forwarding tables in the switch need to be calculated) and monitored by the Experiment Control System (4).

The sub-farm controller forwards the events to one of its worker nodes. It tries to distribute the load evenly between them. It receives trigger decisions and accepted events and forwards them to the appropriate recipient (2).

One such recipient is the L1 decision sorter, which keeps track of event entering the system and decisions made about them and forwards the sorted decisions to the TFC system (9).

When a farm node receives the data it must convert them into the format used by the GAUDI framework. This has to be done as fast as possible, because the time available is limited by the size of the L1 front-end buffers (3).

When the trigger algorithms run, they need to be configured and monitored by means of the ECS (5).

As has been mentioned already all software and hardware components of the system are configured via the ECS. The actual configuration data themselves are stored in the

[4] Although not mentioned in the list of projects explicitly, the selection and integration of a suitable core event builder switch is a very challenging and essential sub-project. It is now listed under the DAQ for two reasons: first it needs good contacts to people and companies which produce and use such high end devices (which is the case at CERN with the IT/CS group), second the switch can in principle be built out of network processor modules (as a fallback solution). But if somebody has the necessary contacts/expertise, this would make a very important project.

configuration data-base (7).

The whole system is evaluated and tested both in a comprehensive top-down simulation (6). A small-scale system is implemented in the software trigger testbed (1).

The division into sub-projects given here can of course be refined, but it is believed that all the projects are reasonably self-contained so that they can be developed in parallel. Obviously good communication between related projects is essential. Except for the DAQ and the testbed, all projects can be developed basically anywhere. The testbed as a physical installation and the DAQ relying heavily on the network processor hardware will be based CERN.

4 Tentative list of projects / work-packages

We list here the following projects, along with some rough estimate of what is involved in terms of resources (hardware / manpower). Where we know we have listed institutes interested or already involved. Also the estimated duration of the project together with the preferred starting date and the required completion date are named. The scope of the projects can certainly be shrunk, expanded, refined etc. However it is understood that commitment to any of the projects includes commitment to commissioning, debugging and maintenance at least in the start up phase of the experiment.

1: DAQ testbed
Description: set-up of a test-bed for the DAQ system, capable of exercising all parts of the data-flow, event-building and event distribution. The system should substitute components by emulators where they are not yet available and be extensible to include finally also the TFC and ECS.
Hardware resources: Network Processors, some PCs, core switch for event building, (in a 2nd stage) sub-farm nodes, sub-farm switch.
Manpower: 1 FTE
Duration: 1 year, can and should be staged to get results by Q3/2003 if possible (for Trigger TDR)
Start date: as soon as possible
Completion date: Q1 to Q2/2004
Institutes: CERN, Lausanne
Related projects: all

2: Farm implementation
Description: Study the software required for operating the sub-farms, the Subfarm Controller (SFC), load balancing, event distribution, interfacing to ECS, software throttling (via ECS), selection of suitable hardware for farm-nodes and SFC
Hardware resources: some PCs, later a switch - this can and will be
tested in the testbed
Manpower: 2 FTE
Duration: 2 years
Start date: Q3/2003
Completion date: Q3/2005 (software) Q2/2006 (hardware selection)
Institutes:

Related projects: Testbed, Farm Installation (CERN, Ph. Gavillet), Dataflow

3: Population of Transient Store
Description: Efficient reception and reformatting of the events in the subfarm node, to present them to the GAUDI algorithms, invocation of the latter Hardware resources: standard PC to start with, high end PCs (test-bed) later for benchmarking
Manpower: 1 FTE
Duration: 1 year
Start date: has started
Completion date: Q2/2004
Institutes: Rio
Related projects: Testbed, Farm implementation, Offline software

4: DAQ control
Description: all control and configuration aspects of the DAQ: setting up of routing tables, loading and monitoring the NPs, switch, interface to ECS and configuration data-base
Hardware resources: to start with none, later to be used in the testbed
Manpower: 2.5 FTE
Duration: 2 years: this project is staged into a design and definition phase and then implementation of many rather independent sub-items, which can be delivered whenever ready
Start date: Q3/2003
Completion date: Q3/2005
Institutes:
Related projects: testbed, DAQ, ECS, Configuration Database

5: Algorithm Control
Description: Starting from existing prototype work to bring the control and monitoring of the trigger algorithms (both L1 and HLT) via ECS into production Hardware Resources: some PCs
Manpower: 1 FTE
Duration: 2 - 3 years (maybe not full-time, need to follow offline developments closely)
Start date: has started - can be taken over eventually
Completion date: Q2/2006
Institutes: currently CERN
Related projects: ECS, Offline Software

6: Simulation
Description: to provide a complete simulation of the complete data flow system and farm from a L0 yes, with timing, simulation of the test-bed
Hardware resources: none
Manpower: 1.5 - 2 FTEs
Duration: 1 year
Start date: has started
Completion date: Q2/2004, but be ready to update it with eventual changes in the system to come up at later stages!
Institutes: CERN + Ferrara (?) + ??

Related projects: testbed, DAQ

7: Configuration Database
Description: to devise a schema for the configuration database, choose an implementation, implant the DB itself plus associated tools: (browsers, extractors, etc.)
Hardware resources: none
Manpower: 2 to 3 FTEs
Duration: 2 years
Start date: as soon as possible
Completion date: First prototype Q2/2004 (when procurement starts) and final Q2/2005
Institutes: CERN + ??
Related Projects: ECS, DAQ, DAQ Control

8: DAQ
Description: To implement the transport of the data from the FE electronics into the sub-farm, including all associated frame-merging, processing, final event-building, select suitable switch for event builder network
Hardware resources: Network Processors, PCs, core switch / router
Manpower: 3 FTE
Duration: 3 years
Start date: has started
Completion date: Q1/2005 (base system at ”40 kHz” for detector commissioning) Q1/2006 for full 1 MHz system
Institutes: CERN
Related Projects: all

9: L1 Decision Sorter
Description: to receive and sort the L1 decisions and transmit them to the Readout Supervisor. An attempt is going to implement this using the NP module. If this fails, custom hardware is needed.
Hardware Resources: None, if implemented as an NP module; electronics design capability, if custom board
Manpower: 1 FTE
Duration: 1 year
Start date: has started, might need a restart
Completion date: Q3/2004
Institutes: CERN (if NP) else ??
Related projects: DAQ

References

[1] *LHCb Online System*, TDR, CERN/LHCC 2001-040, Dec. 2001