FutureDAQ for CBM: On-line Event Selection

H.G. Essel
On behalf of the CBM collaboration

***Abstract*—At the upcoming new Facility for Antiproton and Ion Research FAIR at GSI the Compressed Baryonic Matter experiment CBM requires a new architecture of front-end electronics, data acquisition, and event processing. The detector systems of CBM are a Silicon Tracker System (STS), RICH detectors, a TRD, RPCs, and an electromagnetic calorimeter. The envisioned interaction rate of 10 MHz will produce a data rate of up to 1 TByte/s. Because of the complexity and variability of trigger decisions no common trigger will be applied. Instead, the front-end electronics of all detectors will be self-triggered and marked by time stamps. The full data rate must be switched through a high speed network fabric into a computational network with configurable processing resources for event building and filtering. The decision for selecting candidate events requires tracking, primary vertex reconstruction, and secondary vertex finding in the STS at the full interaction rate. The essential performance factor is now computational throughput rather than decision latency, which results in a much better utilization of the processing resources especially in the case of heavy ion collisions with strongly varying multiplicities. The development of key components is supported by the FutureDAQ project of the European Union (FP6 I3HP JRA1).**

I. The New Facilities

The GSI future project FAIR will provide unprecedented accelerator facilities to investigate physics cases in the fields of nuclear structure physics and nuclear astrophysics, hadron physics, physics of nuclear matter, plasma physics, atomic physics, and applied physics. The FAIR accelerators will provide heavy ion beams up to Uranium at beam energies ranging from 2 – 45 AGeV (for Z/A=0.5) and up to 35 AGeV for Z/A=0.4. The maximum proton beam energy will be 90 GeV.

II. The CBM Experiment

A. The Physics Case

The nucleus-nucleus collisions research program of CBM [1] will focus on the search for

- in-medium modifications of hadrons in super-dense matter as signal for the onset of chiral symmetry restoration,
- a deconfined phase at high baryon densities, and
- the critical endpoint of the deconfinement phase transition.

Many of the signatures pursued with the CBM experiment are based on rare processes. To achieve an adequate sensitivity, the detector systems are designed to operate at interaction rates of up to 10 MHz for A-A collisions and up to several 100 MHz for p-p and p-A collisions.

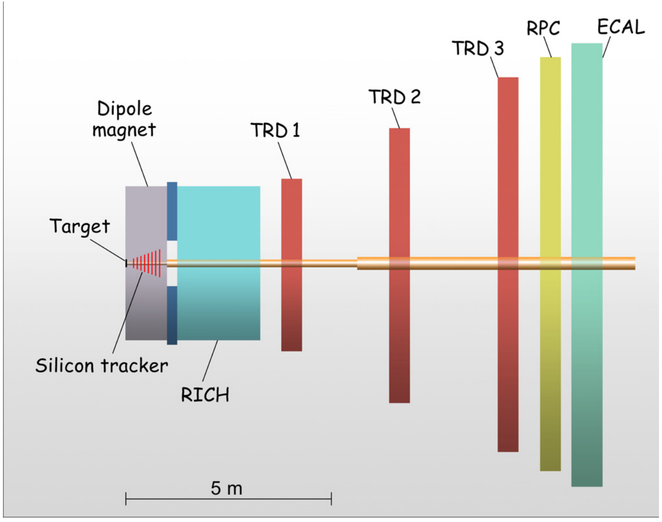

Fig.1. CBM detector setup.

Fig. 1 shows the detector layout. The setup consists of a superconducting dipole magnet with a Silicon Tracker System inside, a Rich Imaging Cherenkov detector (RICH) for electron identification, three Transition Radiation Detectors (TRD), a Time-of-Flight (TOF) wall consisting of a Resistive Plate Chamber (RPC), and an electromagnetic calorimeter.

It is the task of the data acquisition and event selection system to identify the candidate events for the physics signals under study and send them to the archival storage. The most challenging aspect of the detectors is here the measurement of open and hidden charm production in heavy ion collisions down to very low cross sections. The D mesons are identified via the displaced vertices of their decay products, the decision for selecting candidate events thus requires tracking, primary vertex reconstruction, and secondary vertex finding in the STS. In addition, the system has to be configurable to handle a wide range of physics signals, ranging from D and J/Ψ in A-A collisions over low-mass dileptons in p-A collisions to Y in p-p collisions.

B. Triggered DAQ

The conventional system design with triggered front-end electronics allows to keep the event information for a limited time, usually a few microseconds, in the front-end electronics while a fast first level trigger decision is determined from a subset of the data. Upon a positive trigger decision, the data

H.Essel@gsi.de, Gesellschaft für Schwerionenforschung, GSI 64291 Darmstadt, Planckstr. 1, Germany.

acquisition system transports the selected event to higher level trigger processing or archival storage. A system with such a fixed trigger latency constraint is not well matched to the complex algorithms needed for a D trigger, especially in the case of heavy ion interactions, where the multiplicities and thus the numerical effort needed for a decision varies strongly from event to event.

C. *CBM DAQ*

The concept adopted for CBM uses self-triggered front-end electronics, where each particle hit is autonomously detected and the measured hit parameters are stored with precise timestamps in large buffer pools. The event building, done by evaluating the time correlation of hits, and the selection of interesting events is then performed by processing resources accessing these buffers via a high speed network fabric. The large size of the buffer pool ensures that the essential performance factor is the total computational throughput rather than decision latency. Since we avoid dedicated trigger data-paths, all detectors can contribute to event selection decisions at all levels, yielding the required flexibility to cope with the different operation modes.

In this approach we have no physical trigger signal, which prompts a data acquisition system to read a selected event and transport it to further processing or storage. We thus avoid the term 'trigger' in this chapter. The role of the data acquisition system is to transport data from the front-end to processing resources and finally to archival storage. The event selection is done in several layers of processing resources, reminiscent of the trigger level hierarchy in conventional systems.

One consequence of using self-triggered front-end electronics is a much higher data flow coming from the detector front-ends. For CBM, a data rate of about 1 TByte/sec is expected. However, communication cost is currently improving faster over time than processing cost, an observation sometimes termed Gilder's law, making such a concept not only feasible but also cost effective.

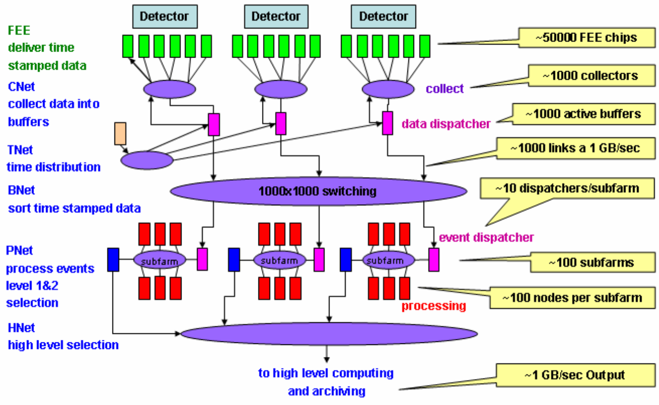

Fig. 2. Overall DAQ architecture

III. Architecture

The communication and processing needed between the front-end electronics, generating digitized detector information, and the archival storage, where the complete context of selected candidate events is recorded, can be structured and organized in several ways. The solution described here is guided by two principles: processing is done after event building and it is done in a structured processor farm. It is well adapted to the type of processing needed in the CBM experiment and leads to a straightforward and modular architecture.

A logical data flow diagram is shown in Fig. 2, indicating the data sources and processing elements as boxes and every form of interconnection networks as ovals. The main components are:

A. *Front-end electronics (FEE)*

The front-end detects autonomously every particle hit and sends the hit parameters together with a precise timestamp and channel address information over the concentrator network (CNet) to a buffer pool. A rough estimate of the data volume generated by a detector channel can be deduced from typical CBM operation and detector parameters: We have 10 MHz interaction rate, 10% occupancy for central collisions, a ratio of 1/4 for minimum bias to central multiplicity, and a typical cluster size of 3 active electronics channels per particle hit. This results in a channel count rate of about 750 kHz. Assuming 8 byte per hit yields a data flow of about 6 MB/sec and channel. For a typical FEE unit with 16 channels this results in a data rate of 100 MB/sec which can be transported over a single GBit serial link.

B. *Clock and time distribution (TNet)*

The timestamps of each hit are used to associate hits with events and also in drift and flight time measurements. Thus not only a time scale, in practice a frequency, but also information about the absolute time has to be communicated to all front-end units. The most stringent requirements come from the START and RPC detectors, where the contribution from the clock jitter should be below 25 ps fwhm.

The most straightforward approach is to distribute a common clock frequency and to provide a mechanism for broadcasting information with clock cycle precise latency to all units. The minimal required functionality is a global clock reset at the begin of the measurement, or alternatively, distribution of tick marks every second as provided by the planned campus-wide frequency and global timing system.

The TNet is thus a dedicated broadcast network, connecting a central controller logically with all front-end units. The last hop to the front-end units may be implemented with the part of the CNet infrastructure, as indicated by the connection of TNet to CNet in Fig. 2.

C. *Concentrator Network (CNet)*

The role of the concentrator network is to collect the data from the individual front-end units and aggregate the traffic on a set of high speed links which connect the detector with the area where the data buffers and the data processing is located. A rough estimate for the total data rate is 1 TB/sec which could be finally transported off the detector with about 1000 links with 10 Gbps each.

The simplest implementation of the CNet is a collection of independent concentrator trees, one for each high speed link. However, a better load balancing or an appropriate degree of failure tolerance is likely to call for a more connected topology. In addition to the hit data transport from the front-end to the buffers, other communication tasks like control traffic or time distribution can be handled by the CNet infrastructure. Such an integrated approach is especially useful in conjunction with low-cost optical links.

D. Active Buffers

The next stage in the data flow is a large buffer pool per link. The units are indicated as (1000) data dispatchers in Fig. 2. They are dubbed 'active buffers' because the data is not only stored but potentially also reformatted and reorganized. They function also as hand-over points between different types of networks, thus logically separating them and allowing for using different technologies in CNet, BNet, and PNet. On the output side of the BNet the active buffers are indicated as event dispatchers (10 per each of the 100 PNet farms). Actually, the two dispatchers will be on one board using bi-directional links in/out of BNet.

E. Build Network (BNet)

For the event selection processing, the data of one event distributed over all data dispatchers must be assembled into one event dispatcher as entry into a farm node. This data reorganization is performed by the build network and the active buffers. In a conventional system, a trigger already defines the context of an event, so all further data processing and transport can be organized in terms of events starting at the FEE. In our case, the FEE sends a stream of time-stamped hits, and it is one of the tasks of the data processing, to first identify at what times interactions occur, and in a second step, to associate the hits with those events.

One possible method for the event definition is to analyze the multiplicity of the silicon tracker hits as a function of the time (from the time stamps). For that the hits of small subsequent time intervals (~2ns) of ca. 50 STS concentrator modules are summed up in the channels of a multiplicity histogram. Peaks in this histogram mark the times of interactions. This *event tagging* processing can be done before or after data traverses the BNet. In the first case, event tagging is handled in the active buffers, and the entities being assembled in the BNet transfers are indeed events. In the second case, only the time stamp information is available, and it is thus natural to assemble all the data of a time interval.

A strict event-by-event approach would lead at the nominal Au+Au interaction rate of 10 MHz to a message rate of 10^{10} messages per second with an average message size of 100 Bytes. However, because the transfer latency is uncritical, it is possible to choose a bigger dispatching unit, either an interval of events, or in the simplest case, a time interval containing a significant number of events. This aggregation reduces the message rate, increases message size, and because event size fluctuations average out, also yields a more uniform message size distribution. A reasonable choice is an event interval of about 100 events or equivalently a time interval in the order of 10 µsec.

Dispatching based on time intervals makes the scheduling simpler. However, tagging the events before the switching allows for the suppression of incoherent background. Fig. 2 indicates that source as well as destination of a BNet transfer is an active buffer. They implement the protocol used on the BNet and are responsible for the organization of the data flow, in particular for traffic shaping and appropriate scheduling of transfers. Because the actual traffic seen by the BNet can be controlled to a large degree and adapted to a given networking technology, it is assumed that the BNet can be based on a commercial off-the-shelf (COTS) technology.

Plausible candidates are Ethernet, Infiniband, or ASI. Fig. 2 only indicates the logical data flow, not a concrete network topology. For one, it is possible to factorize the network in several ways, which allows to build the BNet with a set of medium-sized switches and thus to exploit the usually significantly better price/port ratio of smaller switches. Also, it is possible to merge the data and event dispatchers, one interfacing CNet to Bnet and one interfacing BNet to PNet, into a single entity, leading to a system with half as many BNet ports and bidirectional traffic on all BNet links. Last but not least, the star topology with centralized switches can be replaced with other structures, like a 2D- or 3D-torus topology with distributed switches.

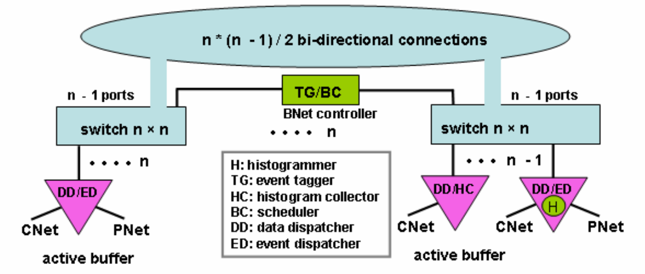

Fig. 3. Generic BNet topology.

Figure 3 shows a schematic view of a factorized switch. With n = 32, 1024 bi-directional channels can be implemented by 496 interconnections. The triangles are the active buffers where the data dispatchers (DD) send the data from CNet through Bnet via the event dispatchers (ED) into the PNet. In addition some of the free inner ports can be used by the BNet controller. Some active buffers also function as histogrammers and histogram collectors. The current approach is to use the BNet itself for all necessary communication of scheduling and histogramming (event tagging).

A simulation framework based on SystemC [2] has been set up for such a configuration with n=10. Fig. 4 shows the simulated occupancy of BNet by the different data types. Over 75% of the nominal bandwidth are used by data tranfers, and only a few percent by meta data.

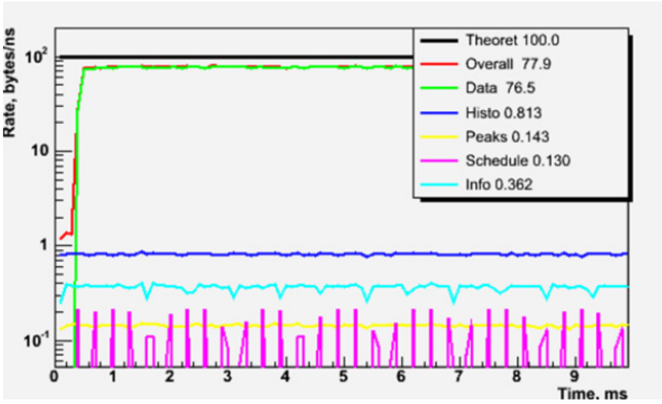

Fig. 4. Network utilization

F. Processing Resources

The first level of event selection processing has to handle the full event rate, and depending how many detector sub-systems are involved in the decision, a substantial fraction of the total data volume. A very rough estimate shows that processing a data flow on the scale of a TByte/sec is likely to require a computational bandwidth on the scale of 10^{15} operations/sec.

With todays technology, the most promising approach is a hybrid system using a combination of hardware processors, implemented with programmable logic components like FPGA's, and software processors, implemented with commodity PC's. The kernels of algorithms which allow highly parallel execution run on hardware processors, the rest in software processors. The aim is to execute most of the operations on hardware processors, which offer the best price/performance ratio for computational bandwidth, but to keep most of the code volume on software processors, which offer much easier program development and maintenance.

Since programmable logic devices are essentially arrays of simple structures, it is expected that they scale well and thus density and speed will improve as the underlying silicon technology evolves. The development of software processors over the relevant time is likely to be more complex. The evolution of single processor speed has apparently reached its limits of complexity and power dissipation, calling for changes in concepts and architectures. This trend is already apparent in recent developments like the compute ASIC for IBM's Blue Gene system or the Cell processor (Sony-Toshiba-IBM [3]). It is also expected that conventional fixed instruction set processor and programmable logic concepts will be coupled, resulting in new forms of configurable computing, like processors with dynamically adaptable instruction sets. A first product in this emerging segment is, for example, the Stretch S5000 series CPUs [4].

The overall architecture is easily adaptable to more integrated forms of configurable computing.

G. Processing Network (PNet)

The processing resources are grouped in farm nodes. Each farm node is organized around a local processing network which provides the communication between the associated processing resources and active buffers, which act as central data repository and as a gateway to the BNet.

The PNet is thus structured into many local networks. The number of hardware and software processors aggregated in one farm node is determined by the amount of resources needed to efficiently handle all the algorithms needed for an event selection decision. Each hardware processor has a dedicated configuration to execute a particular algorithm. In total about a dozen different configurations may be needed. A rough estimate shows that an about equal number of software processors is needed for a balanced load of the whole system.

It is expected that the resources of the farm will be concentrated in a crate or are at least in close proximity. The PNet can therefore use technologies designed for short distance interconnects, plausible candidates are from today's perspective PCIexpress or ASI. As stated already for the BNet, Fig. 2 indicates the logical data flow only, not a concrete network topology. The PNet can be a homogeneous, single technology, switch based star network as suggested by the figure, but many other topologies are possible.

H. High-level Network (HNet)

The task of the event selection processing described up to now is to perform a first reduction, similar to the 'level 1 trigger' in conventional systems. A further reduction will be needed to reduce the data volume to a level suitable for archival storage. This will be handled by a more conventional processing farm. The HNet provides the connection to this high-level computing.

IV. FutureDAQ

We acknowledge the support of the European Community-Research Infrastructure Activity under the FP6 "Structuring the European Research Area" programme (HadronPhysics, contract number RII3-CT-2004-506078). Besides GSI, universities of Heidelberg, Mannheim, Munich, Katowice, Krakow, Warsow, Giessen, Budapest, and Turino are participating.

V. Status

The discussions about the overall concepts as outlined here started in 2004. Simulation frameworks have been set up. A small scale demonstrator for the hierarchy chain will be implemented the next two years.

References

[1] P.Senger, J.Phys.G: Nucl.Part.Phys.28(2002)1869

[2] http://www.systemc.org/

[3] http://researchweb.watson.ibm.com/cell/

[4] Stretch S5000 Technical Overview, http://www.stretchinc.com/products_overview.php