LHCb Trigger and Data Acquisition System

Requirements and Concepts

Beat Jost
Cern / EP
on behalf of the LHCb Collaboration

Presentation given at the
3rd Workshop on Network Based Event-Building and DAQ
Held October 20th 2000 in Lyon, France

Outline

- Introduction
- General Trigger/DAQ Architecture
- Trigger/DAQ functional components
  - Timing and Fast Controls
  - Level-1 Trigger
  - Readout Units/Front-End Multiplexers
  - Sub-Farm Architecture
  - Event-Building Network
  - Controls
- Summary

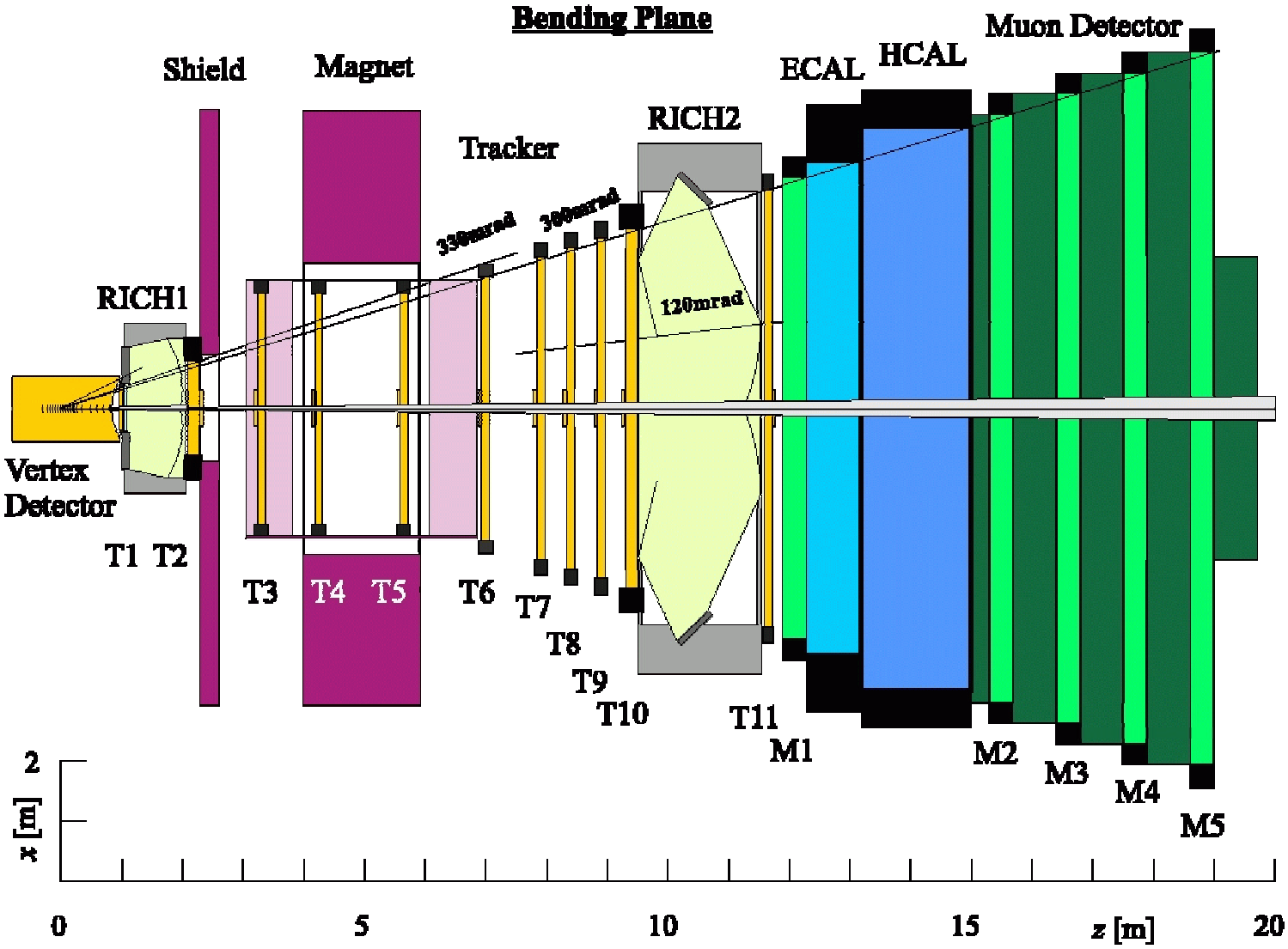
Introduction to LHCb

- Special purpose experiment to measure precisely CP violation parameters in the $B\overline{B}$ system
- Detector is a single-arm spectrometer with one dipole
- Total b-quark production rate is ~75 kHz
- Expected rate from inelastic p-p collisions is ~15 MHz
- Branching ratios of interesting channels range between 10^{-5}-10^{-4} giving interesting physics rate of ~5 Hz

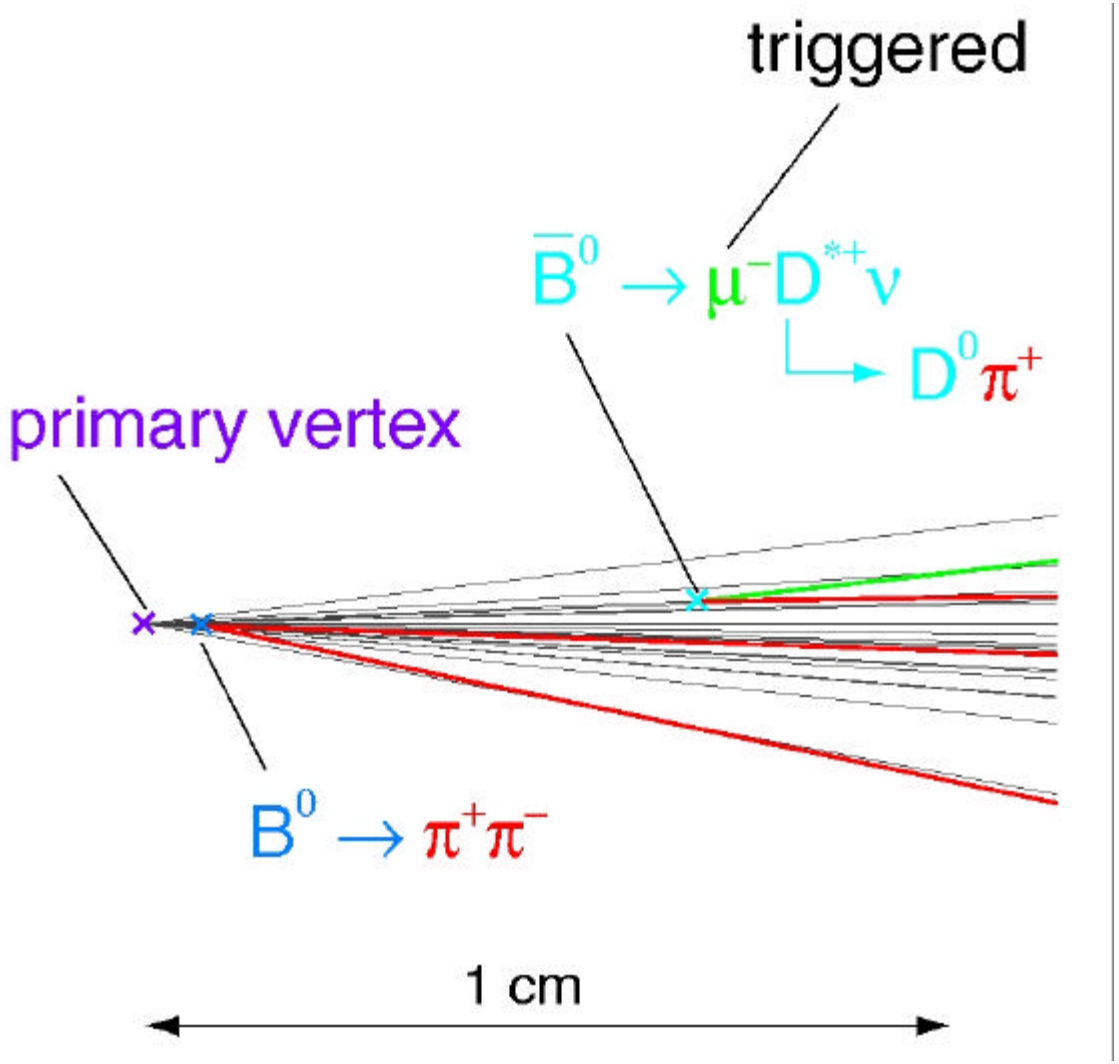
LHCb in Numbers

Number of Channels	~1.1 M
Bunch crossing rate	40 MHz
Level-0 accept rate	1 MHz
Level-1 accept rate	40 kHz
Readout Rate	40 kHz
Event Size	150 kB
Event Building Bandwidth	6 GB/s
Level-2 accept rate	~5 kHz
Level-3 accept rate	~200 Hz
Level-2/3 CPU Power	100 kSI95
Data rate to Storage	~50 MB/s

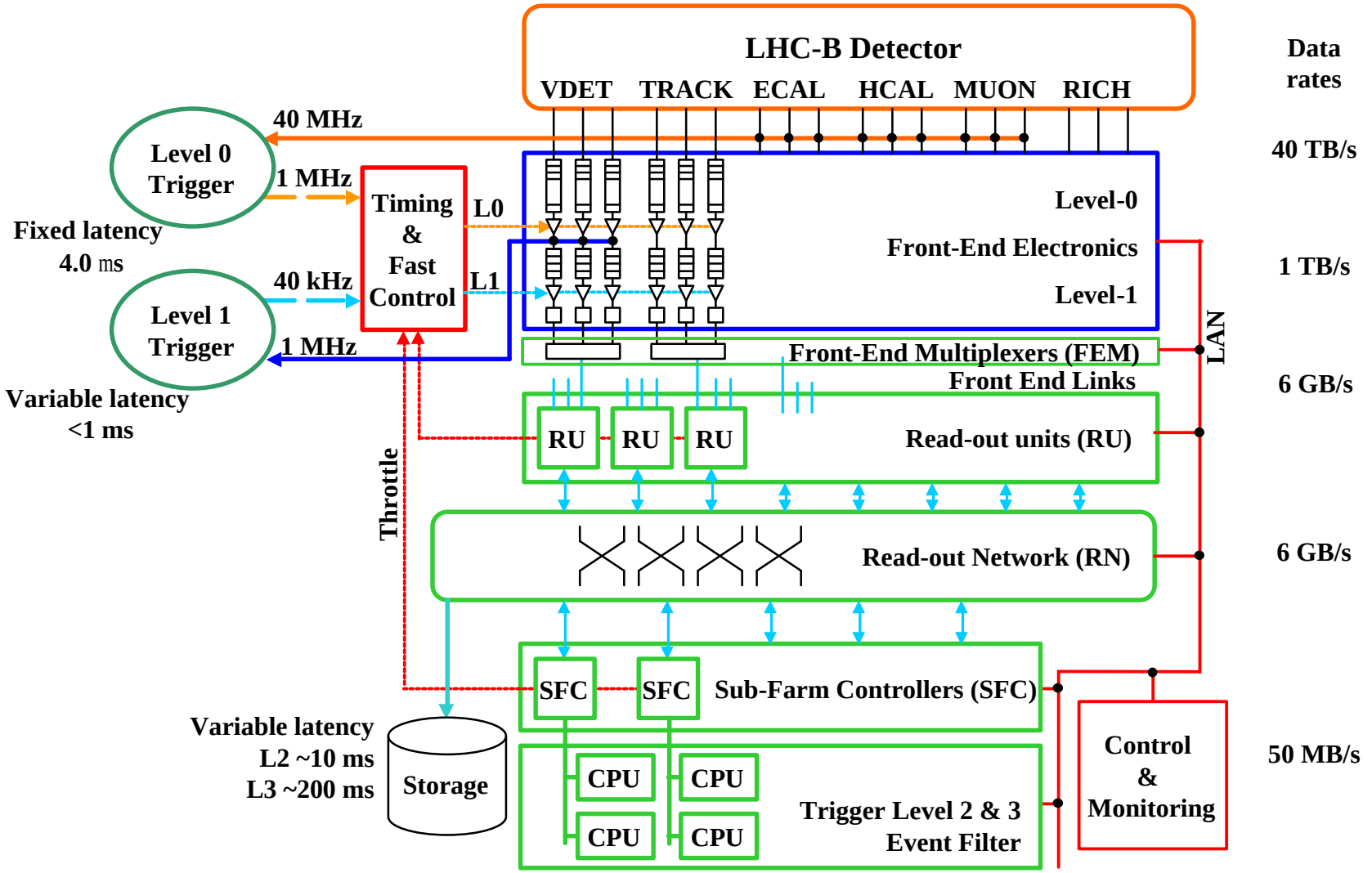
LHCb Detector



Typical Interesting Event

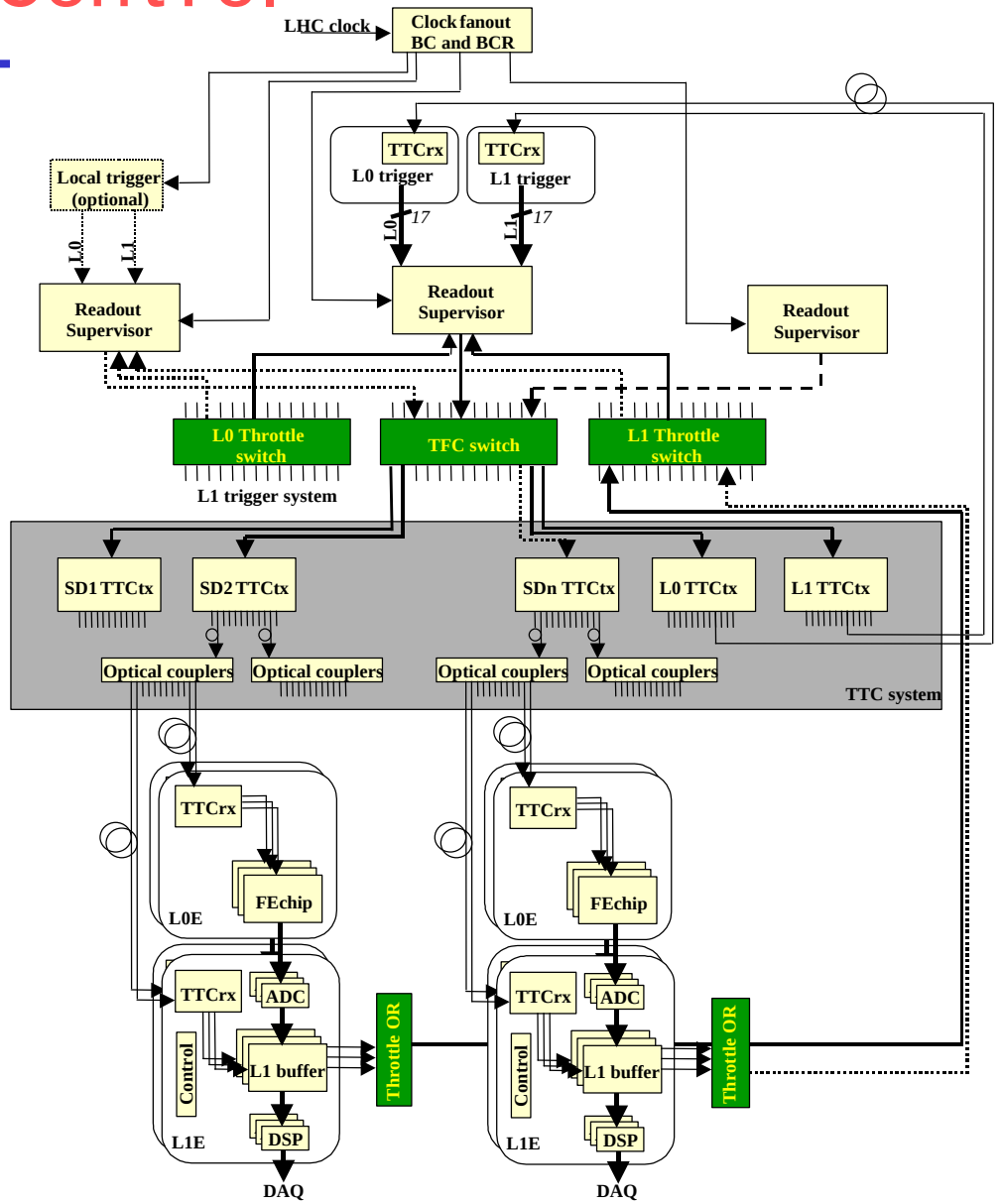


Trigger/DAQ Architecture



Timing and Fast Control

- Provide common and synchronous clock to all components needing it
- Provide Level-0 and Level-1 trigger decisions
- Provide commands synchronous in all components (Resets)
- Provide Trigger hold-off capabilities in case buffers are getting full
- Provide support for partitioning (Switches, ORs)

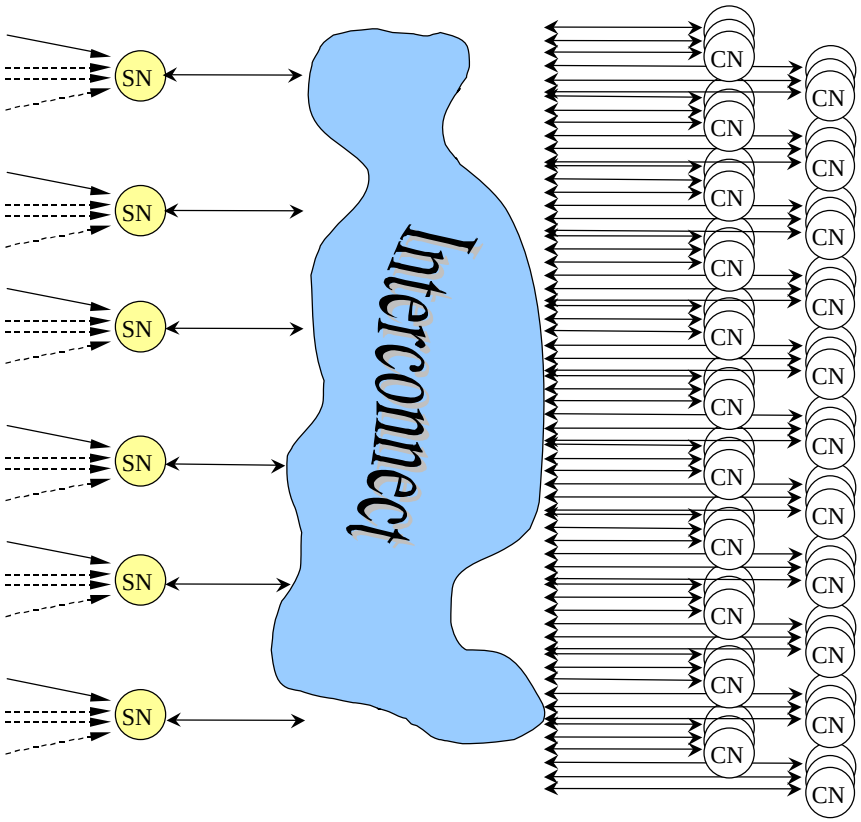


Level-1 Trigger

- Purpose
  - Select events with detached secondary vertices
- Algorithm
  - Based on special geometry of vertex detector (r-stations, φ-stations)
  - Several steps
    - track reconstruction in 2 dimensions (r-z)
    - determination of primary vertex
    - search for tracks with large impact parameter relative to primary vertex
    - full 3 dimensional reconstruction of those tracks
- Expect rate reduction by factor 25
- Technical Problems: 1 MHz input rate, ~4 GB/s data rate, small event fragments, Latency restrictions

Basic Idea:

Network interconnecting the computing nodes of a processor farm to the data sources



Level-1 Trigger (2)

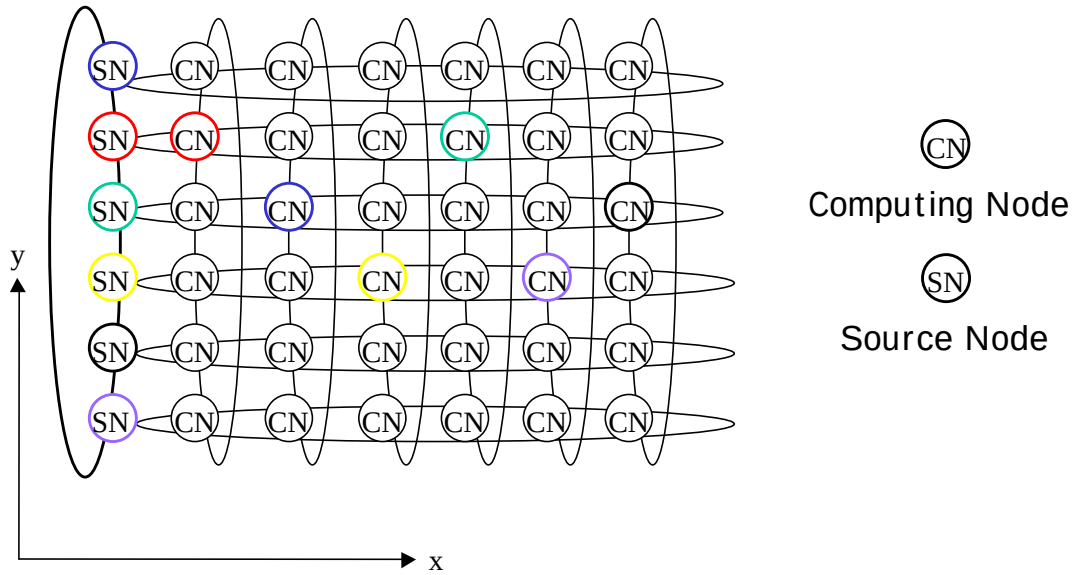
Implementation

- ~32 sources to a network
- Algorithm running in processors (~200 CPUs)
- In principle very similar to DAQ, however the input rate of 1 MHz poses special problems.
- Current studies centered around an SCI based torus topology
  - Simulation of system was done
  - First tests show that data can be written to SCI at 1.5 MHz

Dimensional routing (first x then y).

At any given time not more than one SN must send its data to a certain torus column (see matching color in the sketch).

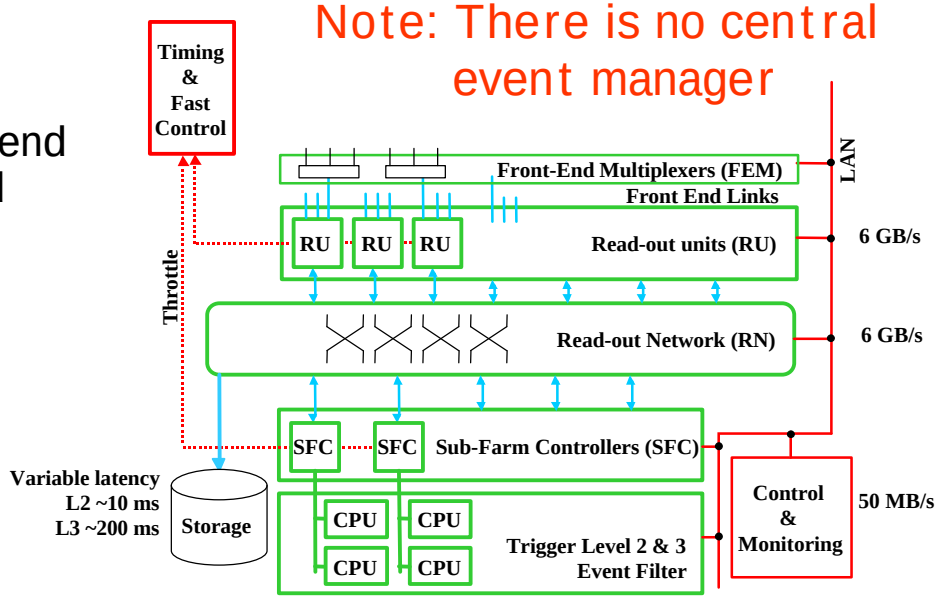
-> need for traffic-shaping



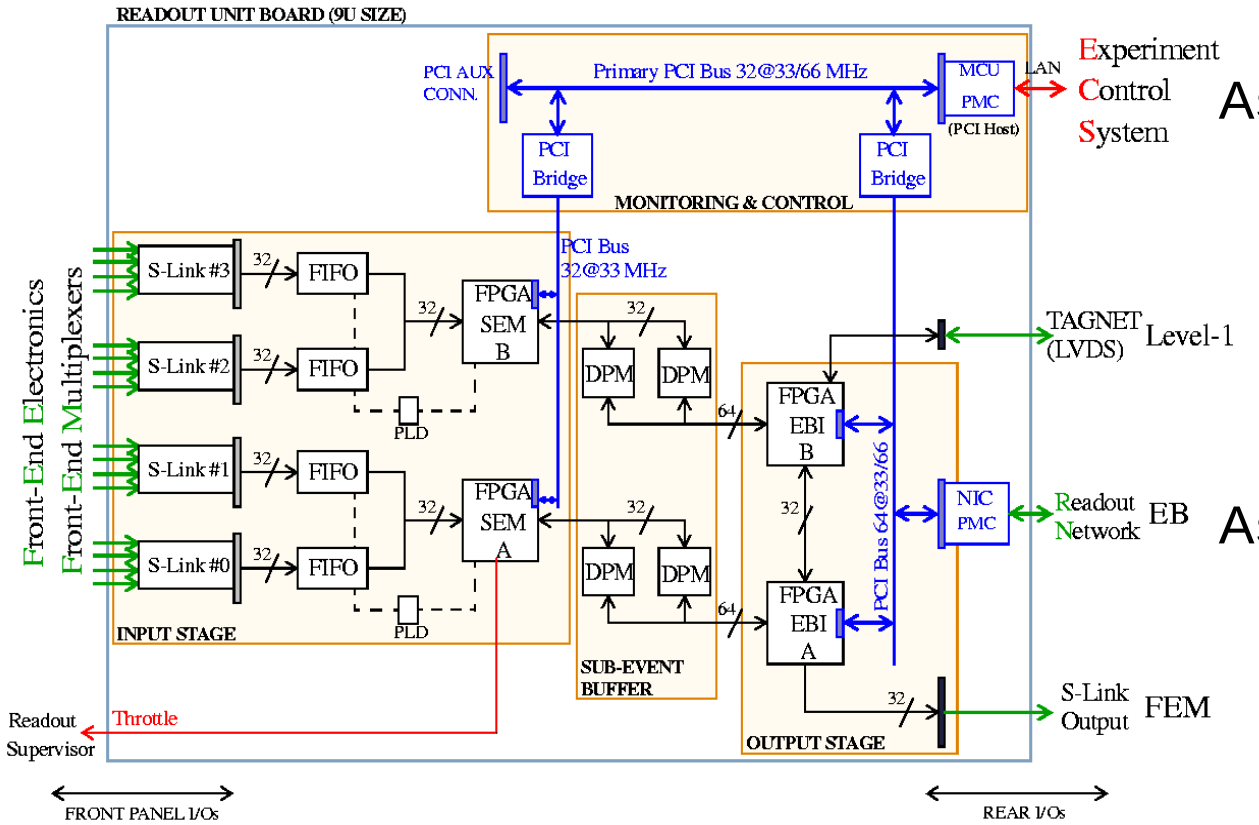
DAQ Functional Components

- Readout Units (RUs)/Front-End Multiplexers (FEM)
  - Multiplex input links (Slink) onto Readout Network links (RU) or Slink (FEM)
  - Merge input fragments to one output fragment
- Subfarm Controllers (SFCs)
  - assemble event fragments arriving from RUs to complete events and send them to one of the CPUs connected
  - dynamic load balancing among the CPUs connected
- Readout Network
  - provide connectivity between RUs and SFCs for event-building
  - provide necessary bandwidth (6 GB/sec sustained)
- CPU farm
  - execute the high level trigger algorithms
  - execute reconstruction algorithm
  - Processing needs: ~100 kSI95, i.e. ~1000 processors

Note: There is no central event manager



RU/FEM Architecture



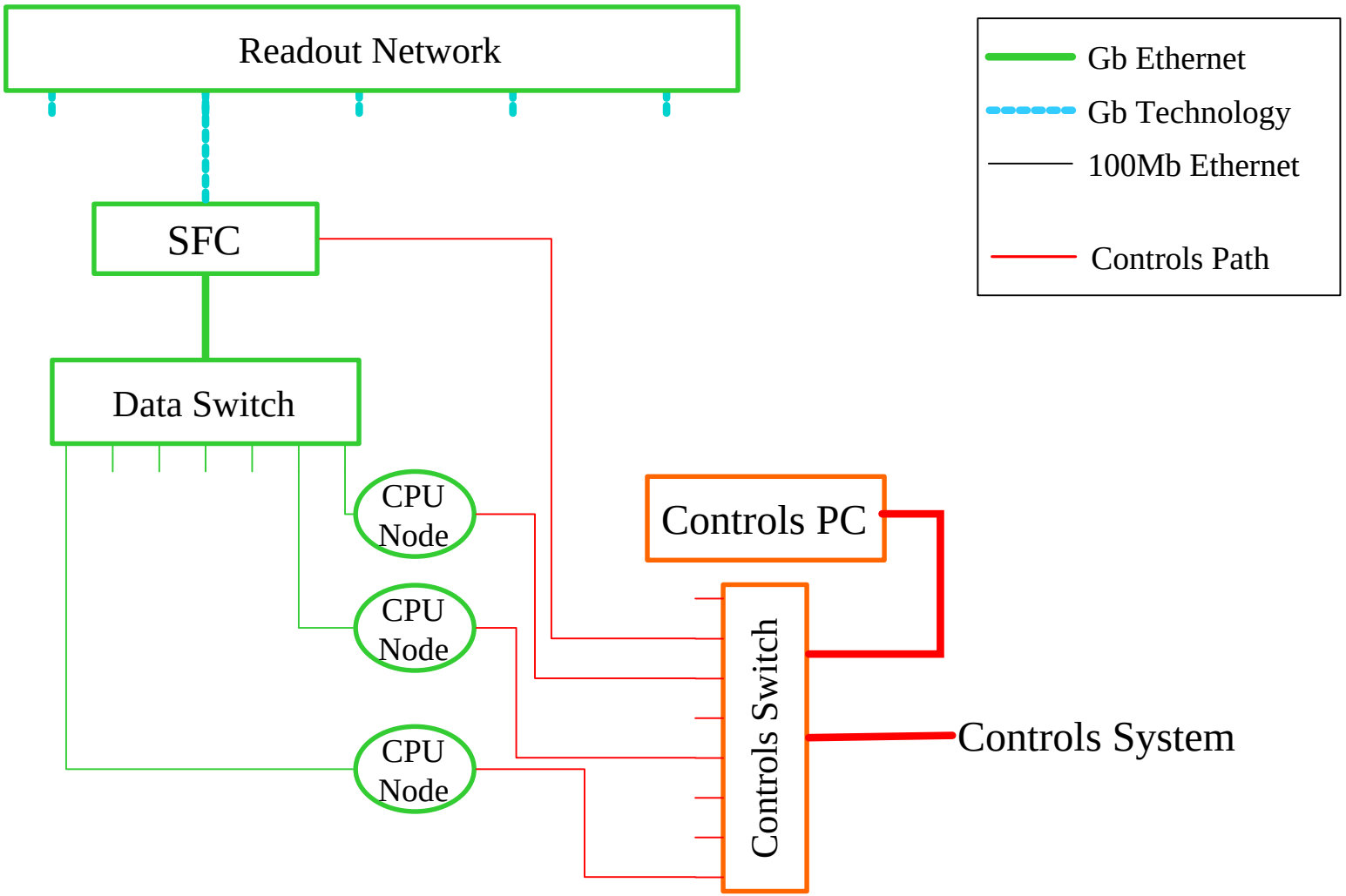
As RU

- 4:1 Multiplexer/EB
- significant buffering
- Output to RN
- possible output blocking
- ~1.5 MHz sub-event building performance

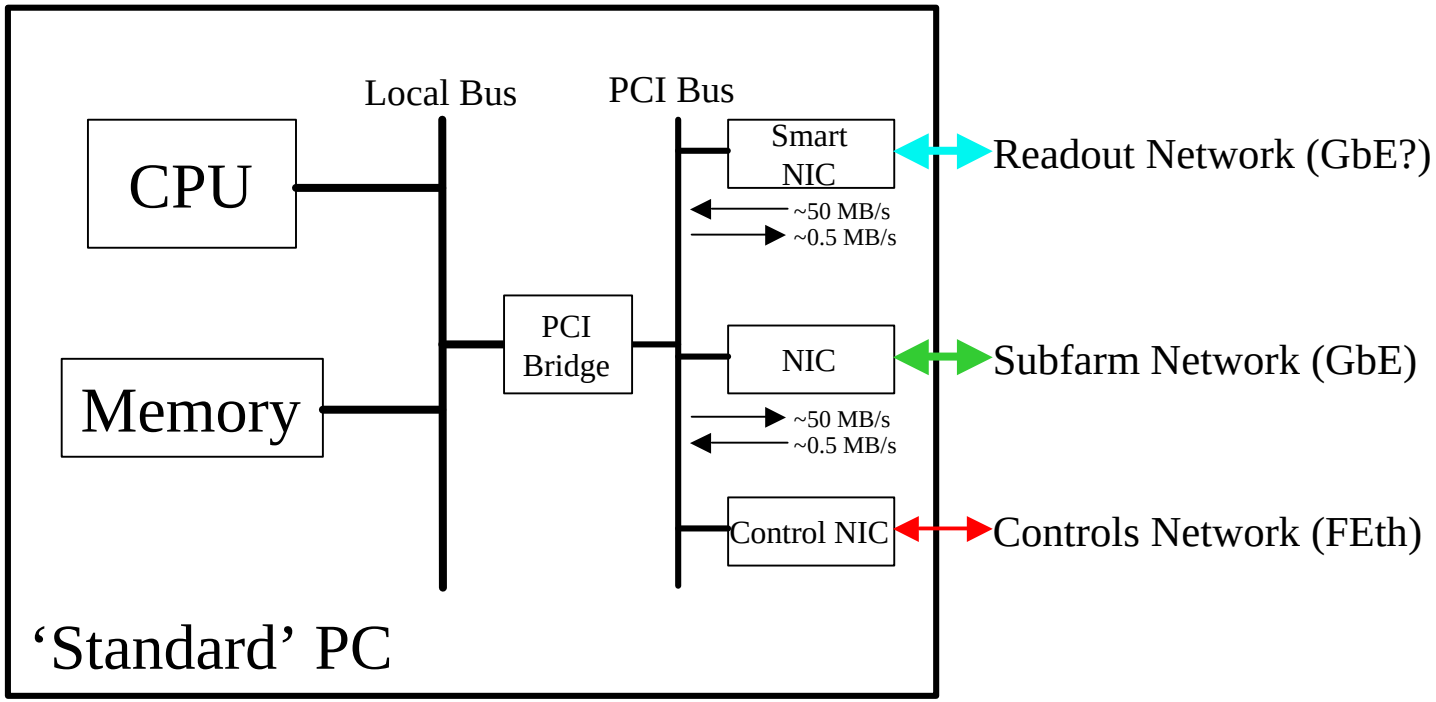
As FEM

- 16:1 Multiplexer/EB
- Minimal/No Buffering
- SLink Output
- no output blocking

Sub-Farm Architecture



SFC Architecture

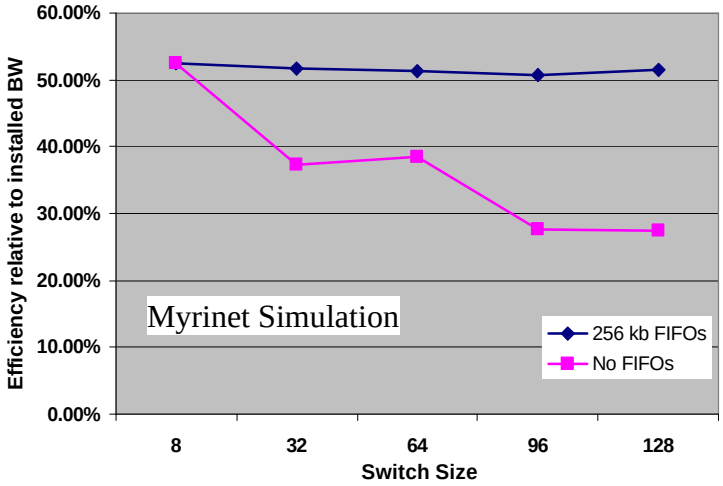


Event-Building Network

- Requirements
  - ➢ 6 GB/s sustained bandwidth
  - ➢ scalable
  - ➢ expandable
  - ➢ ~120 inputs (RUs)
  - ➢ ~120 outputs (SFCs)
  - ➢ affordable and if possible commercial (COTS, Commodity?)
- Readout Protocol
  - ➢ Pure push-through protocol of complete events to one CPU of the farm
  - ➢ Destination assignment following identical algorithm in all RUs (belonging to one partition) based on event number
  - ➚ Simple hardware and software
  - ➚ No central control ® perfect scalability
  - ➚ Full flexibility for high-level trigger algorithms
  - ➘ Larger bandwidth needed (+~50%) compared with phased event-building
  - ➘ Avoiding buffer overflows via 'throttle' to trigger
  - ± Only static load balancing between RUs and SFCs

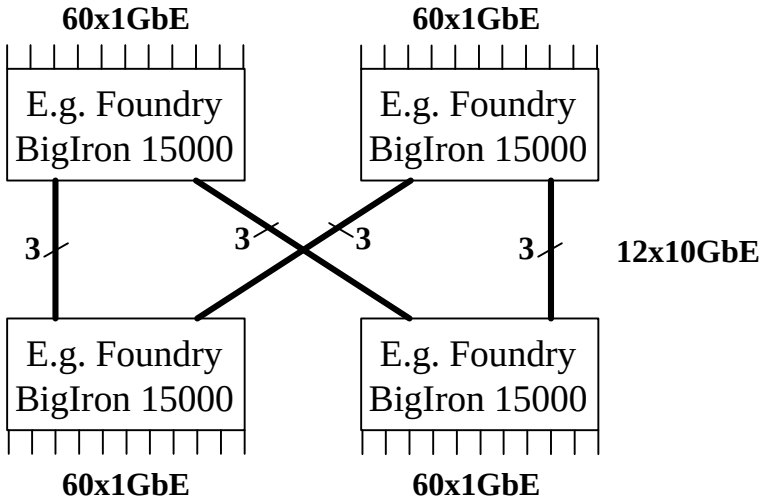
Event-Building Activities (to date)

- Studied Myrinet
  - Tested NIC event-building
  - simulated switching fabric of the size suitable for LHCb
    Results show that switching network could be implemented (provided buffers are added between levels of switches)



- Currently focussing on xGb Ethernet
  - Studying smart NICs (-> Niko's talk)
  - Possible switch configuration for LHCb with ~today's technology (to be simulated...)

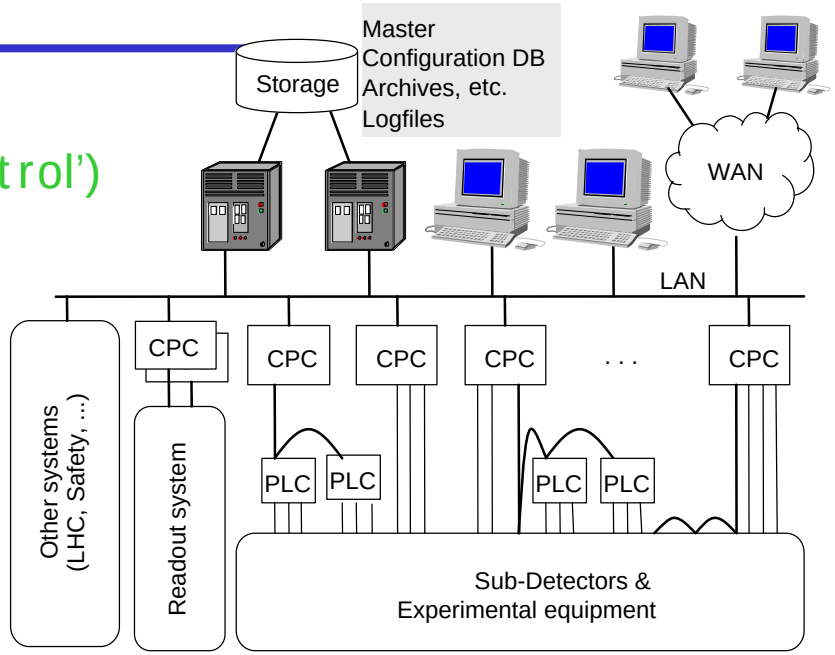
Multiple Paths between sources and destinations!



Controls System

Common integrated controls system

- Detector controls (classical 'slow control')
  - High voltage
  - Low voltage
  - Crates
  - Alarm generation and handling
  - etc.
- DAQ controls
  - Classical RUN control
  - Setup and configuration of all components (FE, Trigger, DAQ, CPU Farm, Trigger algorithms,...)
  - Consequent and rigorous separation of controls and DAQ path



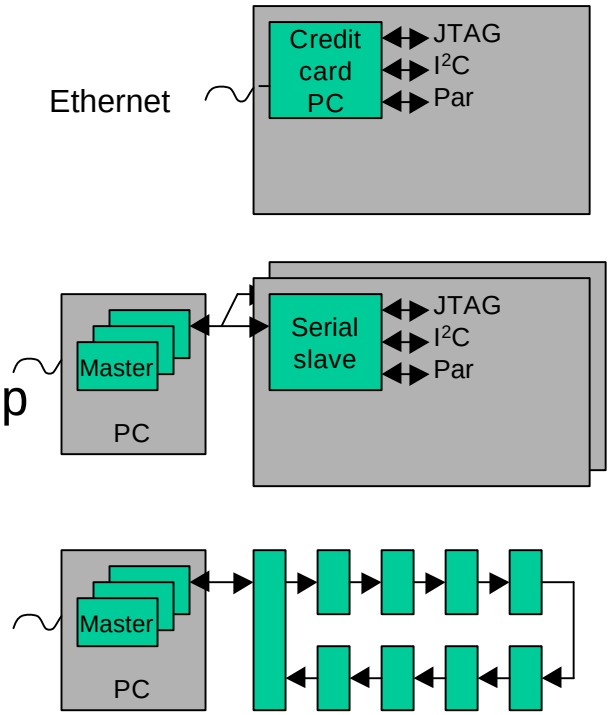
Same system for both functions!

Scale: ~100-200 Control PCs
many 100s of Credit-Card PCs

By itself sizeable Network!
Most likely Ethernet

ECS interface to electronics

- Three solutions
  - No radiation (counting room):
    Ethernet to credit card PC on modules
    Local bus: Parallel bus, I2C, JTAG
  - Low level radiation (cavern):
    10Mbits/s custom serial LVDS twisted pair
    SEU immune antifuse based FPGA interface chip
    Local bus: Parallel bus, I2C, JTAG
  - High level radiation (inside detectors):
    CCU control system made for CMS tracker
    Radiation hard, SEU immune, bypass
    Local bus: Parallel bus, I2C, JTAG

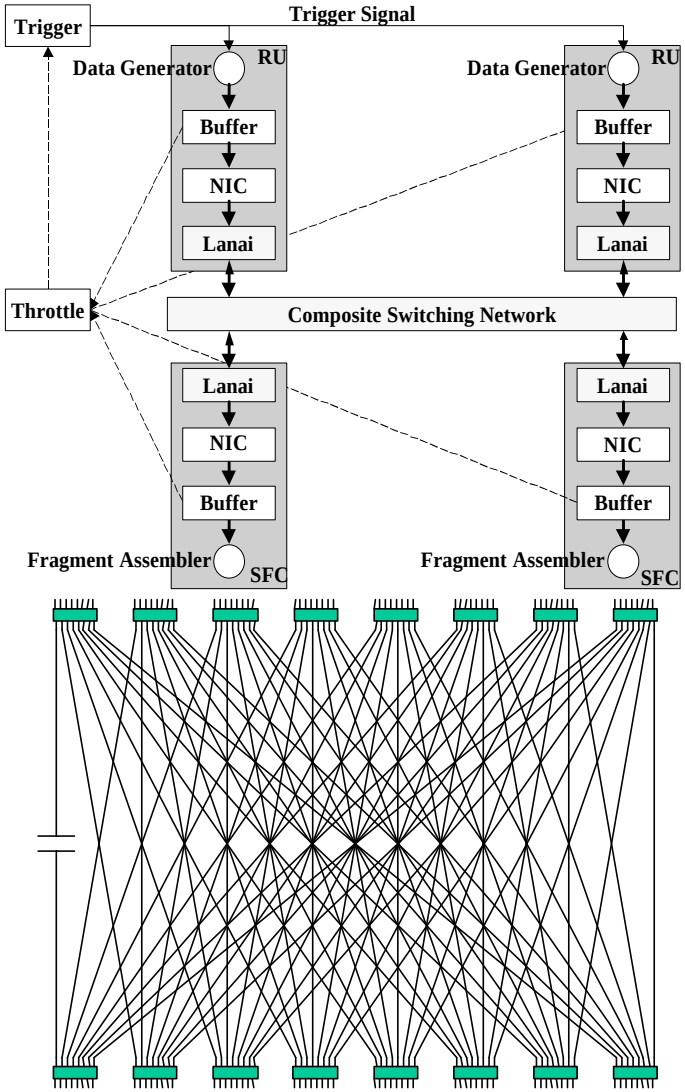


Summary

- LHCb is a special purpose experiment to study CP violation
- Triggering poses special challenges
  - Similarity between inelastic p-p interactions and events with B-Mesons
- DAQ is designed with simplicity and maintainability in mind
  - Push protocol throughout the system
    - Simple, e.g. No central event manager in the event builder
    - no backward communication and definitely no lateral communication
    - Slightly harder bandwidth requirements on readout network (~1.5 times)
  - We are convinced that readout network can be realized at reasonable cost
- Unified approach to Controls
  - Same basic infrastructure for detector controls and DAQ controls
  - Both aspects completely integrated but operationally independent

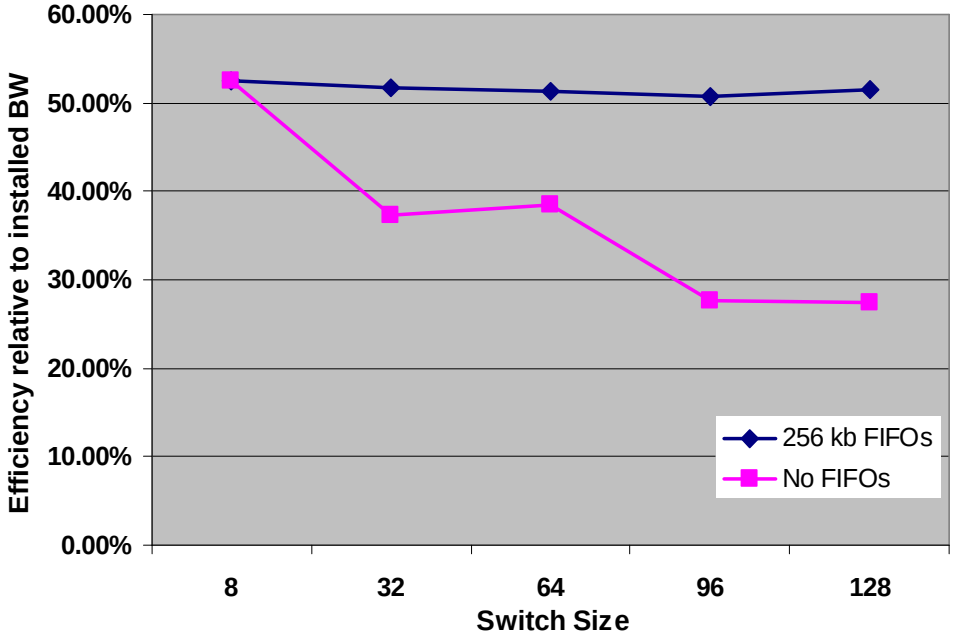
Event-Building Network Simulation

- Simulated technology: Myrinet
  - Nominal 1.28 Gb/s
  - Xon/Xoff flow control
  - Switches:
    - ideal cross-bar
    - 8x8 maximum size (currently)
    - wormhole routing
    - source routing
    - No buffering inside switches
- Software used: Ptolemy discrete event framework
- Realistic traffic patterns
  - variable event sizes
  - event building traffic



Network Simulation Results (Myrinet)

Results don't depend strongly on specific technology (Myrinet), but rather on characteristics (flow control, buffering, internal speed, etc)



Switch Size	Fifo Size	Switching Levels	Efficiency
8x8	NA	1	52.5%
32x32	0	2	37.3%
32x32	256 kB	2	51.8%
64x64	0	2	38.5%
64x64	256 kB	2	51.4%
96x96	0	3	27.6%
96x96	256 kB	3	50.7%
128x128	0	3	27.5%
128x128	256 kB	3	51.5%

FIFO buffers between switching levels allow to recover scalability

50 % efficiency "Law of nature" for these characteristics

Front-End Electronics

- Data Buffering for Level-0 latency
- Data Buffering for Level-1 latency
- Digitization and Zero Suppression
- Front-end Multiplexing onto Front-end links
- Push of data to next higher stage of the readout (DAQ)