



High throughput DAQ systems

Niko Neufeld, CERN/PH

Third I.N.F.N. International School on

**“Architectures, tools and methodologies for
developing efficient large scale
scientific computing applications”**



Ce.U.B. Bertinoro (FC) 23 - 29 October 2011



Outline

- History & traditional DAQ
 - Buses
 - LEP / Tevatron
- LHC DAQ
 - Introduction
 - Network based DAQ
 - Ethernet
 - Scaling Challenges
 - Switches
 - Nodes
 - Challenges (buffer occupancy) Packet-sizes
 - Push & Pull
- Real LHC DAQ architectures (bandwidth vs complexity)
- Eventfilterfarms
- “Ultimate” DAQ
 - Trigger-free / Sampling, ILC, Clic
 - Almost there: CMB
 - Longterm: LHC upgrade
 - Ethernet (again)? / InfiniBand

Disclaimer

- I have been working in this field since 11 years and admit readily to a biased view
- I have selected DAQ systems mostly to illustrate throughput / performance → not mentioned does not mean that it is not an interesting system
- “High throughput” brings a focus on large experiments: the greatest *heroism in DAQ* is found in small experiments, where the DAQ is done by one or two people part-time and in test-beams. I pay my respects to them!

Tycho Brahe and the Orbit of Mars

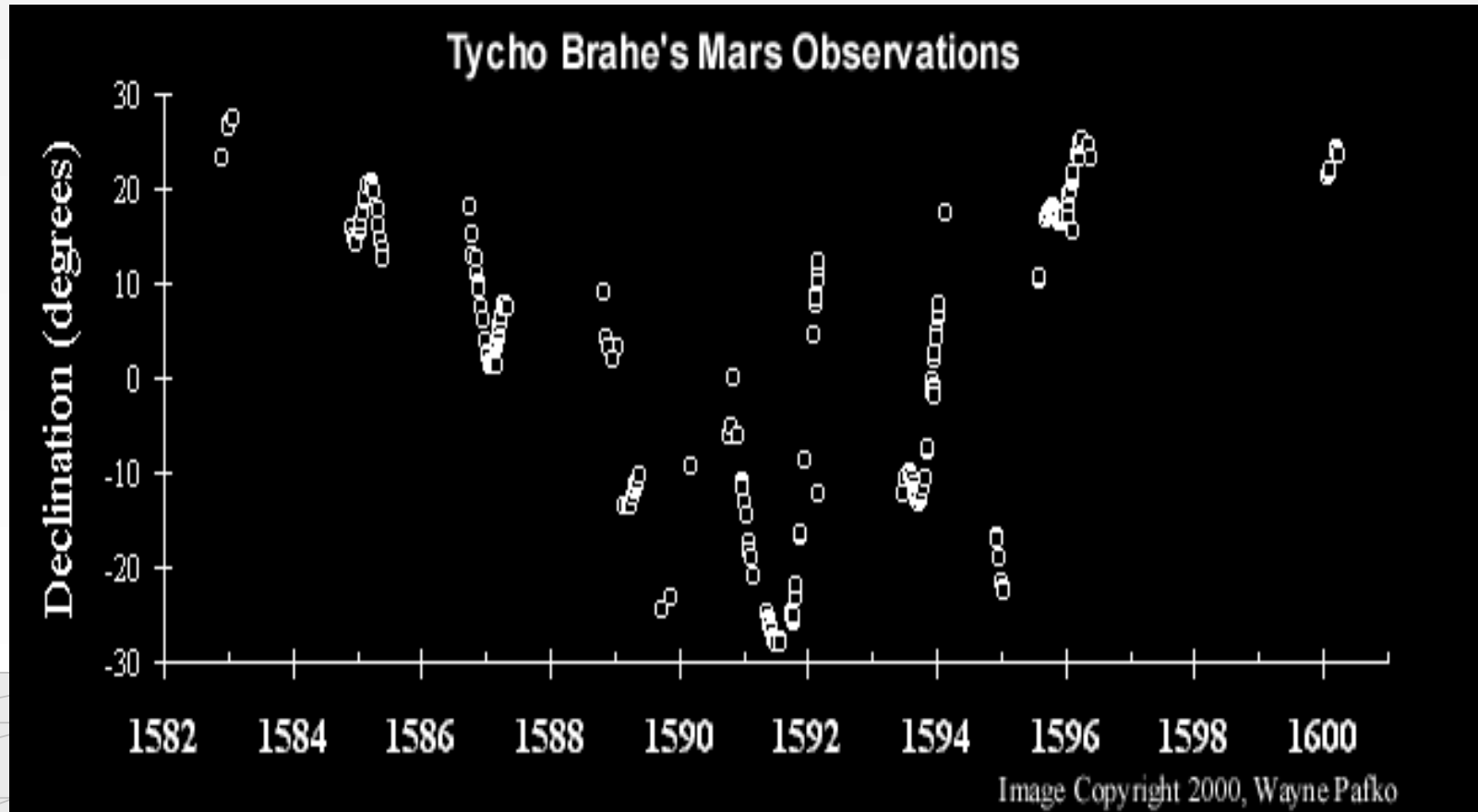
*I've studied all available charts of the planets and stars and none of them match the others. There are just as many measurements and methods as there are astronomers and all of them disagree. What's needed is **a long term project** with the aim of mapping the heavens **conducted from a single location** over a period of several years.*

Tycho Brahe, 1563 (age 17).



- First measurement campaign
- Systematic data acquisition
 - Controlled conditions (same location, same day and month)
 - Careful observation of boundary conditions (weather, light conditions etc...) - important for data quality / systematic uncertainties

The First Systematic Data Acquisition



- Data acquired over 18 years, normally every month
- Each measurement lasted at least 1 hr with the naked eye
- Red line (only in the animated version) shows comparison with modern theory

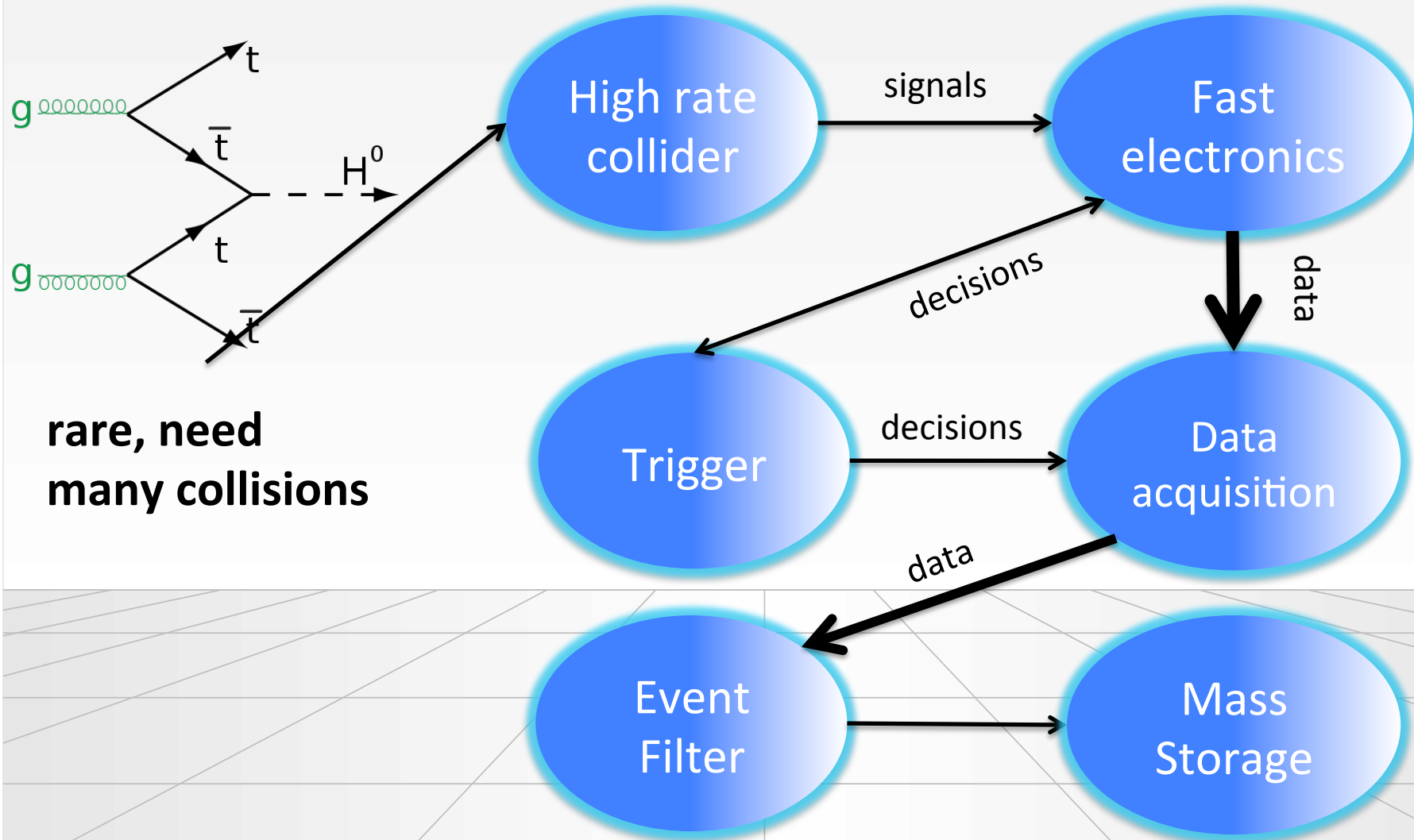
Tycho's DAQ in today's Terminology

- Trigger = in general something which tells you when is the “right” moment to take your data
 - In Tycho's case the position of the sun, respectively the moon was the trigger
 - the **trigger rate** $\sim 3.85 \times 10^{-6}$ Hz (one measurement / month) compare with LHCb 1.0×10^6 Hz
- Event-data (“event”) = the summary of all sensor data, which are recorded from an individual physical event
 - In Tycho's case the entry in his logbook (about 100 characters / entry)
 - In a modern detector the time a particle passed through a specific piece of the detector and the signal (charge, light) it left there
- Band-width (bw) (“throughput”) = Amount of data transferred / per unit of time
 - “Transferred” = written to his logbook
 - “unit of time” = duration of measurement
 - $bw_{\text{Tycho}} = \sim 100 \text{ Bytes / h} = 0.0003 \text{ kB/s}$
 - $bw_{\text{LHCb}} = 55.000 \text{ Bytes / us} = 55000000 \text{ kB/s}$

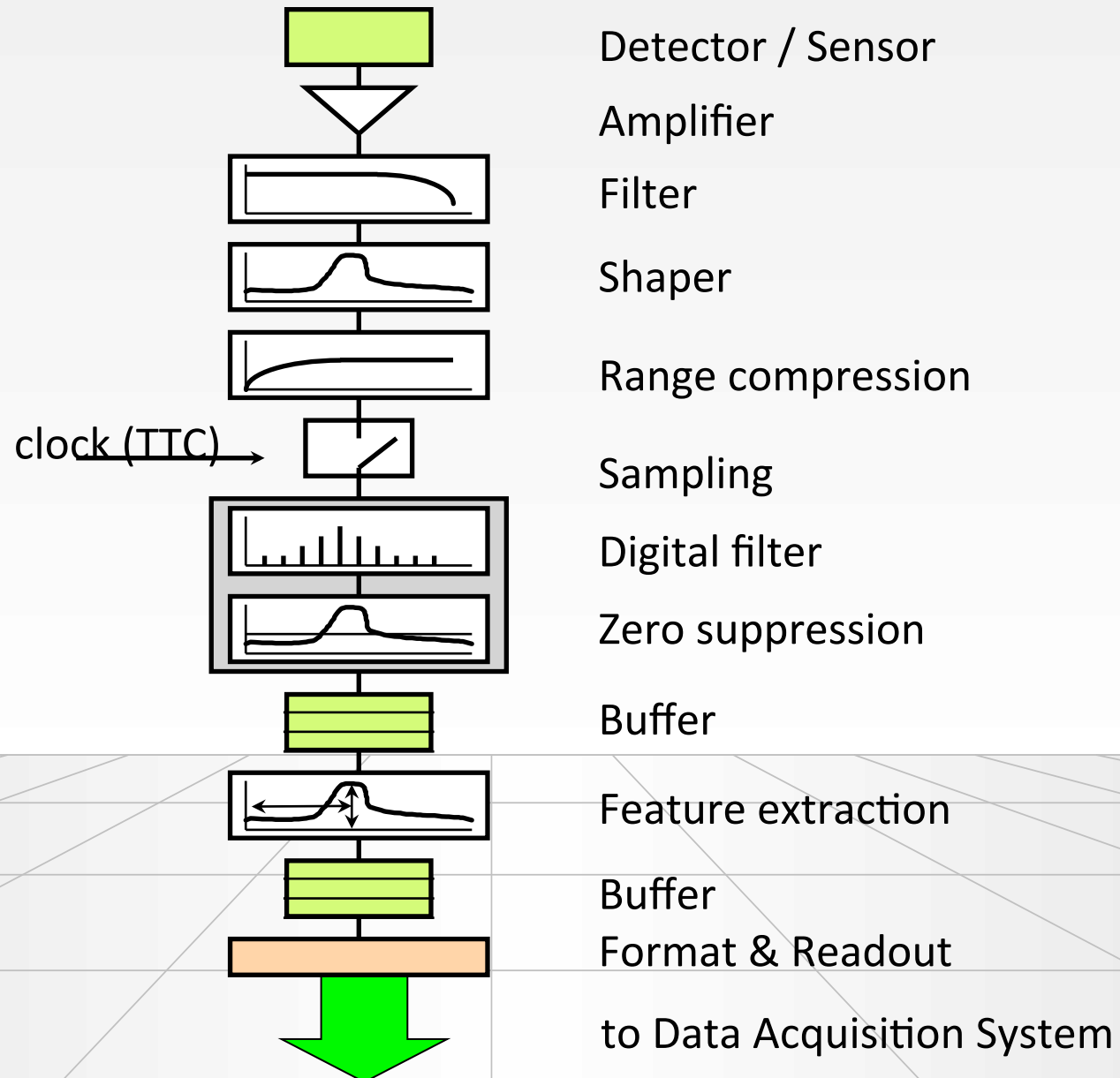
Lessons from Tycho

- Tycho did not do the correct analysis (he believed that the earth was at the center of the solar system) of the Mars data. This was done by Johannes Kepler (1571-1630), eventually paving the way for Newton's laws → good data will always be useful, even if you yourself don't understand them!
- The size & speed of a DAQ system are not correlated with the importance of the discovery!

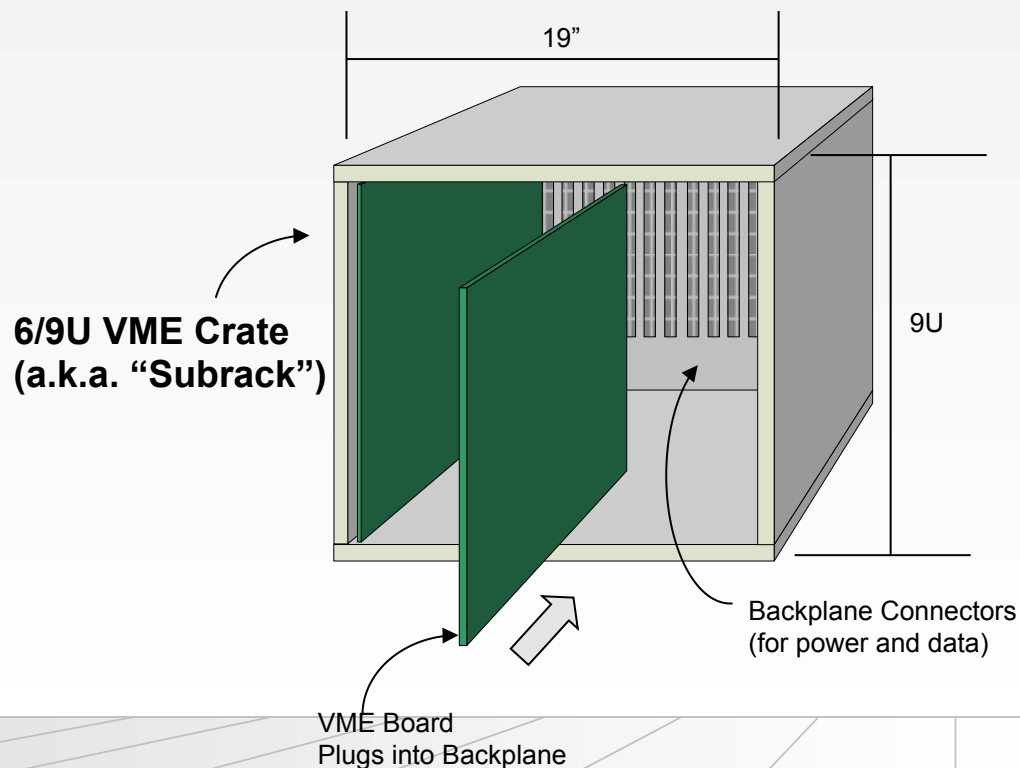
Physics, Detectors, Trigger & DAQ



Before the DAQ - a detector channel



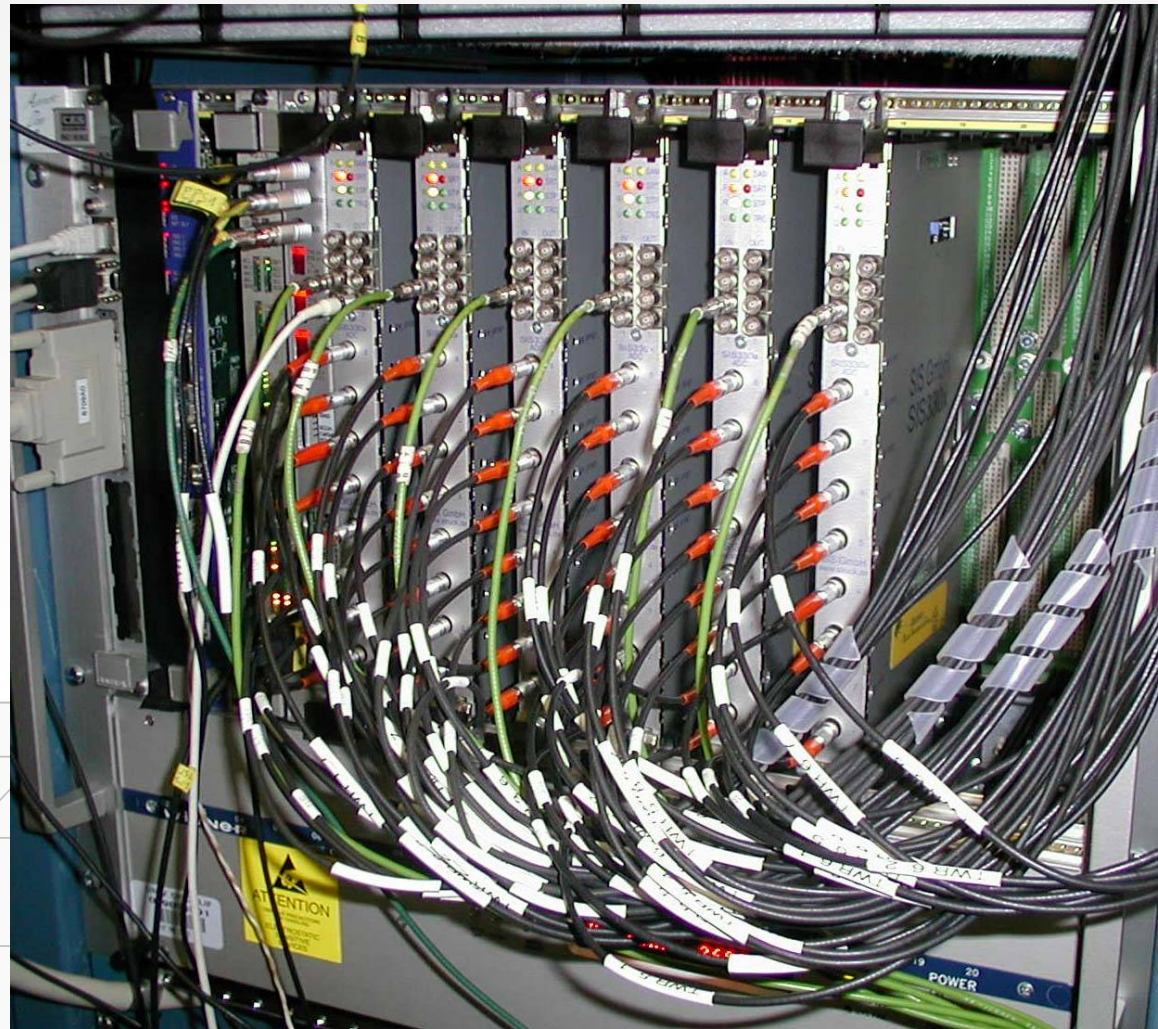
Crate-based DAQ



- Many detector channels are read-out on a dedicated PCB ("board")
- Many of these boards are put in a common chassis or **crate**
- These boards need
 - Mechanical support
 - Power
 - A standardized way to access their data (our measurement values)
- All this (and more 😊) is provided by standards for (readout) electronics such as **VME** (IEEE 1014), Fastbus, Camac , ATCA, uTCA

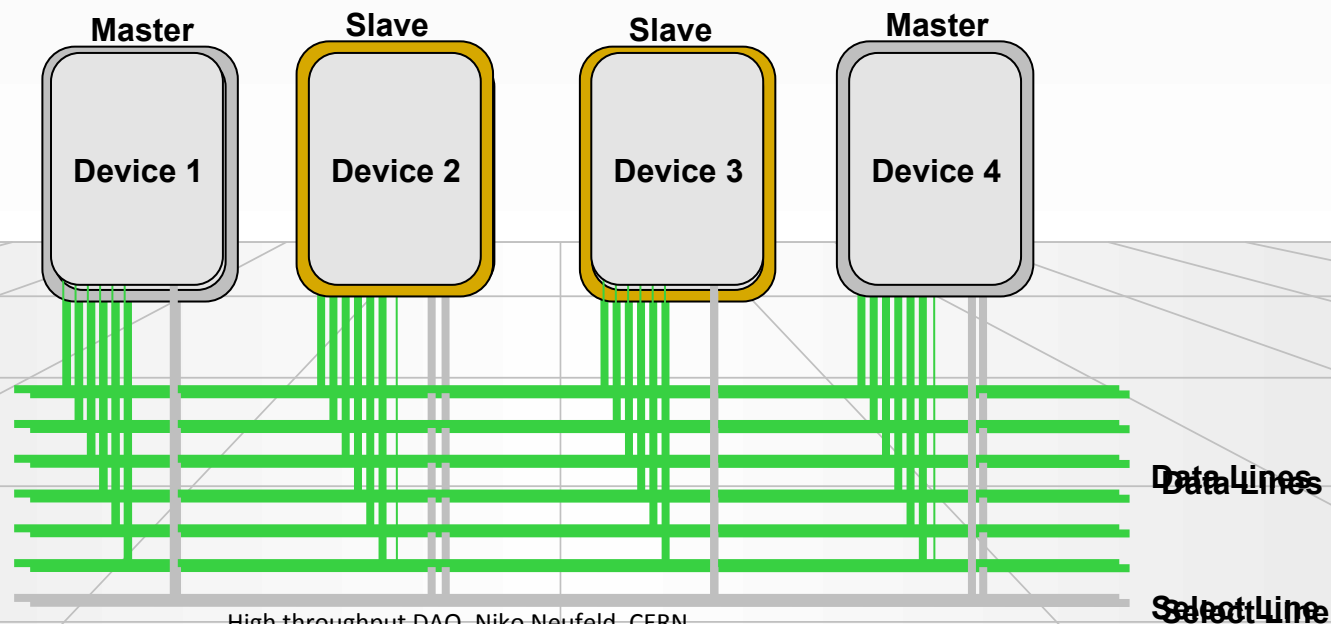
Example: VME

- **Readout boards** in a *VME-crate*
 - mechanical standard for
 - electrical standard for power on the backplane
 - signal and protocol standard for communication on a *bus*



Communication in a Crate: Buses

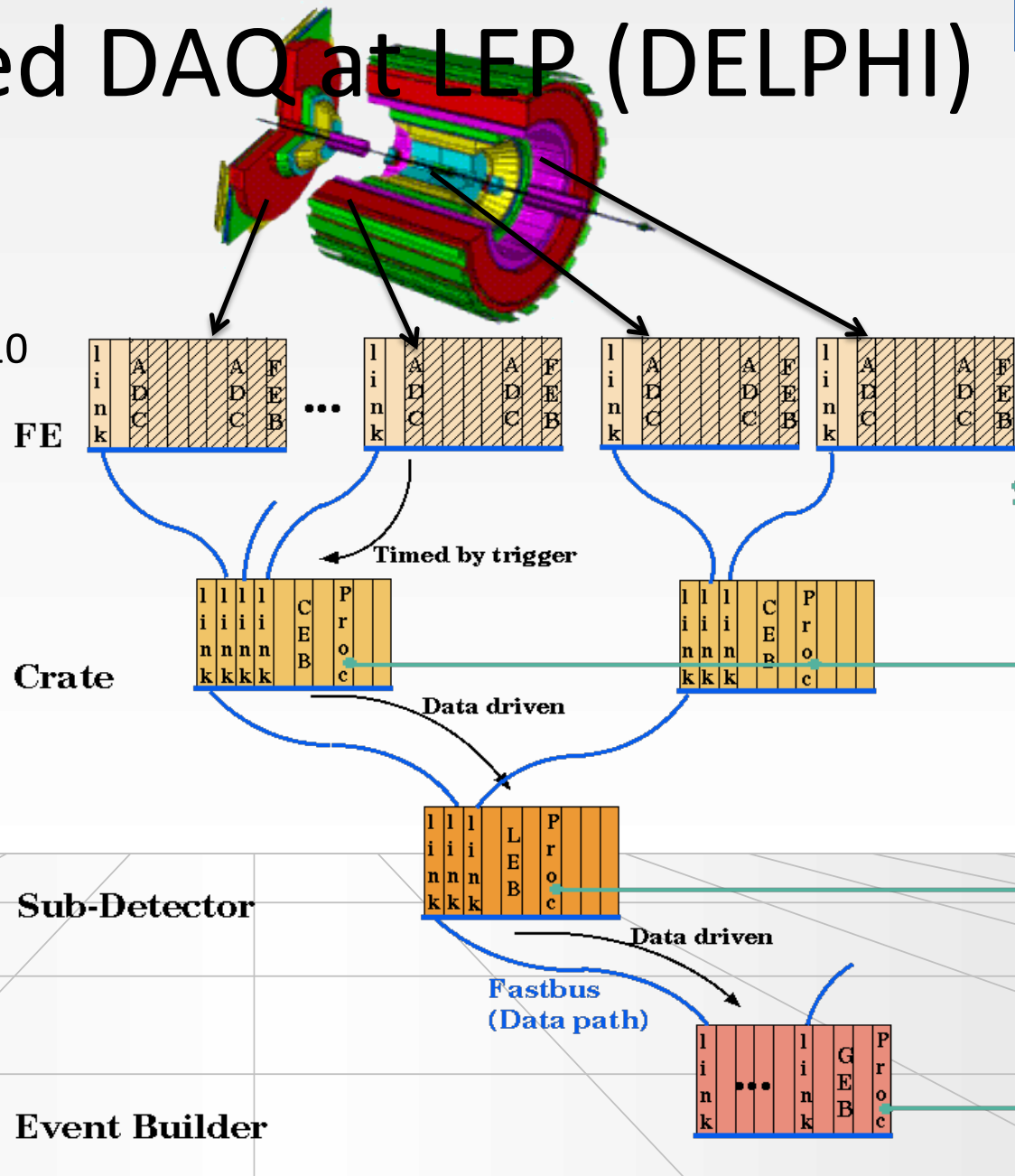
- A bus connects two or more devices and allows them to communicate
- The bus is **shared** between all devices on the bus → arbitration is required
- Devices can be **masters** or **slaves** and can be uniquely identified ("**addressed**") on the bus
- Number of devices and physical bus-length is limited (**scalability!**)
 - For synchronous high-speed buses, physical length is correlated with the number of devices (e.g. PCI)
 - Typical buses have a lot of control, data and address lines
- Buses are typically useful for systems $\ll 1$ GB/s



Crate-based DAQ at LEP (DELPHI)

- 200 Fastbus crates, 75 processors
- total event-size ~ 100 kB
- rate of accepted events $O(10 \text{ Hz})$

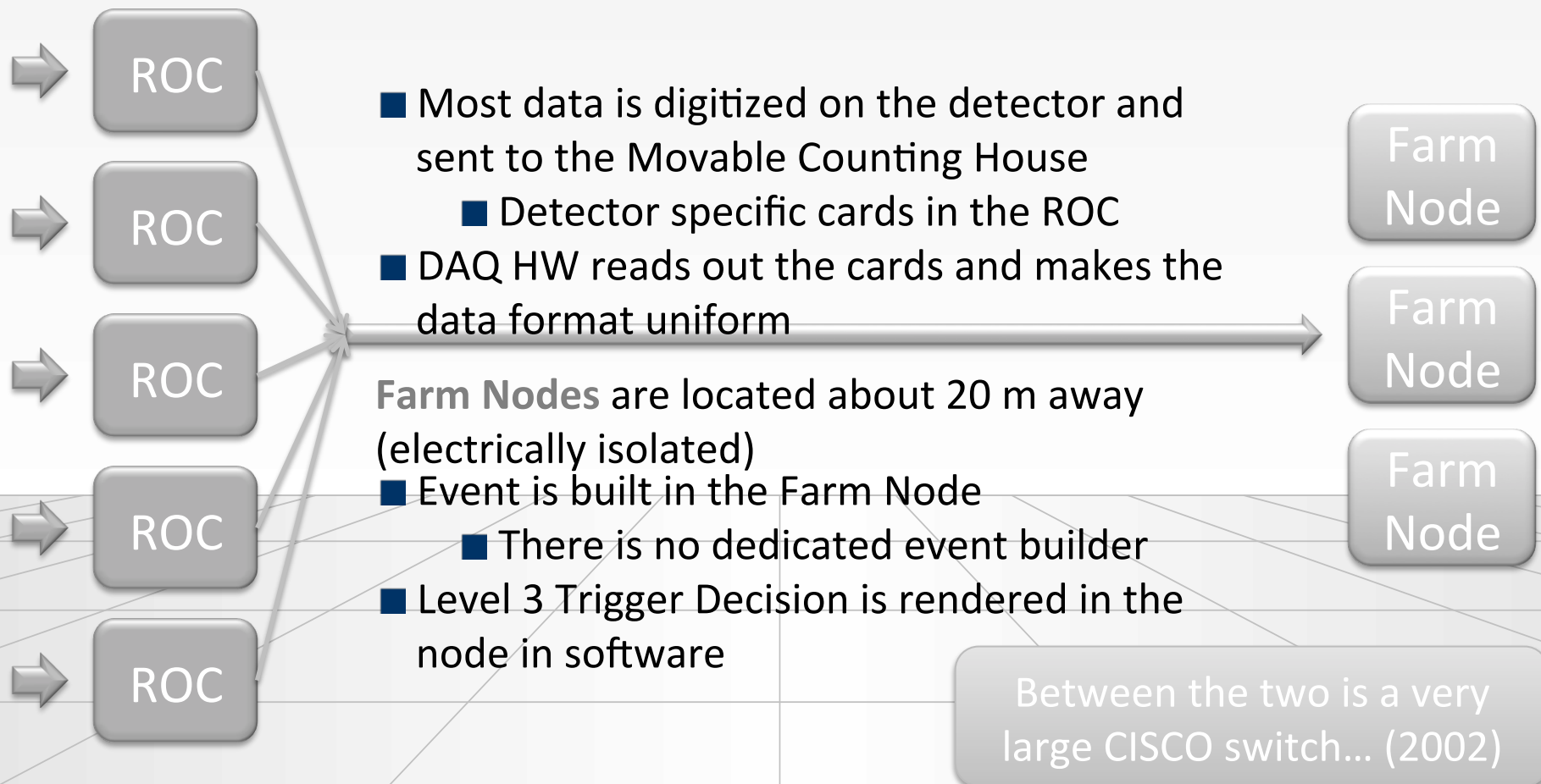
- **FE data**



- **Full Event**

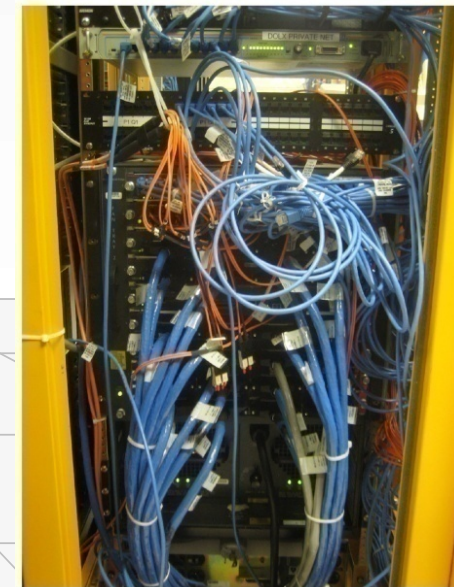
Combining crates and LANs: the D0 DAQ (L3)

Read Out Crates are VME crates that receive data from the detector. Event-size 300 kB, at ~ 1 kHz



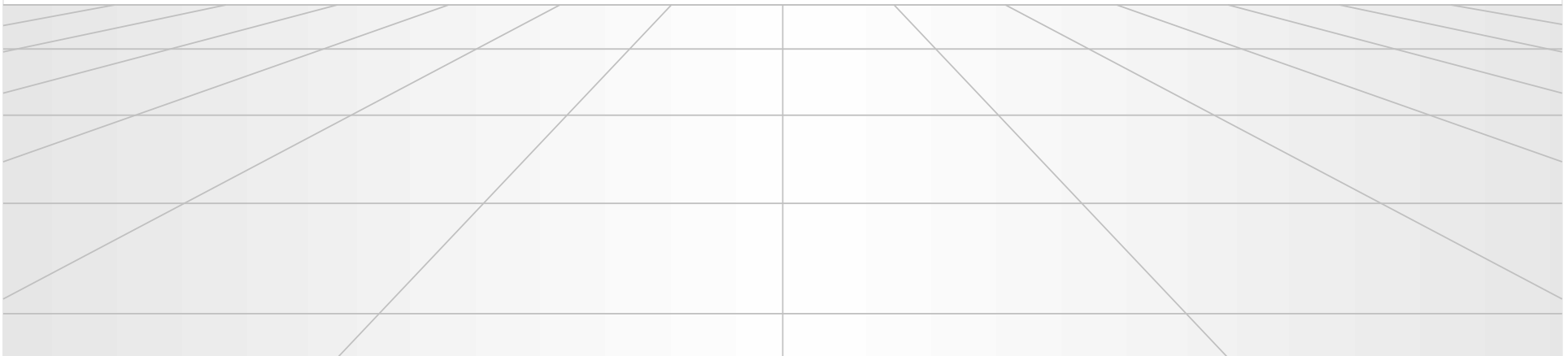
The DO DAQ in 2002 - 2011

- ROC's contain a Single Board Computer to control the readout.
 - VMIC 7750's, PIII, 933 MHz
 - 128 MB RAM
 - VME via a PCI Universe II chip
 - Dual 100 Mb ethernet
 - 4 have been upgraded to Gb ethernet due to increased data size
- Farm Nodes: 288 total, 2 and 4 cores per pizza box
 - AMD and Xeon's of differing classes and speeds
 - Single 100 Mb Ethernet
 - Less than last CHEP!
- CISCO 6590 switch
 - 16 Gb/s backplane
 - 9 module slots, all full
 - 8 port GB
 - 112 MB shared output buffer per 48 ports

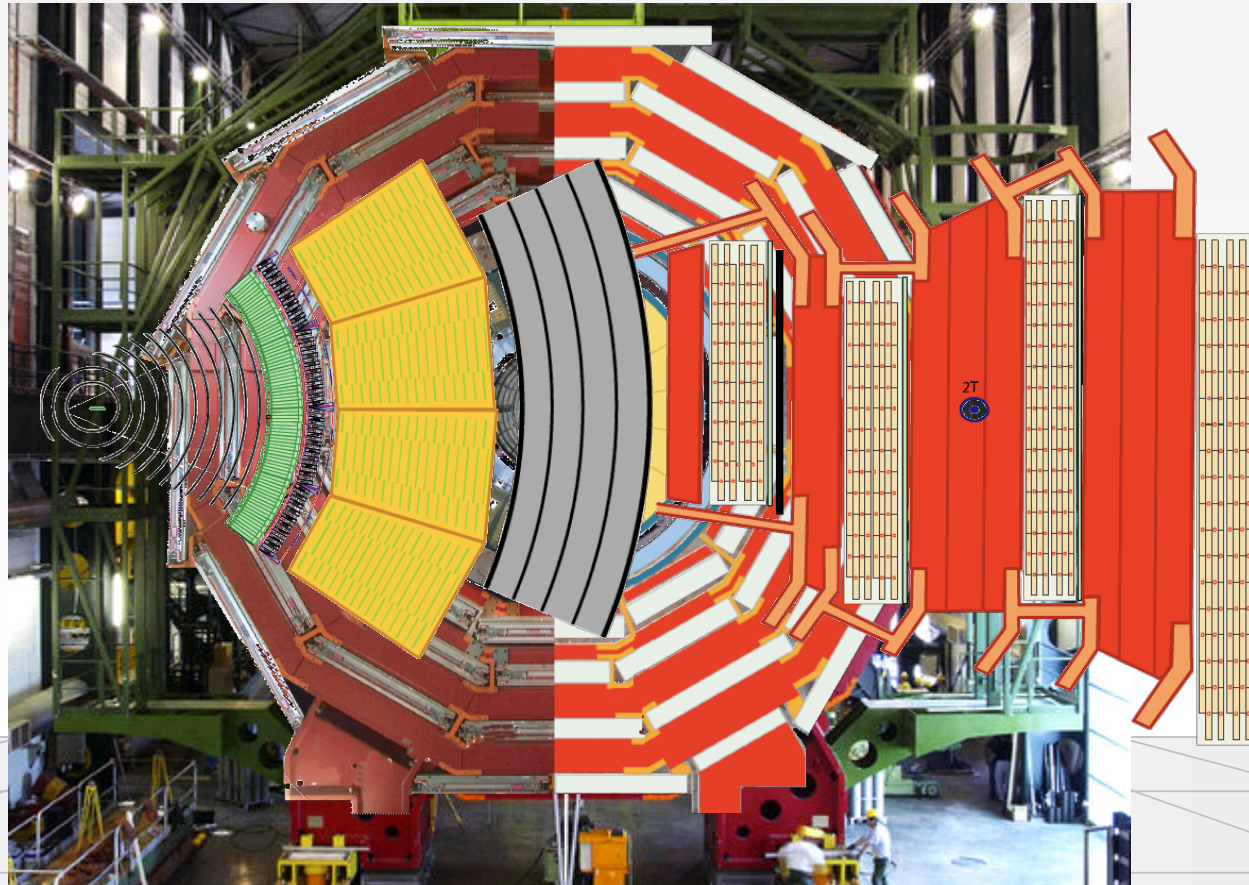


LHC DAQ

DAQ for multi-Gigabyte/s experiments



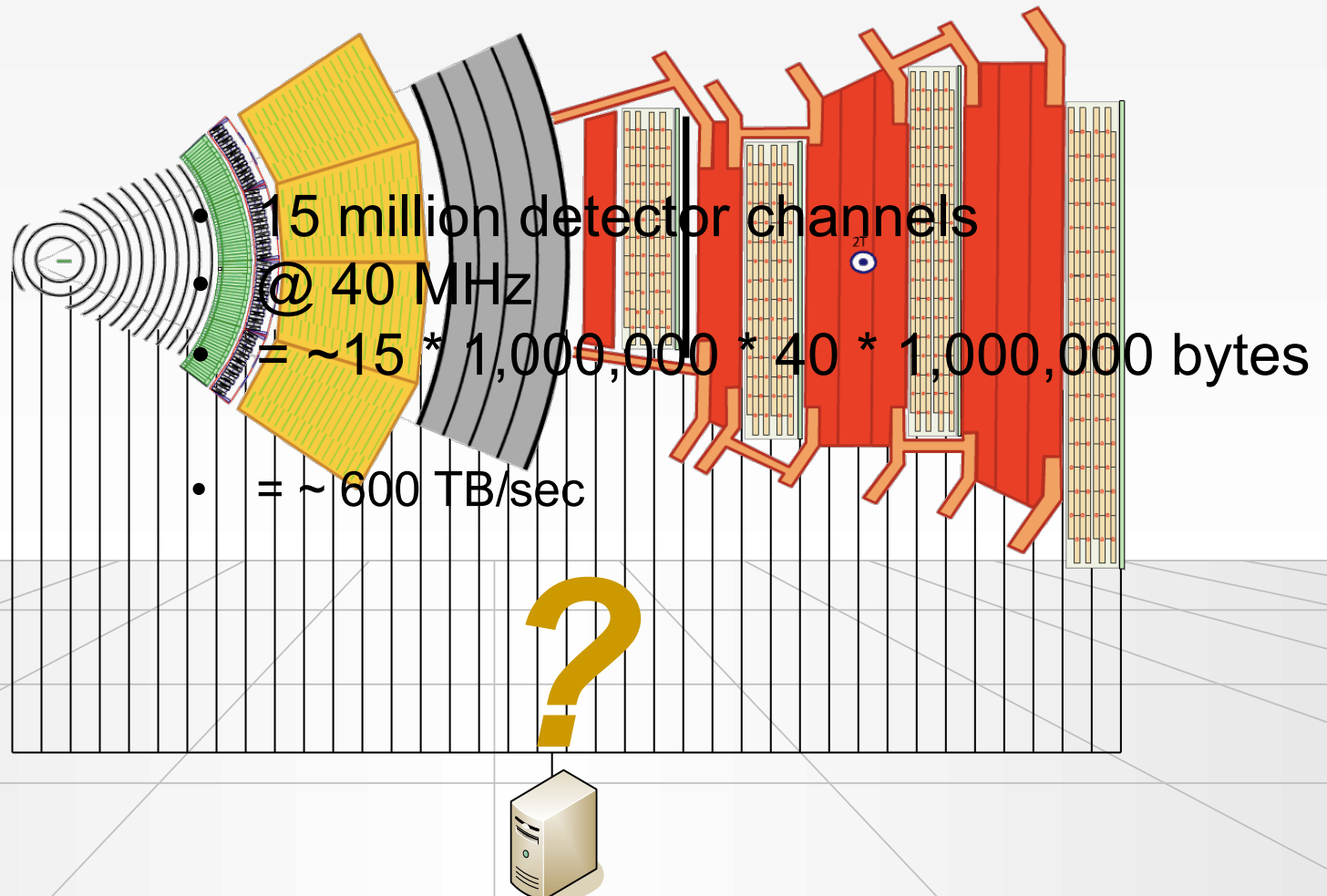
Moving on to Bigger Things...



The CMS Detector

High throughput DAQ, Niko Neufeld, CERN

Moving on to Bigger Things...



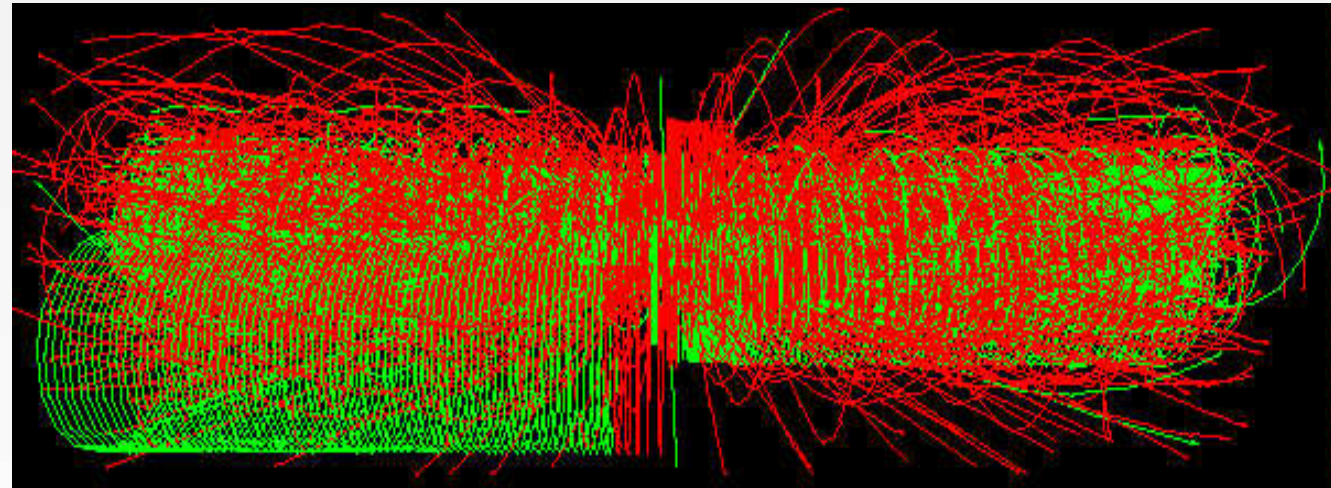
Know Your Enemy: pp Collisions at 14 TeV at 10^{34} $\text{cm}^{-2}\text{s}^{-1}$

- $\sigma(\text{pp}) = 70 \text{ mb}$
--> $>7 \times 10^8 / \text{s}$
(!)
- In ATLAS and CMS* 20 min bias events will overlap

• $\text{H} \rightarrow \text{ZZ}$

$\text{Z} \rightarrow \mu\mu$

$\text{H} \rightarrow 4 \text{ muons}$:
the cleanest
("golden")
signature

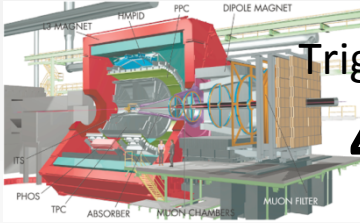
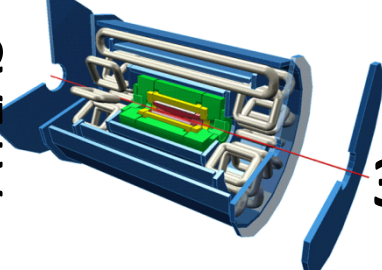
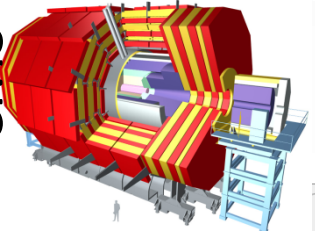
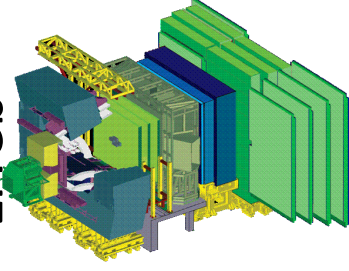


Reconstructed tracks
with $p_t > 25 \text{ GeV}$

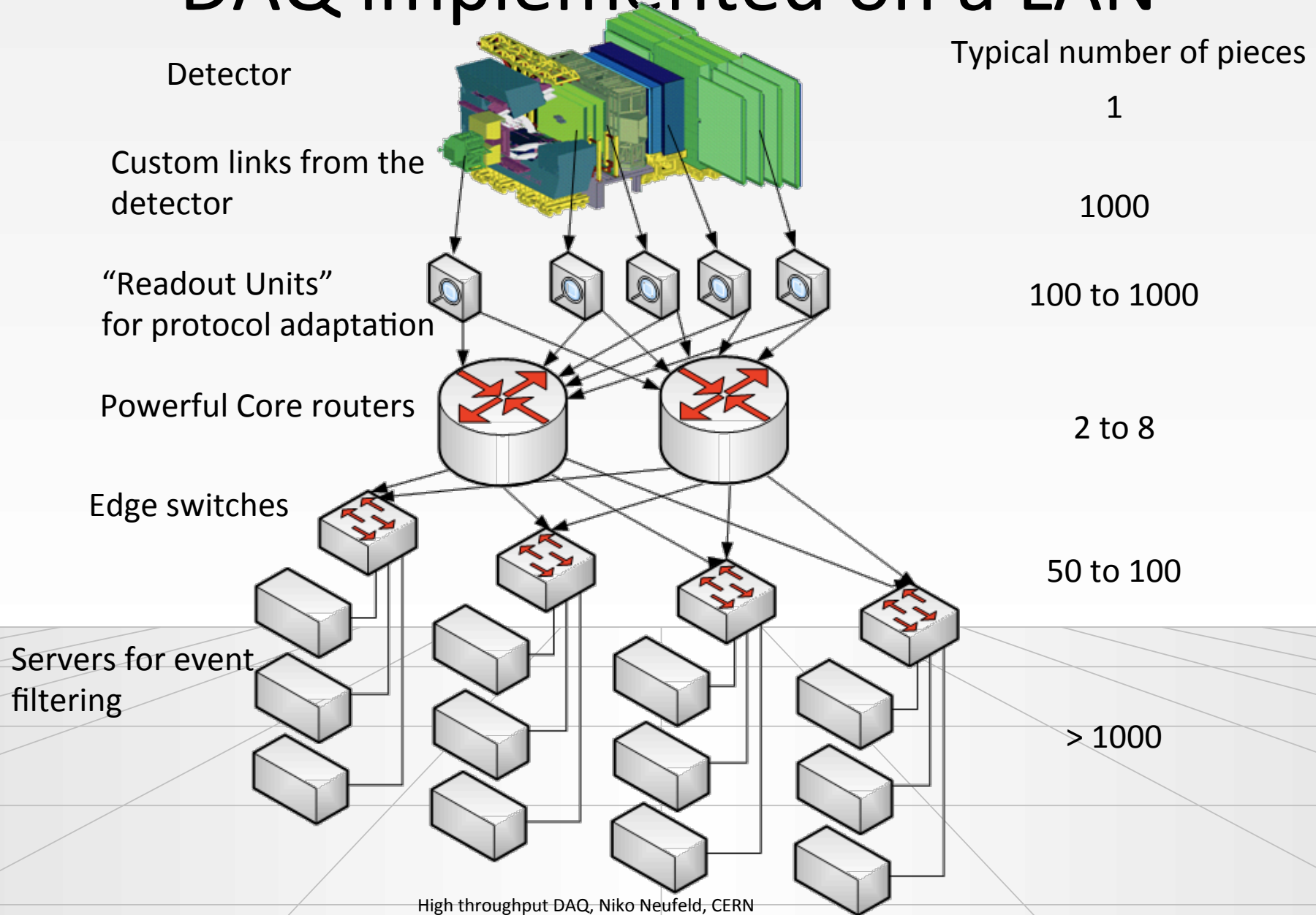
**And this
(not the H though...)
repeats every 25 ns...**

*)LHCb @ $2 \times 10^{33} \text{ cm}^{-2}\text{s}^{-1}$ isn't much nicer and in Alice (PbPb) it will be even worse

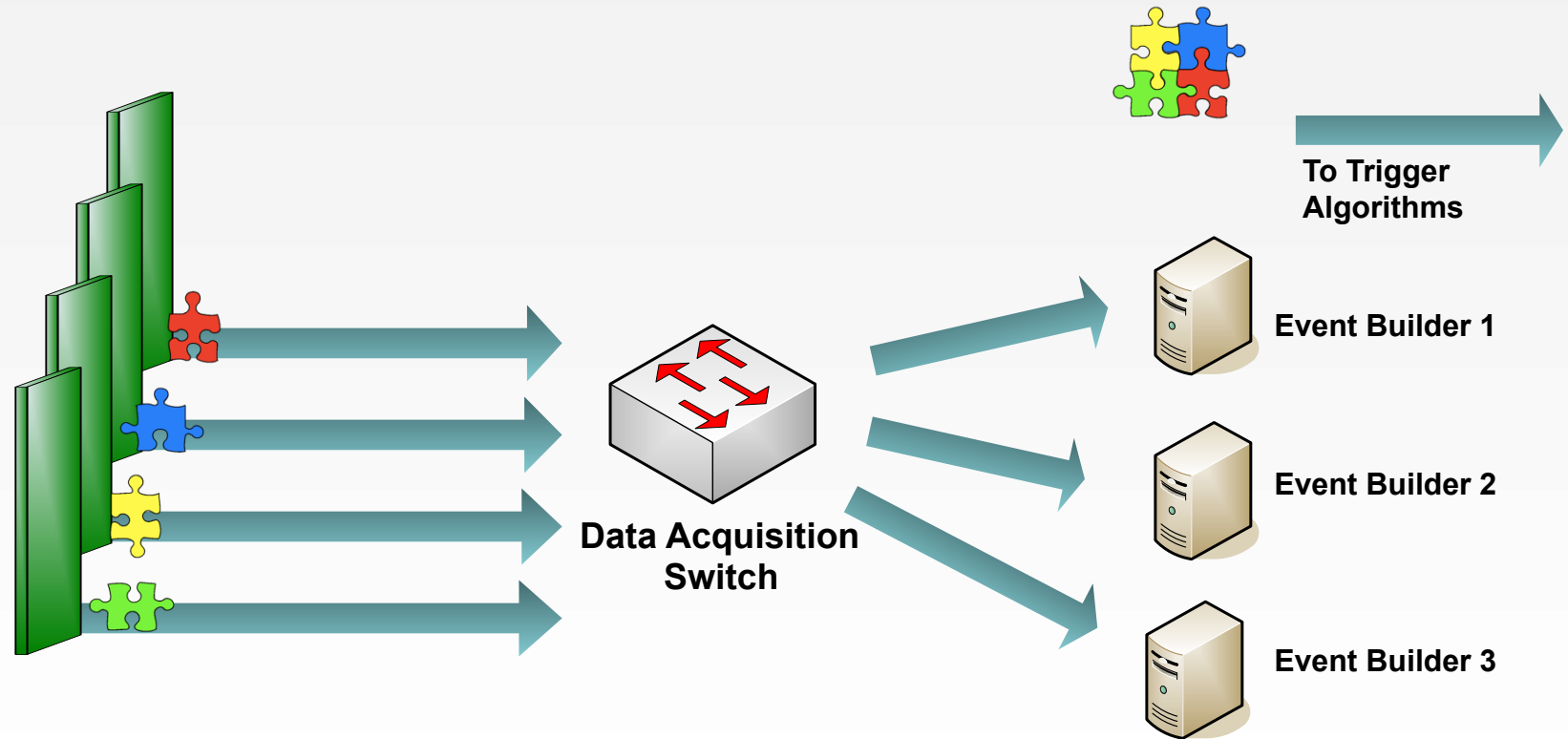
LHC Trigger/DAQ parameters

	#	Level-0,1,2	Event	Network	Storage
	Trigger	Rate (Hz)	Size (Byte)	Bandw.(GB/s)	MB/s (Event/s)
ALICE		Pb-Pb 500	5×10^7	25	1250 (10^2)
		p-p 10^3	2×10^6		200 (10^2)
ATLAS		LV-1 10^5 LV-2 3×10^3	1.5×10^6	4.5	300 (2×10^2)
CMS		LV-1 10^5	10^6	100	~ 1000 (10^2)
LHCb		LV-0 10^6	5.5×10^4	55	150 (2×10^3)

DAQ implemented on a LAN



Event Building over a LAN



1 Event fragments are received from detector front-end

2 Event fragments are read out over a network to an event builder

3 Event builder assembles fragments into a complete event

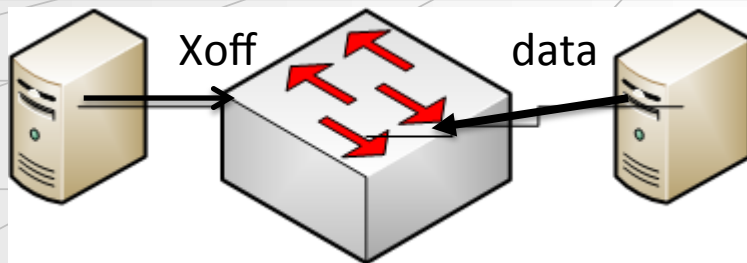
4 Complete events are processed by trigger algorithms

One network to rule the all

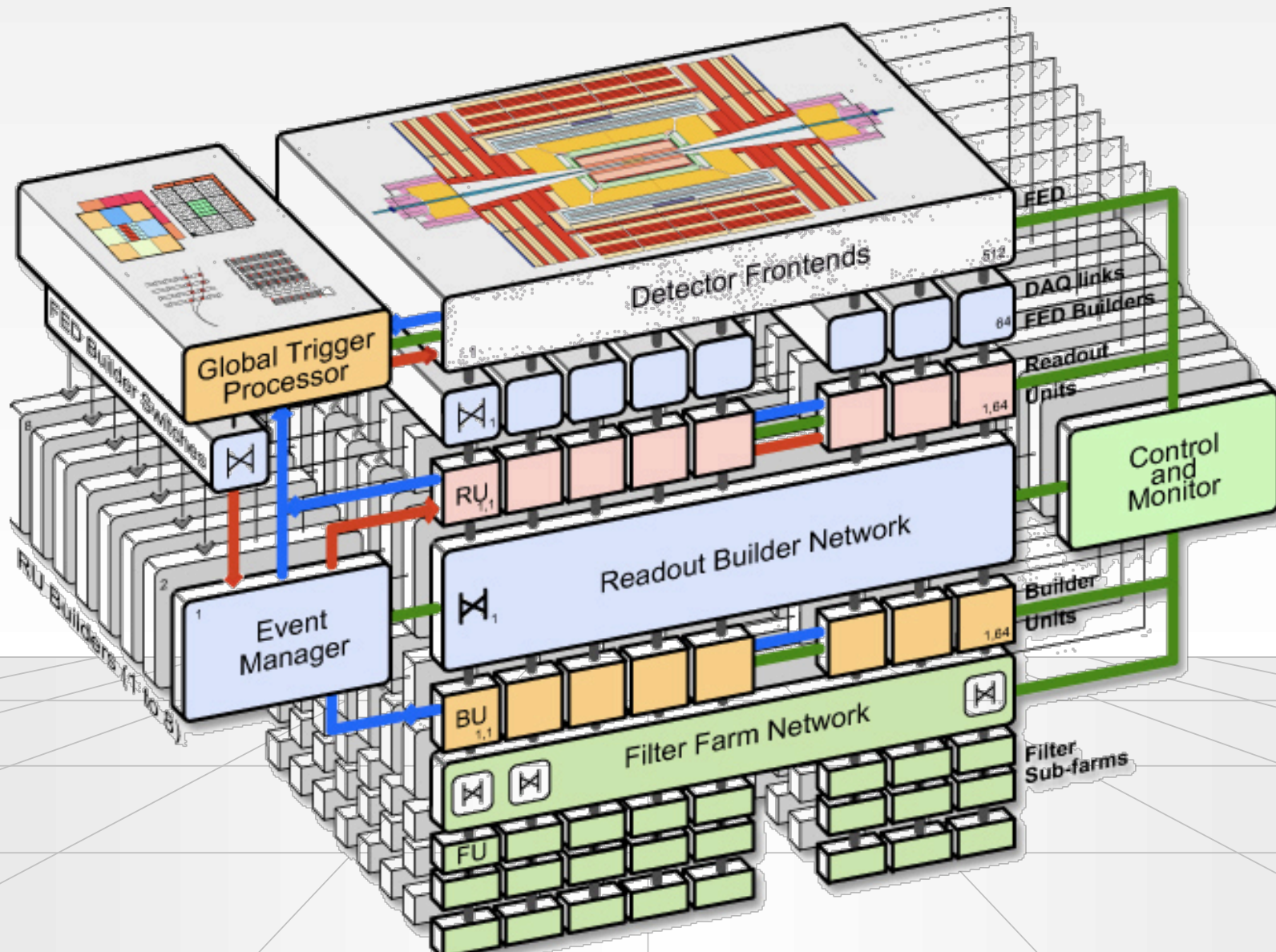
- Ethernet, IEEE 802.3xx, has almost become synonymous with Local Area Networking
- Ethernet has many nice features: cheap, simple, cheap, etc...
- Ethernet does not:
 - guarantee delivery of messages
 - allow multiple network paths
 - provide quality of service or bandwidth assignment (albeit to a varying degree this is provided by many switches)
- Because of this raw Ethernet is rarely used, usually it serves as a transport medium for IP, UDP, TCP etc...



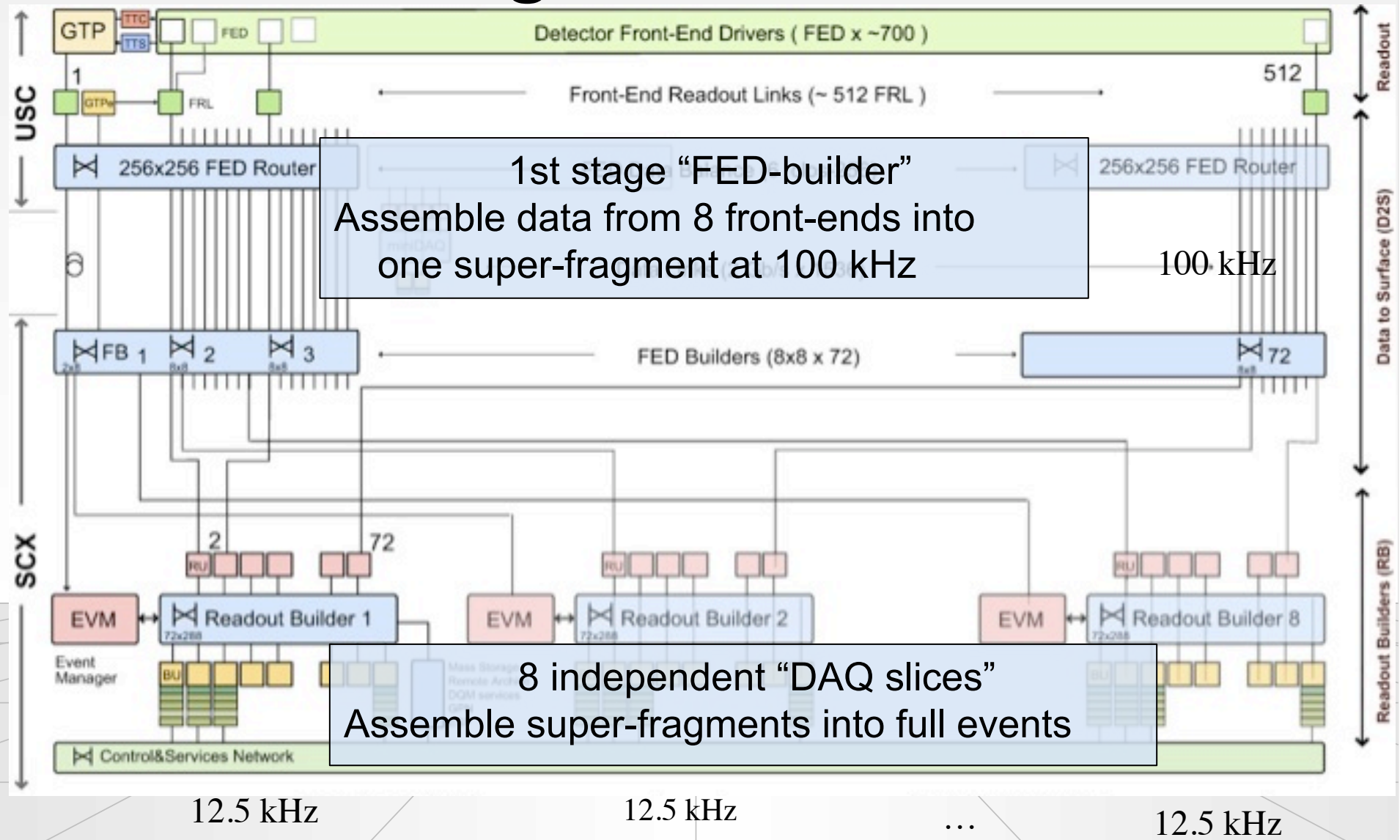
- Flow-control in standard Ethernet is only defined between immediate neighbors
- Sending station is free to throw away x-offed frames (and often does ☹)



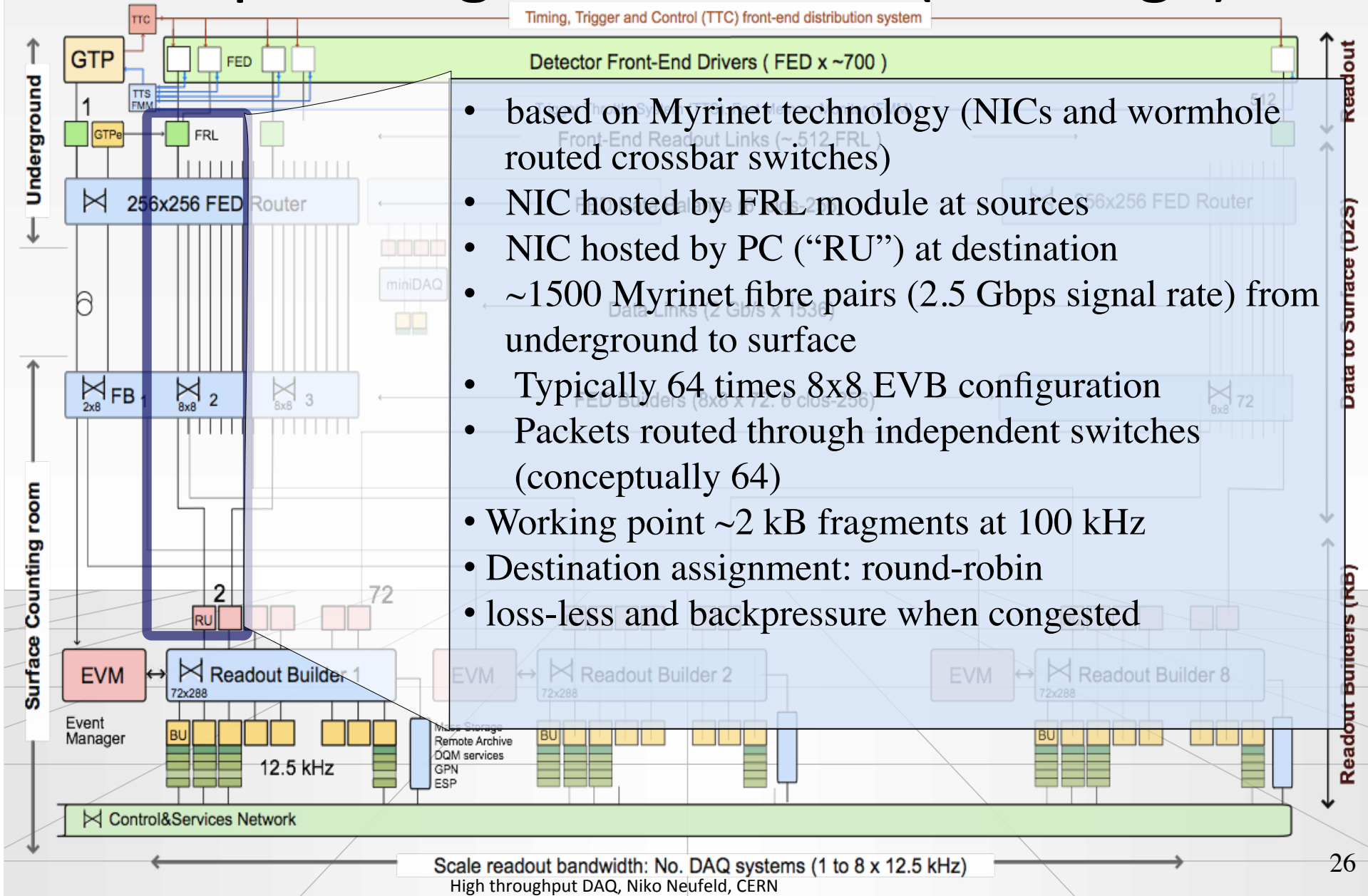
CMS Data Acquisition



2-Stage Event Builder

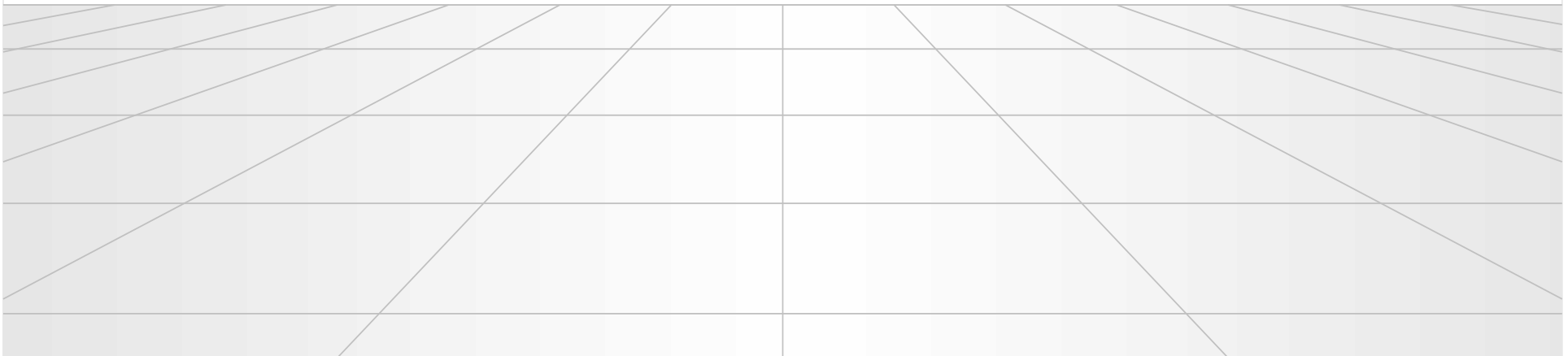


Super-Fragment Builder (1st stage)

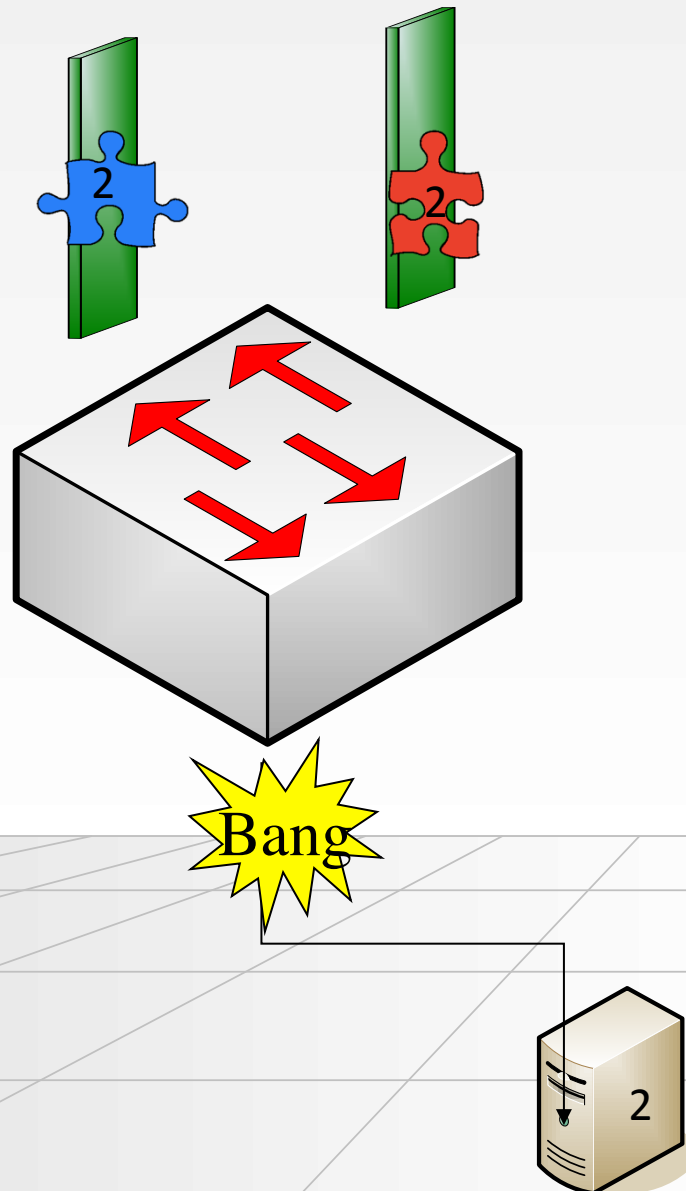


- based on Myrinet technology (NICs and wormhole routed crossbar switches)
- NIC hosted by FRL module at sources
- NIC hosted by PC (“RU”) at destination
- ~1500 Myrinet fibre pairs (2.5 Gbps signal rate) from underground to surface
- Typically 64 times 8x8 EVB configuration
- Packets routed through independent switches (conceptually 64)
- Working point ~2 kB fragments at 100 kHz
- Destination assignment: round-robin
- loss-less and backpressure when congested

Scaling in LAN based DAQ

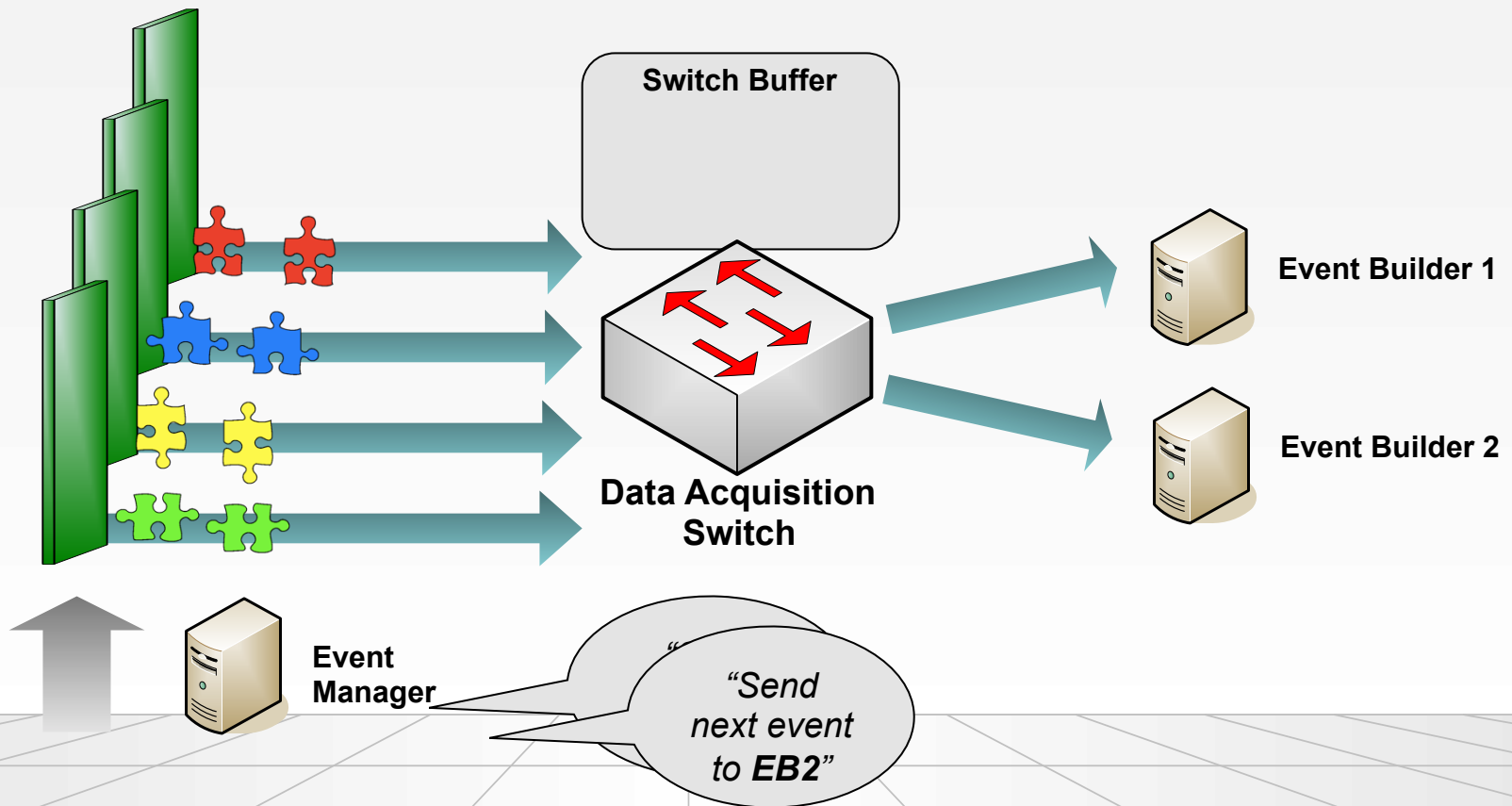


Congestion



- "Bang" translates into random, uncontrolled packet-loss
- In Ethernet this is perfectly valid behavior and implemented by many low-latency devices
- Higher Level protocols are supposed to handle the packet loss due to *lack of buffering*
- This problem comes from *synchronized sources sending to the same destination at the same time*

Push-Based Event Building



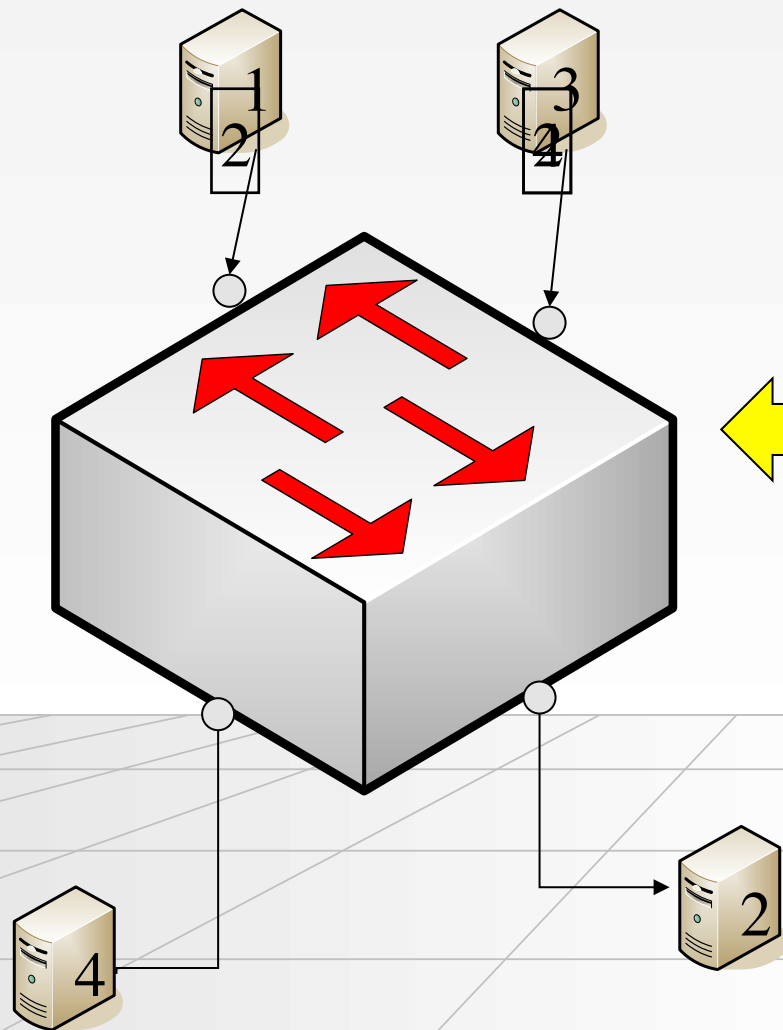
1 Event Manager tells readout boards where events must be sent (round-robin)

2 Readout boards do not buffer, so switch must

3 No feedback from Event Builders to Readout system

Cut-through switching

Head of Line Blocking



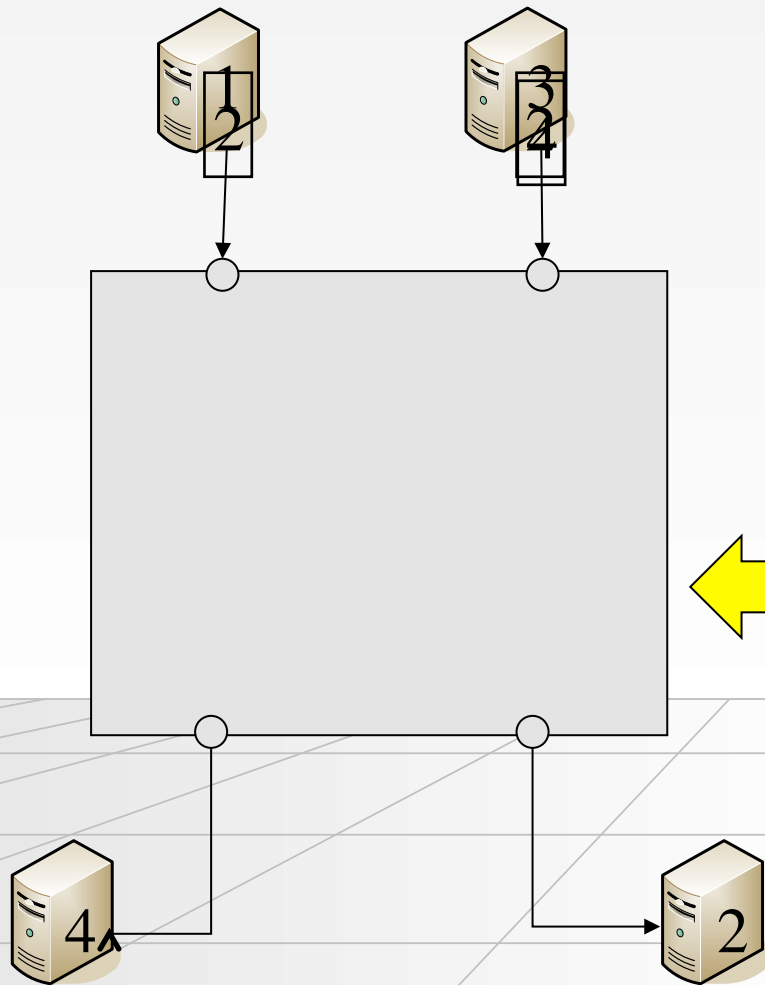
- The reason for this is the First in First Out (FIFO) structure of the input buffer
 - Queuing theory tells us* that for random traffic (and infinitely many switch ports) the throughput of the switch will go down to 58.6% → that means on 100 MBit/s network the nodes will "see" effectively only ~ 58 MBit/s
- Packet to node 4 must wait even though port to node 4 is free

*) "Input Versus Output Queuing on a Space-Division Packet Switch"; Karol, M. et al. ; IEEE Trans. Comm., 35/12

Using more of that bandwidth

- Cut-through switching is excellent for low-latency (no buffering) and reduces cost (no buffer memories), but “wastes” bandwidth
 - It’s like building more roads than required just so that everybody can go whenever they want immediately
- For optimal usage of installed bandwidth there are in general two strategies:
 - Use store-and-forward switching (next slide)
 - Use traffic-shaping / traffic-control
 - Different protocols (“pull-based event-building”), multi-level readout
 - end-to-end flow control
 - virtual circuits (with credit-scheme) (InfiniBand)
 - Barrel-shifter

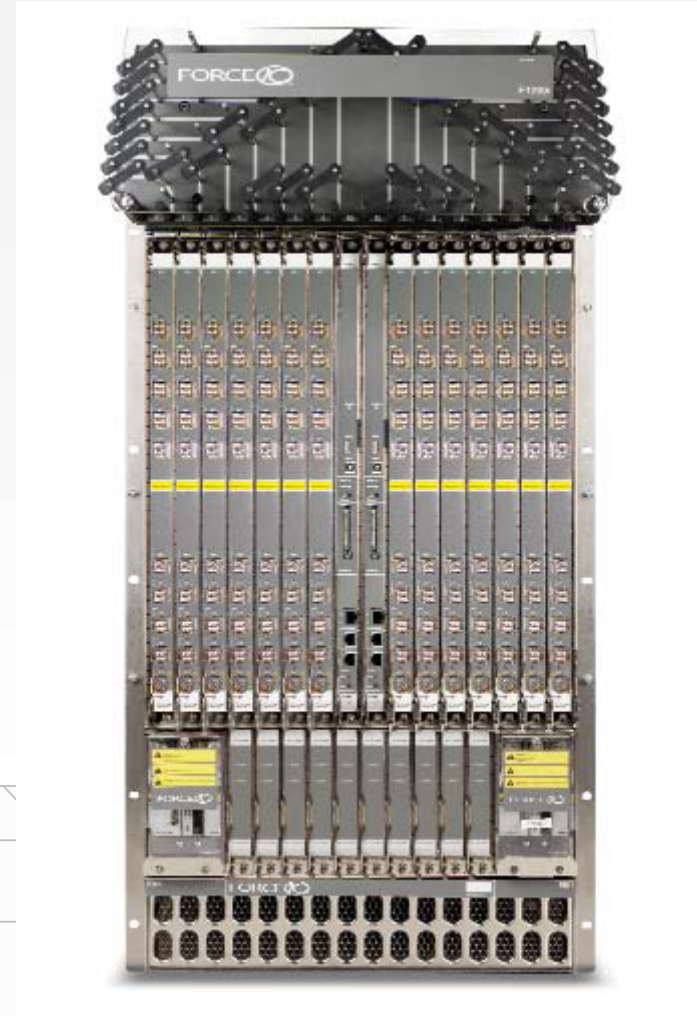
Output Queuing



- In practice virtual output queueing is used: at each input there is a queue → for n ports $O(n^2)$ queues must be managed
 - Assuming the buffers are large enough(!) such a switch will sustain random traffic at 100% nominal link load
- Packet to node 2 waits at output port 2. Way to node 4 is free

Store-and-Forward in the LHC DAQs

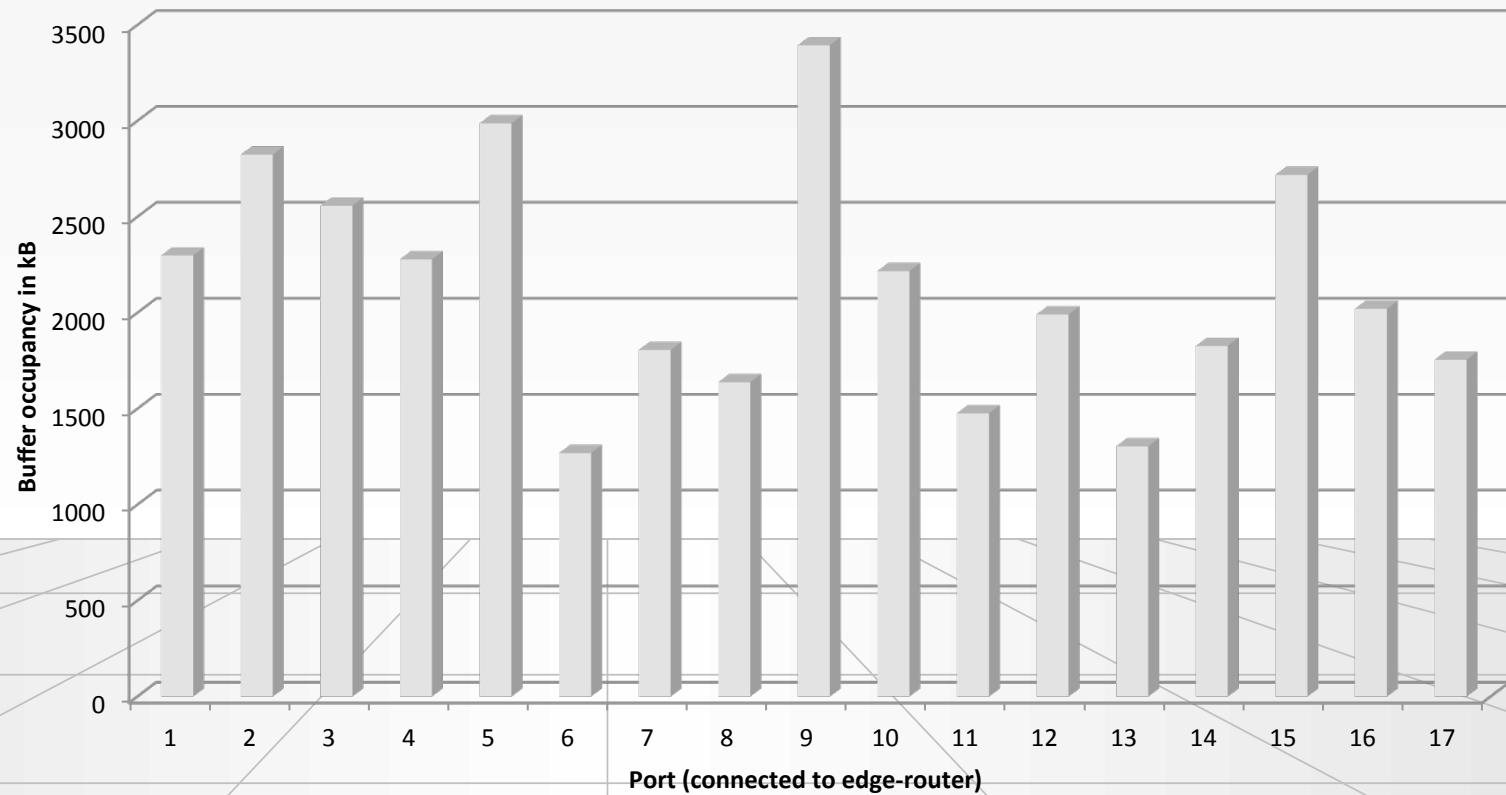
- **256 MB shared memory / 48 ports**
- Up to 1260 ports (1000 BaseT)
- Price / port ~ 500 - 1000 USD
- Used by all LHC experiments
- 6 kW power, 21 U high
- Loads of features (most of them unused 😊 in the experiments)



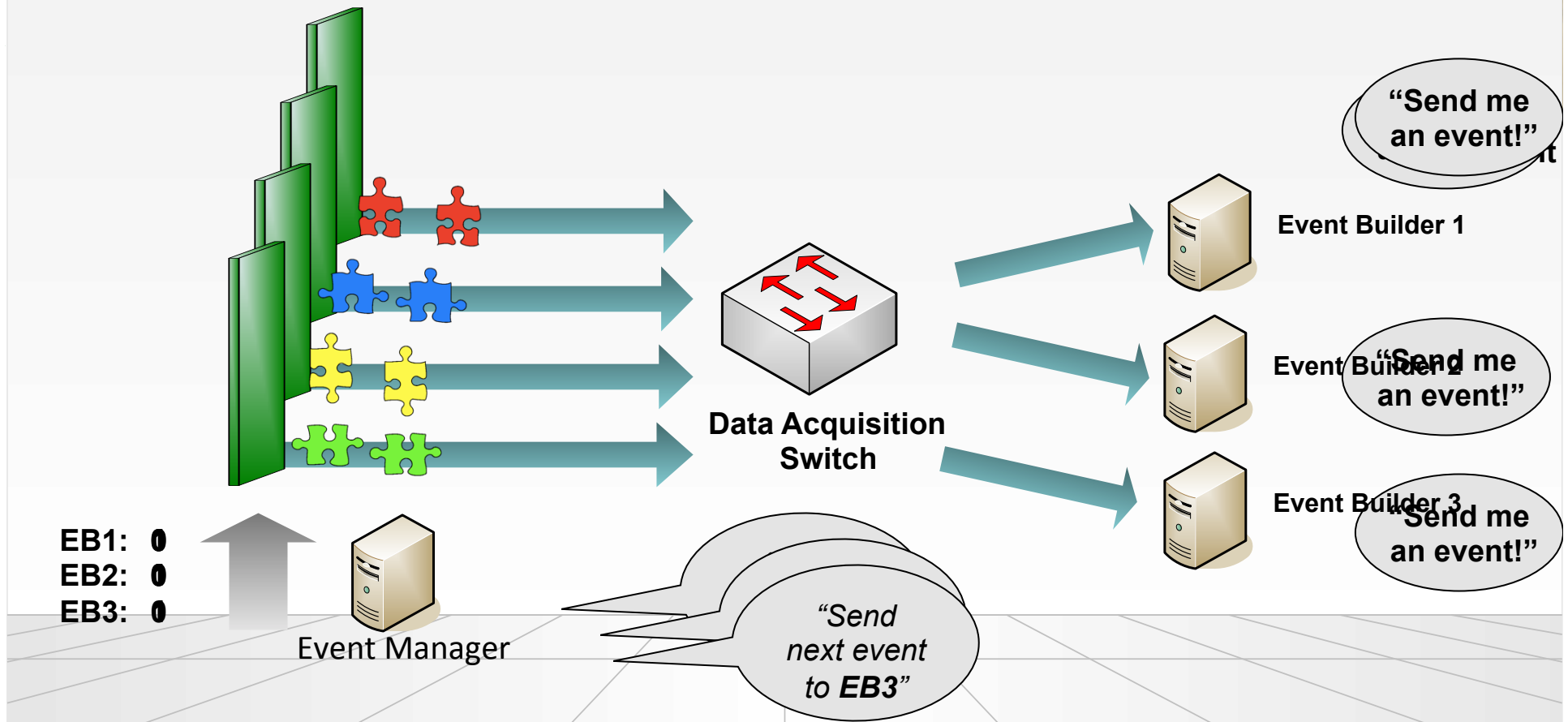
Buffer Usage in F10 E1200i

Buffer usage in core-router with a test using 270 sources @ 350 kHz event-rate

- 256 MB shared between 48 ports
- 17 ports used as “output”
- **This measurement for small LHCb events (50 kB) at 1/3 of nominal capacity**



Push-Based Event Building with store& forward switching and load-balancing



1

Event Builders notify Event Manager available capacity

2

Event Manager ensures that data are sent only to nodes with available capacity

3

Readout system relies on feedback from Event Builders

DAQ networks beyond push and store & forward

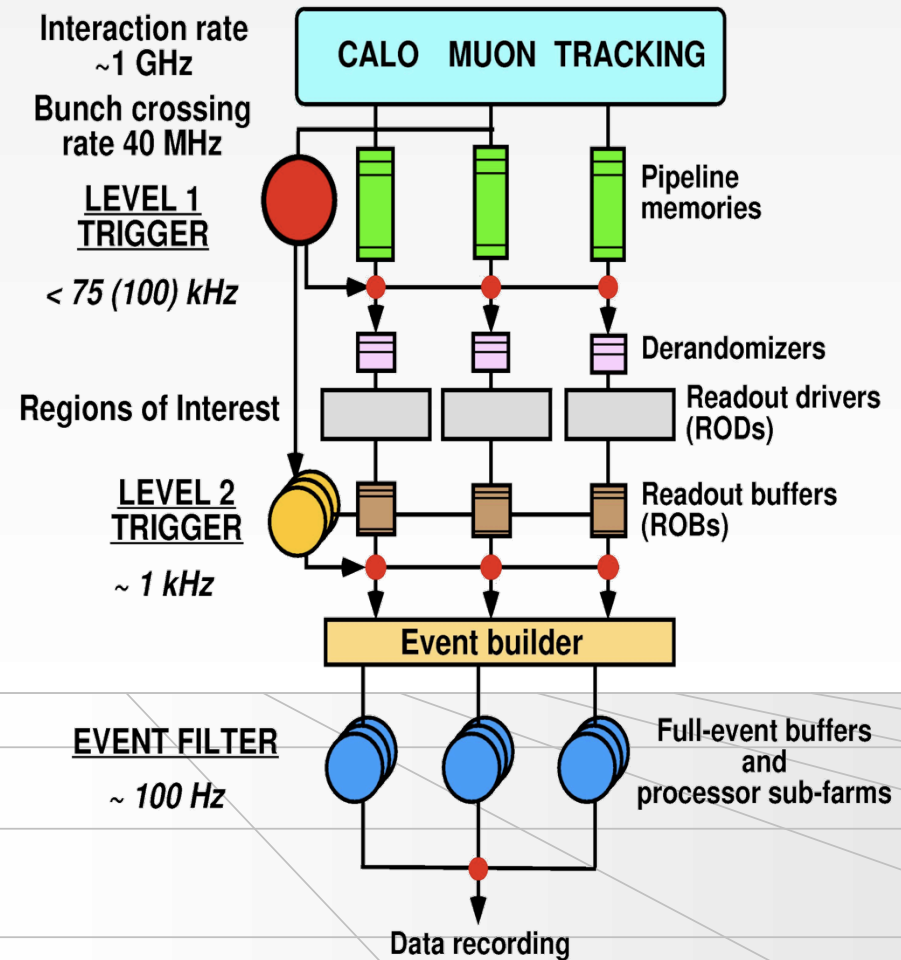
- Reducing network load
- Pull-based event-building
- Advanced flow-control



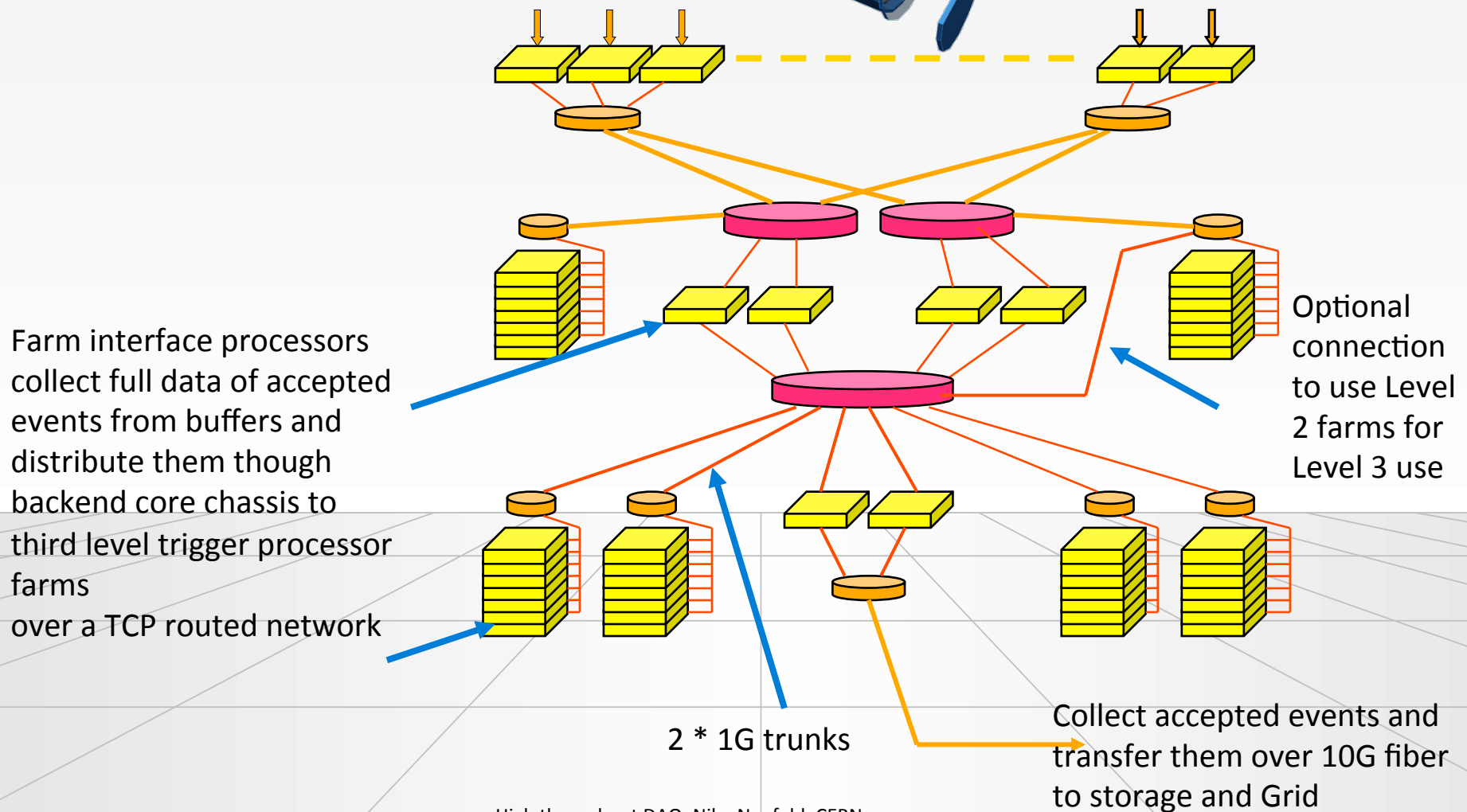
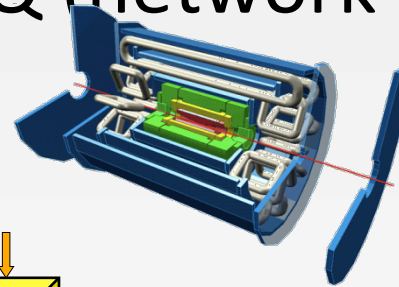
Progressive refinement: ATLAS



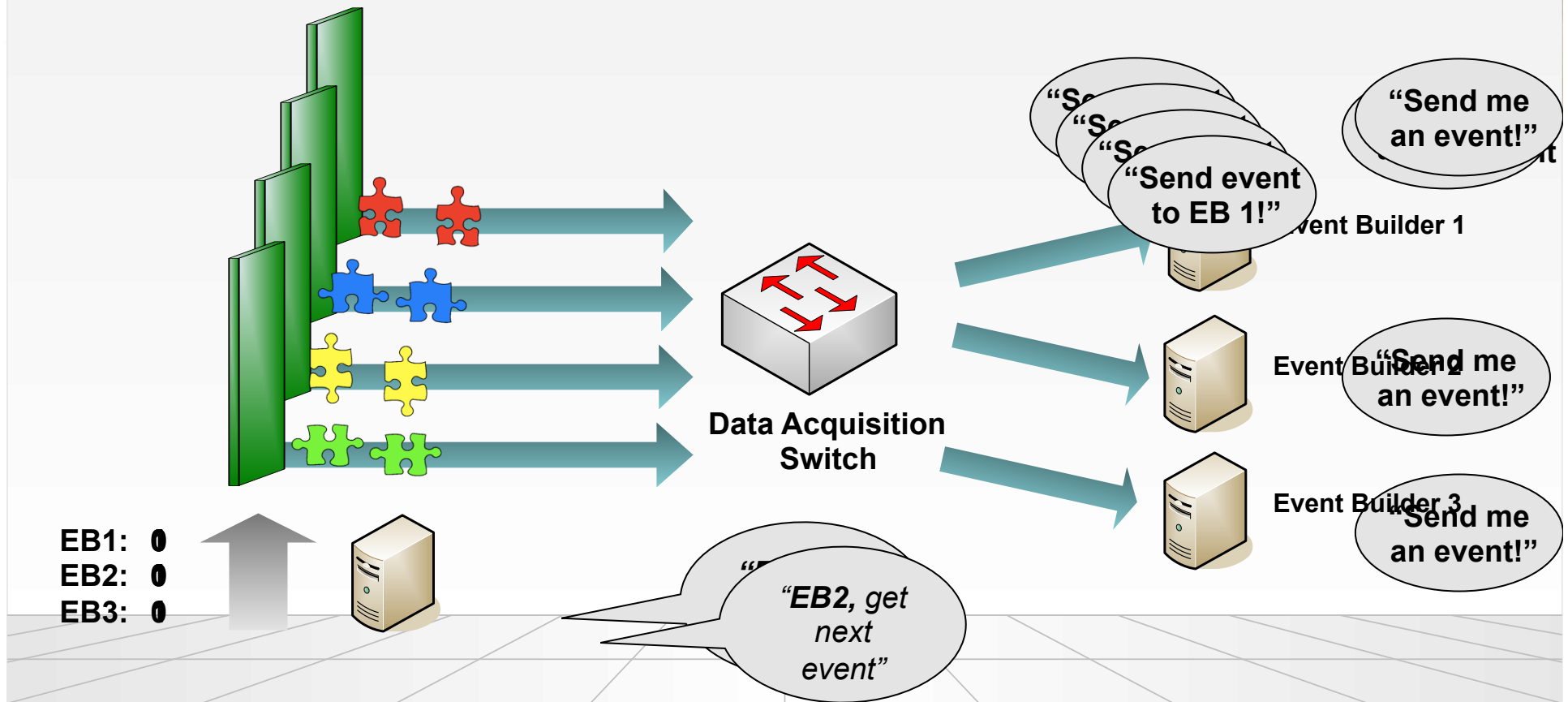
- 3-level trigger
- Partial read-out (possible because of the nature of ATLAS physics) at high rate
- Full read-out at relatively low rate
- → smaller network



ATLAS DAQ (network view)



Pull-Based Event Building



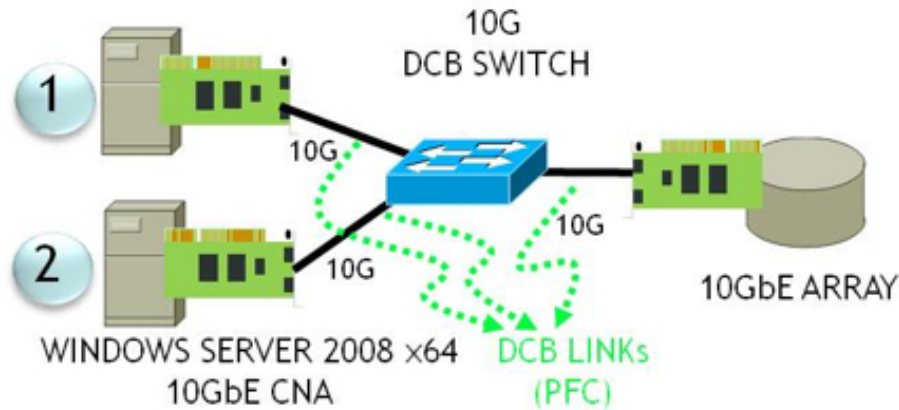
1 Event Builders notify Event Manager of available capacity

2 Event Manager elects event-builder node

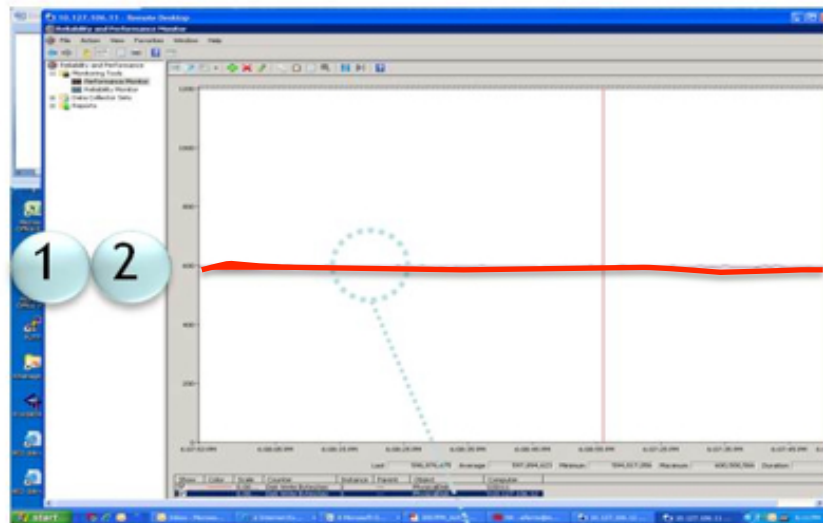
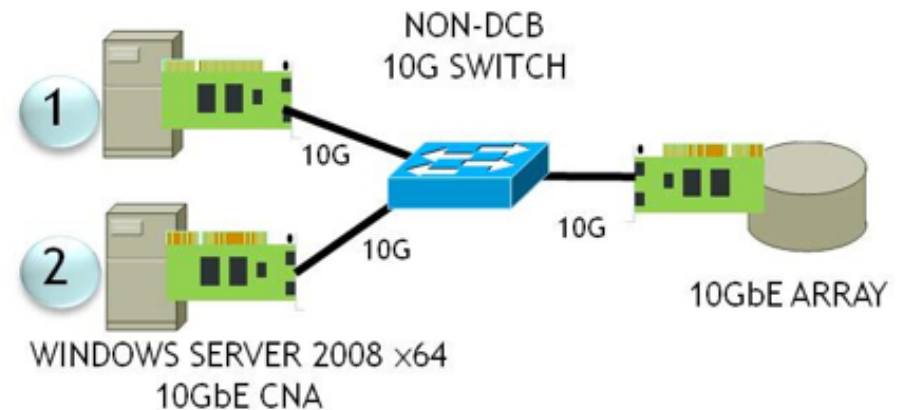
3 Readout traffic is driven by Event Builders

Advanced Flow-control DCB, QCN

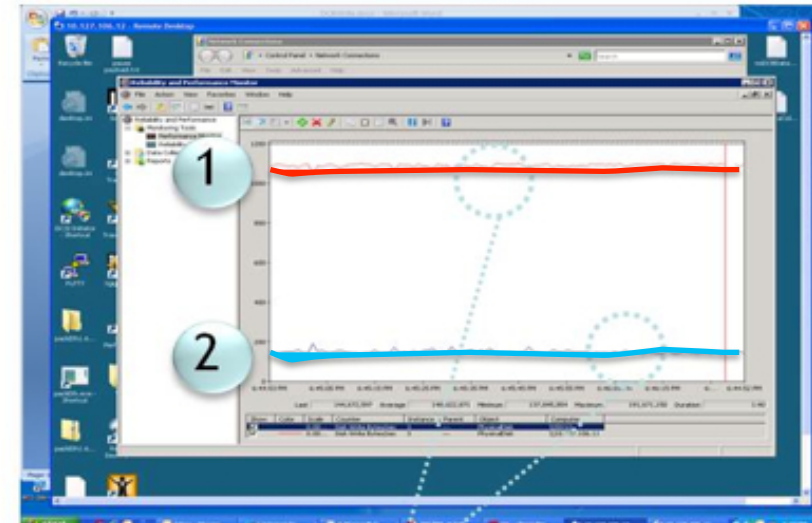
iSCSI WITH DCB



iSCSI WITHOUT DCB



Balanced iSCSI throughput (600MB/s, 600MB/s)
Steady packet streams (no TCP burstiness)

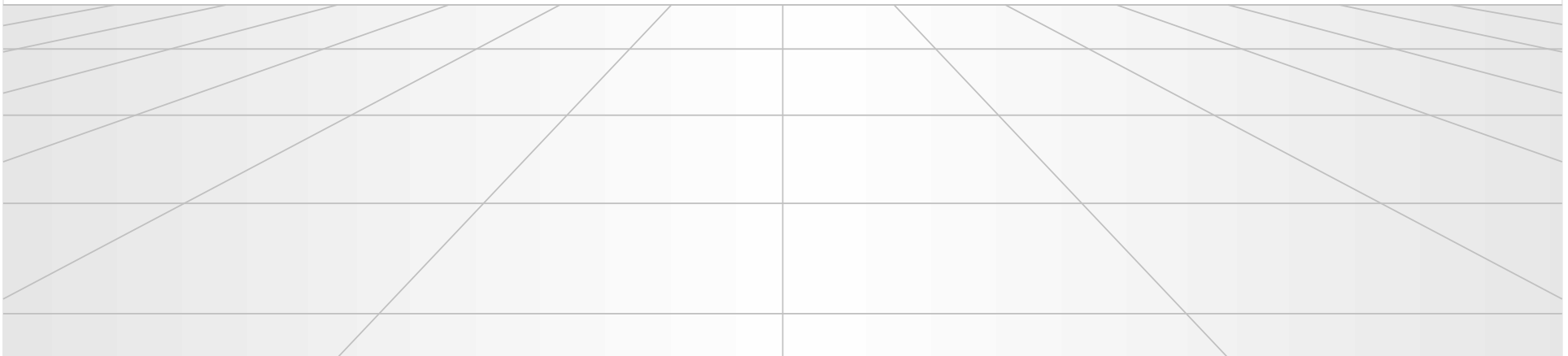


Unbalanced iSCSI throughput (1100MB/s, 100MB/s)
Typical TCP burstiness

Why is DCB interesting?

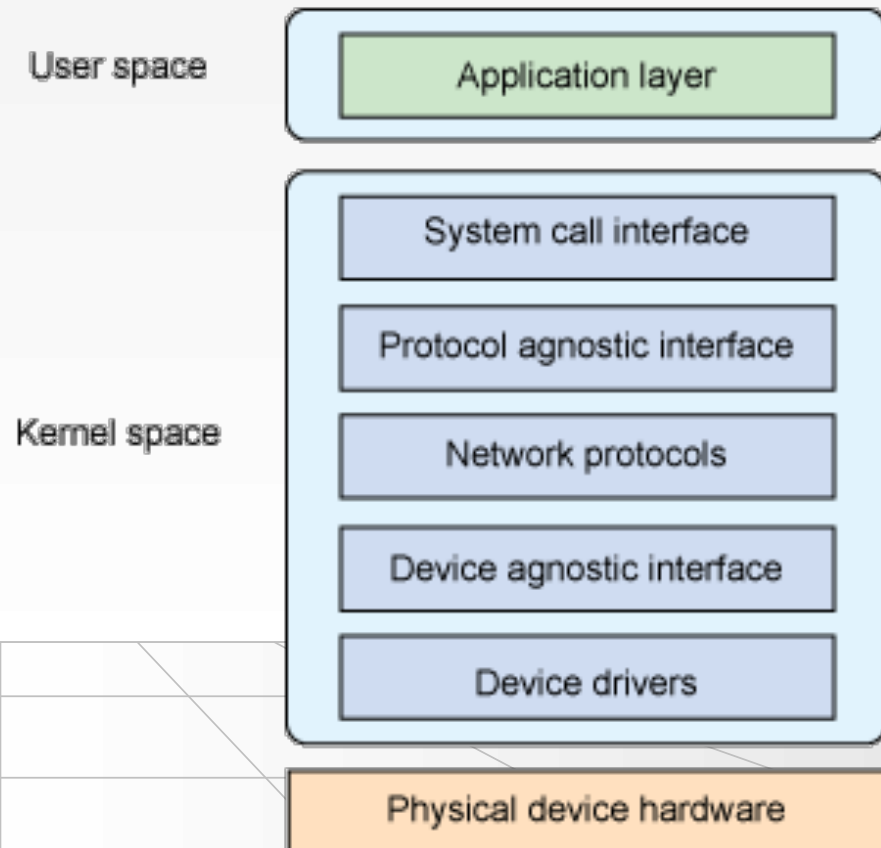
- Data Center Bridging tries to bring together the advantages of Fibrechannel, Ethernet and InfiniBand on a single medium
- It has been triggered by the storage community (the people selling Fibrechannel over Ethernet and iSCSI)
- It achieves low latency and reliable transport at constant throughput through flow-control and buffering in the sources
- 802.1Qau, 802.1Qbb, 802.3bd could allow us to go away from store&forward in DAQ LANs (→ extensive R&D required)

Moving the data in and through a PC

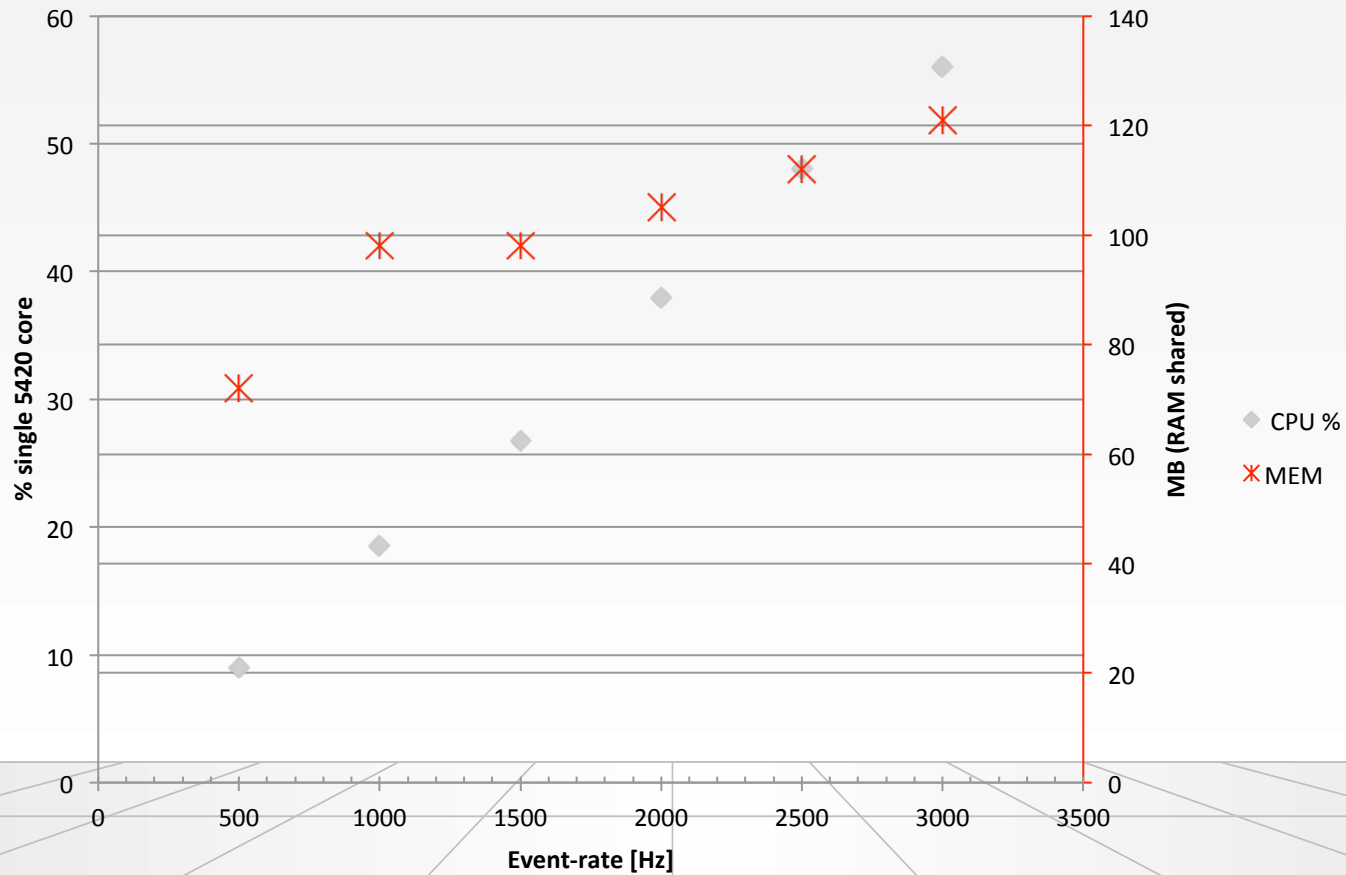


Sending and receiving data

- Multiple network protocols result in multiple software layers
- Data moving can be expensive
 - Passing data through the layers sometimes cannot be done without copying
 - Header information needs to be stripped off → splicing, alignment etc...
- In Ethernet receiving is quite a bit more expensive than sending
- Holy grail is **zero-copy event-building** (difficult with classical Ethernet, easier with InfiniBand)



The cost of event-building in the server



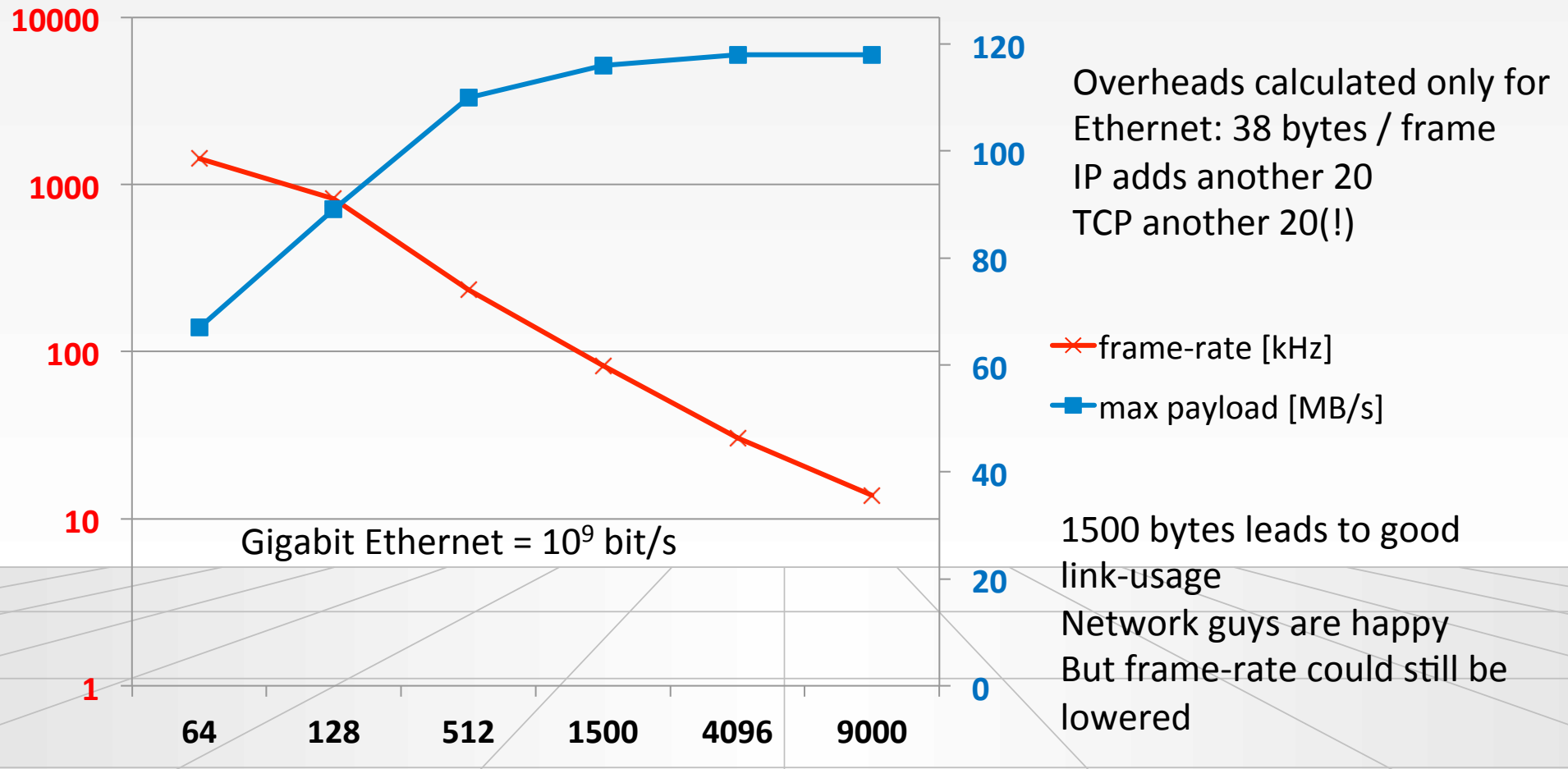
% CPU of one Intel 5420 core (a 4 core processor running at 2.5 GHz)

MEM is resident memory (i.e. pages locked in RAM)

Precision of measurements is about 10% for CPU and 1% for RAM

Frame-size / payload & frame-rate

Or: why don't they increase the MTU?

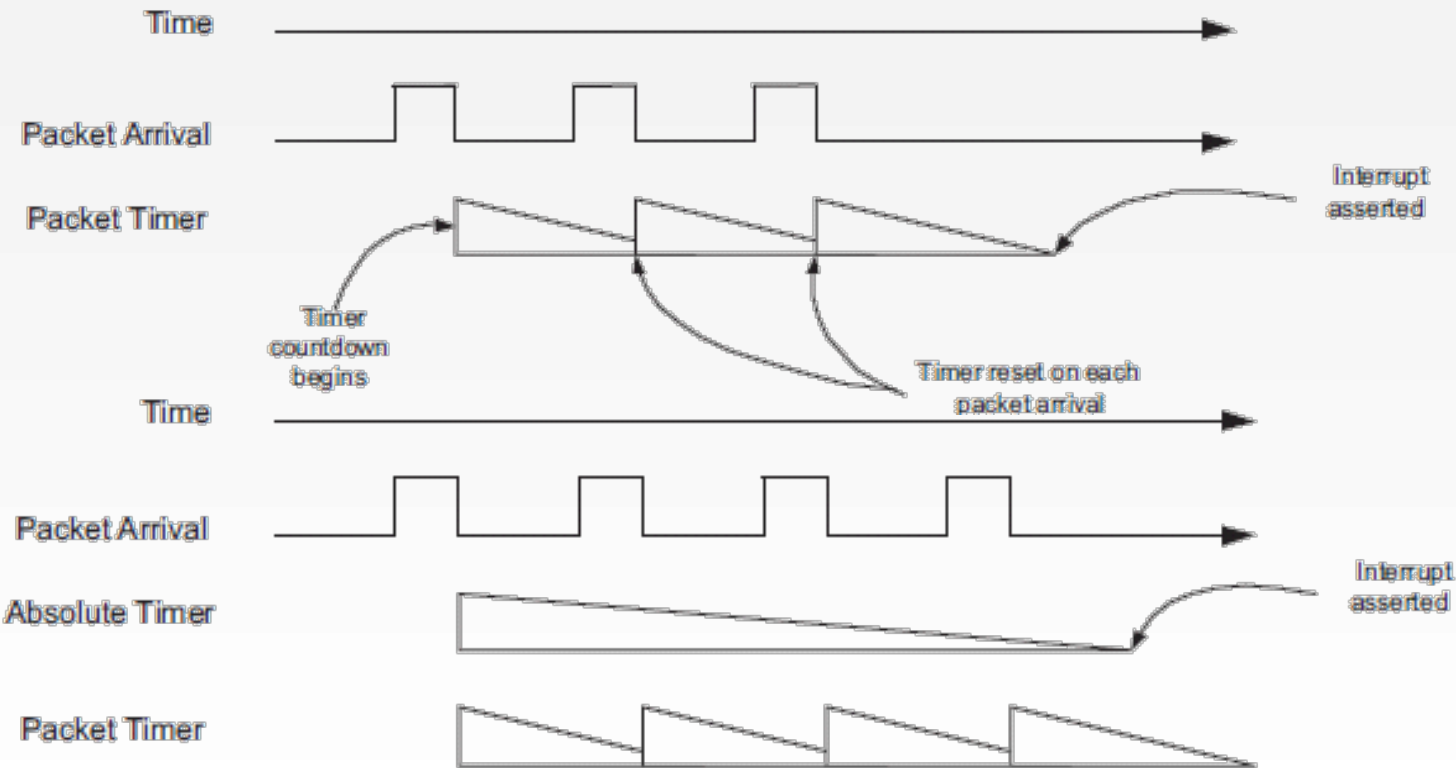


Tuning for bursty traffic

- In general provide for lots of buffers in the kernel, big socket buffers for the application and tune the IRQ moderation
- Examples here are for Linux, 2.6.18 kernel, Intel 82554 NICs (typical tuning in the LHCb DAQ cluster)

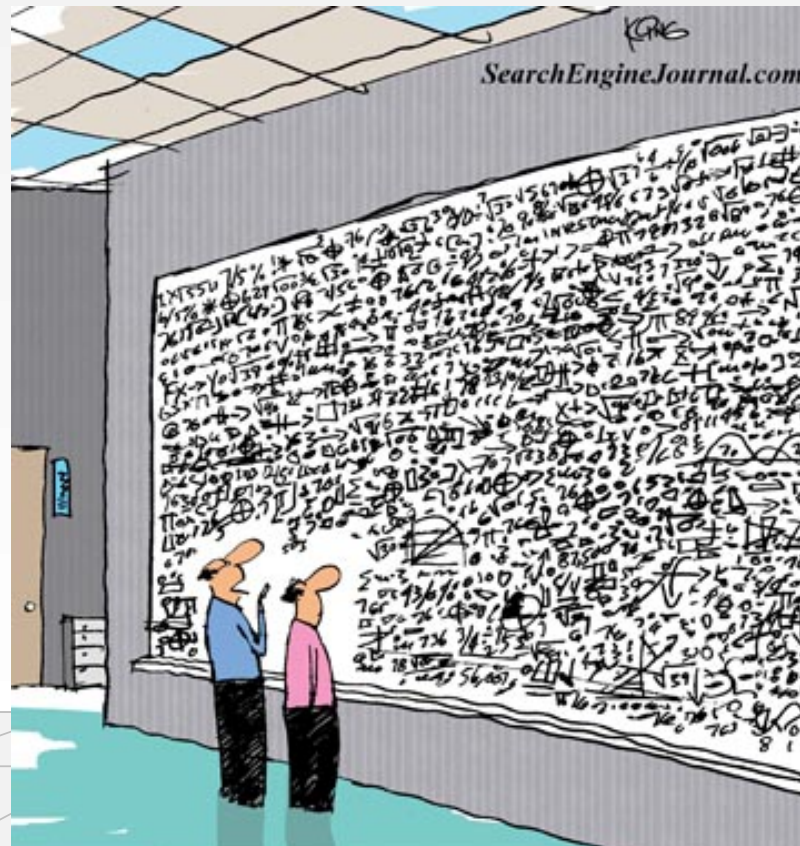
```
/sbin/ethtool -G eth1 rx 1020 # set number of RX descriptors
in NIC to max
# the following are set with sysctl -w
net.core.netdev_max_backlog = 4000
net.core.rmem_max = 67108864
# the application is tuned with setsockopt()
```

Interrupt Moderation



- Careful tuning necessary – multi-core machines can take more IRQs (but: spin-locking...)
- Can afford to ignore at 1 Gbit/s but will need to come back to this for 10 Gbit/s and 40 Gbit/s

High Level Trigger Farms



And that, in simple terms, is what we do in the High Level Trigger

Event-filtering

- Enormous amount of CPU power needed to filter events
 - Alternative is not to filter and store everything (ALICE)
- Operating System: Linux SLC5 32-bit and 64-bits: standard kernels, no (hard) real-time
- Hardware:
 - PC-server (Intel and AMD): rack-mount and blades
 - All CPU-power local: no grid, no clouds (yet?)

Online Trigger Farms 2011

	ALICE	ATLAS	CMS	LHCb
# cores	2700	17000	10000	15500
total available power (kW)		~ 2000 ⁽¹⁾	~ 1000	550
currently used power (kW)		~ 250	450 ⁽²⁾	~ 145
total available cooling power	~ 500	~ 820	800 (currently)	525
total available rack-space (Us)	~ 2000	2400	~ 3600	2200
CPU type(s)	AMD Opteron, Intel 54xx, Intel 56xx	Intel 54xx, Intel 56xx	Intel 54xx, Intel 56xx	Intel 54xx, Intel 56xx

(1) Available from transformer (2) PSU rating

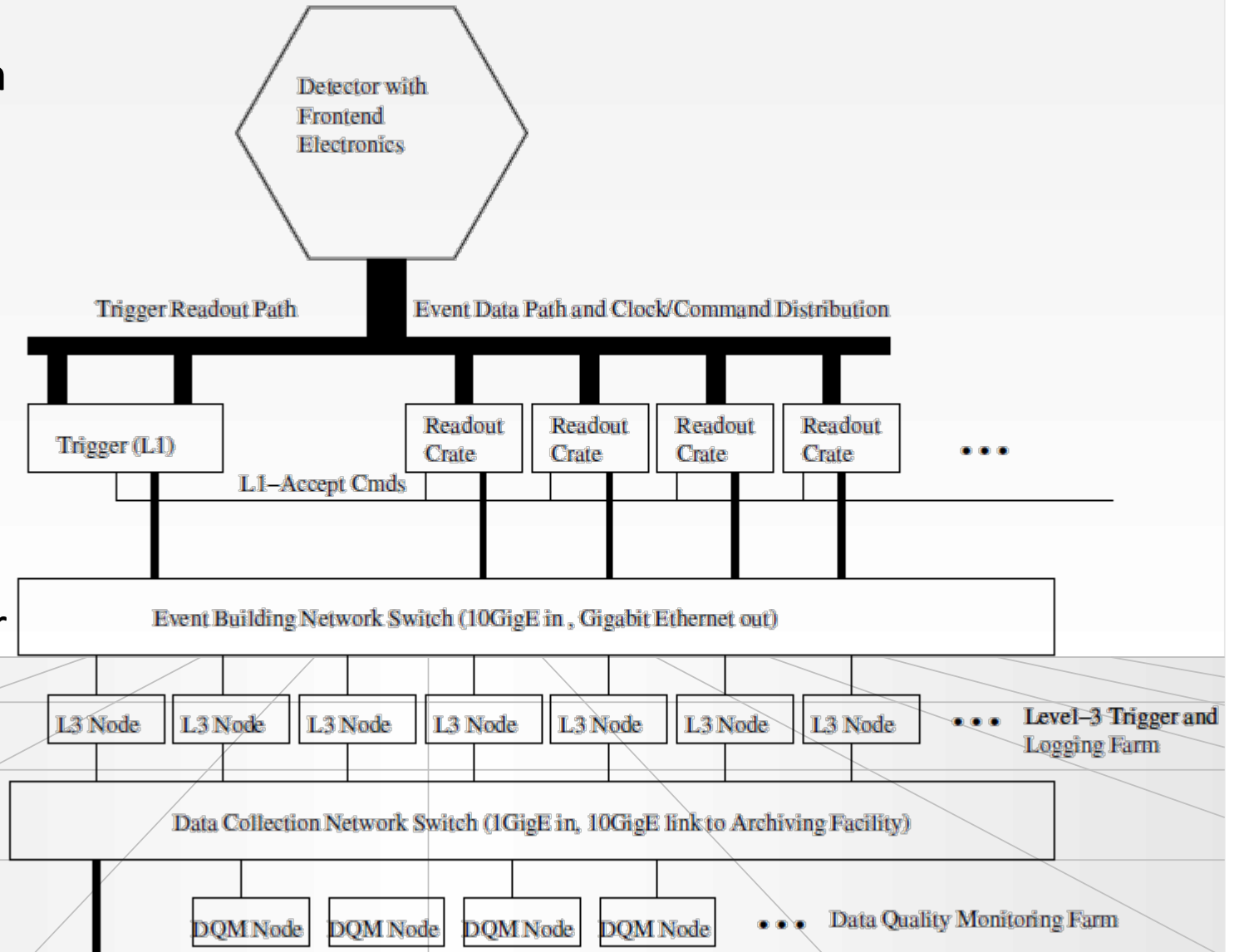
Faster, Larger – the future

A bit of marketing for upcoming DAQ systems



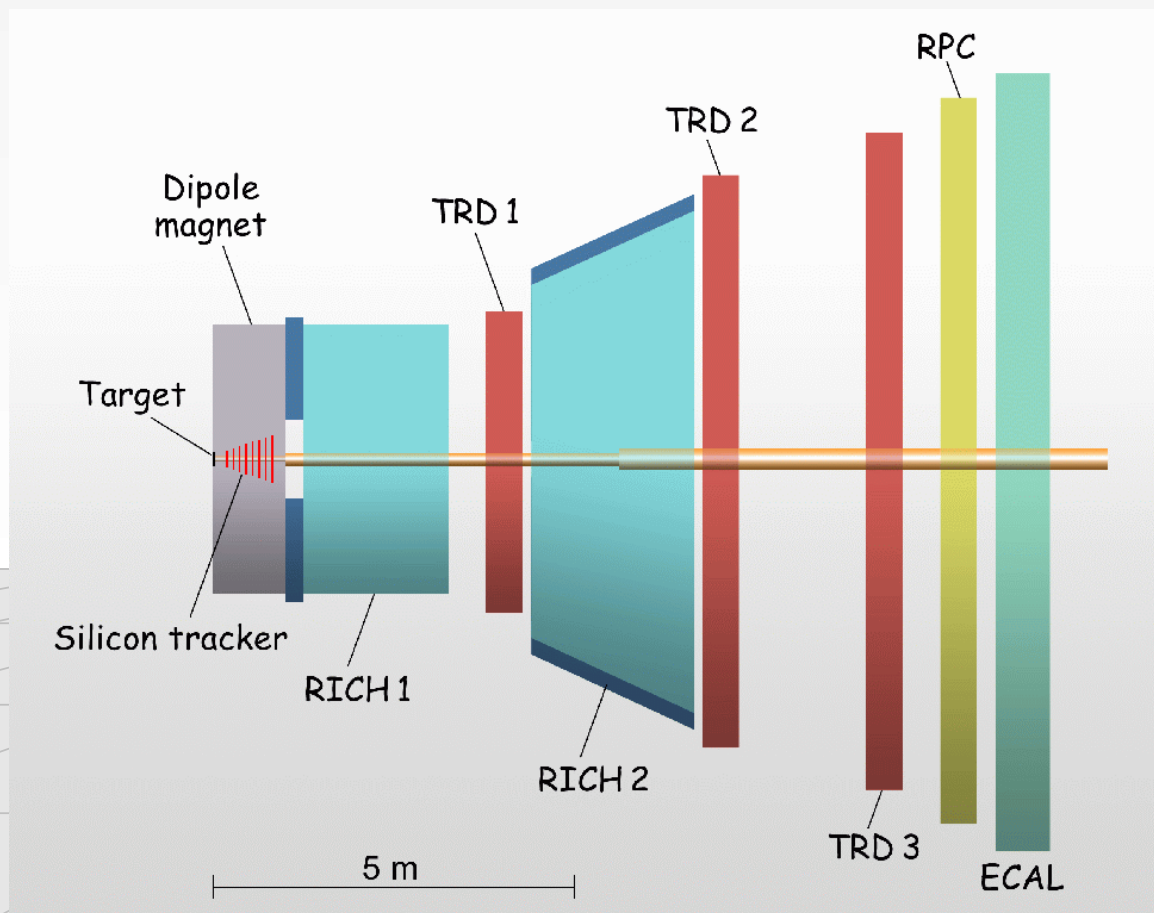
SuperB DAQ

- Collection in Readout-Crates
- Ethernet read-out
- 60 Gbit/s
- → if you look for a challenge in DAQ, you must join LHCb 😊 - or work on the SLHC



Compressed Baryonic Matter (CBM)

- Heavy Ion experiment planned at future FAIR facility at GSI (Darmstadt)
- Timescale: ~2014



Detector Elements

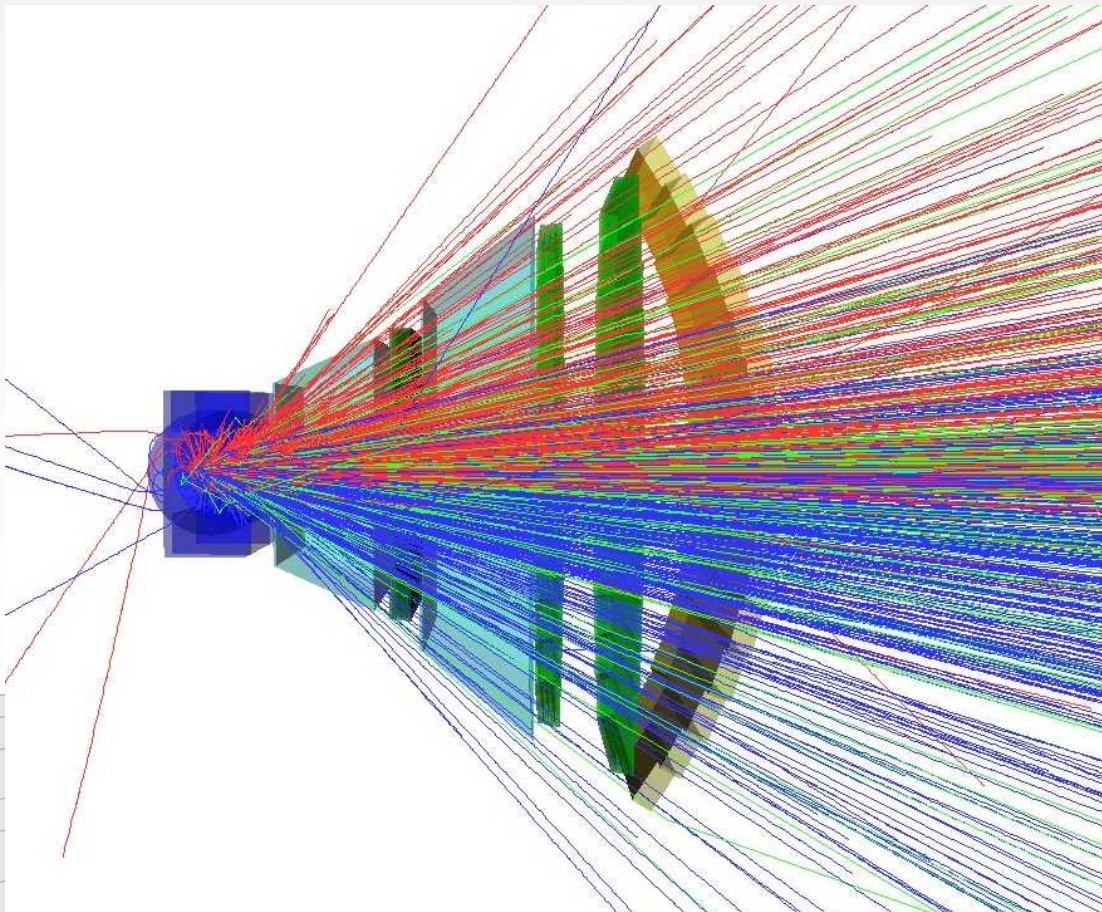
- **Si** for Tracking
- **RICH** and **TRDs** for Particle identification
- **RPCs** for ToF measurement
- **ECal** for Electromagnetic Calorimetry

Average Multiplicities:

160 p
 400 π^-
 400 π^+
 44 K^+
 13 K
 800 γ
1817 total at 10 MHz

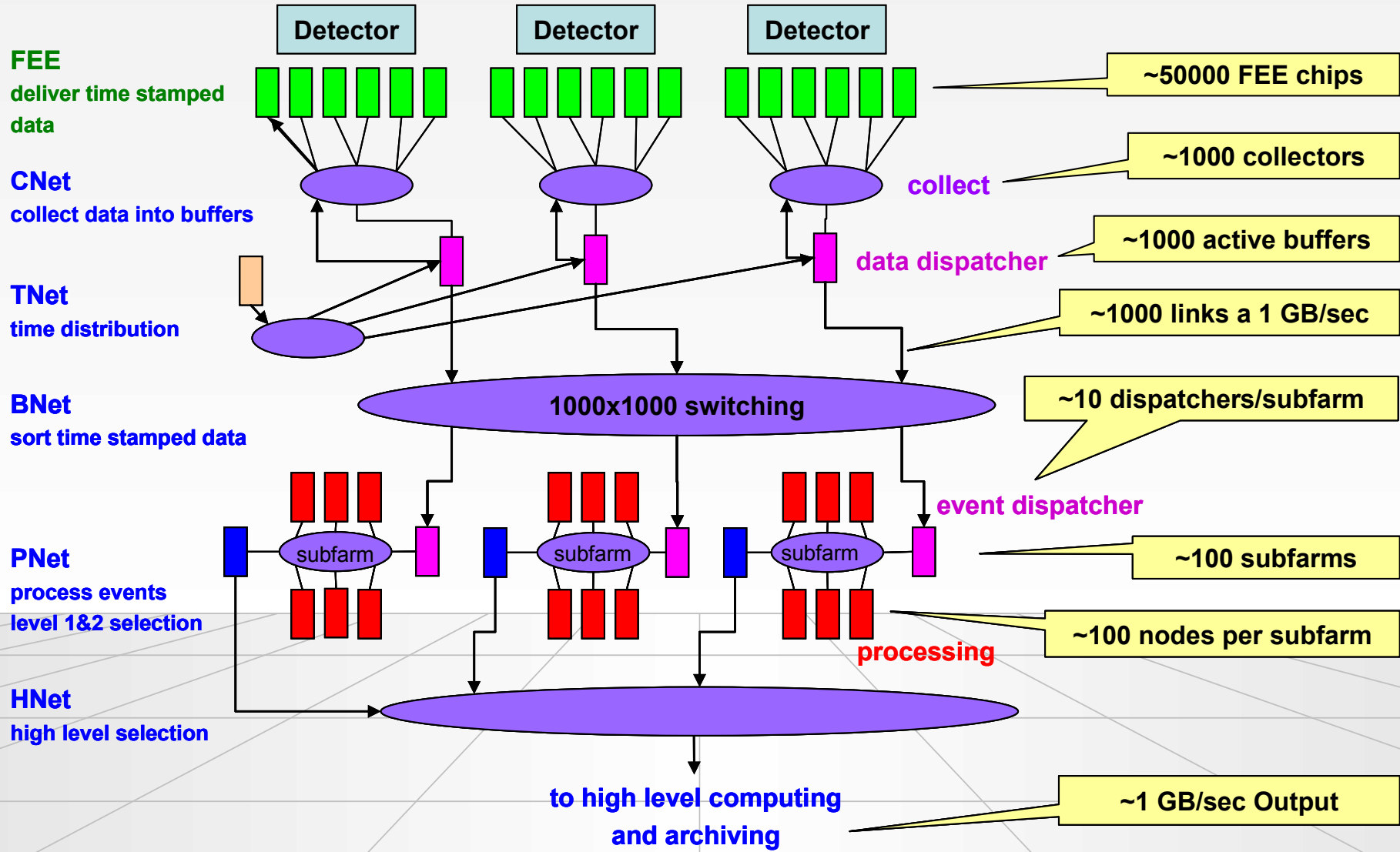
High Multiplicities

Quite Messy Events... (cf. Alice)



- ❑ Hardware triggering problematic
 - Complex Reconstruction
 - 'Continuous' beam
- ❑ **Trigger-Free Readout**
 - 'Continuous' beam
 - Self-Triggered channels with precise time-stamps
 - Correlation and association later in CPU farm

CBM DAQ Architecture



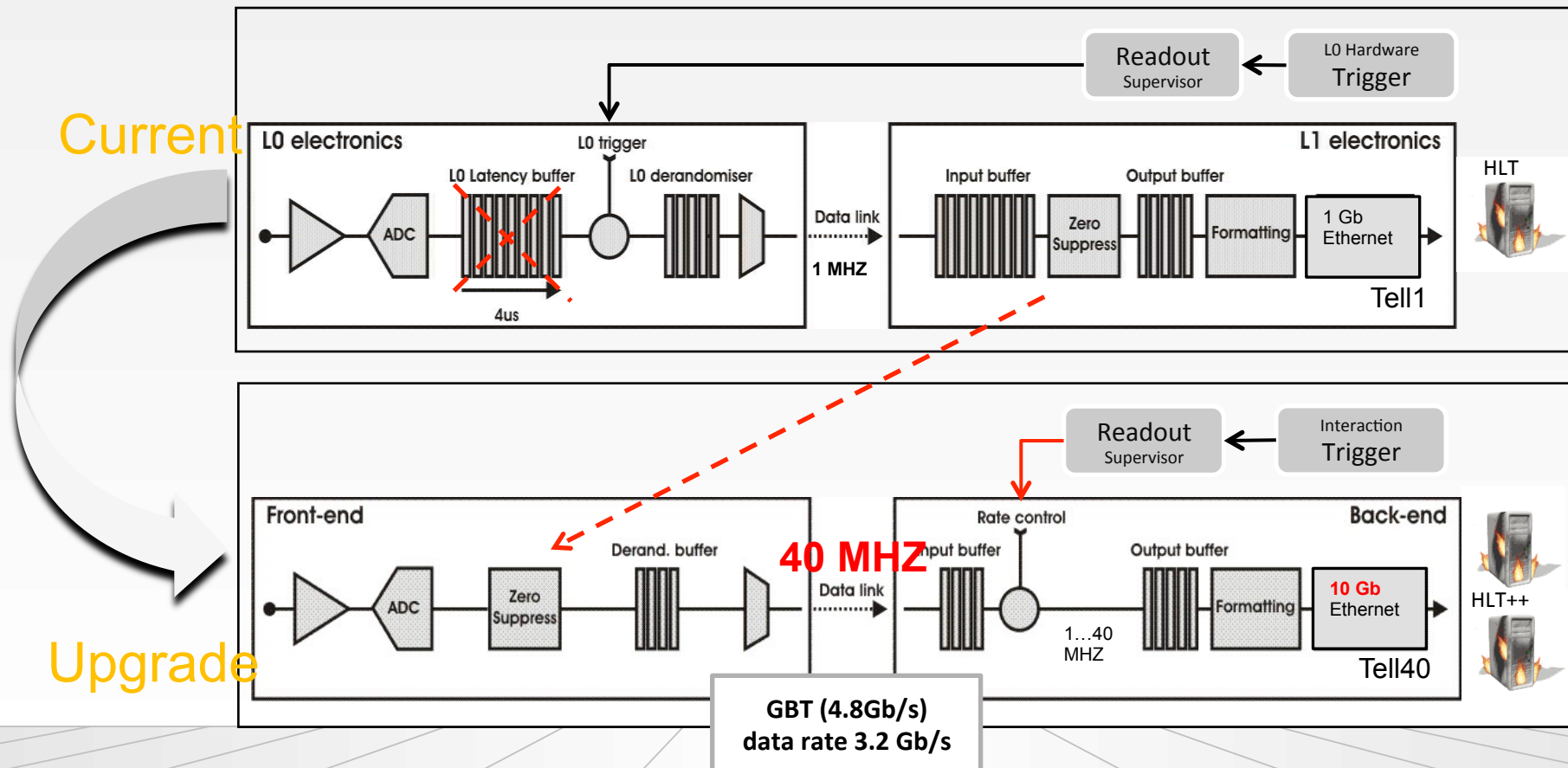
CBM Characteristics/Challenges

Very much network based

→ 5 different networks

- Very **low-jitter (10 ps)** timing distribution network
- Data collection network to link detector elements with front-end electronics (**link speed O(GB/s)**)
- **High-performance** (\sim **O(TB/s)**) event building switching network connecting O(1000) Data Collectors to O(100) Subfarms
- Processing network within a subfarm interconnecting O(100) processing elements for triggering/data compression
- Output Network for collecting data ready for archiving after selection.

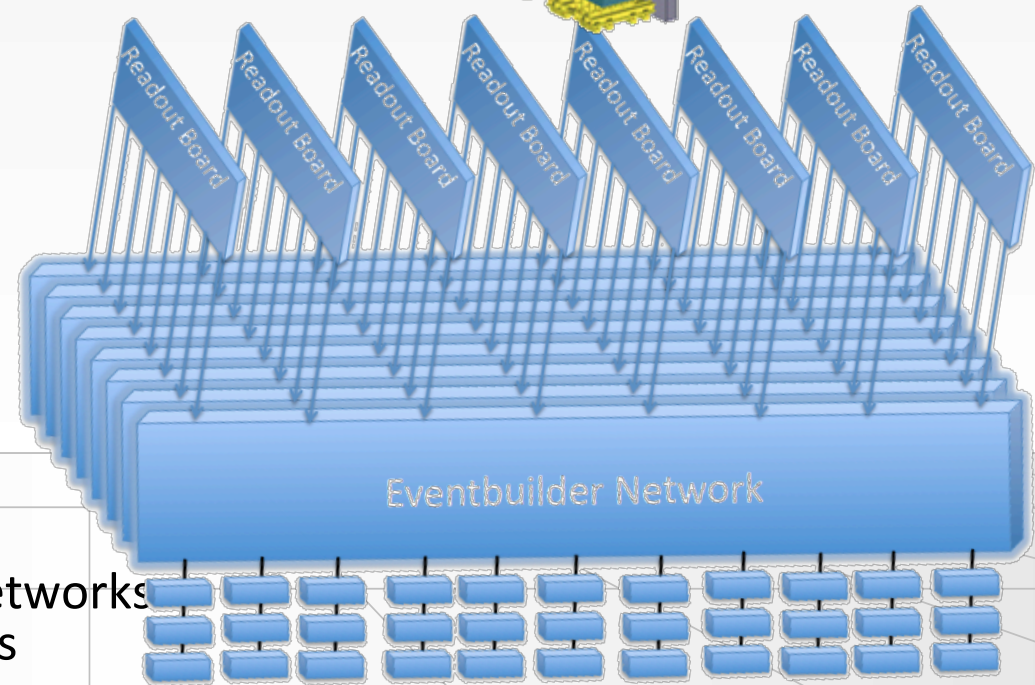
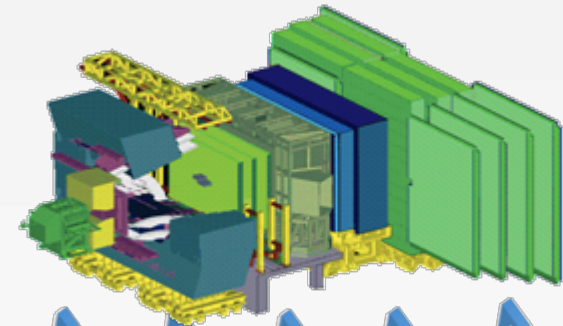
LHCb DAQ Architecture from 2018



- All data will be readout @ collision rate 40 MHz by all frontend electronics (FEE) → **a trigger-free read-out!**
- Zero-suppression will be done in FEEs to reduce the number of the GigaBit Transceiver (GBT) links

Requirements on LHCb DAQ Network

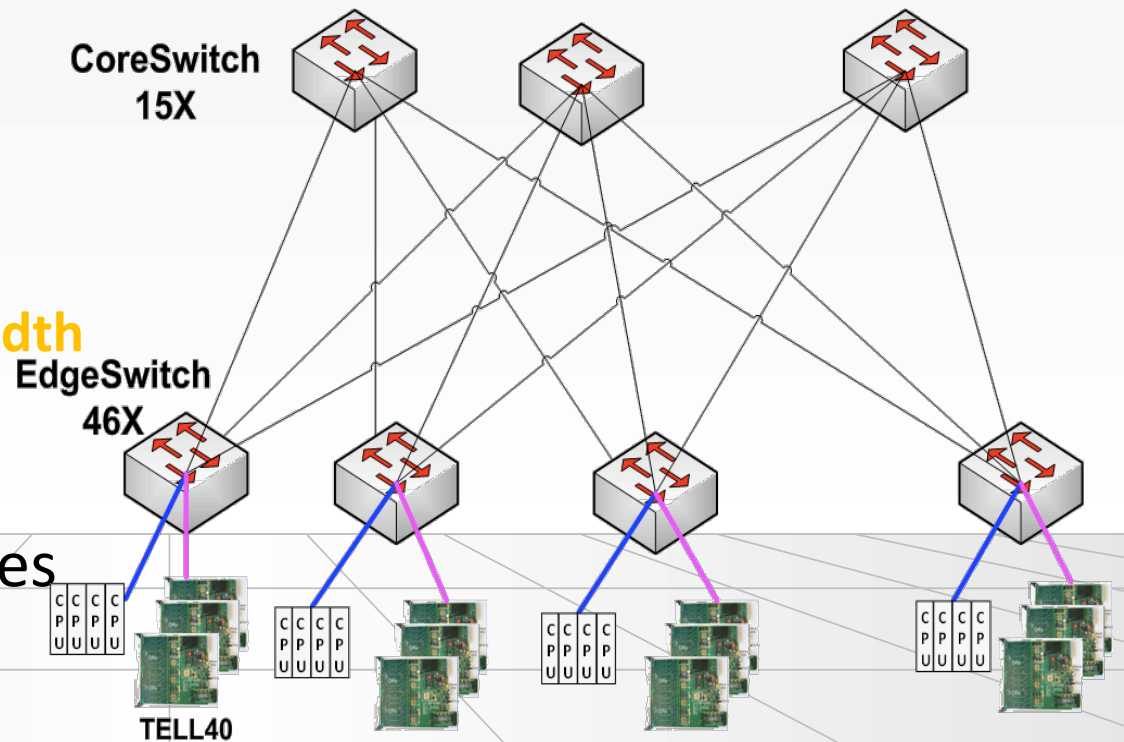
- Design:
 - 100 kB@30 MHz (10 MHz out of 40 MHz are empty) → 24 Tbit/s network required
 - Keep average link-load at 80% of wire-speed
 - ~5000 Servers needed to filter the data (depends on Moore's Law)
- We need:
 - (input) Ports for Readout boards (ROB) from the detector: 3500x10 Gb/s
 - (output) Ports for Event Filter Farm (EFF): 5000x10 Gb/s
 - Bandwidth: 34 Tb/s (unidirectional, including load-factor)
- Scaling: build several (8) sub-networks or slices. Each Readout Board is connect to each slice.



Fat-Tree Topology for One Slice

- 48-port 10 GbE switches
- Mix readout-boards (ROB) and filter-farm-servers in one switch
 - 15 x readout-boards
 - 18 x servers
 - 15 x uplinks

Non-block switching
use 65% of installed bandwidth
(classical DAQ only 50%)



- Each slice accomodates
 - 690 x inputs (ROBS)
 - 828 x outputs servers

Ratio (server/ROB) is adjustable

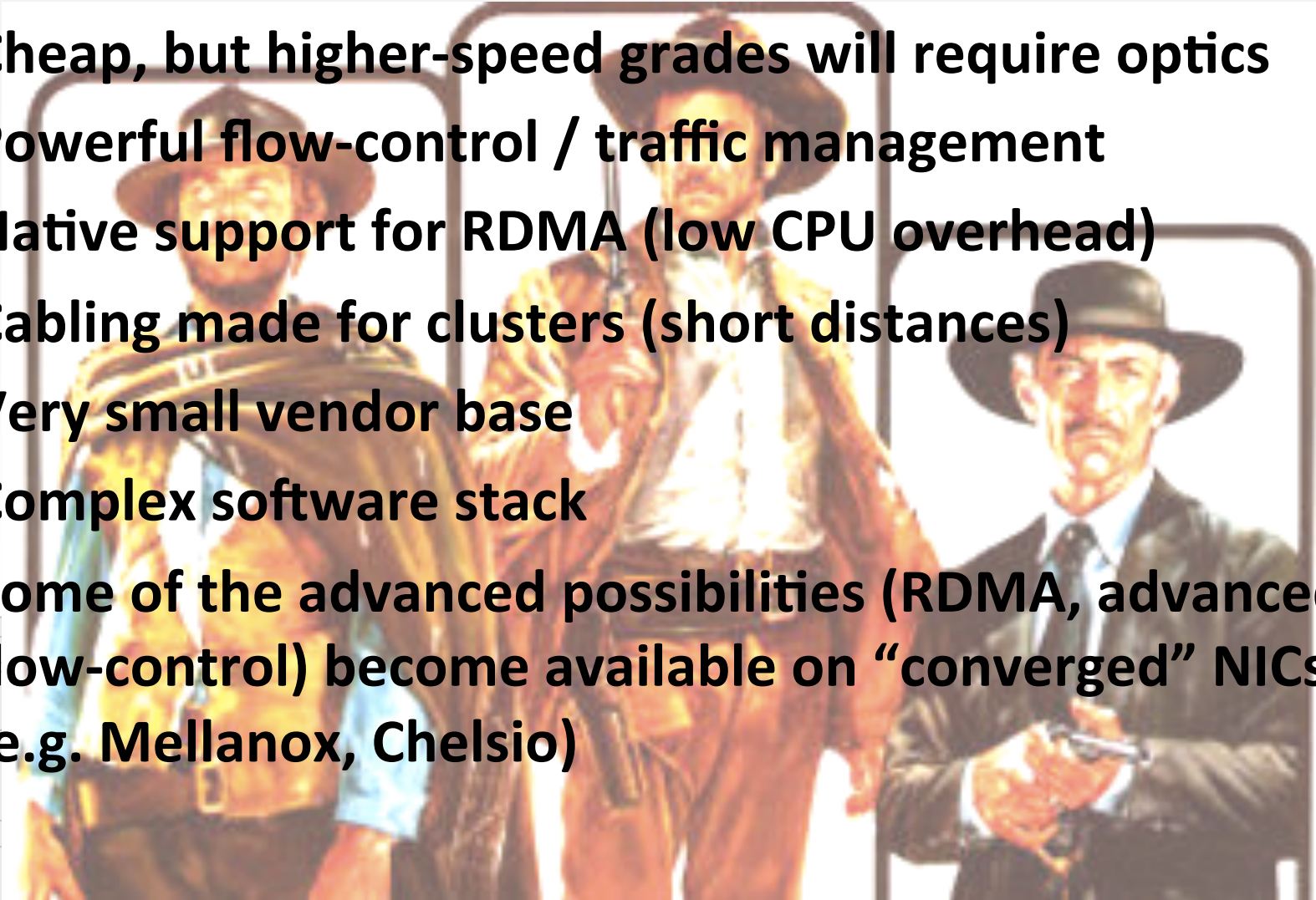
InfiniBand

	SDR	DDR	QDR	FDR	EDR	HDR	NDR
1X	2 Gbit/s	4 Gbit/s	8 Gbit/s	14 Gbit/s	25 Gbit/s	125 Gbit/s	750 Gbit/s
4X	8 Gbit/s	16 Gbit/s	32 Gbit/s	56 Gbit/s	100 Gbit/s	500 Gbit/s	3000 Gbit/s
12X	24 Gbit/s	48 Gbit/s	96 Gbit/s	168 Gbit/s	300 Gbit/s	1500 Gbit/s	9000 Gbit/s

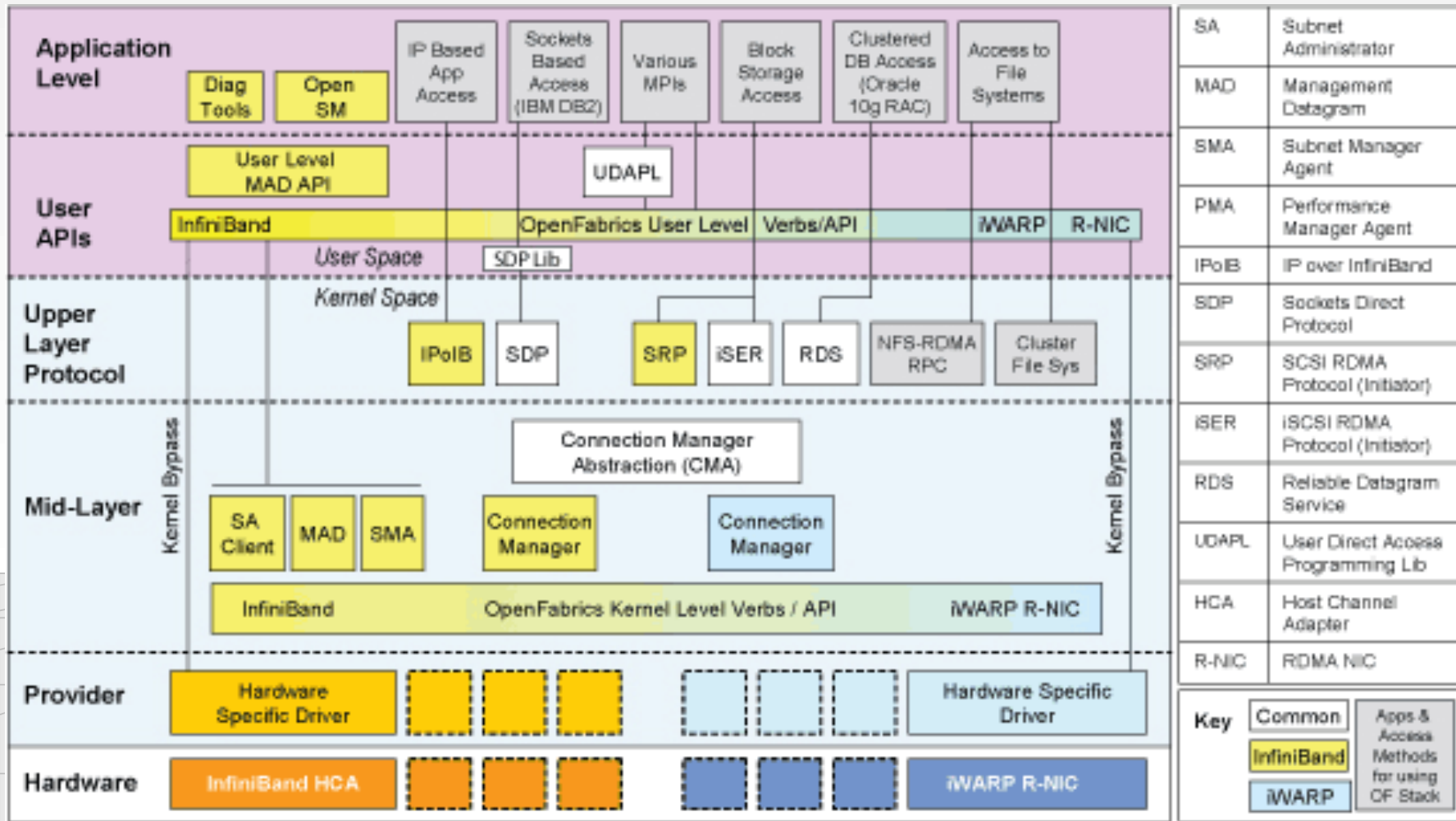
- High bandwidth (32 Gbit/s, 56 Gbit/s ...) – always a step ahead of Ethernet 😊
- Low price / switch-port
- Very low latency

InfiniBand: the good, the bad and the ugly

- **Cheap, but higher-speed grades will require optics**
- **Powerful flow-control / traffic management**
- **Native support for RDMA (low CPU overhead)**
- **Cabling made for clusters (short distances)**
- **Very small vendor base**
- **Complex software stack**
- **Some of the advanced possibilities (RDMA, advanced flow-control) become available on “converged” NICs (e.g. Mellanox, Chelsio)**



Is the future of DAQ in the OFED?



Future DAQ systems (choices)

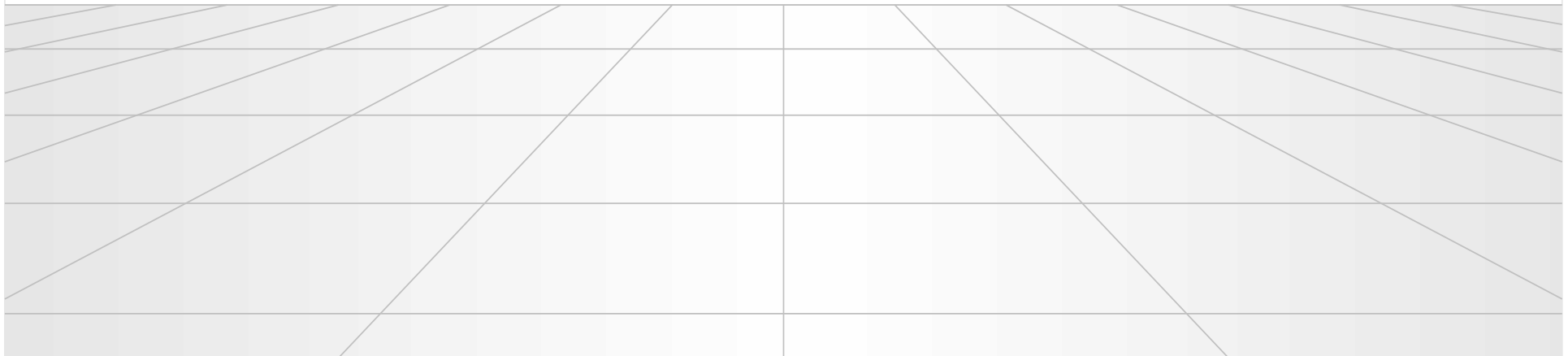
- Certainly LAN based
 - InfiniBand deserves a serious evaluation for high-bandwidth (> 100 GB/s)
 - In Ethernet if DCB works, might be able to build networks from smaller units, otherwise we will stay with large store&forward boxes
- Trend to “trigger-free” → do everything in software → bigger DAQ will continue
 - Physics data-handling in commodity CPUs
- Will there be a place for multi-core / coprocessor cards (Intel MIC / CUDA)?
 - IMHO this will depend on if we can establish a development framework which allows for longterm maintenance of the software by non-“geek” users, much more than on the actual technology

Summary and Future

- Large modern DAQ systems are based entirely on Ethernet and big PC-server farms
- Bursty, uni-directional traffic is a challenge in the network and the receivers, and requires substantial buffering in the switches
- The future:
 - It seems that buffering in switches is being reduced (latency vs. buffering)
 - Advanced flow-control is coming, but it will need to be tested if it is sufficient for DAQ
 - Ethernet is still strongest, but InfiniBand looks like a very interesting alternative
 - Integrated protocols (RDMA) can offload servers, but will be more complex



Publicita / Publicité / Commercial





ISOTDAQ 3rd International School 2012 OF TRIGGER AND DATA ACQUISITION

isotdaq.web.cern.ch

Registration until 1 November 2011

1 - 8 February 2012
Cracow, Poland



Topics

Trigger

NIM Electronics
Front-end Electronics
FPGA Programming

Data Acquisition

ADC, TDC, Detector Readout
Event & Buffer Management
DAQ Control Software
Storage Technologies

Data Transfer Technologies

VMEbus, xTCA
PCI, PCI-X
Data Networks

Review Talks

LHC Experiments
ATLAS TDAQ Architecture
CMS TDAQ Architecture

Advisory Committee

Enrico Pasqualucci (INFN Roma)
Gokhan Unel (UCI)
Kostas Kordas (AUTH)
Livio Mapelli (CERN)
Markus Joos (CERN)
Niko Neufeld (CERN)
Robert McLaren (CERN)
Speranza Falciano (INFN Roma)
Christoph Schwick (CERN)
Hannes Sakulin (CERN)
Janusz Chwastowski (CUT, IFJ PAN)
Krzysztof Korcyl (IFJ PAN, CUT)
Michal Turala (IFJ PAN)



photo from www.flickr.com/photos/bazylek

Thanks

- I acknowledge gratefully the help of many colleagues who provided both material and suggestions: Guoming Liu, Beat Jost, David Francis, Frans Meijers, Gordon Watts, Clara Gaspar