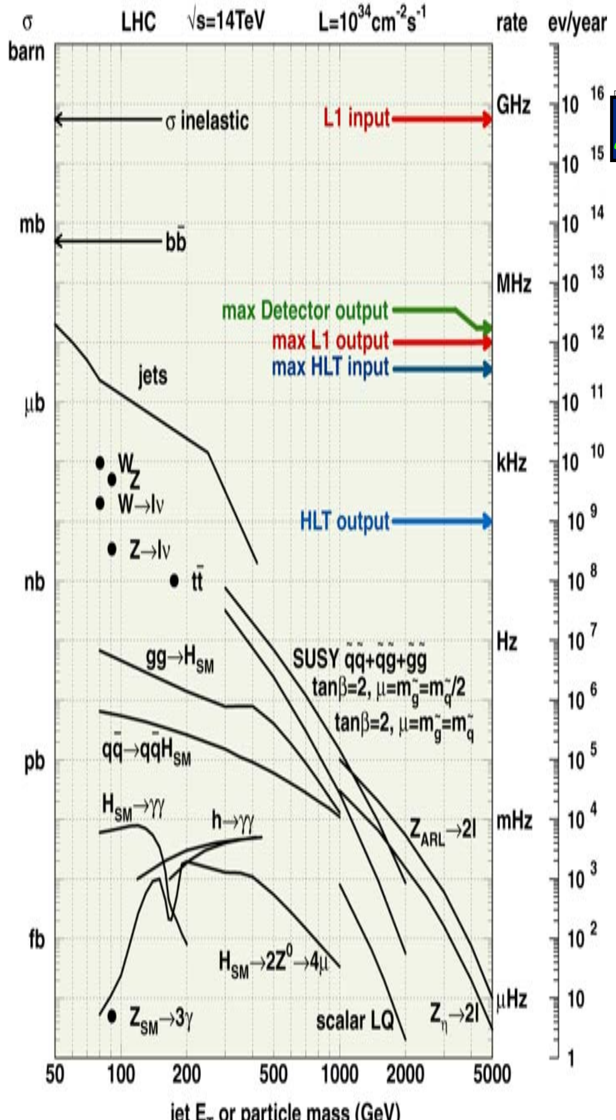
Trigger and DAQ systems (at the LHC)

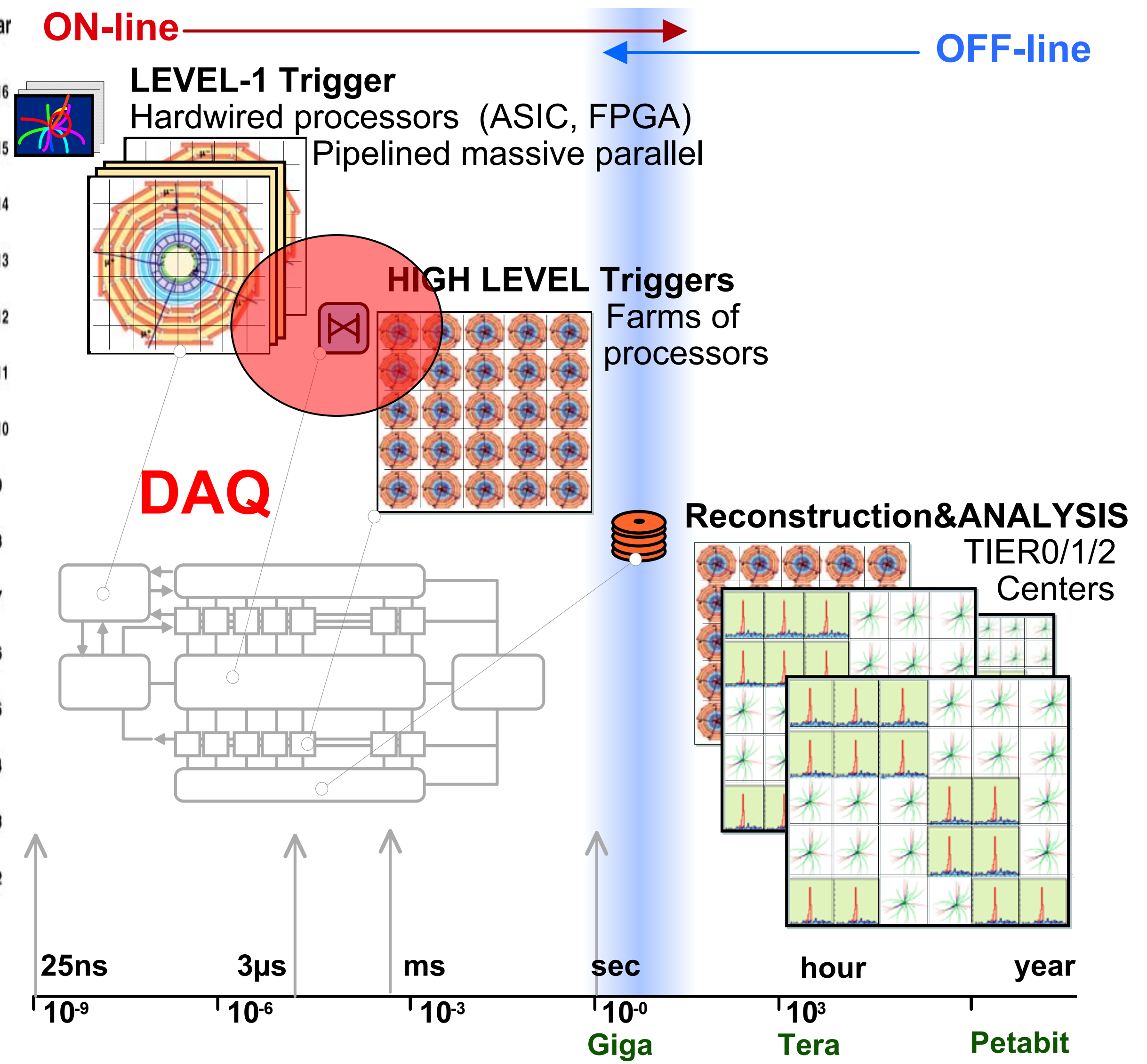
Paris Sphicas
CERN/PH and Univ. of Athens
Summer Student Lectures
August 3 2004

- **Introduction**
- **Level-1 Trigger**
- **DAQ**
  - **Readout**
  - **Switching and Event Building**
  - **Control and Monitor**
- **High-Level trigger**

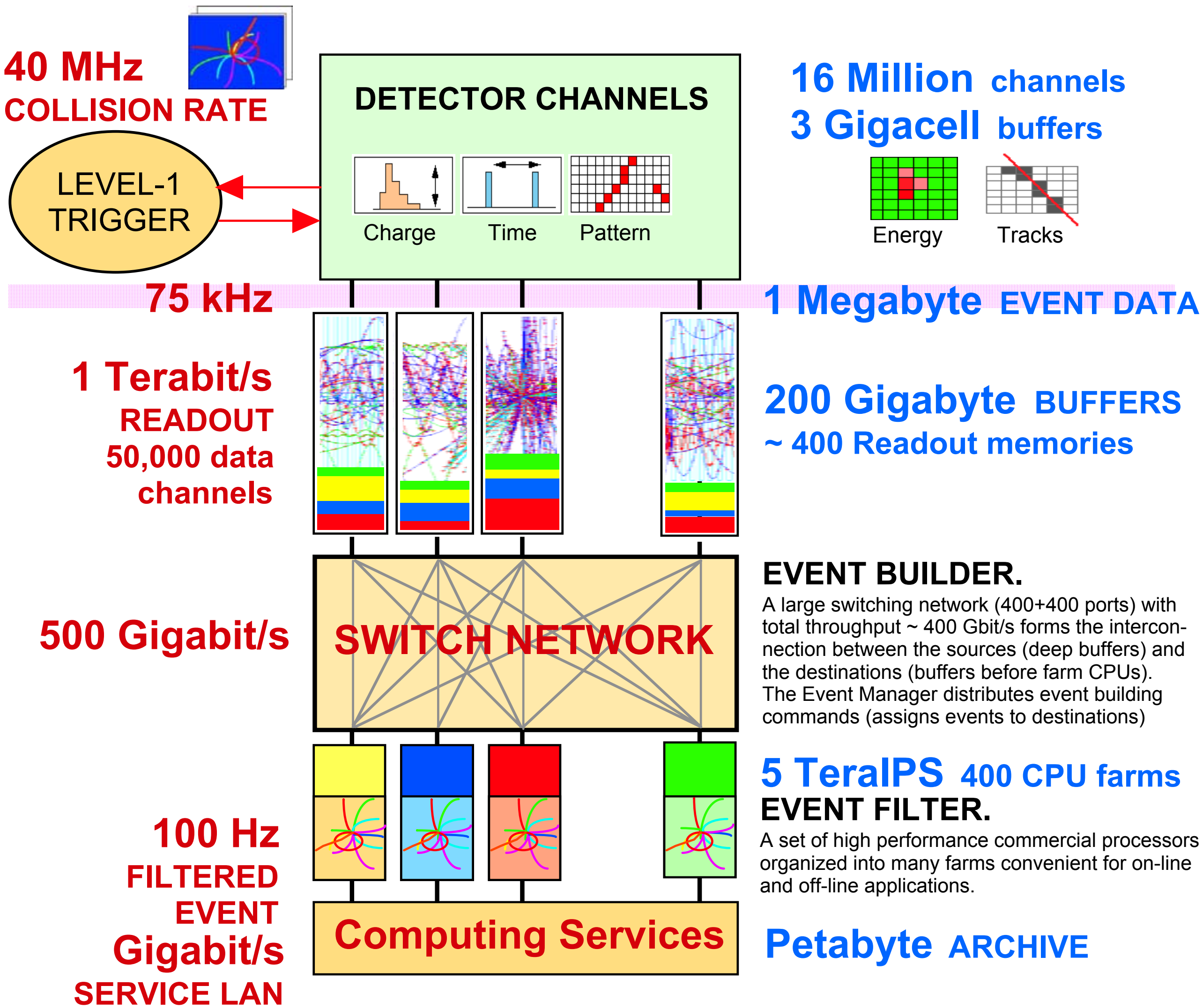
DAQ system

Physics selection at the LHC

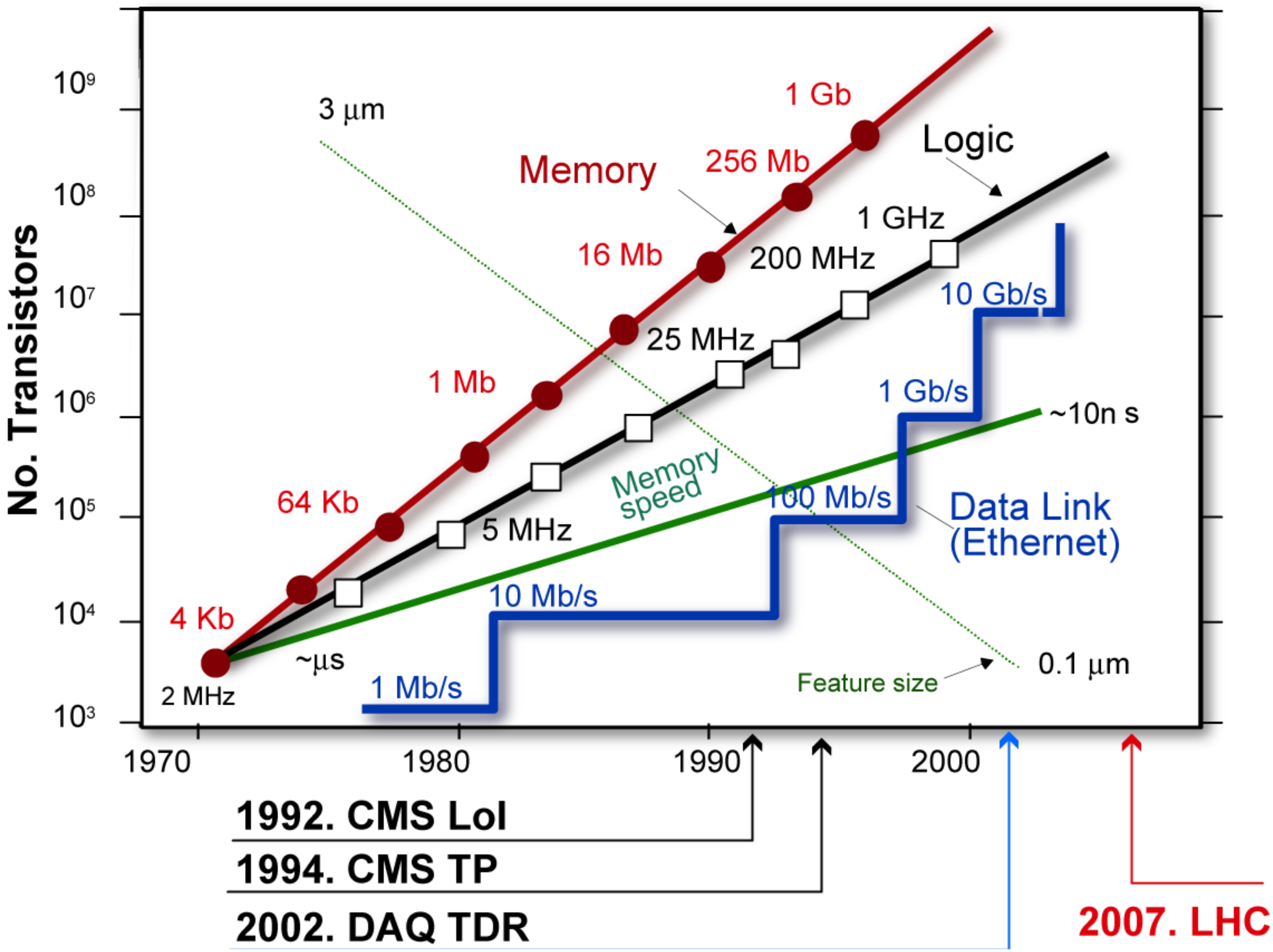




Online Selection Flow in pp

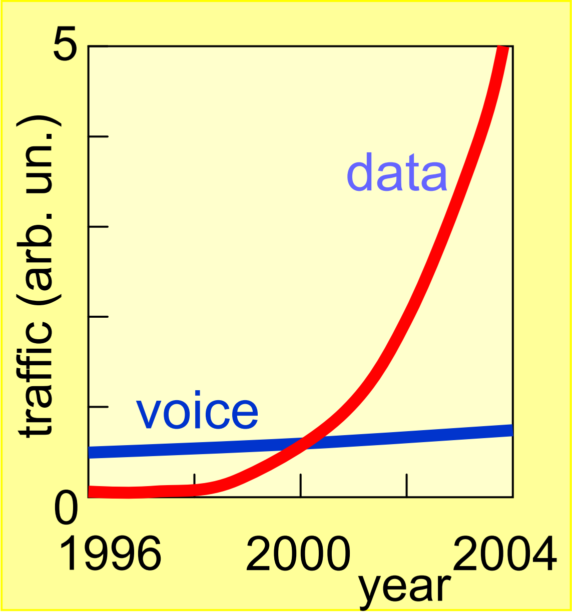


Technology evolution



Internet Growth (a reminder)

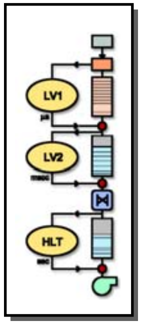
- **100 million new users online in 2001**
- **Internet traffic doubled every 100 days**
- **5000 domain names added every day**
- **Commerce in 2001: >$200M**
- **1999: last year of the voice**
- **Prices(basic units) dropping**
- **Need more bandwidth**
- **Conclusion:**
  - **It'll go on; can count on it.**

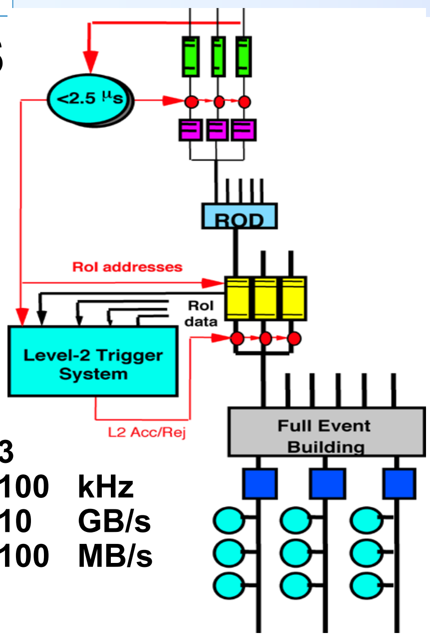


Pietro M. DI VITA / Telecom ITALIA
Telecom99

Trigger/DAQ systems: grand view

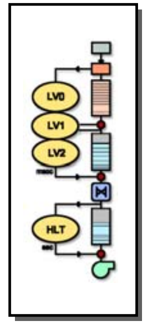
ATLAS

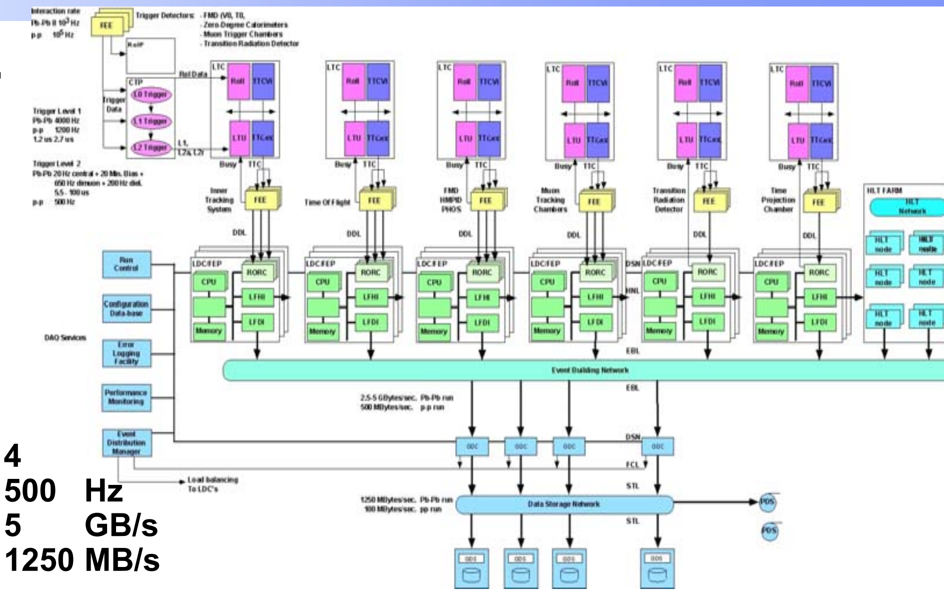




Levels	3	
LV-1 rate	100	kHz
Readout	10	GB/s
Storage	100	MB/s

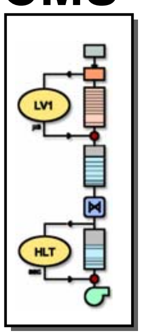
ALICE

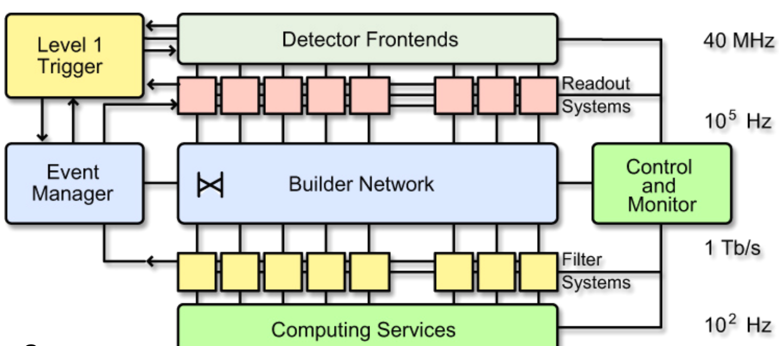




Levels	4	
LV-1 rate	500	Hz
Readout	5	GB/s
Storage	1250	MB/s

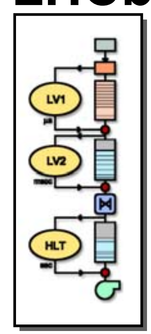
CMS

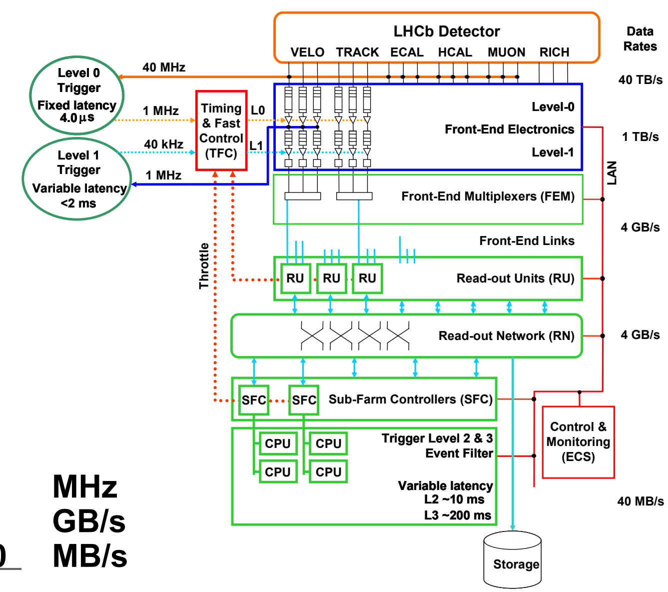




Levels	2	
LV-1 rate	100	kHz
Readout	100	GB/s
Storage	100	MB/s

LHCb

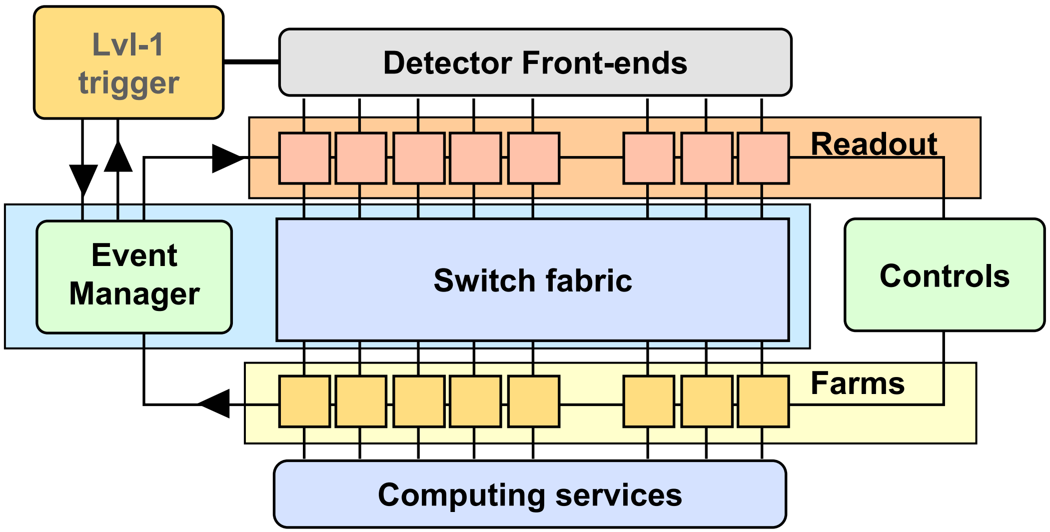




Levels	3	
LV-1 rate	1	MHz
Readout	4	GB/s
Storage	40	MB/s

Trigger/DAQ: basic blocks

- **Current Trigger/DAQ elements**



Detector Front-ends, feed Lvl-1 trigger processor

Readout Units: buffer events accepted by Lvl-1 trigger

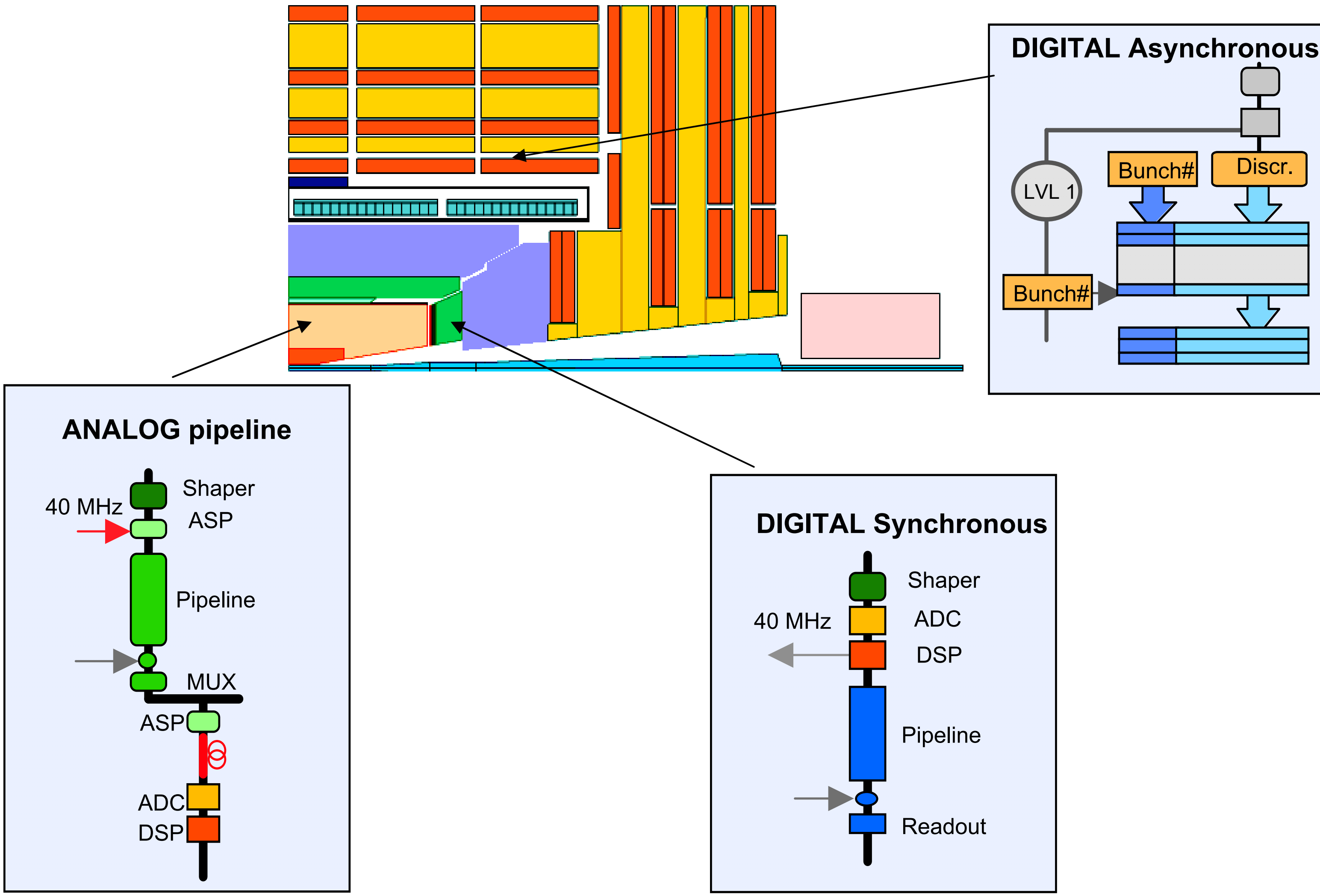
Switching network: interconnectivity with HLT processors

Processor Farm

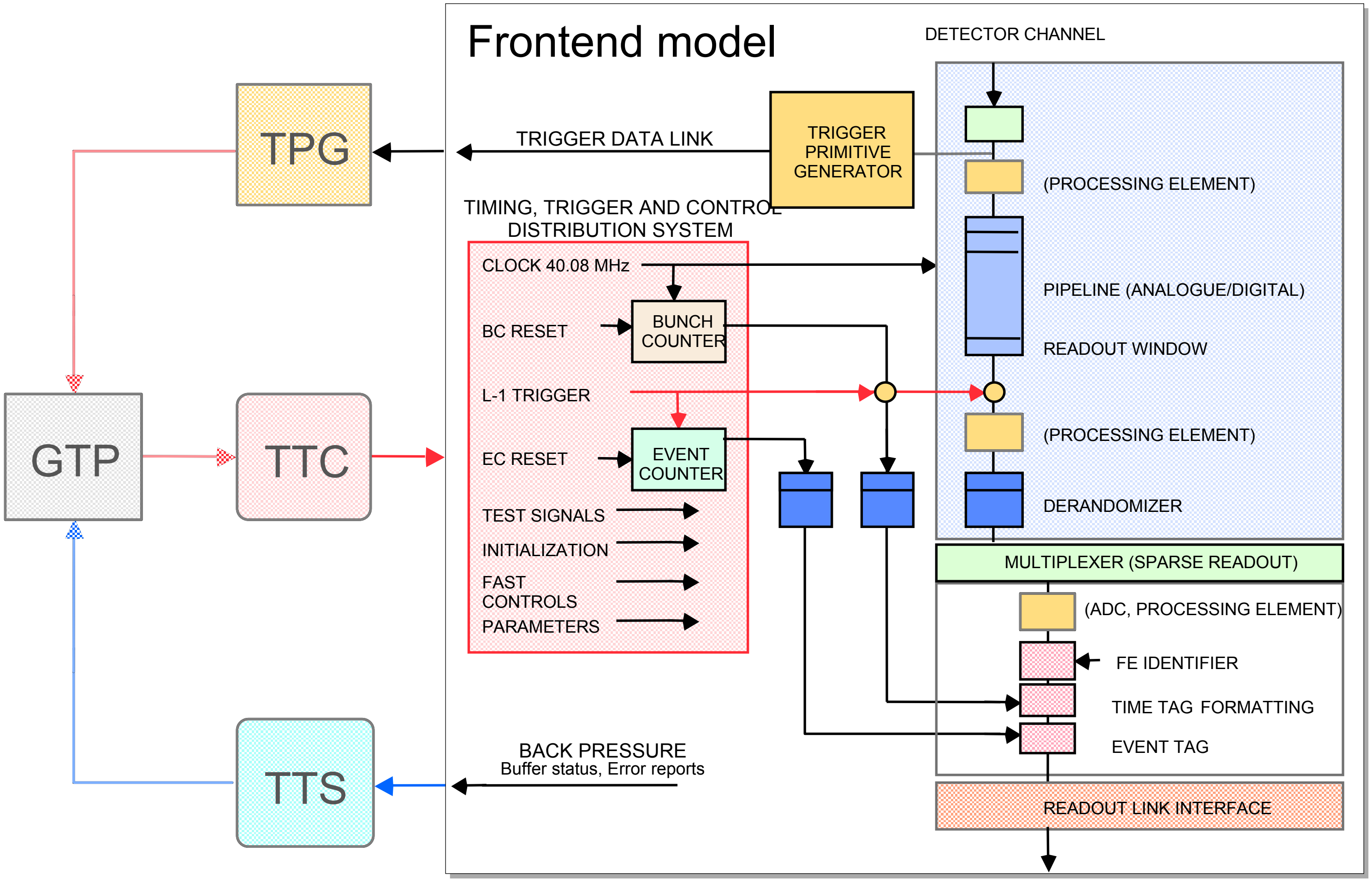
+ control and monitor

Readout

Detector Readout: front-end types

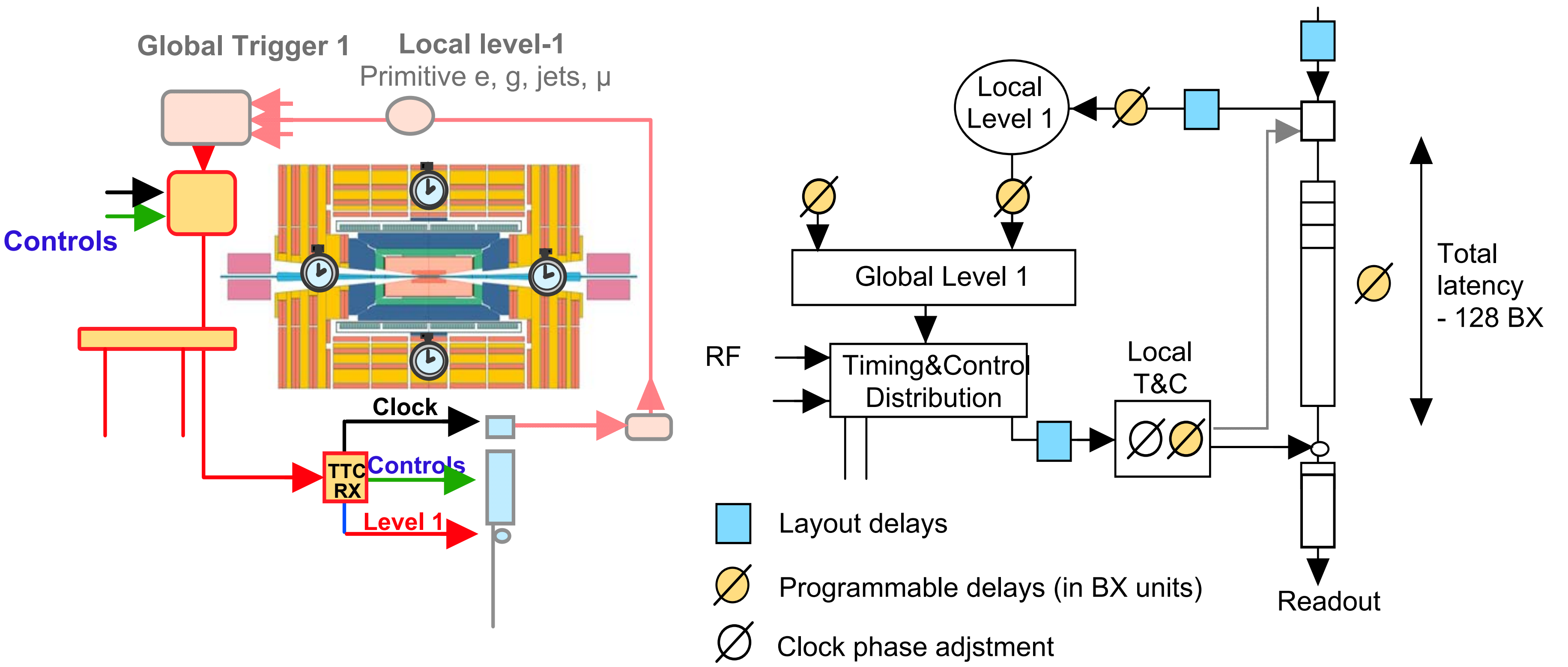


Readout: Front-End electronics (model)



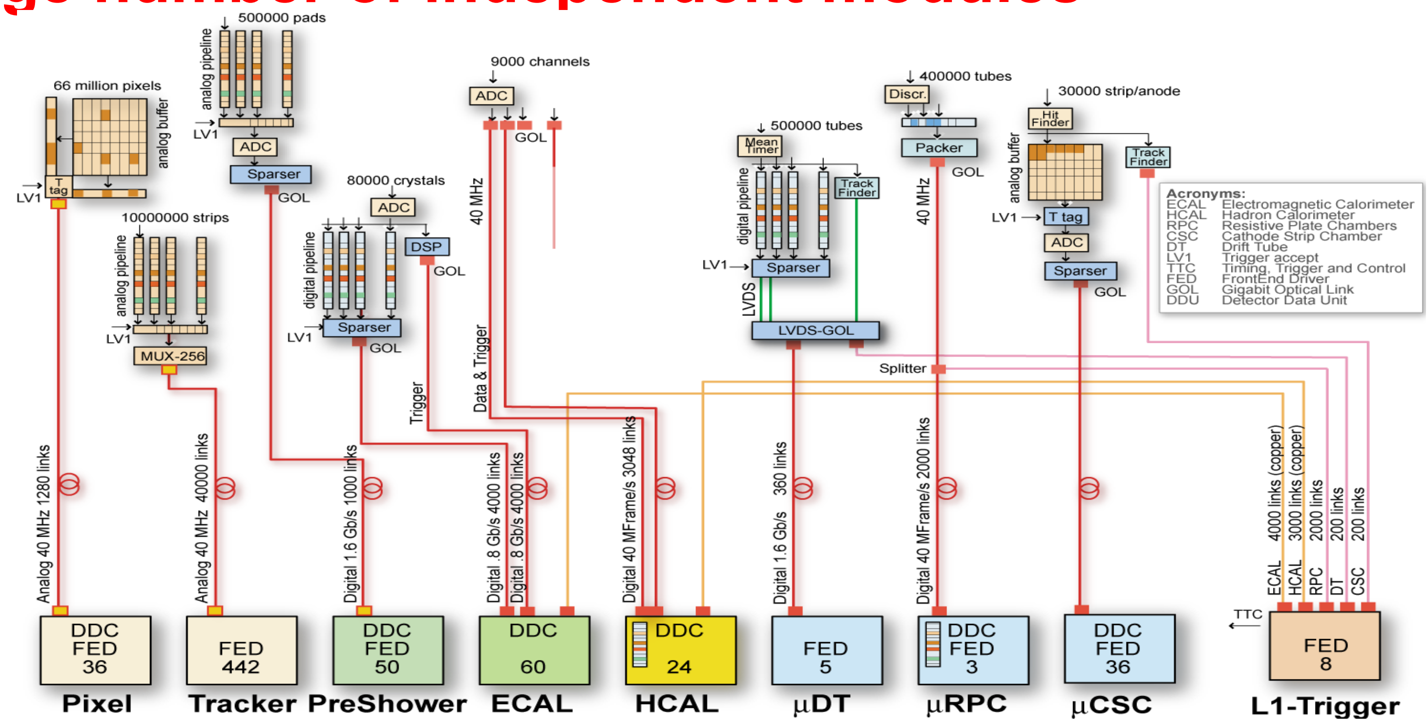
Clock distribution & synchronization

- **Trigger, Timing & Control (TTC); from RD12**



Need standard interface to front-ends

- **Large number of independent modules**



DAQ

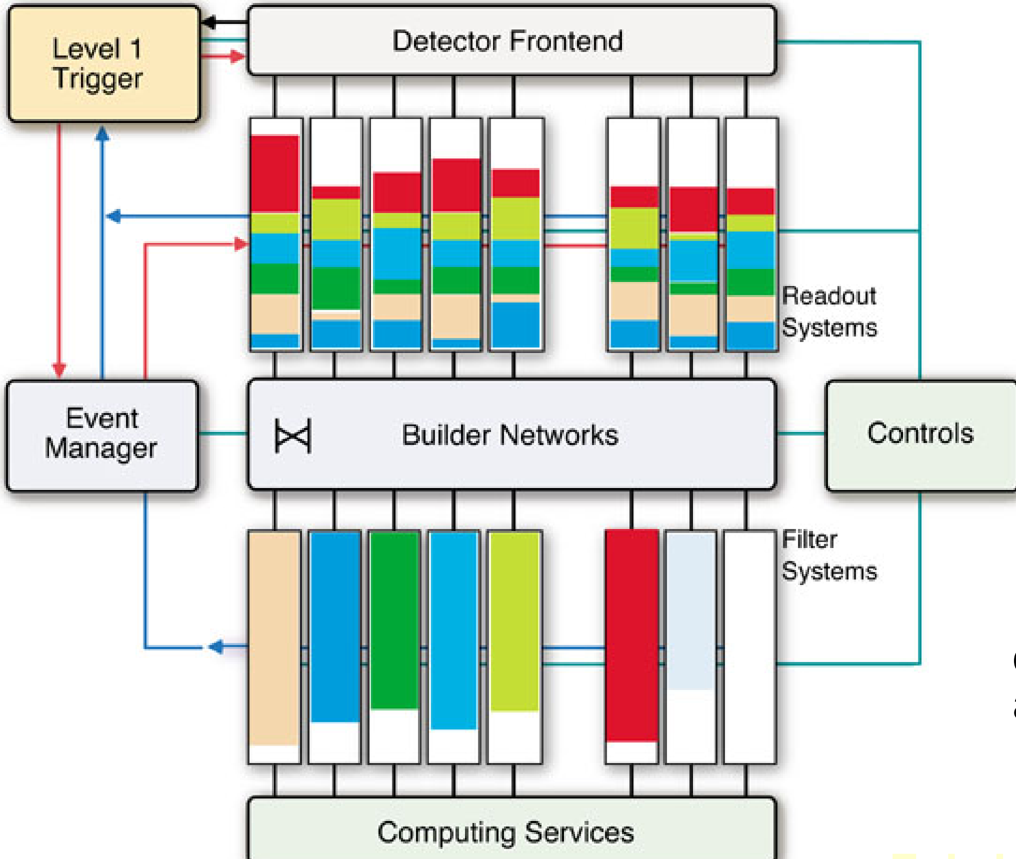
Readout Units/Drivers/Buffers

- **Currently, dual-ported data access**
  - Additional ports for control
  - DAQ element with lowest latency (~μs), highest rate
  - Basic tasks:
    - Merge data from N front-ends
    - Send data onto processor farm
    - Store the data until no longer needed (data sent or event rejected)
  - Issues:
    - Input interconnect (bus/point-to-point link/switch)
    - Output interconnect (bus/point-to-point link/switch)
    - Sustained bandwidth requirement (200-800 MB/s)

Event Building

Event Building

- **Form full-event-data buffers from fragments in the readout. Must interconnect data sources/destinations.**



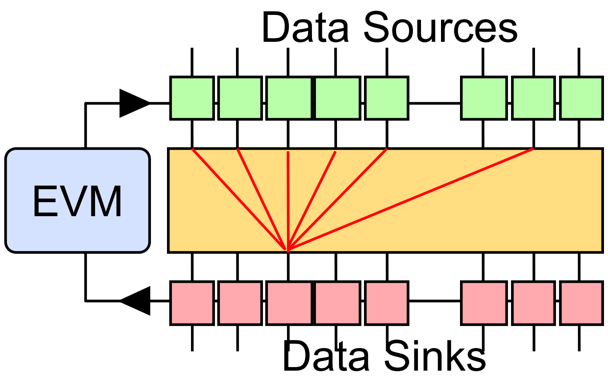
Event fragments :
Event data fragments are stored in separated physical memory systems

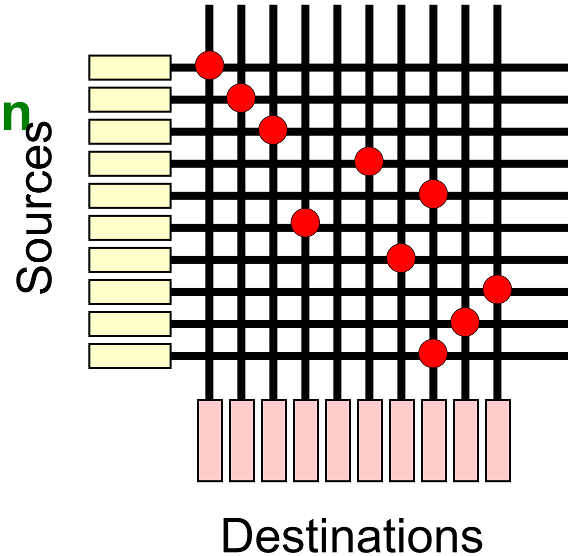
Full events :
Full event data are stored into one physical memory system associated to a processing unit

Hardware:
Fabric of switches for builder networks
PC motherboards for data Source/Destination nodes

Event Building via a Switch

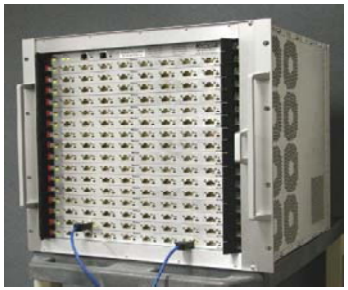
- **Three major issues:**
  - **Link utilization**
  - **The bottleneck on the outputs**
  - **The large number of ports needed**
- **Space-division: crossbar**
  - **Simultaneous transfers between any arbitrary set of inputs and outputs**
    - **Can be both self-routing and arbiter-based (determine connectivity between S's and D's for each cycle); the faster the fabric, the smaller the arbitration complexity**
    - **Does not solve Output Contention issue**
    - **Need *Traffic Shaping***



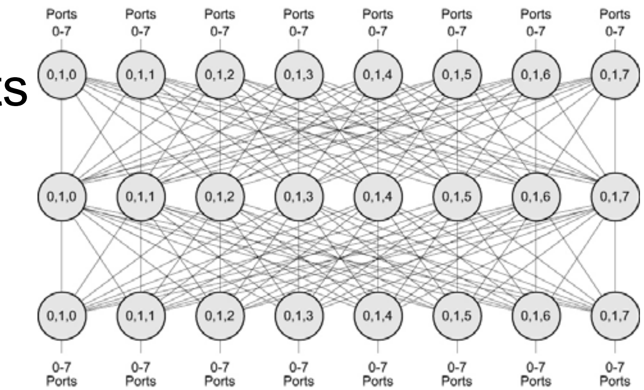


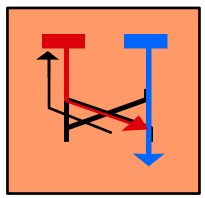
Switching technologies

Myricom: Myrinet 2000



- Switch: **Clos-128 @ 2.5** Gb/s ports
- **NIC**: M3S-PCI64B-2 (**LANai9**)
- **Custom Firmware**

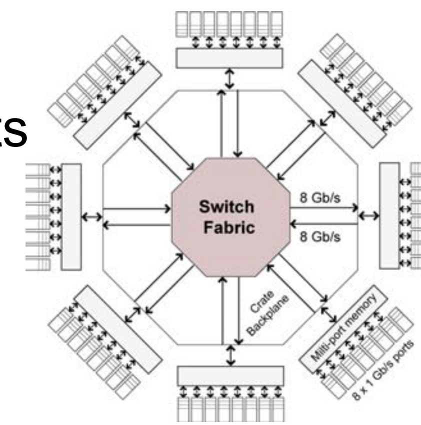


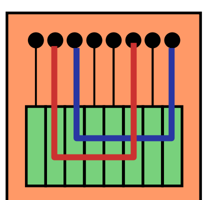


wormhole data
transport with flow
control at all stages

Gigabit Ethernet

- Switch: Foundry **FastIron64 @ 1.2** Gb/s ports
- **NIC**: **Alteon** (running standard firmware)





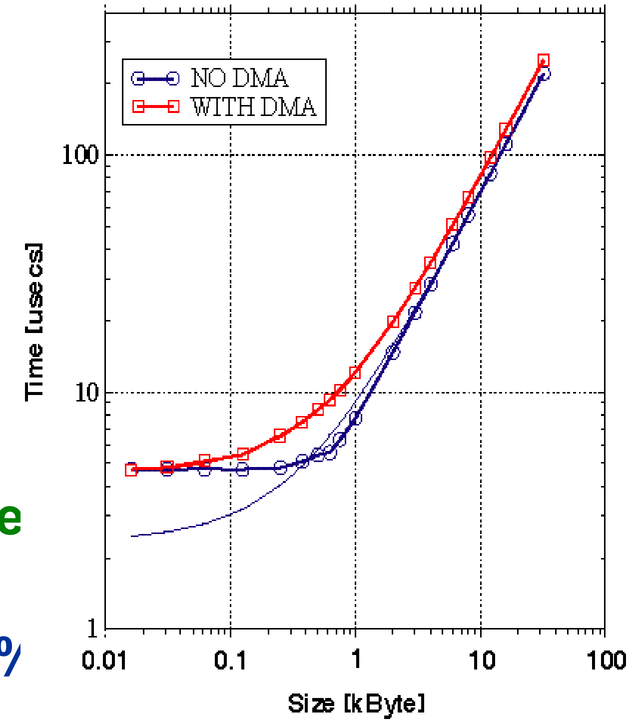
Implementation:
Multi-port memory system R/W bandwidth greater than sum of all port speeds
Packet switching
Contention resolved by Output buffer.
Packets can be lost.

Infiniband

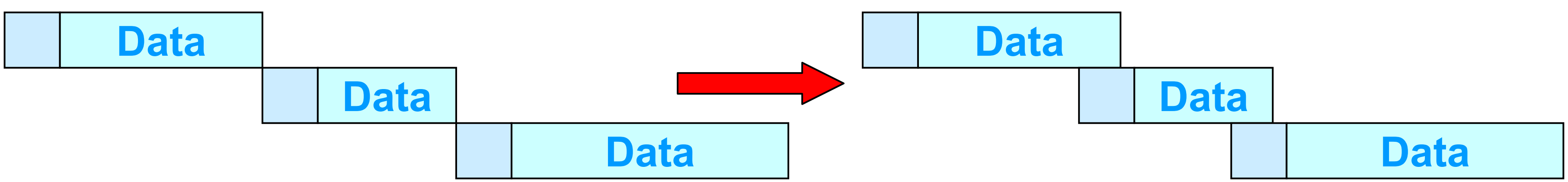
- 2.5 Gb/s demo products. First tests completed recently.

Link utilization

- **Fit transfer time vs s(ize)**
  - **Clearly, $T = T_0 + s/V_{max}$**
  - **Example: extract T_0 and V_{max}**
    - **$T_0 = 1\mu s$**
    - **$V_{max} = 140$ MB/s**
  - **But plateau at $5\mu s$**
    - **Full overhead (including software setup etc)**
  - **Overall link utilization efficiency: 92%**

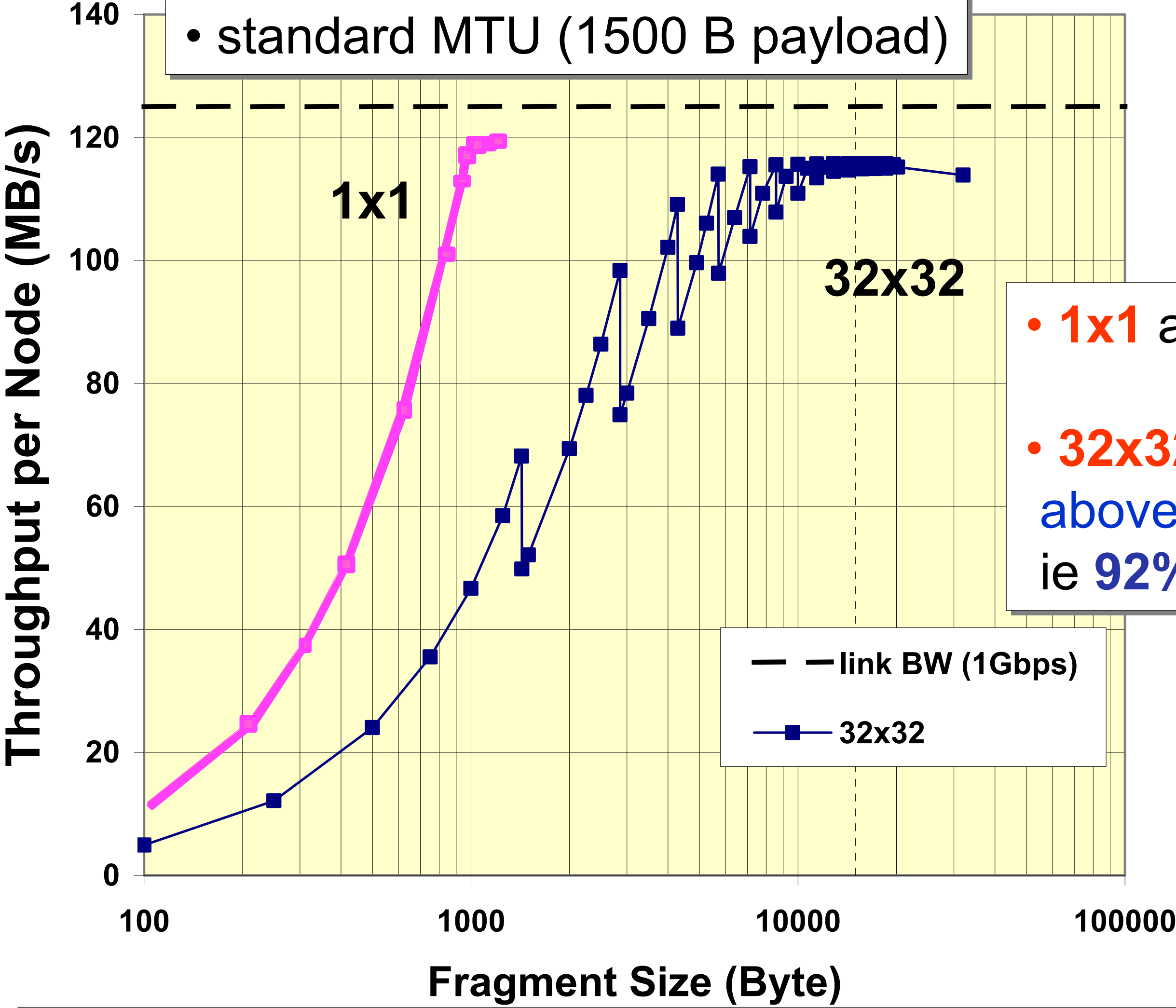


- **Special I/O drivers to overlap the overhead operations with the actual data transfer**



Gigabit Ethernet-based 32x32 EVB

- Alteon AceNIC
- standard MTU (1500 B payload)



- **1x1** asymptotically to 125 MB/s
- **32x32** saw tooth due to MTU

  above **10k**: plateau of 115 MB/s

  ie **92%** of link speed (1Gbps)

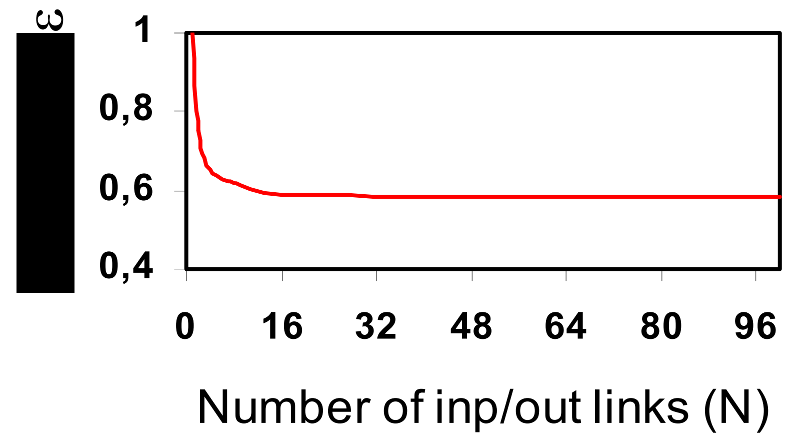
Next issue: clash on output link (traffic-shaping)

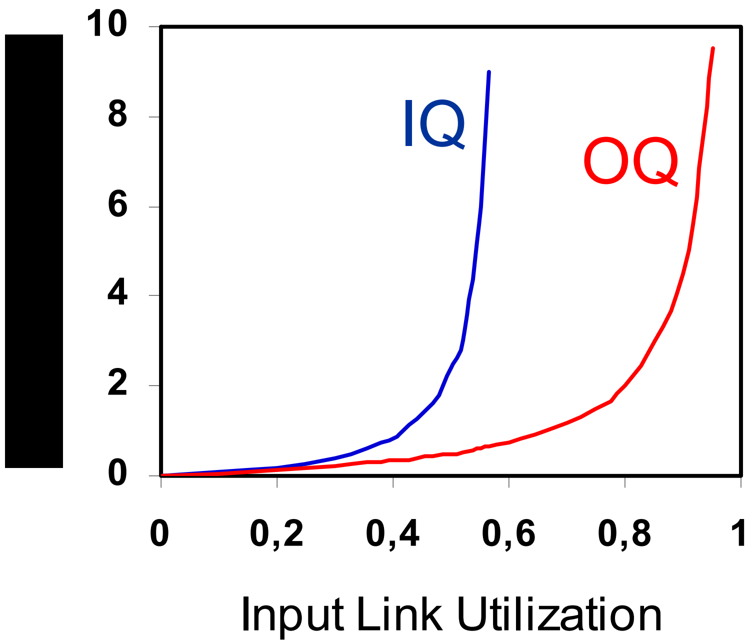
Performance of IQ/OQ switches

- **IQ switches, random traffic:**

$$\varepsilon = 2 - \sqrt{2} \approx 0.59 \ \text{ for } N \to \infty$$

M.J.Karol, M.G.Hluchyj and S.P.Morgan, "Input vs Output Switching on a Space Division Packet Switch", IEEE Trans. Commun., vol. 2, pp. 277-287, 1989.

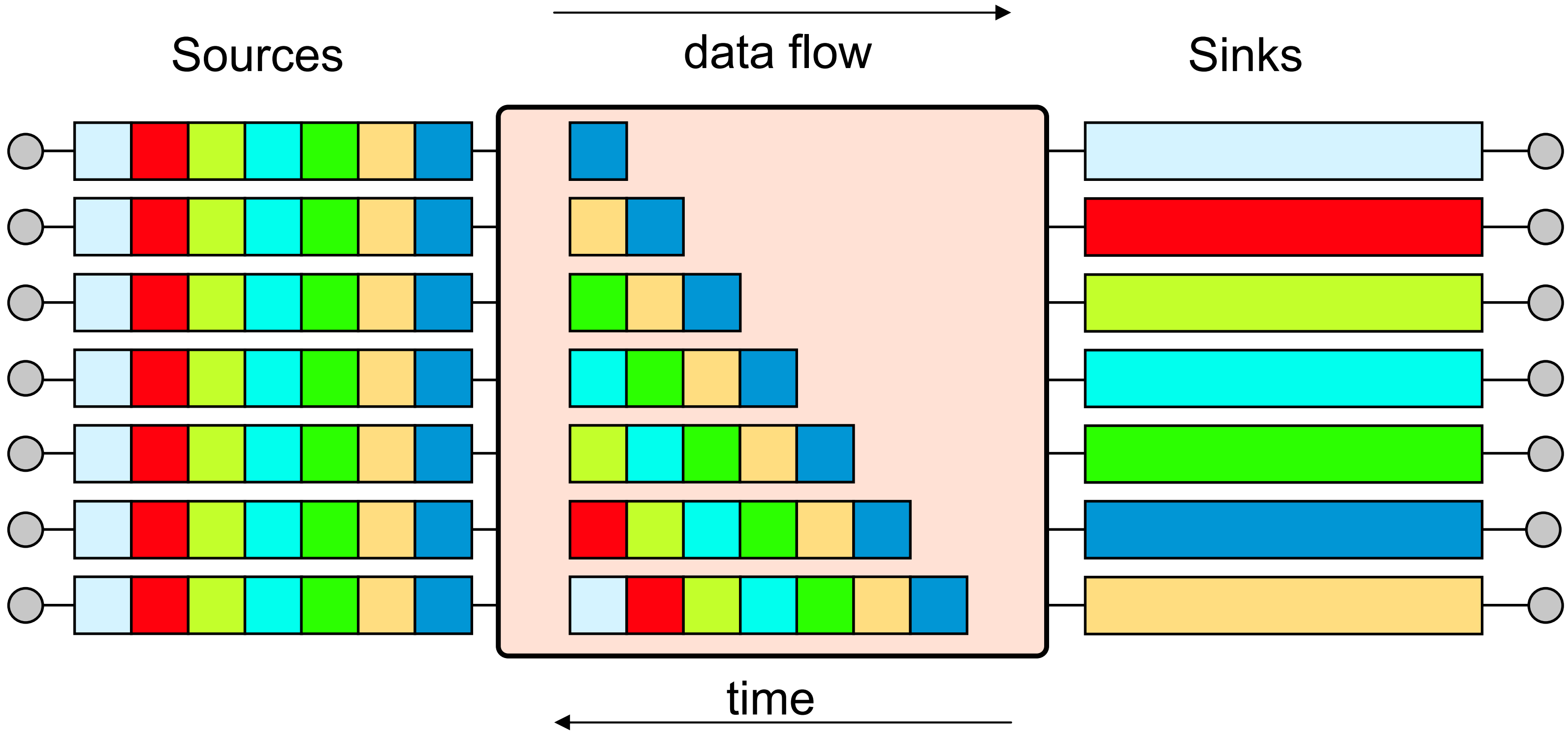




- **Best performance: OQ**
  - **Bandwidth of the memory used for the output FIFOs becomes prohibitively large**
    (write-access to FIFOs is N times faster than the input link speeds)

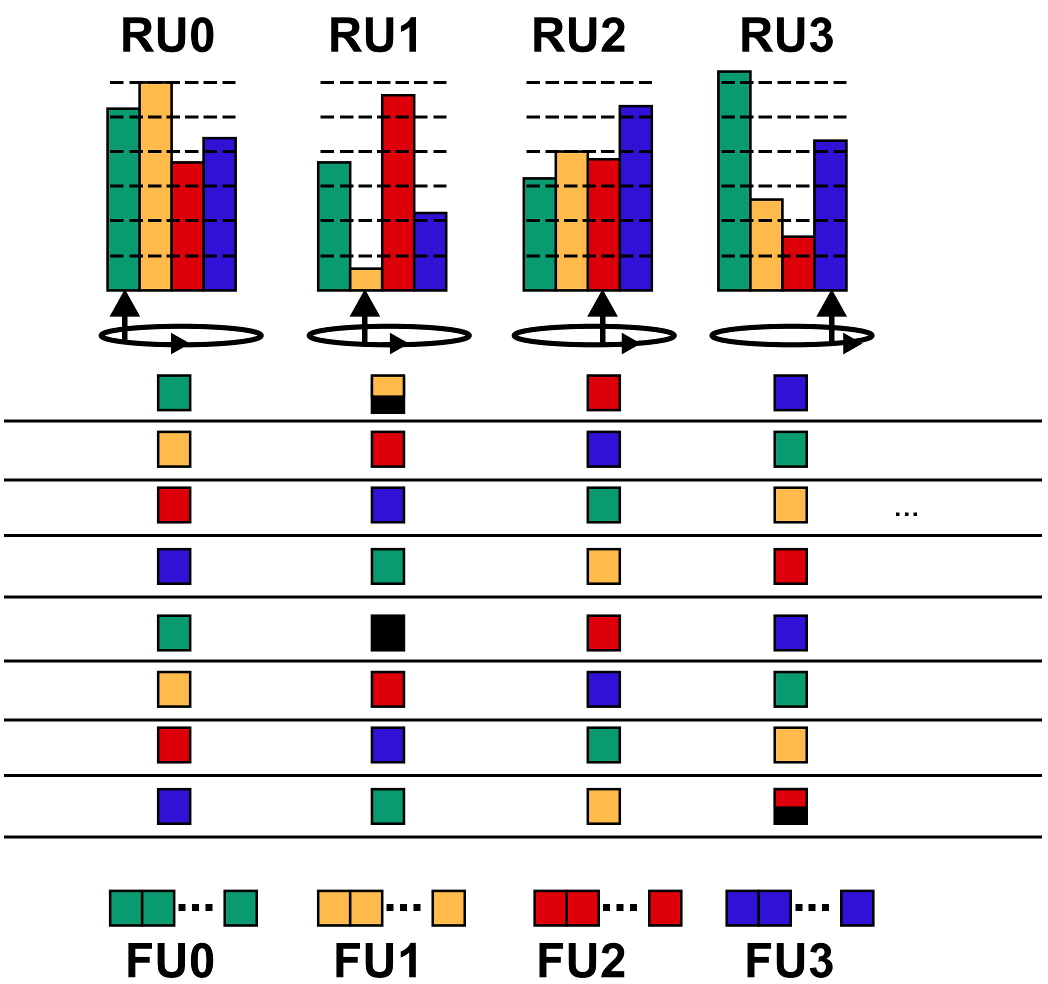
EVB traffic shaping: barrel shifter

- **Barrel-shifter: principle**

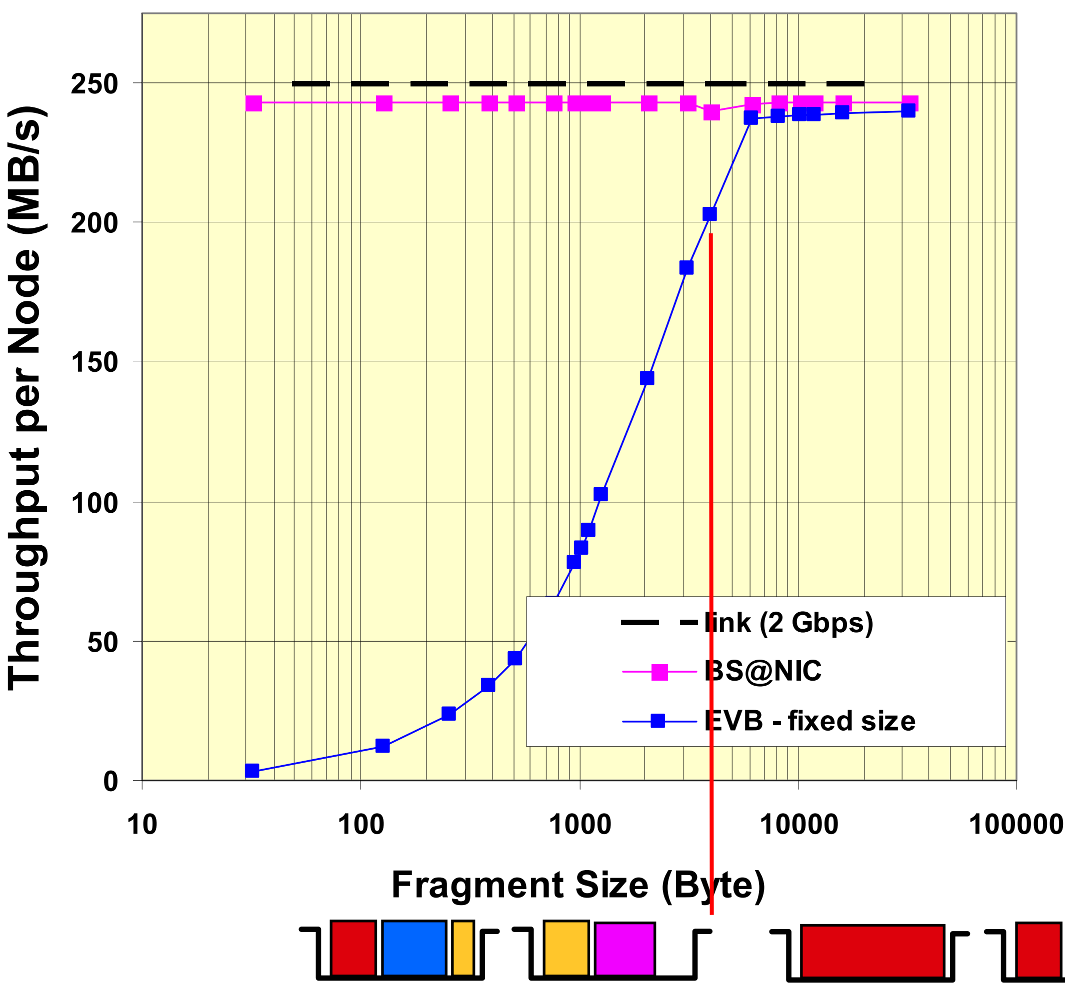


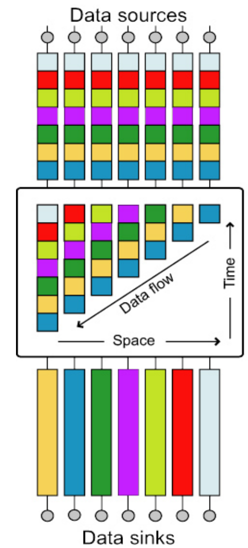
Barrel-shifting with variable-size events

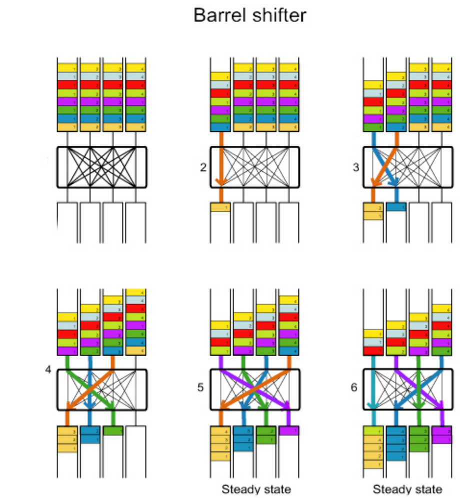
- **Demonstrator**
  - Fixed-block-size with barrel-shifter
  - Basic idea taken from ATM (and time-division-muxing)
  - As seen in composite-switch analysis, this should work for large N as well
  - Currently testing on 64x64… (originally: used simulation for N≈500; now ~obsolete)



A Myrinet-based 32x32 EVB

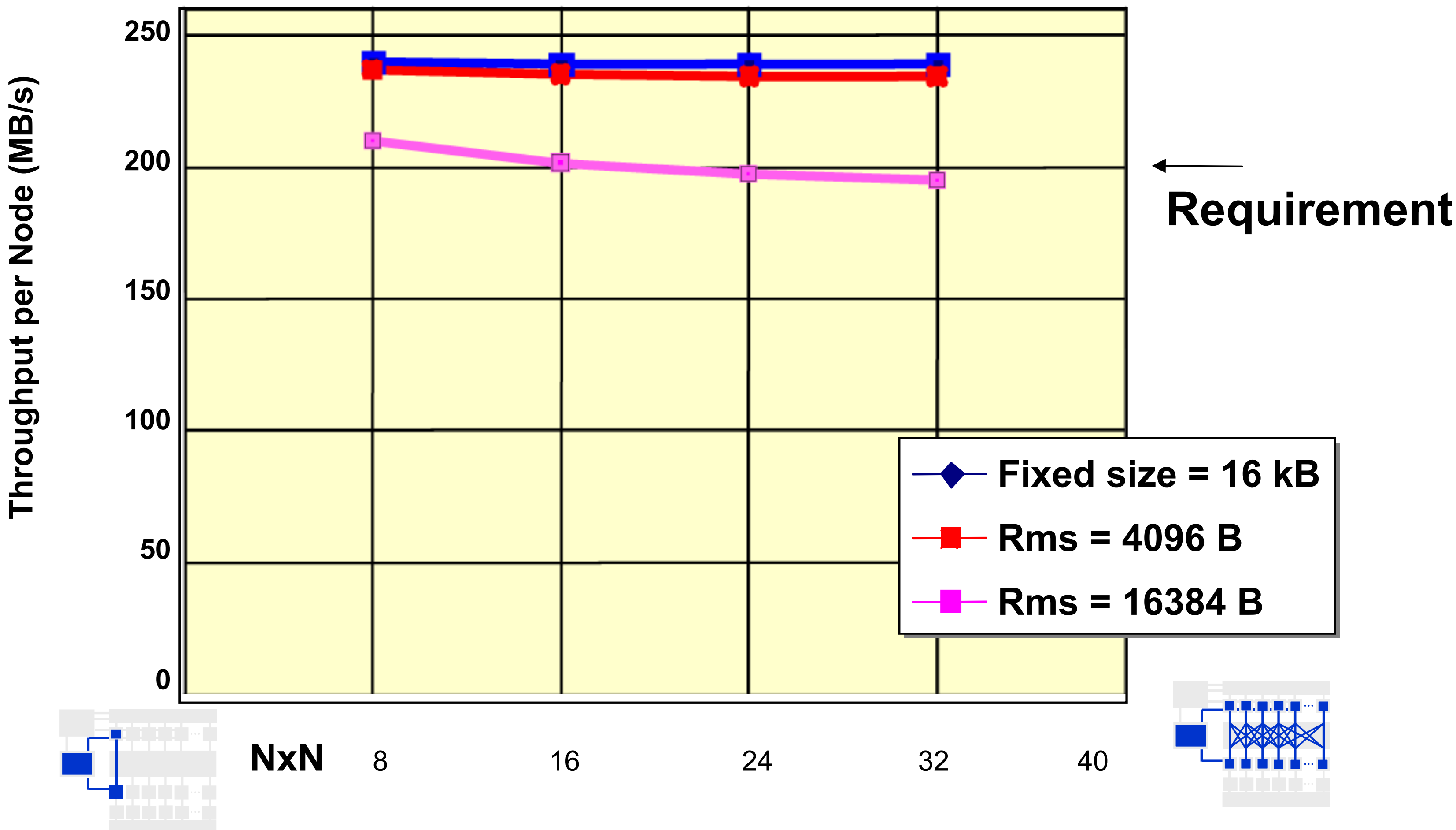






- Fixed-size event fragments
  below 4k: Fragment < BS carrier
  above 4k: Fragment > BS carrier
- Throughput at **234 MB/s**
  = **94% of link Bandwidth**

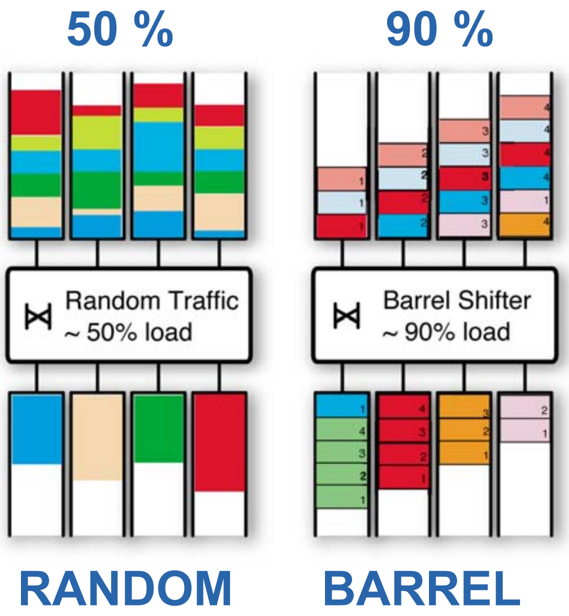
Barrel-shifter scaling: Myrinet



From 8x8 to 32x32: Scaling observed (as expected from barrel shifter)

EVB summary

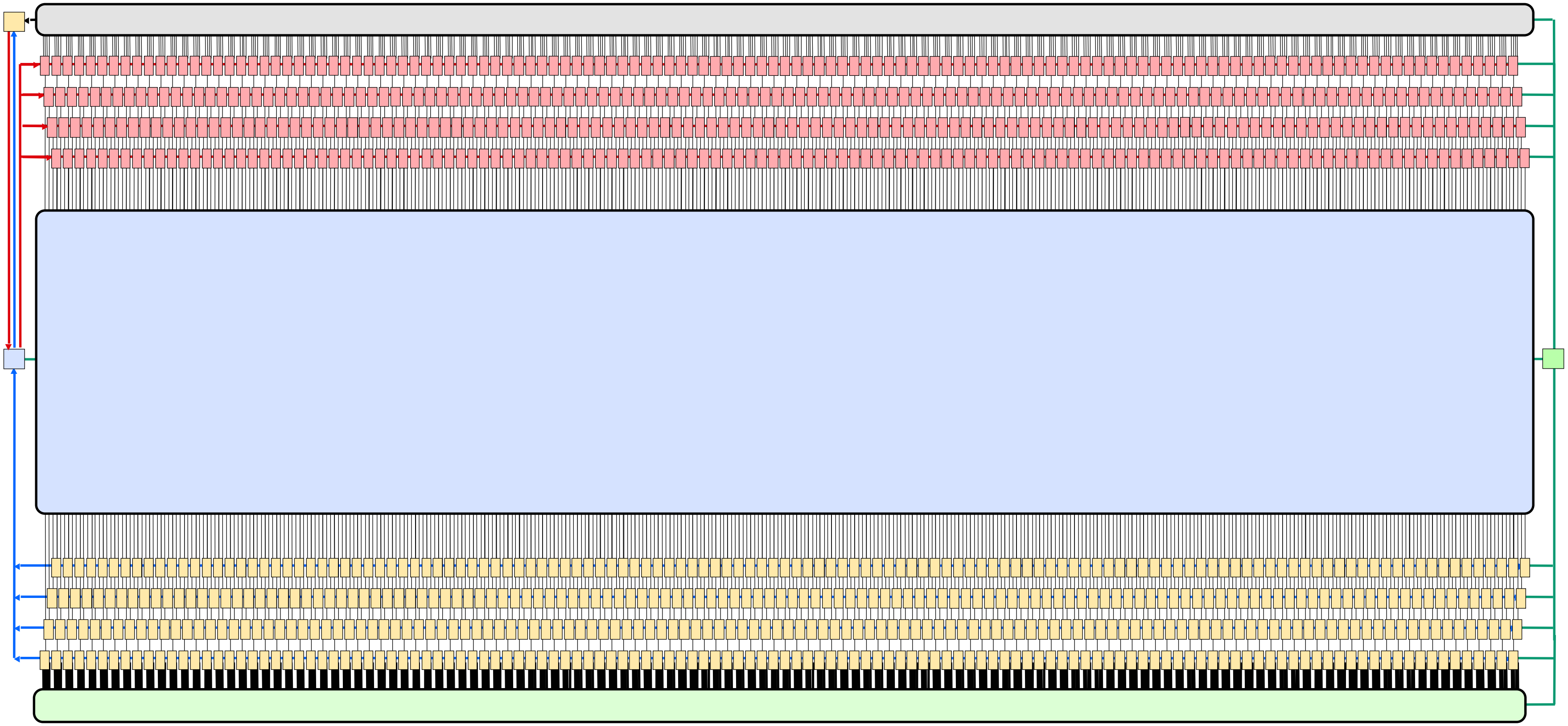
- **EVB traffic pattern “special”; need “traffic-shaping”**
  - Two limits to this:
    - Random traffic: need switch with factor 2 more bandwidth than throughput needed
    - Barrel: can work with ~90% efficiency
  - Clear demonstration at 32x32
    - Larger systems (e.g. ALICE) have also been demonstrated, but not at near-100% loads
      - ➔ They serve as demonstrations of all the software and system aspects involved in the system



Control and Monitor

Control & Monitor (I)

- **Unprecedented scale; example: 1000 interconencted units**

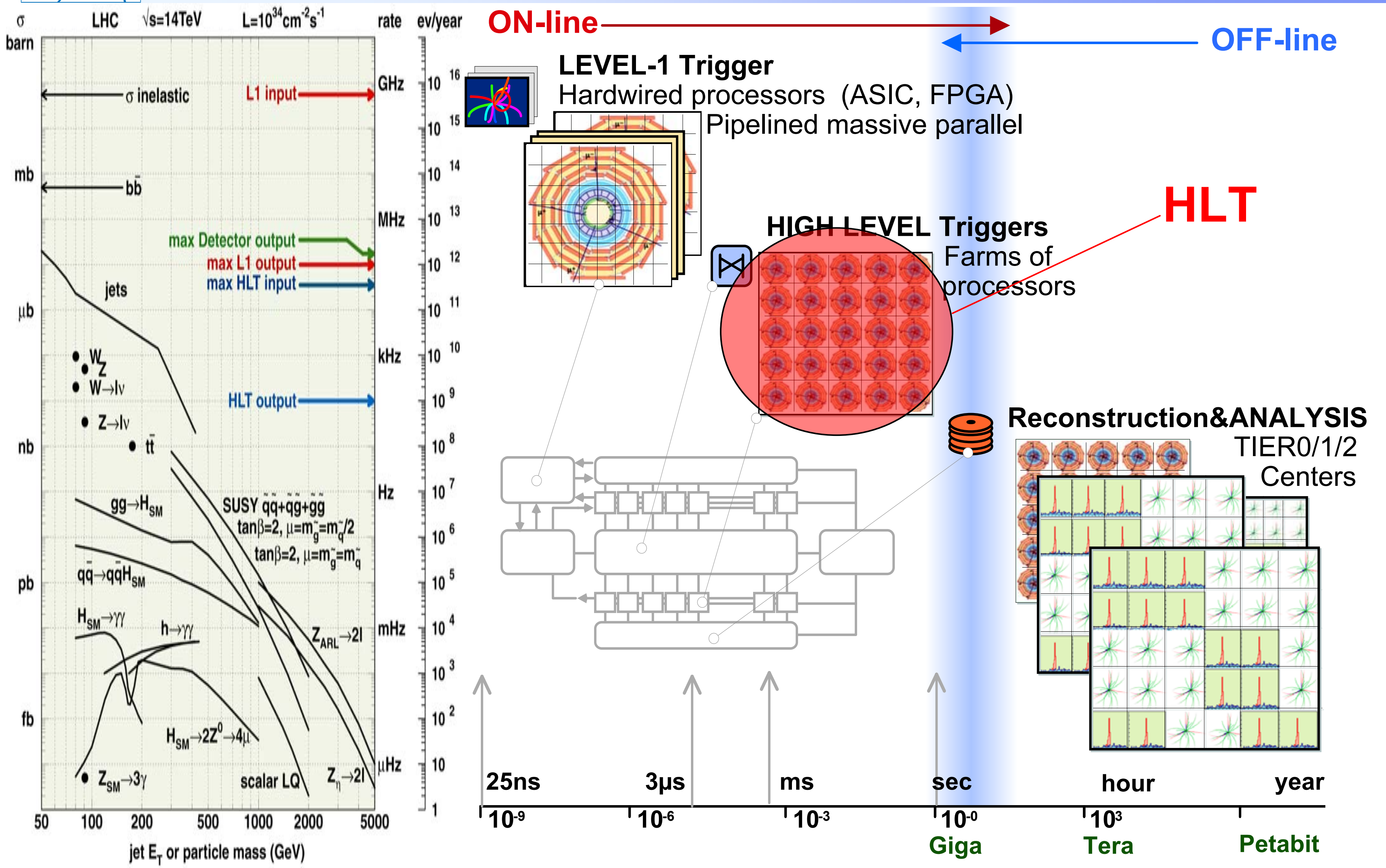


Control & Monitor (II)

- **Challenges:**
  - Large N (on everything)
  - Disparity in time scales (μs–s; from readout to filtering)
  - Need to use standards for
    - Communication (Corba? Too heavy? Right thing? SOAP!)
    - User Interface (is it the Web? Yes…)
  - Physics monitoring complicated by factor 500 (number of sub-farms);
    - Need merging of information; identification of technical, one-time problems vs detector problems
- **Current work:**
  - Create toolkits from commercial software (SOAP, XML, HTTP etc); integrate into packages, build "Run Control" on top of it;
- **Detector Control System: DCS. All of this for the ~10^7 channels… SCADA (commercial, standard) solutions**

High-Level Trigger

Physics selection at the LHC



Branches

1. **Throughput of ~32 Gb/s is enough (ALICE)**
   - ALICE needs 2.5 GB/s of "final EVB"
   - Then proceed no further; software, control and monitor, and all issues of very large events (storage very important)
2. **Need more bandwidth, but not much more (e.g. LHCb; event size ~100 kB @ 40 kHz = 4 GB/s = 32 Gb/s)**
   - Implement additional capacity
3. **Need much more than this; CMS+ATLAS need 100 GB/s = 800Gb/s**
   - Two solutions:
     - Decrease rate by using a Level-2 farm (ATLAS)
       - ➔ Thus, two farms: a Level-2 and Level-3 farm
     - Build a system that can do 800 Gb/s (CMS)
       - ➔ Thus, a single farm

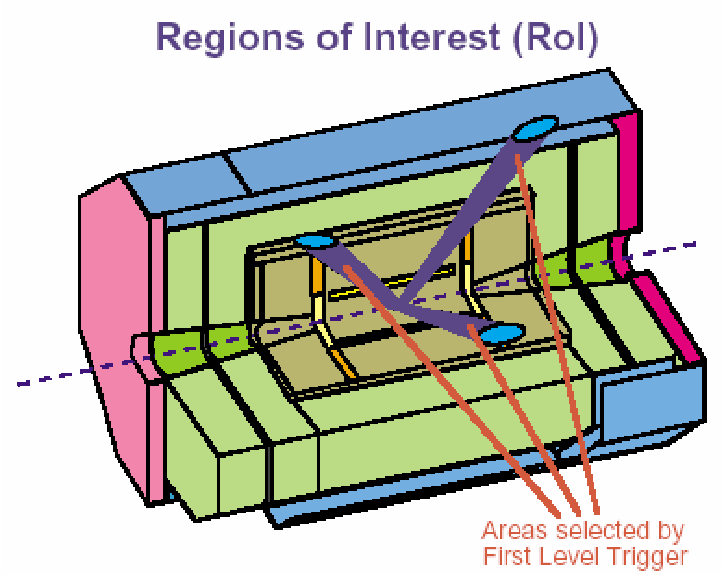
100 GB/s case: Level-2/Level-3 vs HLT

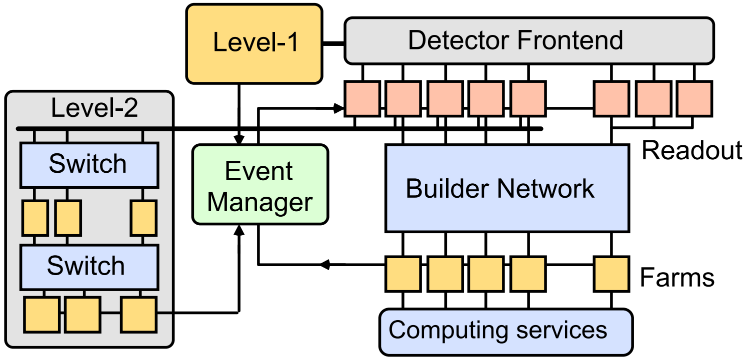
- **Level-2 (ATLAS):**
  - Region of Interest (ROI) data are ~1% of total
  - Smaller switching network is needed (not in # of ports but in throughput)
  - But adds:
    - Level-2 farm
    - "ROB" units (have to "build" the ROIs)
    - Lots of control and synchronization
  - Problem of large network → problem of Level-2

- **Combined HLT (CMS):**
  - Needs very high throughput
  - Needs large switching network
  - But it is also:
    - Simpler (in data flow and in operations)
    - More flexible (the entire event is available to the HLT – not just a piece of it)
  - Problem of selection → problem of technology

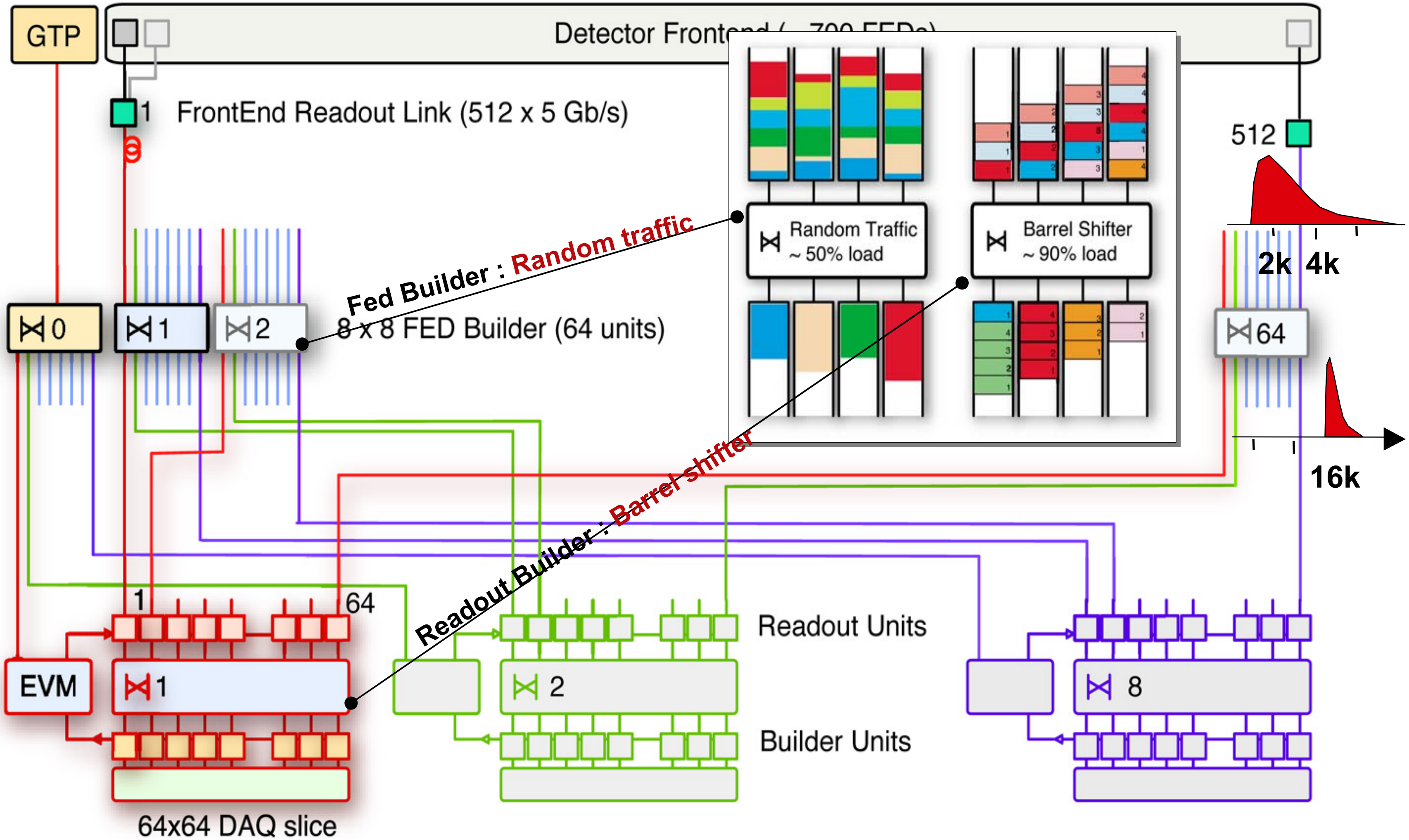
ATLAS: from demonstrator to full EVB

- **With Regions of Interest:**
  - If the Level-2 delivers a factor 100 rejection, then input to Level-3 is 1-2 kHz.
  - At an event size of 1-2 MB, this needs 1-4 GB/s
    - An ALICE-like case in terms of throughput
    - Dividing this into ~100 receivers implies 10-40 MB/s sustained – certainly doable
  - Elements needed: ROIBuilder, L2PU (processing unit),





Detector readout & 3D-EVB



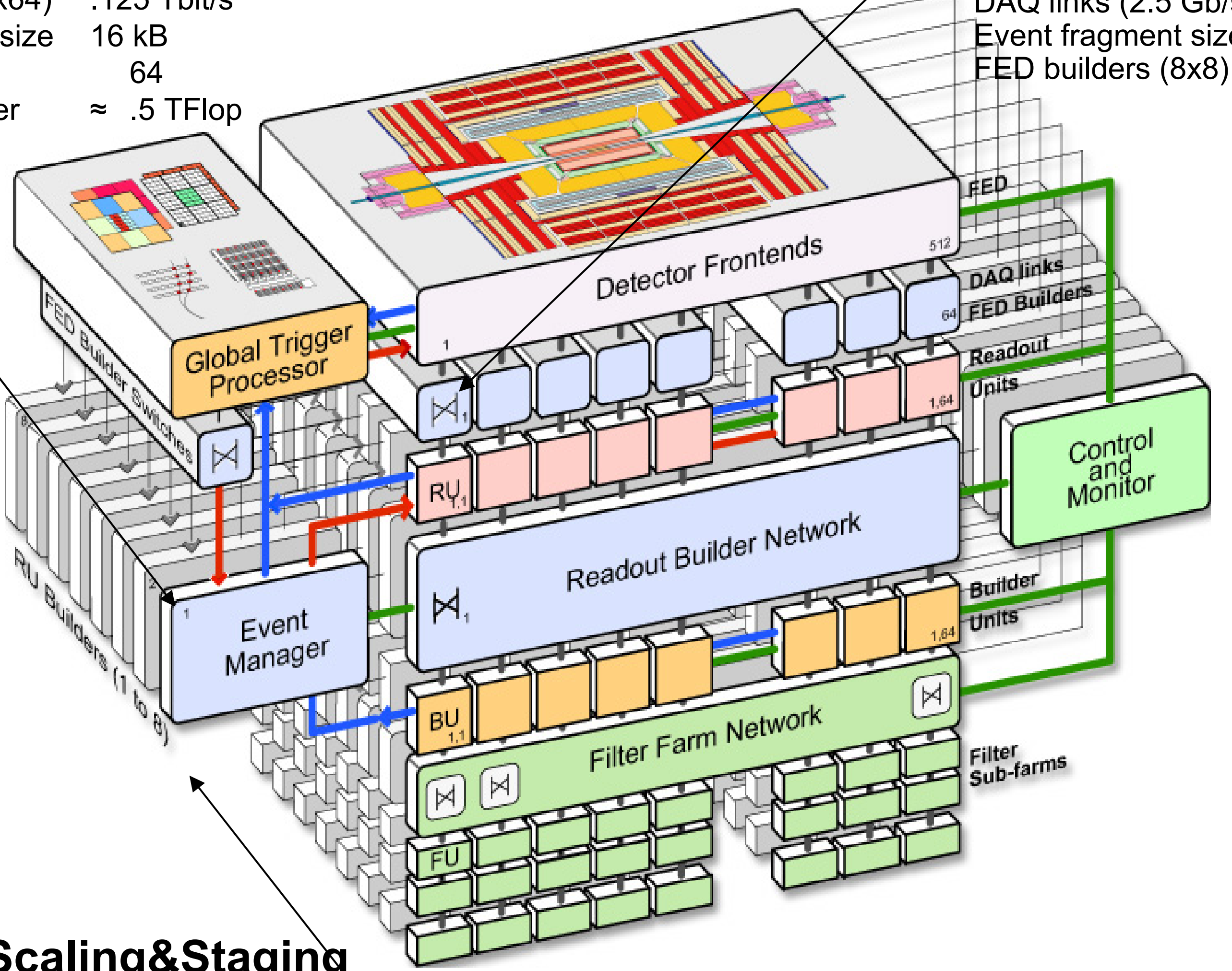
3D-EVB: DAQ staging and scaling

DAQ unit (1/8th full system):

Lv-1 max. trigger rate	12.5 kHz
RU Builder (64x64)	.125 Tbit/s
Event fragment size	16 kB
RU/BU systems	64
Event filter power	≈ .5 TFlop

Data to surface:

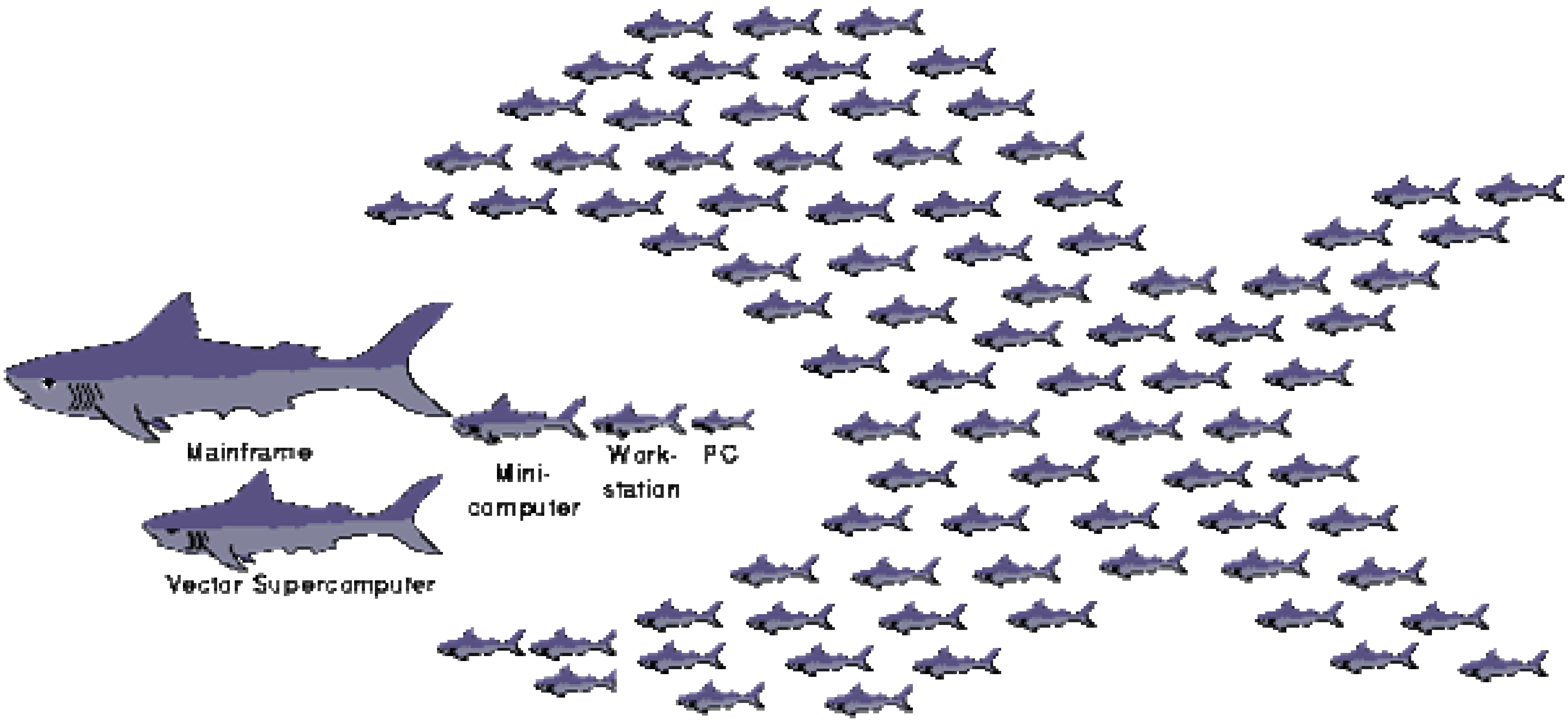
Average event size	1 Mbyte
No. FED s-link64 ports	> 512
DAQ links (2.5 Gb/s)	512+512
Event fragment size	2 kB
FED builders (8x8)	≈ 64+64



DAQ Scaling&Staging

Filter Farm

Processor Farm: the 90's super-computer; the 2000's large computer



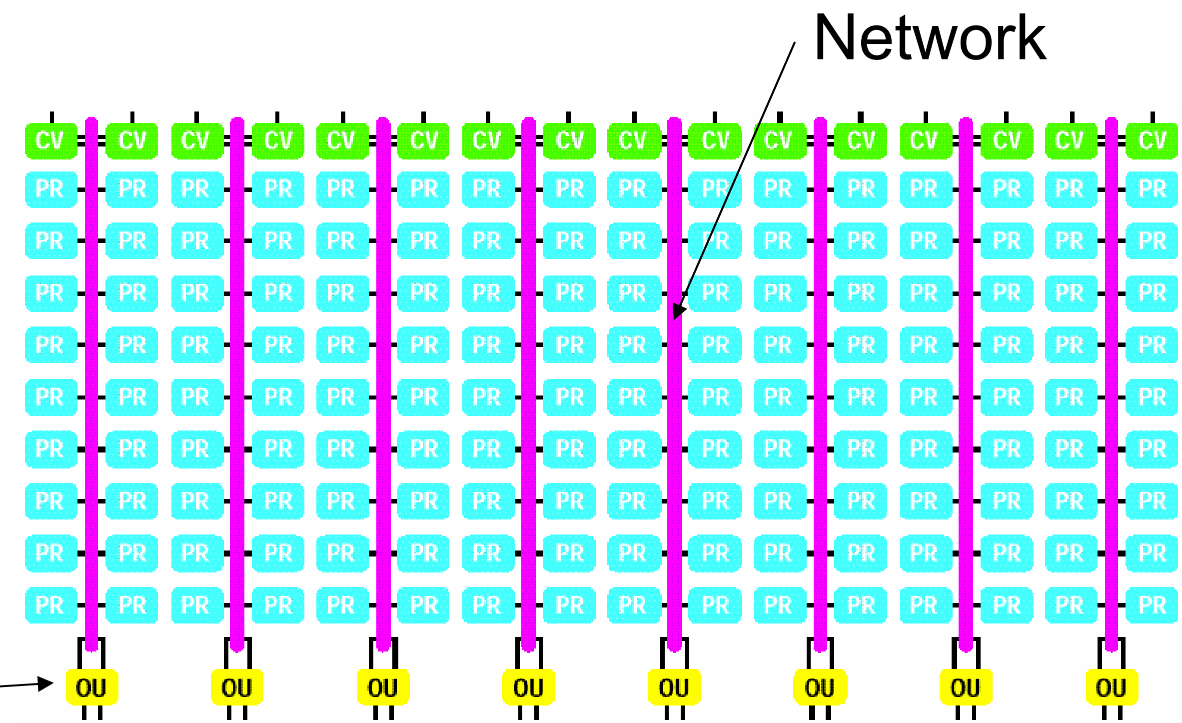
NOW

Found at the NOW project (http://now.cs.berkeley.edu)

Processor Engines

- **Final stage of the filtering process: almost an offline-quality reconstruction & selection**
  - **Need real programmable processors; and lots of them**
  - **(Almost) all experiments in HEP: using/will use a processor farm**

Input



Output

CDF example from Run II start

Processor Engine (II)

- **PC+Linux: the new supercomputer for scientific applications**

obswww.unige.ch/~pfennige/gravitor/gravitor_e.html

www.cs.sandia.gov/cplant/

Processor Farms: summary

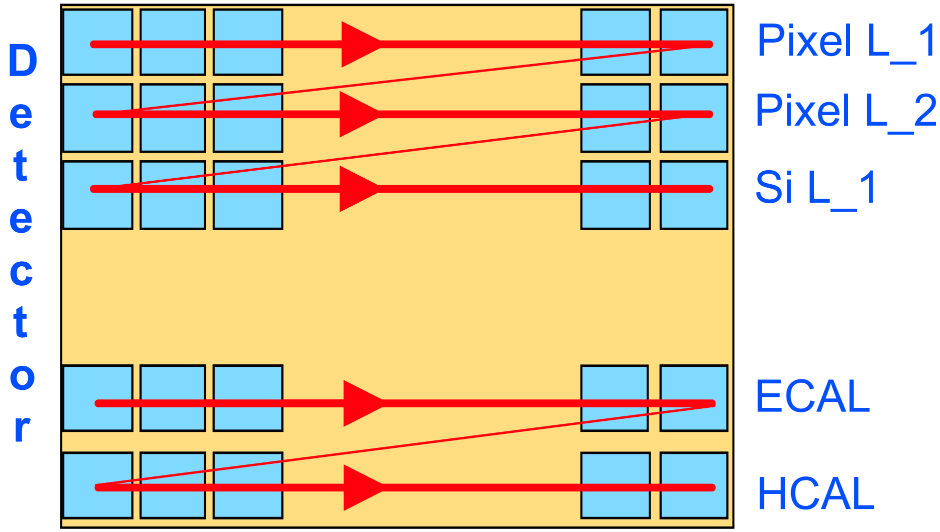
- **Explosion of number of farms installed**
  - **Very cost-effective**
    - **Linux is free but also very stable, production-quality**
    - **Interconnect: Ethernet, Myrinet (if more demanding I/O); both technologies inexpensive and performant**
  - **Large number of message-passing packages, various API's on the market**
    - **Use of a standard (VIA?) could be the last remaining tool to be used on this front**
  - **Despite recent growth, it's a mature process: basic elements (PC, Linux, Network) are all mature technologies. Problem solved. What's left: Control & Monitor.**
    - **Lots of prototypes and ideas. Need real-life experience.**
      - **➔ Problem is human interaction**

HLT algorithms and performance

HLT requirements and operation

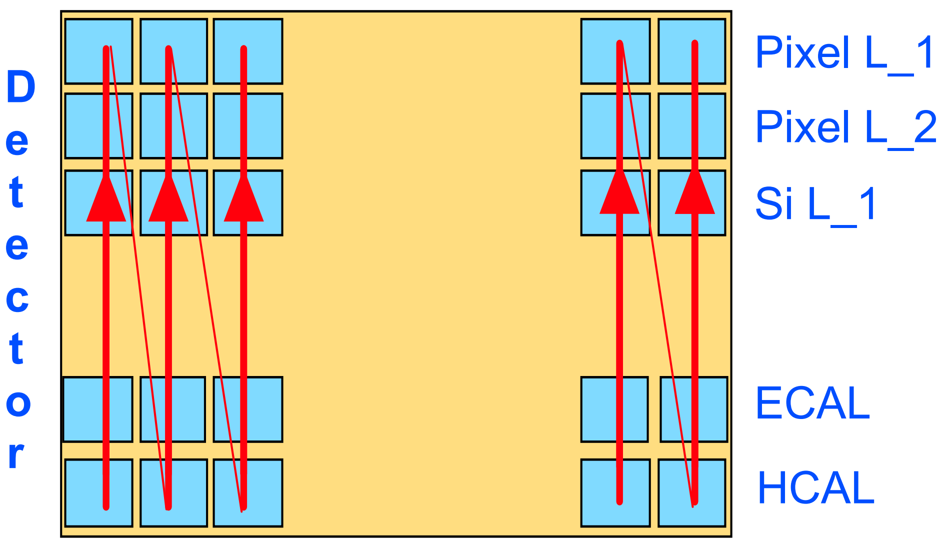
- **Strategy/design guidelines**
  - Use offline software as much as possible
    - Ease of maintenance, but also understanding of the detector
- **Boundary conditions:**
  - Code runs in a single processor, which analyzes one event at a time
  - HLT (or Level-3) has access to full event data (full granularity and resolution)
  - Only limitations:
    - CPU time
    - Output selection rate (~10^2 Hz)
    - Precision of calibration constants
- **Main requirements:**
  - Satisfy physics program (see later): high efficiency
  - Selection must be inclusive (to discover the unpredicted as well)
  - Must not require precise knowledge of calibration/run conditions
  - Efficiency must be measurable from data alone
  - All algorithms/processors must be monitored closely

HLT (regional) reconstruction (I)



Global

- process (e.g. DIGI to RHITs) each detector fully
- then link detectors
- then make physics objects

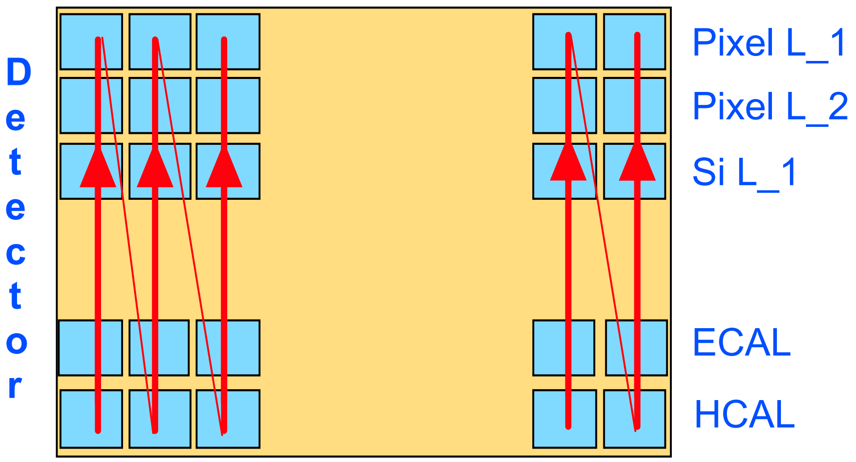


Regional

- process (e.g. DIGI to RHITs) each detector on a "need" basis
- link detectors as one goes along
- physics objects: same

HLT (regional) reconstruction (II)

- **For this to work:**
  - Need to know where to start reconstruction (seed)
- **For this to be useful:**
  - Slices must be narrow
  - Slices must be few
- **Seeds from Lvl-1:**
  - e/γ triggers: ECAL
  - μ triggers: μ sys
  - Jet triggers: E/H-CAL



- **Seeds ≈ absent:**
  - Other side of lepton
  - Global tracking
  - Global objects (Sum E_T, Missing E_T)

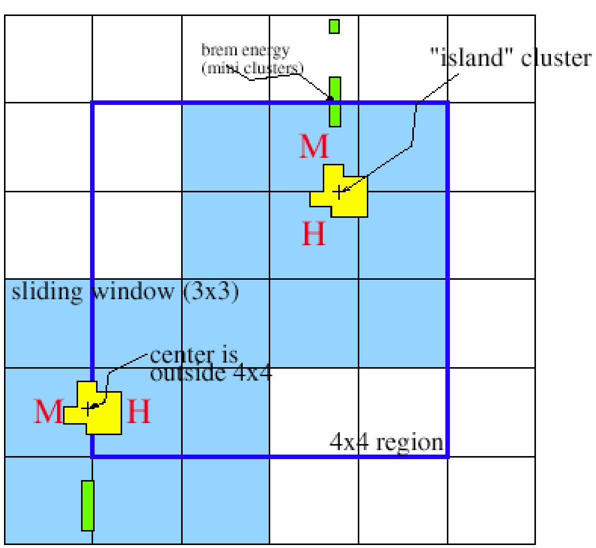
Example: electron selection (I)

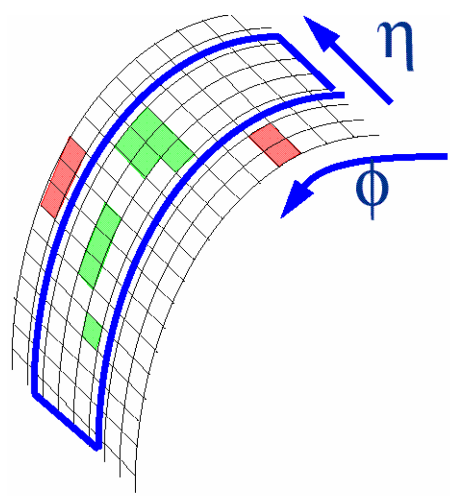
- **"Level-2" electron:**
  - **1-tower margin around 4x4 area found by Lvl-1 trigger**
  - **Apply "clustering"**
  - **Accept clusters if H/EM < 0.05**
  - **Select highest E_T cluster**

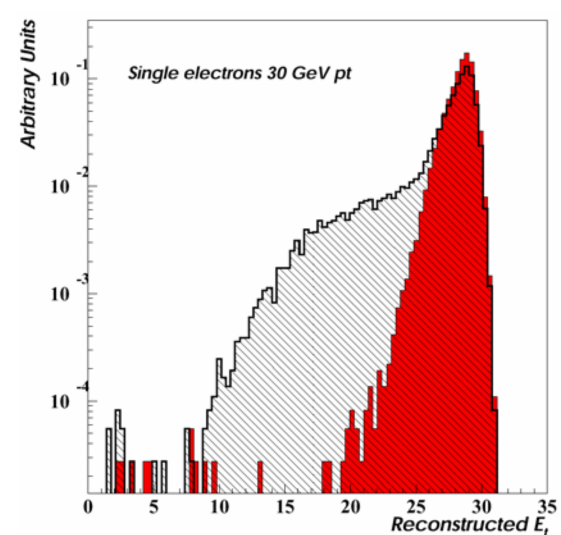
- **Brem recovery:**
  - **Seed cluster with $E_T > E_T^{min}$**
  - **Road in ϕ around seed**
  - **Collect all clusters in road**

  → "supercluster"

  and add all energy in road:

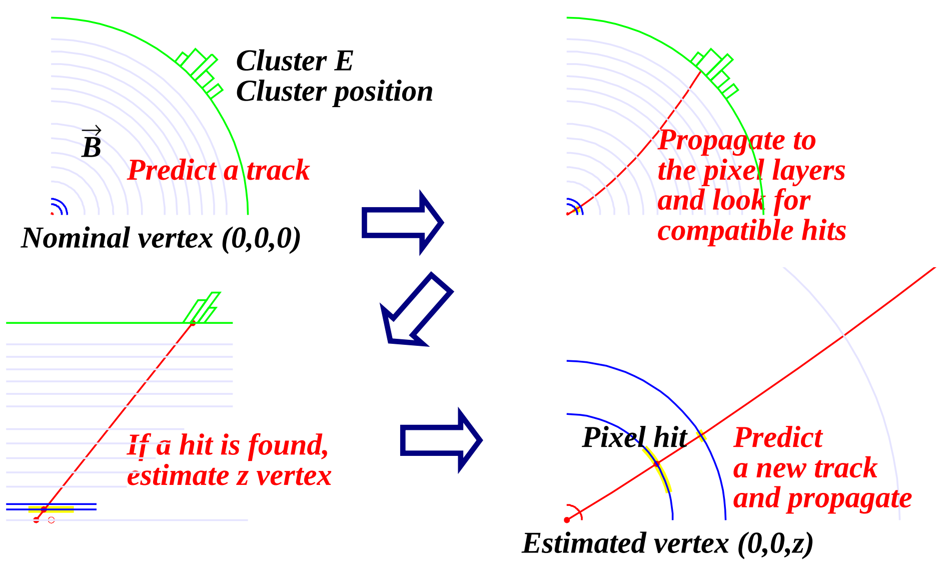


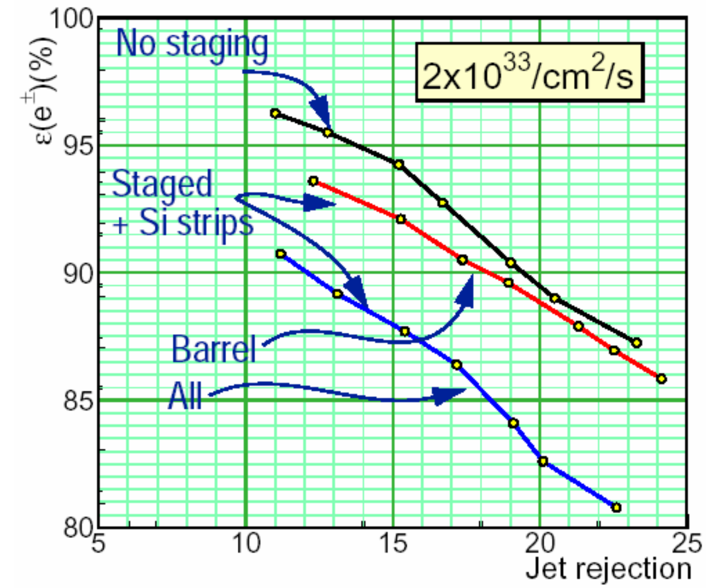




Example: electron selection (II)

- **"Level-2.5" selection: add pixel information**
  - **Very fast, high rejection (e.g. factor 14), high efficiency (ε=95%)**
    - **Pre-bremsstrahlung**
    - **If # of potential hits is 3, then demanding ≥ 2 hits quite efficient**

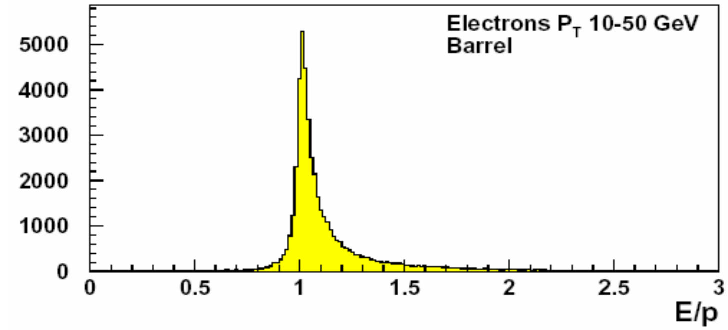


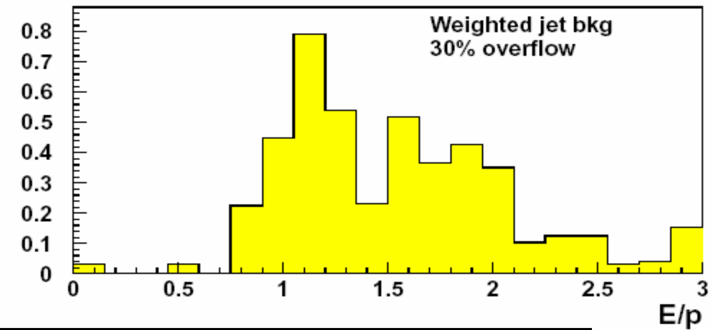


No staging: 3 cylinders + 2 disks
Staged: 2 cylinders + 1 disk

Example: electron selection (III)

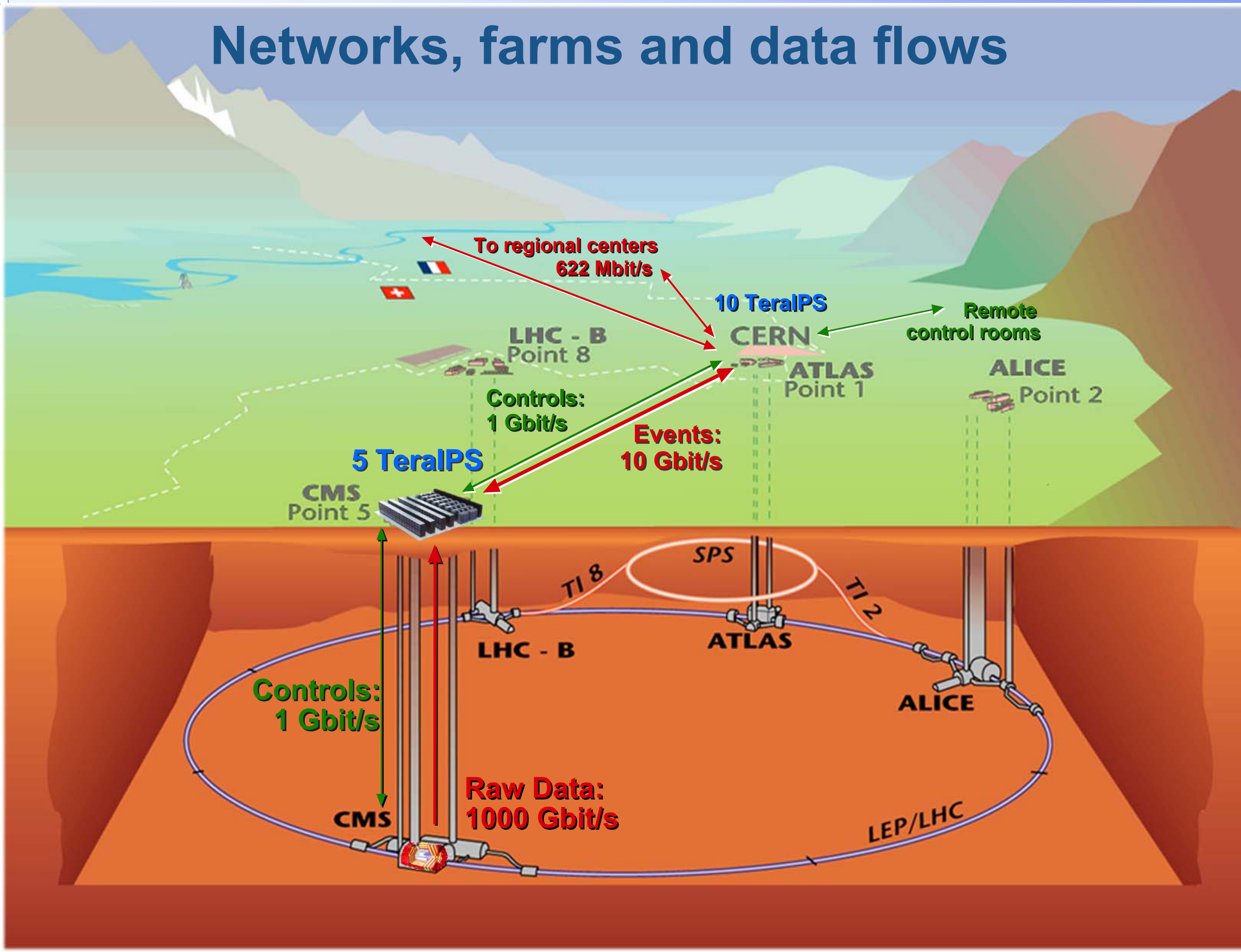
- **"Level-3" selection**
  - Full tracking, loose track-finding (to maintain high efficiency):
  - Cut on E/p everywhere, plus
    - Matching in η (barrel)
    - H/E (endcap)
  - Optional handle (used for photons): isolation



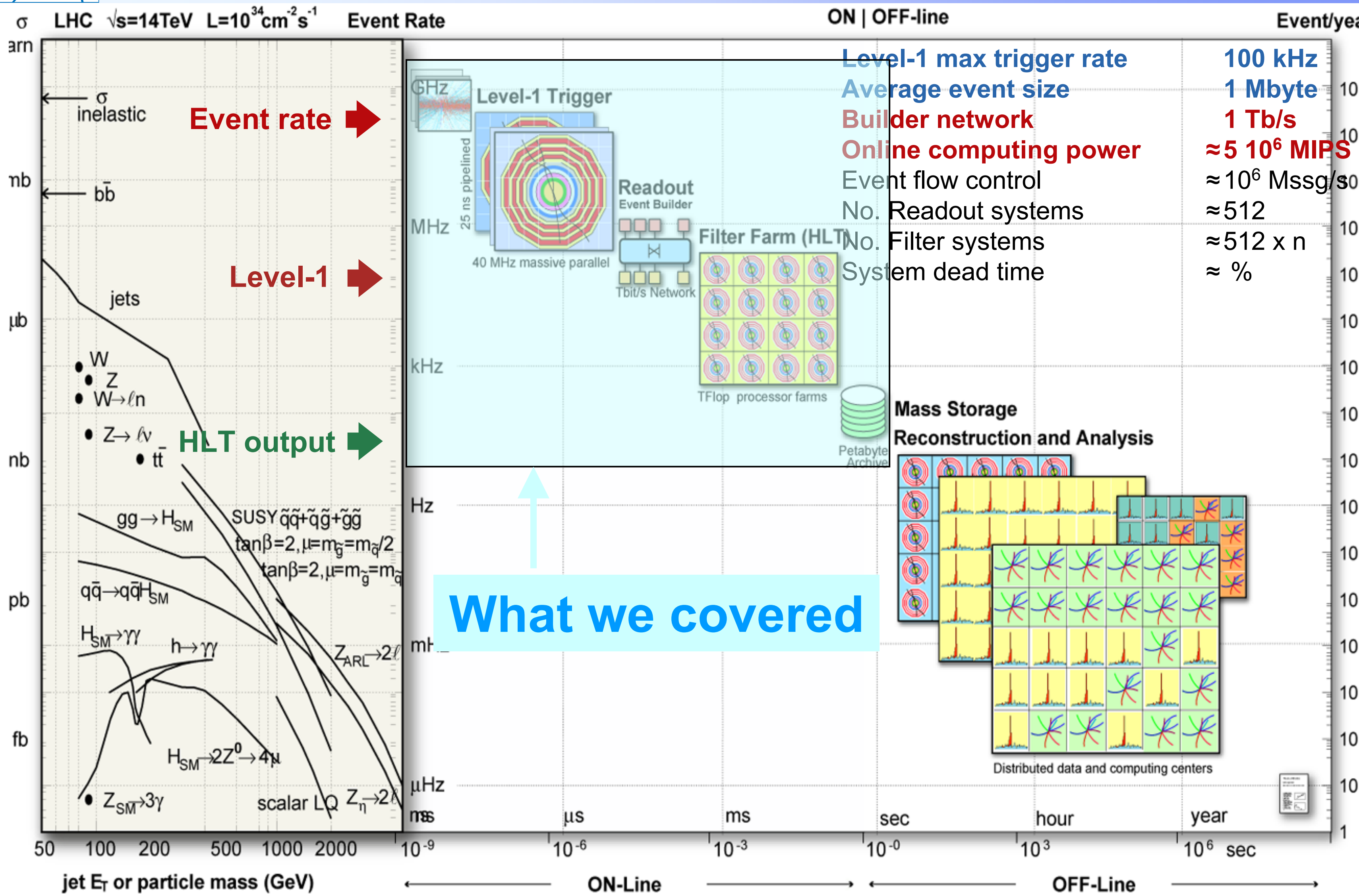


	Signal	Background	Total
Single e	$W \to e\nu$: 10 Hz	$\pi^{\pm}/\pi^0$ overlap: 5 Hz π^0 conversions: 10 Hz b/c $\to$ e: 8 Hz	33 Hz
Double e	$Z \to ee$: 1 Hz	~0	1 Hz
Single γ	2 Hz	3 Hz	5 Hz
Double γ	~0	5 Hz	5 Hz
			44 Hz

After the Trigger and the DAQ/HLT



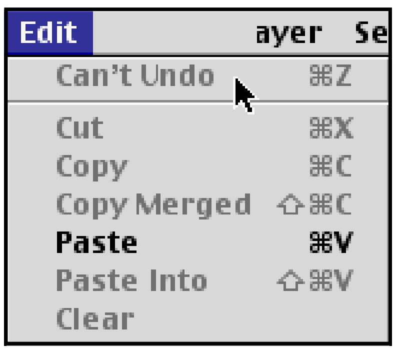
Online Physics Selection: summary

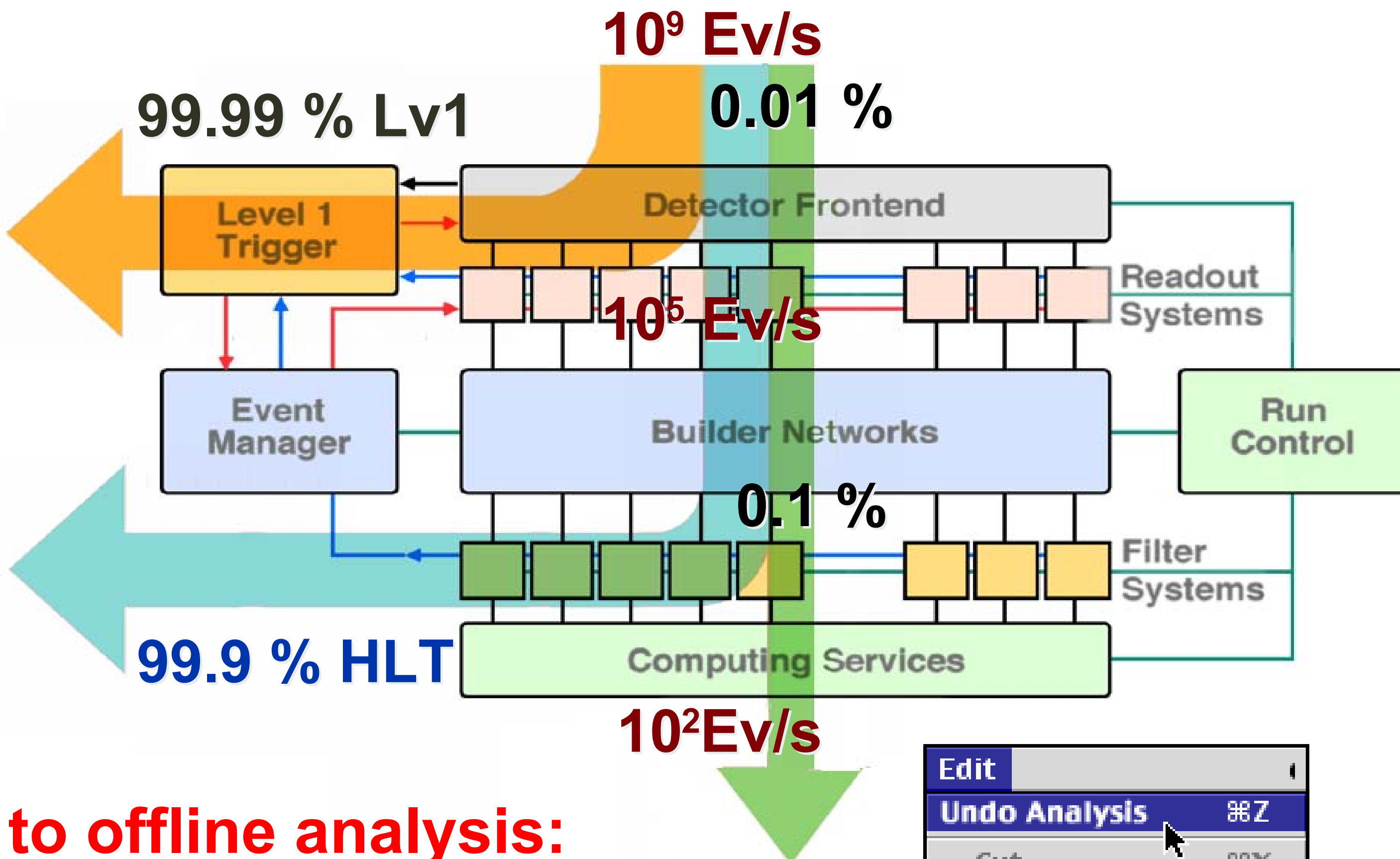


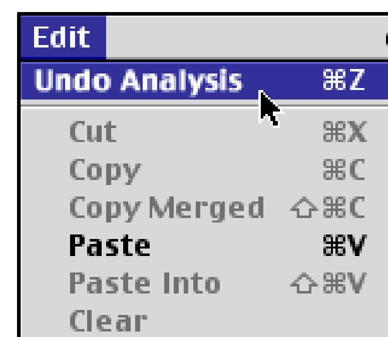
(Grand) Summary

- The Level-1 trigger takes the LHC experiments from the 25 ns timescale to the 10-25 μs timescale
  - Custom hardware, huge fanin/out problem, fast algorithms on coarse-grained, low-resolution data
- Depending on the experiment, the next filter is carried out in one or two (or three) steps
  - Commercial hardware, large networks, Gb/s links.
  - If Level-2 present: low throughput needed (but need Level-2)
  - If no Level-2: three-dimensional composite system
- High-Level trigger: to run software/algorithms that as close to the offline world as possible
  - Solution is straightforward: large processor farm of PCs
  - Monitoring this is a different issue
- All of this must be understood, for it's done online.

A parting thought







With respect to offline analysis:

Same hardware (Filter Subfarms)

Same software

But different situations