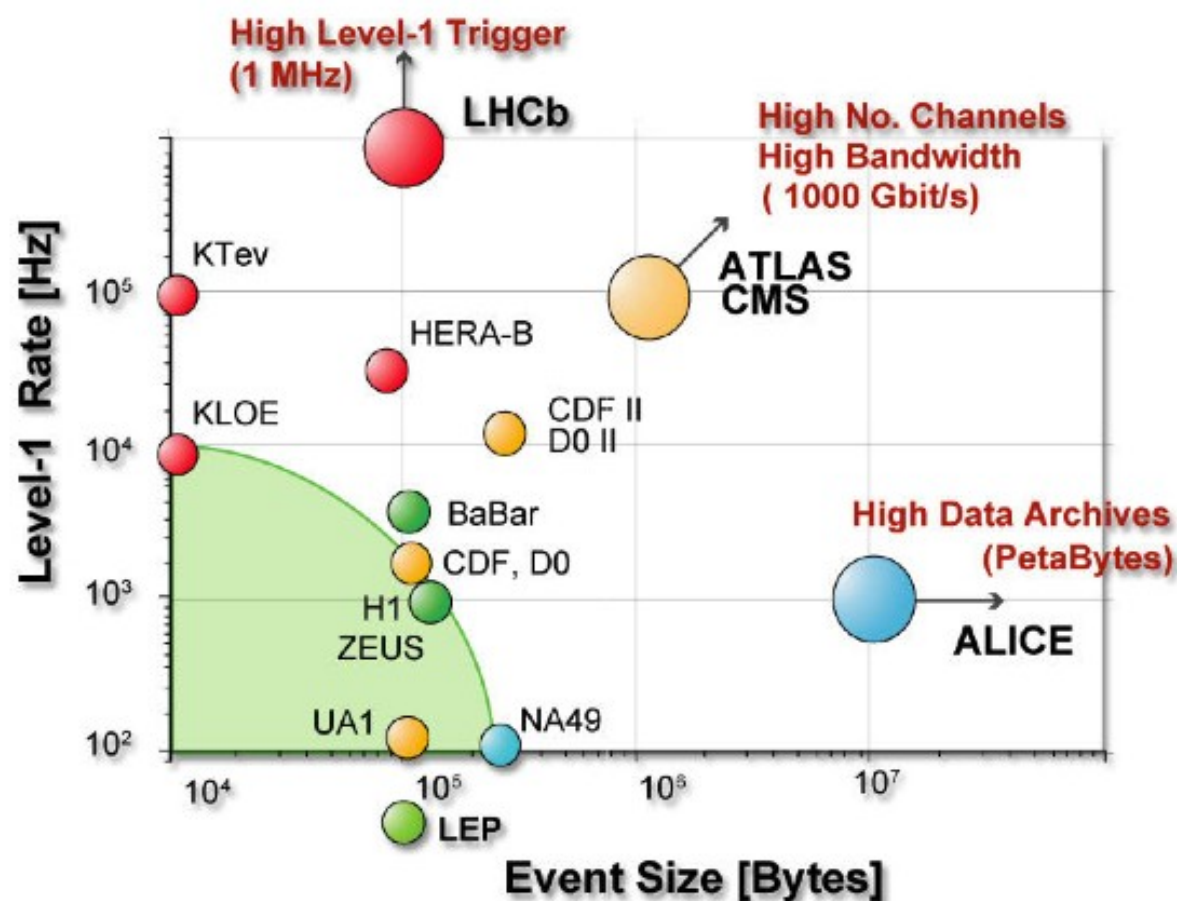


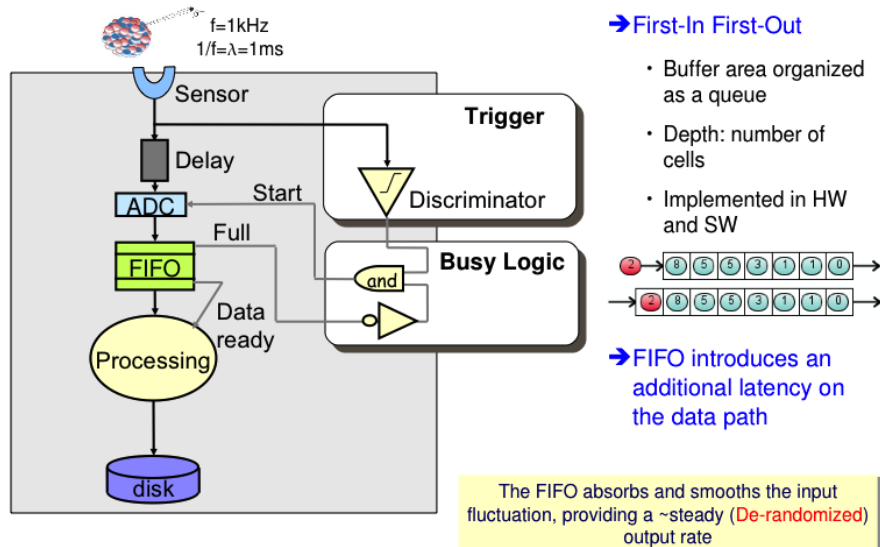
# TDAQ design: from test beam to medium size experiments



# How do we go

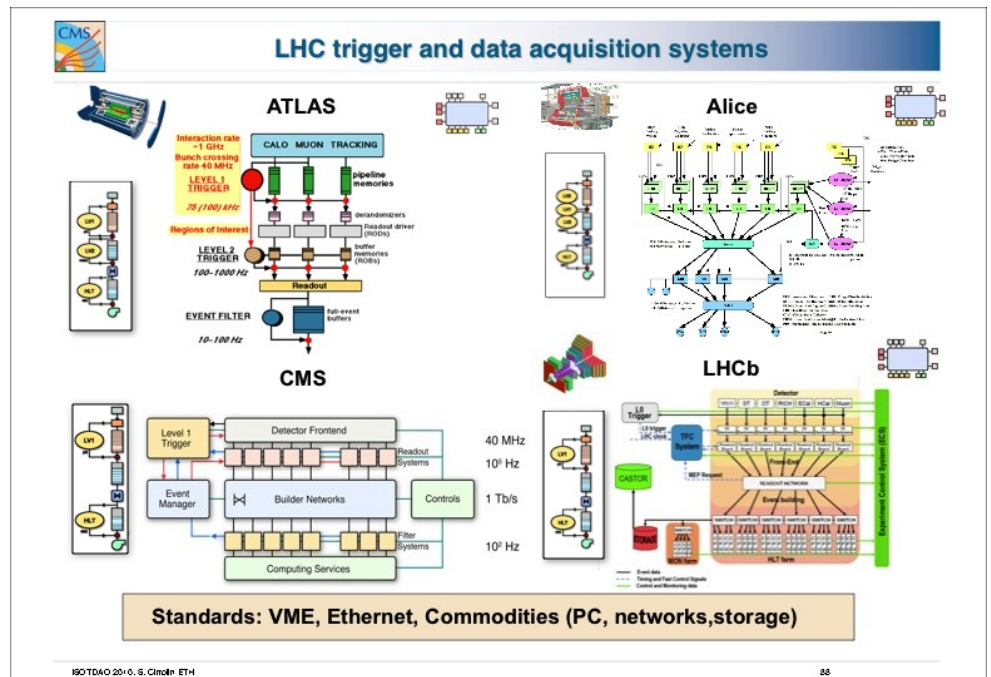


## Basic DAQ: De-randomization



← from here

to here →



February 10<sup>th</sup> 2011

Introduction to Data Acquisition - W.Vandelli - ISOTDAQ2011

15

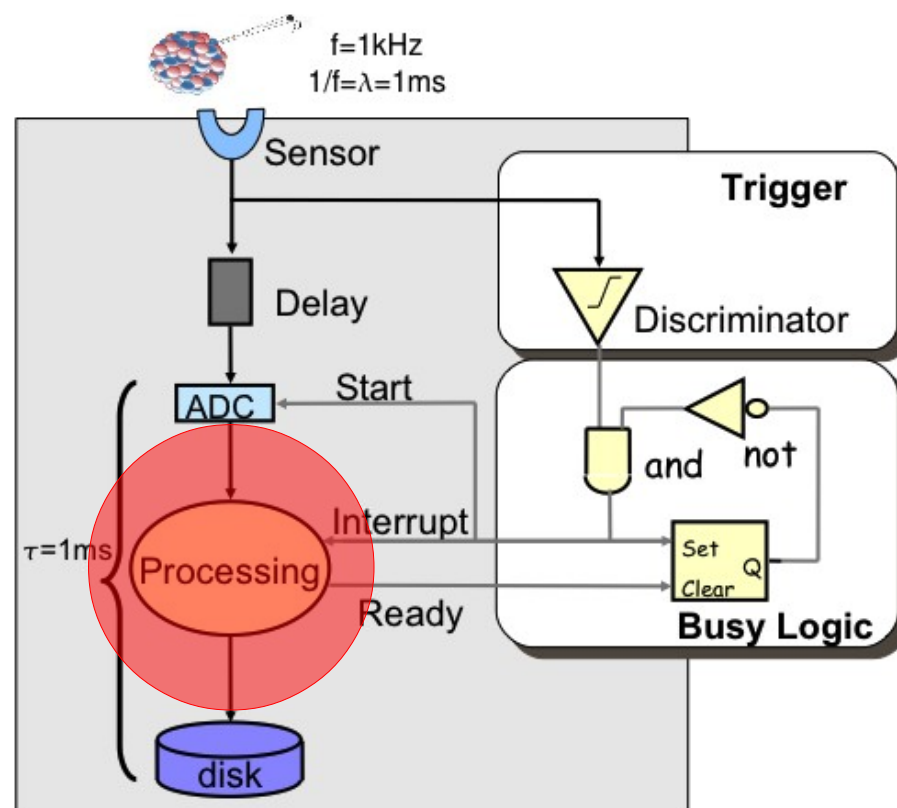
# Outline

- Step 1: Increasing the rate
- Step 2: Increasing the sensors
- Step 3: Multiple Front-Ends
- Step 4: Multi-level Trigger
- Step 5: Data-Flow control
- Trends etc

# Step One: increasing the rate

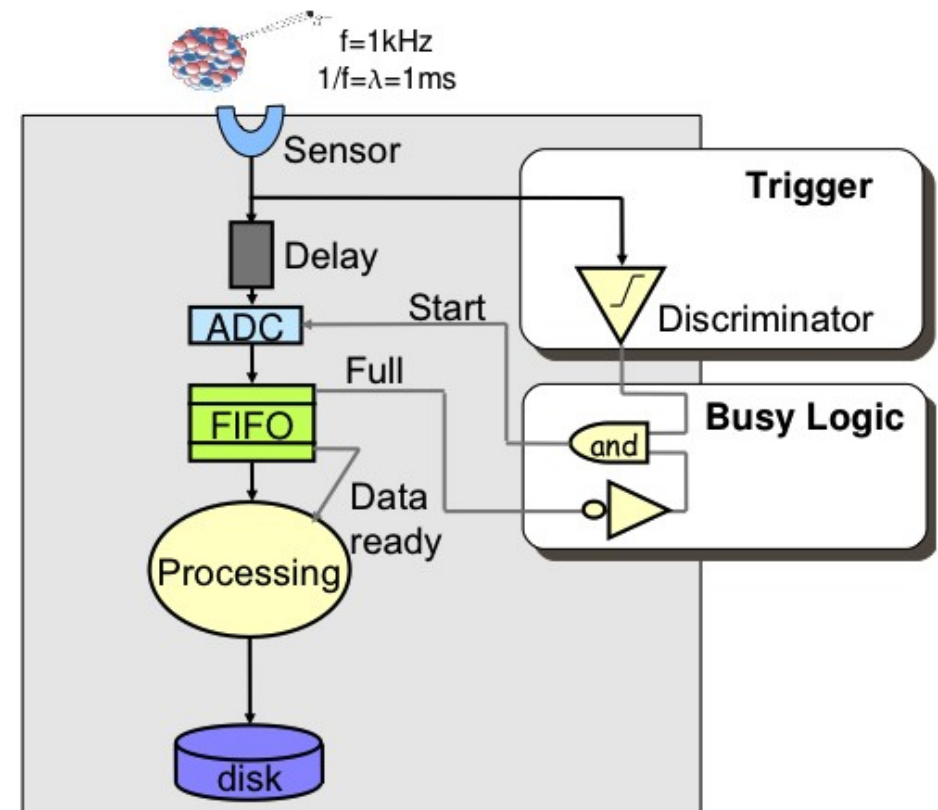
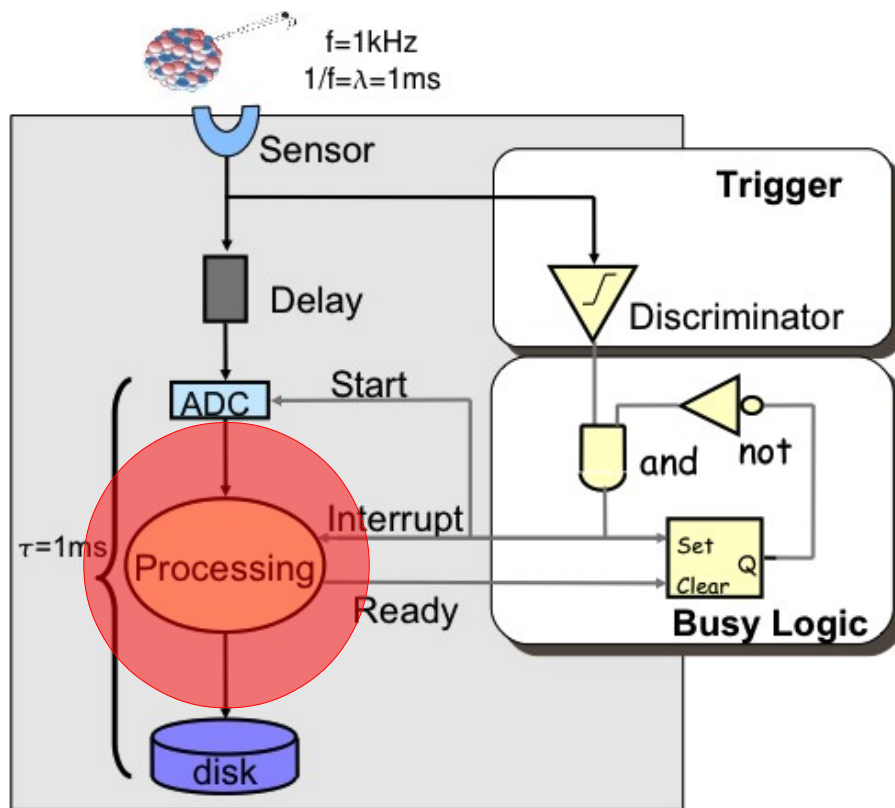
## Processing:

- Wait for ADC (poll/irq)
- Read it
- Clear it
- Re-format data
- write to disk



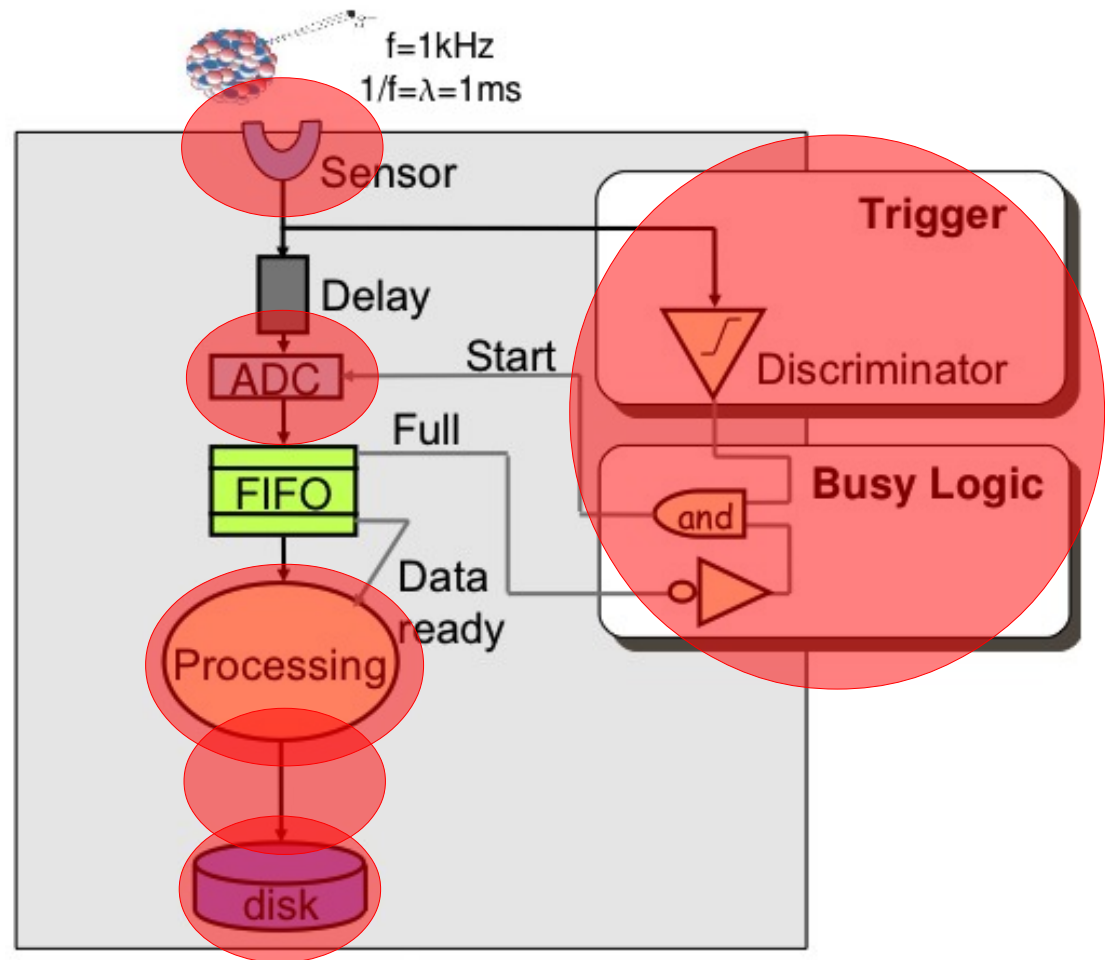
# Derandomisation

- Processing here is an evident bottleneck
- Buffering decouples the problem



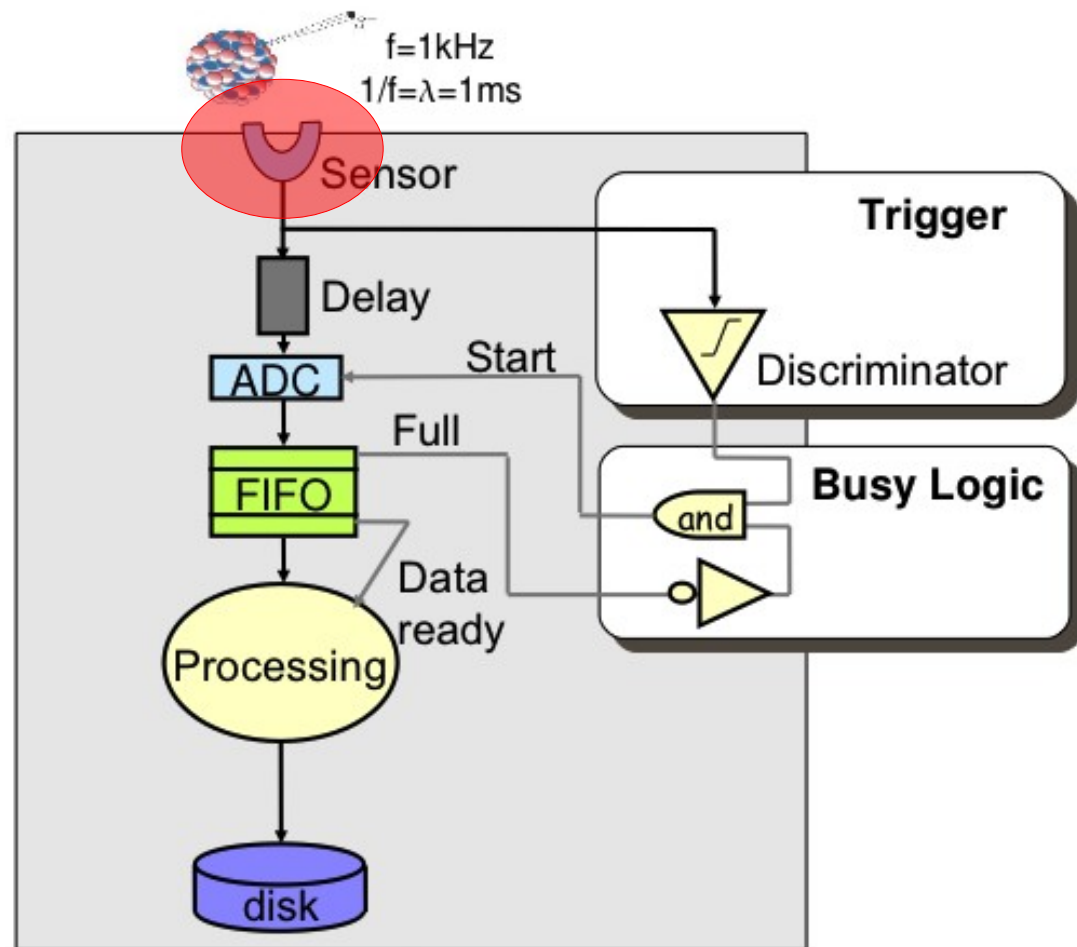
# Is it over? no.

- Even in a simple DAQ there are many other possible limits



# Is it over? no: the sensor

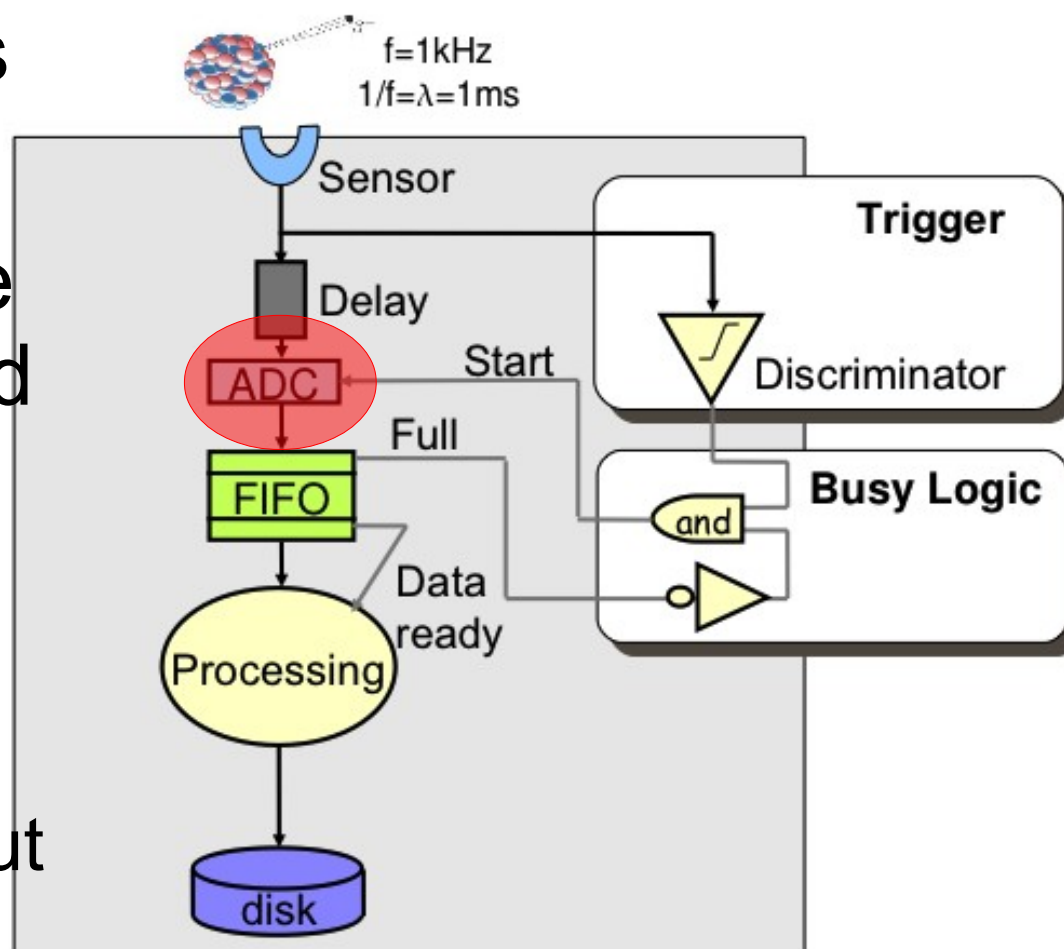
- Sensors are limited by physical processes
  - drift times in gases
  - charge collection
- choose fast processes
- also the (hidden) analog F.E. imposes limits
- split the sensors, each gets less rate:  
“increase granularity”





# Is it over? no: the ADC

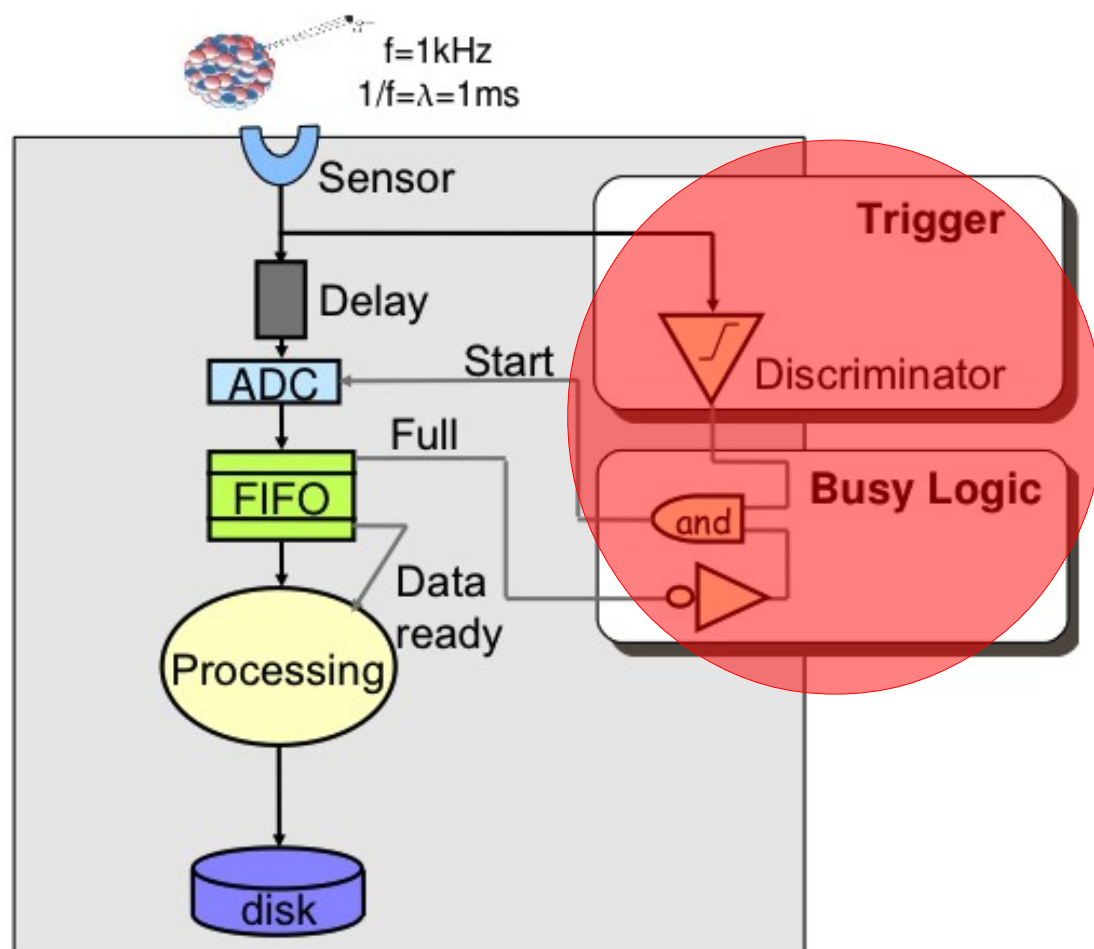
- Analog/Digital F.E. is also limited
- Faster ADCs pay the price in precision and power consumption
- Alternatives:
  - analog buffers
  - see Detector Readout lecture





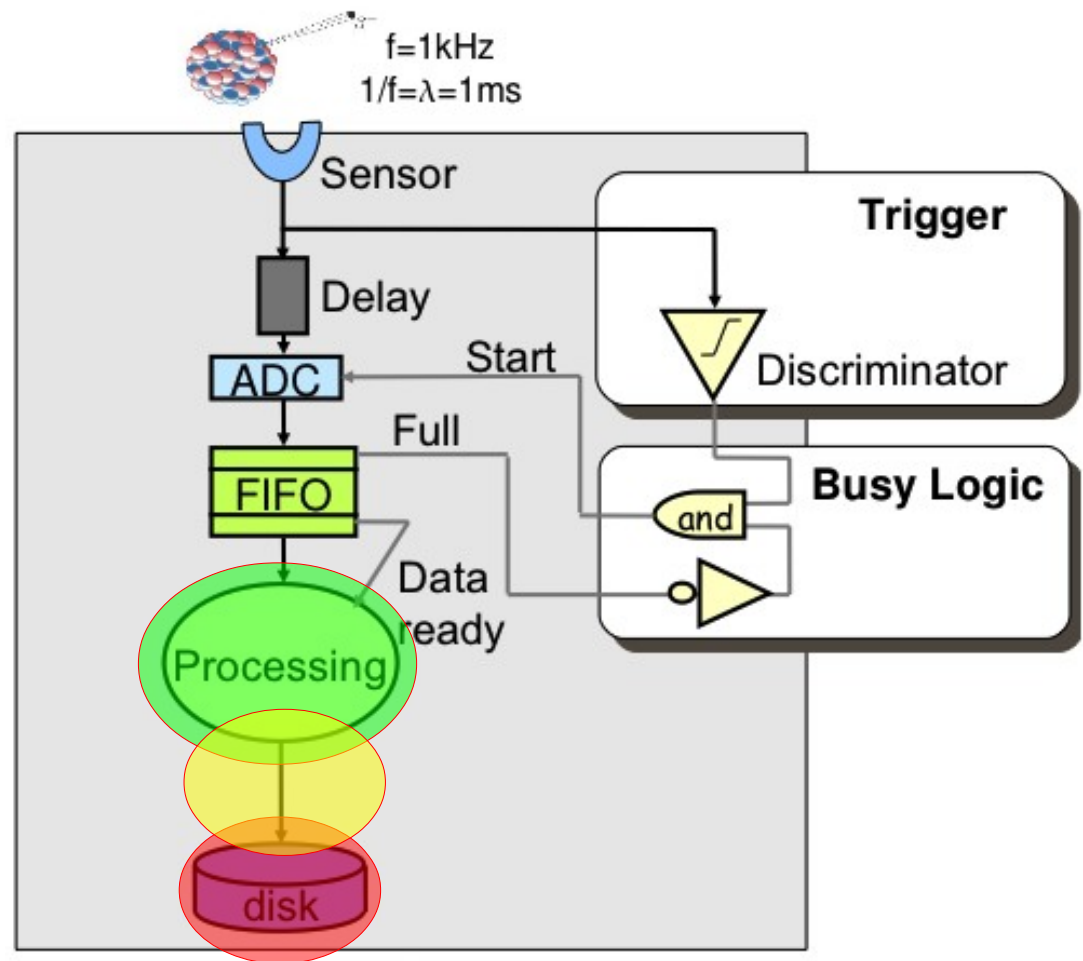
# Is it over? no: the Trigger

- A simple trigger is fast (so I lied, not an issue?)
- a complex trigger logic may not be so fast even when all in hardware
- to get a single answer all information must be collected in a single point
  - in one step:  
too many cables
  - in many steps:  
delays



# Is it over? no: the dataflow

- Data Processing is quite easy and scalable
- Data Transport may not be easy
- Final storage is expensive (and at some point not easy either)



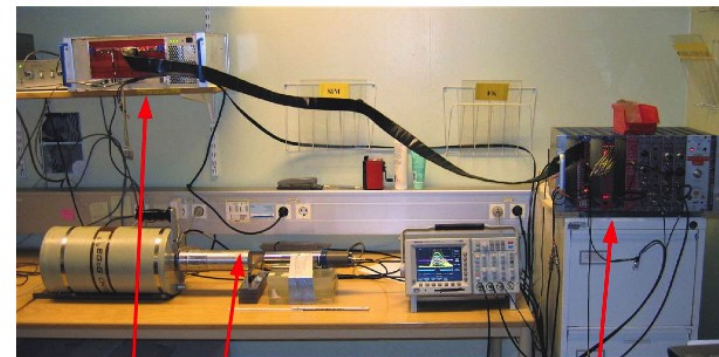
# A little example



## Ge crystal for isotope identification



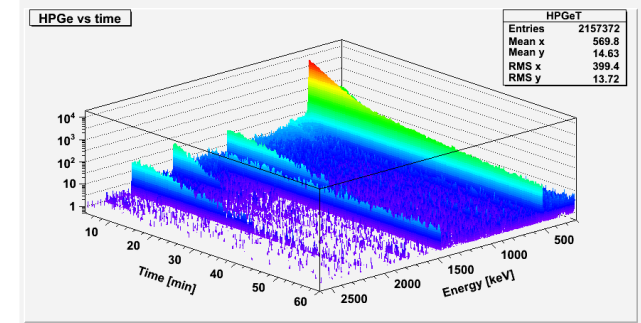
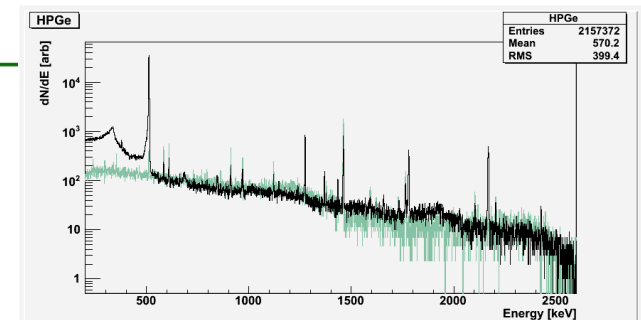
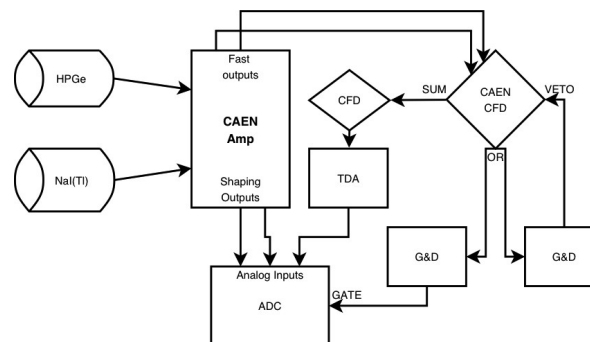
- HPGe + NaI Scintillator  
High res spectroscopy and beta+ decay identification
- minimal trigger with busy logic
- Peak ADC with buffering, zero suppression
- VME SBC with local storage
- Rate limit  $\sim 14\text{kHz}$ 
  - HPGe signal shaping for charge collection
  - PADC conversion time
- 3x12 bits data size  
(coincidence in an ADC channel)  
+32bit ms timestamp
- Root for monitor & storage



Crystal HPGe

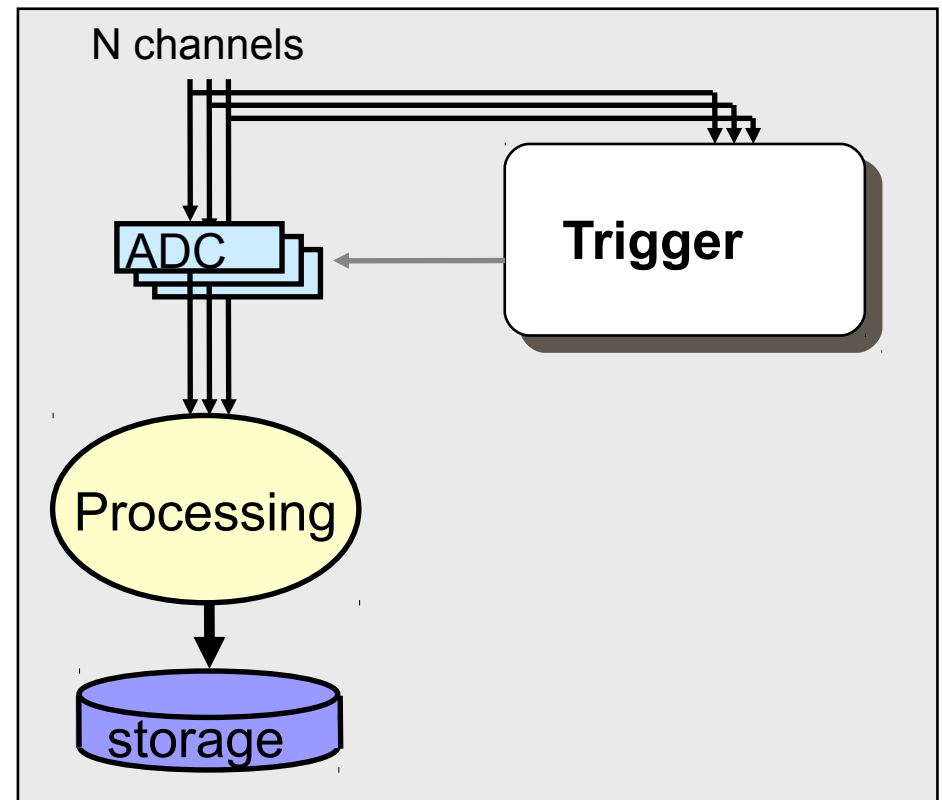
Readout (ADC)

Trigger & front-end

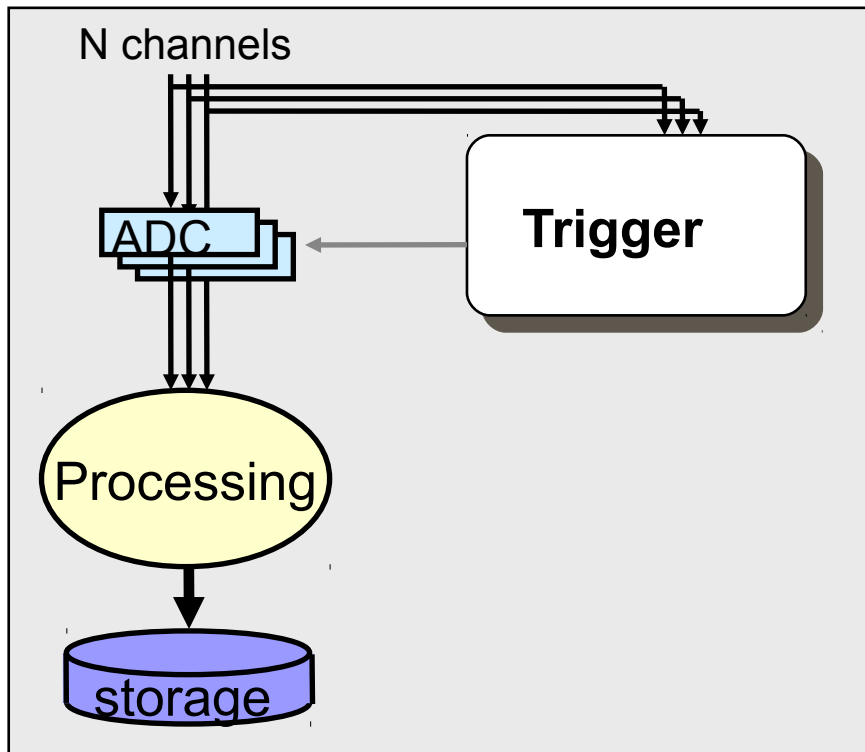


# Step two: increasing the sensors

- Multiple channels (usually with FIFOs)
- Single, all-HW trigger
- single processing unit
- single I/O

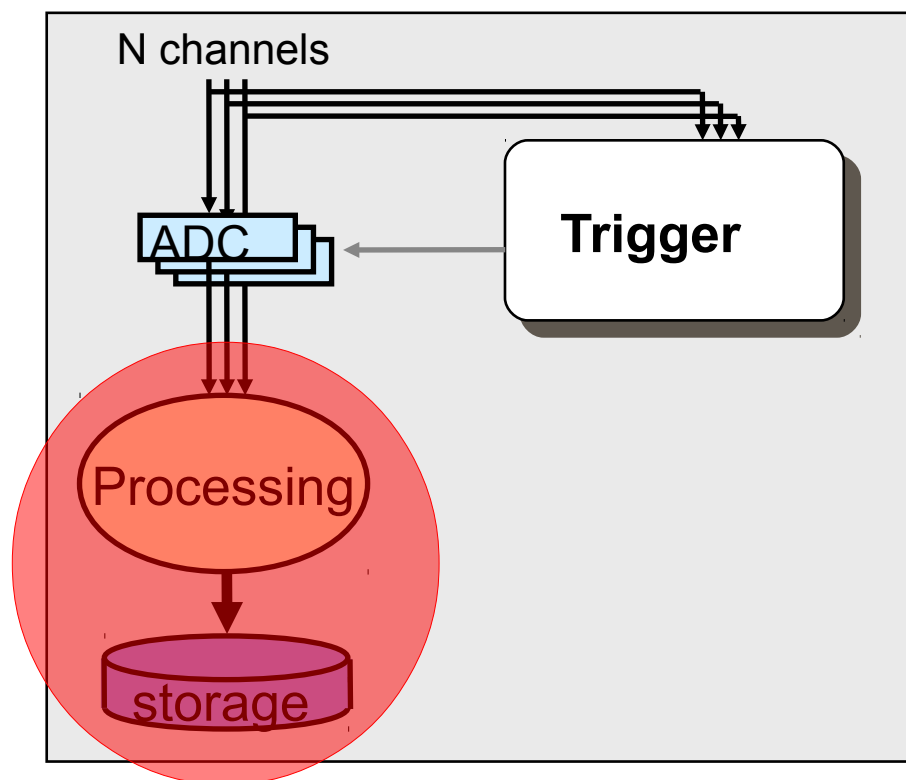


# multi-channels, single FE PU



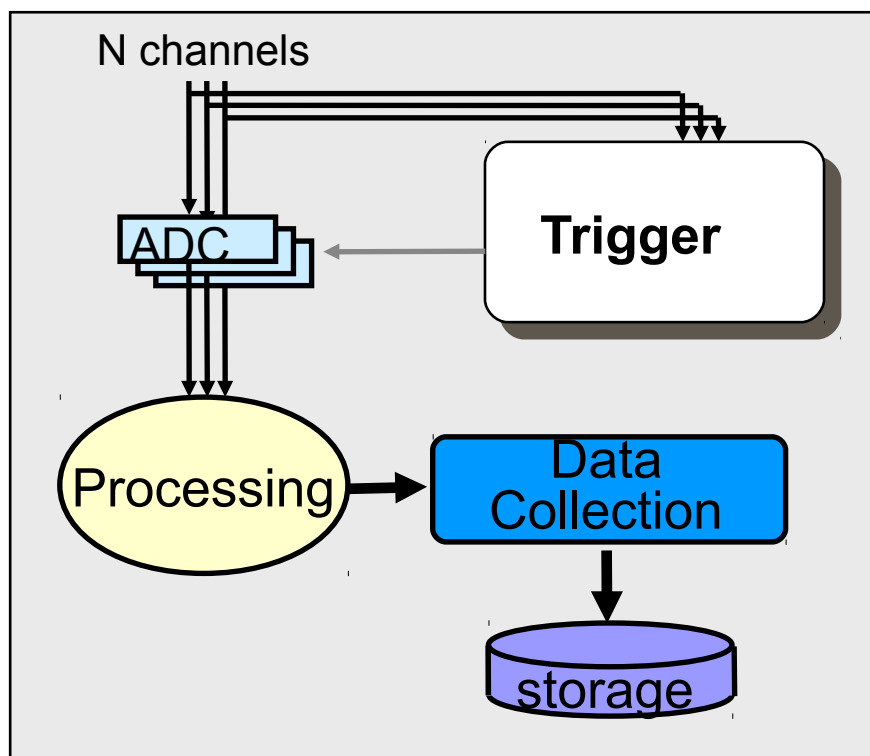
- common architecture in test beams and small experiments
- Usually the rates limited by (interesting) physics itself, not TDAQ system
- or by the sensors

# Bottlenecks: PU and Storage



- A single Processing Unit can be a limit
  - collate / reformat / compress data can be heavy for an F.E. CPU
  - simultaneously writing storage
- Final storage too:
  - VME up to 50MB/s  
-> 1TB in 5h  
too many disks in a week!

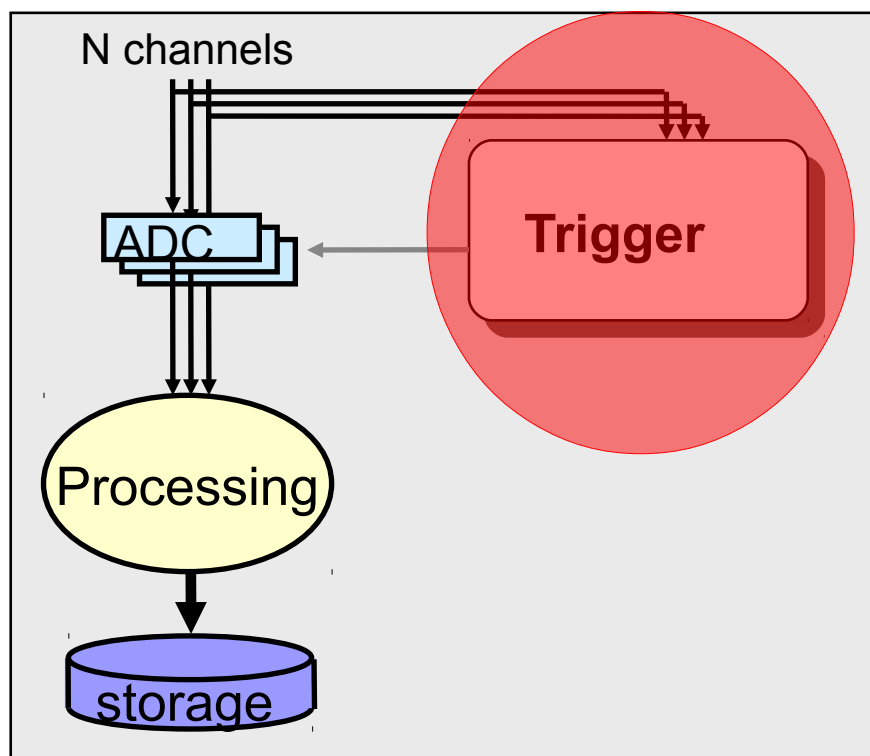
# Solution: Decouple FE from Storage



- A dedicated “Data Collection” unit to format / compress and store
- Free FE for smarter processing or decreased dead time on non-buffered ADCs



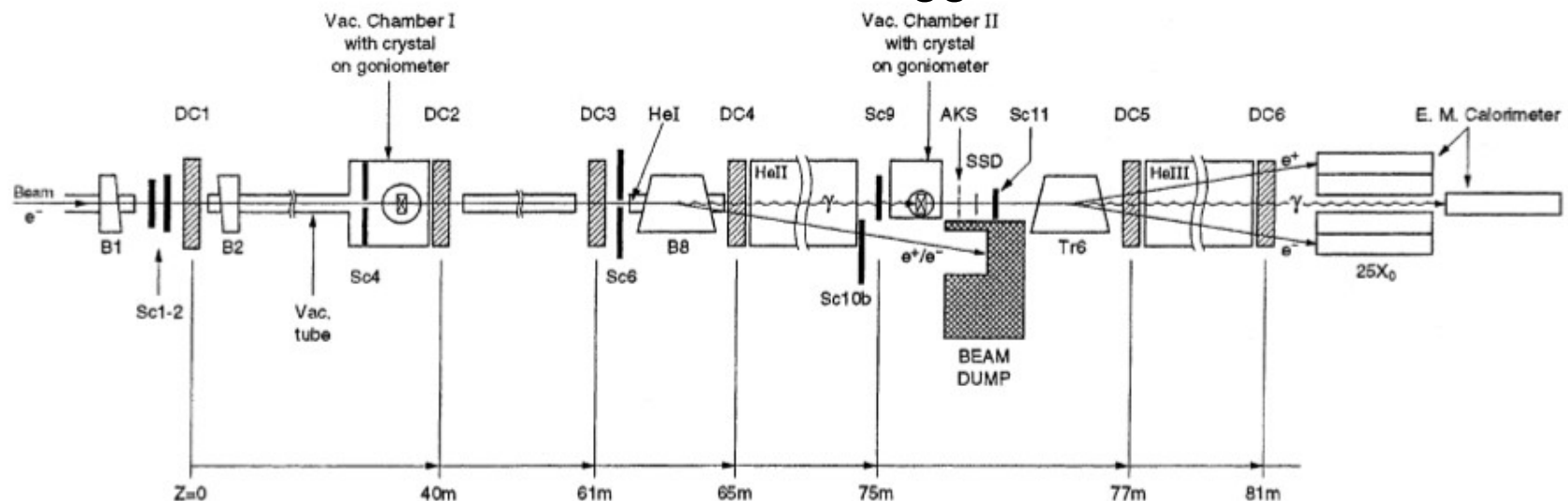
# Bottlenecks: Trigger ?



- To reduce data rates (to avoid storage issues) a non-trivial trigger is needed.
- With the number of channels that a VME can support we may already hit manageability limits for discrete logic
- Integrated, programmable logic came to rescue

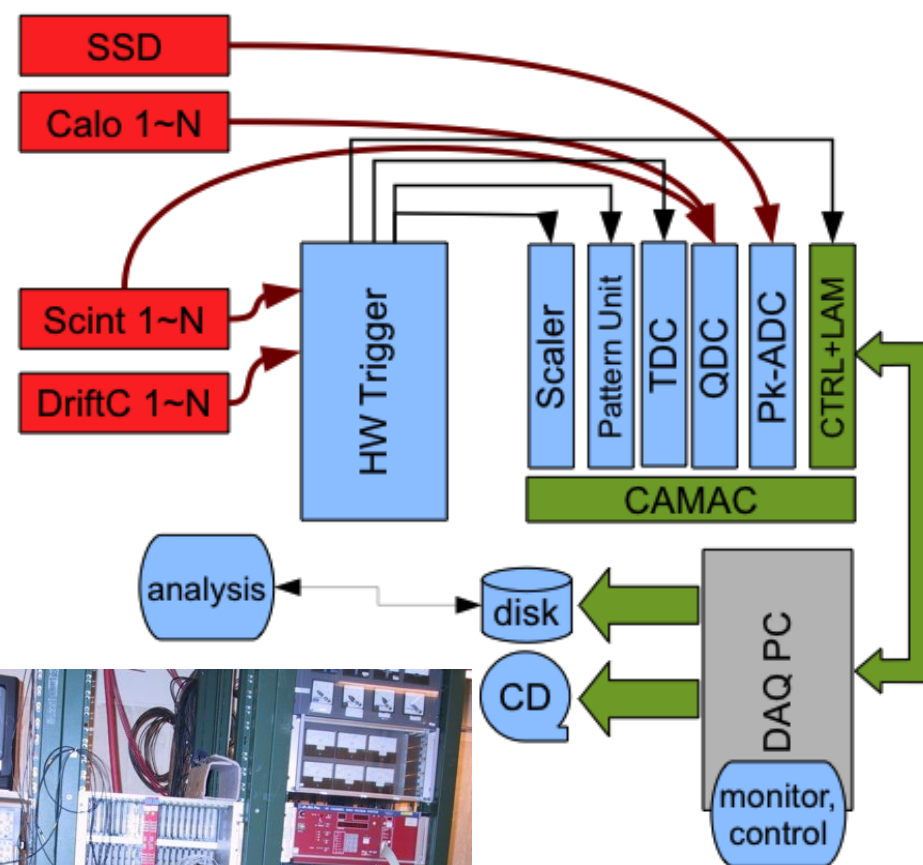
# A real example: NA43/63

- Radiation emission effects:  
Coherent emission in crystals and structured target, LPM suppression...
- 80~120GeV  $e^-$  from CERN SPS slow extraction
- 2s spill every 13.5s
- Needs very high angular resolution
- Long baseline + high-res, low material detectors  
→ Drift Chambers
- 10 kHz limit on beam for radiation damage
- results in typical 2~3 kHz physics trigger



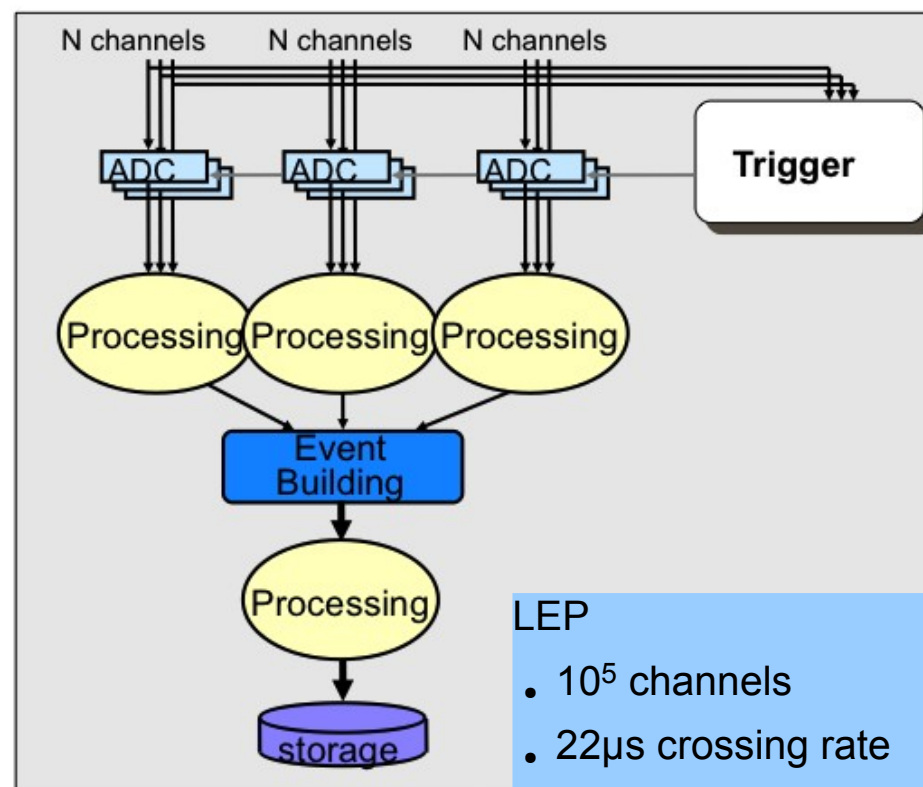
# A real example: NA43/63

- 30~40 TDC, 6~16 QDC, 0~2 PADC  
(depends on measurement)
- CAMAC bus  
1MB/s, no buffers, no Z.S.
- single PC readout
- NIM logic trigger (FPGA in 2009)
  - pileup rejection
  - fixed deadtime



# Step Three: Multiple FEs

- LEP experiments were typical examples
- complex detectors, not very high rate physics, nor background
- little pileup, limited channel occupancy
- simpler, slow gas-based main trackers

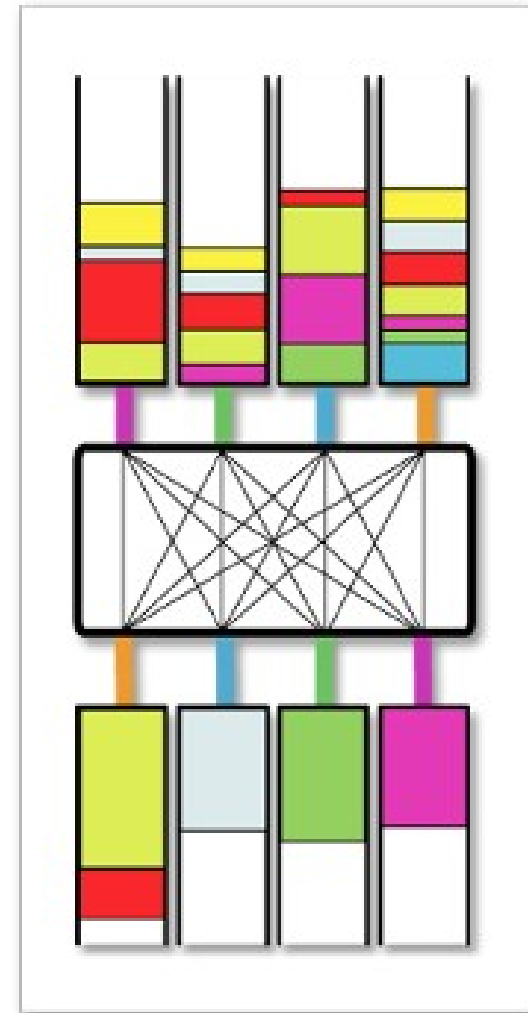


## LEP

- $10^5$  channels
- $22\mu\text{s}$  crossing rate
  - no event overlap
- single interaction

# Event Building ?

- Event “fragments”
  - in detector/sector-specific pipeline
- keep track of which event they belong to
  - timestamp or
  - L1 trigger #
- gather every fragment to single location



# A minimal example



MineralPET TechDemo :

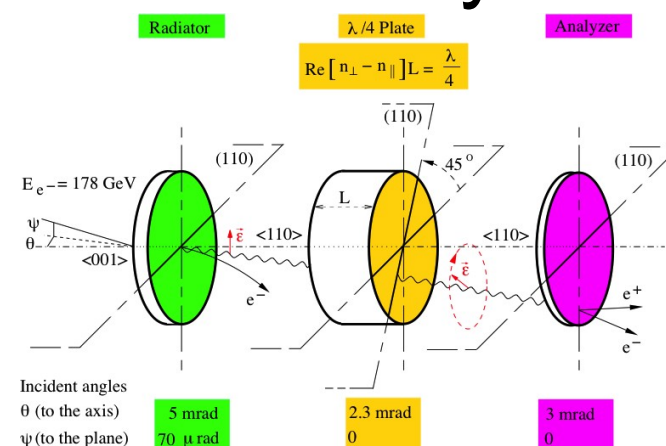
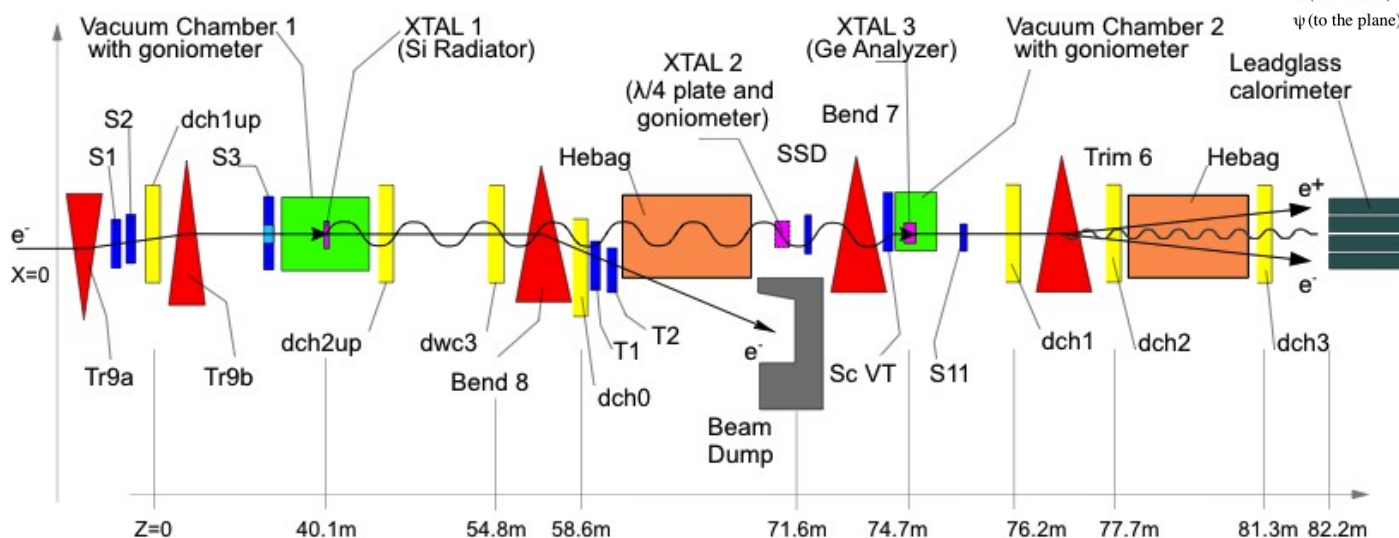
- 16 position-sensitive scintillators
- 2 \* 32-Ch PeakADC
- 1 \* 64-Ch TDC
- 8 kHz readout, ~256bytes events
  - single trigger, not interested in absolute rates, so it can run near saturation

- Today's VME modules do buffering, zero suppression etc.
- best throughput achieved by block transfers of full buffers
- as soon as you use more than one module :
  - unpack blocks into events
  - merge data from same event across all sources
- “Network” design collapsed in a single system



# A small size example: NA59

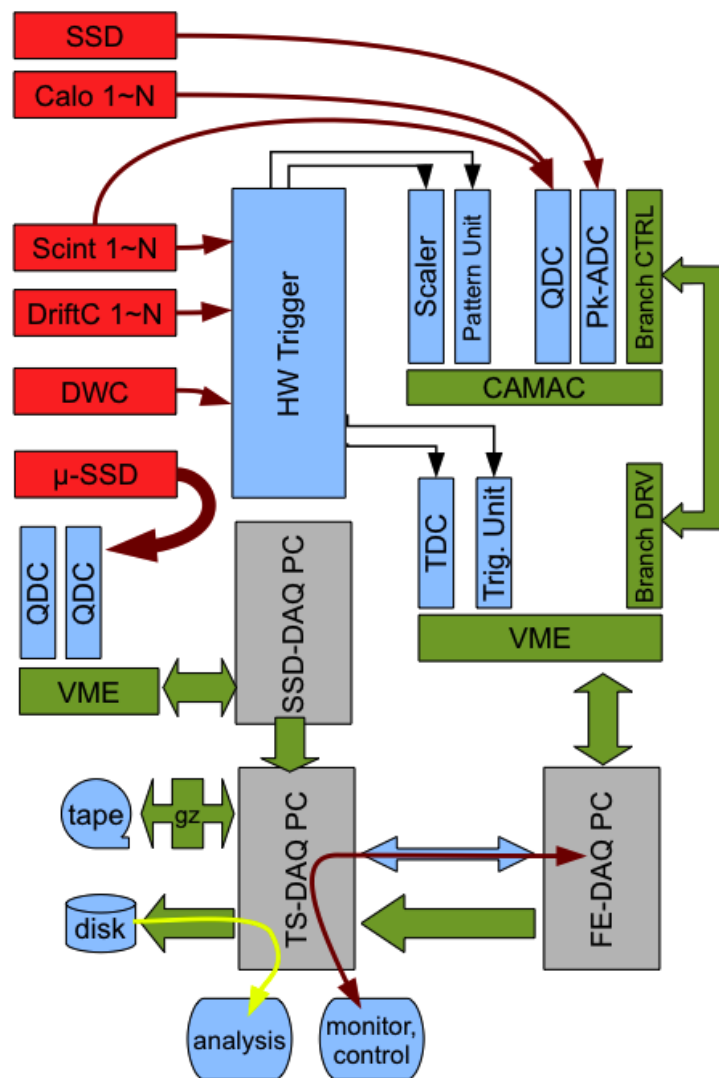
- 80~120GeV e<sup>-</sup> from CERN SPS slow extraction
- Radiation polarization conversion in crystals
- 2s spill every 13.5s



- Drift Chambers and Delay Wire chambers
- ~10μm resolution
- ~10μrad resolution

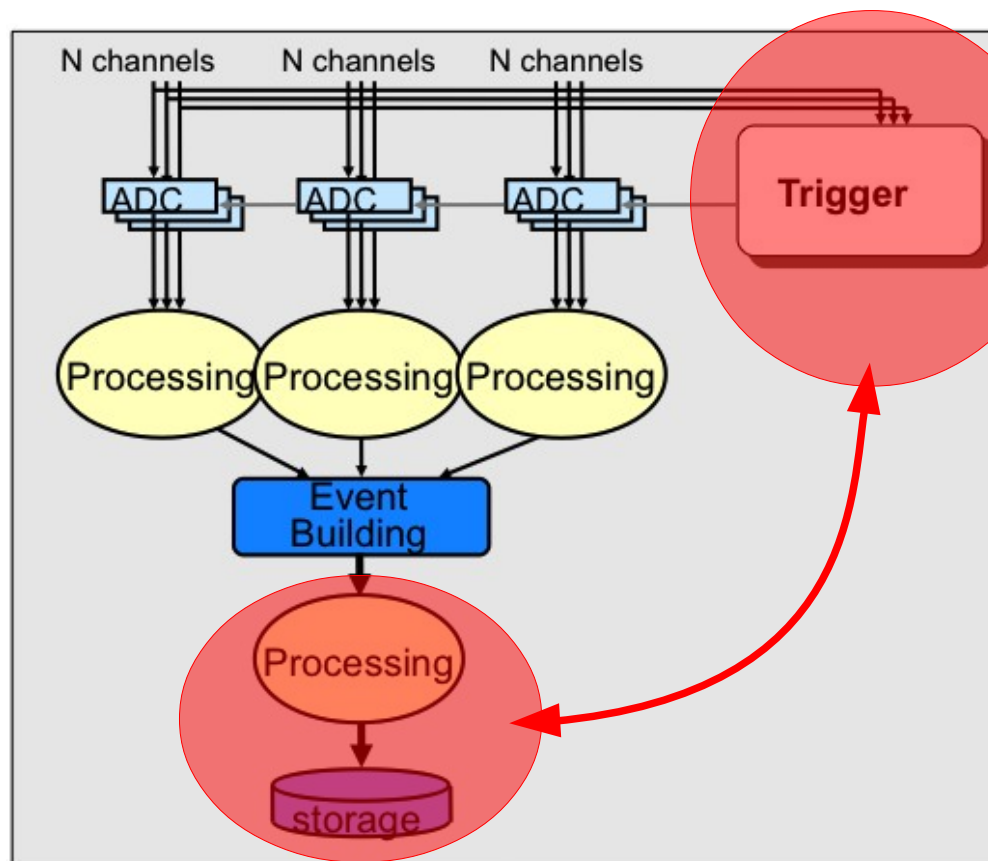


# An small size example: NA59



- Main VME+CAMAC FE
- Silicon Tracker FE
- Decoupled “Block Building” and Storage
- SPS: 2s spill in 13.5s  
take advantage of idle duty cycle for processing & storage
- Physics and detectors limit the rate to ~4kHz
- Event size ~280bytes  
→ 840kB/s  
*not far from LEP data rates!*

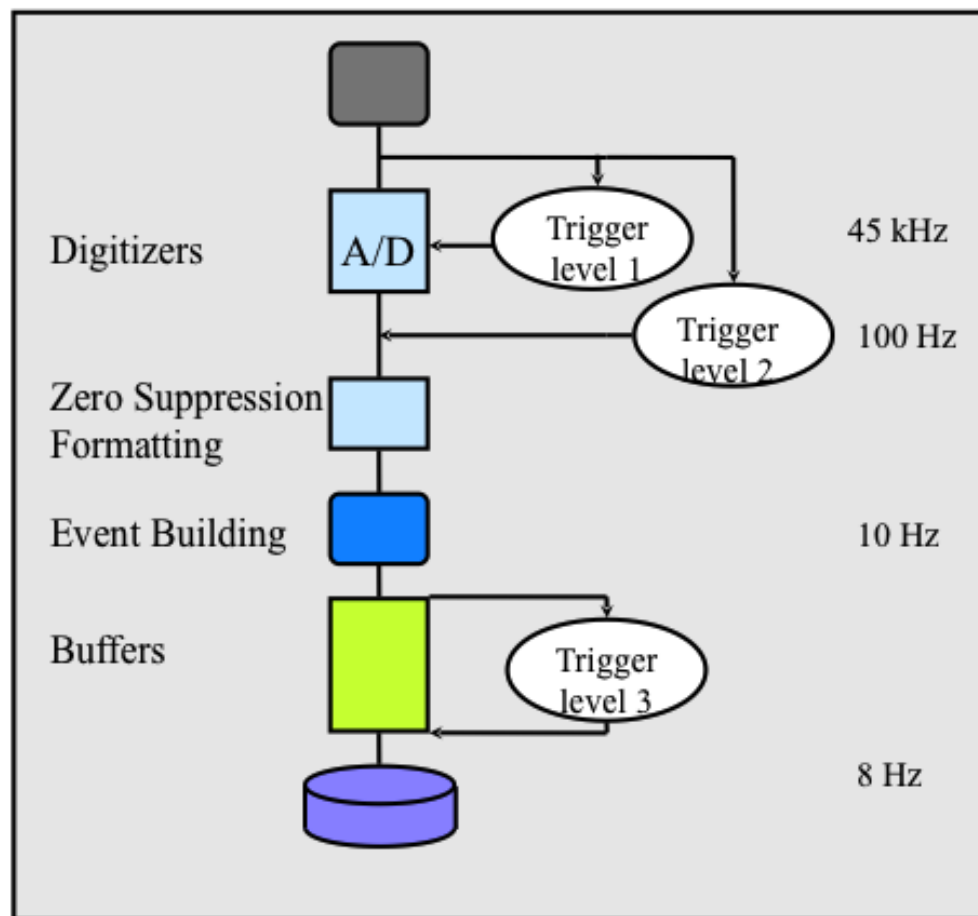
# Bottlenecks?



- Trigger complexity vs storage
- Single HW trigger is not sufficient to reduce rate
- Introduce L2 Trigger
- Introduce HLT

# Step four: Multi-level trigger

Typical Trigger / DAQ structure at LEP



- More complex filters
- but slower
- applied later in the chain

*see Trigger lectures*

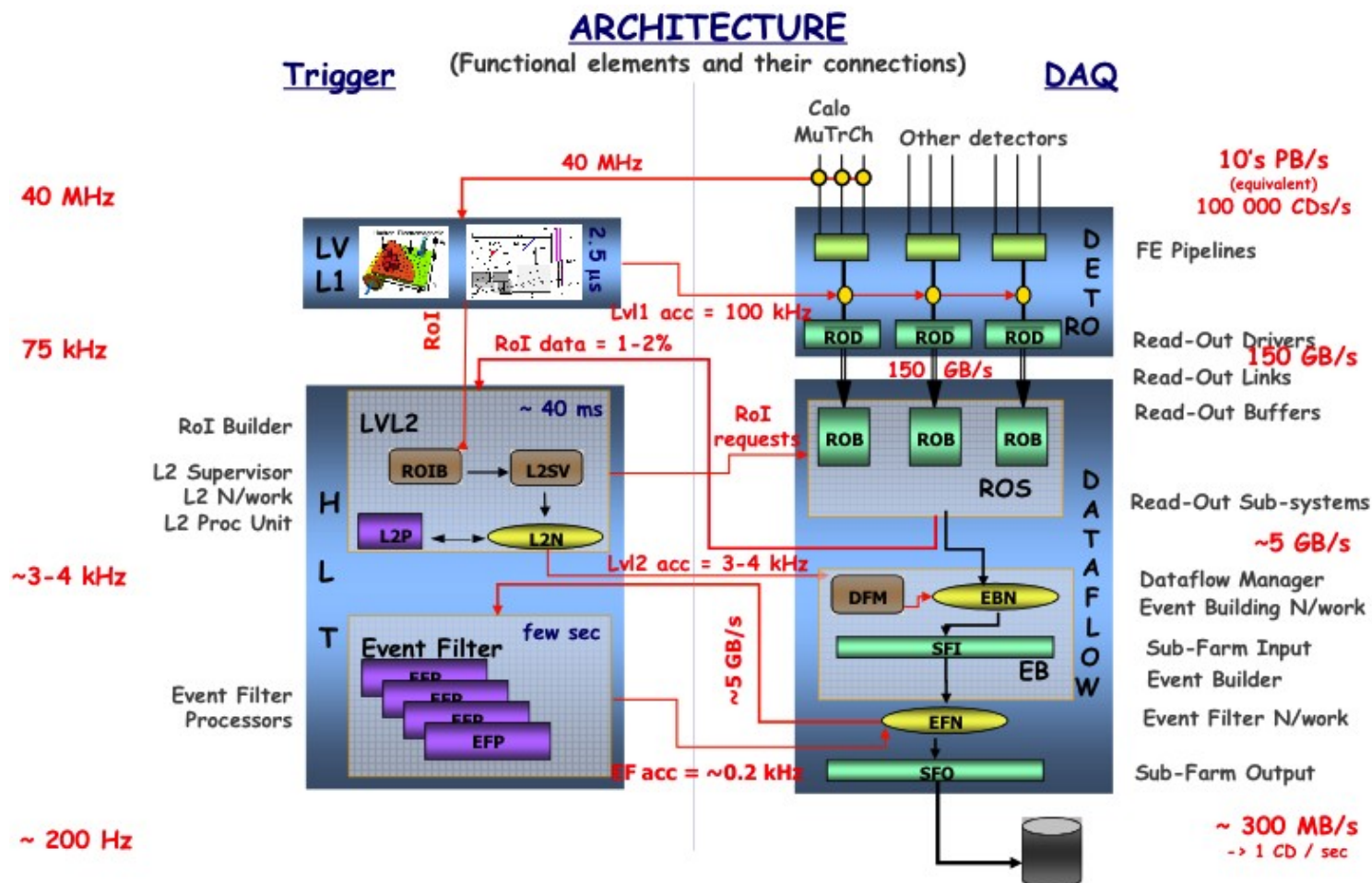
## LEP

- $10^5$  channels
- 22 $\mu$ s crossing rate
  - no event overlap
- single interaction
- L1  $\sim 10^3$  Hz
- L2  $\sim 10^2$  Hz
- L3  $\sim 10^1$  Hz
- 100kB/ev  $\rightarrow$  1MB/s

# ATLAS: oh my!

## LHC

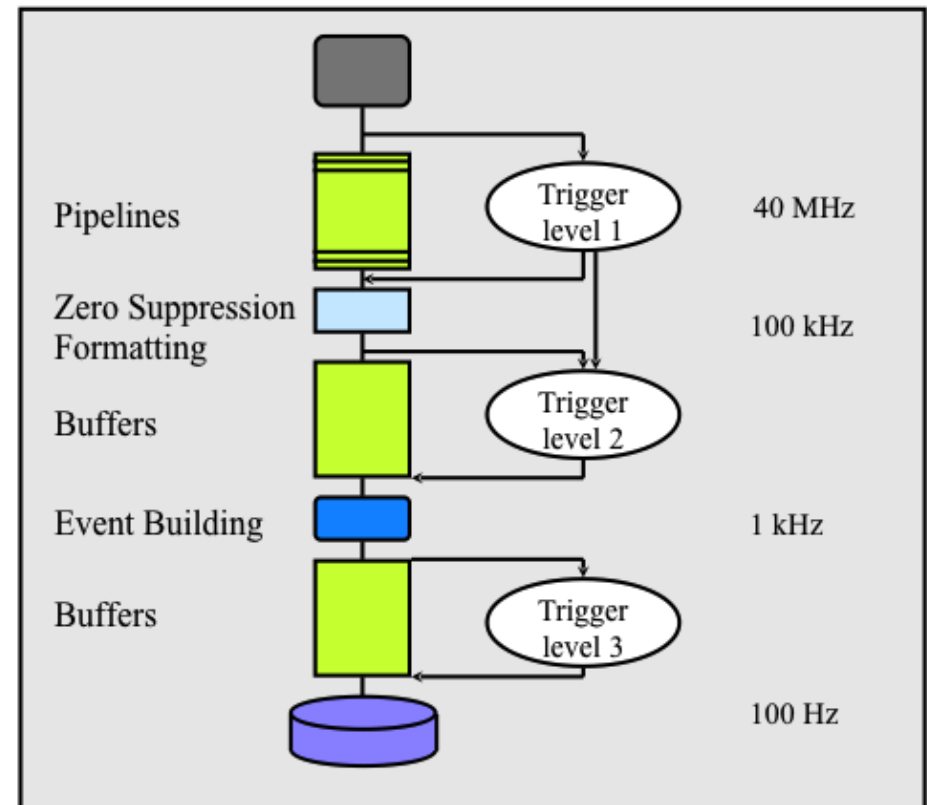
- $10^7$  channels
- 25ns crossing rate
  - high event overlap
- 20 interactions
- $L1 \sim 10^5$  Hz
- $L2 \sim 10^3$  Hz
- $L3 \sim 10^2$  Hz
- $1\text{MB/ev} \rightarrow 100\text{MB/s}$



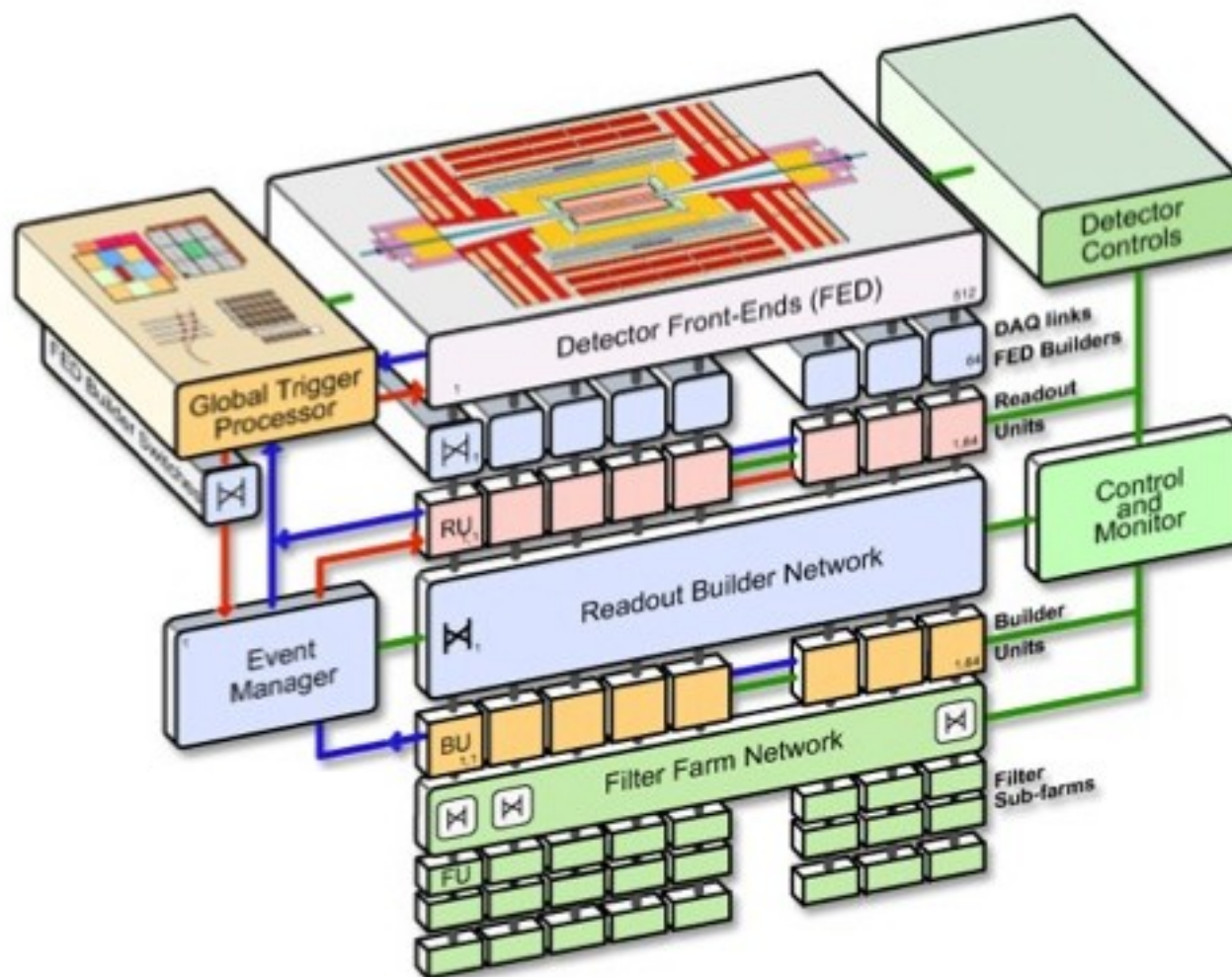
# Actually, it's “just”

- Still 3-level trigger
- buffers everywhere
- L2 on CPU, not HW, but limited to ROIs
- L3 using offline algorithms
- “economical” design: the least CPU and network for the job

see “*TDAQ for LHC*” lecture



# CMS: oh my!

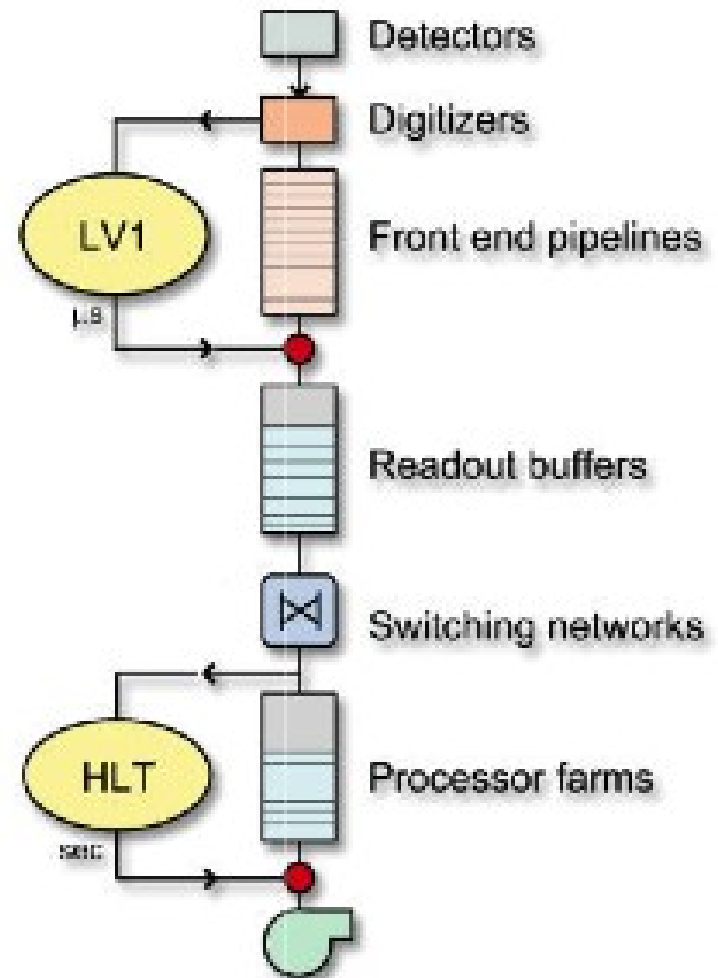


CMS TDAQ Design - S. Cittolin @ISOTDAQ 2010

# Actually it's “just”

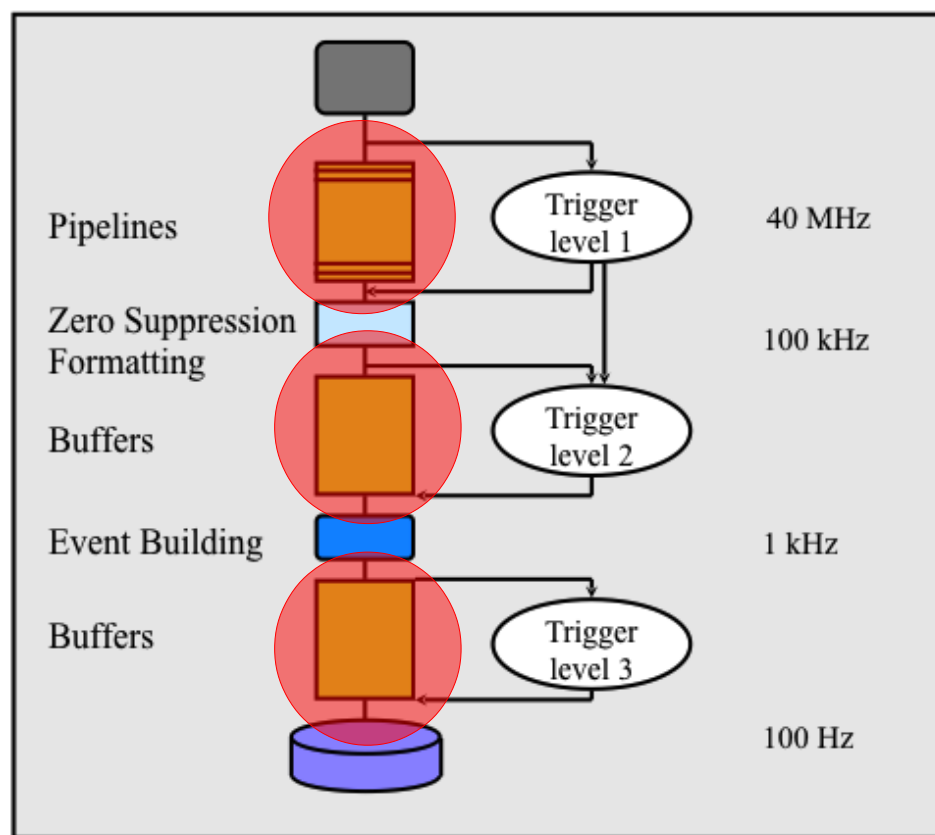
- Only two trigger levels
- Intermediate event building step (RB)
- larger network switching

see “TDAQ for LHC” lecture





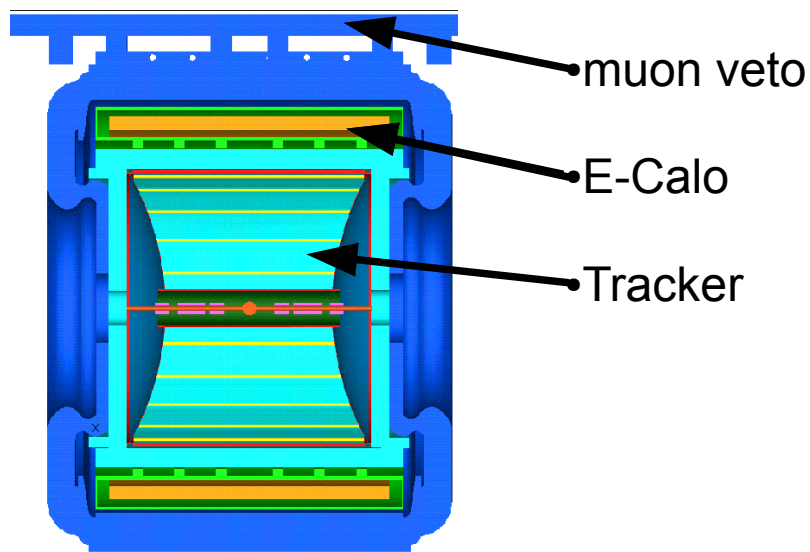
# Step Five: Data Flow control



- Buffers are not the final solution:  
they can overflow
  - bursts
  - unusual event sizes
- Discard
  - local, or
  - “backpressure”,  
tells lower levels to discard
  - up the chain to a single point, else efficiency becomes unknown

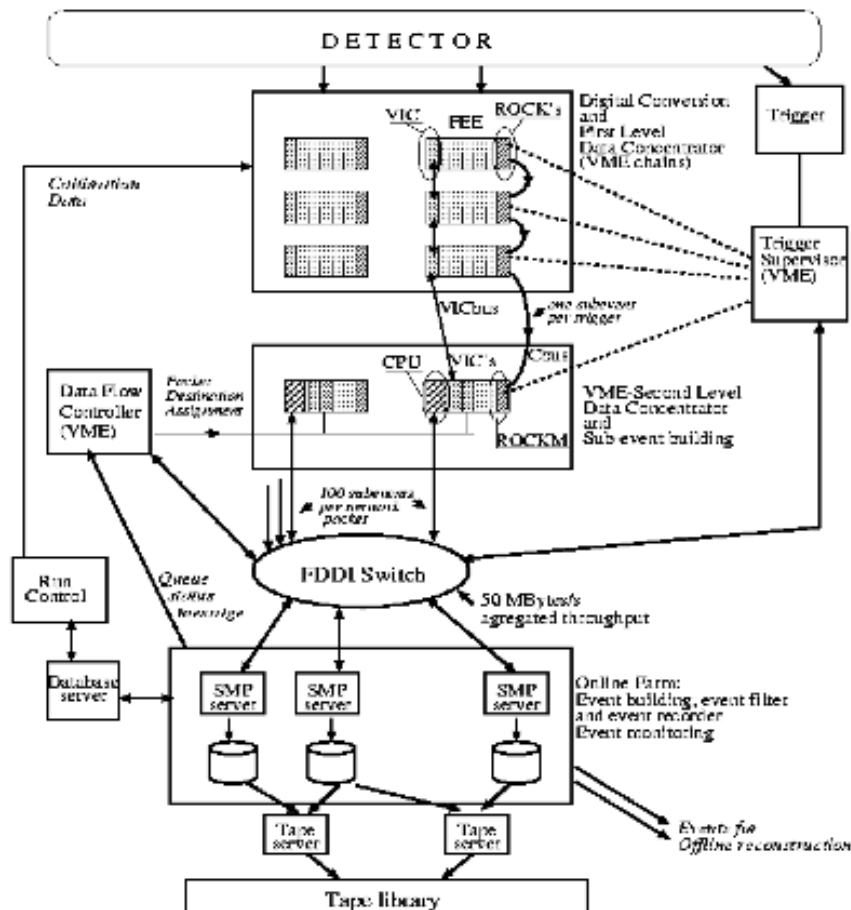
# A counter-example: Kloe

- DAΦNE e<sup>+</sup>e<sup>-</sup> collider in Frascati
- CP violation parameters in the Kaon system
- “factory”: rare events in a high rate beam



- $10^5$  channels
- 2.7ns crossing rate
  - large event overlap
- single interaction
  - “double hit” rejection
- L1  $\sim 10^4$  Hz  
2 $\mu$ s fixed dead time
- L2  $\sim 10^4$  Hz  
~COTS, background rejection only
- 5kB/ev  $\rightarrow$  50MB/s

# A counter-example: Kloe



- High rate of small events
- Fixed L1 dead time:  $2\mu\text{s}$
- synchronous
- not so much need for buffering at FE
- **push** architecture vs pull used in ATLAS  
*see Software lecture*
- no backpressure

# Trends

- Integrate synchronous, low latency in the front end
  - the limitations discussed do not disappear, but become “local”
  - all-HW implementation
  - isolated in a replaceable component
- Use networks as soon as possible
- Deal with dataflow instead of latency
- Use COTS network and processing
- Use “network” design already at small scale
  - easily get high performance with commercial components

- *(6) It is easier to move a problem around (for example, by moving the problem to a different part of the overall [network] architecture) than it is to solve it.  
(6a) (corollary). It is always possible to add another level of indirection.*

*RFC 1925 The Twelve [Networking] Truths*

# Back to basics ?

- *(12) In [protocol] design, perfection has been reached not when there is nothing left to add, but when there is nothing left to take away.*

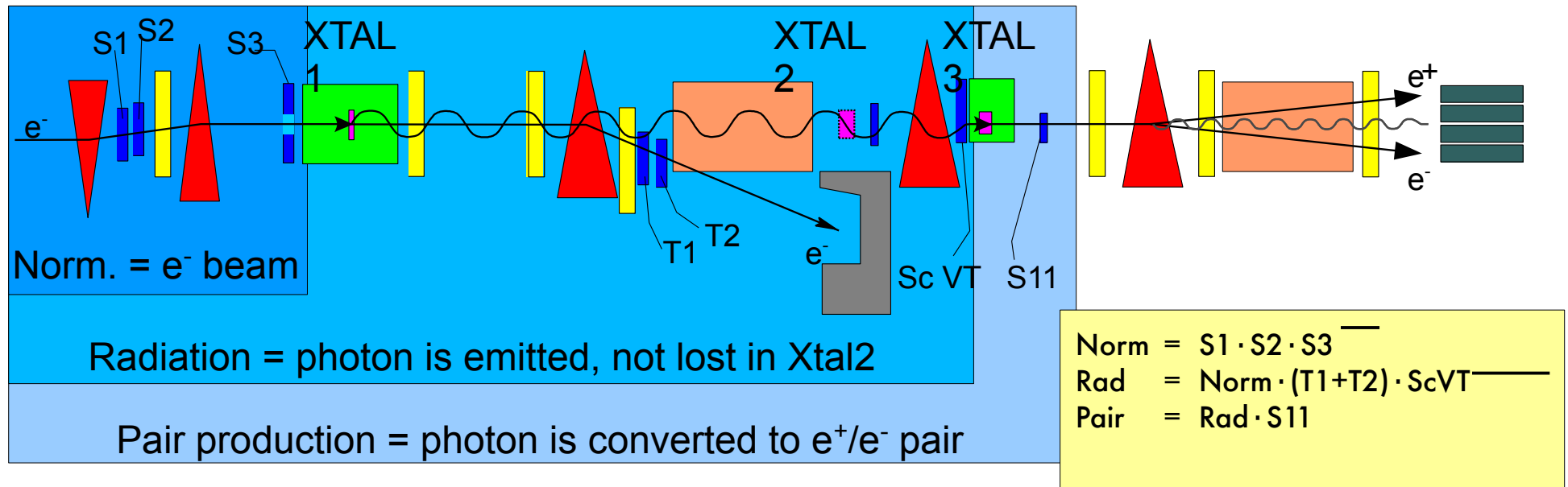
*RFC 1925 The Twelve [Networking] Truths*

- After adding all these levels of buffering, indirection, preselection, pre-preselection..
- What if we threw it all away?
- And we looked instead (e.g. for next generation Linear Colliders) towards “triggerless” systems, where all data flows to Event Building, and selection is done fully in software?

*See e.g. Patrick Le Dû @SNOWMASS 2001*

# SPARE SLIDES

# NA59 Trigger - physical view

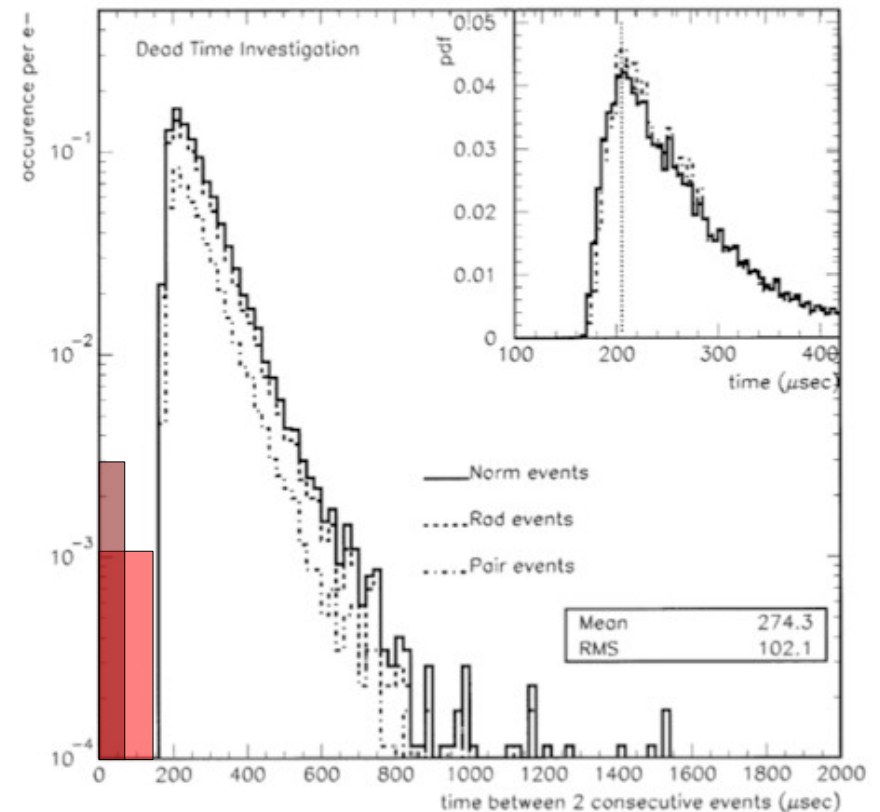


- Different types of events get different prescaling before readout
  - Give more chances to interesting (Rad, Pair) events, reduce storage
- Add calibration events in the mix
- Reject event if another particle arrives within drift time of DCs
  - Would not be distinguishable – so no central drift chambers at LHC exp.
- Fully implemented in HW
  - discrete NIM modules, about 2 crates



# NA59: Validate Trigger & DAQ

- Instrument your DAQ for performance
  - But careful because `gettimeofday` yields!
- Check dead time via  $\Delta t_{\text{event}}$ 
  - Most Probable 205 $\mu\text{s}$ , avg 275 $\mu\text{s}$
  - minimum 170 $\mu\text{s}$
  - VME readout time 160 $\mu\text{s}$  (bus analyzer)
  - 60 $\mu\text{s}$  CAMAC ADC (Lecroy 2249A)
- Compare with real rates
  - Scalers with no busy veto



# NA59 Validate Trigger & DAQ

- Instrument your DAQ for performance
  - But careful because gettimeofday yields!
- Check dead time via  $\Delta t_{\text{event}}$ 
  - Most Probable 205 $\mu\text{s}$ , avg 275 $\mu\text{s}$
  - minimum 170 $\mu\text{s}$
  - VME readout time 160 $\mu\text{s}$  (bus analyzer)
  - 60 $\mu\text{s}$  CAMAC ADC (Lecroy 2249A)
- Compare with real rates
  - Scalers with no busy veto
- Compare for different trigger types (*dem*
- Analyse minimum-bias (NORM) events to check that the HW trigger cuts actually behave as expected

