Trigger and DAQ

Alberto Annovi
INFN Pisa

Acknowledgments

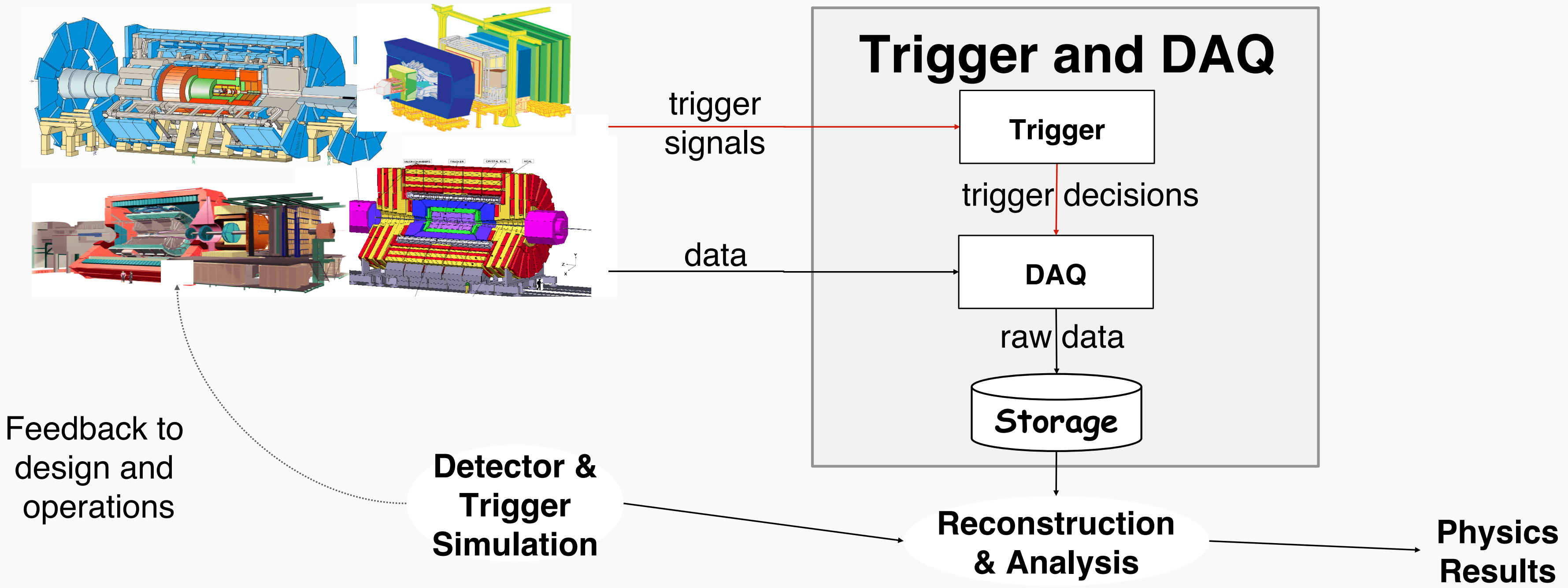
- Most material courtesy of **Andrea Negri**, and **Alessandro Thea**
- Errors are mine

Outline

- T&DAQ Introduction
- DAQ
  - Introduction
  - Basic DAQ concepts
  - Scaling up
  - Do it yourself
- Trigger
  - Introduction to trigger concepts

Overview

- Overall the main role of T & DAQ is to process the signals generated in a detector and saving the interesting information on a permanent storage



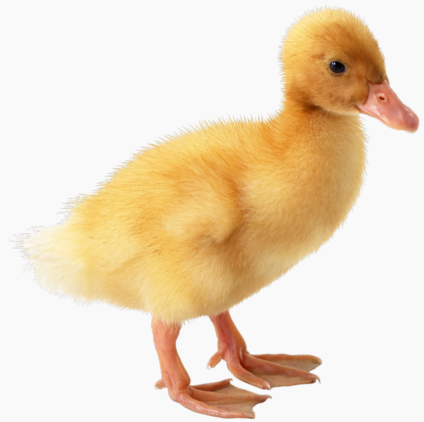
- I'll mostly refer to DAQ in High-Energy Physics

Trigger & DAQ

- Trigger
  - Either selects interesting events
    or rejects boring ones, in real time
  - i.e. with minimal *controlled* **latency**
    - time it takes to form and distribute its decision
- DAQ
  - Gathers data produced by detectors: **Readout**
  - Forms complete events:
    Data Collection and **Event Building**
  - Possibly feeds several trigger levels: **HLT**
  - Stores event data: **Data Logging**
  - Provides **Run Control**, **Configuration**, **Monitoring**

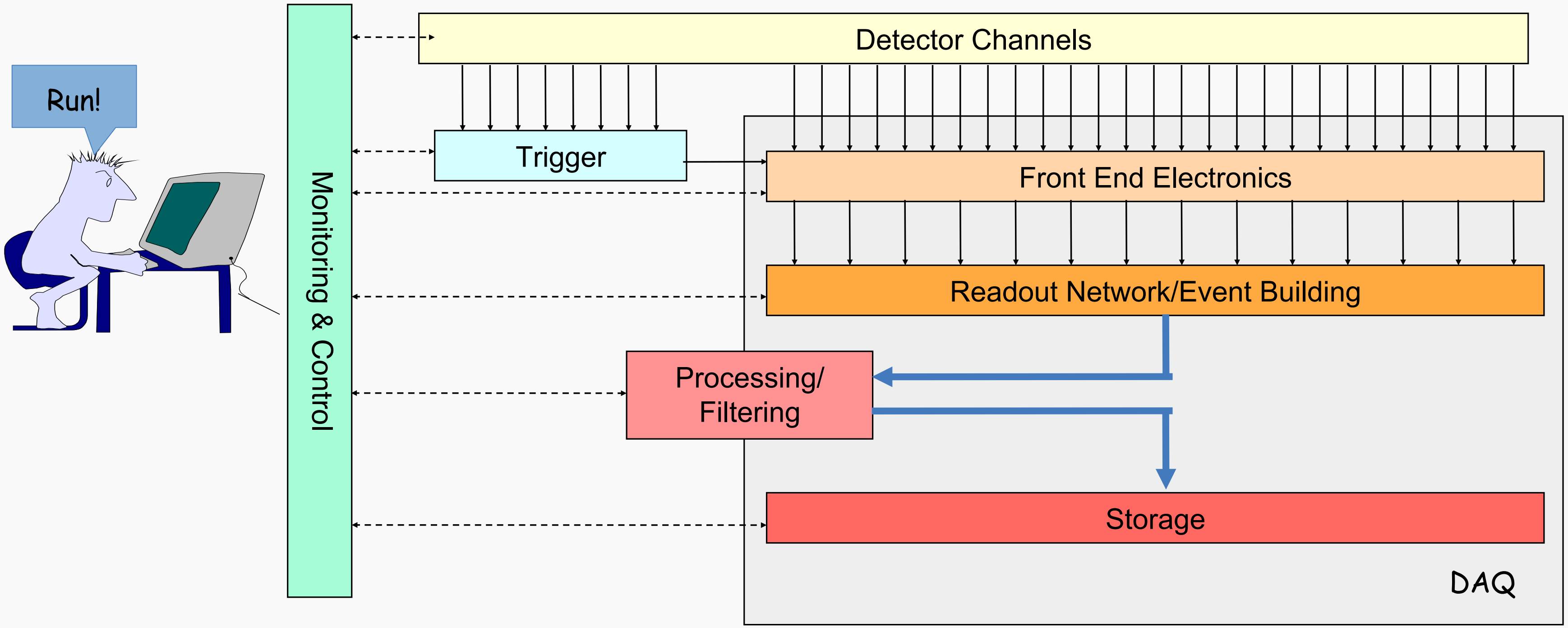
Trigger & DAQ

- Trigger
  - Either selects interesting events
    or rejects boring ones, in real time
  - i.e. with minimal *controlled* **latency**
    - time it takes to form and distribute its decision



- DAQ
  - Gathers data produced by detectors: **Readout**
  - Forms complete events:
    Data Collection and **Event Building**
  - Possibly feeds several trigger levels: **HLT**
  - Stores event data: **Data Logging**
  - Provides **Run Control**, **Configuration**, **Monitoring**

Trigger, DAQ and Controls



DAQ Introduction

- Data AcQuisition is an heterogeneous field
  - An alchemy of physics, electronics, networking, hacking and experience
  - Boundaries not well defined
  - … money and personnel matter as well

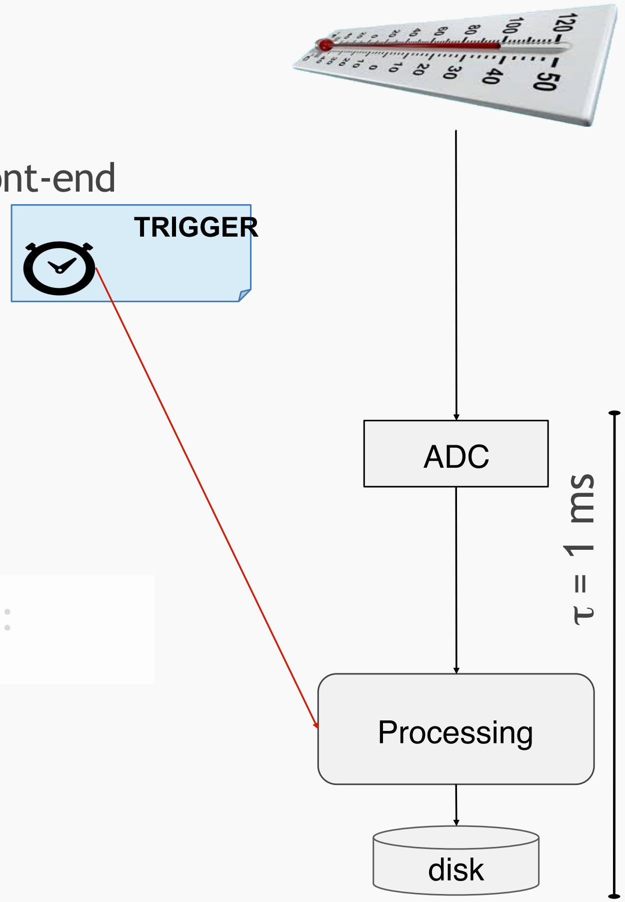
- Aim of this part is to introduce the **basic DAQ concepts** avoiding as many technological details as possible

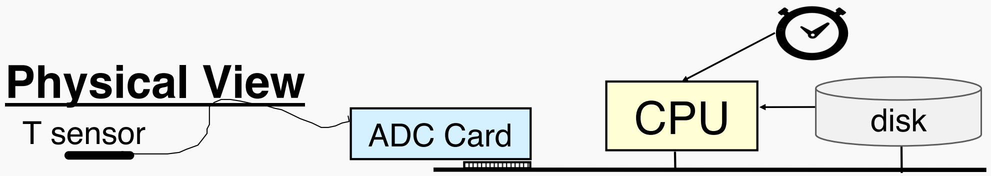
DAQ Outline

- Basic DAQ concepts
  - Digitization, Latency
  - Deadtime, Busy, Backpressure
  - De-randomization
- Scaling up
  - Readout and Event Building
  - Buses vs Network
- Do it yourself

Basic DAQ: periodic trigger

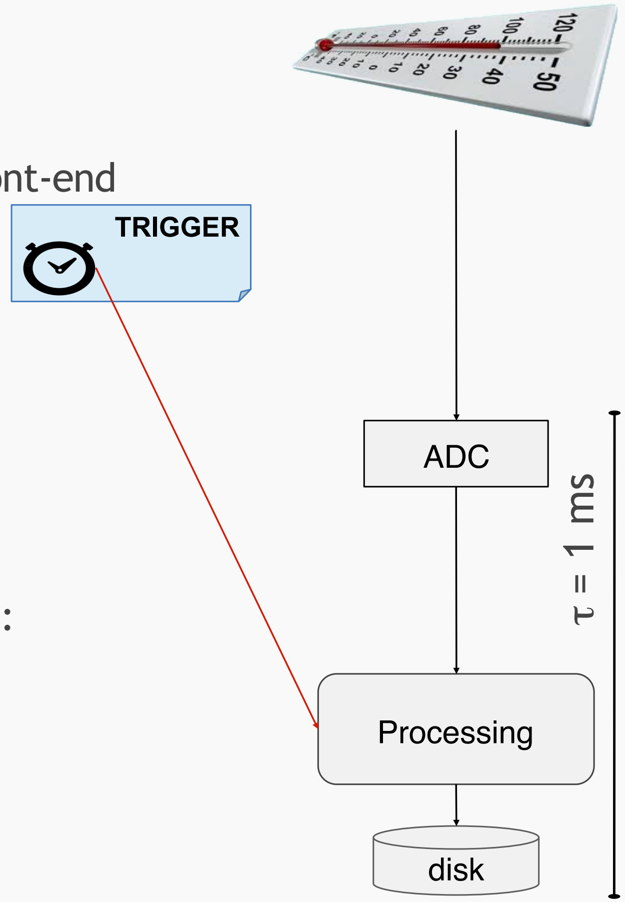
- Es: measure temperature at a fixed frequency
  - ADC performs analog to digital conversion, digitization (our front-end electronics)
  - CPU does readout and processing
- System clearly limited by the time τ to process an "event"
  - ADC conversion + CPU processing + Storage
- The DAQ maximum sustainable rate is simply the inverse of τ, e.g.:
  - τ = 1 ms $\rightarrow$ R = 1/τ = 1 kHz

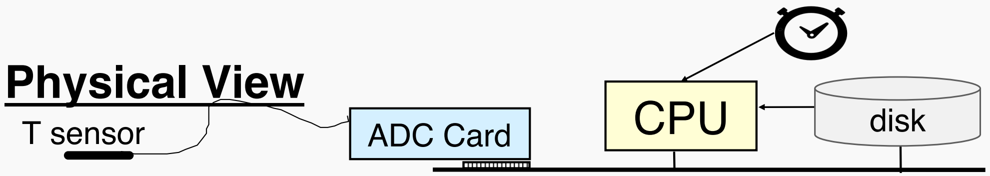




Basic DAQ: periodic trigger

- Es: measure temperature at a fixed frequency
  - ADC performs analog to digital conversion, digitization (our front-end electronics)
  - CPU does readout and processing
- System clearly limited by the time τ to process an "event"
  - ADC conversion + CPU processing + Storage
- The DAQ maximum sustainable rate is simply the inverse of τ, e.g.:
  - $\tau = 1$ ms $\rightarrow R = 1/\tau = 1$ kHz

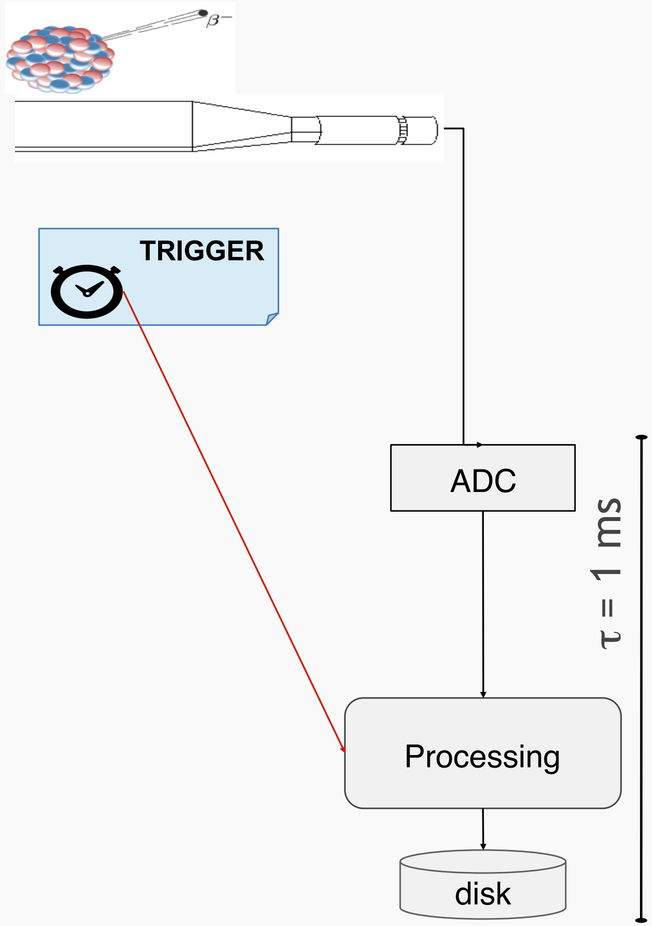




Basic DAQ: “real” trigger

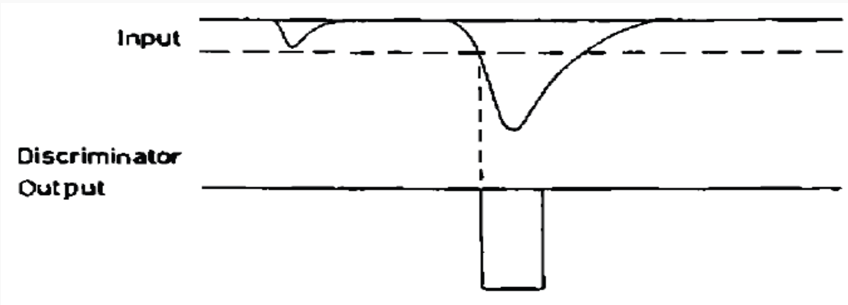
- Events asynchronous and unpredictable
  - E.g.: beta decay studies
- A physics trigger is needed
  - **Discriminator**: generate an output signal only if amplitude of input pulse is greater than a certain threshold

  - delay introduced to compensate for the **trigger latency**

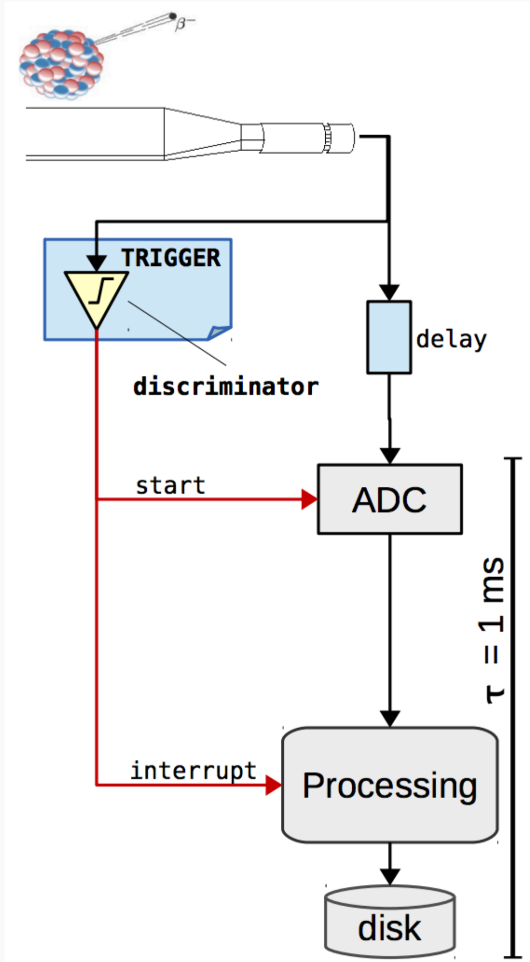


Basic DAQ: “real” trigger

- Events asynchronous and unpredictable
  - E.g.: beta decay studies
- A physics trigger is needed
  - **Discriminator**: generate an output signal only if amplitude of input pulse is greater than a certain threshold

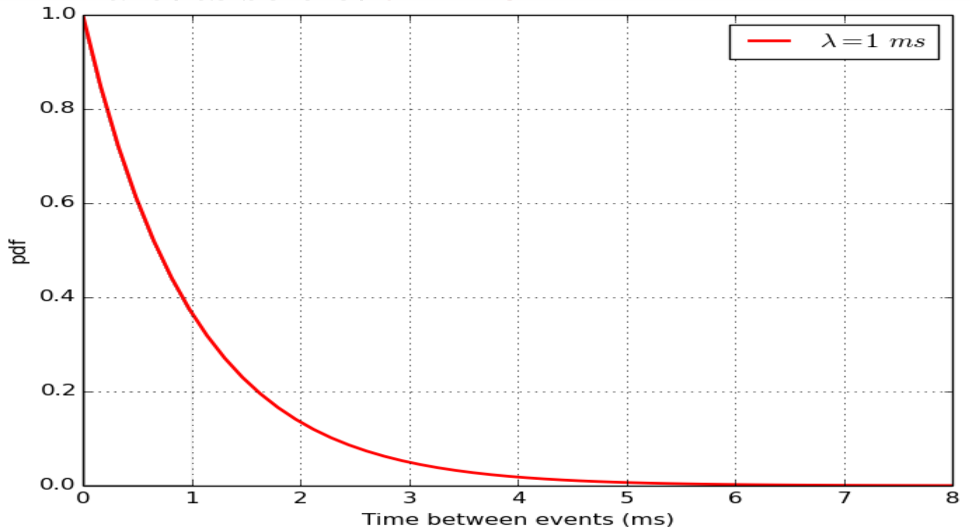


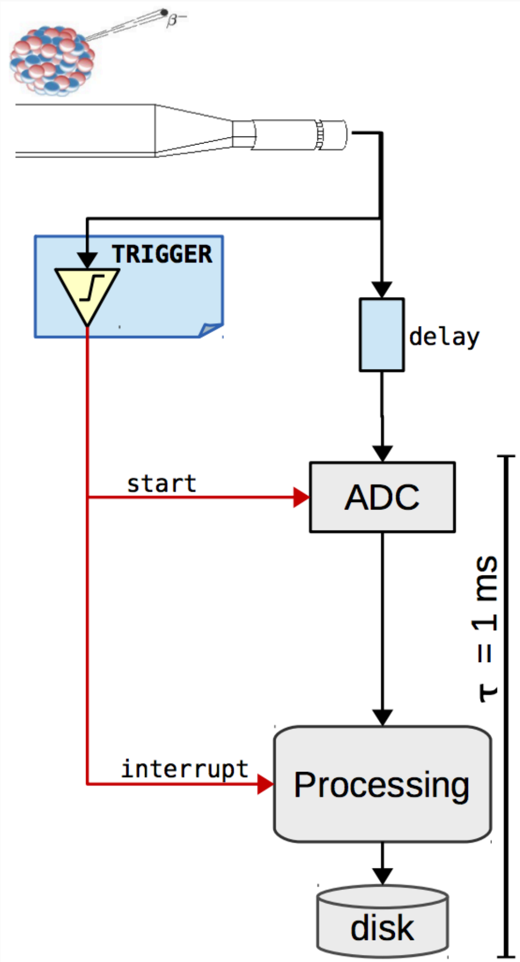
  - delay introduced to compensate for the **trigger latency**



Basic DAQ: “real” trigger

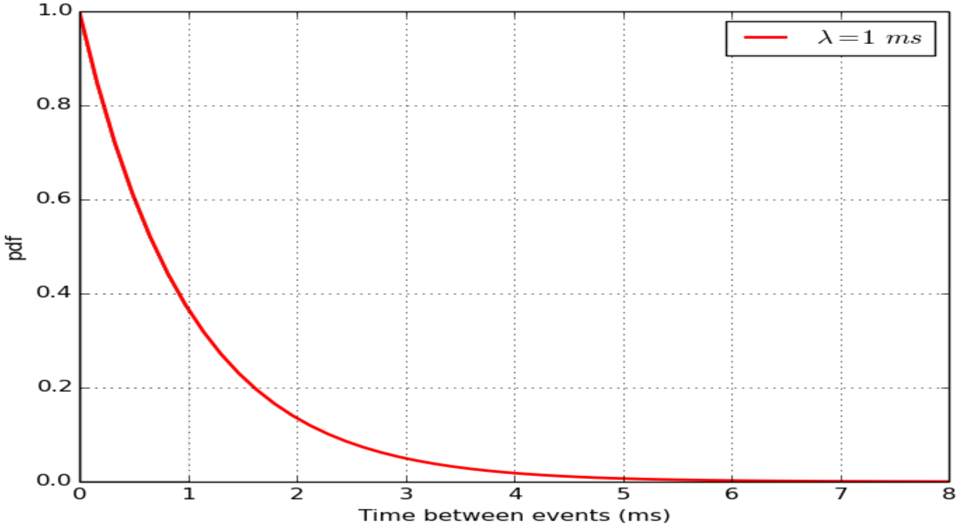
- Stochastic process
  - Fluctuations in time between events
- Let's assume for example
  - a process rate f = 1 kHz, i.e. λ = 1 ms
  - and, as before, τ = 1 ms

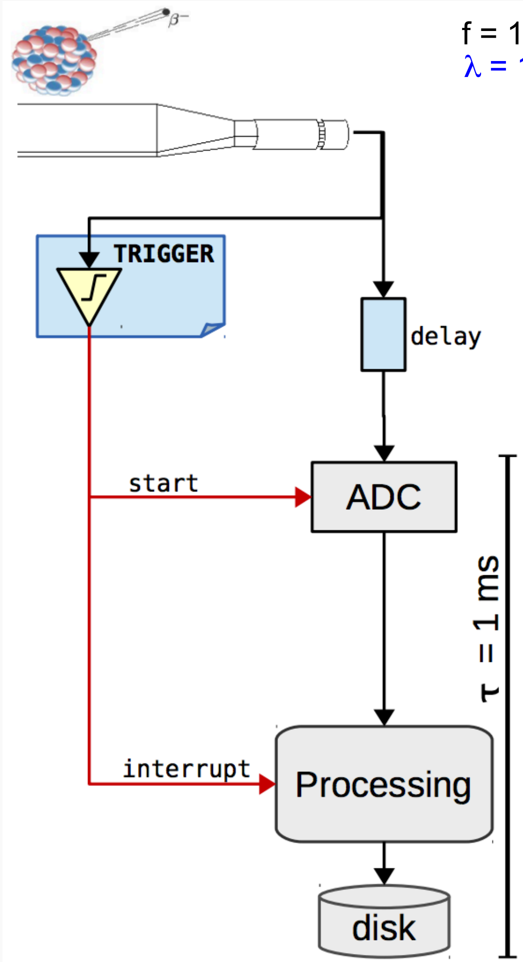




Basic DAQ: “real” trigger

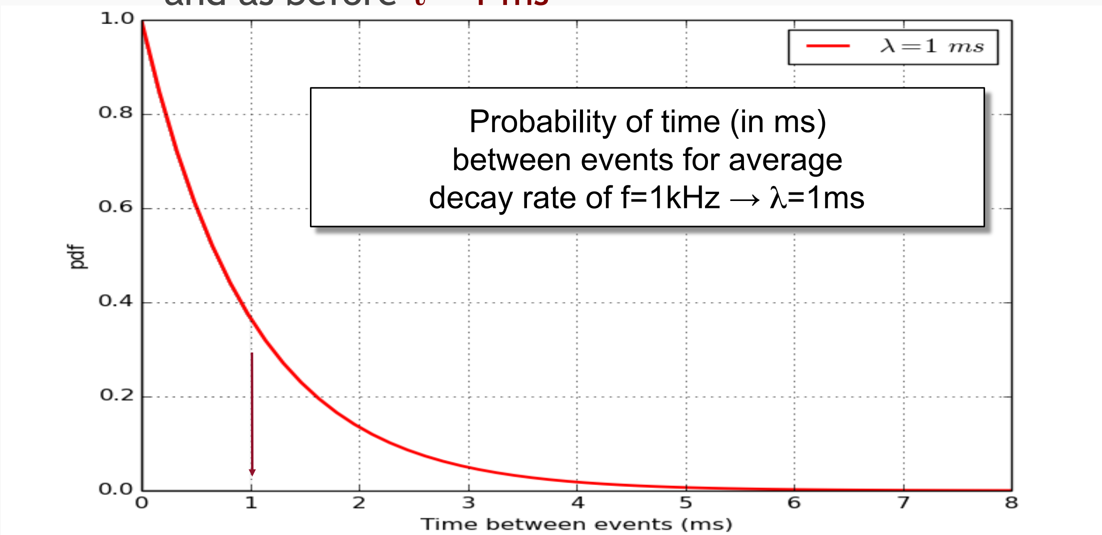
- Stochastic process
  - Fluctuations in time between events
- Let's assume for example
  - a process rate f = 1 kHz, i.e. λ = 1 ms
  - and as before τ = 1 ms

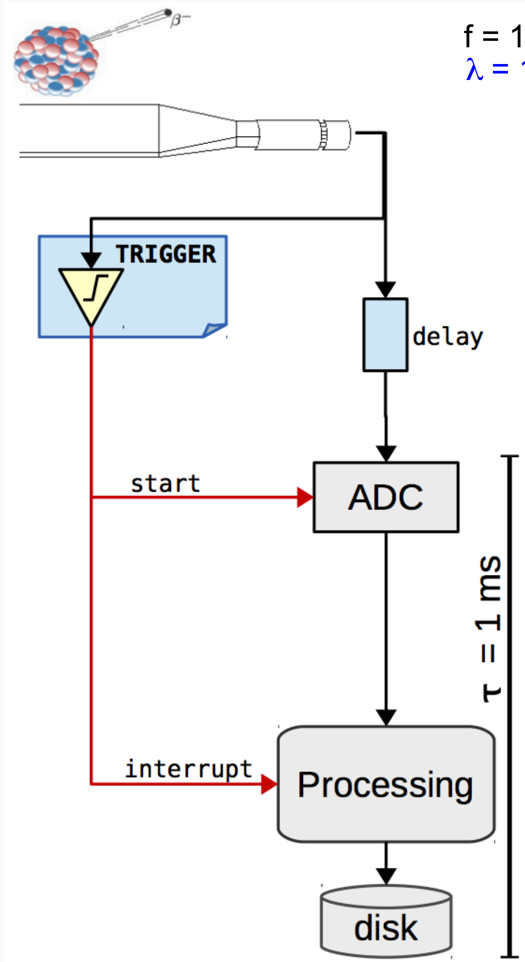




Basic DAQ: “real” trigger

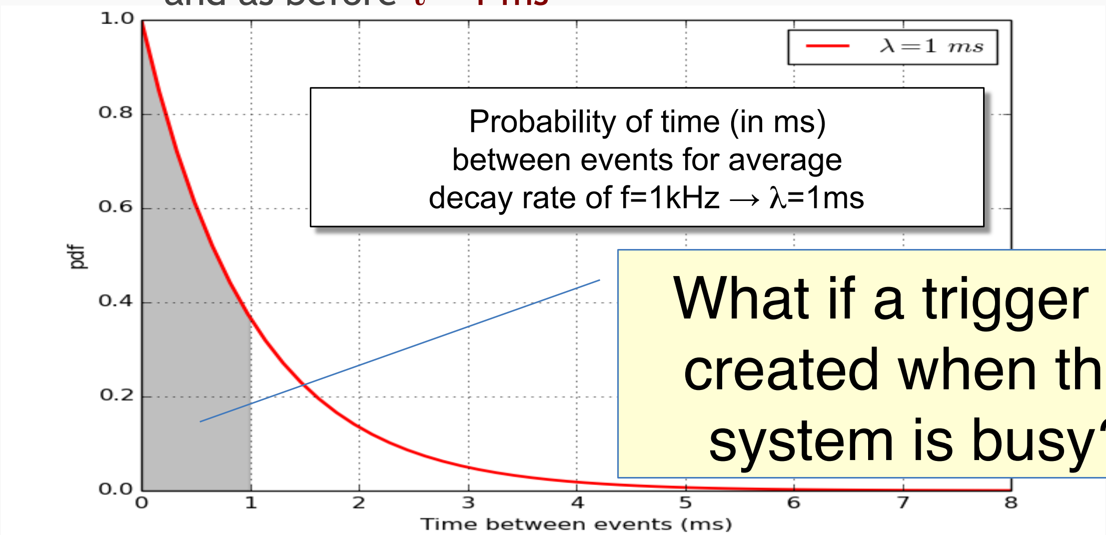
- Stochastic process
  - Fluctuations in time between events
- Let's assume for example
  - a process rate f = 1 kHz, i.e. λ = 1 ms
  - and as before τ = 1 ms



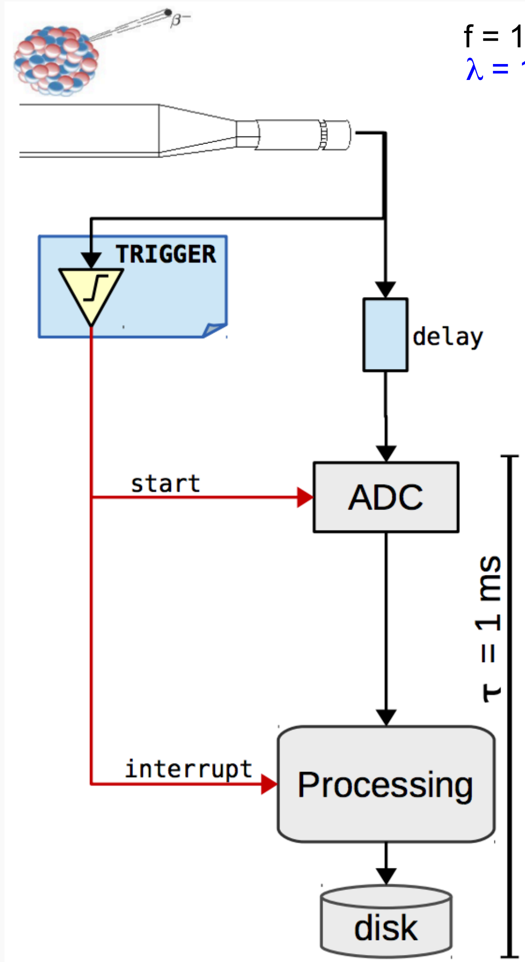


Basic DAQ: “real” trigger

- Stochastic process
  - Fluctuations in time between events
- Let's assume for example
  - a process rate f = 1 kHz, i.e. λ = 1 ms
  - and as before τ = 1 ms

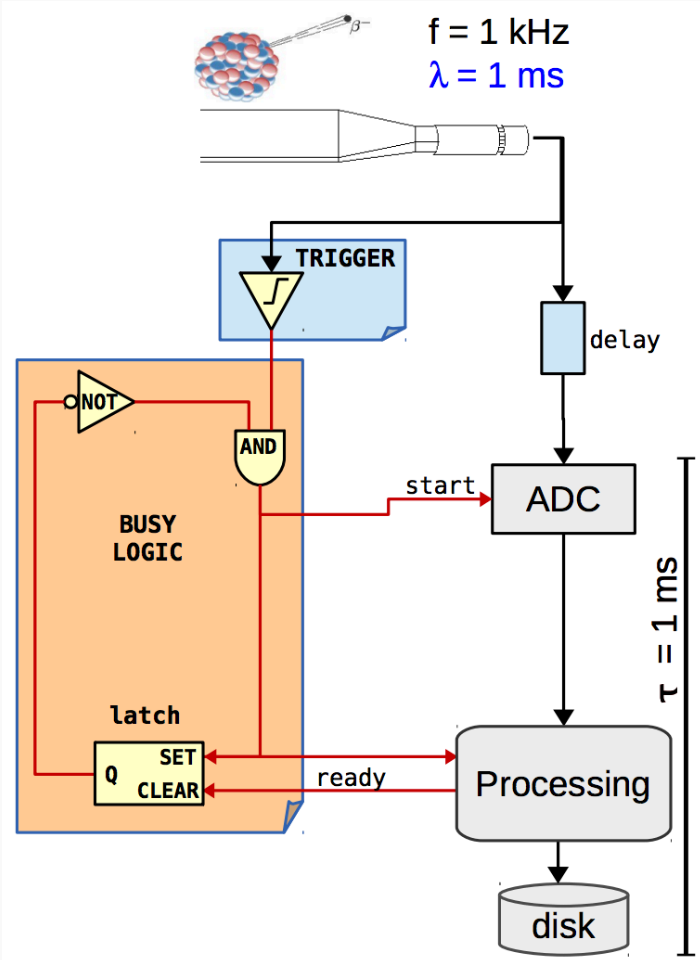


What if a trigger is created when the system is busy?



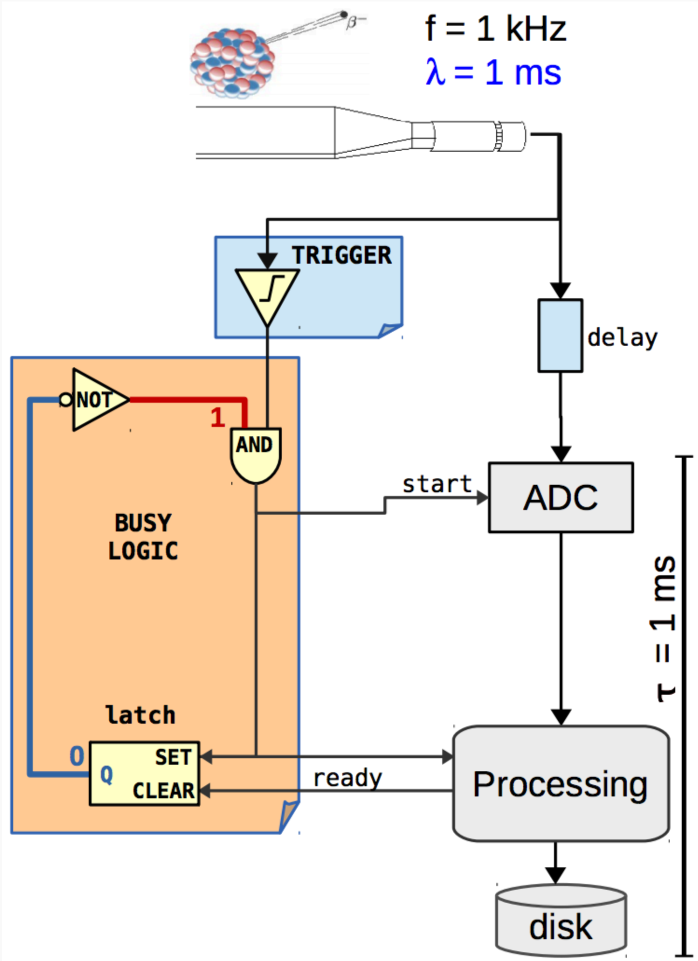
Basic DAQ: "real" trigger

- **Busy logic** avoids triggers while the system is busy in processing
  - E.g.: AND port and a latch
- Latch (**flip-flop**):
  - a bistable circuit that changes state (Q) by signals applied to the control inputs (SET, CLEAR)



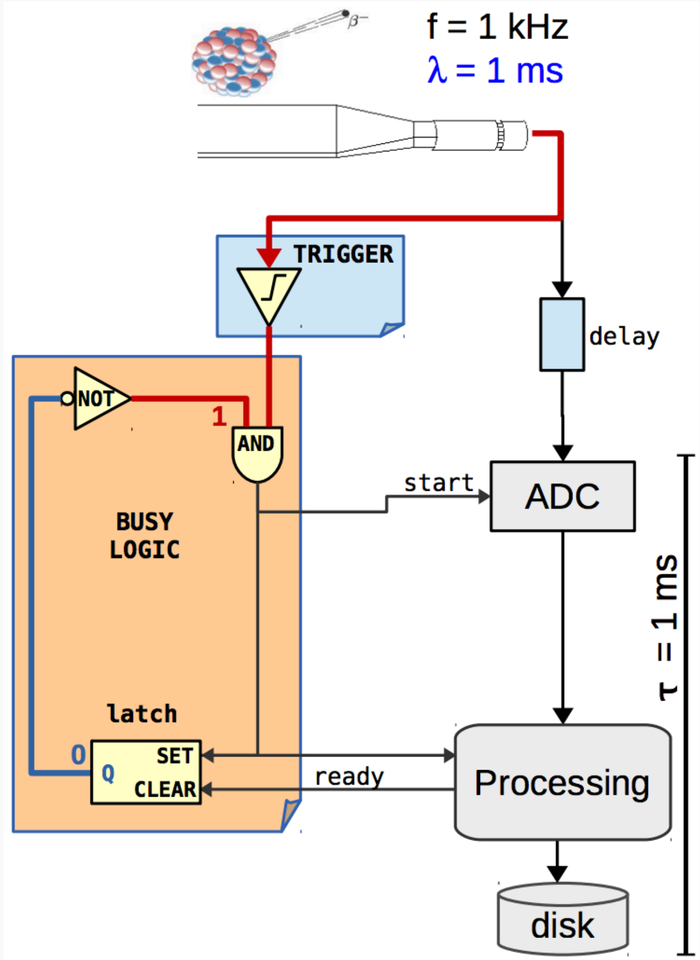
Basic DAQ: “real” trigger

- **Busy logic** avoids triggers while the system is busy in processing
  - E.g.: AND port and a latch
- Latch (**flip-flop**):
  - a bistable circuit that changes state (Q) by signals applied to the control inputs (SET, CLEAR)
  - At the beginning the flip-flop state is down and so one input of the AND port is always up



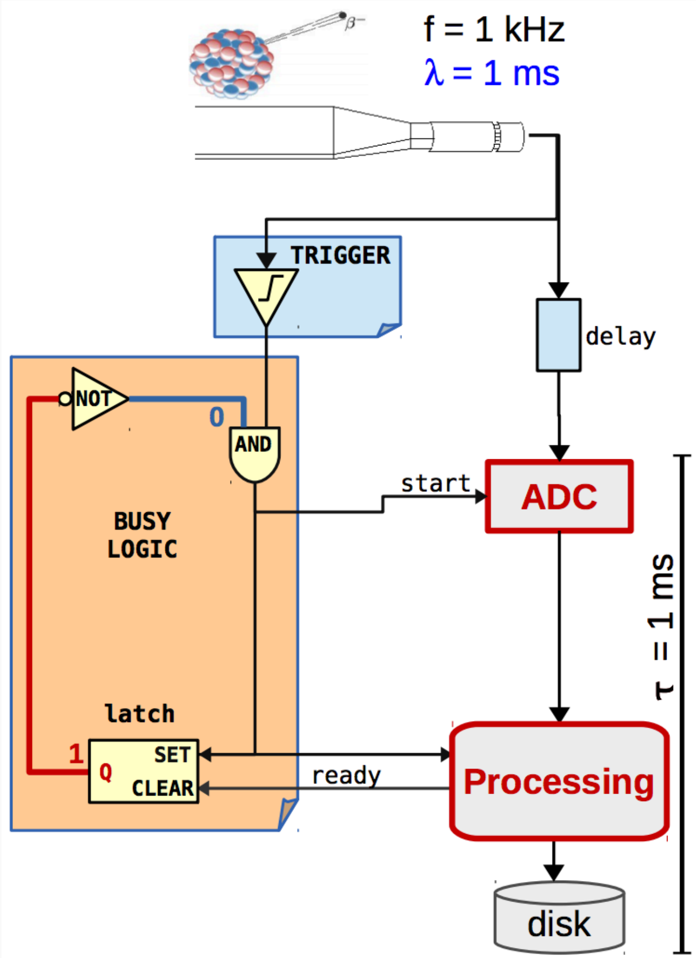
Basic DAQ: “real” trigger

- **Busy logic** avoids triggers while the system is busy in processing
  - E.g.: AND port and a latch
- Latch (**flip-flop**):
  - a bistable circuit that changes state (Q) by signals applied to the control inputs (SET, CLEAR)
  - If a trigger arrives, the AND port is open
    - ADC is started
    - Flip-flop is flipped



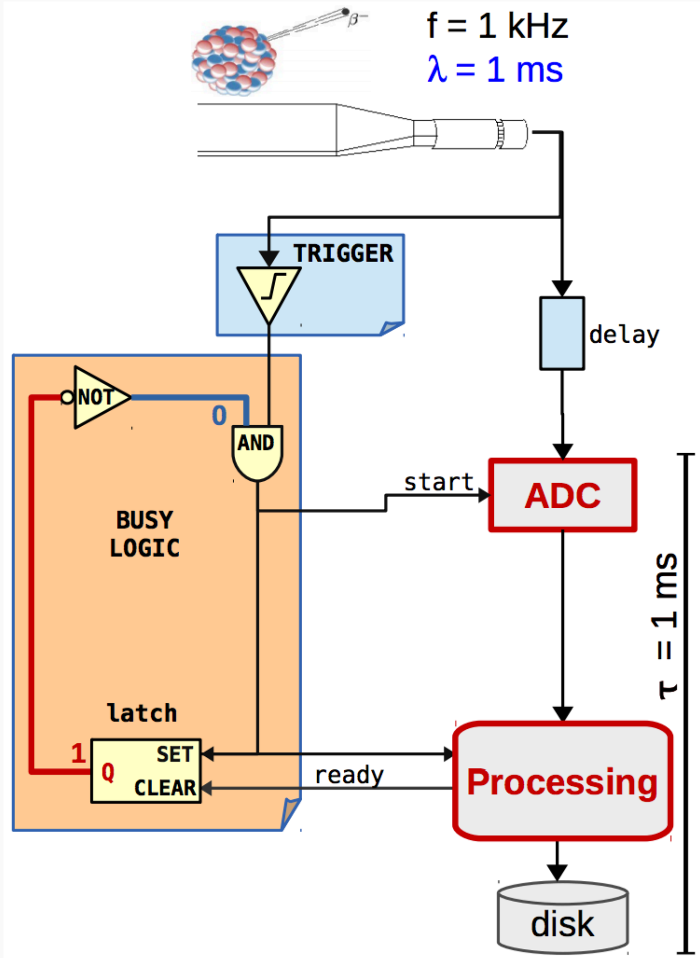
Basic DAQ: “real” trigger

- **Busy logic** avoids triggers while the system is busy in processing
  - E.g.: AND port and a latch
- Latch (**flip-flop**):
  - a bistable circuit that changes state (Q) by signals applied to the control inputs (SET, CLEAR)
  - If a trigger arrives, the AND port is open
    - ADC is started
    - Flip-flop is flipped



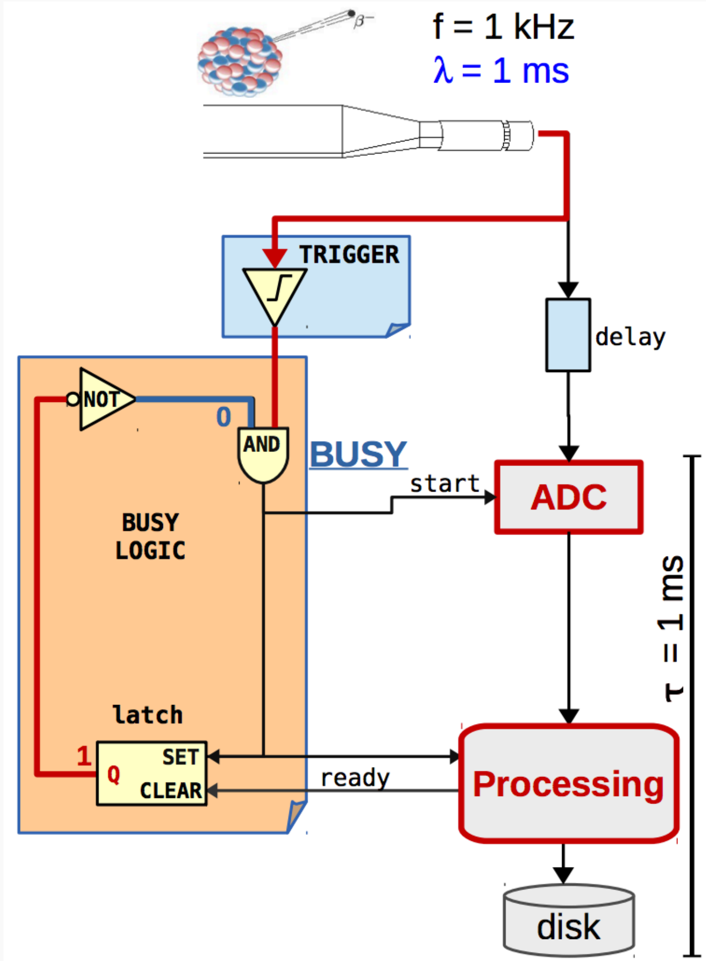
Basic DAQ: “real” trigger

- **Busy logic** avoids triggers while the system is busy in processing
  - E.g.: AND port and a latch
- Latch (**flip-flop**):
  - a bistable circuit that changes state (Q) by signals applied to the control inputs (SET, CLEAR)
  - Flip-flop is flipped
    - Now an AND input port is steadily down



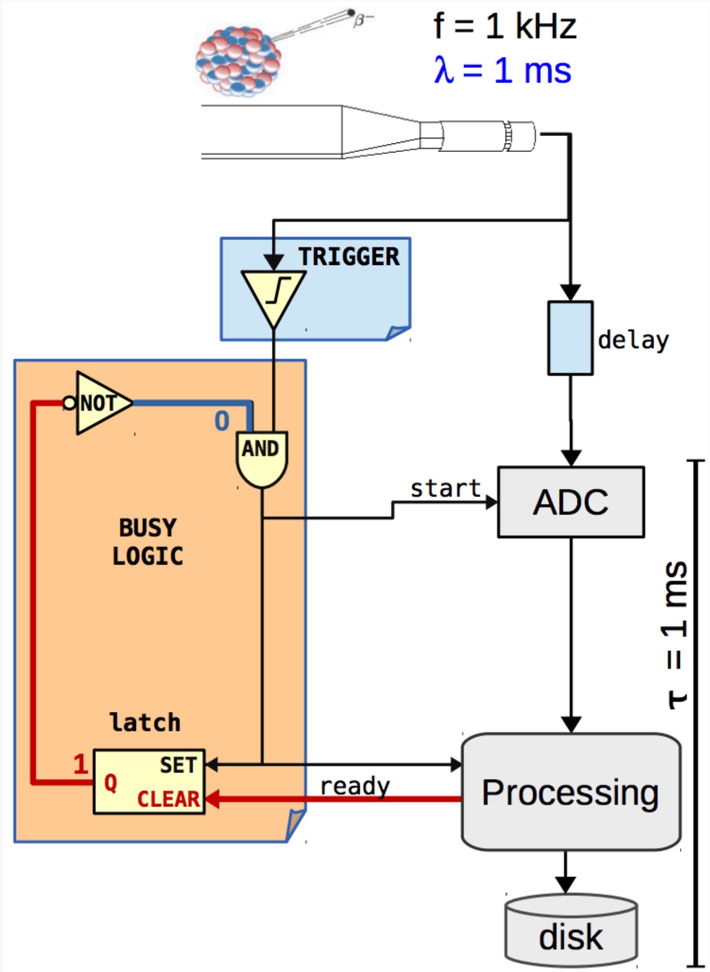
Basic DAQ: “real” trigger

- **Busy logic** avoids triggers while the system is busy in processing
  - E.g.: AND port and a latch
- Latch (**flip-flop**):
  - a bistable circuit that changes state (Q) by signals applied to the control inputs (SET, CLEAR)
  - Any new trigger is inhibited by the AND port
    - busy



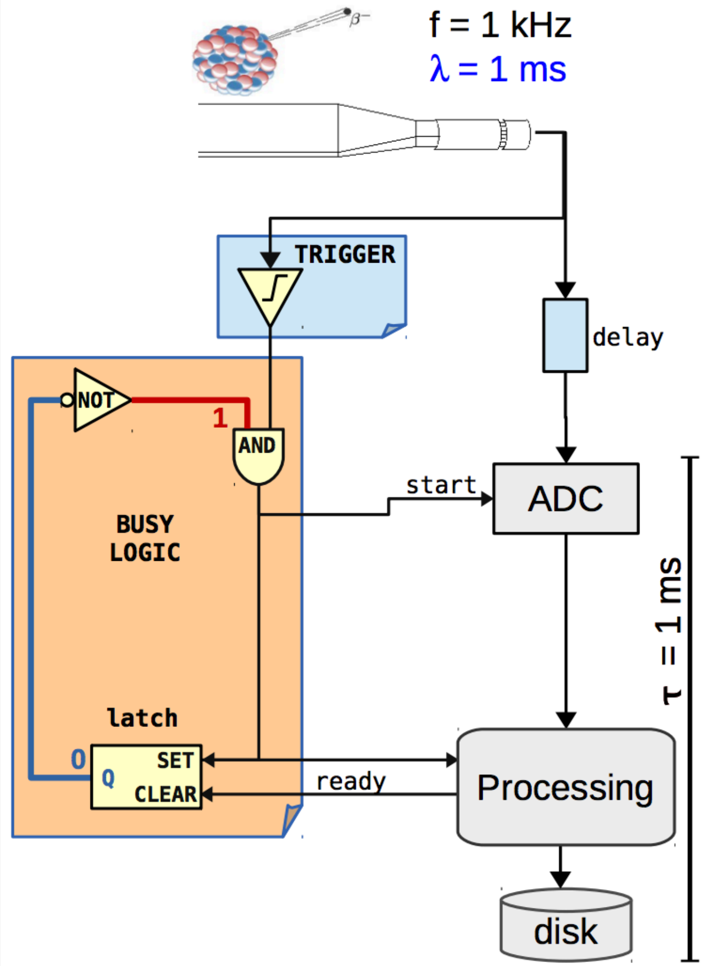
Basic DAQ: “real” trigger

- **Busy logic** avoids triggers while the system is busy in processing
  - E.g.: AND port and a latch
- Latch (**flip-flop**):
  - a bistable circuit that changes state (Q) by signals applied to the control inputs (SET, CLEAR)
  - At the end of processing flip-flop is flipped again
    - The system is ready to accept a new trigger

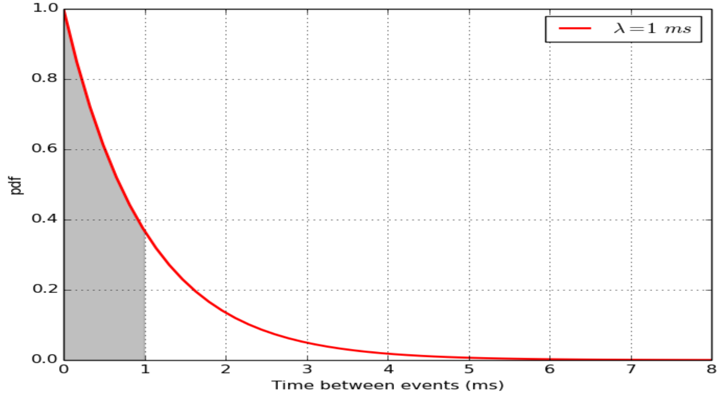


Basic DAQ: "real" trigger

- **Busy logic** avoids triggers while the system is busy in processing
  - E.g.: AND port and a latch
- Latch (**flip-flop**):
  - a bistable circuit that changes state (Q) by signals applied to the control inputs (SET, CLEAR)
  - At the end of processing flip-flop is flipped again
    - The system is ready to accept a new trigger



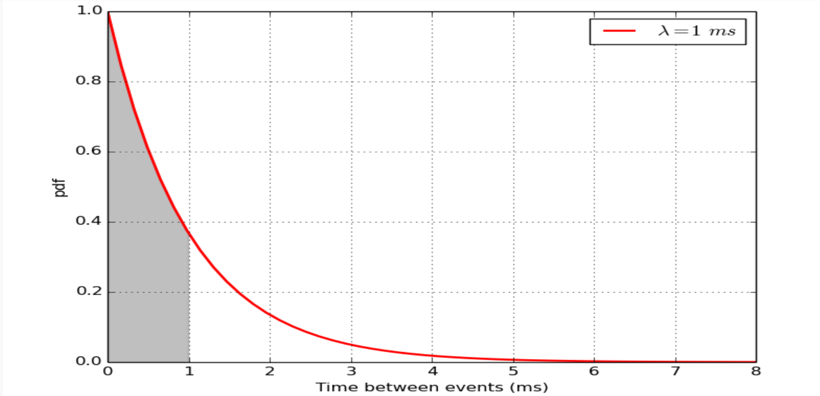
Deadtime and efficiency



- Which (average) DAQ rate can we achieve now?
  - Reminder: w/ a synchronous trigger τ = 1 ms the limit is 1 kHz
- Definitions
  - **f:** average rate of physics phenomenon (input)
  - ν: average rate of DAQ (output)
  - τ: **deadtime**, the time the system requires to process an event, without being able to handle other triggers
  - probabilities: P[busy] = $\nu\tau$; P[free] = 1 - $\nu\tau$
- Therefore:

$$\nu = f\,P[free] \Rightarrow \nu = f(1-\nu\tau) \Rightarrow \nu = \frac{f}{1+f\tau}$$

Deadtime and efficiency



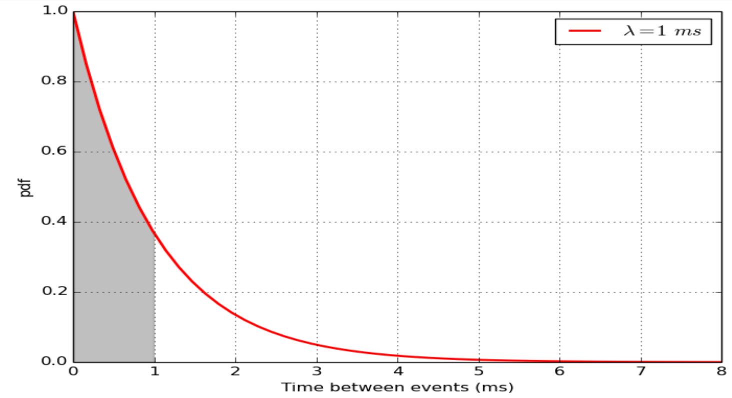
- Which (average) DAQ rate can we achieve now?
  - Reminder: w/ a synchronous trigger τ = 1 ms the limit is 1 kHz

- Definitions
  - **f:** average rate of physics phenomenon (input)
  - ν: average rate of DAQ (output)
  - τ: **deadtime**, the time the system requires to process an event, without being able to handle other triggers
  - probabilities: P[busy] = $\nu\,\tau$; P[free] = 1 - $\nu\,\tau$

- Therefore:

$$\nu = f\,P[free] \Rightarrow \nu = f(1-\nu\,\tau) \Rightarrow \nu = \frac{f}{1+f\,\tau}$$

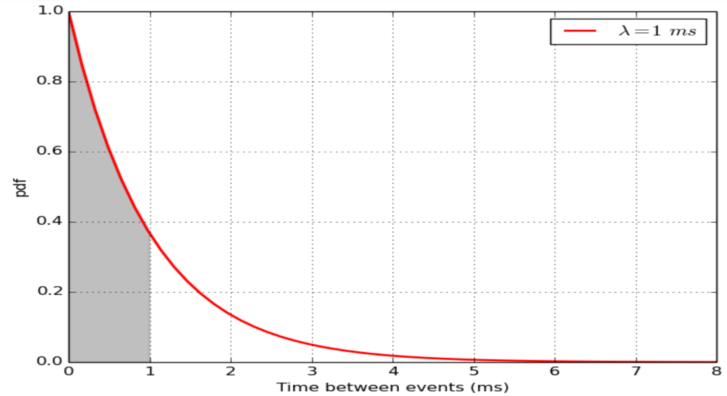
Deadtime and efficiency



- Which (average) DAQ rate can we achieve now?
  - Reminder: w/ a synchronous trigger τ = 1 ms the limit is 1 kHz
- Definitions
  - **f:** average rate of physics phenomenon (input)
  - ν: average rate of DAQ (output)
  - τ: **deadtime**, the time the system requires to process an event, without being able to handle other triggers
  - probabilities: P[busy] = $\nu\,\tau$;   P[free] = 1 - $\nu\,\tau$
- Therefore:

$$\nu = f\,P[free] \Rightarrow \nu = f(1 - \nu\tau) \Rightarrow \nu = \frac{f}{1 + f\tau}$$

Deadtime and efficiency



- Which (average) DAQ rate can we achieve now?
  - Reminder: w/ a synchronous trigger τ = 1 ms the limit is 1 kHz

- Definitions
  - **f:** average rate of physics phenomenon (input)
  - ν: average rate of DAQ (output)
  - τ: **deadtime**, the time the system requires to process an event, without being able to handle other triggers
  - probabilities: P[busy] = $\nu\,\tau$; P[free] = 1 - $\nu\,\tau$

- Therefore:

$$\nu = f\,P[free] \Rightarrow \nu = f(1 - \nu\,\tau) \Rightarrow \nu = \frac{f}{1 + f\,\tau}$$

Deadtime and efficiency

- Due to stochastic fluctuations
  - DAQ rate always < physics rate $\quad \nu = \dfrac{f}{1+f\tau} < f$
  - Efficiency always < 100% $\quad \epsilon = \dfrac{N_{saved}}{N_{tot}} = \dfrac{1}{1+f\tau} < 100\%$

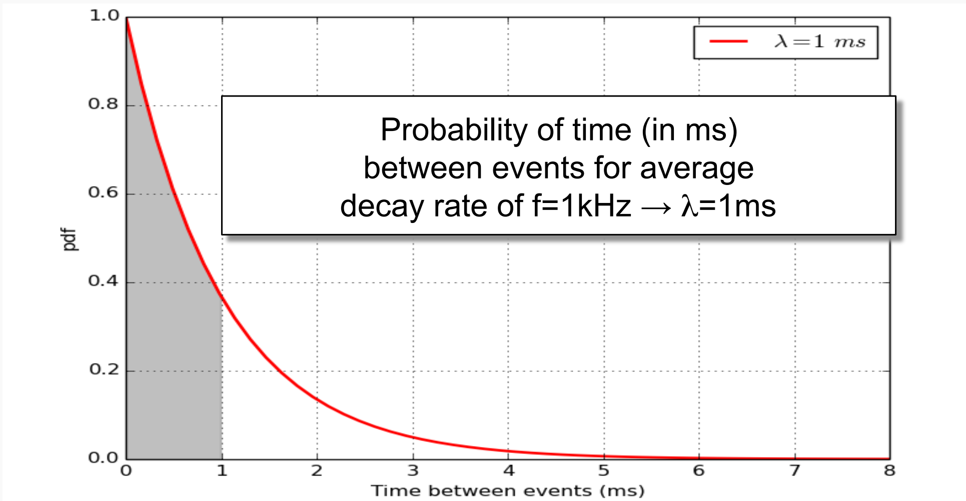
- So, in our specific example

$$\left| \begin{array}{l} f = 1\,kHz \\ \tau = 1\,ms \end{array} \right. \rightarrow \left| \begin{array}{l} \nu = 500\,Hz \\ \epsilon = 50\% \end{array} \right.$$

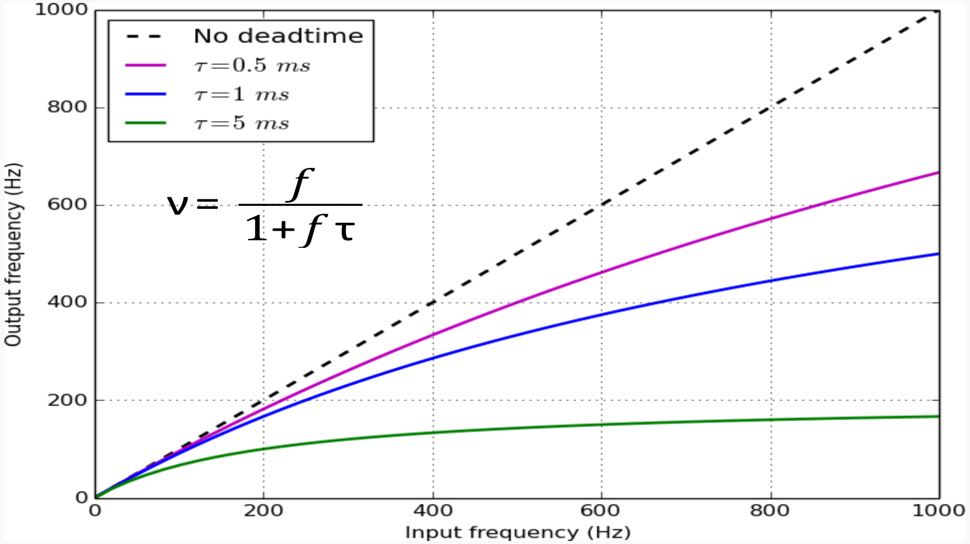
Deadtime and efficiency

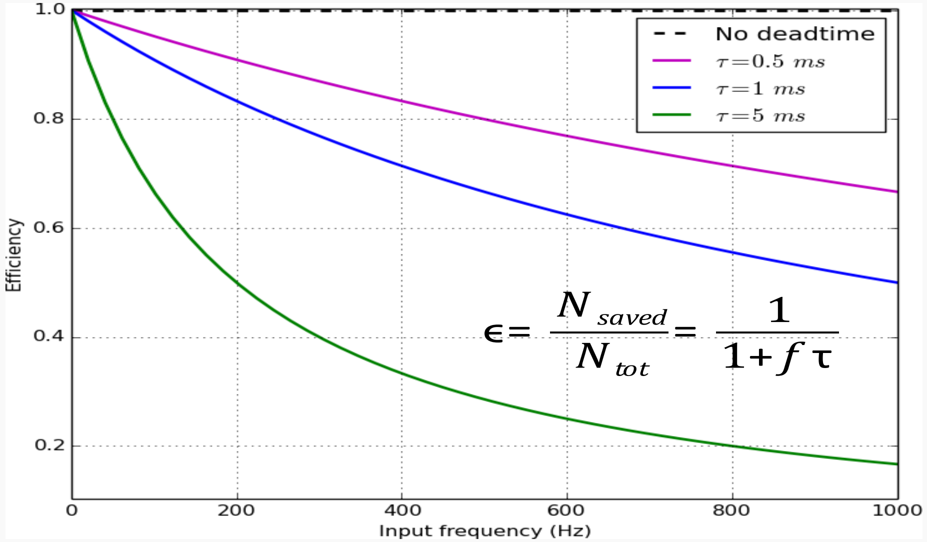
- Due to stochastic fluctuations
  - DAQ rate always < physics rate $$\nu = \frac{f}{1+f\tau} < f$$
  - Efficiency always < 100% $$\epsilon = \frac{N_{saved}}{N_{tot}} = \frac{1}{1+f\tau} < 100\%$$
- So, in our specific example

$$\left| \begin{array}{l} f = 1\,kHz \\ \tau = 1\,ms \end{array} \right. \rightarrow \left| \begin{array}{l} \nu = 500\,Hz \\ \epsilon = 50\% \end{array} \right.$$



Deadtime and efficiency

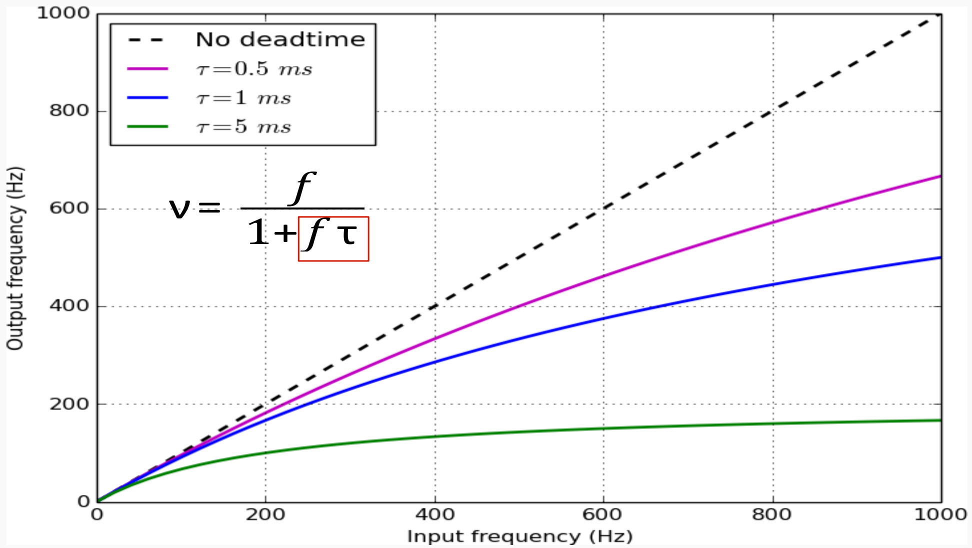


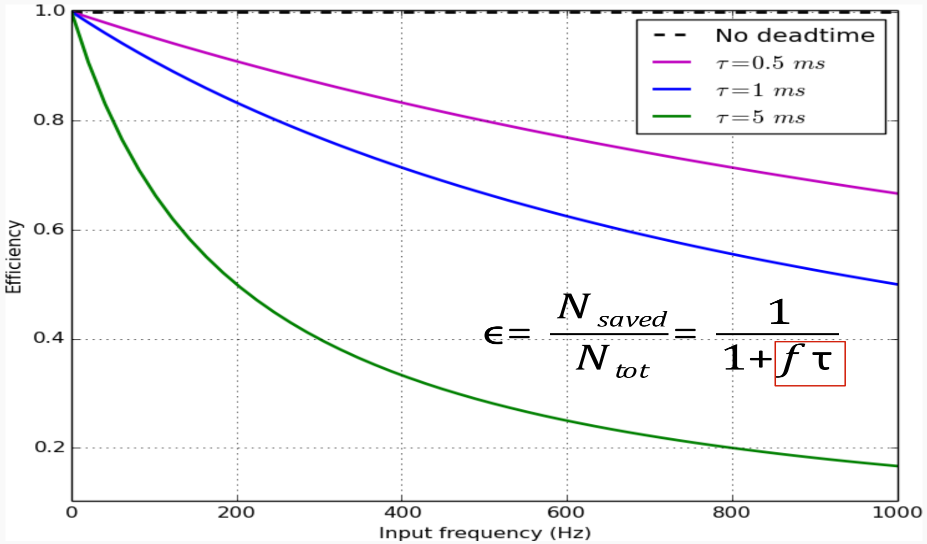


- In order to obtain ε~100% (i.e.: ν~f) → $f\tau << 1$ → $\tau << \lambda$
  - E.g.: ε~99% for f = 1 kHz → τ < 0.01 ms → $1/\tau$ > 100 kHz
  - To cope with the input signal fluctuations,
    we have to **over-design** our DAQ system by a factor 100!

- How can we mitigate this effect?

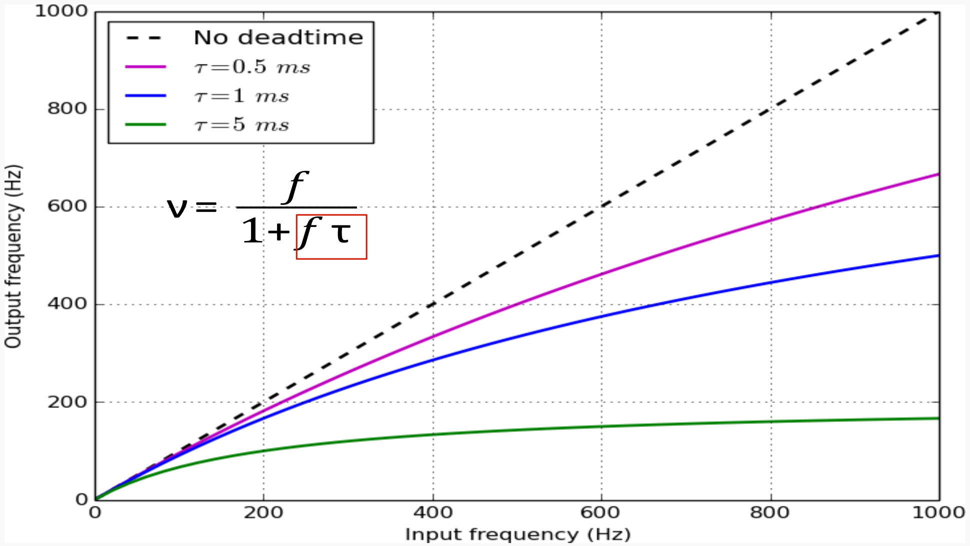
Deadtime and efficiency



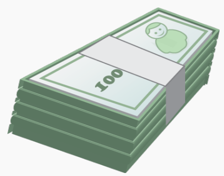


- In order to obtain ε~100% (i.e.: ν~f) $\rightarrow f\tau << 1 \rightarrow \tau << \lambda$
  - E.g.: ε~99% for f = 1 kHz $\rightarrow \tau$ < 0.01 ms $\rightarrow 1/\tau$ > 100 kHz
  - To cope with the input signal fluctuations,
    we have to **over-design** our DAQ system by a factor 100!
- How can we mitigate this effect?

Deadtime and efficiency



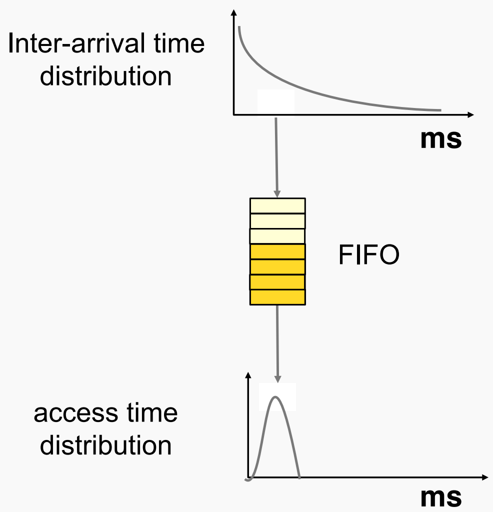
- In order to obtain ε~100% (i.e.: ν~f) → fτ << 1 → τ << λ
  - E.g.: ε~99% for f = 1 kHz → τ < 0.01 ms → $1/\tau$ > 100 kHz
  - To cope with the input signal fluctuations,
    we have to **over-design** our DAQ system **by a factor 100!**



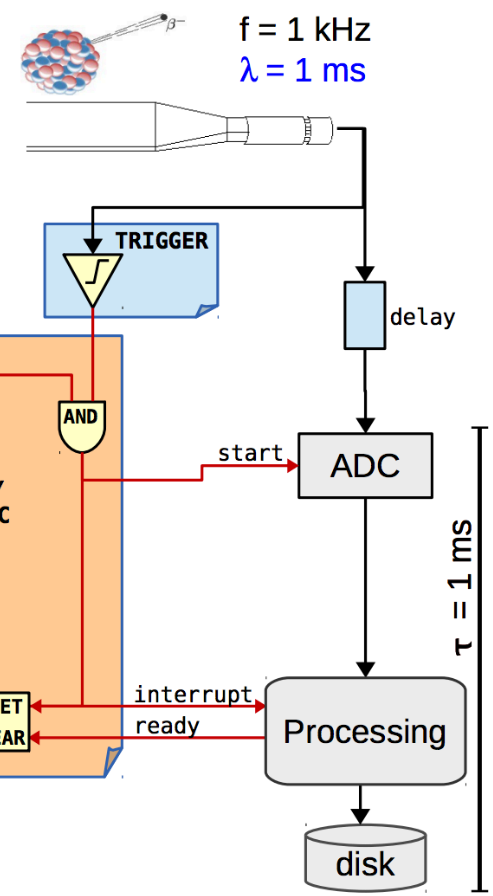
- How can we mitigate this effect?

De-randomization

- Input fluctuations can be absorbed and smoothed by a queue
  - A First In First Out can provide a ~steady and **de-randomized** output rate

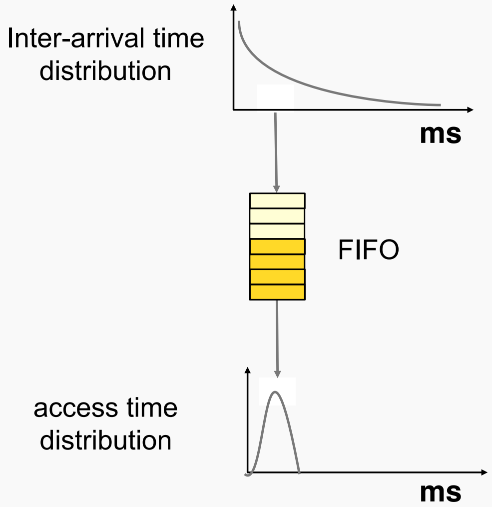


  - It introduces additional latency to the data path
  - The effect of the queue depends on its depth

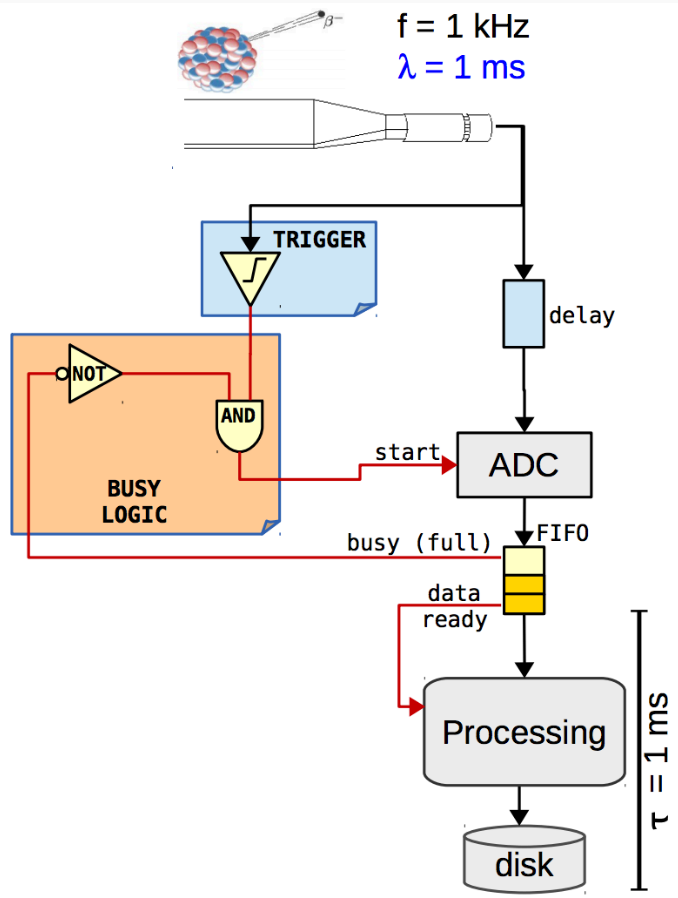


De-randomization

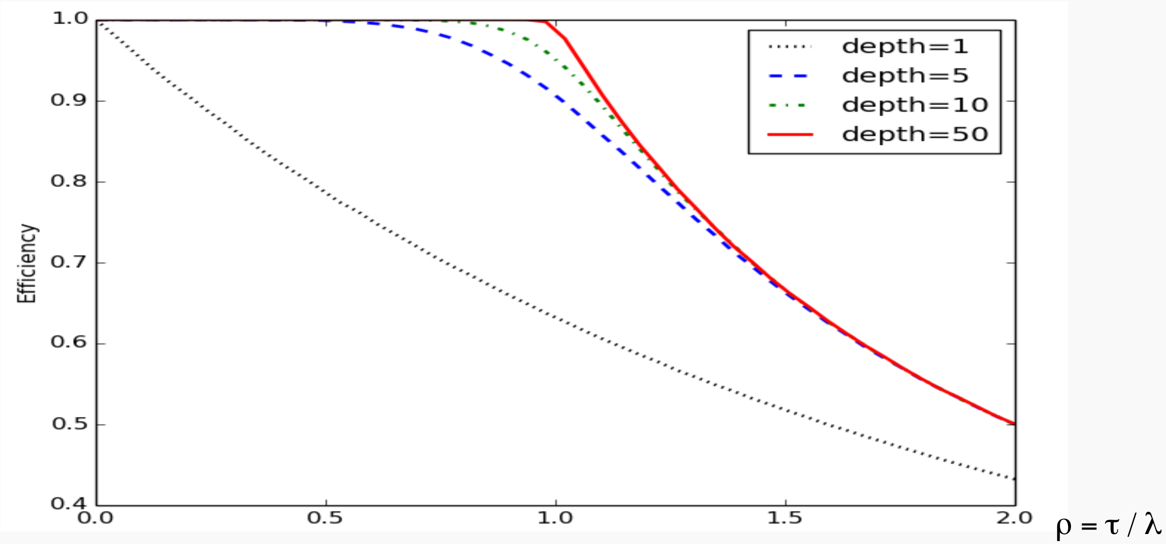
- Input fluctuations can be absorbed and smoothed by a queue
  - A First In First Out can provide a ~steady and **de-randomized** output rate

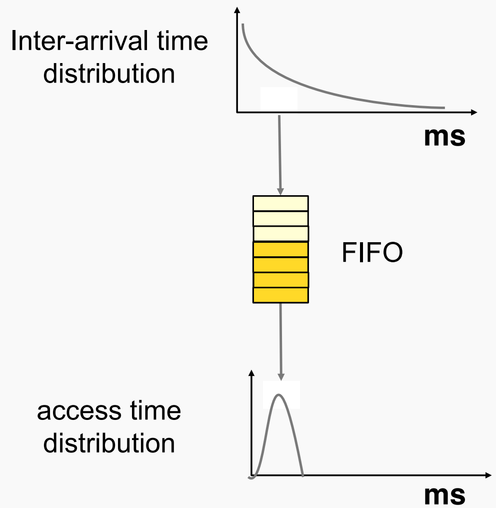


  - It introduces additional latency to the data path
  - The effect of the queue depends on its depth



Queuing theory

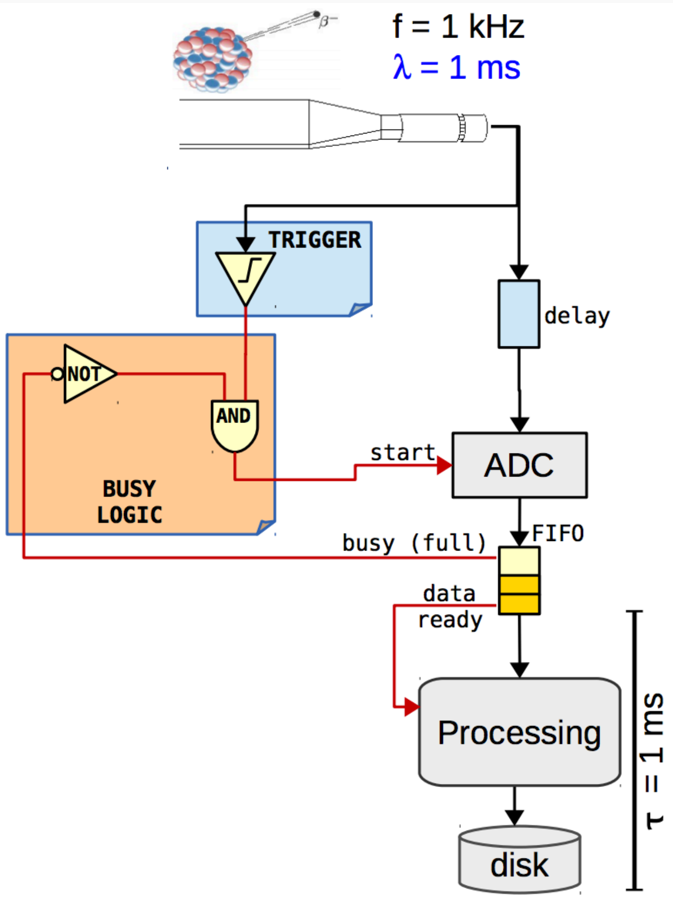




- Efficiency vs traffic intensity ($\rho = \tau / \lambda$) for different queue depths
  - ρ **> 1:** the system is overloaded ($\tau > \lambda$)
  - ρ **<< 1:** the output is over-designed ($\tau << \lambda$)
  - ρ **~ 1:** using a queue, high efficiency obtained even w/ moderate depth
- Analytic calculation possible for very simple systems only
  - Otherwise MonteCarlo simulation is required

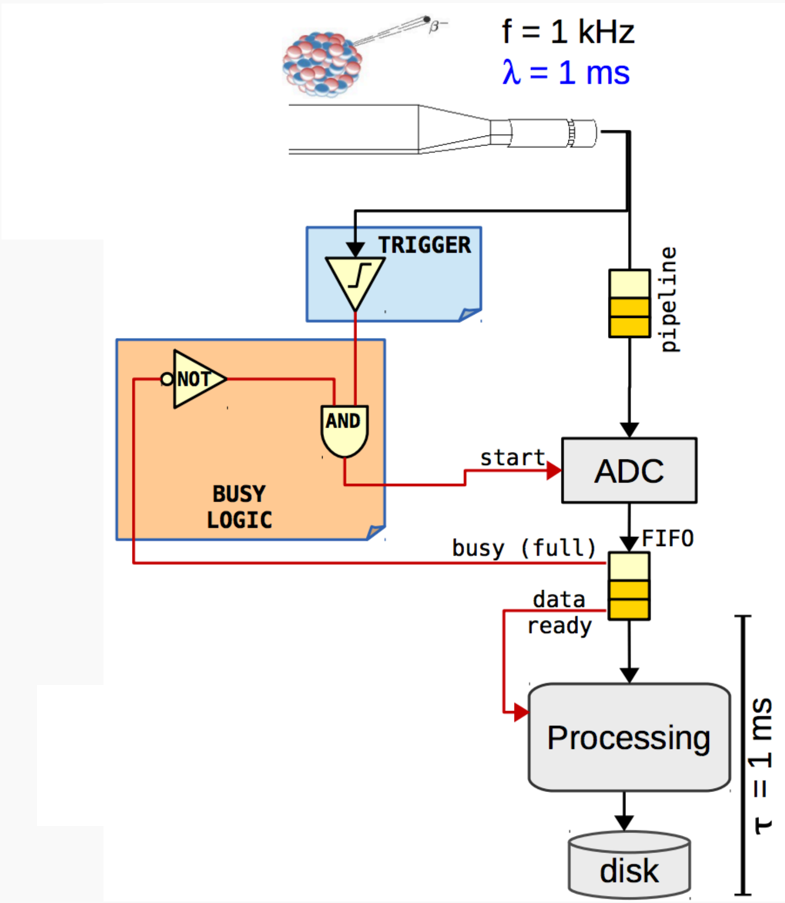
De-randomization summary

- Almost 100% efficiency with minimal deadtime achievable if
  - ADC is able to operate at rate >> f
  - Data processing and storing operate at a rate ~ f
- The FIFO decouples the low latency front-end from the data processing
  - Minimize the amount of “unnecessary” fast components
- Could the delay be replaced with a “FIFO”?
  - Analog pipelines, heavily used in LHC DAQs



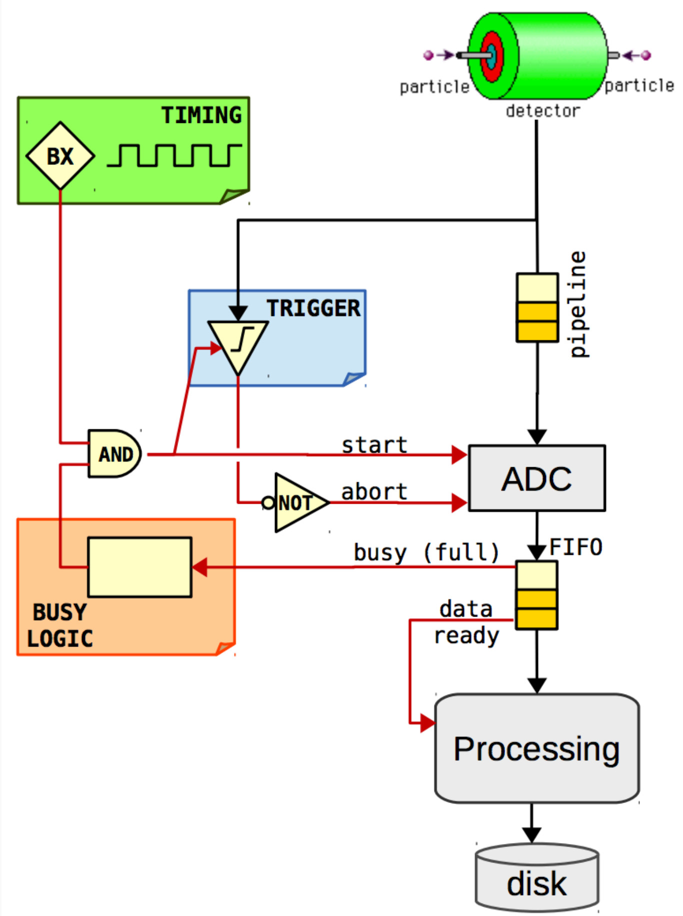
De-randomization summary

- Almost 100% efficiency with minimal deadtime achievable if
  - ADC is able to operate at rate >> f
  - Data processing and storing operate at a rate ~ f
- The FIFO decouples the low latency front-end from the data processing
  - Minimize the amount of “unnecessary” fast components
- Could the delay be replaced with a “FIFO”?
  - Analog pipelines, heavily used in LHC DAQs



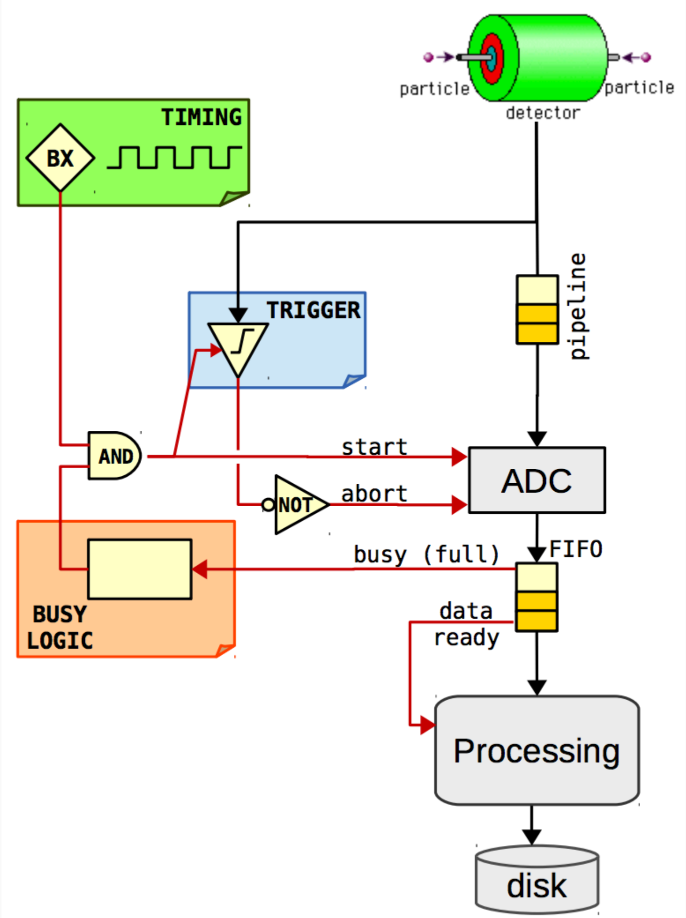
Collider setup

- Particle collisions are synchronous
  - So, do we still need de-randomization buffers?
- But the time distribution of triggers is random
  - Good events are unpredictable
- De-randomization still needed
- More complex busy logic to protect buffers and detectors
  - Eg: accept n events every m bunch crossings
  - Eg: prevent certain trigger patterns



Collider setup

- Particle collisions are synchronous
  - So, do we still need de-randomization buffers?
- But the time distribution of triggers is random
  - Good events are unpredictable
- De-randomization still needed
- More complex busy logic to protect buffers and detectors
  - Eg: accept n events every m bunch crossings
  - Eg: prevent certain trigger patterns

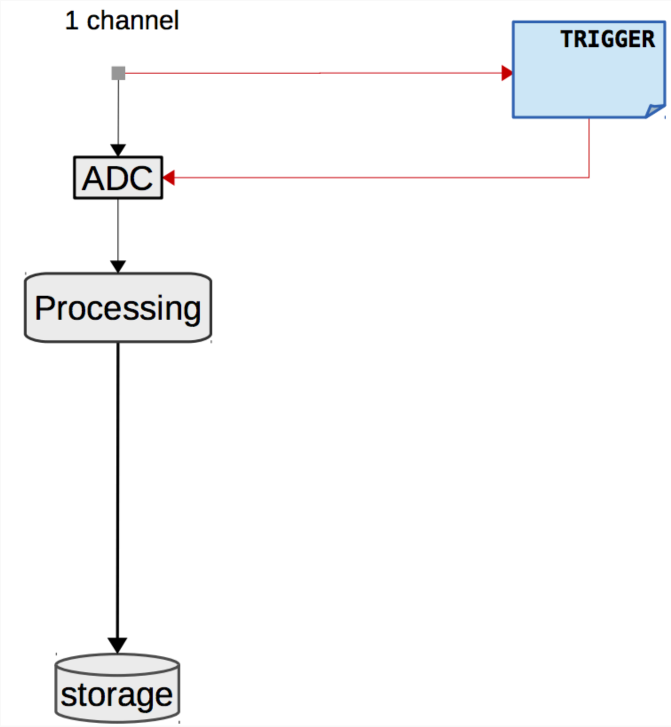


DAQ Outline

- Basic DAQ concepts
  - Digitization, Latency
  - Deadtime, Busy, Backpressure
  - De-randomization
- Scaling up
  - Readout and Event Building
  - Buses vs Network
- Do it yourself

Adding more channels

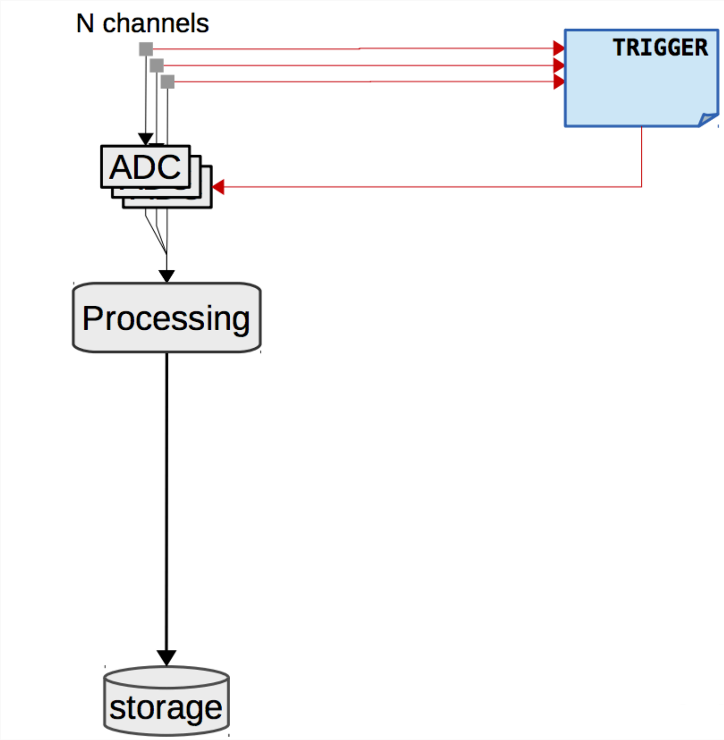
- Adding more channels requires a hierarchical structure committed to the data handling and conveyance



Front-End	data digitization data buffering
Readout	data extraction data formatting
Event Building	event assembly
Event Filtering	event rejection
Event Logging	file storage

Adding more channels

- Adding more channels requires a hierarchical structure committed to the data handling and conveyance



Front-End | data digitization, data buffering

Readout | data extraction, data formatting, data buffering

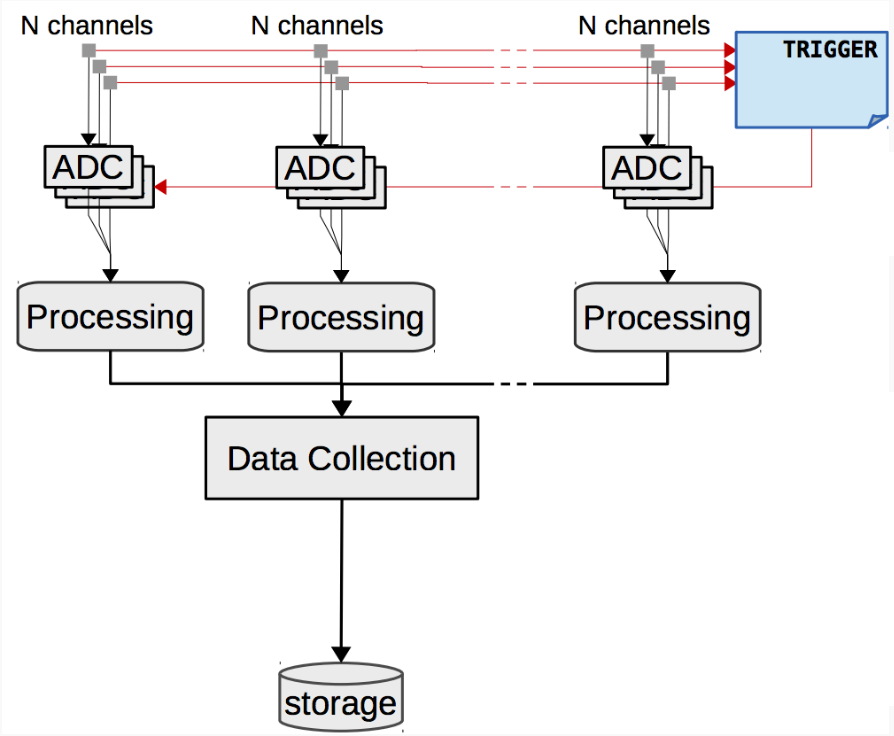
Event Building | event assembly

Event Filtering | event rejection

Event Logging | file storage

Adding more channels

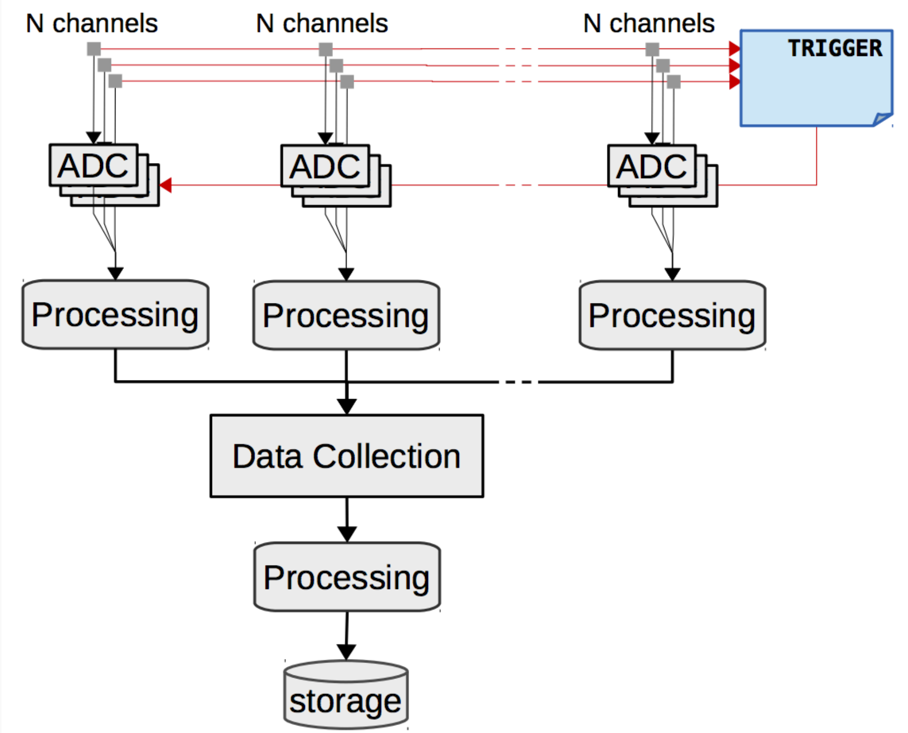
- Adding more channels requires a hierarchical structure committed to the data handling and conveyance



Front-End	data digitization data buffering
Readout	data extraction data formatting data buffering
Event Building	event assembly event buffering
Event Filtering	event rejection
Event Logging	file storage

Adding more channels

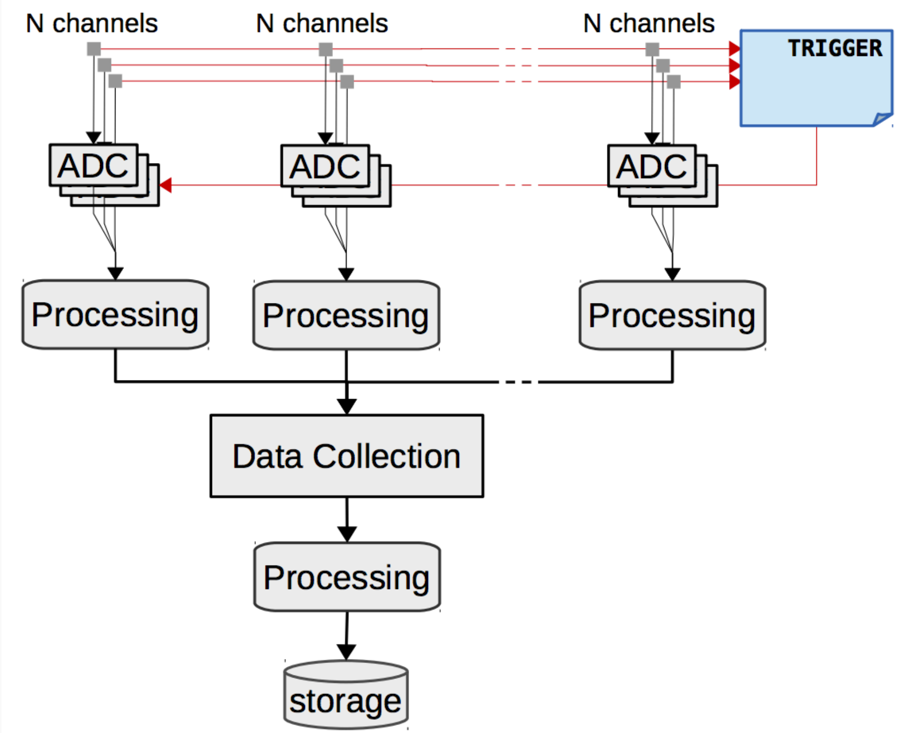
- Adding more channels requires a hierarchical structure committed to the data handling and conveyance



Front-End	data digitization data buffering
Readout	data extraction data formatting data buffering
Event Building	event assembly event buffering
Event Filtering	event rejection event buffering
Event Logging	file storage

Adding more channels

- Adding more channels requires a hierarchical structure committed to the data handling and conveyance



Front-End | data digitization
data buffering

Readout | data extraction
data formatting
data buffering

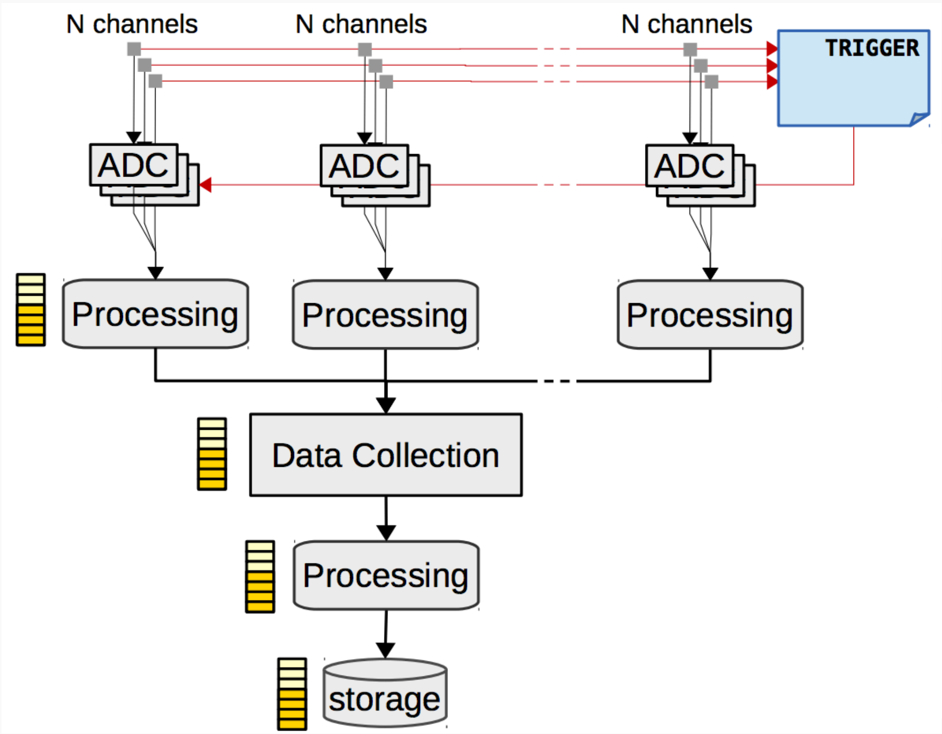
Event Building | event assembly
event buffering

Event Filtering | event rejection
event buffering

Event Logging | file storage
file buffering

Adding more channels

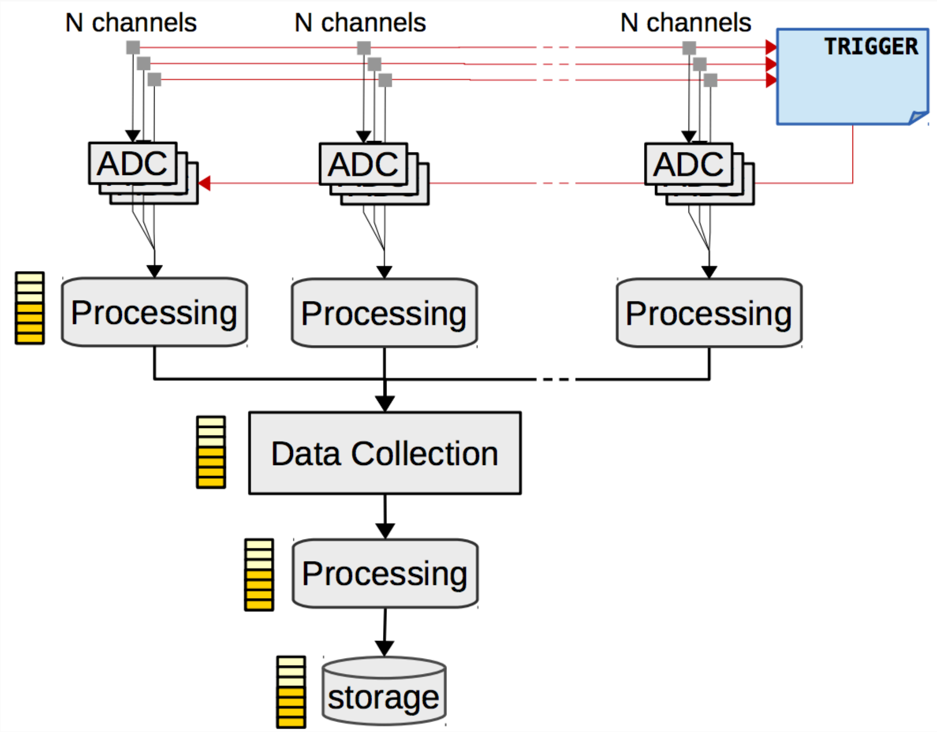
- Buffering usually needed at every level
  - DAQ can be seen as a multi level buffering system



Front-End	data digitization data buffering
Readout	data extraction data formatting data buffering
Event Building	event assembly event buffering
Event Filtering	event rejection event buffering
Event Logging	file storage file buffering

Backpressure

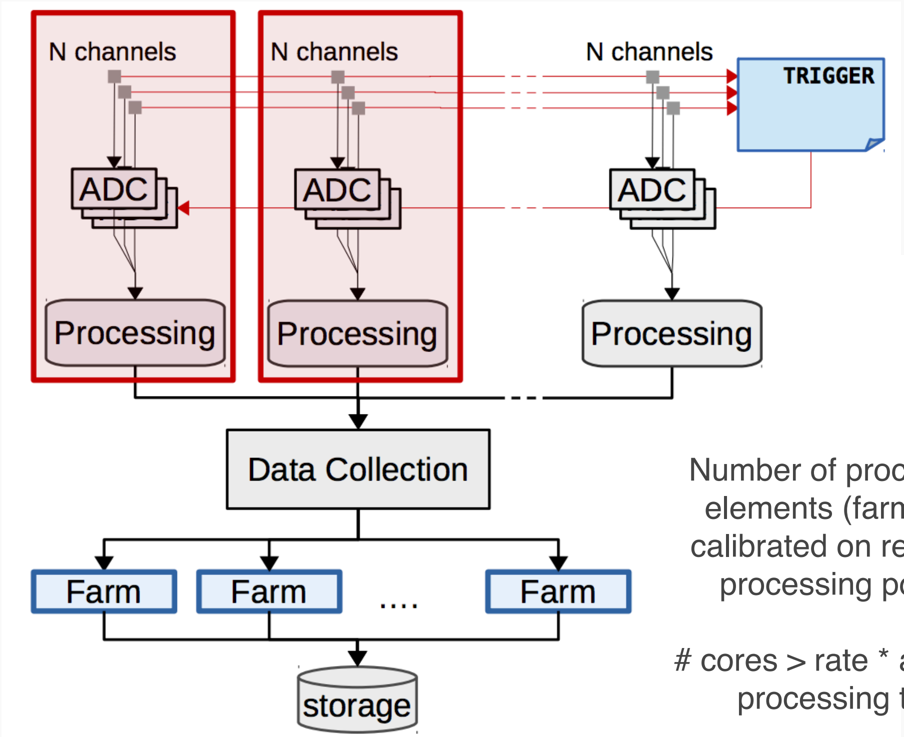
- If a system/buffer gets saturated
  - the “pressure” is propagated upstream (**back-pressure**)



- Up to exert busy to the trigger system
- Debugging: where is the source of backpressure?
  - follow the buffers occupancy via the monitoring system

Building blocks

- Reading out data or building events out of many channels requires many components



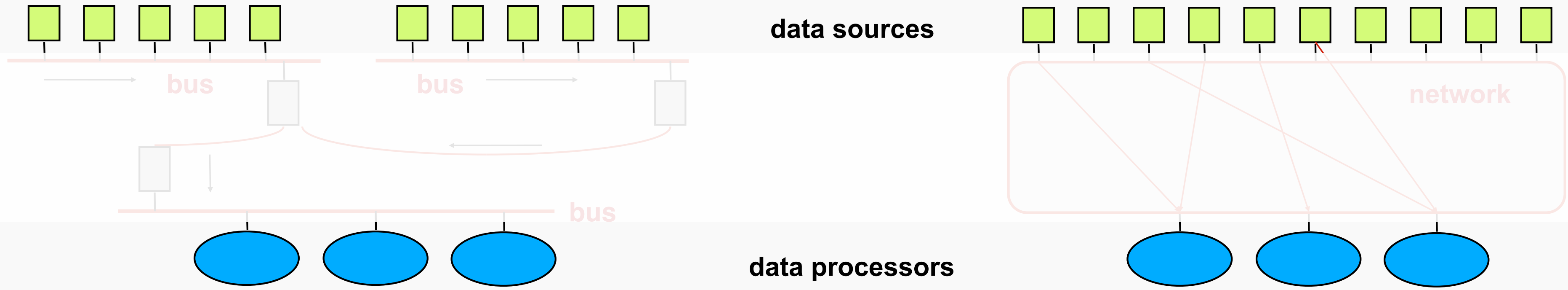
Number of processing elements (farm size) calibrated on required processing power:

cores > rate * average processing time

- In the design of our hierarchical data-collection system, we have to better define "**building blocks**"
  - Readout crates
  - HLT racks
  - event building groups
  - daq slices
- "Building blocks" allow to partition the DAQ and take data with part of the experiment

Readout Topology

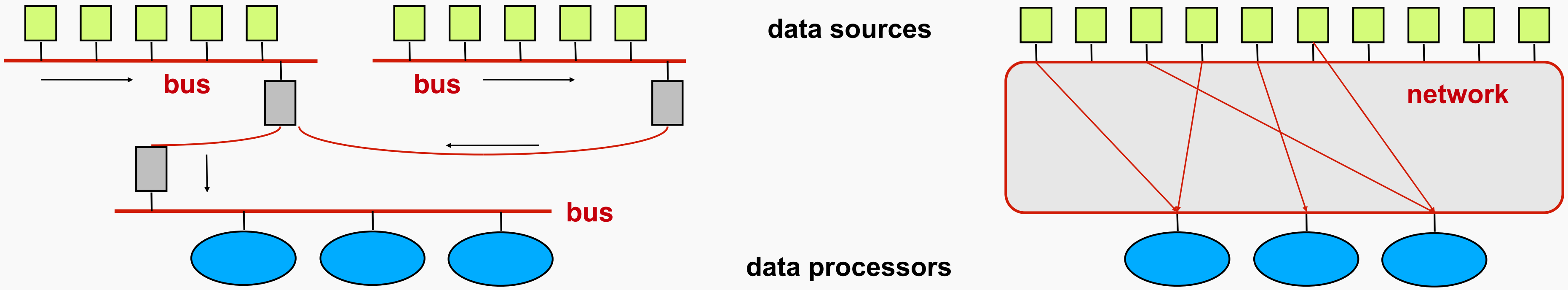
- How to organize the interconnections inside the building blocks and between building blocks?
  - How to connect data sources and data destinations?
  - Two main classes: **bus** or **network**



  - Warning: bus and network are generic concepts that can be easily confused with their most common implementations

Readout Topology

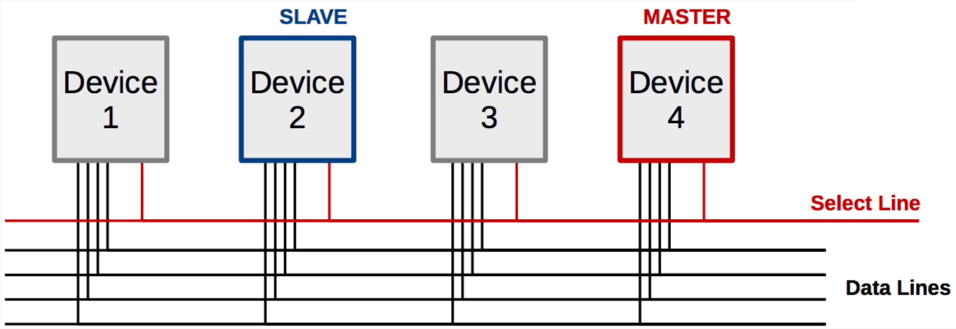
- How to organize the interconnections inside the building blocks and between building blocks?
  - How to connect data sources and data destinations?
  - Two main classes: **bus** or **network**

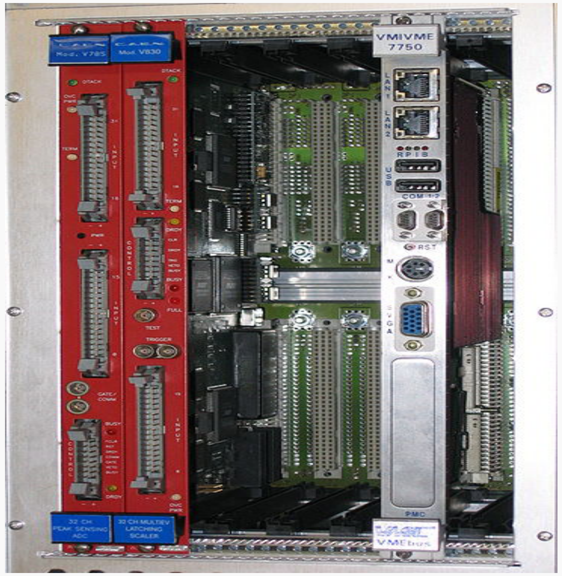


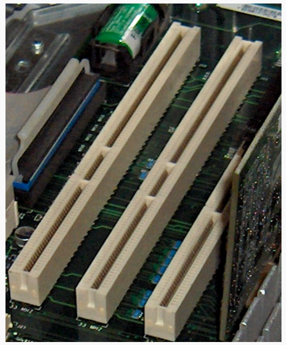
  - Warning: bus and network are generic concepts that can be easily confused with their most common implementations

Buses

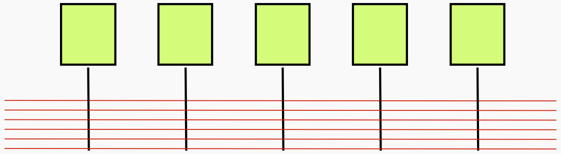
- Devices connected via a **shared bus**
  - Bus → group of electrical lines
- Sharing implies **arbitration**
  - Devices can be **master** or **slave**
  - Devices can be addresses (uniquely identified) on the bus
- E.g.: SCSI, Parallel ATA, VME, PCI …
  - local, external, crate, long distance, …





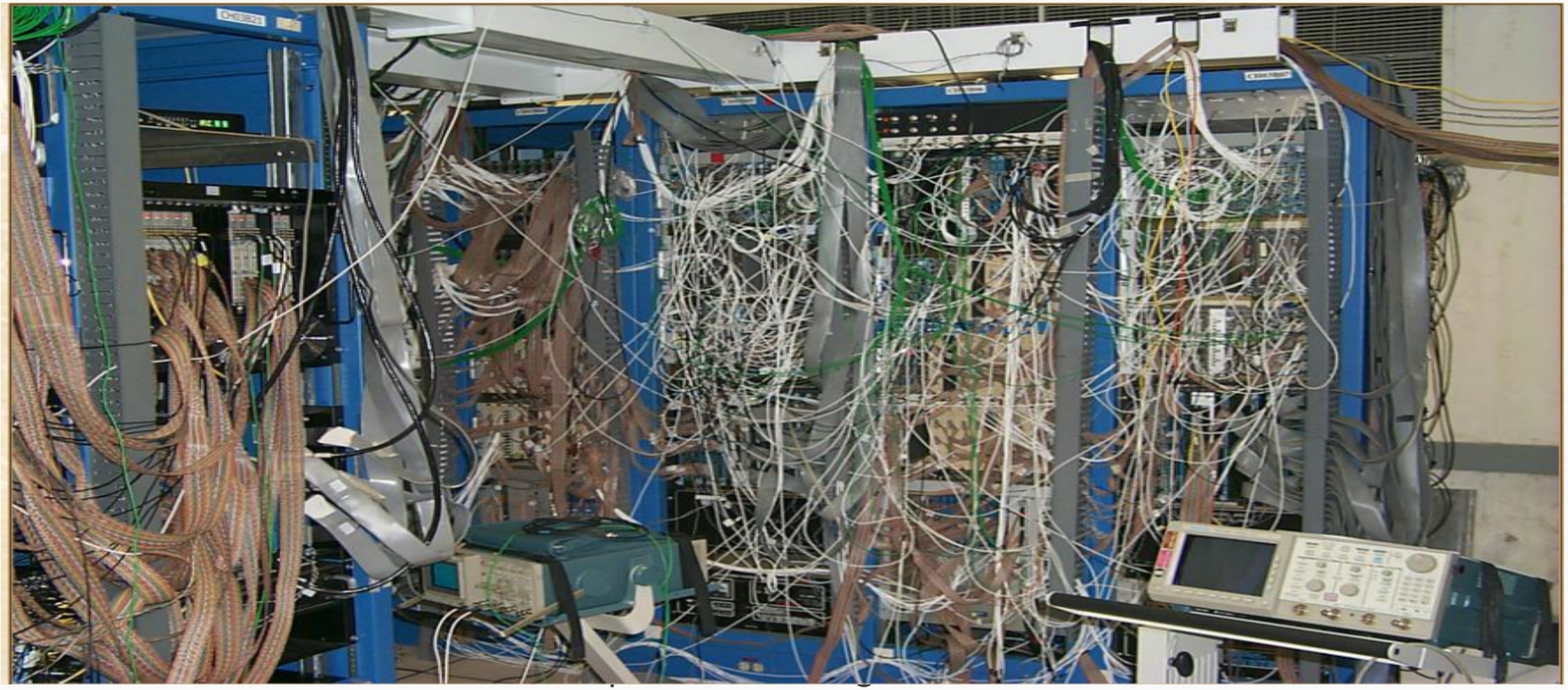


Bus facts



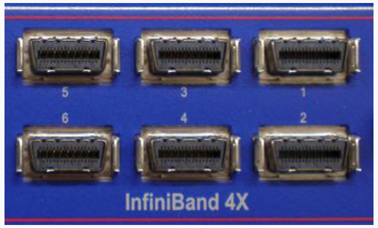
- Simple :-)
  - Fixed number of lines (bus-width)
  - Devices have to follow well defined interfaces
    - Mechanical, electrical, communication, ...
- **Scalability** issues :-(
  - Bus bandwidth is shared among all the devices
  - Maximum bus width is limited
  - Maximum number of devices depends on bus length
  - Maximum bus frequency is inversely proportional to the bus length
  - On the long term, other “effects” might limit the scalability of your system

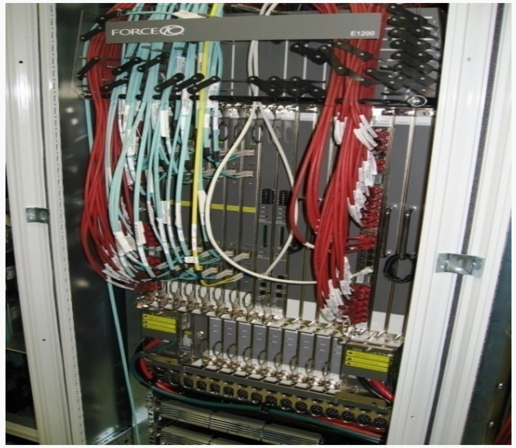
Bus facts

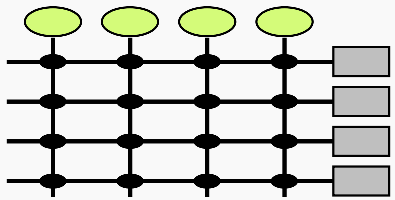


- On the long term, other “effects” might limit the scalability of your system

Network



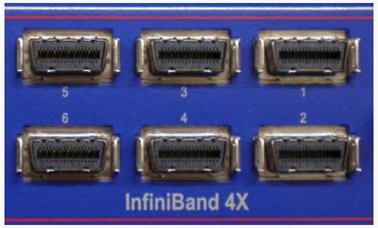


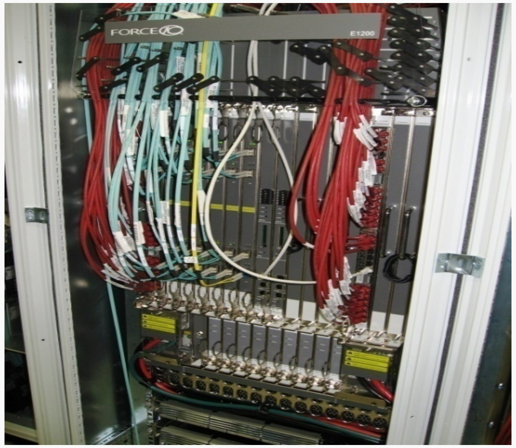


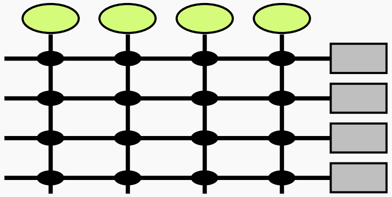
- All devices are **equal**
  - Devices <u>communicate directly</u> with each other via messages
  - No arbitration, simultaneous communications
- Examples:
  - Telephone, Ethernet, Infiniband, …
- In switched networks, **switches** move messages between sources and destinations
  - Find the right path
  - Handle **congestions** (two messages with the same destination at the same time)
    - The key is ….

Network

- All devices are **equal**
  - Devices <u>communicate directly</u> with each other via messages
  - No arbitration, simultaneous communications
- Examples:
  - Telephone, Ethernet, Infiniband, …
- In switched networks, **switches** move messages between sources and destinations
  - Find the right path
  - Handle **congestions** (two messages with the same destination at the same time)
    - The key is …. **buffering**

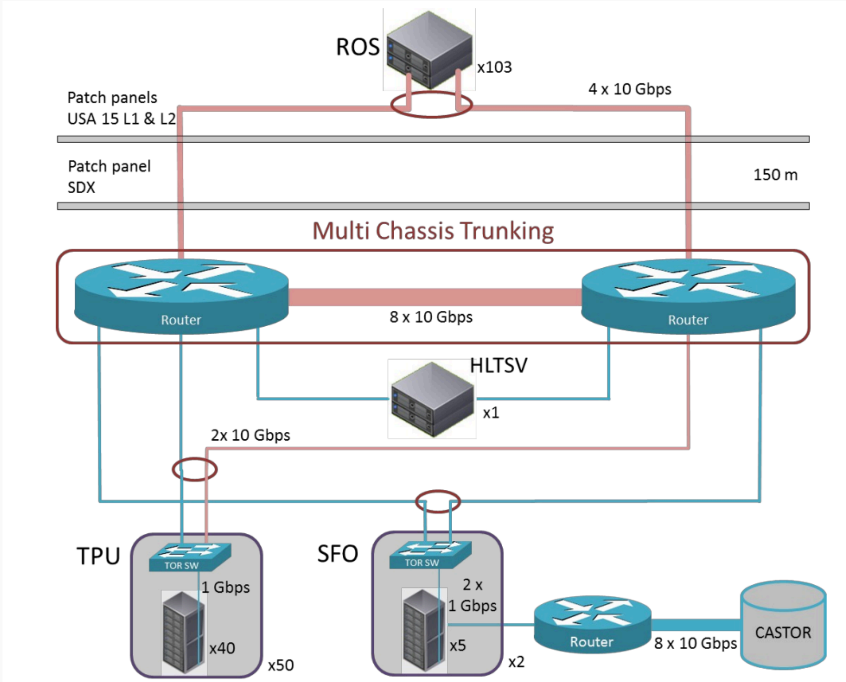






Network

- Networks scale well (and allow redundancy)
  - They are the backbones of LHC DAQ systems



DAQ Outline

- Basic DAQ concepts
  - Digitization, Latency
  - Deadtime, De-randomization
- Scaling up
  - Readout and Event Building
  - Buses vs Network
- Do it yourself

DAQ Mentoring

- Study the trigger properties
  - Periodic or stochastic, continuous or bunched
- Consider the needed efficiency
  - It is good to keep operation margins, but avoid over-sizing
- Identify the fluctuation sources and size adequate buffering mechanisms
  - Watch out: (deterministic) complex systems introduce fluctuations: multi-threaded software, network communications, ...
- An adequate buffer is not a huge buffer
  - Makes your system less stable and responsive, prone to divergences and oscillations. Overall it decreases reliability

DAQ Mentoring

- Keep it simple, keep under control the number of free parameters without losing flexibility
  - Have you ever heard about SUSY phase-space scans? Do you really want something like that for your DAQ system?
- Problems require perseverance
  - Be careful, a rare little glitch in your DAQ might be the symptom of a major issue with your data
- In any case, ...

DAQ Mentoring

- Keep it simple, keep under control the number of free parameters without losing flexibility
  - Have you ever heard about SUSY phase-space scans? Do you really want something like that for your DAQ system?
- Problems require perseverance
  - Be careful, a rare little glitch in your DAQ might be the symptom of a major issue with your data
- In any case, ...

Today's trigger menu

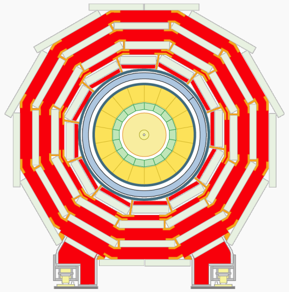
Introduction to trigger concepts

- What is all this triggering fuss about?
- Requirements and constraints
- Efficiencies and how to measure them
- Examples, examples, example…

The problem is…

[Slide taken from from *F. Winklmeier*, CERN, 2016]

…that modern large-scale experiments are **really BIG**

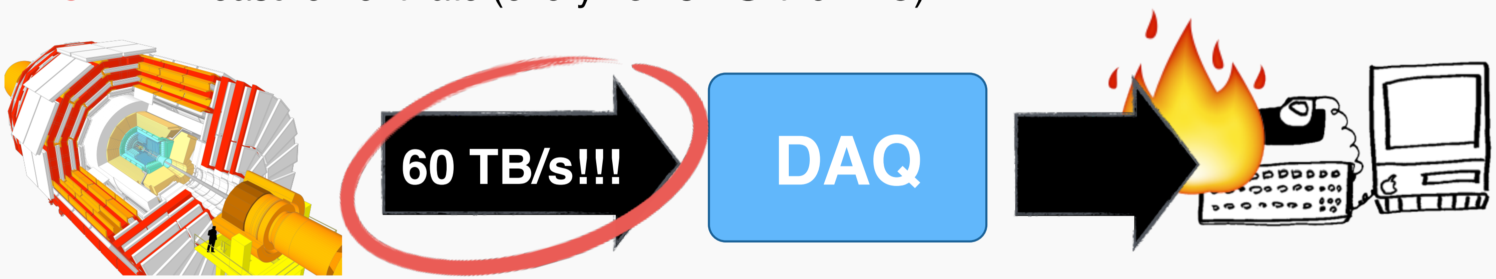


i.e. LHC experiments (ATLAS/CMS)

- **~100M channels**
- **~1-2 MB** of RAW data per collected event

… and **really FAST**

- **~40 MHz** measurement rate (every 25 ns - @ the LHC)

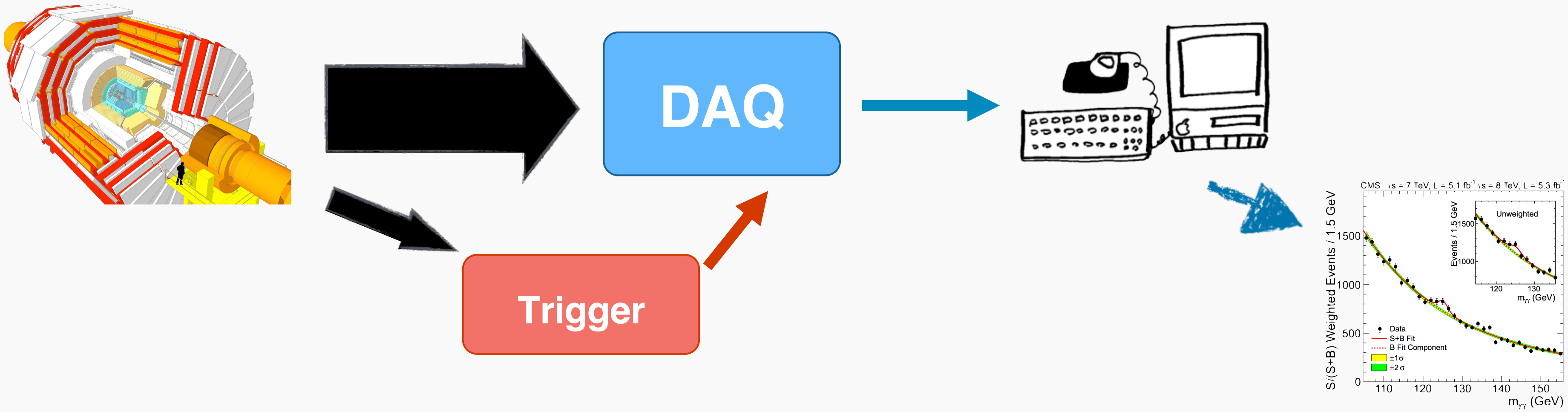


Data volume is a *key problem* in modern large-scale experiments

Definitions - Trigger

The **Data AcQuisition (DAQ)** system collects the data from the different parts of the detector, converts the data in a suitable format and saves it to permanent storage

The **Trigger** is the system that decides, in real time, whether to read out or discard the measurements corresponding to each observed interaction for offline analysis.



…a bit like…

the elusive subject
New physics
(somewhere in there)

bystanders, obtrusions,
lookalikes
“Old” physics

the shutter release
Trigger signal

the photographer
Trigger

the camera
Detector & DAQ

… taking a photo…

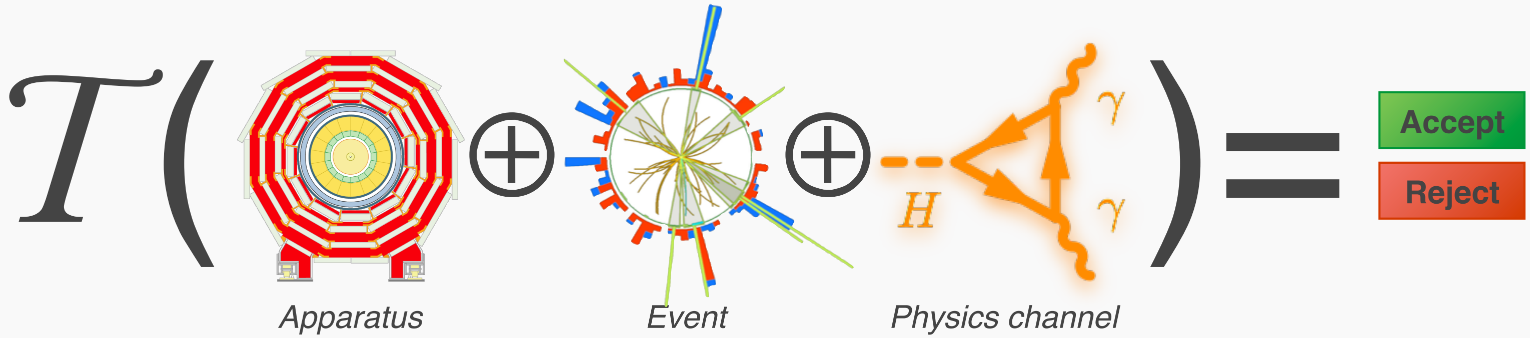
- Tune **aperture**, **timing**, **focus** → Ensure good accuracy
- Check **light levels** → Ensure good sensitivity
- Hold your hand **steady** → Ensure good synchronisation
- Continuously **monitor** the camera screen → Analize the stream of data
- **Shoot** when the subject in position → Start capturing the ‘event’

What is a trigger system, in practice?

[Slide taken from from *A. Hocker*, CERN, 2009]

Hardware/software processor filtering the event stream based upon a ‘quick look’ at the data

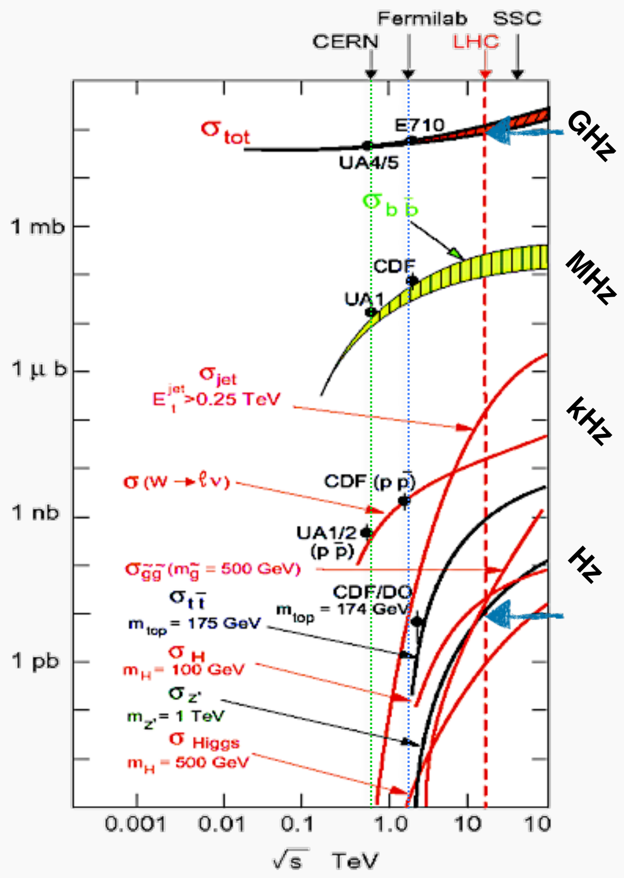
- Look at (almost) **all events**, select **most interesting** ones, collect **all detector information** and **store it** for offline analysis (for a reasonable amount of money)
- It must accept interactions at a **rate low enough** for storage and reconstruction



Key questions:

- **What is “interesting” and what not?**
- **How selective must it be?**
- **How fast must it to be?**

What to trigger on (at the LHC)?



Expected production rate for process X:

$$R_X^{prod} = \sigma_X \times \mathscr{L}$$

Process	Cross section (nb) @ 14 TeV	Production rates (Hz) @ $\mathscr{L}$= 10^{34} $cm^{-2}s^{-1}$
inelastic	10^8	10^9
$b\bar{b}$	5×10^5	5×10^6
$W \to \ell\nu$	15	150
$Z \to \ell\ell$	2	20
$t\bar{t}$	1	10
H	0.05	0.5

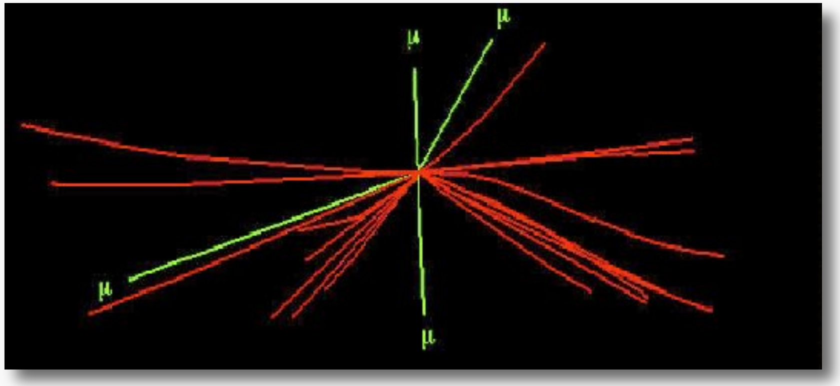
Many **orders of magnitude** between QCD background and primary physics channels rates

$$\frac{\sigma_{tot}}{\sigma_H} \approx 10^{11}$$

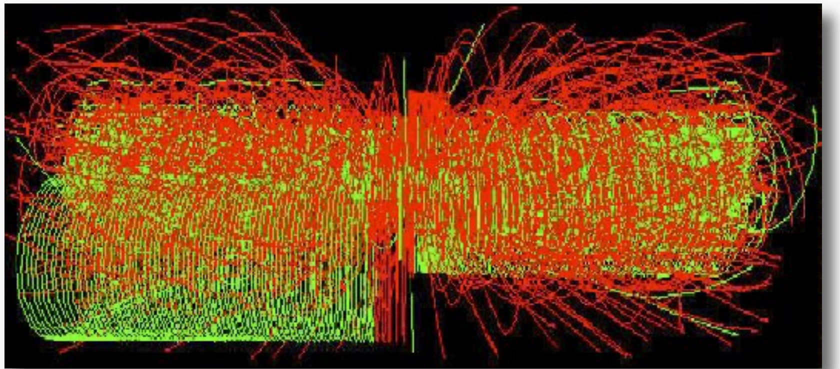
Even if it would be feasible, saving all events at hadron collider is not useful

How to identify the interesting events?

Only high-p_T tracks Higgs → 4μ



High-pT tracks +30 MinBias

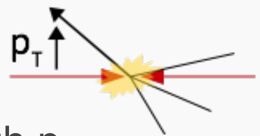


Example: Higgs → 4μ events

- "Pretty interesting" process
- Key to the 2012 Higgs discovery

In practice:

- Hidden under tons of "well known" physics
  - Lots of low momentum particles
- Exploit the physics signature to identify the underlying Higgs decay
  - 4 high-momentum *μ*



"Interesting" physics usually is high-p_T

What trigger rate can be afforded?

The **Data AcQuisition (DAQ)** system collects the data from the different parts of the detector, converts the data in a suitable format and saves it to permanent storage.

DAQ Bandwidth mainly constraint by available technologies, size and costs:

- **Finite storage capabilities**
- **Finite computing power** (available for online and offline processing)

Maximum allowed trigger rate determined by typical event size:

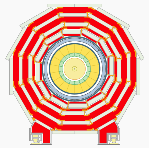
$$B_{DAQ} = R_T^{max} \times S_E$$

DAQ Bandwidth — **Maximum Trigger rate** — **Event size**

How many particles per event?

How many FE channels?

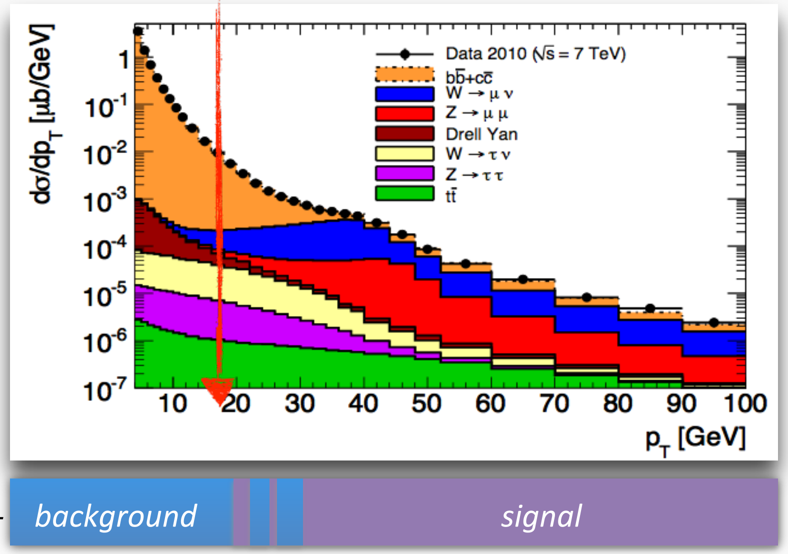
FE channels in ATLAS/CMS



- $O(10^7)$ in inner detector
- $O(10^5)$ in calorimeters
- $O(10^6)$ in muon detectors

Requirement 1: High Background Rejection

Inclusive single muon p_T spectrum



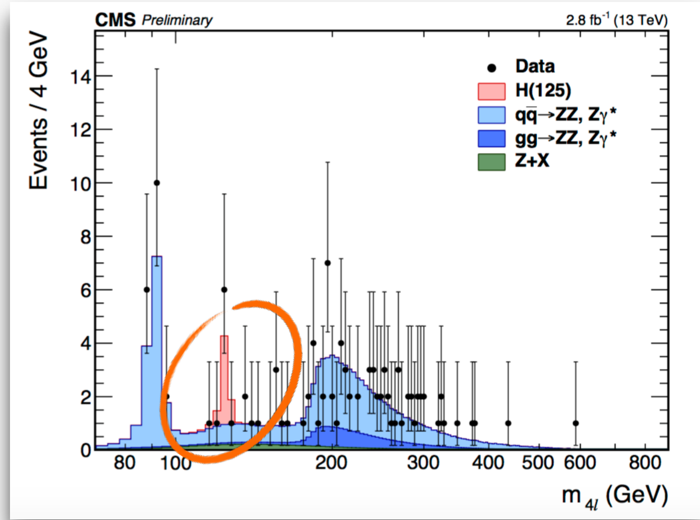
$\mu\ p_T$

$$R_{bkg} = 1 - \frac{N_{bad}^{accepted}}{N_{bad}^{prod/exp}}$$

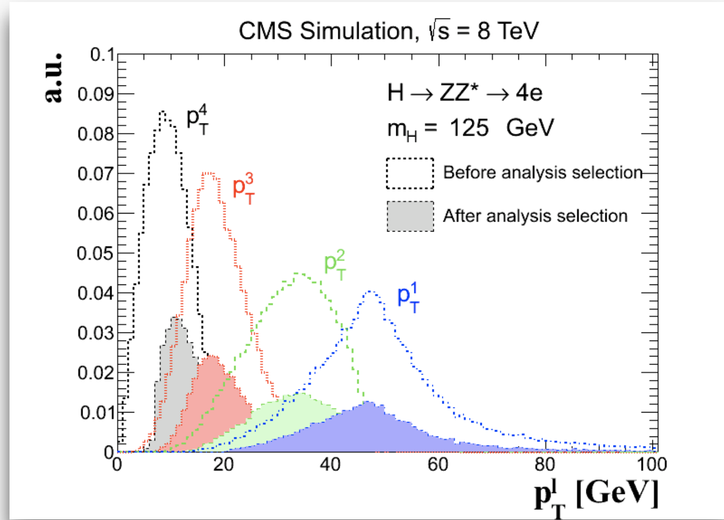
Background rejection (Rate control)

- Instrumental or physics background
  - Need to identify characteristics which can suppress the background
  - Need to demonstrate solid understanding of background rate and shapes
- Backgrounds sometimes known with large uncertainties
  - Make your trigger flexible and robust

Requirement 2: High Signal Efficiency



4-leptons invariant mass,
selected events for H→ZZ*→4l

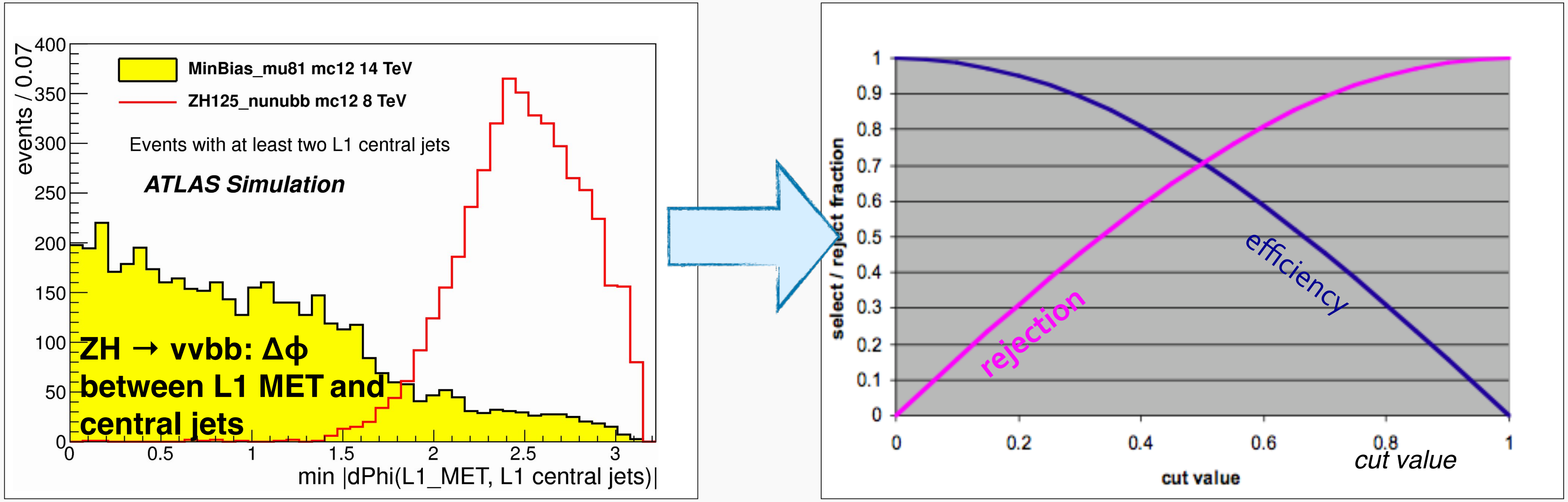


$$\epsilon_{trg} = \frac{N_{good}^{accepted}}{N_{good}^{prod/exp}}$$

Maximise trigger acceptance

- **Ideal**: complete acceptance for all events of interest
- **In practice**, aim for: trigger thresholds lower than any conceivable analysis cut

In real life: find the best compromise possible



Basic requirements (I)

Need **high efficiency** for selecting processes for physics analysis

- Selection should not have biases that affect physics results
- Event losses must be low (and known)
- …

Need **large reduction** of rate from unwanted high-rate processes

(according to the capabilities of DAQ and also of offline computing!)

- Instrumental background
- High-rate physics processes that are not relevant for analysis

System must be **affordable**

- e.g algorithms executed at high rate must be fast

Basic requirements (II)

Robustness is essential

- Trigger is a 'mission critical' system; no data can be taken without it
  - It must function, and function predictably, under all experimental conditions
- Simple and inclusive triggers are preferable, whenever possible

Highly flexible, to react to changing conditions (e.g. wide luminosity range)

- Programmable thresholds, high granularity to maintain uniform performance, ability to follow luminosity, beam-size and vertex position changes
- Long term effectiveness, to reach physics results even after >10 years of data taking

And now, into the details…

What is the trigger looking for?

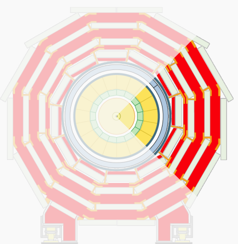
Trigger Signatures: Features distinguishing new physics from the bulk of the SM cross-section

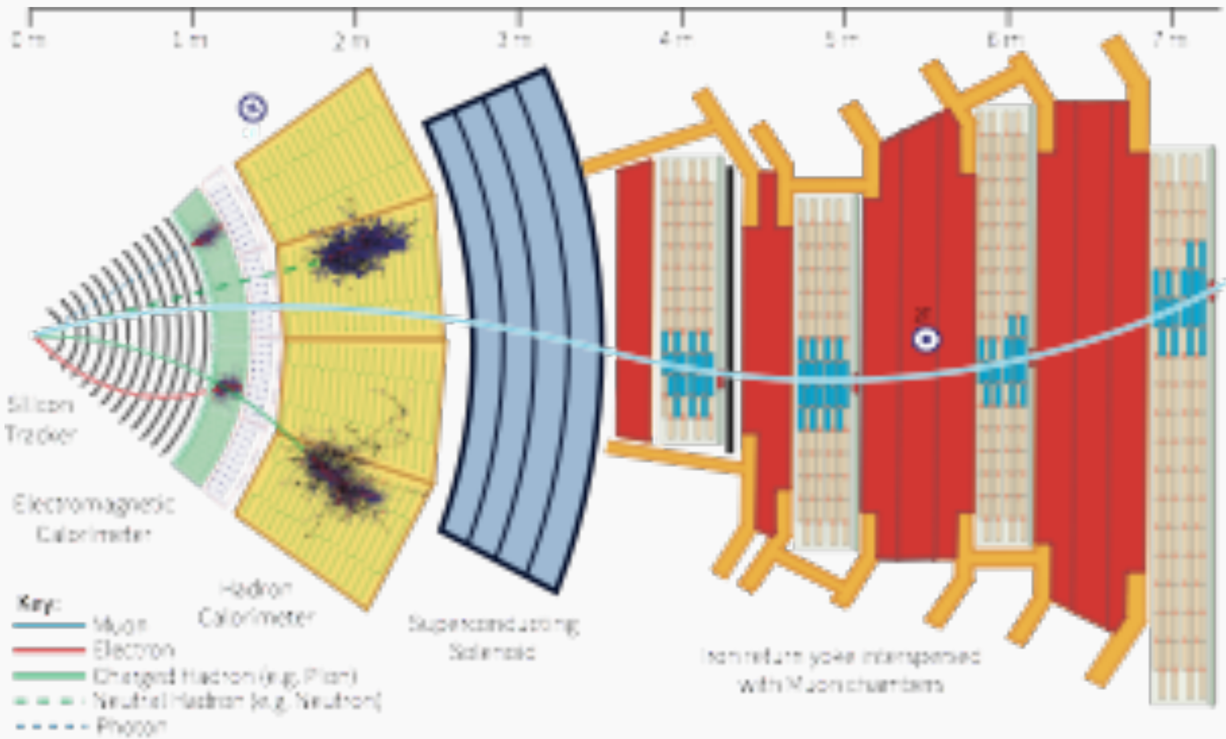
High p_T e,ɣ

High p_T jets

High p_T μ

Global quantities
Total and Missing Transverse Energy





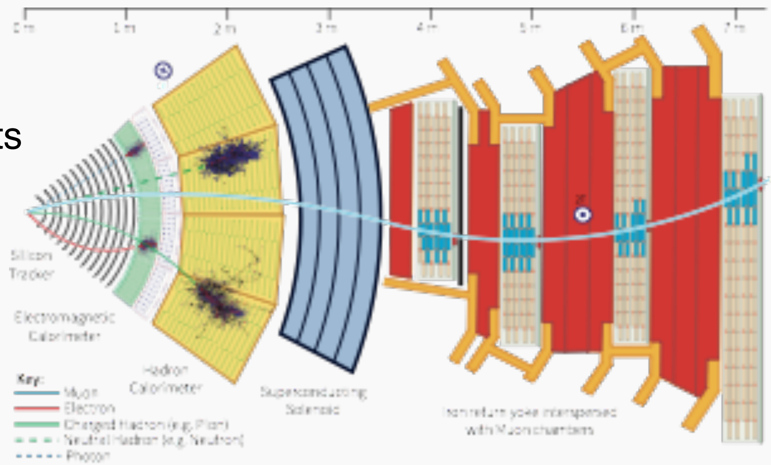
Trigger algorithms

Trigger decision result of several **trigger algorithms**

- Operate on trigger information from subdetector(s) to identify signatures
- Generally, ***several algorithms operate in parallel to find different signatures****
  - e.g. calorimeter information used to find electrons + jets in parallel
- Algorithms must cover whole detector in an unbiased way
- Output is a count or list of trigger signatures, possibly with additional information
  - Object pt, position, charge, 'quality', etc

Some algorithms are 'global' over the whole detector

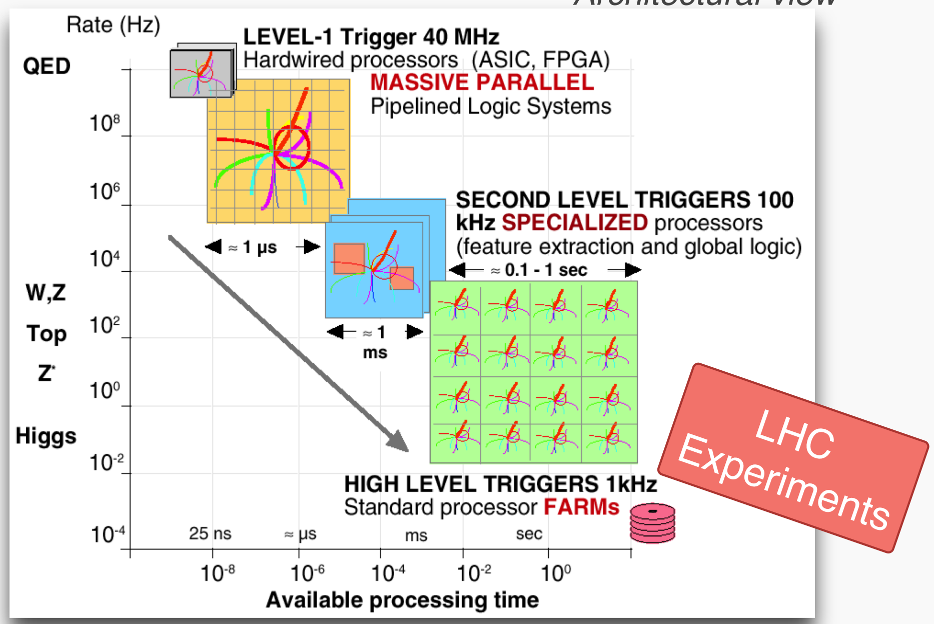
- Examples: Missing Et, Total Et, Ht, global object counts



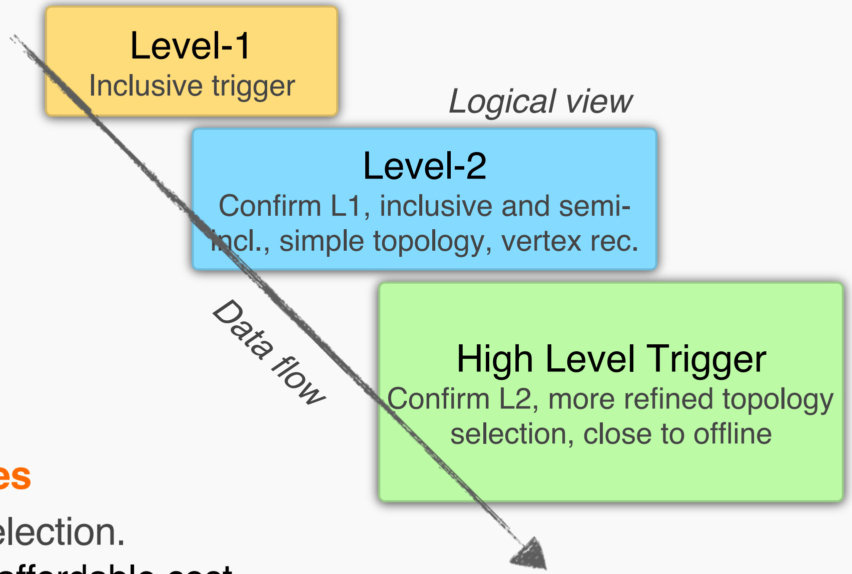
* or, to be precise, objects

Multi-level triggers

Architectural view



Logical view



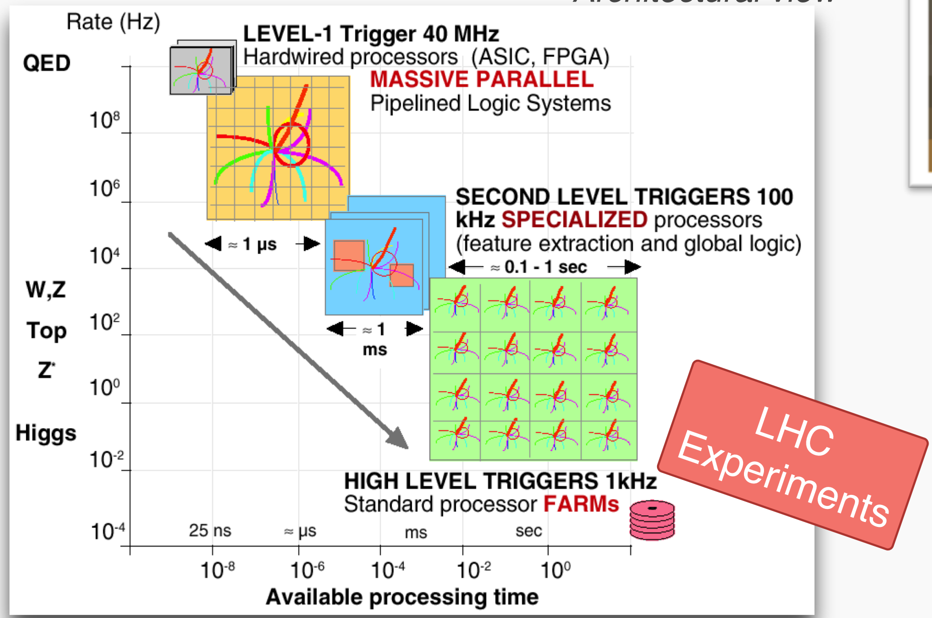
Trigger decision divided in **multiple sequential stages**

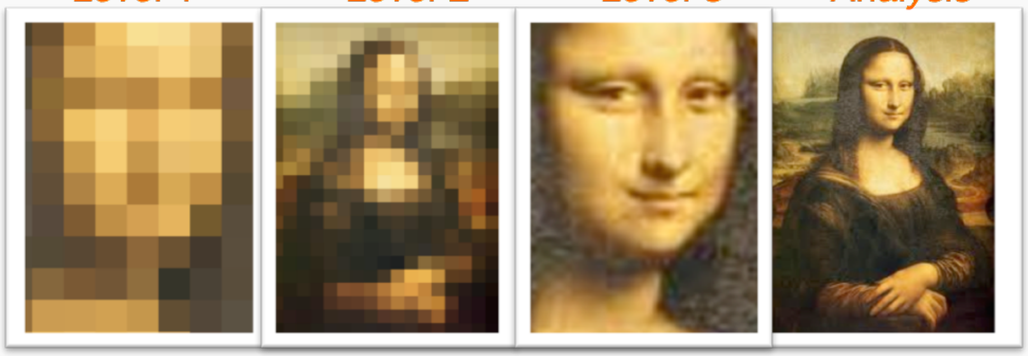
Progressive reduction in rate after each stage of selection.

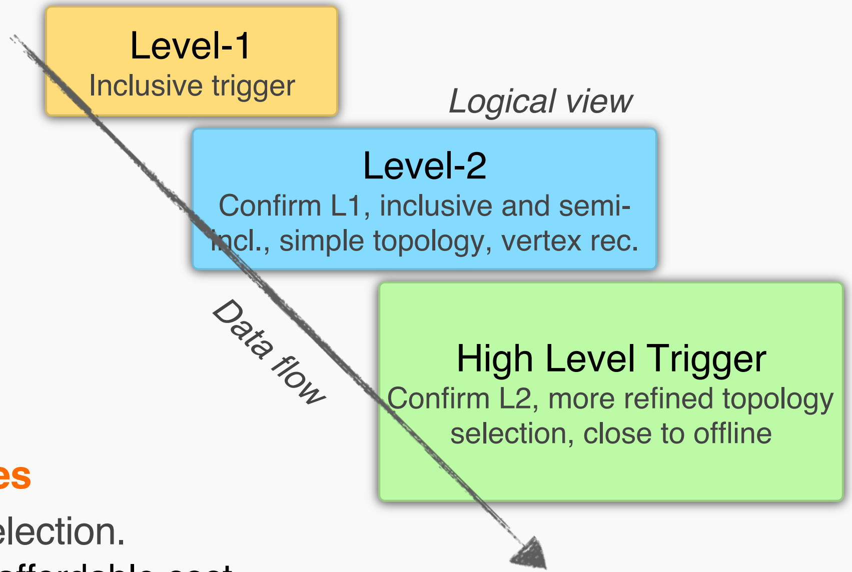
- Allows use of more and more complex algorithms at affordable cost

Multi-level triggers

Architectural view







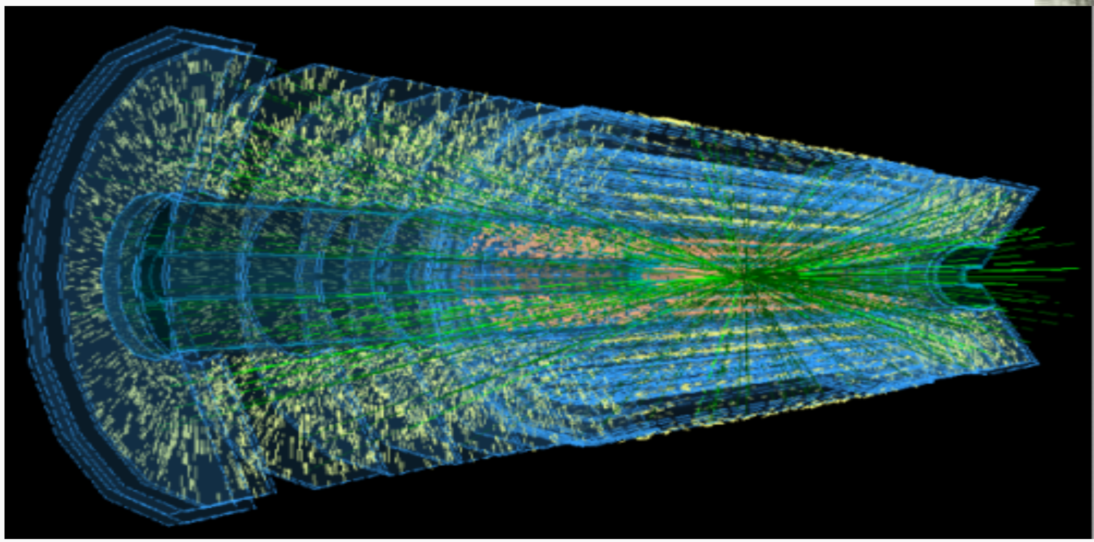
Trigger decision divided in **multiple sequential stages**

Progressive reduction in rate after each stage of selection.

- Allows use of more and more complex algorithms at affordable cost

CPU intensive triggers: the (real-time) tracking problem

Particle Tracking numbers:
~ 2000 charge tracks / “event”
~ 100k strip and pixel hits / “event”
Up to ~80 pp collisions / “event”.
Non-local combinatorial problem
of associating hits into a track.



Full tracking with CPUs requires
order of 1 second on a x86_64 core.
For 100kHz, 10^5 cores would be
required along with adequate networking.

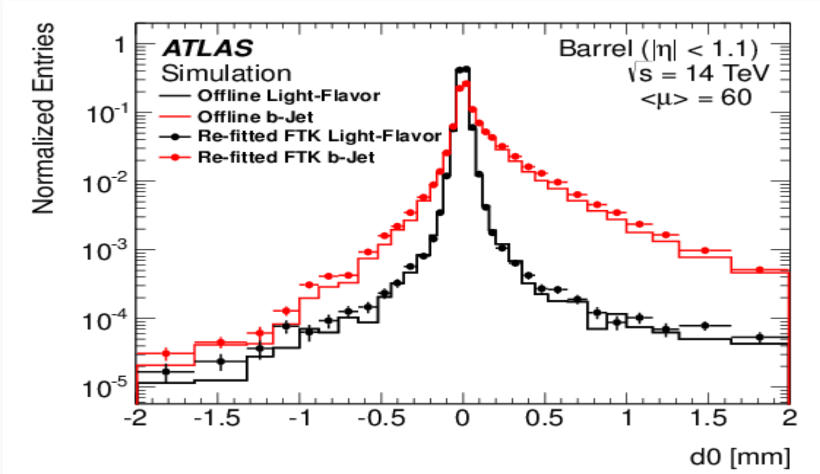
ATLAS Fast TracKer (FTK)

Dedicated, hardware-based tracking.

Provides tracking to the HLT for the full event at 100kHz.

Finds and fits tracks O(100 μs) in the ID silicon.

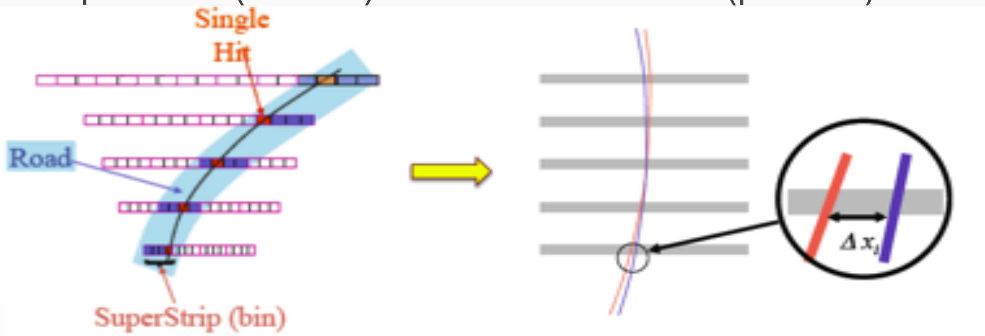
Processing performed in two steps



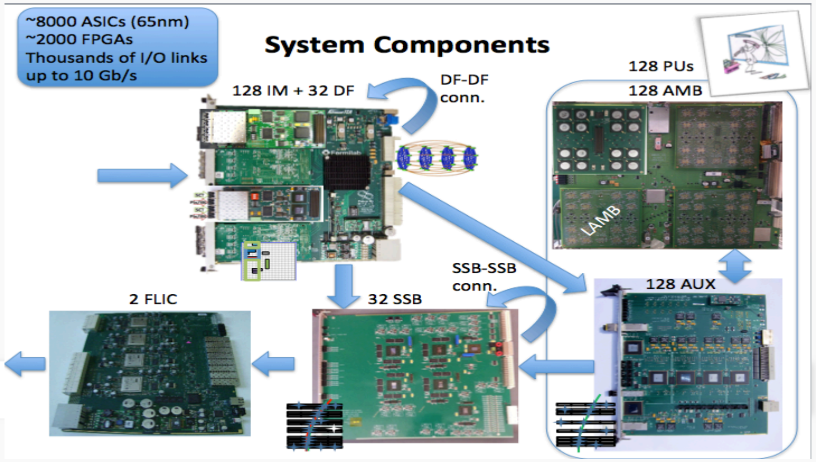
Light jet rejection using FTK+HLT refit compared to offline reconstruction

hit pattern matching to pre-stored patterns (coarse)

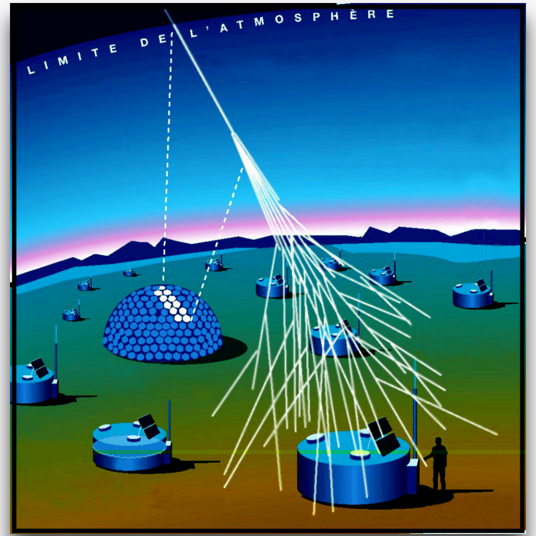
subsequent linear fitting in FPGAs (precise)



Associative memory ASIC



Simple signatures: Auger observatory (fluorescence)

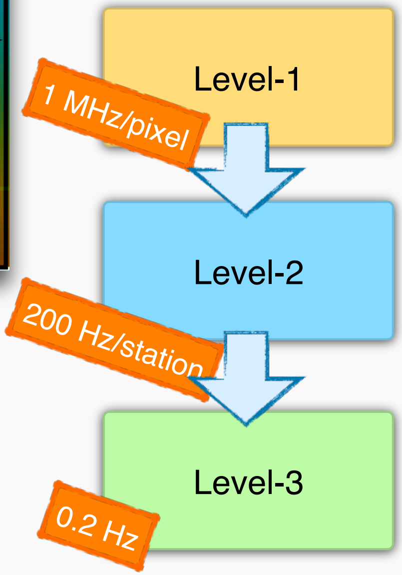


Surface Detector: array of ~1600 water Cherenkov stations over 3000 km^2 on ground, to identify secondary particles
Florescent Detector: 4 UV telescopes measure the shower Energy longitudinally

Detect air showers generated by cosmic rays above 10^{17} eV

- Expected rate < 1/km2/century.
- 2 large area fluorescence detectors

3-level trigger installed on each detector

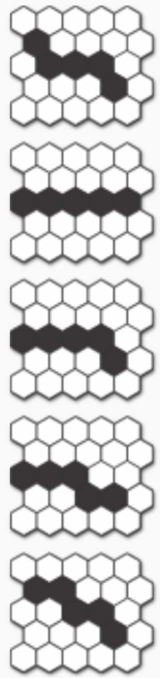


L1: (local) select active pixels
- ADC counts > threshold
- ADC digitises every 100 ns (time resolution)
- ADC values stored for 100 μs in local buffers
- Synchronised via GPS clock signal

L2: (local) identifies track segments
- Geometrical criteria with programmable pattern recognition algorithms

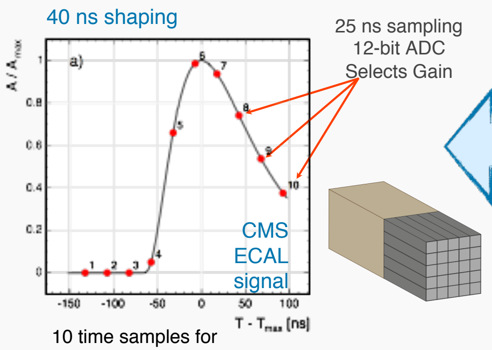
L3: (central) 3-D correlation between L2 triggers

Example of L2 patterns

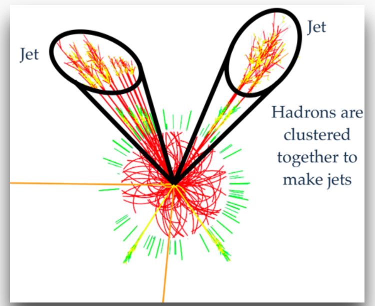


Multiple signatures: the CMS calorimeter trigger

High energy e, γ, τ, jets, missing E_T, ΣE_T identification



10 time samples for amplitude and time reconstruction



1: Dedicated Front-End electronics

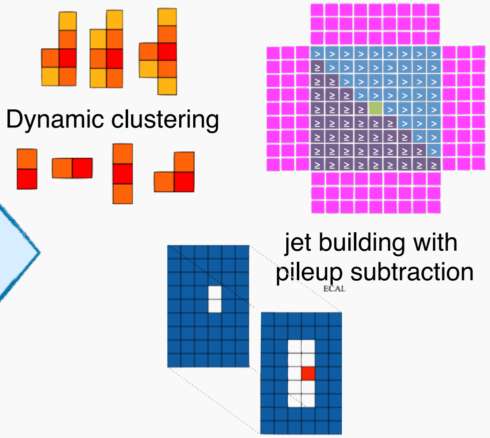
- ECAL/HCAL front-end, shapes, digitises and sums energy in trigger primitives

2: Level-1 Trigger

- Dedicated high-speed processors to apply clustering algorithms and programmable E_T thresholds

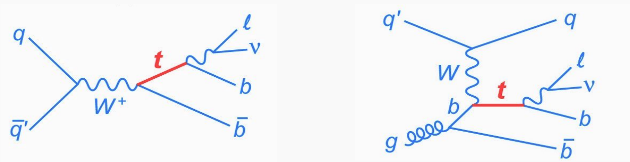
3: High-Level triggers

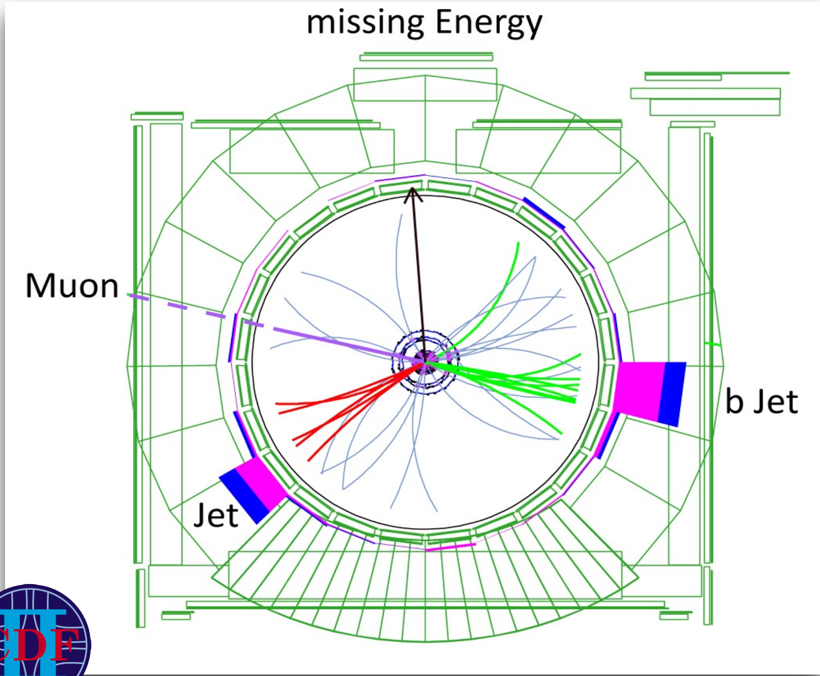
- Topological variables and tracking information
  - e/jet separation using cluster shapes
  - e/γ separation using tracking
- Isolation criteria
- Close to offline



Shape veto,
H/E, isolation, calibration

CDF - Multi objects trigger





CDF single top event

Signal characterisation:

- **1 high pT lepton**, in general isolated
- **Large MET** from high energy neutrino
- **2 jets**, 1 of which is a b-jets

L1-Trigger objects

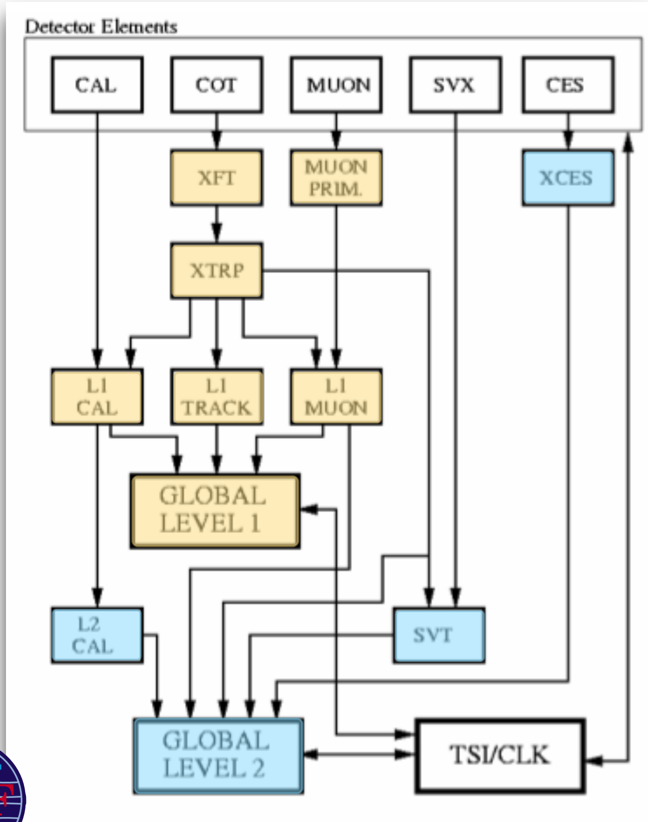
- Central tracking (XFT* p_T>1.5GeV)
- Calorimeter
  - Electron (Cal +XFT)
  - Photon (Cal)
  - Jet (Cal EM+HAD)
- Missing ET, SumET
- Muon (Muon + XFT)

L2-Trigger objects

- L1 information
- SVT (displaced track, impact parameter)
- Jet cluster
- Isolated cluster
- Calorimeter ShowerMax (CES)

*XFT=eXtremely Fast Tracker

CDF - Multi objects trigger



CDF single top event

Signal characterisation:

- **1 high pT lepton**, in general isolated
- **Large MET** from high energy neutrino
- **2 jets**, 1 of which is a b-jets

L1-Trigger objects

- Central tracking (XFT* p_T>1.5GeV)
- Calorimeter
  - Electron (Cal +XFT)
  - Photon (Cal)
  - Jet (Cal EM+HAD)
- Missing ET, SumET
- Muon (Muon + XFT)

L2-Trigger objects

- L1 information
- SVT (displaced track, impact parameter)
- Jet cluster
- Isolated cluster
- Calorimeter ShowerMax (CES)

*XFT=eXtremely Fast Tracker

Trigger efficiency

Trigger is just another "cut" in the physics analysis event selection

- Trigger efficiency must be precisely known for cross-section measurements, etc.
  - For each trigger algorithm, at each trigger level

$$\sigma_X = \frac{N_{candidates} - N_{bkg}}{A \cdot \epsilon_{total} \cdot \int \mathcal{L}\,dt}$$

$$A \cdot \epsilon_{total} = A \cdot \epsilon_{Tracking} \cdot \epsilon_{Reco} \cdot \epsilon_{L1-Trg} \cdot \epsilon_{L2-Trg} \cdot \epsilon_{L3-Trg} \cdot \epsilon_{vertex} \cdot \epsilon_{analysis}$$

Trigger efficiency measurement

Definition

- Usually measured w.r.t. offline-reconstructed objects
  - e.g. # triggered electrons vs # offline electrons

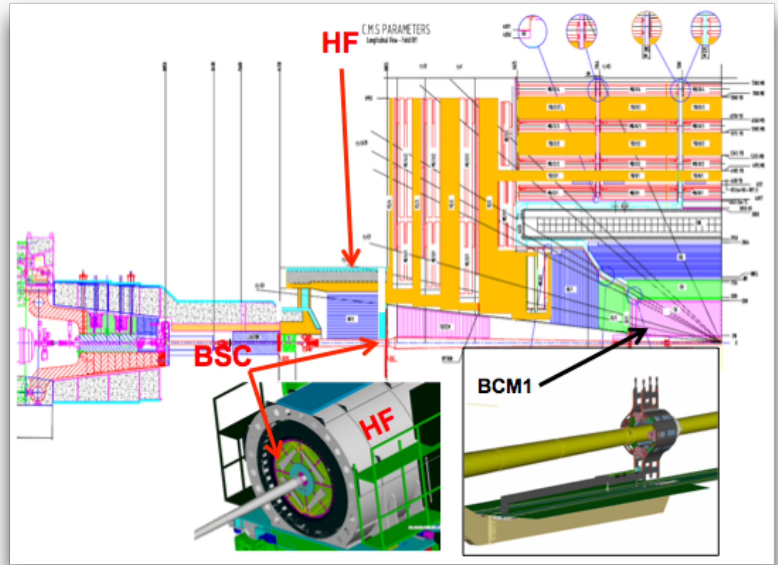
$$\text{efficiency}(\mathscr{A}) = \epsilon_{\mathscr{A}} = \frac{N^{\text{trigger}}_{\mathscr{A}}}{N^{\text{offline}}}$$

Measurement via

- Monte-Carlo simulation
  - This is not sufficient for analysis purposes; performance varies with lumi, time, detector performance
- Relative to a looser (prescaled) trigger
  - e.g. Use 40 GeV jet trigger to measure 60 GeV jet eff.
- Independent trigger
  - Trigger on one physics signature, measure a different one
- Tag-and-probe

Level-1 Trigger performance measurement

Forward, Beam scintillators (BSC), Beam pickups (BPTX)



$$\epsilon_{\mathscr{A}} = \frac{N^{\text{trigger}}_{\mathscr{A}}}{N^{\text{offline}}_{\text{minbias}}}$$

- Relative to **zero and minimum bias** (MB) triggers:
  - i.e. triggers with no requirements at all - a.k.a. “did something happen yet?”
  - Recorded in small quantities (heavily prescaled) for offline trigger performance estimate
- **Zero bias**: e.g. trigger on random (filled) bunches
- **Minimum bias**: trigger on minimum detector activity

Trigger efficiency measurement - High Level Trigger

$$\epsilon_{\mathcal{A}} = \frac{N_{\mathcal{A}}^{\text{trigger}}}{N_{\text{passthrough}}^{\text{offline}}}$$

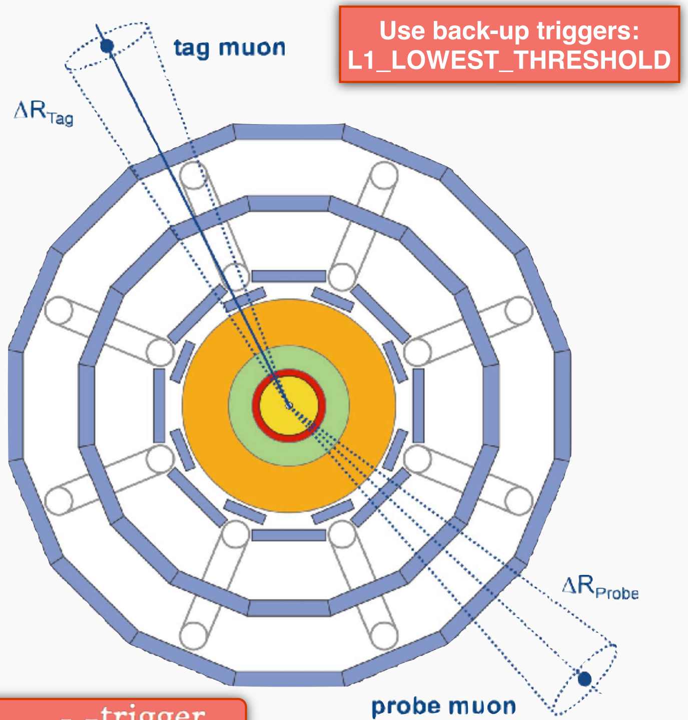
Efficiency easily measurable exploiting the L1 pass-through back-up triggers

L2 muon with p_T>10 GeV

$$\text{Eff (L2MU10)} = \frac{\text{\# Events passing L2MU10}}{\text{\# Events passing L2MU10\_PASSTHROUGH}}$$

"**L2MU10_PASSTHROUGH**" selected events
where the Level-1 trigger had already found a muon,
but **does not apply** L2MU10

Efficiency measurement with “Tag and Probe” technique



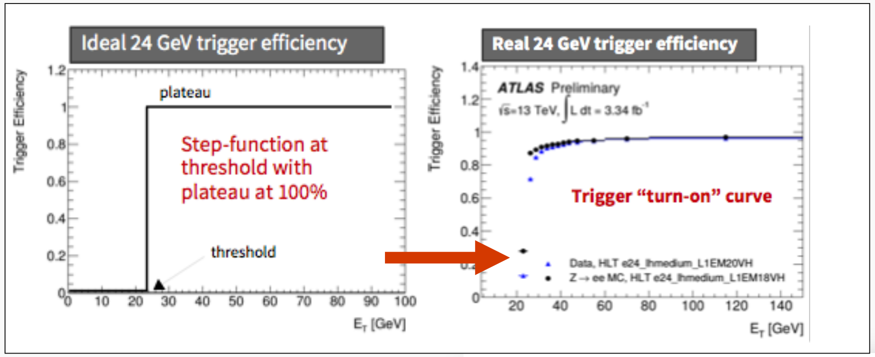
$$\epsilon_{\mathscr{A}} = \frac{N^{\text{trigger}}_{\mathscr{A}}}{N^{\text{offline}}_{\text{probe}}}$$

Exploit a well know physics process (e.g. Z→ll) to select a very clean sample

- Applicable on specific signatures (typically leptons)
- Requires careful fake control

How?

- **Online**: Trigger on independent signature (the **Tag**)
- **Offline**: Reconstruct the event and identify the candidate signature (the **Probe**)
  - e.g, tight offline requirements and Z mass selection
- **Offline**: measure trigger efficiency on the Probe



Trigger turn-on curves

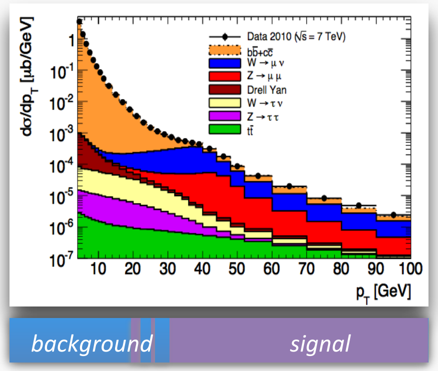
Trigger pT thresholds do not result step function in pT due to

- Resolution Inefficiencies
- Trigger/offline differences

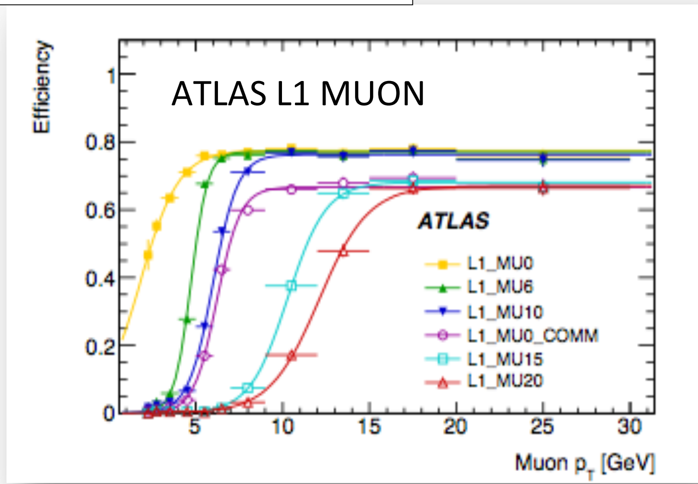
Trigger behaviour better described by Error function, usually called **trigger turn-on**

Understanding of the step region is critical

- Efficiency changes very quickly and contamination from background can be important
  - **Sharp**, better background suppression
  - **Slow**, can be better extrapolated and systematic error can be reduced



$\mu\ p_T$



$$rate = \int L \bullet d\sigma / dp_T \bullet \varepsilon(p_T) = \int rate(p_T) \bullet \varepsilon(p_T) \bullet dp_T$$

Efficiency dependencies on…

The trigger behaviour, can vary rapidly due to significant changes in

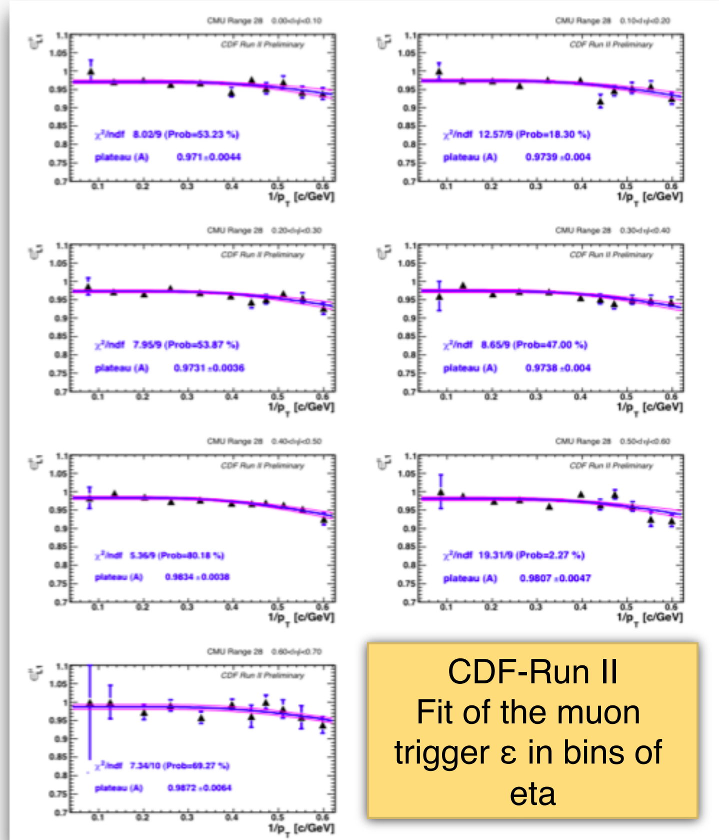
- Detector
- Trigger hardware
- Trigger algorithms
- Trigger definition

Analysis must track of all these changes

Multi-dimensional study of the efficiency:

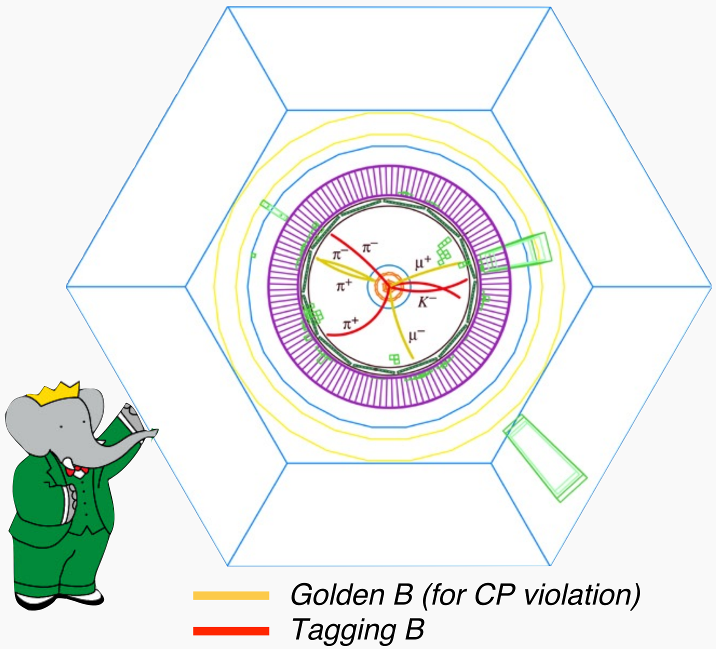
$\varepsilon(p_T, \eta, \varphi, \text{run\#})$

- Fit the turn-on curves for different bins of η, φ, p_T



CDF-Run II
Fit of the muon trigger ε in bins of eta

BaBar - exclusive trigger for precision measurements



Golden event in the BaBar Detector e^+e^- collision producing a B and an anti-B

Primary goal
minimise the trigger efficiency systematics by selecting a very specific signature

Trigger objects

- Charged tracks in the drift chamber, with different pT cuts: long track (0.18 GeV), short track (0.12 GeV)
- EM calorimeter clusters with different ET cuts

Search for well-defined topology

- Number of objects, optional geometrical separation cuts or matching between tracks and clusters

Accurate studies on signal and background to determine the efficiency measurement error

How many signatures & algorithms?

Physics triggers

- **Discovery experiments**: multiple inclusive selections ensure wide open search windows
- **Precision experiments**: multiple triggers for multiple measurements

Calibration triggers

- Detectors calibrations
- Detectors and trigger efficiency measurements
- Tagging efficiency
- Energy scale measurements

Background triggers

- Instrumental and physics background
- Better description of the background can be extrapolated from data than from Monte Carlo
- Understand resolutions, including the under-threshold population

Monitor triggers

- To monitor the trigger itself (remember, lost events are lost for ever!)

Bulk of the recorded events

Rate allocation

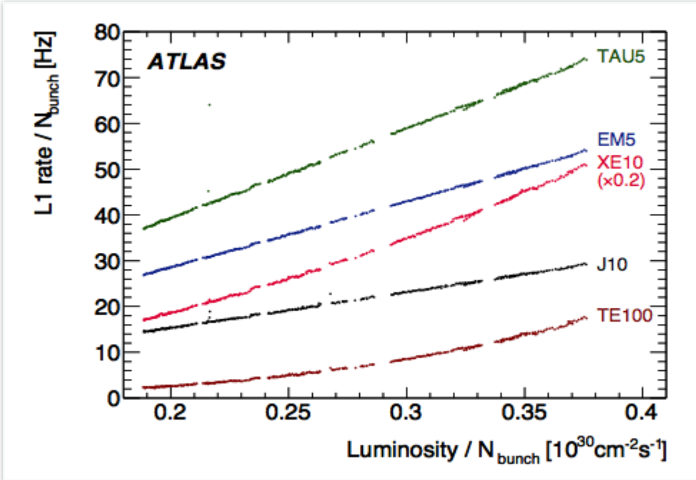
Target : the final allowed DAQ bandwidth

The rate allocation to each trigger signature

- Physics goals (plus calibration, monitoring samples)
- Required efficiency and background rejection
- Bandwidth consumption

$$R_i = \mathscr{L} \int \frac{d\sigma}{dp_T} \cdot \epsilon_i (p_T) \, dp_T$$

Rates scale linearly with luminosity,
Pile-up effects break linearity.



Rate extrapolation

- **Trigger design and commissioning**: use large samples of simulated data, including large cross-section backgrounds
  - Large uncertainties due to detector response and background cross-sections
- **During running** (at colliders), (some) rates can be extrapolated to higher Luminosity

The neverending struggle: Physics interest vs system bandwidth…

Lower thresholds would be desirable, but the physics coverage must be balanced against considerations of the offline computing cost

- **How accommodate a broad physics program?**
- **And cope with increasing rates?**

Trigger menus!

Trigger Menu

Defines the Physics program/reach of the experiment

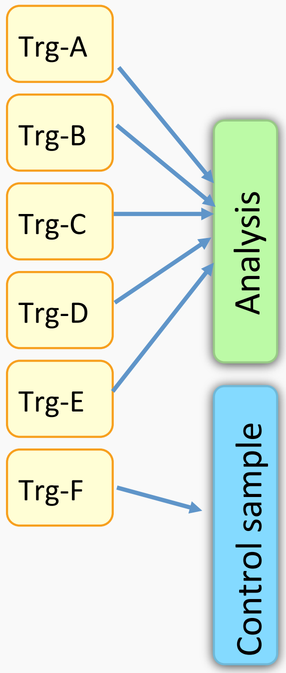
- Collection of physics trigger, associated back-ups, triggers for calibration and monitoring

It must be

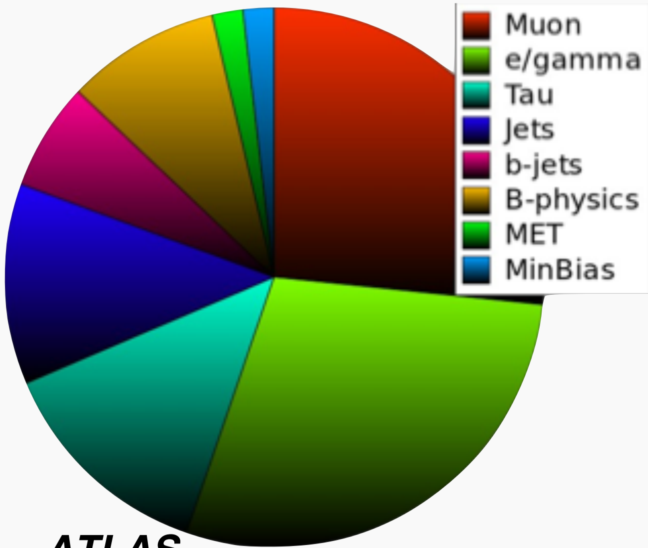
- Redundant to ensure data collection efficiency
- Sufficiently flexible to face possible variations of the environment and physics goals
  - e.g. detectors, machine luminosity,…

Central to the physics program

- Each analysis served by multiple triggers and different samples
  - from the most inclusive to the most exclusive
- Ideally, it will collect events (some, at least) from all relevant processes
  - (to provide physics breadth and control samples)



Trigger strategy @ colliders: the ATLAS menu



ATLAS
Trigger rates per signature at 10^{33}

Inclusive triggers for signal samples

- **Single high-p_T objects**
  - e/μ/γ (p_T>20 GeV)
  - jets (p_T>100 GeV)
- **Multi-object events**
  - e-e, e-μ, μ-μ, e-τ, e-γ, μ-γ, etc… to further reduce the rate

Back-up triggers for monitoring and performance studies (often pre-scaled)

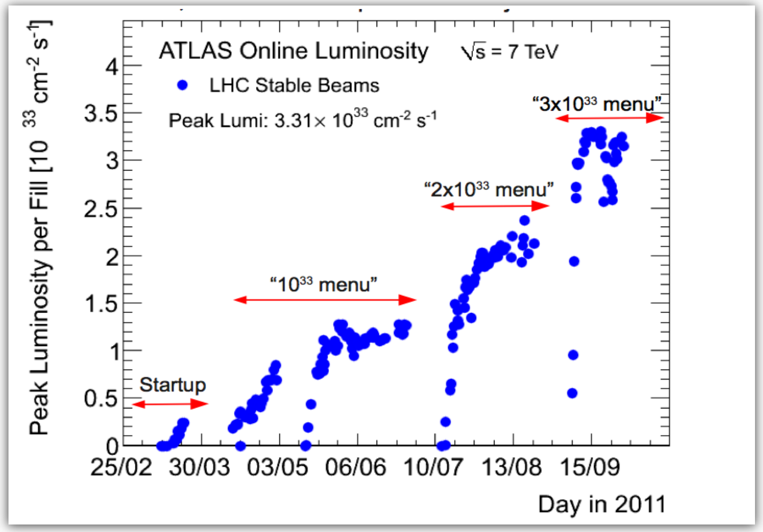
- Jets (p_T>8, 20, 50, 70 GeV)
- Inclusive leptons (p_T > 4, 8 GeV)
- Lepton + jet

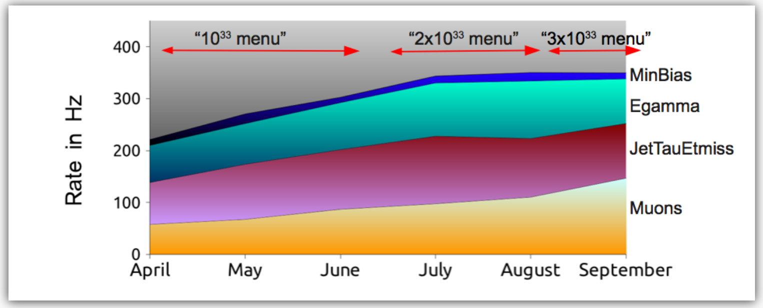
Example: ATLAS menu $3x10^{33}$

Priority List for >300 Hz

Chain	Physics	Unique rate L1 (Hz)	Unique rate L2 (Hz)	Unique rate EF (Hz)	*Sorted by* Problem level
EF_xe60_verytight_noMu	SUSY/Exotics	0	0	0.5	EF (pileup)
EF_j100_a4tc_EFFS_ht400	SUSY	0	0	2.5	EF
EF_4j45_a4tc_EFFS	SUSY/SM	0	0	2	EF
EF_5j30_a4tc_EFFS		0	5	3	EF
EF_j240_a10tc_EFFS	Exotics/SM	0	0	1	EF
EF_tau29_loose1_xs45_loose_noMu_3L1J10	Higgs	0	40	5	EF
EF_b10_medium_4j30_a4tc_EFFS	Top/Higgs	0	4	10	EF
EF_2mu4_BmumuX	B-physics	0	7	0.9	EF
EF_2mu4_Jpsimumu		0	6	1.7	EF
EF_mu4mu6_DiMu		0	25	6.5	EF
EF_mu4mu6_DiMu_DY20	SM	0	10	5?	EF
EF_2MUL1_l2j30_HV_allMS	Exotics	0	?	?	EF
EF_mu20i_medium	$5x10^{33}$ prep.	0	15	3	EF
EF_mu18_MG_medium	Many	0	0	60	EF
EF_mu18_medium		0	0	60	EF
EF_e60_loose	(Exotics)	0	5	7	EF,client
EF_mu15/18/22_njX?	SUSY/??	100	10	?	EF,non-validated
EF_g22_hiptrt?	Exotics	0	?	$< 1?$	non-validated
EF_e15_medium_xe40_noMu	SUSY/Exotics	310	70?	1.3	L2 (pileup)
EF_j55_a4tc_EFFS_xe55_medium_noMu_dphi2j30xe10		70	210	1.5	L2
EF_e10_medium_mu6_topo_medium	Higgs	1200	9	1	L1
EF_tau20_medium_e15_medium	Higgs	3700	10	1	L1
EF_xe60_tight_noMu	SUSY	680?	150?	1	L1,L2 (pileup),EF
EF_e10_medium_mu6	Higgs/SUSY	1200	75	10	L1, EF
EF_l2j30_Trackless_HV_L1MU6	Exotics	1500?	0.5	0.5	L1
Total extra rate		6500	600	100	Peak at 3×10^{33}

The ATLAS trigger during commissioning





ATLAS start-up in 2008: **$L=10^{31}$ cm^{-2} s^{-1}**

- **Level-1**: Low p_T thresholds and loose selection
  - In parallel, deploy high thresholds and multi-objects triggers for validation (to be used as back-up triggers)
- **HLT**: running in pass-through mode for offline validation or with low thresholds

Evolved rapidly with the increase in LHC luminosity

- Increased p_T thresholds
- Algorithms for complex signatures added
- Maintain stable trigger conditions for important physics results (for conferences)
- Maintained the balance between physics streams
  - electrons, muons, jets, minimum-bias

Inclusive trigger example: from CDF

Trigger Chain: Inclusive High-p_T Central Electron

Level 1

- EM Cluster $E_T > 8$ GeV
- Rϕ Track $p_T > 8$ GeV

Level 2

- EM Cluster ET > 16 GeV
- Matched Track $p_T > 8$ GeV
- Hadronic / EM energy < 0.125

Level 3

- EM Cluster $E_T > 18$ GeV
- Matched Track $p_T > 9$ GeV
- Shower profile consistent with e^-

To efficiently collect
W, Z, tt, tb, WW, WZ, ZZ,
Wγ, Zγ, W', Z', etc...

✓Use resolution at L2/L3 to improve purity
✓only really care about L1 efficiency near L2 threshold

Back-up trigger example: from CDF

Back-up Triggers for central Electron 18 GeV

W_NOTRACK
- L1: EMET > 8 GeV && MET > 15 GeV
- L2: EMET > 16 GeV && MET > 15 GeV
- L3: EMET > 25 GeV && MET > 25 GeV

NO_L2
- L1: EMET > 8 GeV && rφ Track pT > 8 GeV
- **L2: AUTO_ACCEPT**
- L3: EMET > 18 GeV && Track pT > 9 GeV && shower profile consistent with e-

NO_L3
- L1: EMET > 8 GeV && rφ Track pT > 8 GeV
- L2: EMET > 8 GeV && Track pT > 8 GeV && Energy at Shower Max > 3 GeV
- **L3: AUTO_ACCEPT**

L2/L3 Passthrough

✓ **Factorize efficiency into all the components:**
- ✓ efficiency for track and EM inputs determined separately
- ✓ separate contributions from all the trigger levels

Redundant, inclusive trigger example: from CDF

L1_EM8_PT8 feeds - inclusive

- Inclusive high-pT central electron chains
- Di-lepton chains (ee, eμ, eτ)
- Several back-up triggers
- 15 separate L3 trigger chains in total

A ttbar cross section analysis uses

- Inclusive high-p_T central e chains
- Inclusive high-p_T forward e chains
- MET + jet chains
- Muon chains

Trigger menus must be

Inclusive:
Reduce the overhead for the program analysis

Redundant:
Issues in a single one detector or in a trigger input do not affect physics (reduced efficiency but still the measurement is possible)

Concluding remarks

The Data AcQuisition is a key element of any experiment

- Robustness and reliability essential for good data and smooth operations

The trigger strategy is a trade-off between physics requirements and affordable systems and technologies

- A good design is crucial – then the work to maintain optimal performance is easy

Here we just reviewed the main trigger requirements coming from physics

- High efficiency – rate control
- Excellent knowledge of the trigger selection on signal and background
- Flexibility and redundancy

References

References

- *N. Ellis*, “Trigger and Data Acquisition”, Proceedings from the 5th CERN-Latin American School of High-Energy Physics, 15 - 28 Mar 2009, Colombia, CERN Report Number CERN-2010-001.
- *A. Hocker*, “Trigger and Data Analysis”, Lectures at HCPSS, 8-17 June 2009, CERN.
- *R. Frühwirth*, *M. Regler*, *R.K. Bock*, *H. Grote* and *D. Notz*, “Data Analysis Techniques for High-Energy Physics”, Cambridge University Press, 2nd Edition, August 200 , ISBN 0521635489.
- F. Winklmeier “Particle Detectors – Trigger/DAQ”, CERN Academic Training Lecture Programme, 12th May 2016.