Introduction to Triggering

Thanks to Monika Wielers and Alessandro Thea for providing material.

What is a trigger?

- You might think of this

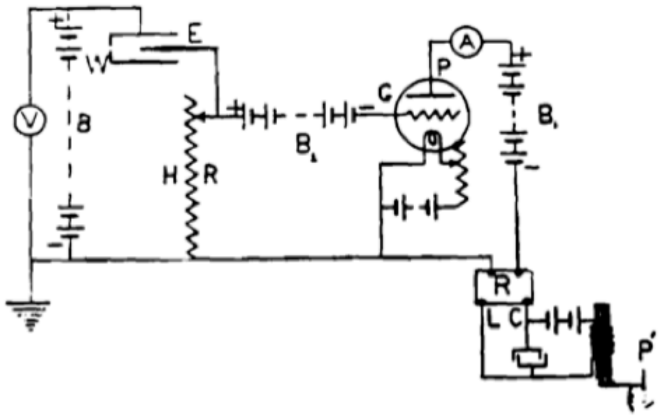
- But that’s not what we are discussing here

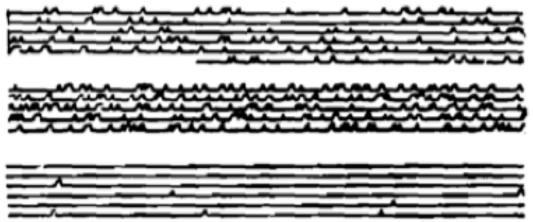
- Trigger – in general:
  - Something which tells you when to take your data
  - If you can’t keep everything, process data to decide quickly what to keep
  - Uses “simple” criteria to make the decision
  - As we just heard, Trigger and DAQ are ‘interwoven’

- Concentrate here on LHC trigger systems (apologies for the ATLAS bias)

Not a new concept

ON THE AUTOMATIC REGISTRATION OF
α-PARTICLES, β-PARTICLES
AND γ-RAY AND X-RAY PULSES





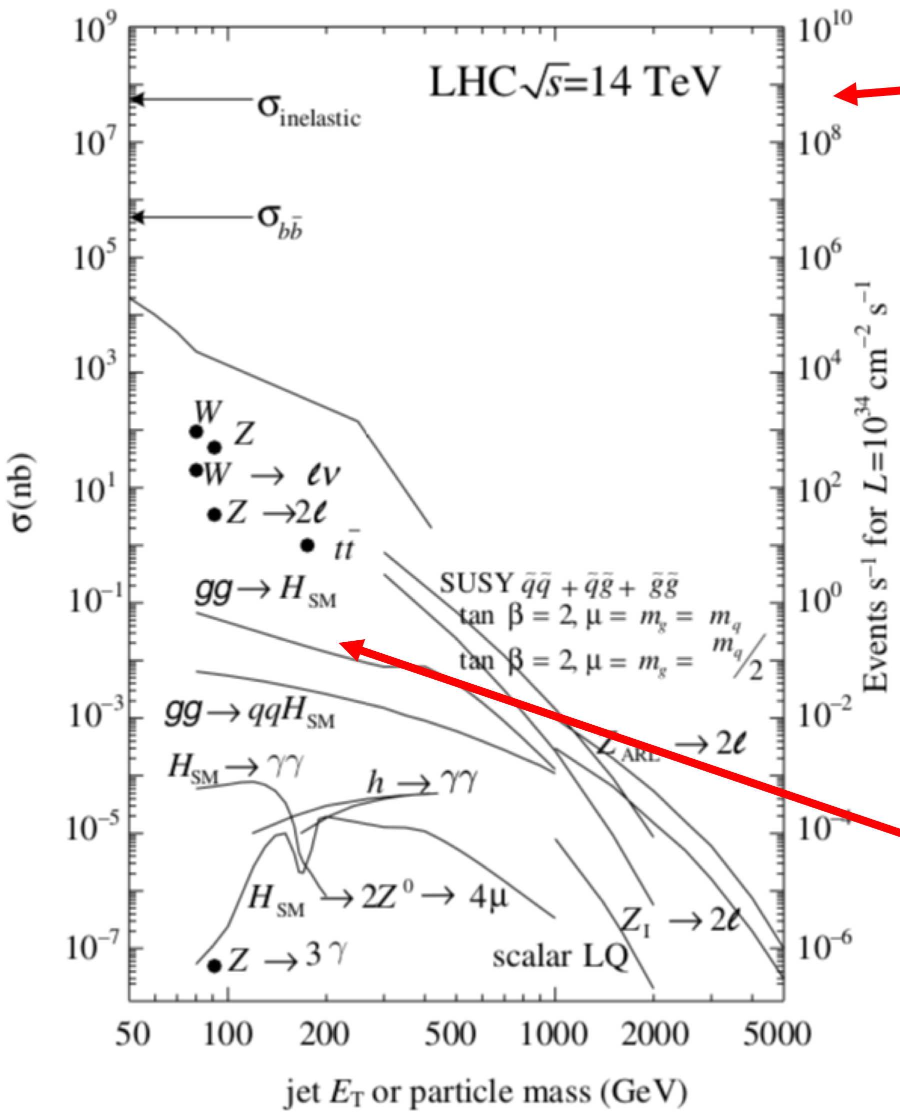
Alois F. Kovarik
Sheffield Scientific School
Yale University
New Haven, Conn.
January 25, 1919

Phys. Rev. 13, 272 , 1st April 1919

"… visual or audible methods of counting are quite trying on the nerves … A self-recording device would therefore be an obvious improvement."

A lot has happened since then
But trigger and DAQ can still be quite trying on the nerves

What is the problem?



Inelastic cross-section -> GHz of events (40MHz bunch crossing x ~40 p-p interactions per crossing)

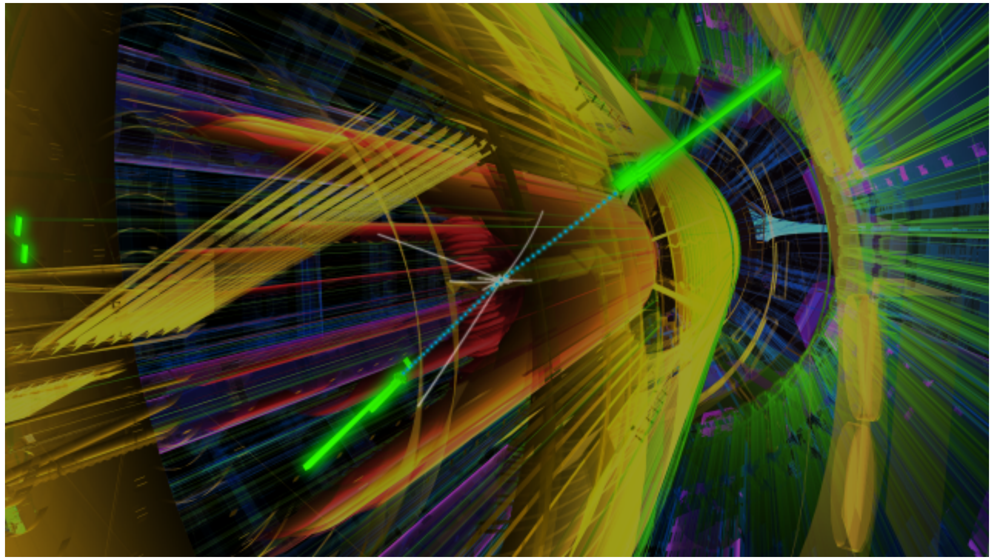
Process	Cross-section (nb)	Production Rate (Hz)
Inelastic	10^8	10^9
$b - bbar$	5×10^5	5×10^6
$W \rightarrow l\nu$	15	150
$Z \rightarrow l\nu$	2	20
$t - tbar$	1	10
$H(125)SM$	0.05	0.5

Some of the interesting processes only happen very rarely : ~1 in 10^{11}

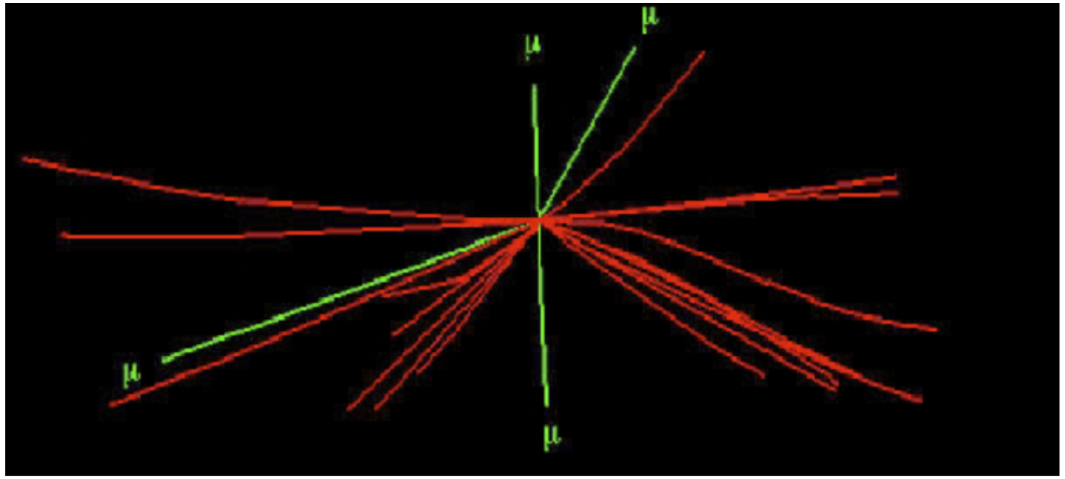
Any we want to find them!

How do we decide which events are interesting??

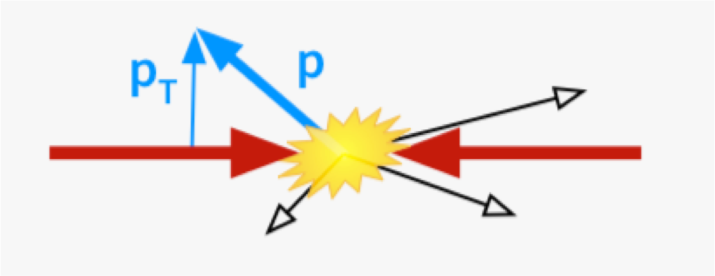
- One Higgs in every 10 billion pp interactions
- $H \rightarrow \gamma\gamma$ is even rarer, BR~10^{-3}
- 1 $H \rightarrow \gamma\gamma$ per 10 trillion interactions
- Need to find them all



How do we decide which events are interesting??

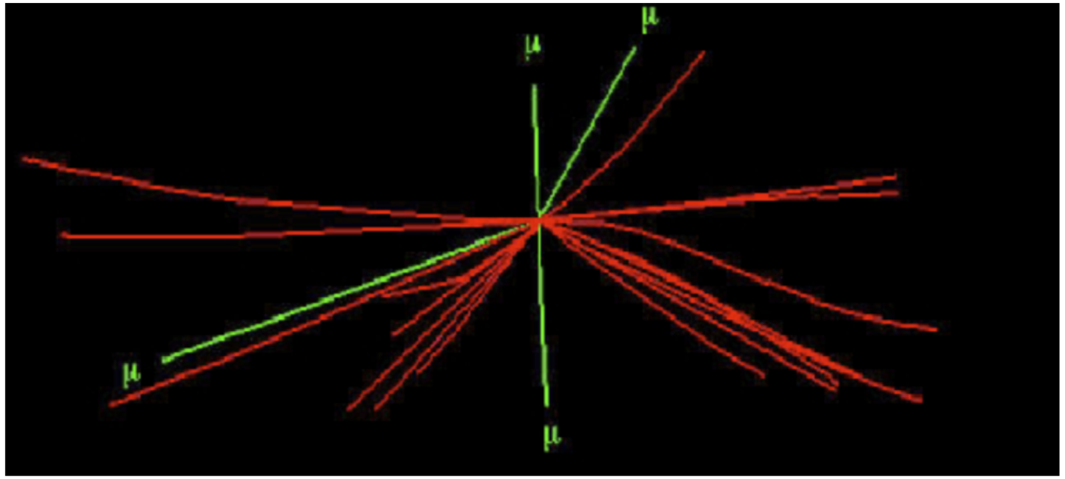


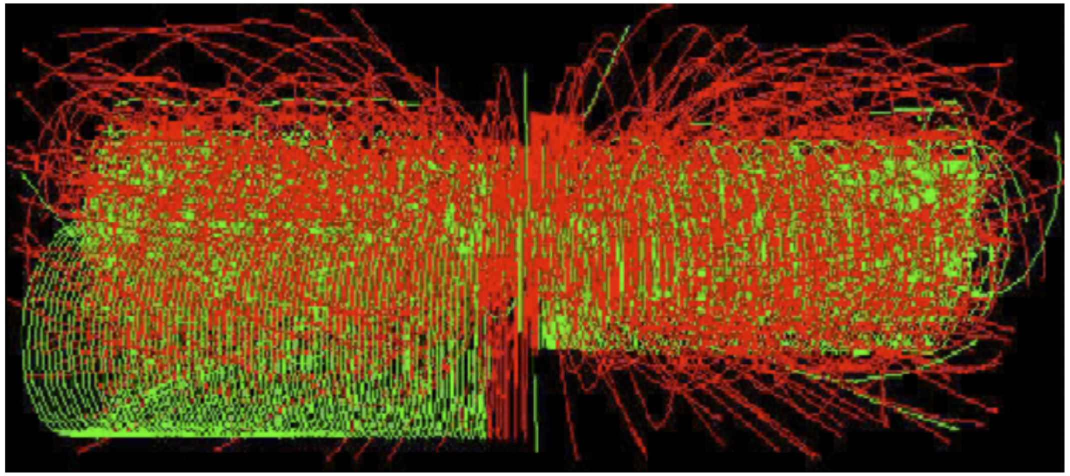
Interesting events usually have high pT particles



- $H \rightarrow 4\mu$, $p_T(\mu)$ ~ 30-50 GeV
- $H \rightarrow \gamma\gamma$, $p_T(\gamma)$ ~ 50-60 GeV

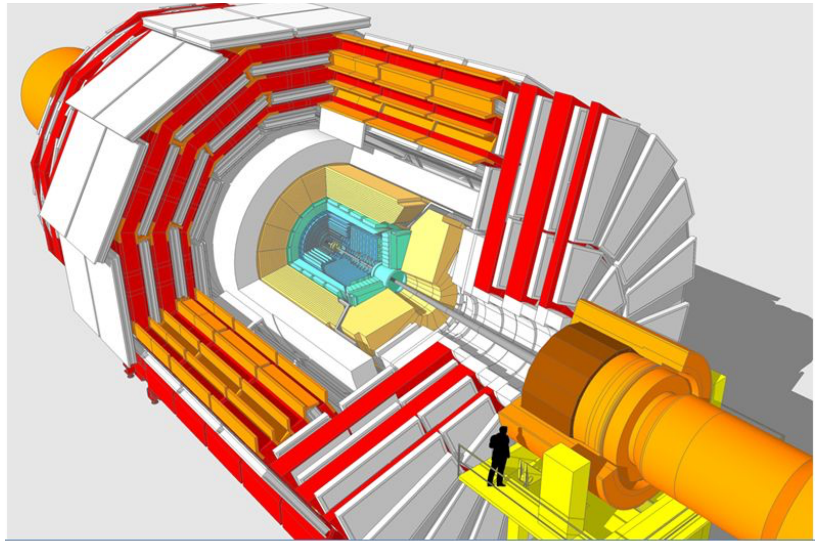
How do we decide which events are interesting??





- But hidden in a mass of low-pT pileup (~98%)
- Need sophisticated algorithms….
- Would be great to record all the data and sort it out “at leisure” offline
- BUT…….

Data volumes are a challenge

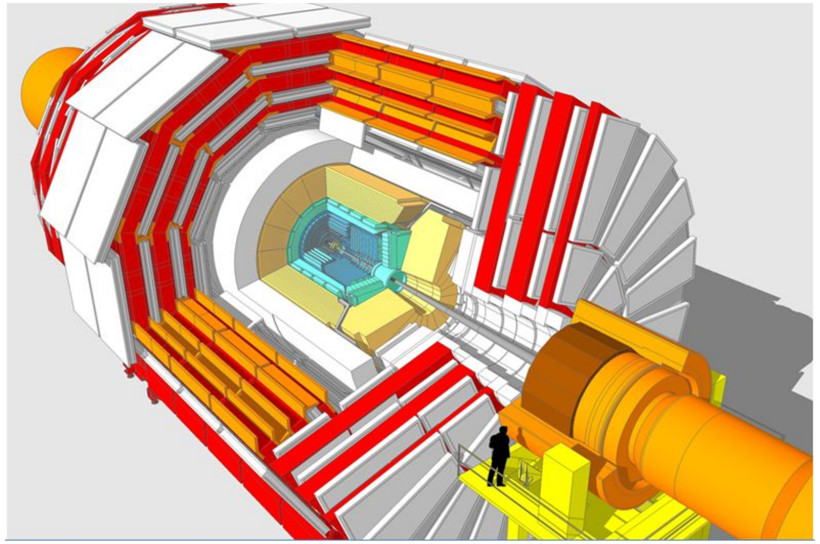


- Modern large-scale experiments are **BIG**

e.g. LHC experiments:

- ~100M channels
- ~1-2 MB raw data per measurement

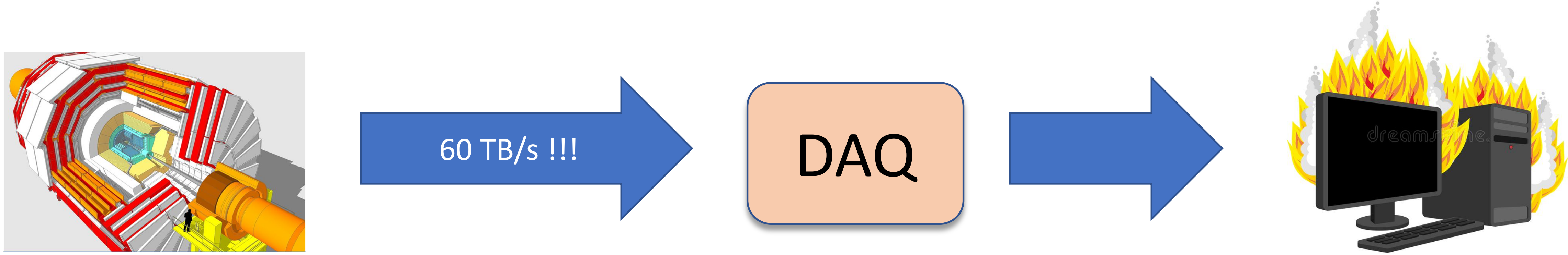
Data volumes are a challenge



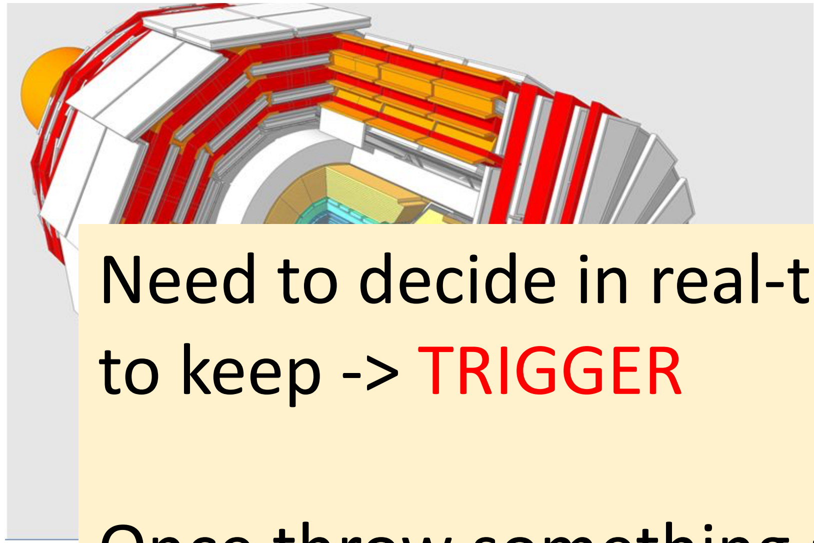
Modern large-scale experiments are **BIG**

e.g. LHC experiments:
- ~100M channels
- ~1-2 MB raw data per measurement

…. and **FAST** : 40MHz bunch-crossing rate



Data volumes are a challenge



Modern large-scale experiments are **BIG**

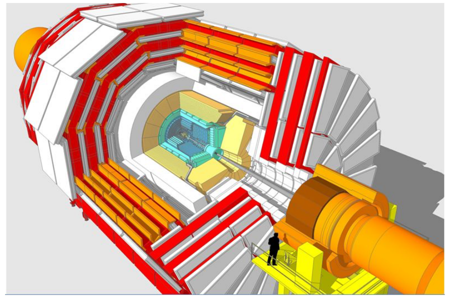
e.g. LHC experiments:

rement

Need to decide in real-time which collisions (“events”) to keep -> TRIGGER

Once throw something away it is gone for ever – better get it right first time!

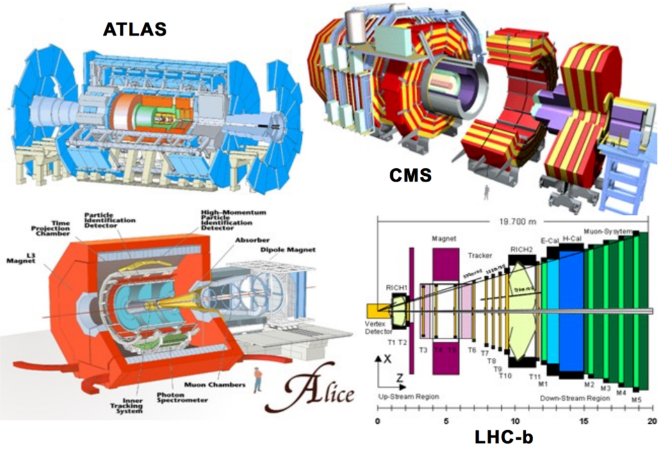
…. and **FAS**

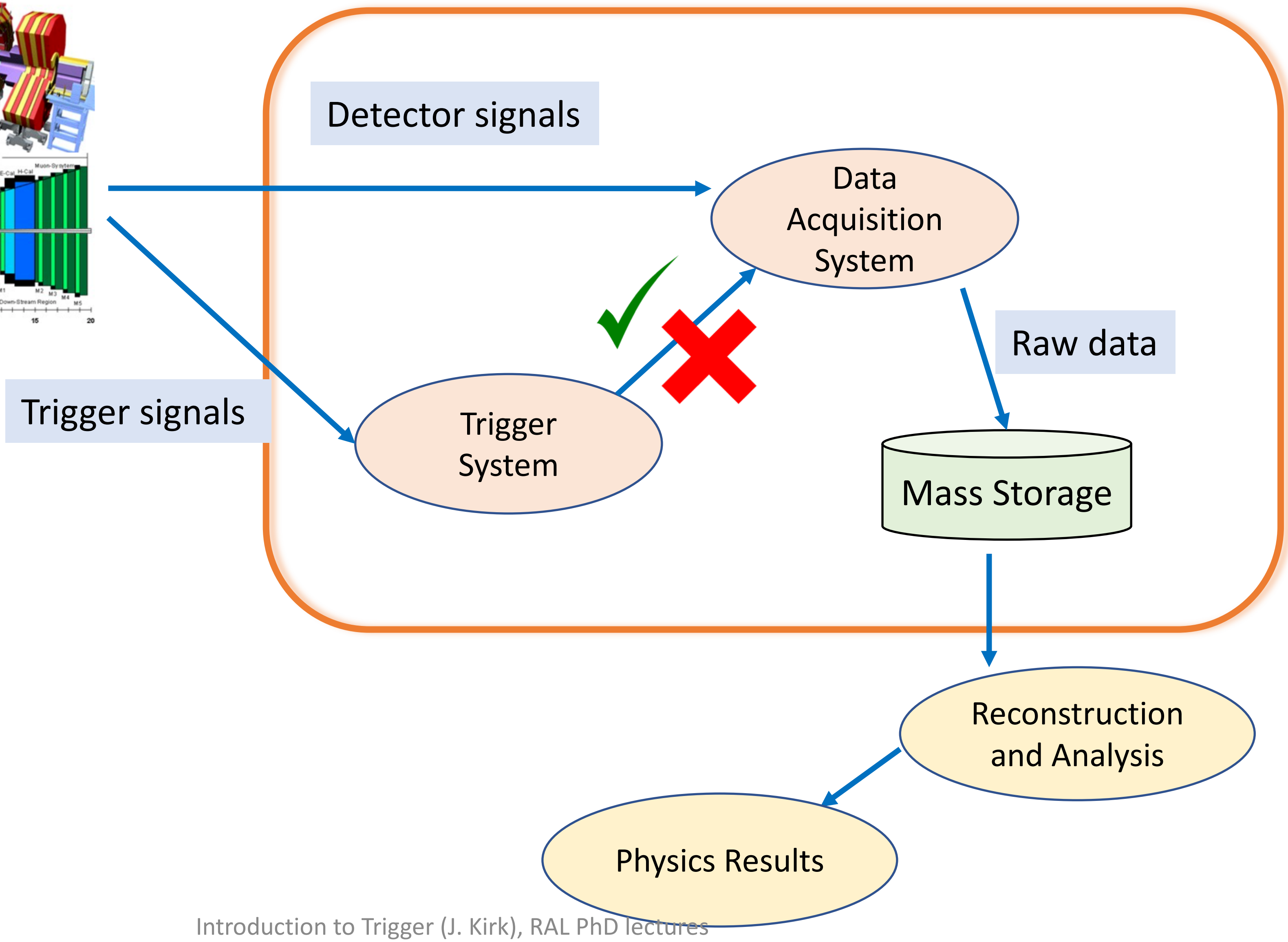


60 TB/s !!!

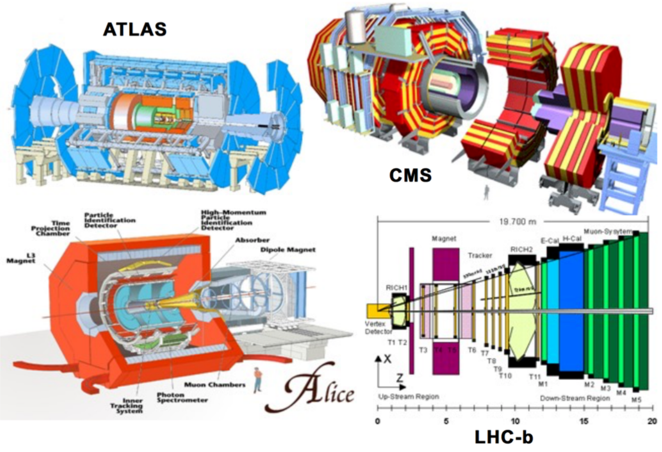
DAQ

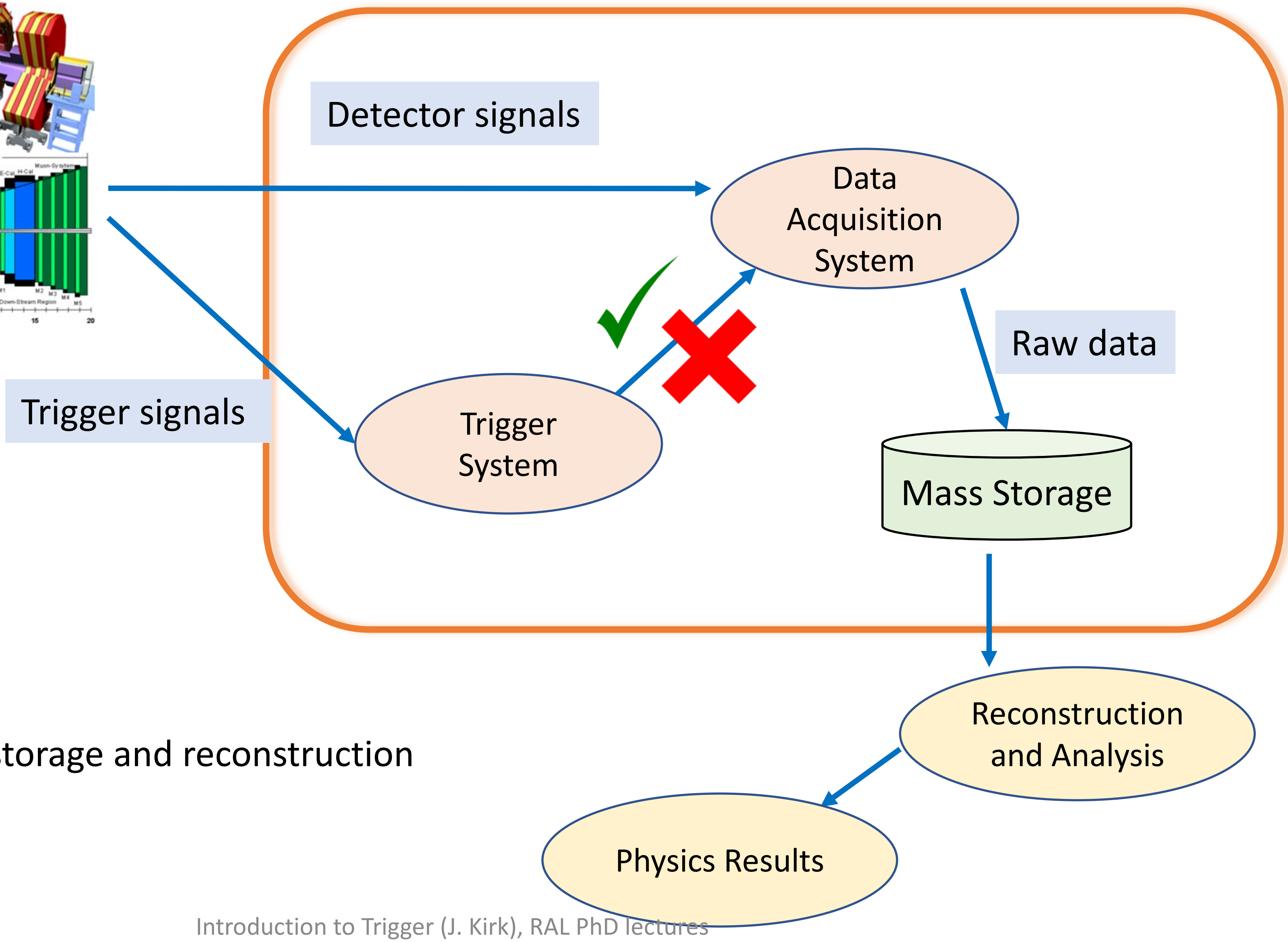
General overview





General overview





Trigger must:

- Take a QUICK look at events
- Select the most INTERESTING
- Keep rate LOW ENOUGH for storage and reconstruction

Keep rate low enough

DAQ system collects the data from different parts of the detector, converts data to a suitable format and saves it to permanent storage

Bandwidth constrained by:

- Finite storage capacity
- Cost of computing power (used online and offline)

Average event size determines the trigger rate:

$$BW_{DAQ} = Rate_{Trig}^{max} \; x \; Size^{Ev}$$

Keep rate low enough

DAQ system collects the data from different parts of the detector, converts data to a suitable format and saves it to permanent storage

Bandwidth constrained by:

- Finite storage capacity
- Cost of computing power (used online and offline)

Average event size determines the trigger rate:

Number of FE channels
Number of particles per event
~ 1-2 MB

$$BW_{DAQ} = Rate_{Trig}^{max} \; x \; Size^{Ev}$$

~ 1 GB/s

Keep rate low enough

DAQ system collects the data from different parts of the detector, converts data to a suitable format and saves it to permanent storage

Bandwidth constrained by:

- Finite storage capacity
- Cost of computing power (used online and offline)

Average event size determines the trigger rate:

$$BW_{DAQ} = Rate_{Trig}^{max} \; x \; Size^{Ev}$$

Number of FE channels
Number of particles per event
~ 1-2 MB

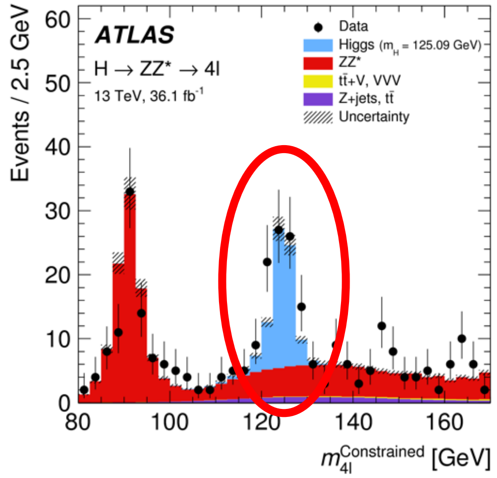
~ 1 GB/s

~ 1 kHz

Select interesting events

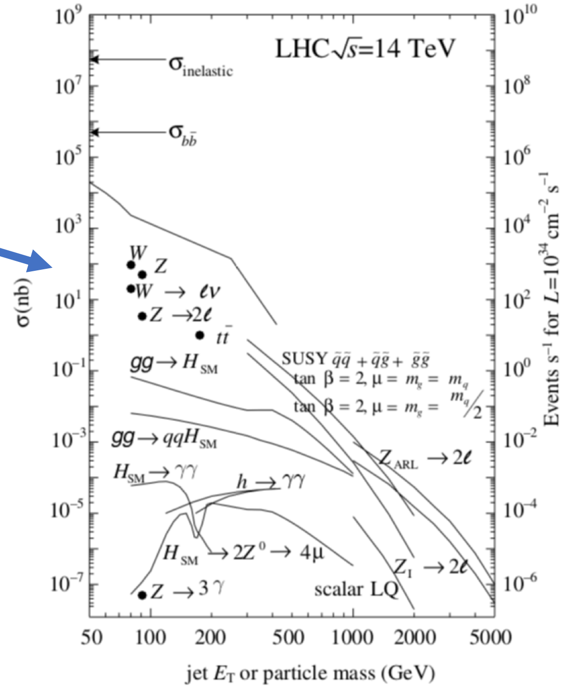
Physics goals:

- Higgs properties
- Search for Beyond Standard Model particles : SUSY, extra dimensions, new gauge bosons, black holes…..
- Many interesting Standard Model studies



All has to fit in ~1 kHz of data written to storage

Non-trivial: e.g. $W \rightarrow l\nu$ ~150 Hz (10^{34} $cm^{-2}s^{-1}$)

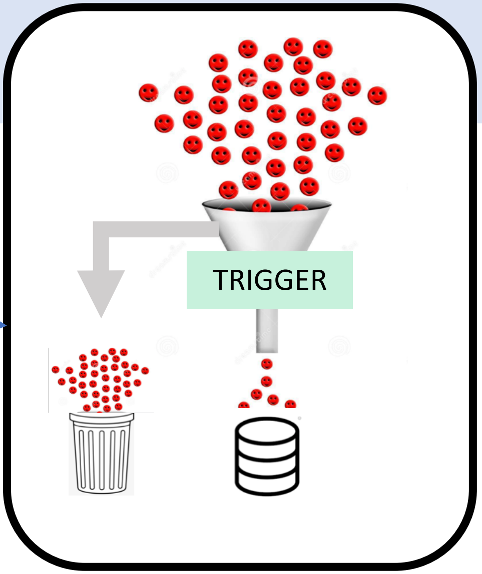


"Good" physics can become an enemy!!

- Trigger "menu" decides which events to keep (more later)

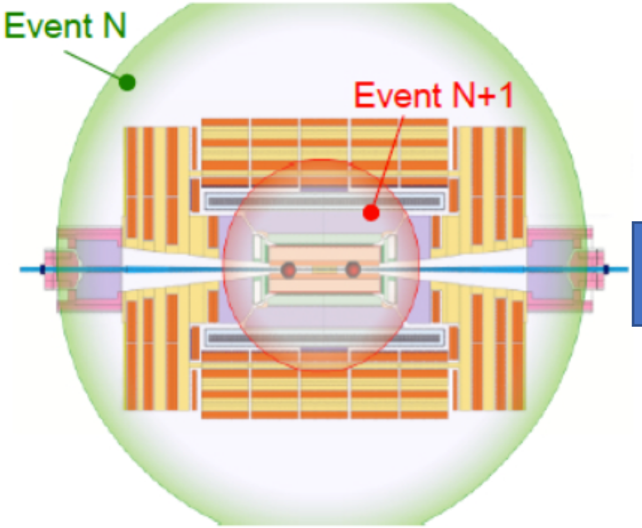
Needs to be FAST

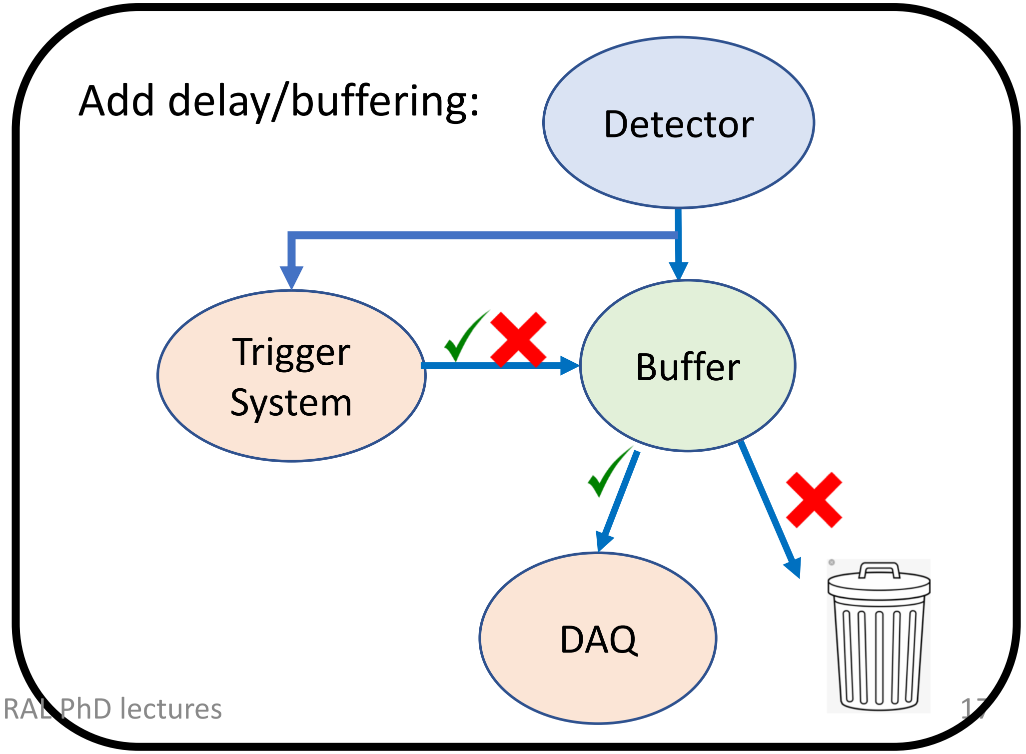
Ideally decide each bunch crossing whether or not to keep an event



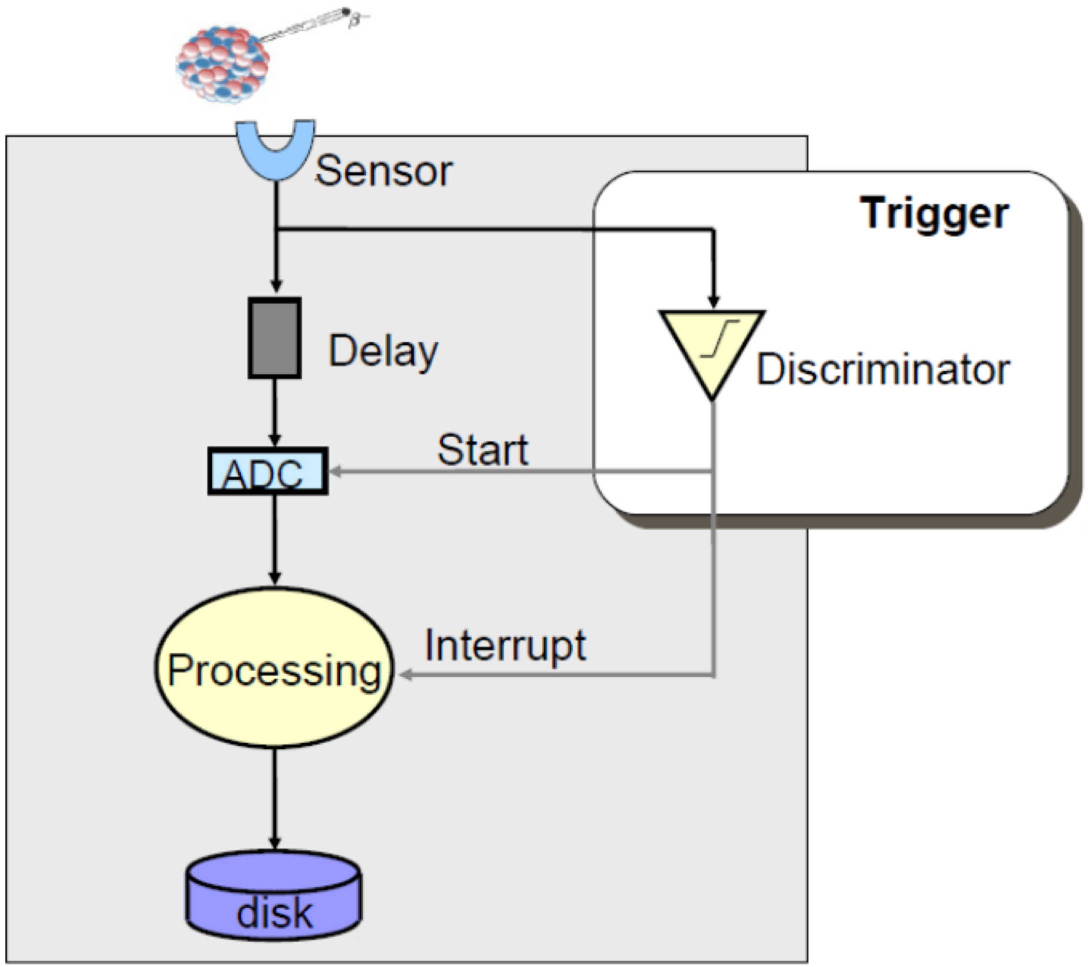
Huge amount of data per bunch crossing : O(10^6-10^8) channels

Some detectors need >25ns to readout and integrate signals





Simple trigger

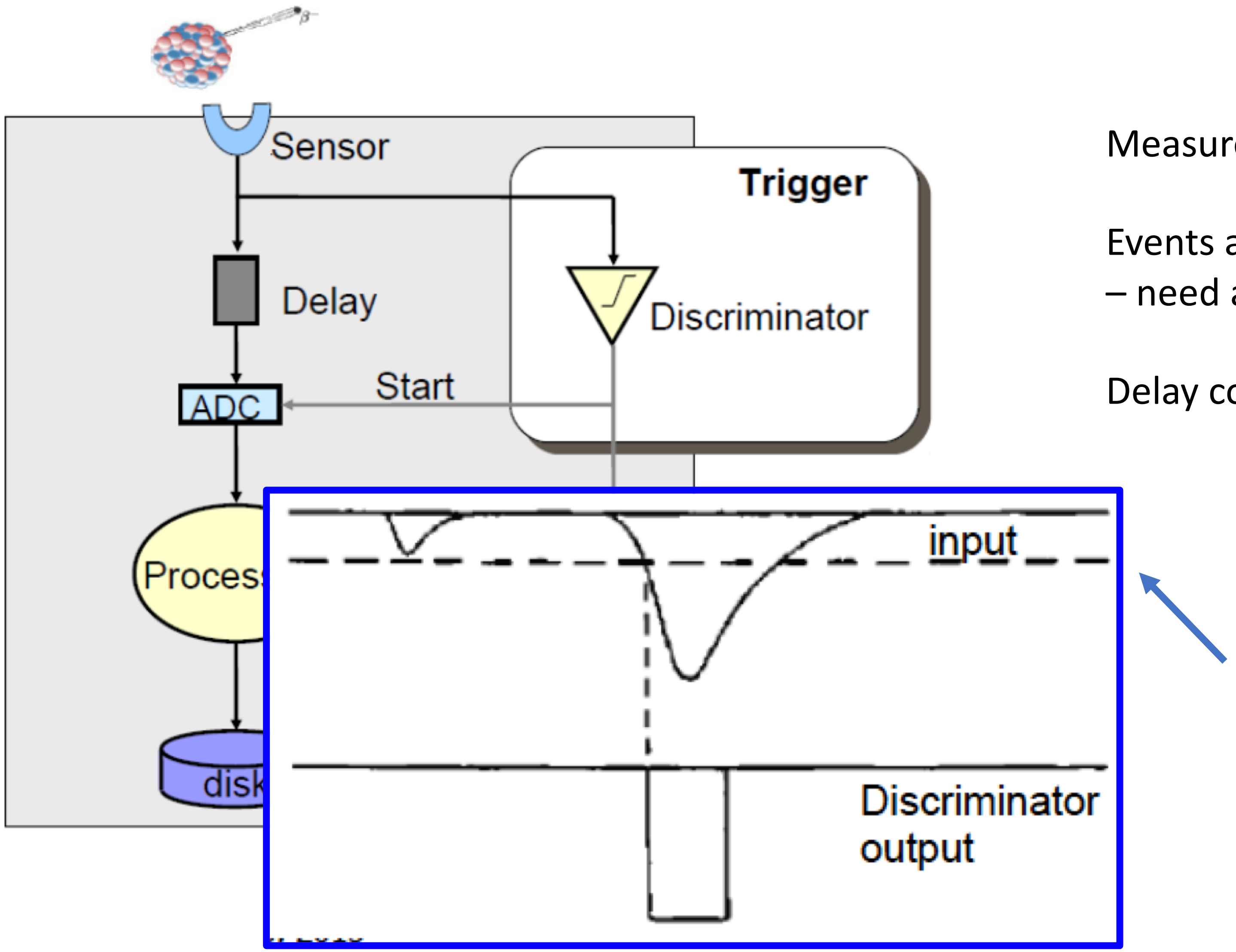


Measure β decay properties

Events are asynchronous and unpredictable – need a trigger

Delay compensates for trigger latency

Simple trigger



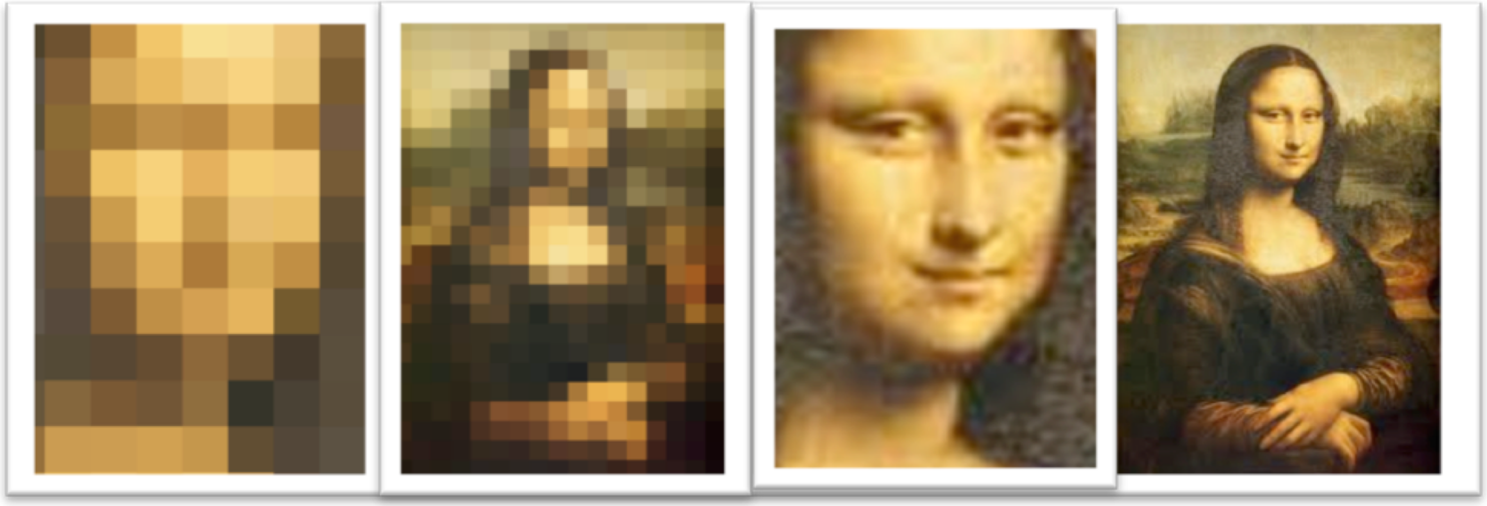
Measure β decay properties

Events are asynchronous and unpredictable – need a trigger

Delay compensates for trigger latency

Discriminator:
Generate output signal when input amplitude is greater than some threshold

Multi-level trigger



May not be possible to take final trigger decision in a single step:

- ➢ Multi-level triggers

More and more complex algorithms are applied on lower and lower data rates

- ➢ First level with short latency, working at higher rates
- ➢ Higher levels apply further rejection, with longer latency (more complex algorithms)

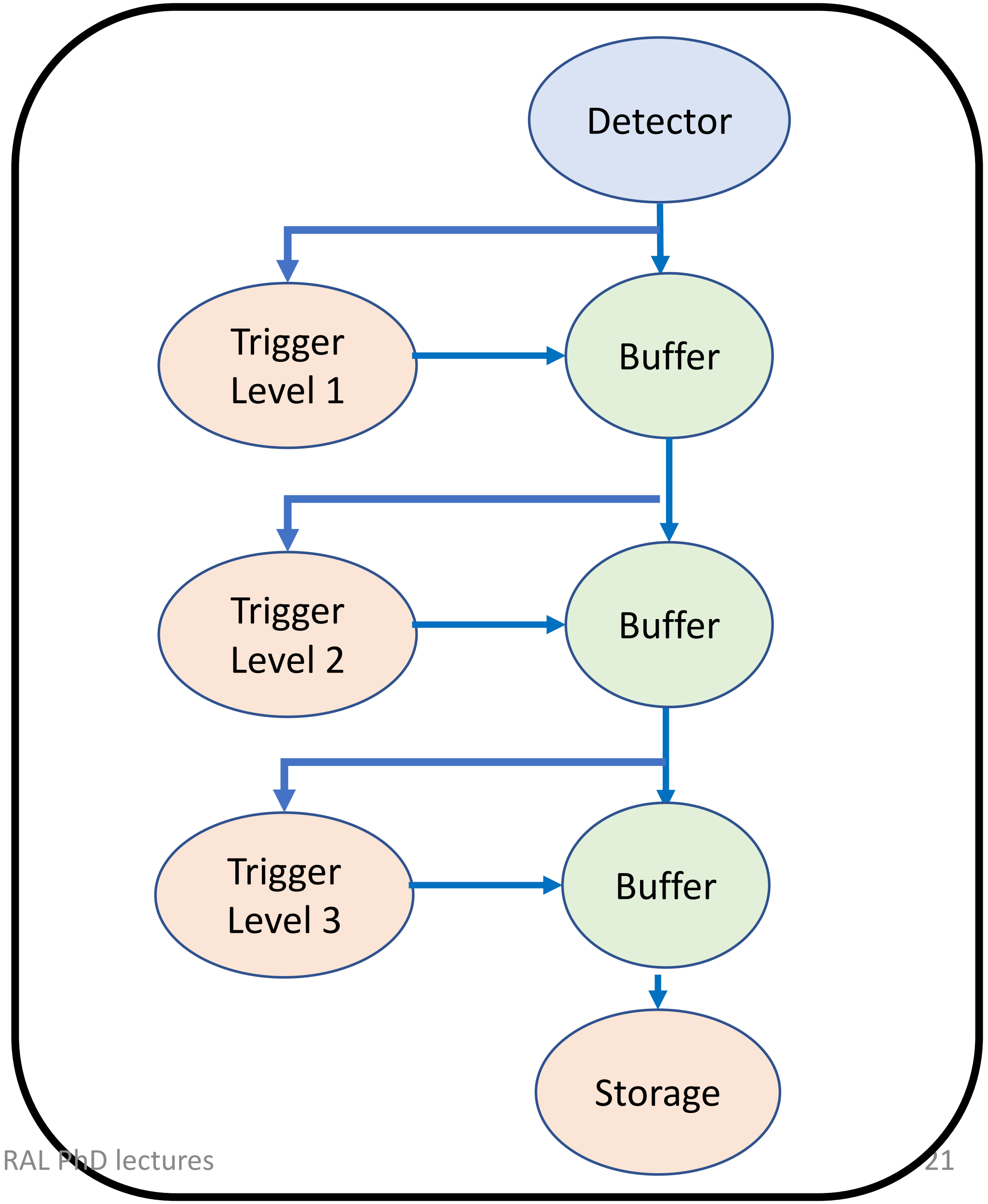
Larger buffers
Longer latency
Lower event rate
Larger event fragment size
Higher algorithmic complexity
Access to higher granularity information

Multi-level trigger

May not be possible to take trigger decision in a single place:

- Too many readout units
- Too far away (cables)
- Too long decision time

Distribute decision over a number of steps e.g. reject ~90% of events at each step



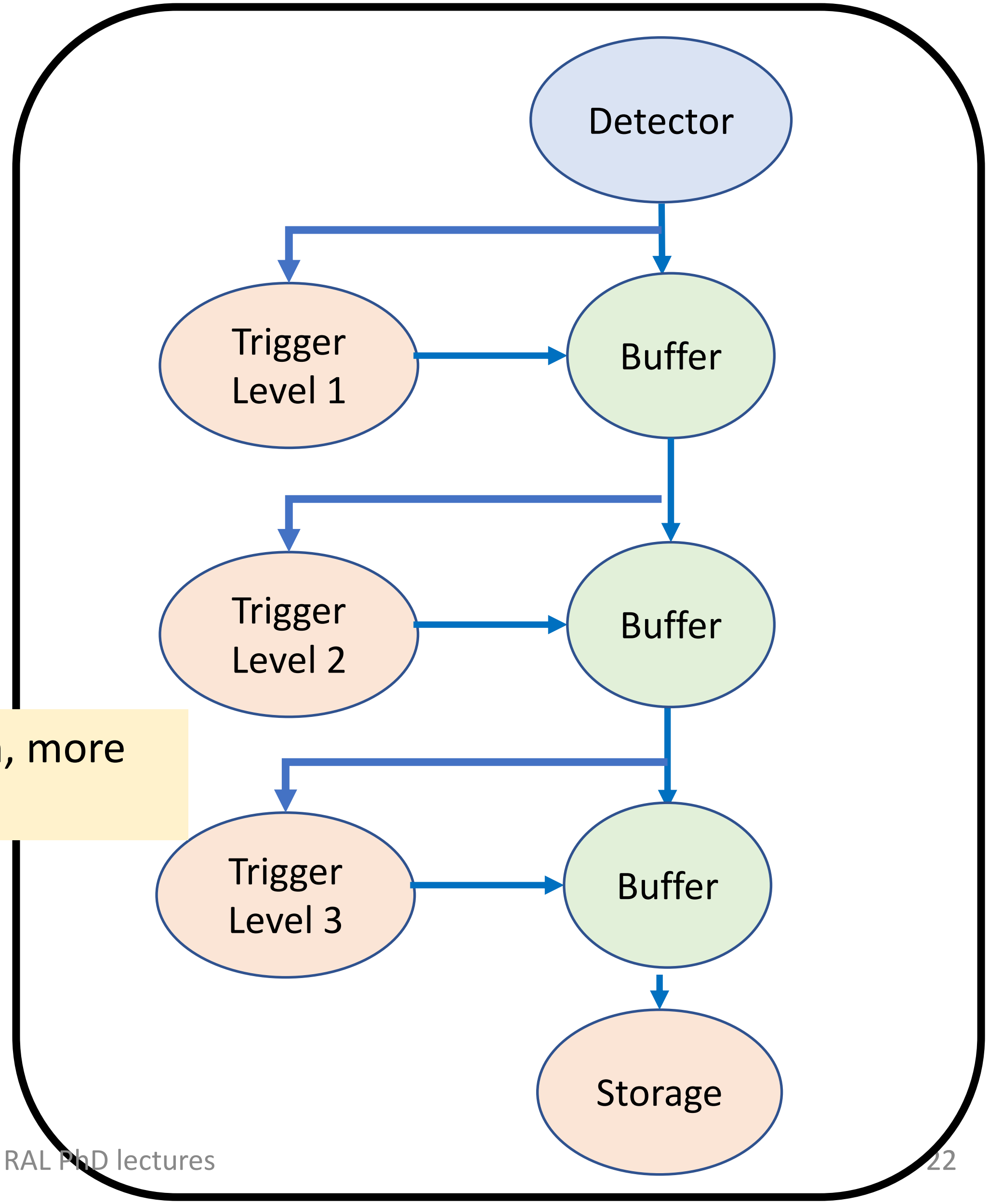
Multi-level trigger

May not be possible to take trigger decision in a single place:

- ➢ Too many readout units
- ➢ Too far away (cables)
- ➢ Too long decision time

Distribute decision over a number of steps
e.g. reject ~90% of events at each step

Longer latency, higher granularity information, more complex algorithms

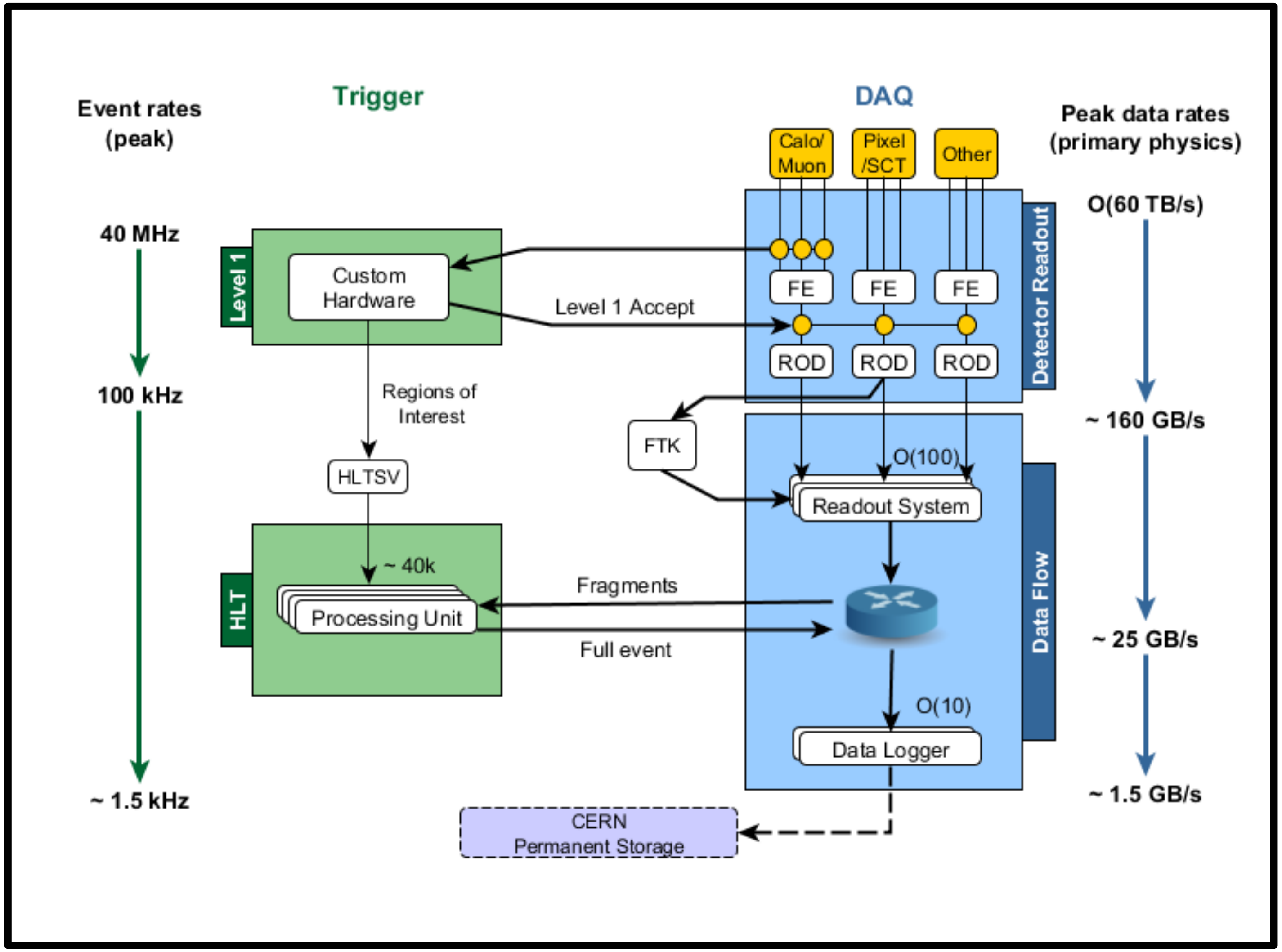


Multi-level trigger

May not be possible to take trigger decision in a single place:

- Too many readout units
- Too far away (cables)
- Too long decision time

Distribute decision over a number of steps e.g. reject ~90% of events at each step

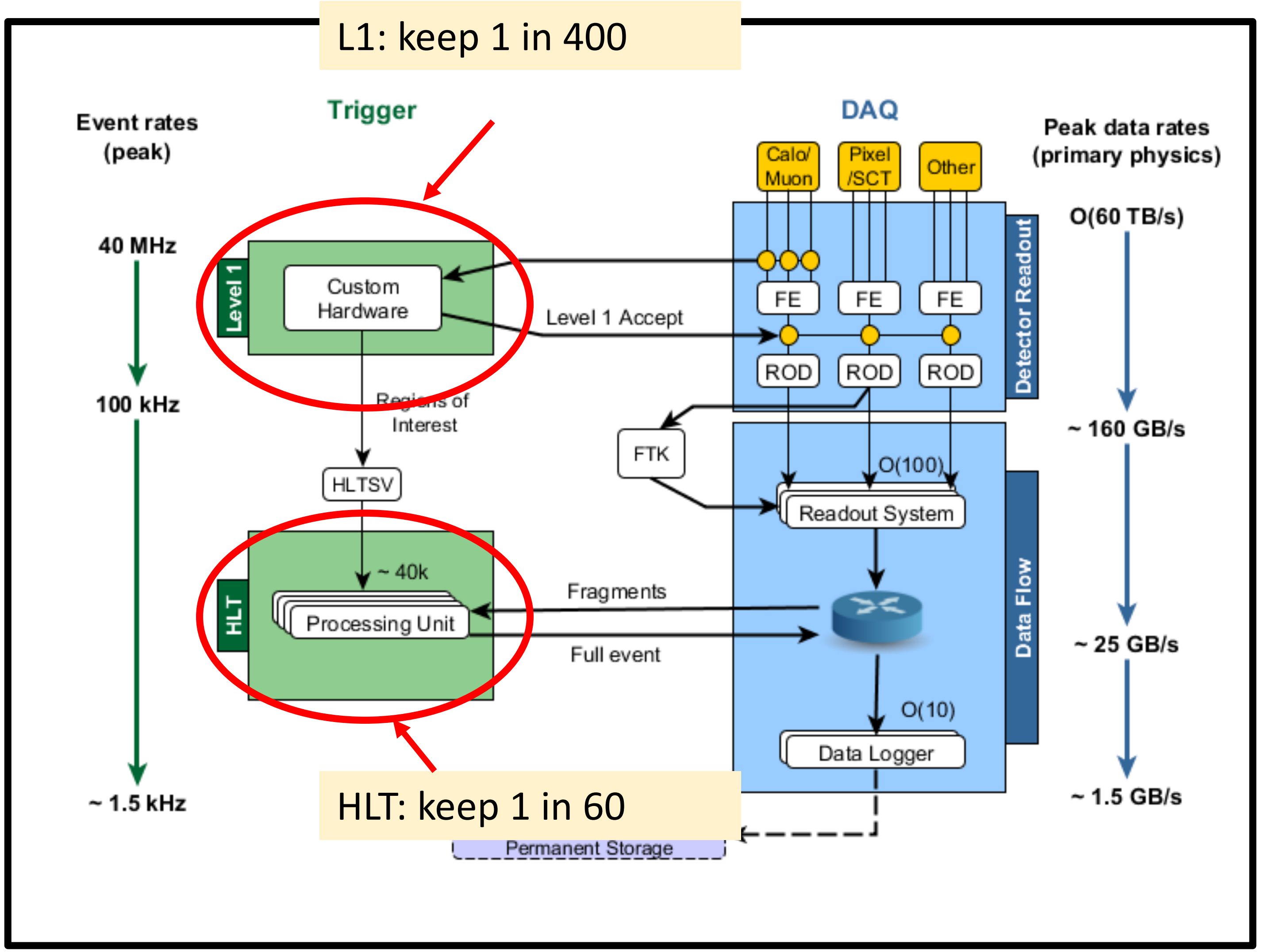


Multi-level trigger

May not be possible to take trigger decision in a single place:

- ➢ Too many readout units
- ➢ Too far away (cables)
- ➢ Too long decision time

Distribute decision over a number of steps e.g. reject ~90% of events at each step

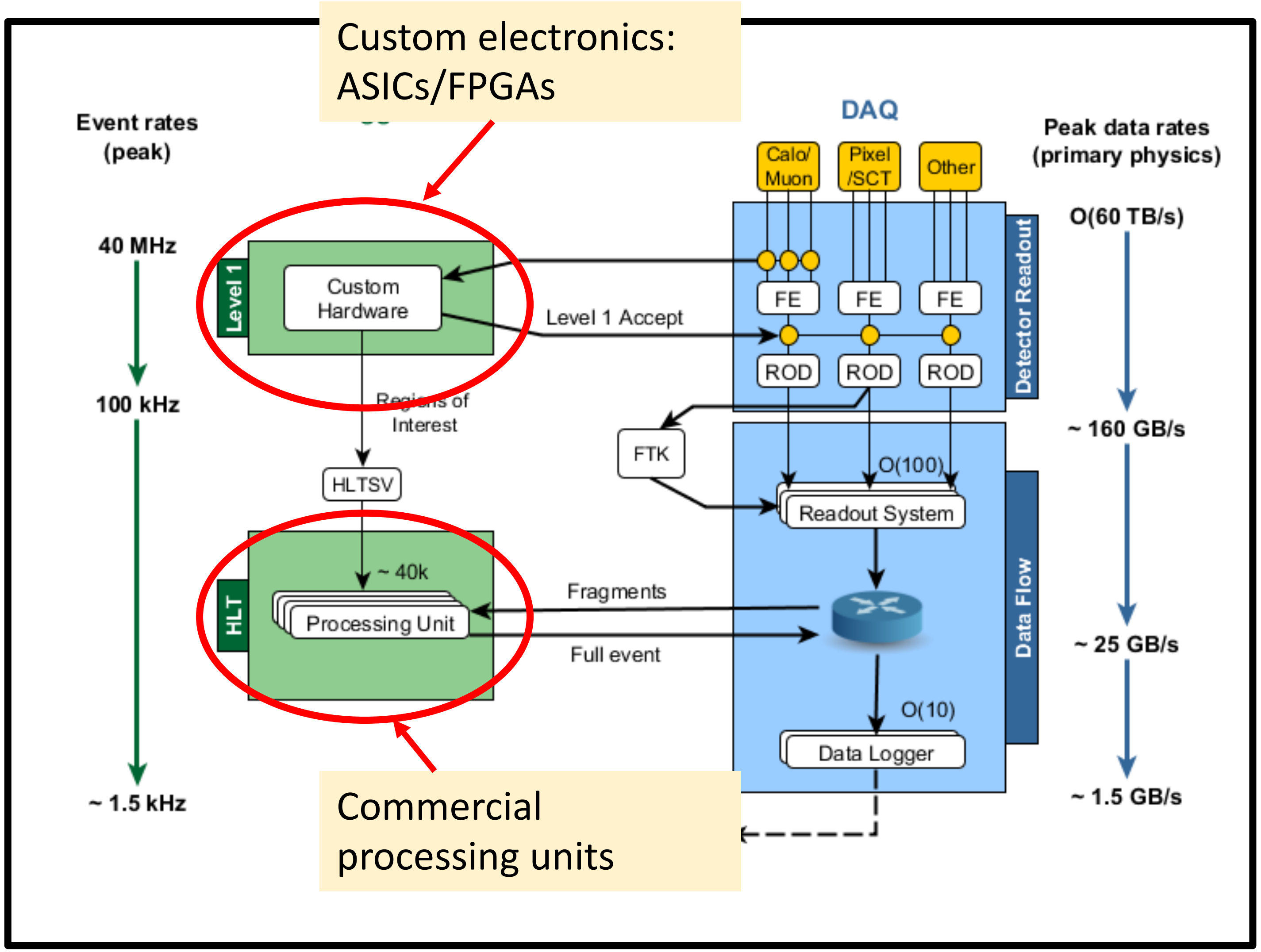


Multi-level trigger

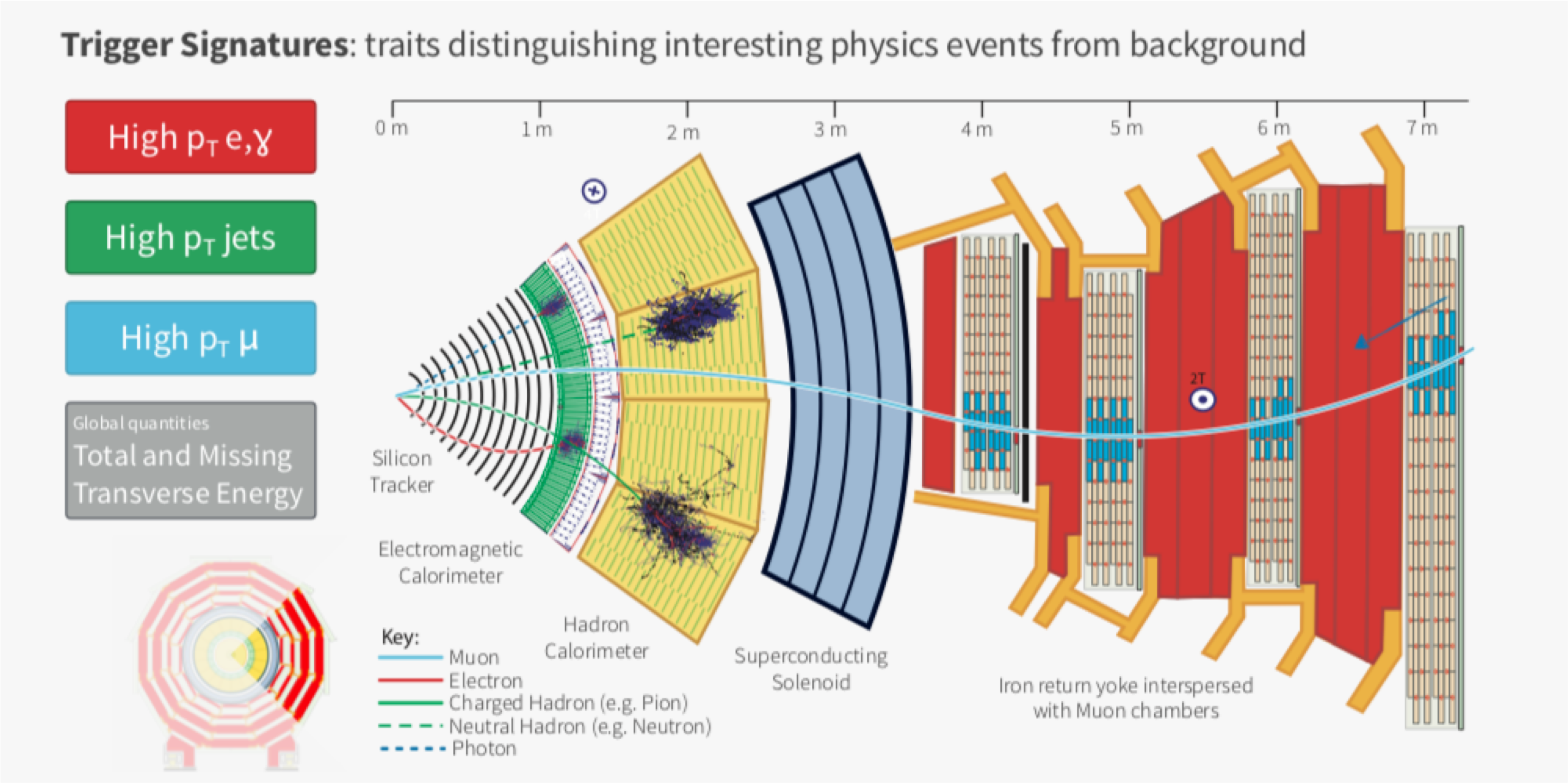
May not be possible to take trigger decision in a single place:

- Too many readout units
- Too far away (cables)
- Too long decision time

Distribute decision over a number of steps e.g. reject ~90% of events at each step



What is the trigger looking for??

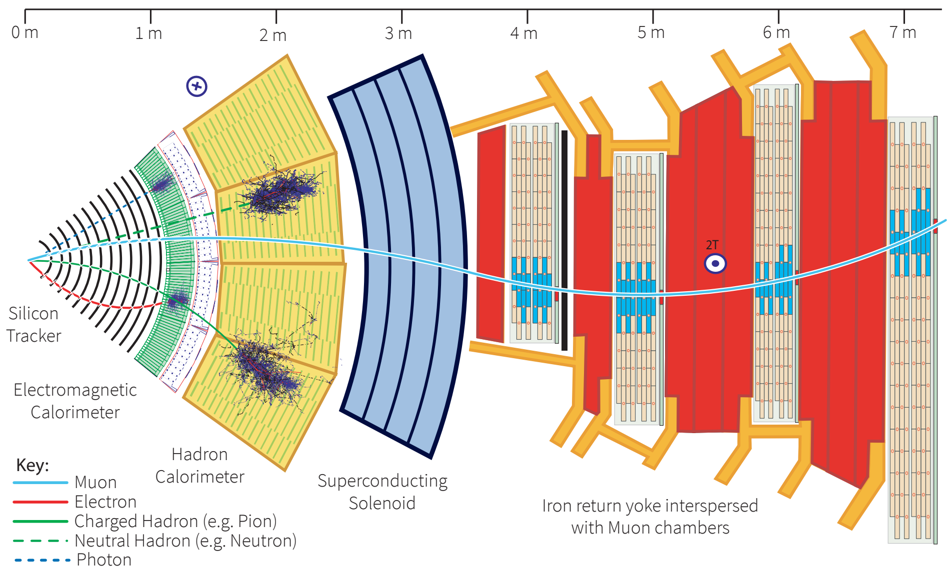


Trigger Algorithms

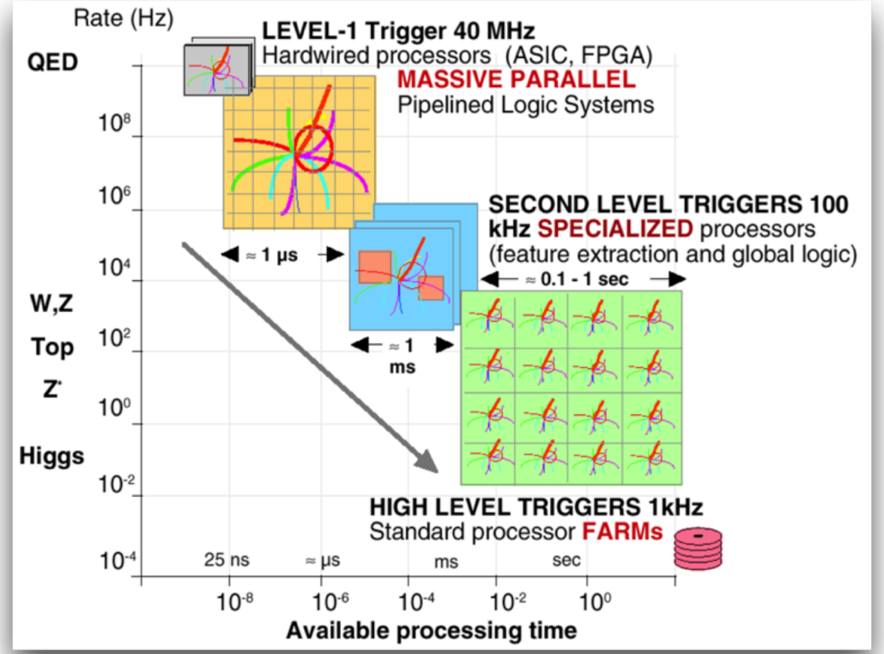
Trigger selection based on multiple trigger algorithms

- Exploiting reduced data from (sub)detector(s) to identify physics signatures
- Generally, several algorithms operate in parallel to find different signatures (trigger objects)
  e.g. calorimeter data used to find electrons + jets in parallel
- Algorithms must cover whole detector in an unbiased way
- The algorithm output is a counter or list of signatures, possibly with extra information
  - # of objects
  - pT, position, charge, 'quality', etc… for each object

Some algorithms are 'global'
- i.e. they encompass the whole detector
- Examples: Missing ET, Total ET, HT, global object counts



Multi-level trigger



Level-1
Inclusive trigger

Level-2
Confirm L1, inclusive and semi-incl., simple topology, vertex rec.

High Level Trigger
Confirm L2, more refined topology selection, close to offline

Reduction in rate
Increased complexity of algorithms

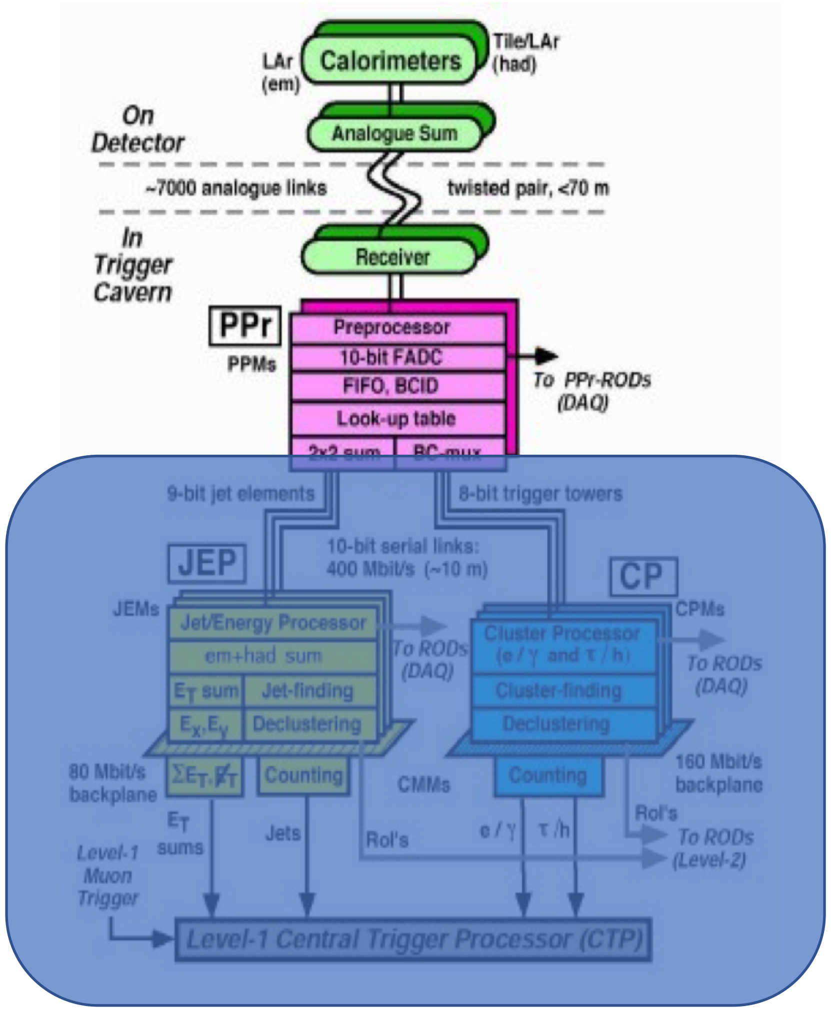
Simple example : single electron

L1 Electron: p_T > Thresh

Inner Detector track associated to L1 electron

Combined electron
- p_T > Thresh
- Isolation

L1 Example : ATLAS Calorimeter Trigger



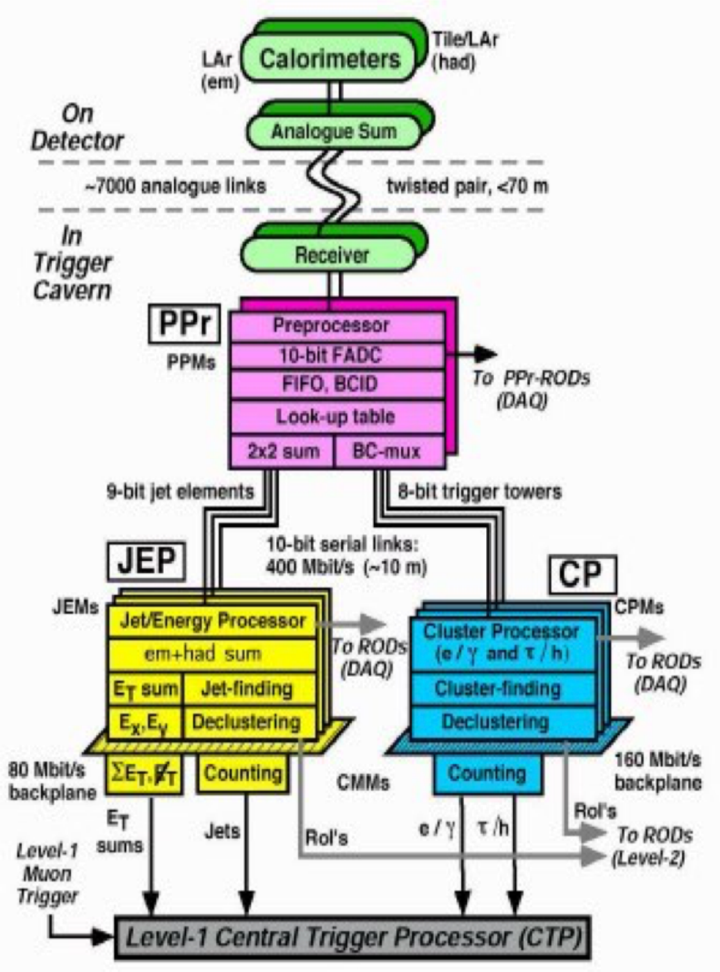
On-detector:

- Sum of analog signals from cells to form towers

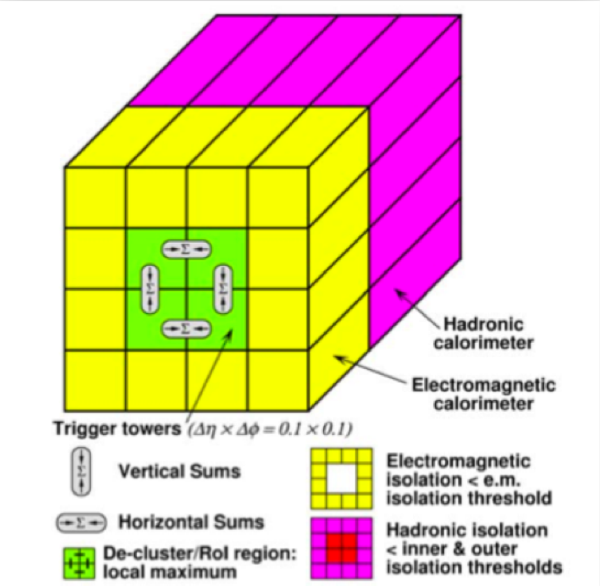
Off-detector - L1 Trigger

- Pre-processor board
- ADCs with 10-bit resolution
- ASICs to perform the trigger algorithm
  - Assign energy (ET) via Look-Up tables
  - Apply threshold on ET
  - Peak-finder algorithm to assign the BC

L1 Example : ATLAS Calorimeter Trigger



- Implemented in FPGAs, the parameters of the algorithms can be easily changed
- Total of 5000 digital links connect PPr to JEP and CP, 400 Mb/s



Electron/photon trigger

- Sum energy in calorimeter cells into Em and Had towers
- Search in 4x4 towers for local (1x2/2x1) maximum

Can do similar for Jets, Taus, missing E_T

Central/Global Trigger

L1 (Calo and Muon) use reduced granularity to provide fast (<3μs) information on particle candidates.
May only be Muon/Calo but still a lot of info

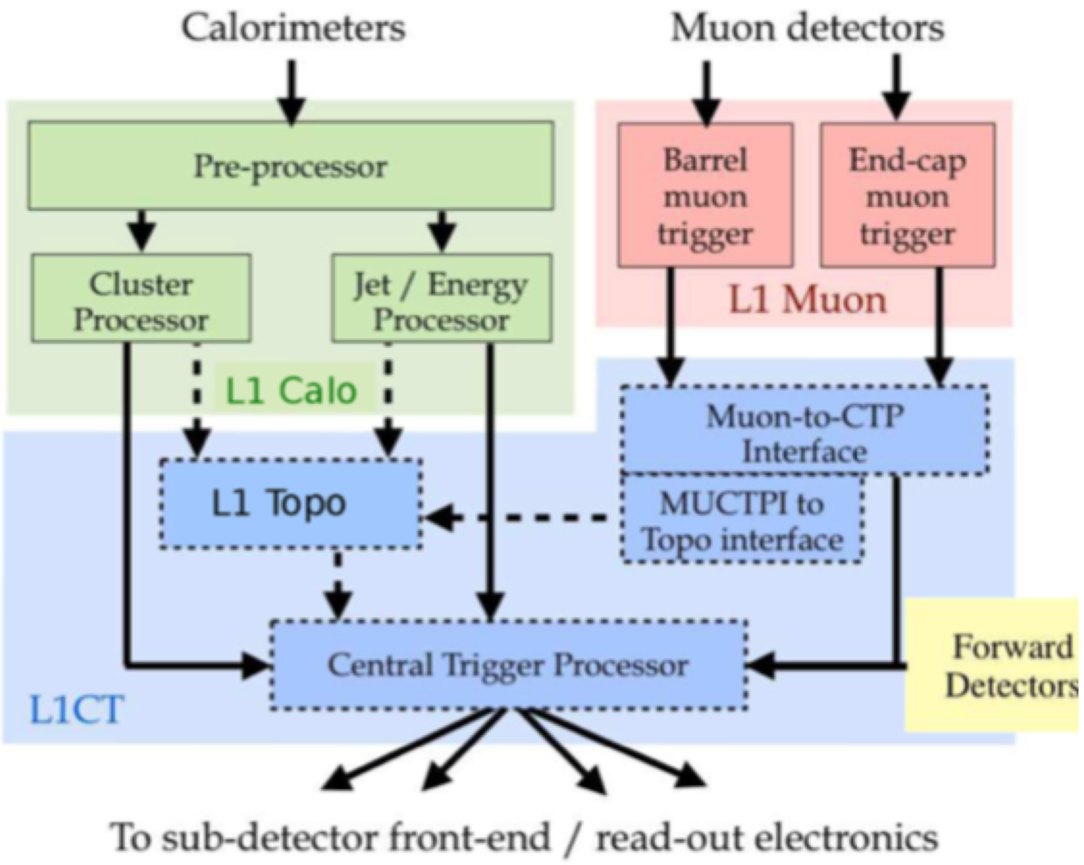
- Electrons, muons, taus, jets, total and missing energy
- location, E_T, p_T threshold passed

Can also look at topological constraints

- More complex checks: $\Delta\varphi, M_{JJ}, \Delta R$

Central trigger decides pass/fail

If pass, collate data from whole detector and send to **High Level Trigger** (rate ~100kHz)

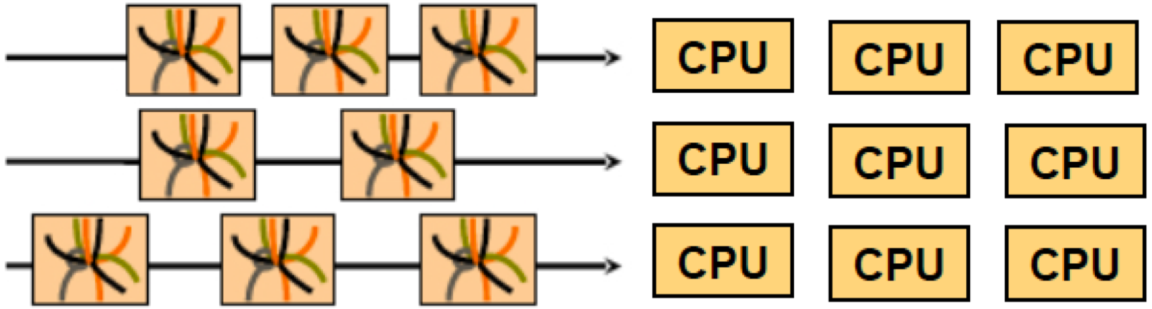


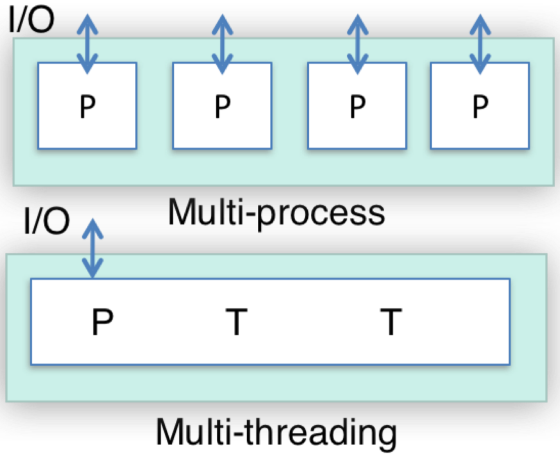
High Level Trigger: software trigger

Still need to reduce rate for storage:
- ➢ 100 kHz → 1kHz

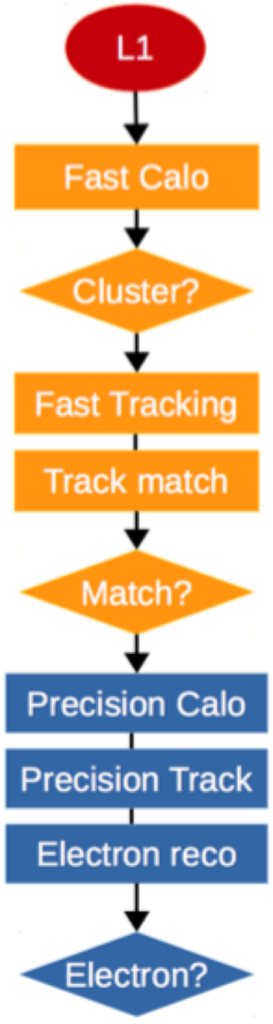
Networked computer farm

Early rejection:
    Reduce data and resources (CPU, memory….)
Event-level parallelism:
    Process more events in parallel
    Multi-processing or/and multi-threading





Example: HLT trigger algorithm



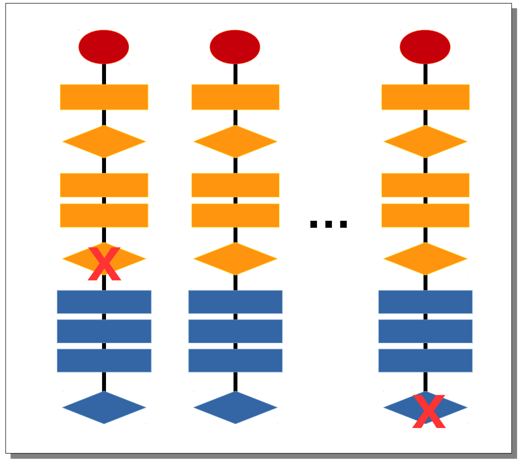
Offline reconstruction too slow to be used directly

- Takes >10s per event but HLT usually needs << 1s L1

Requires step-wise processing with early rejection

- Fast reconstruction
  - Trigger-specific algorithms
  - L1-guided regional reconstruction
- Precision reconstruction
  - Offline (or very close to) algorithms
  - Full detector data available

Stop processing as soon as one step fails

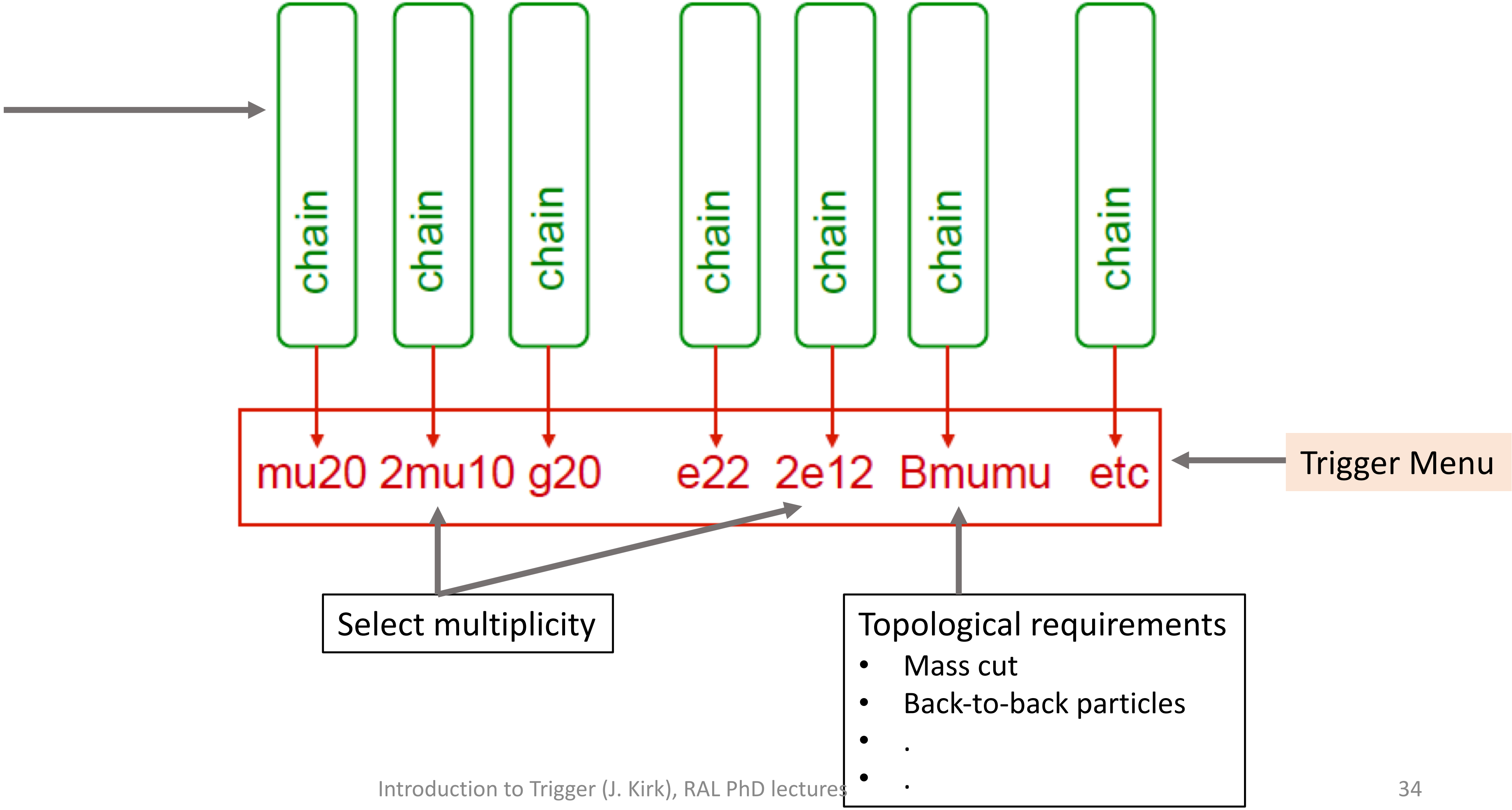


Trigger chain

Trigger menu

Chains identify different particles:

- Electron
- Muon
- Photon
- Tau
- Jets
- B-jets
- Missing E_T
- .
- .
- .



Trigger menu

Trigger menu:

- Define the physics programme of the experiment, i.e. what is recorded
- Each physics group defines a set of chains
- Trigger menu

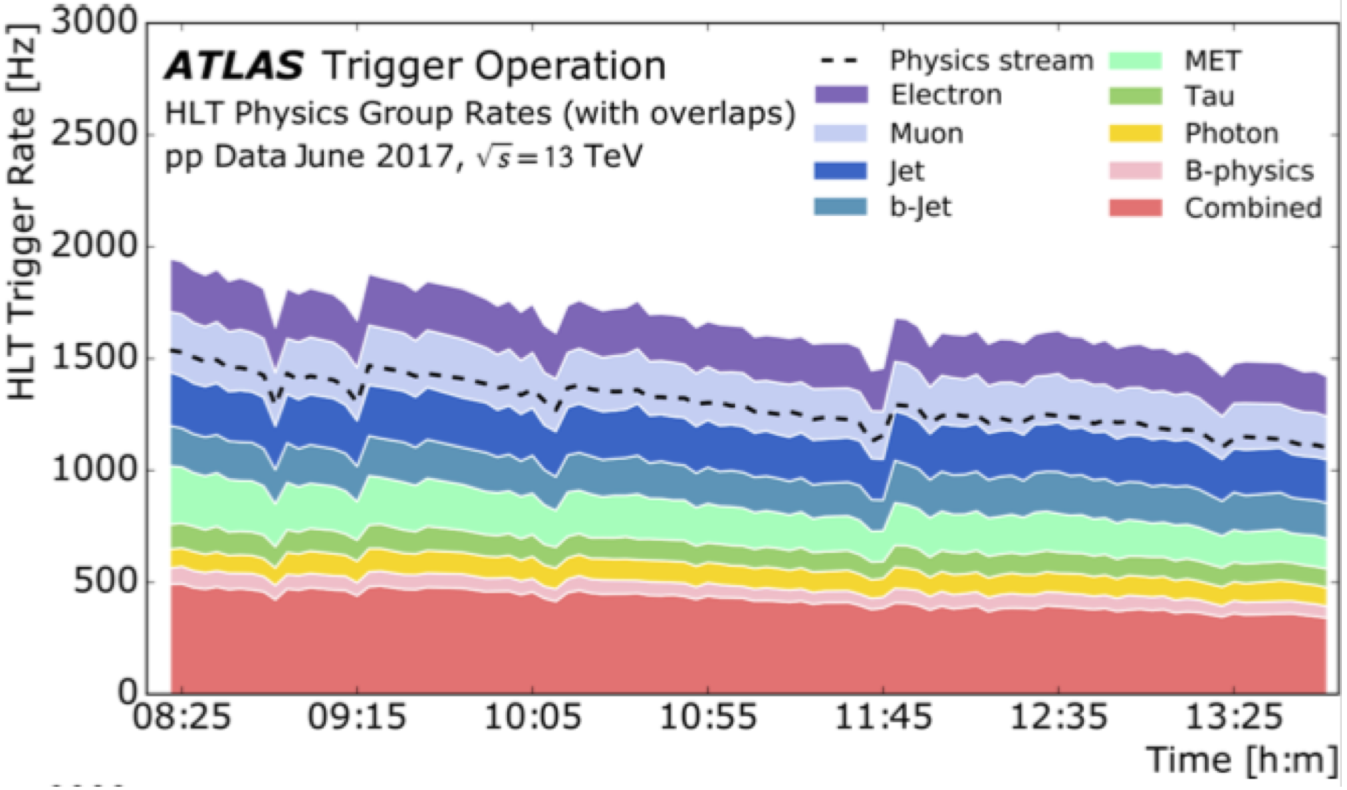
Event is recorded if at least one chain passes.

Menu design is driven by:

- Physics
- Rate limitations at L1/HLT
- Online resources (CPU/bandwidth)

Menu consists of:

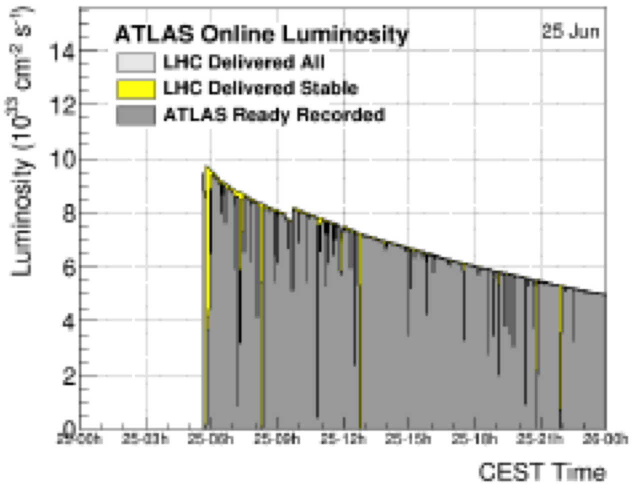
- Primary physics triggers
- Support triggers – measure efficiencies and backgrounds
- Calibration triggers – needed by detector groups (e.g. L1 only triggers)
- Monitoring triggers – check everything is working (e.g. $Z \to ll$)



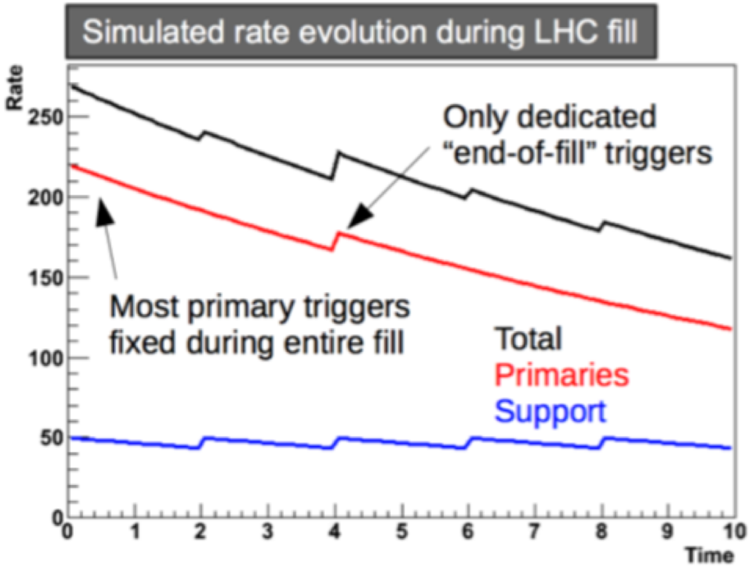
Trigger prescales

Menu varies with luminosity, time and running conditions

Not all triggers run at full rate:
- Rate might be too high
- Sub-sample may be enough to fulfil needs (support triggers)
- Add triggers as luminosity drops – 'optimal' use of resources



Trigger prescale reduces rate:
- Prescale N means accept '1 in N' events passing this trigger
- Prescale can be fractional
- Apply L1 or HLT prescales
- Can change during run – lower prescales as luminosity drops, add in 'end-of'fill' triggers



Designing a menu

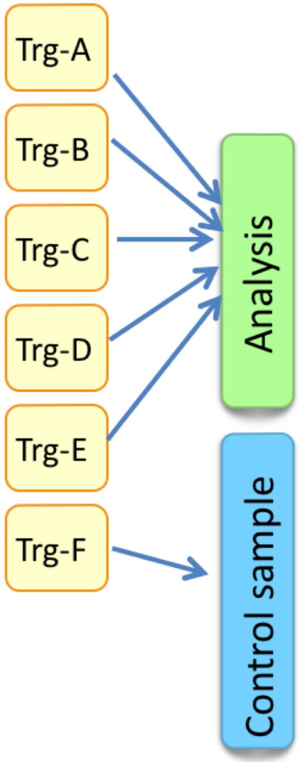
Defines the physics programme/reach of the experiment
Collection of physics trigger, associated back-ups, triggers for calibration and monitoring

It must be

- Redundant - to ensure the efficiency measurement
- Flexible - to adapt to changes of the environment and the physics goals, e.g. detectors, machine luminosity,…

Central to the physics programme

- 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
  - provide physics breadth and control samples



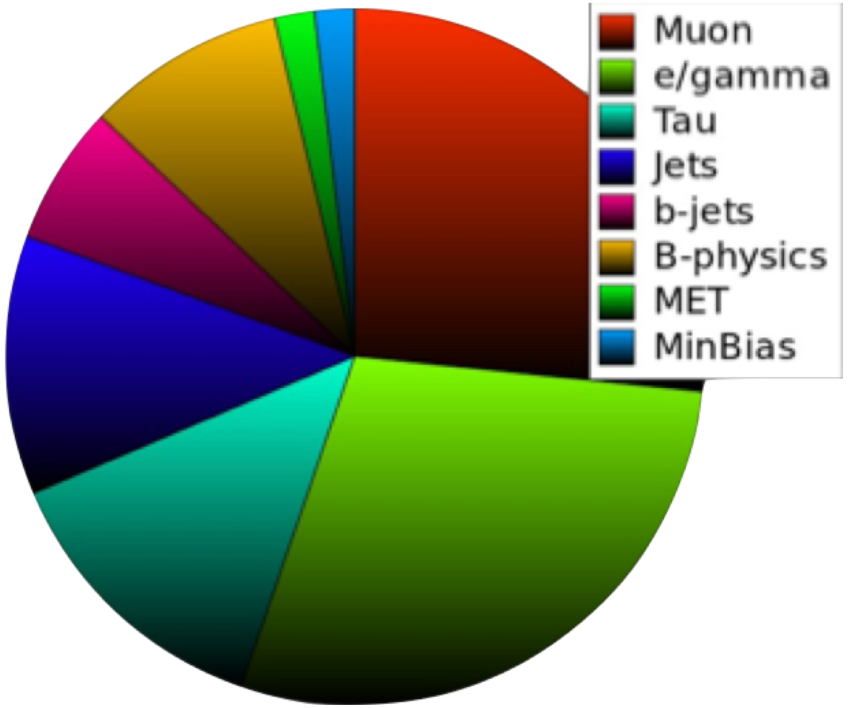
Rate Allocation : “Physics versus Bandwidth”

Lower thresholds always desirable, but the physics coverage must be balanced against offline computing cost

Target : the final available DAQ bandwidth

The rate allocation to each trigger signature
- Physics goals (plus calibration, monitoring samples)
- Required efficiency and background rejection
- Bandwidth consumption

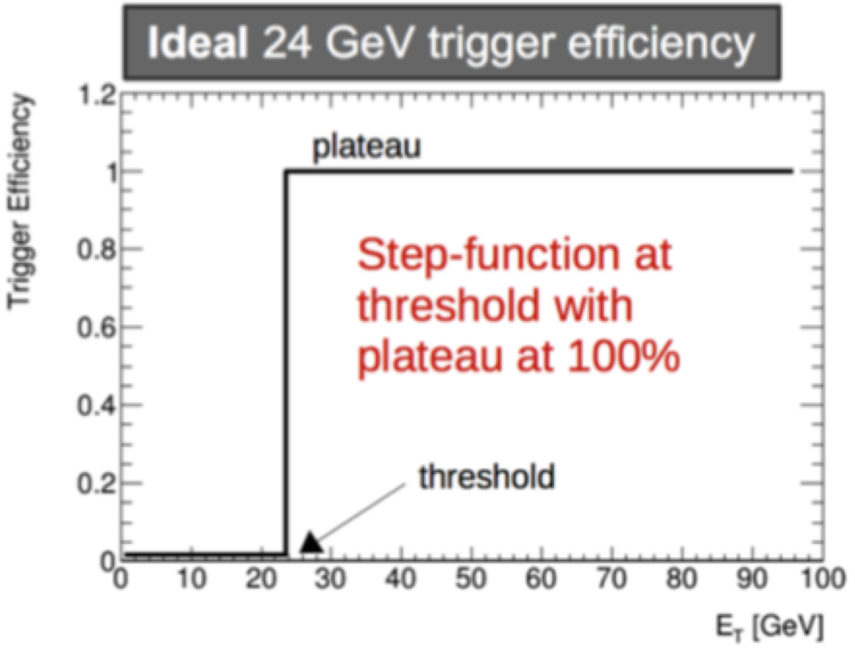
When designing the menu: check predicted rates using previously recorded unbiased data



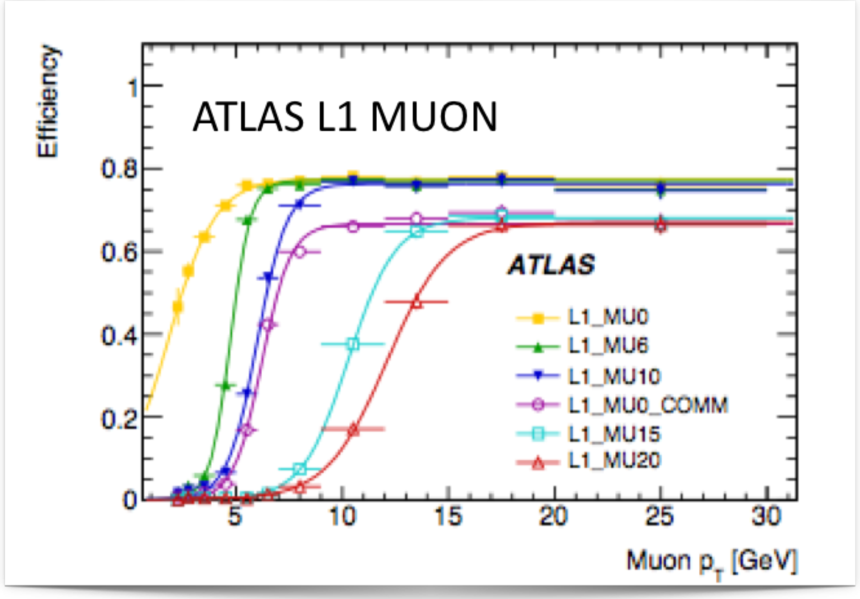
Physics analysis perspective

Physics analyser wants to know:

- Where is the trigger **turn-on**?(maximal efficiency wrt offline objects)
- What is the peak **efficiency**? (is it 100%, or do I need a scale factor)
- Is it **prescaled**? (Do I need a correction?)



- Resolutions
- Inefficiencies
- Online/offline differences

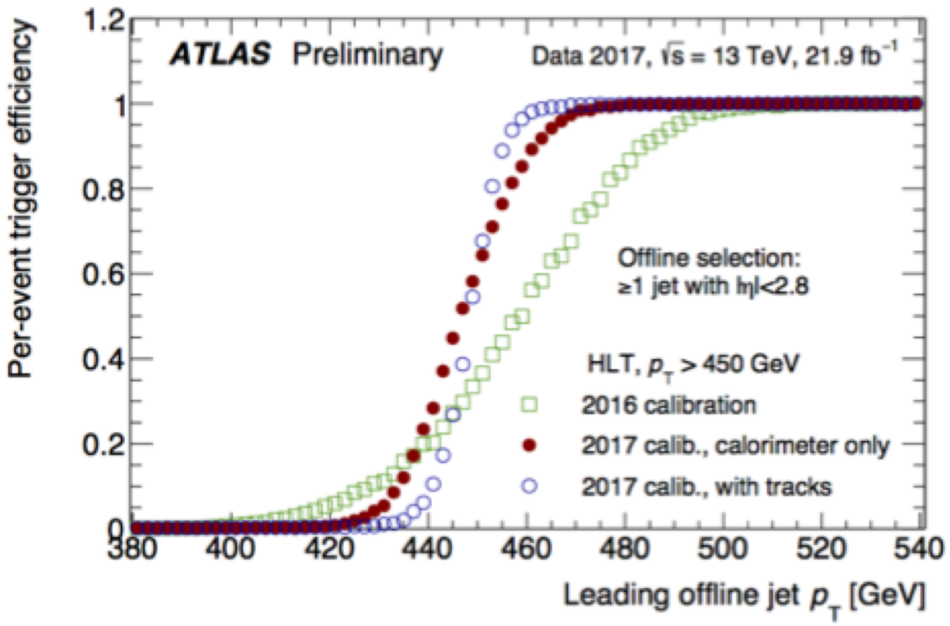


Measuring efficiency

How to measure the efficiency of your trigger?

Define efficiency w.r.t the offline reconstructed objects

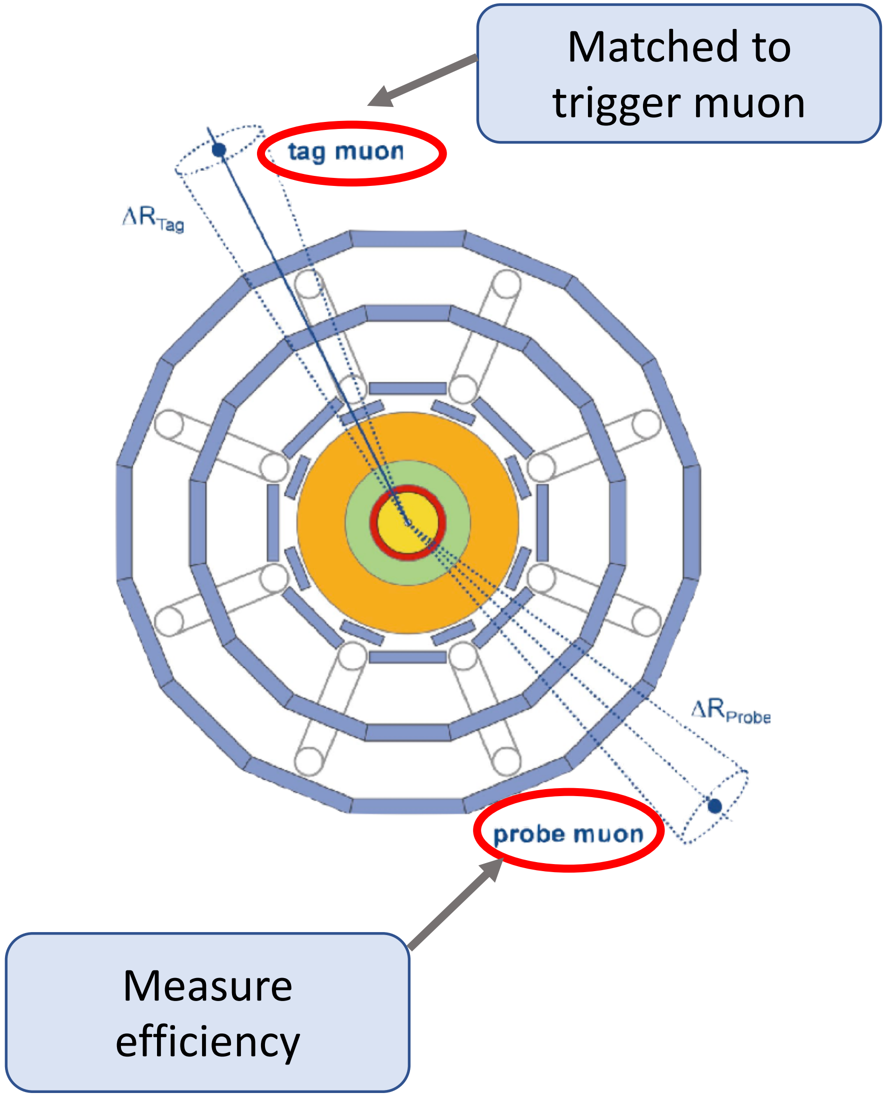
$$\epsilon_{trigger} = \frac{N_{trigger}}{N_{offline}}$$



Various methods:

- Tag-and-probe
  - Trigger on one particle (tag), measure how often another (probe) passes trigger, e.g. $Z \rightarrow ll, \;\; J/\psi \rightarrow ll$
- Boot-strap
  - Use sample triggered by looser (prescaled) trigger to measure efficiency of higher threshold trigger
- Orthogonal trigger
  - Use sample triggered by one trigger (e.g. muon trigger) to measure efficiency of a different, independent trigger (e.g. jet trigger)
- Simulation/emulation : MC

Measuring efficiency : tag-and-probe



Exploit a well know physics process (e.g $Z \rightarrow ll,\ \ J/\psi \rightarrow ll$) to select a very clean sample

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

How?
Online:
- Trigger on independent signature (e.g. single muon)

Offline:
- Reconstruct the event, e.g. 2 muons in Z mass window (use tight selection for high purity)
- Match offline muon to trigger muon (Tag)
- Measure trigger efficiency for other (Probe) muon

Efficiency as a function of:

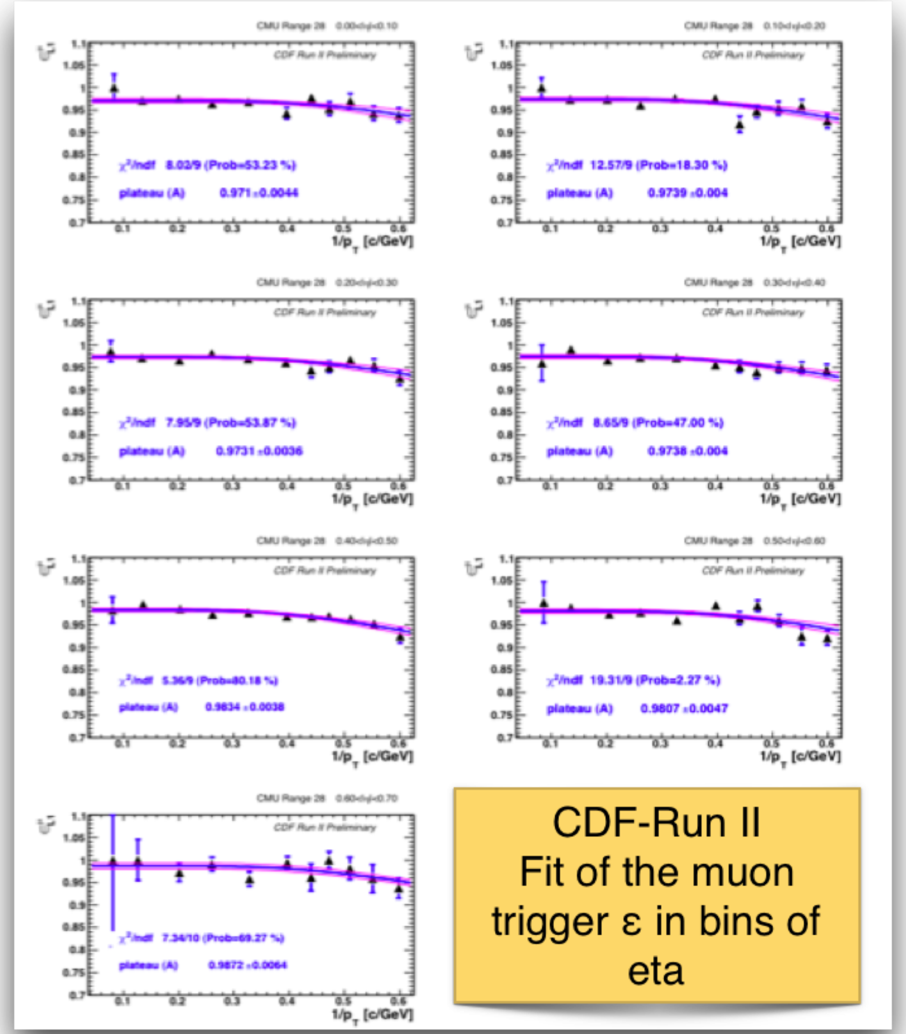
Trigger efficiency can vary rapidly due to changes in

- Detector geometry (η, φ)
- Trigger hardware (ageing, dead channels, etc…)
- Trigger definition
- Trigger algorithms

Analyses must track all of these changes

Multi-dimensional study of the efficiency:
$\varepsilon(p_T, \eta, \varphi, run\#)$

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



Monte Carlo and Scale Factors

Triggers have to be emulated in simulated data (Monte Carlo)

BUT… MC samples are produced **before** the data is recorded

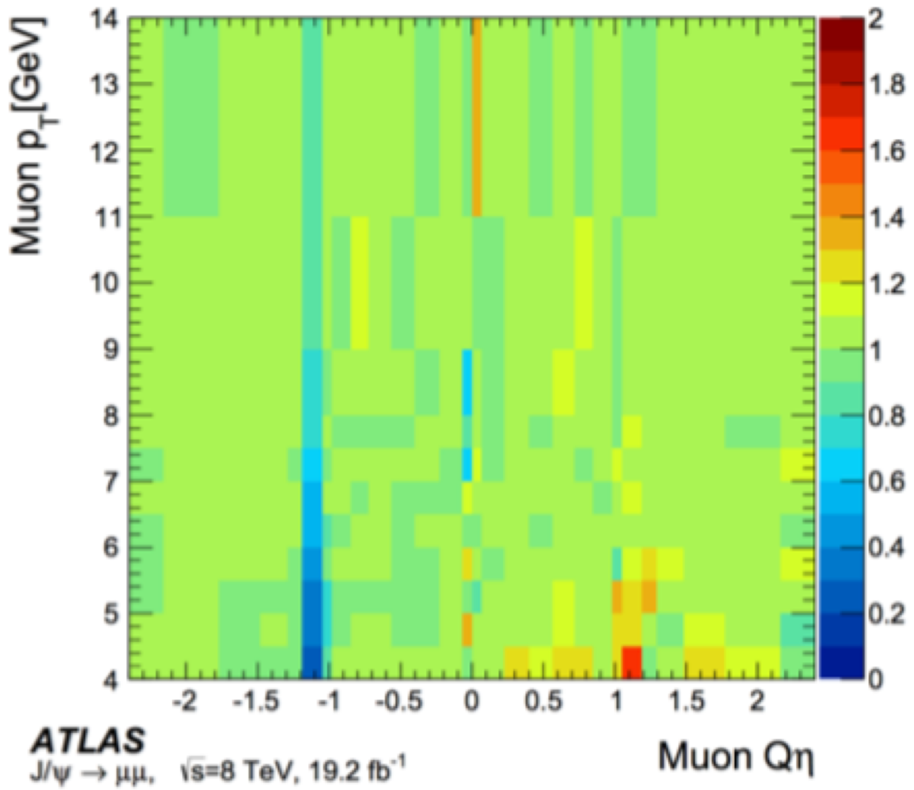
MC contains best-guess trigger menu (plus backups for possible future triggers). **Never** think of everything

Differences between data/MC occur due to:

- ➢ Different running conditions: pileup, luminosity profile
- ➢ Trigger menu changes
- ➢ Improvements/ bug fixes

Scale factors used:

- ➢ Correct MC to match observed data
- ➢ Paramaterise in terms of p_T, η, ϕ, etc.



How to design a trigger

Understand your physics:
- What particles are in final state, how high is pT?

Understand the existing trigger menu
- Is there a trigger in place that will accept your events? → It's covered – job done!!

If not think up a new trigger:
- Can you combine particles, e.g. muon + 2 b-jets
- Can you use topology of event, .e.g. invariant mass, back-to-back topologies
- Remember trigger selection should be looser than offline

Keep it simple
- Less bias
- Less need for supporting triggers

Would other analyses profit from your trigger?
- More analyses there are the more likely your trigger will be accepted to run online

If possible, base it on an already existing trigger
- Already validated

Check the rate
- Will this new trigger fit into the trigger menu or do you need a prescale?

How to design a trigger

Simple

- Easy to commission, debug and understand

Inclusive

- One trigger for many analyses
- Discover the unexpected!

Robust

- Trigger runs millions of times per second – strange conditions will occur, be prepared
- Be immune to detector problems
- Be prepared for changes in beam conditions

Redundant

- Signal selected by more than one trigger
- Help understand biases and efficiencies
- Safety – backup in case of high rates or problems

Don’t forget supporting triggers for efficiency measurement and background studies

Be creative

Trigger Level Analysis

Search analyses don't tend to like prescaled triggers:

- Immediate efficiency loss at trigger level
- Signal events could be lost

Prescales used to keep rates under control

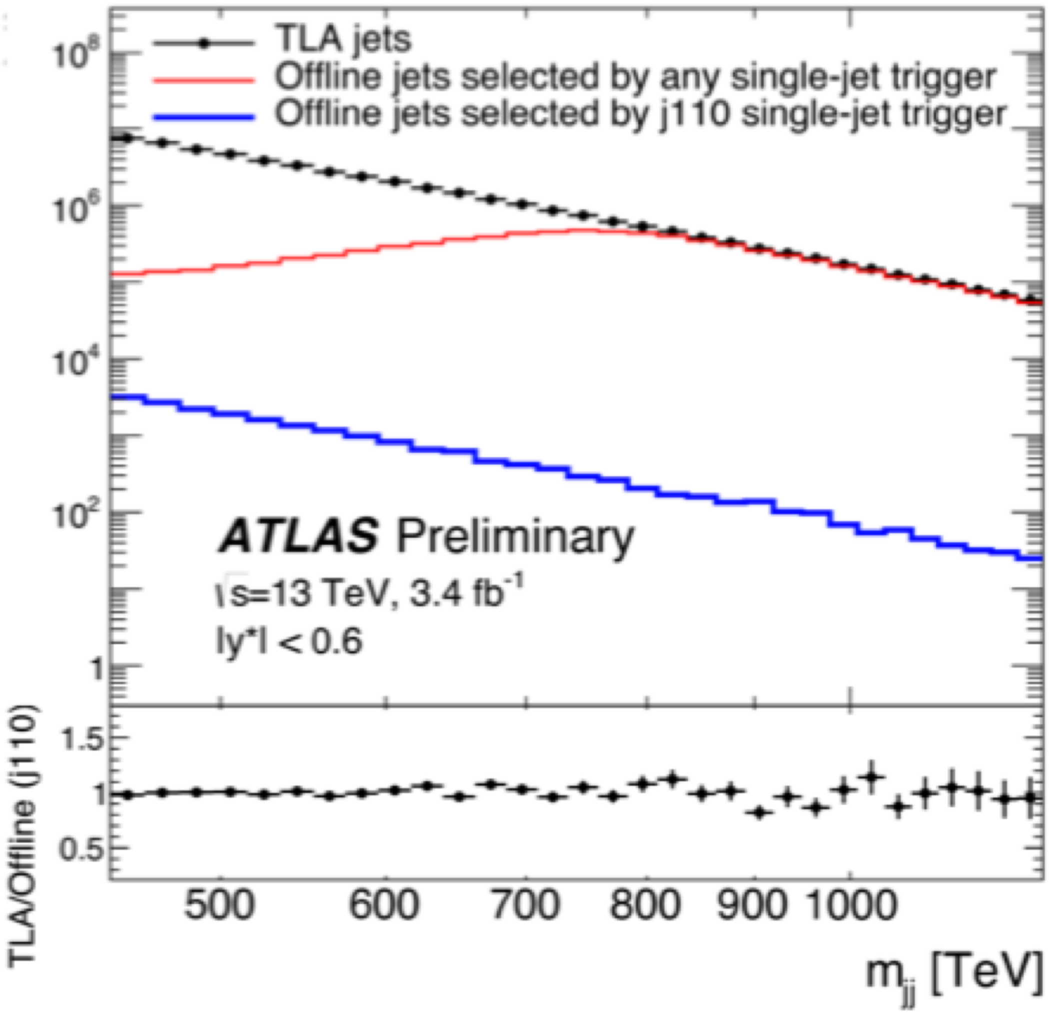
Have another dial to turn: event size

Reduce the size of your event by only saving objects you need

- Run unprescaled again!!

TLA jets:

- Only save the leading few HLT trigger jets with selective variables
- Form di-jet invariant mass and push below threshold allowed for normal jet triggers

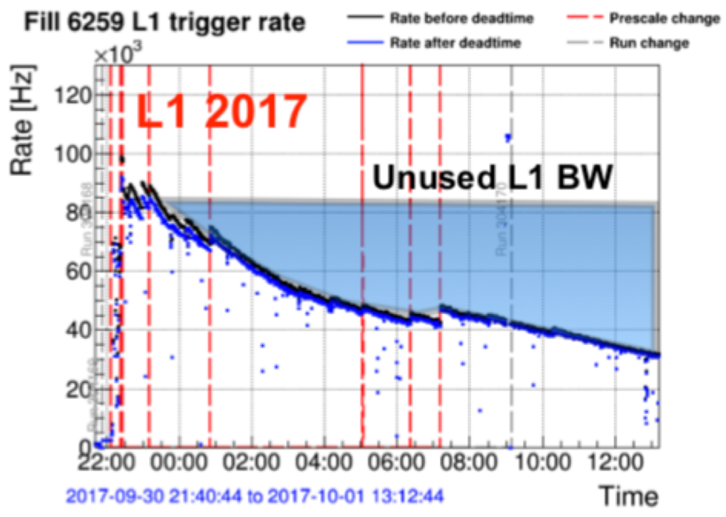


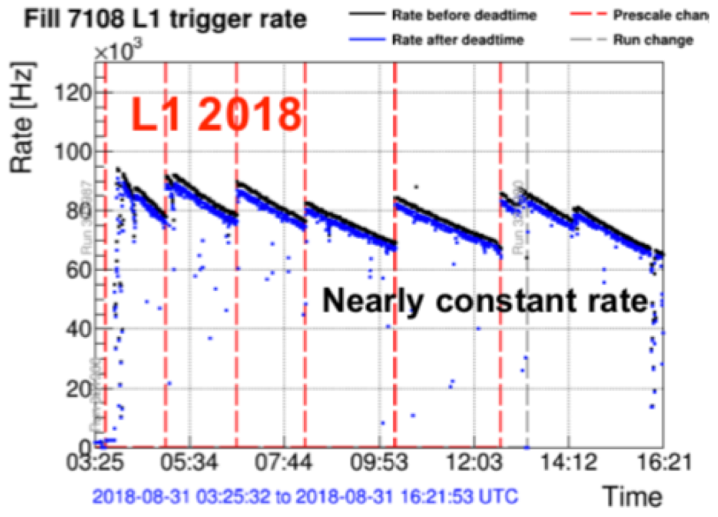
Parked / delayed data streams

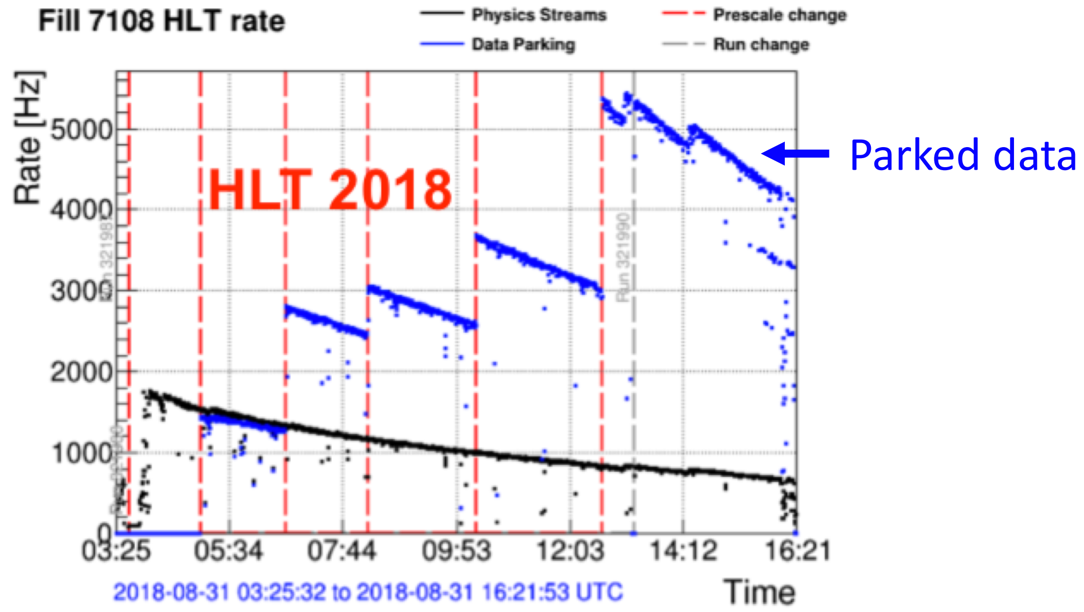
What if we don't mind waiting for the data?

B-physics analyses at CMS/ATLAS
- Often take a while for analyses to be completed (manpower, complicated analysis)
- Can afford to wait O(6 months) after data taking for the data
- During the run "park" the data – reconstruct when computer power is available (between fills or end of year)

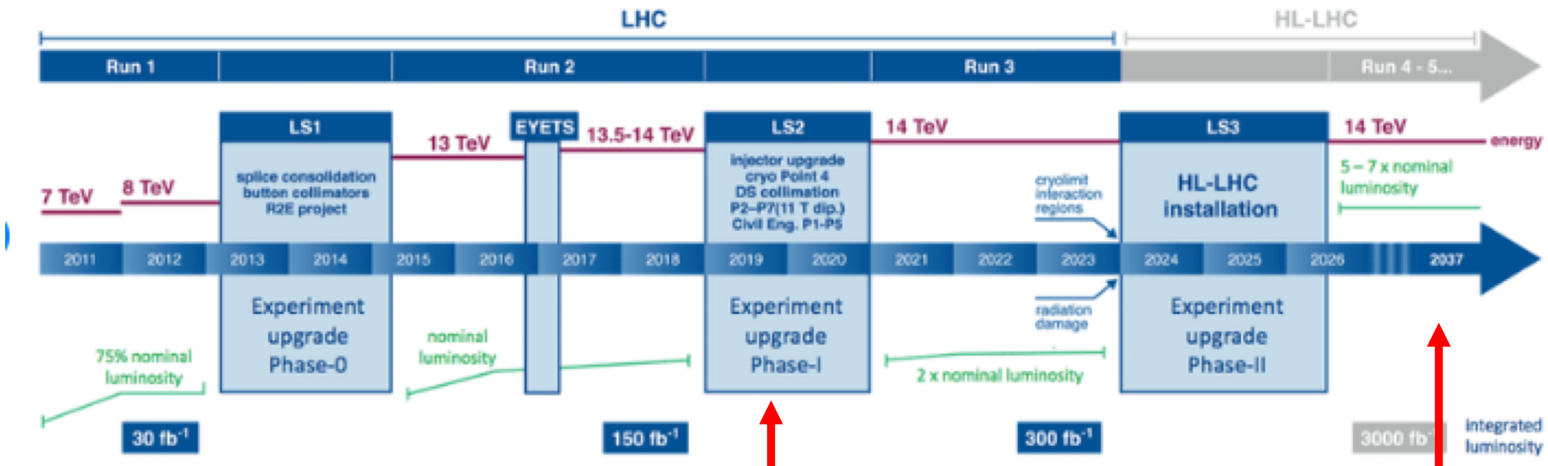
CMS recorded 10^{10} unbiased B hadron decays in 2018







Future



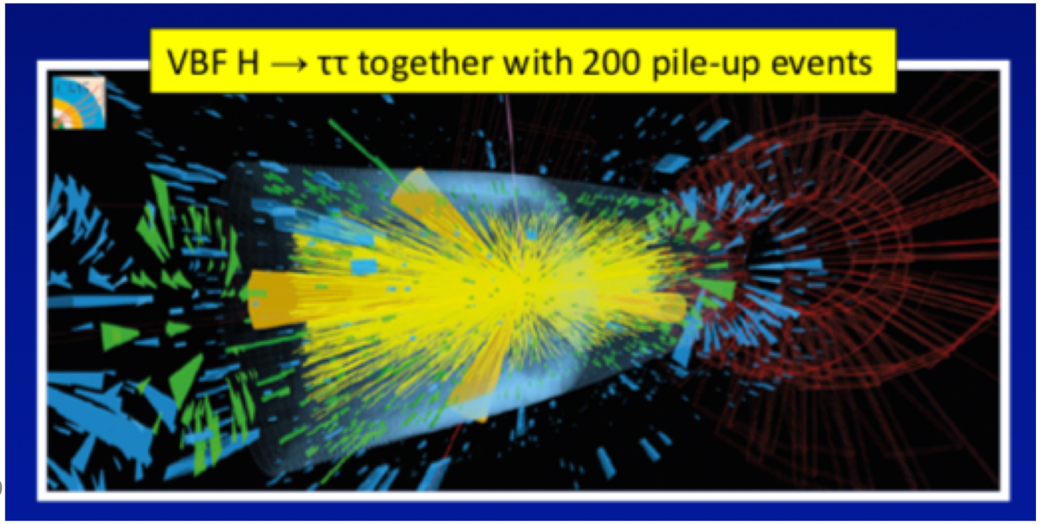
We are here

Pileup ~200

Trigger driven by physics needs and accelerator environment

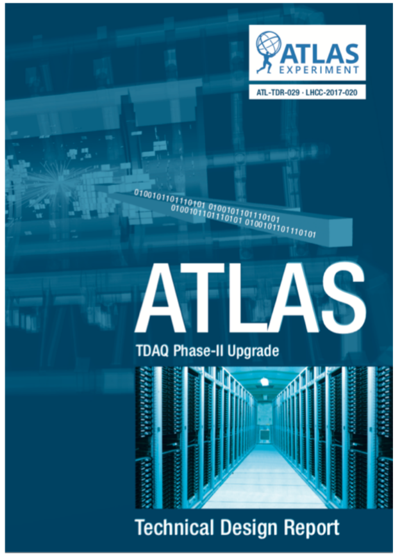
Future HL-LHC:

- ~200 interactions per bunch crossing (pileup)
- High granularity detectors (more channels)
- Larger event size



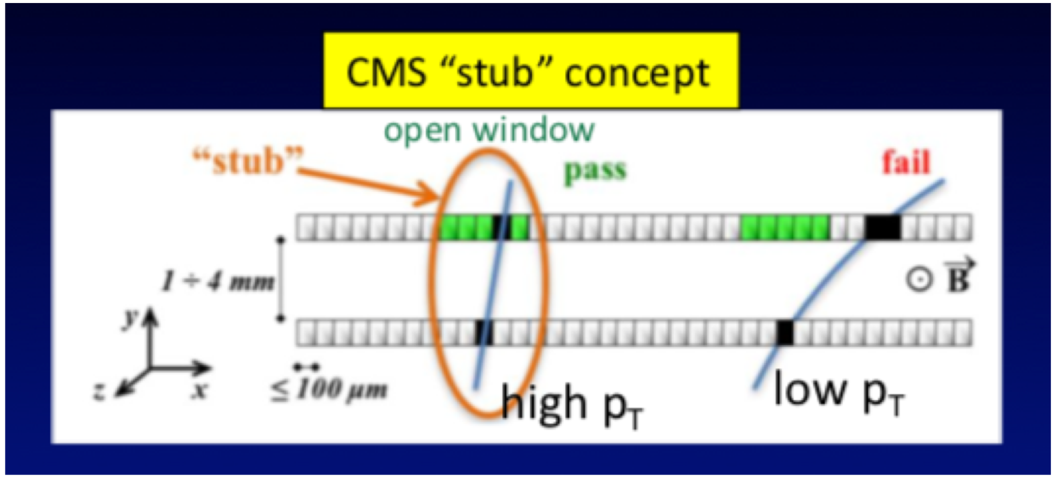
Future

BUT physics need to keep trigger thresholds close to today's values
Already planning for this…..
(And some already started)



- L1 latency increases to ~10-12.5μs (~2.5-3.2 μs today)
- Readout rate increases 750-1000 kHz (100 kHz today)
- Rate to storage ~7.5-10 kHz (~1 kHZ today)

- L0/L1 upgrade to use higher granularity, more complex algorithms
- Early use of tracking
- Use of FPGAs, GPUs, Machine Learning and multi-threading at HLT



Summary

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

Introduced some concepts and nomenclature – hopefully useful for your work

Main trigger requirements:

- High efficiency but with control of rates
- Knowledge of effect of trigger selection on signal and background events
- Flexibility and redundancy

Trigger is vital - if don't trigger an event it is lost forever!