



Science and
Technology
Facilities Council

Introduction to Data Acquisition

Will Panduro Vazquez, RAL PPD

*Advanced Graduate Lectures on practical Tools,
Applications and Techniques in HEP*
18th June 2026

Acknowledgements and Disclaimer

- Lecture and much of the material inherited from Alessandro Thea (now CERN, but latterly of this parish) – many thanks!
- I work on ATLAS – much of this talk will be collider-based, but the concepts are general

Outline

- *Introduction*
 - What is DAQ?
 - Example System architecture
- *Basic DAQ concepts*
 - Digitization, Latency
 - Deadtime, Busy, Backpressure
 - De-randomization
- *Scaling up*
 - Readout and Event Building
 - Buses vs Network
- *DAQ challenges at the LHC and beyond*

What is DAQ?

- **Data Acquisition (DAQ)** is
 - the process of **sampling signals** that **measure** real world physical conditions and **converting** the resulting samples into **digital** numeric values that can be manipulated by a computer
- **Ingredients:**
 - **Sensors**: convert physical quantities to electrical signals
 - **Analog-to-digital converters**: convert conditioned sensor signals to digital values
 - **Processing** and **storage** elements
- **In practice:**
 - A complex mix of **physics, electronics, computer science, networking** (etc...)

DAQ duties

- Gather data produced by detectors
 - **Readout**
 - Form complete events
 - **Data Collection** and **Event Building**
 - Possibly feed extra processing levels
 - Store event data
 - **Data Logging**
- } **Data Flow**
- Manage operations
 - **Control, Configuration, Monitoring**

Interlude: data vs *interesting* data

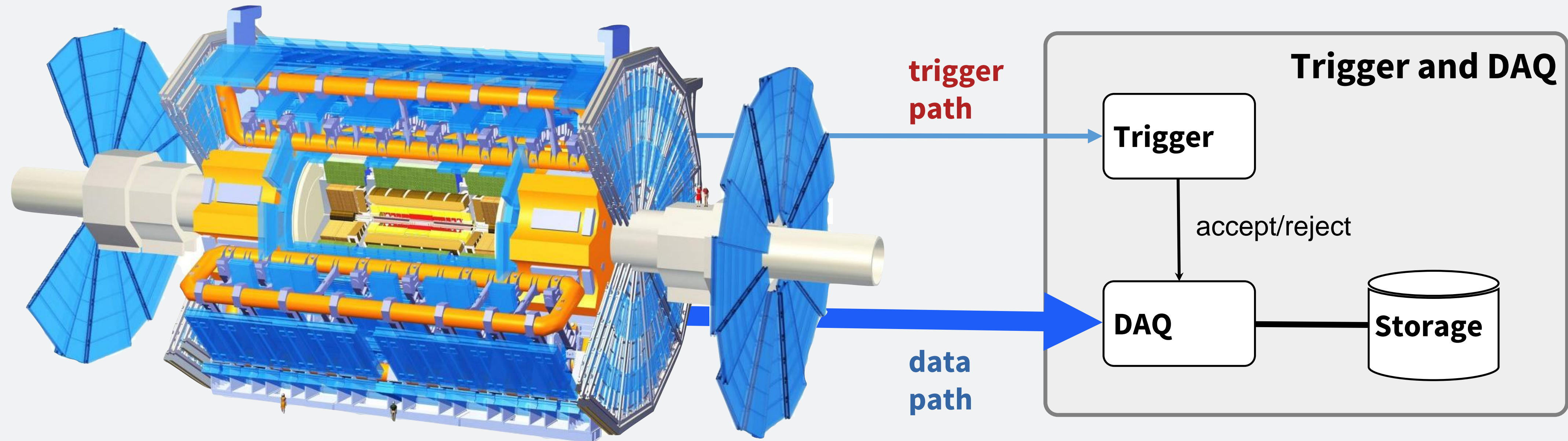
- *Interesting physics data* typically a small fraction of sampled signals
 - **really, Really, REALLY** small
- Storing all recorded data is unpractical (and costly)
 - sometimes technically unfeasible
- Online data reduction before permanent storage imperative
- That's the job of the **Trigger!**
 - DAQ and Trigger deeply entwined (often referred to as TDAQ)
 - Trigger Lecture on Tuesday - Dr. Tim Martin



Trigger in a nutshell

- Selects interesting events **AND** rejects the rest, *in real time*
 - **Selective:** efficient for “signal” and resistant to “background”
 - **Simple and robust:** Must be predictable at all times!
 - **Fast:** Late is no better than never
- With minimal *controlled latency*
 - time it takes to form and distribute its decision
- The implementation of “Trigger” has significantly evolved in the past decades
 - **LEP:** trigger the sampling of (slow) detector
 - **LHC:** trigger the readout of on-detector buffers (*Level-1*) or trigger transfer to permanent storage (*High Level Trigger - HLT*)

From detector to TDAQ



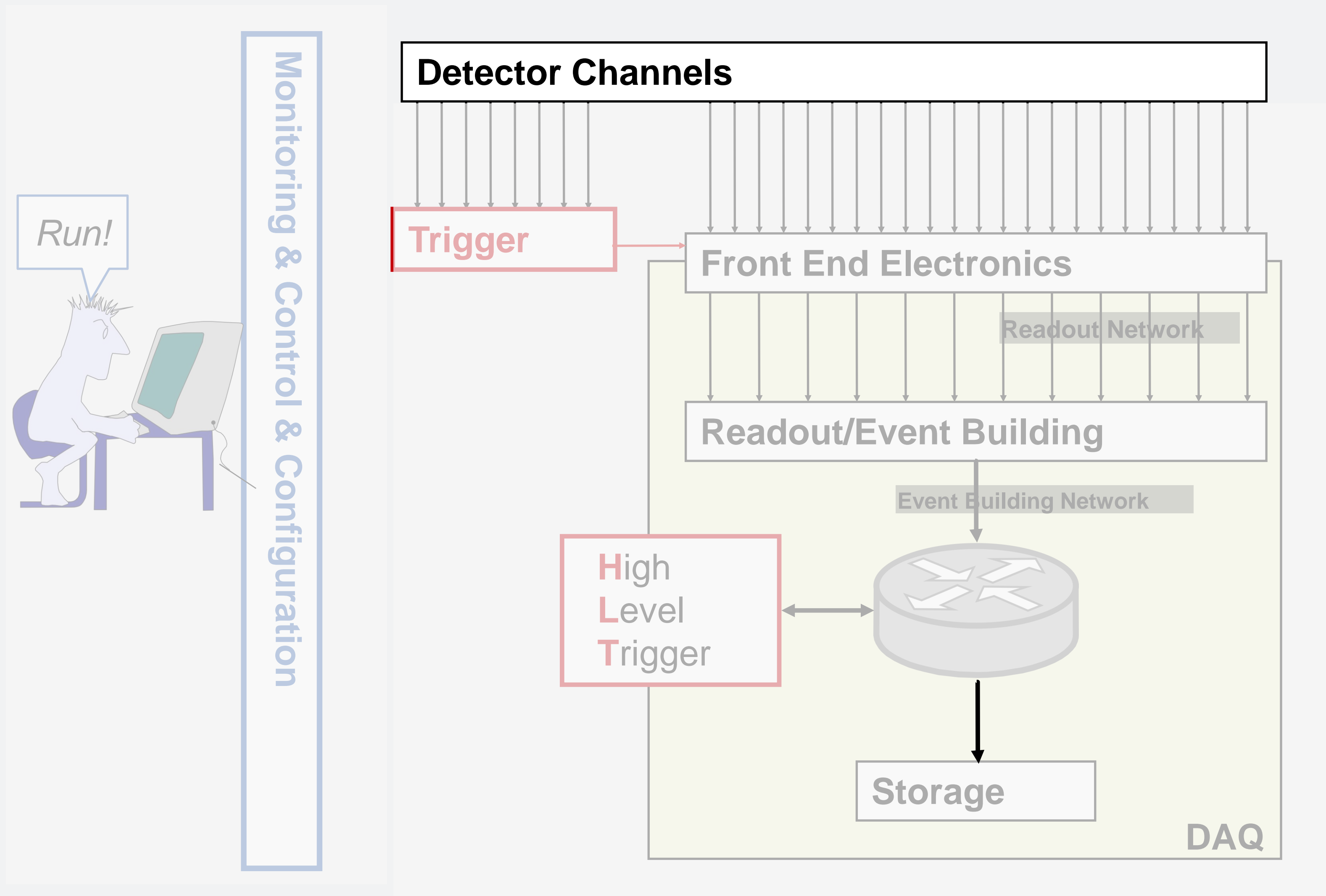
■ Trigger path

- From specific detectors to trigger logic
- Continuous streaming of trigger data
- Dedicated connections

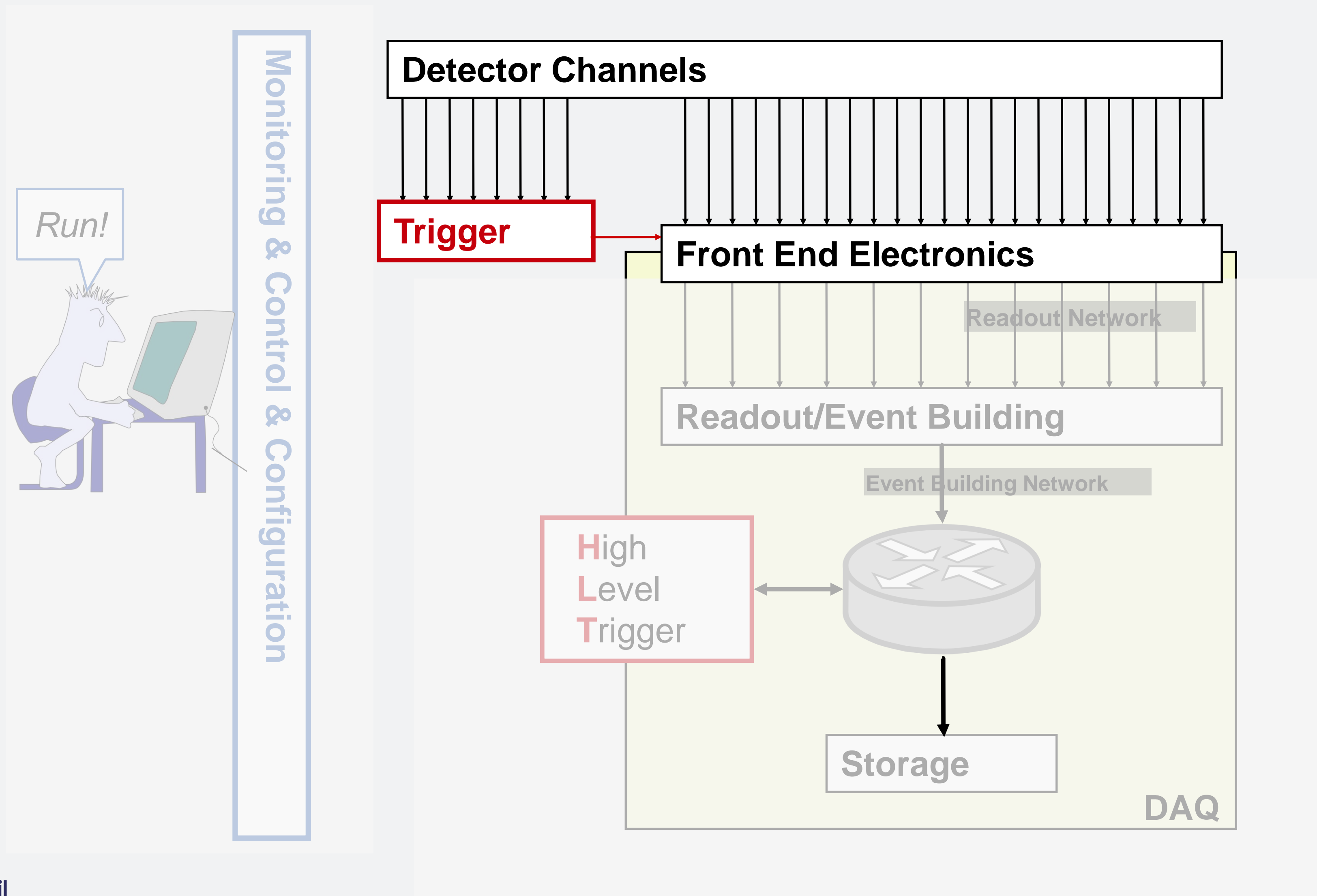
■ Data path

- From all the detectors to readout
- Transmission on positive trigger decision

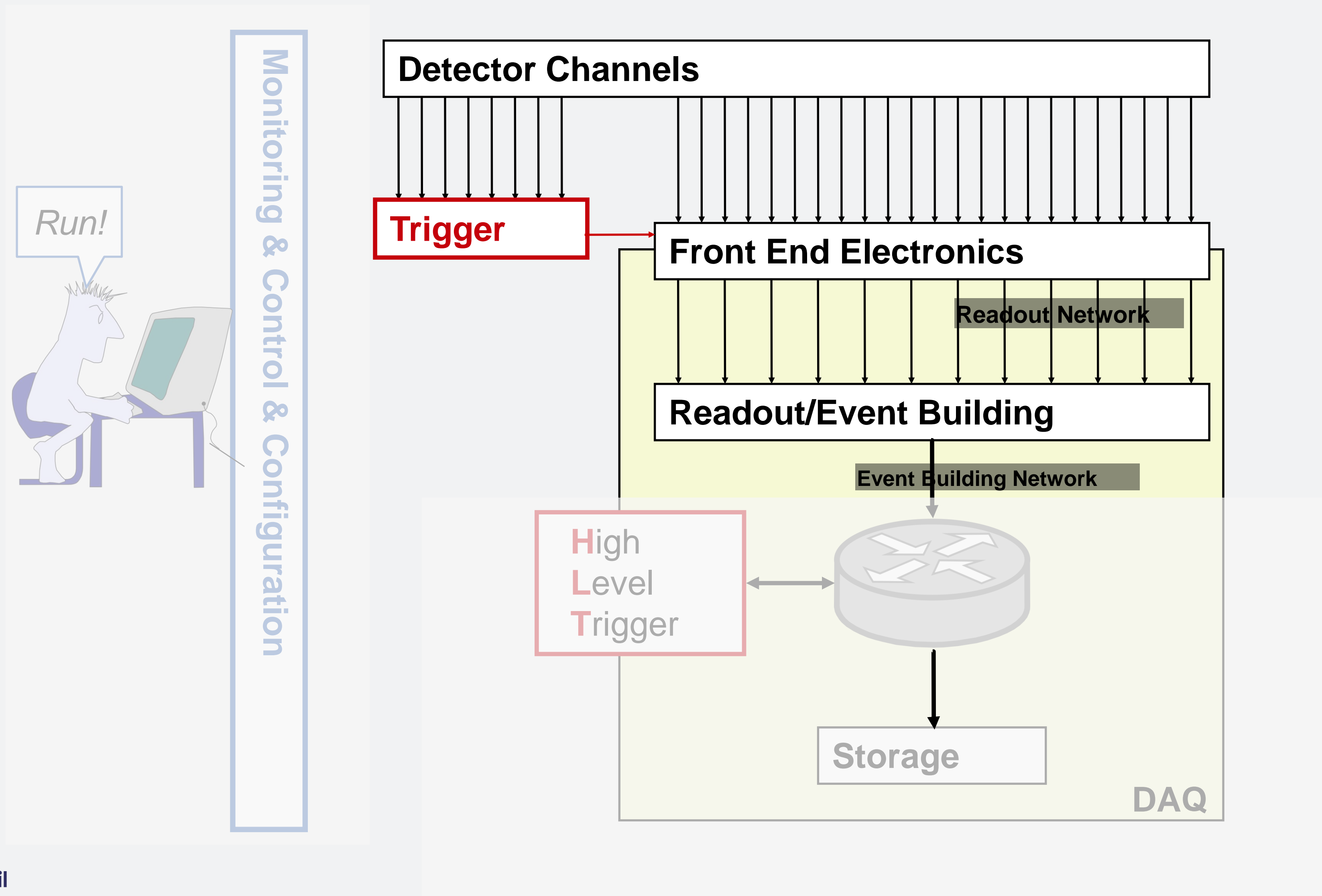
TDAQ Architecture



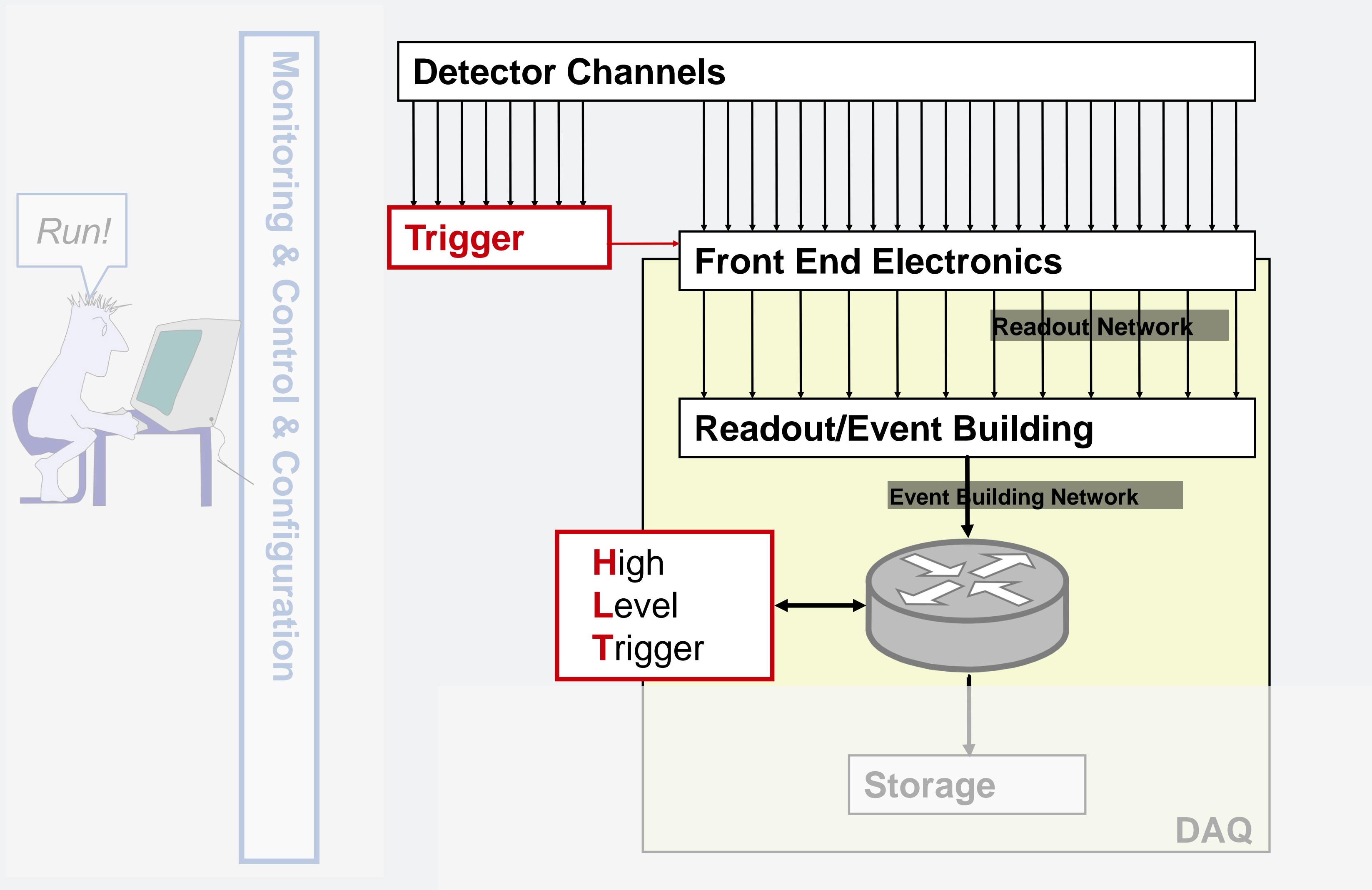
TDAQ Architecture



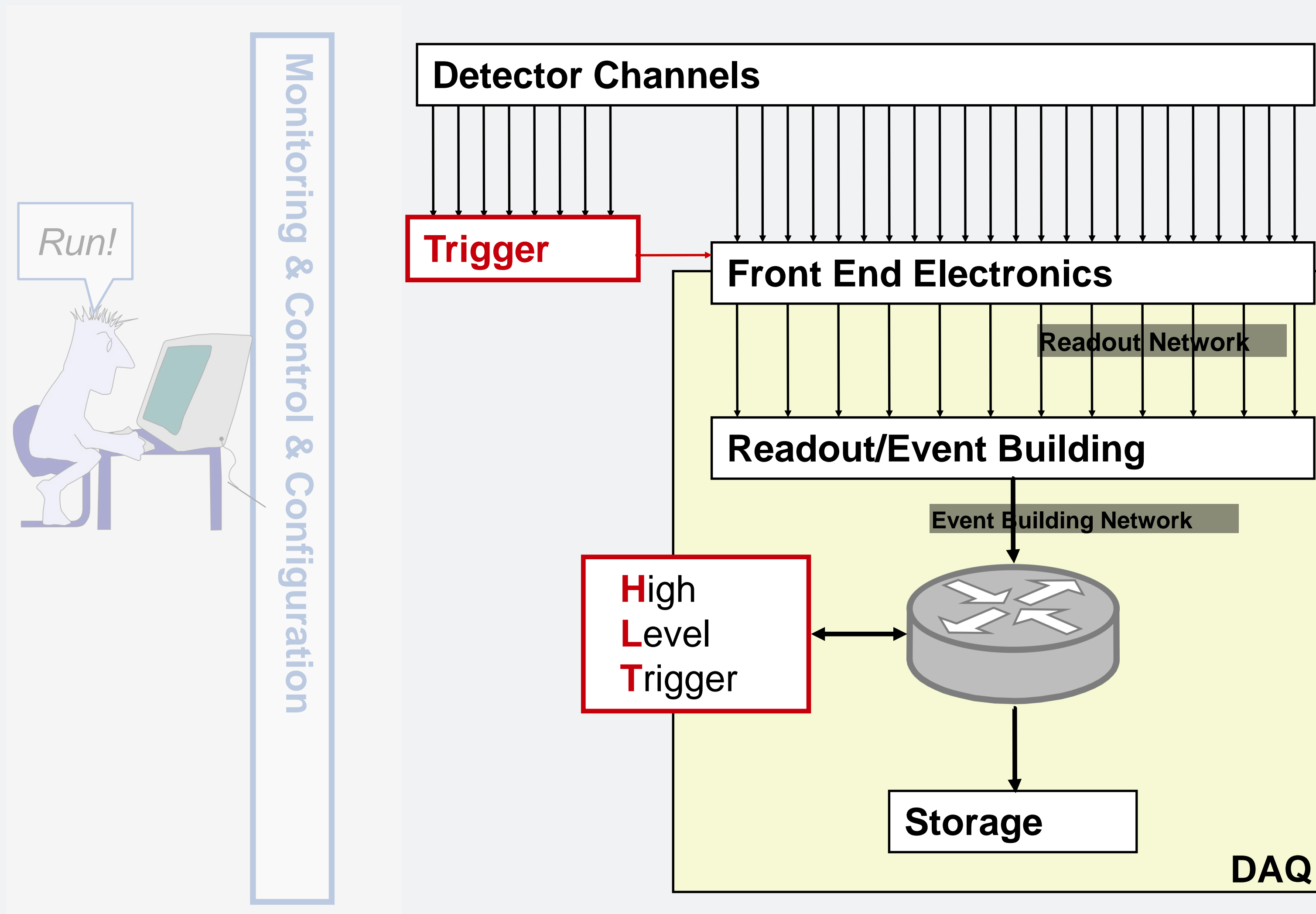
TDAQ Architecture



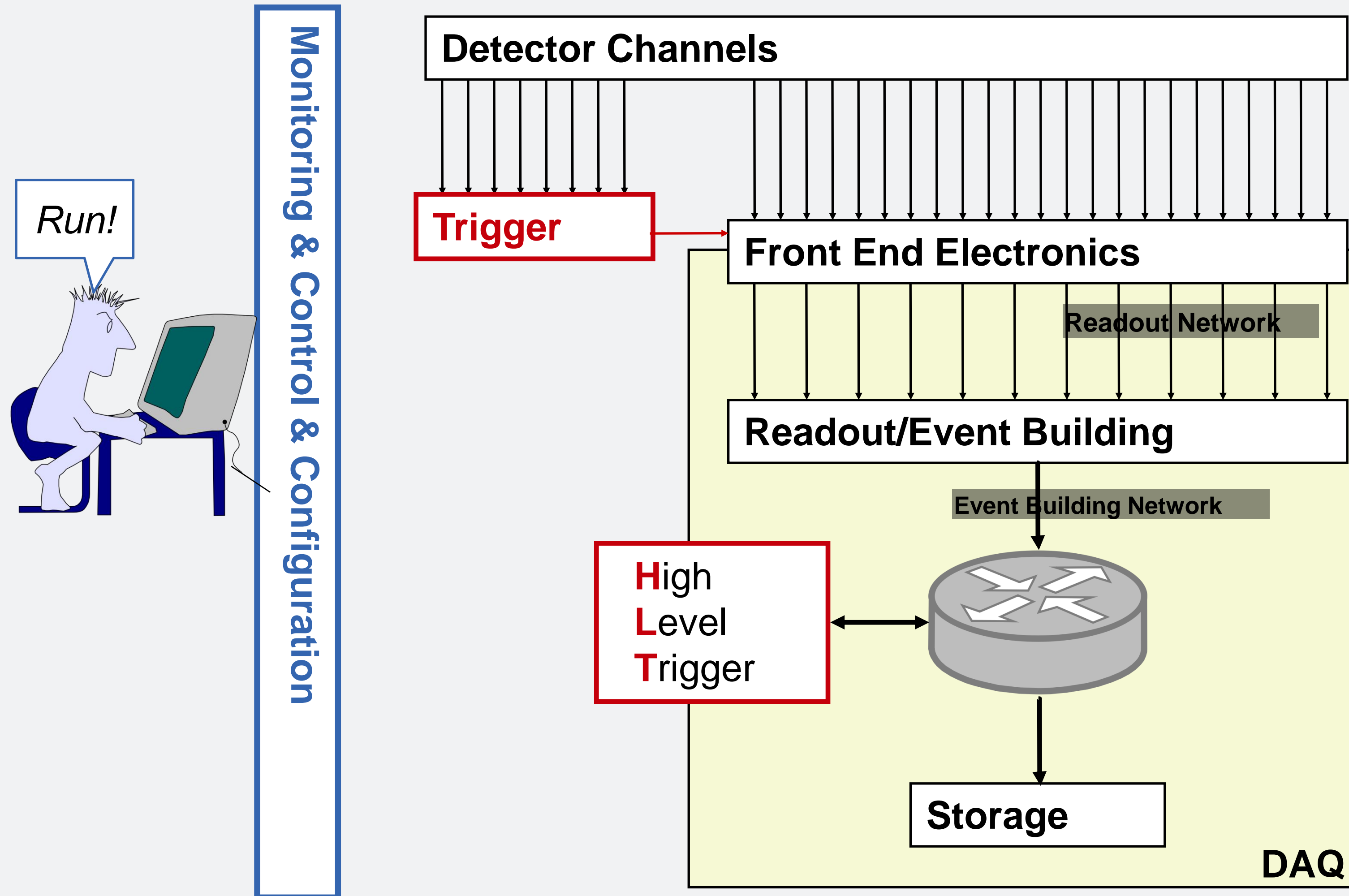
TDAQ Architecture



TDAQ Architecture



TDAQ Architecture



Outline

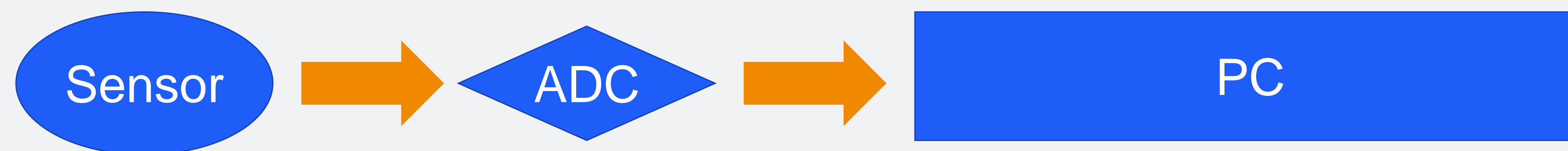
- *Introduction*
 - What is DAQ?
 - Example System architecture
- *Basic DAQ concepts*
 - Digitization, Latency
 - Deadtime, Busy, Backpressure
 - De-randomization
- *Scaling up*
 - Readout and Event Building
 - Buses vs Network
- *DAQ Challenges at the LHC and beyond*



with a toy model

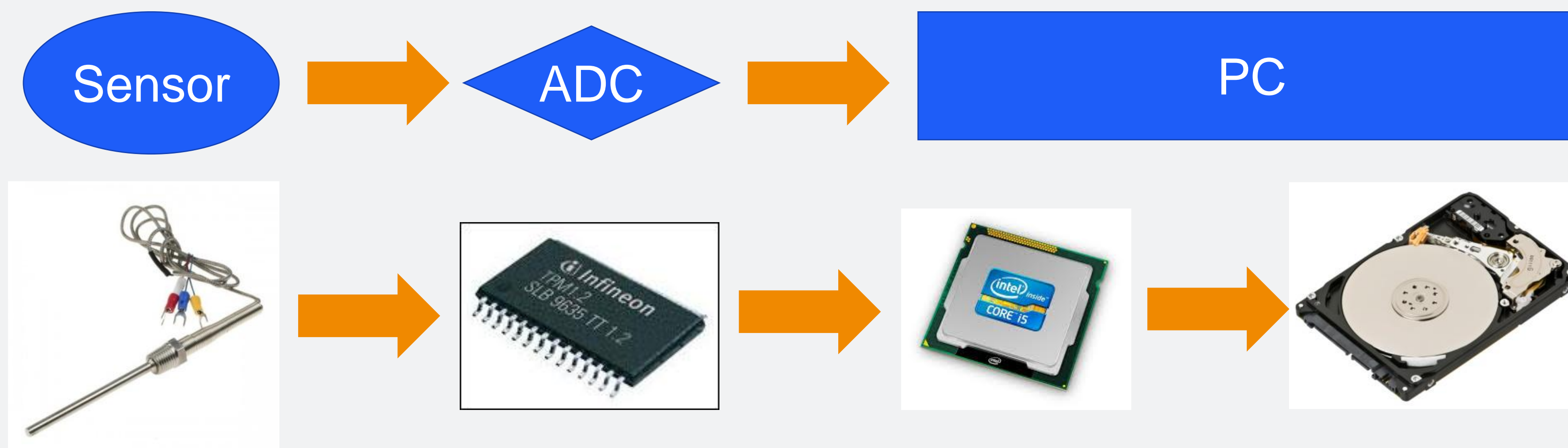
DAQ Basics

- Simple example: temperature sensor



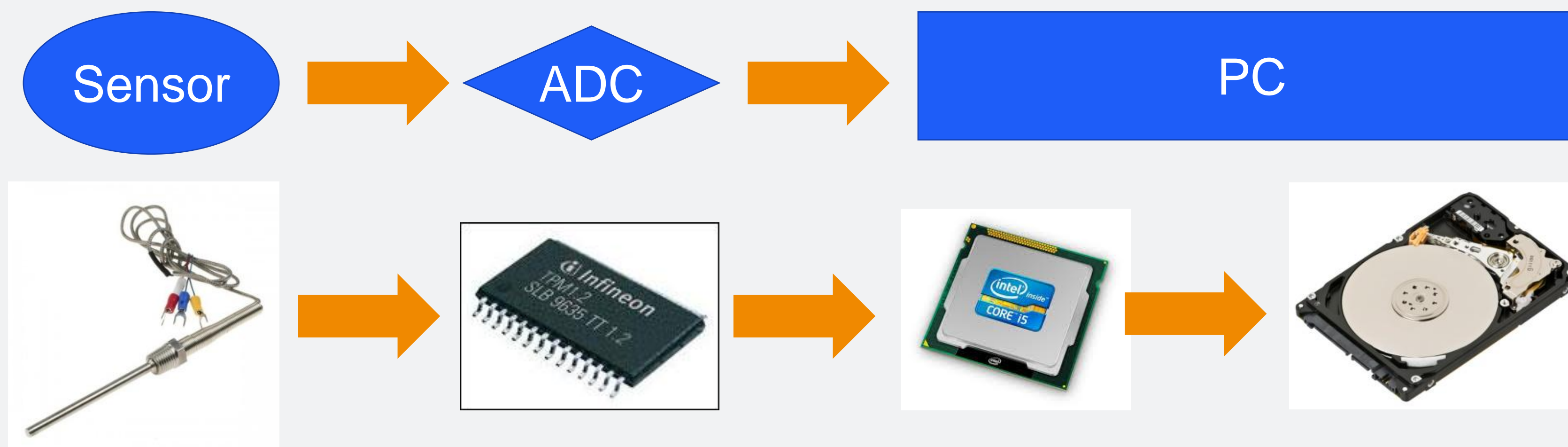
DAQ Basics

- Simple example: temperature sensor



DAQ Basics

- Simple example: temperature sensor



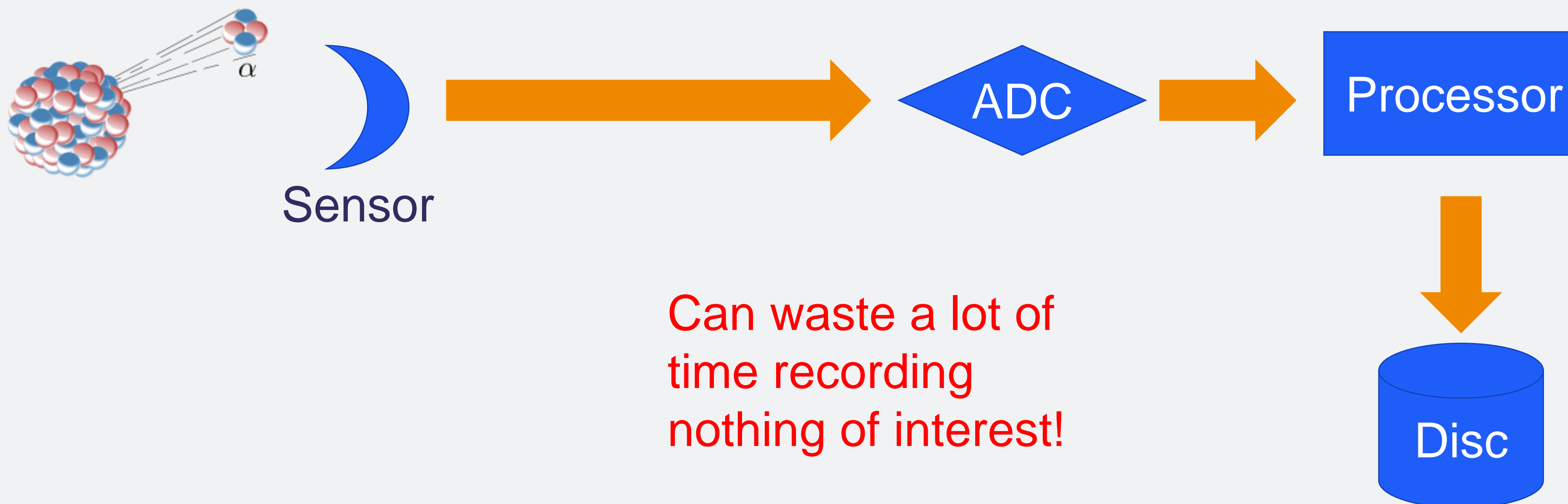
Limited by:

Analogue-to-Digital conversion rate

Data recording rate

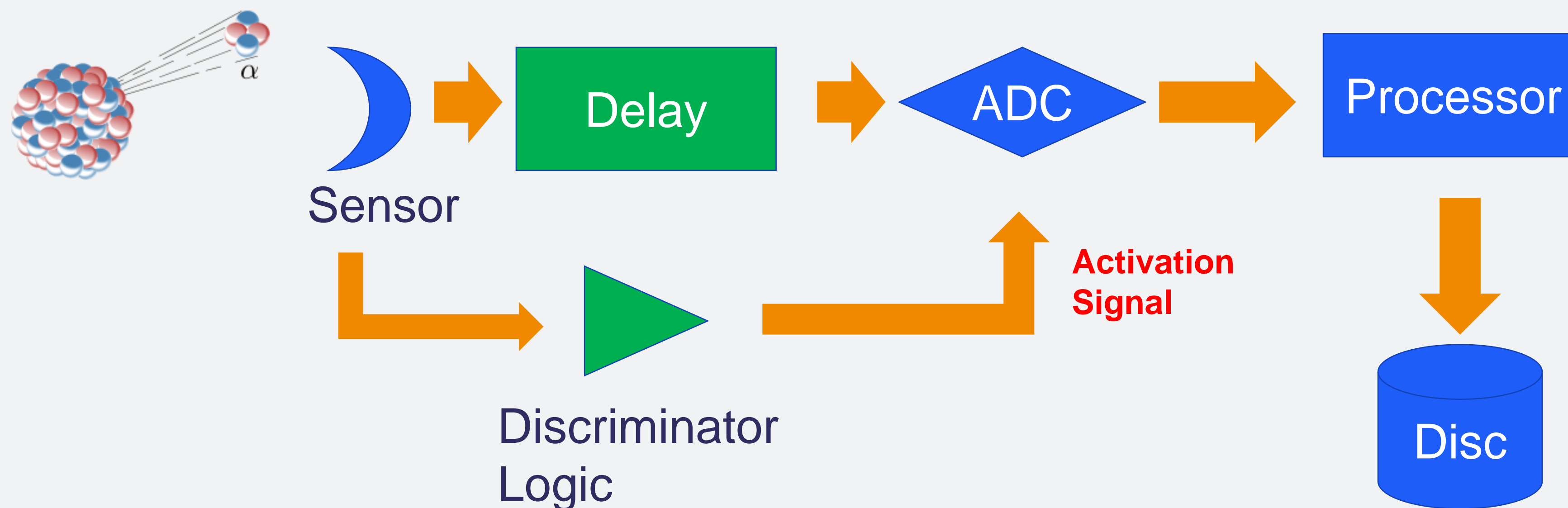
DAQ Basics

- What if process is unpredictable?
 - e.g. radioactive decay



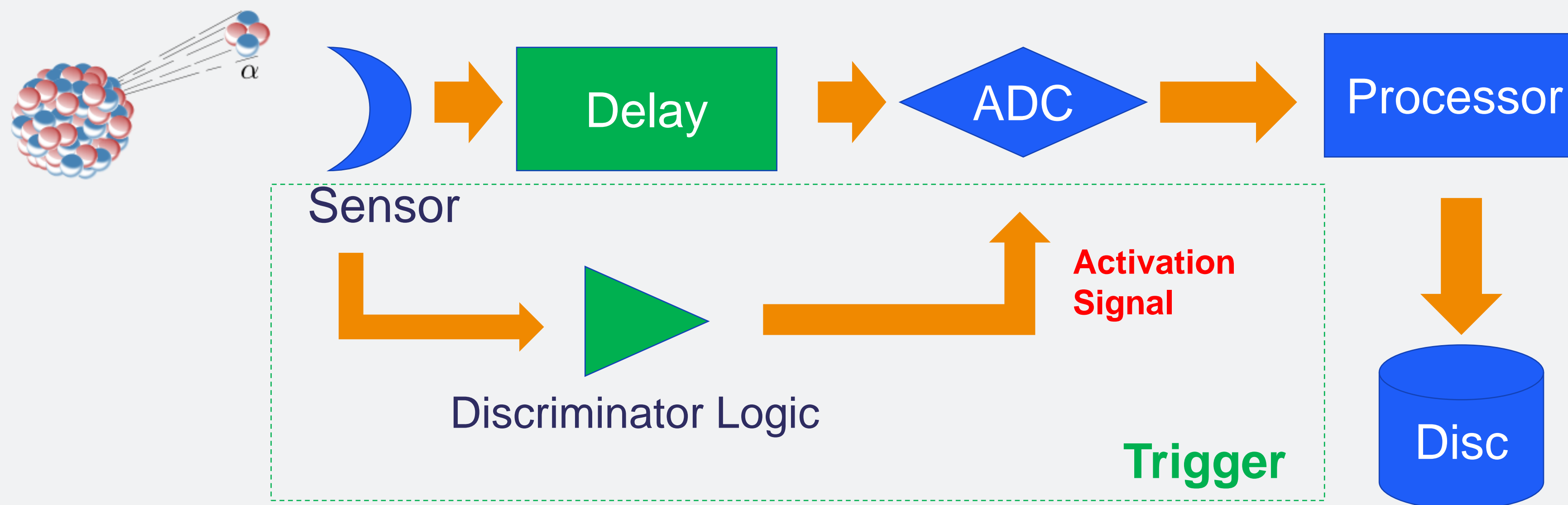
DAQ Basics

- What if process is unpredictable?
 - e.g. radioactive decay



DAQ Basics

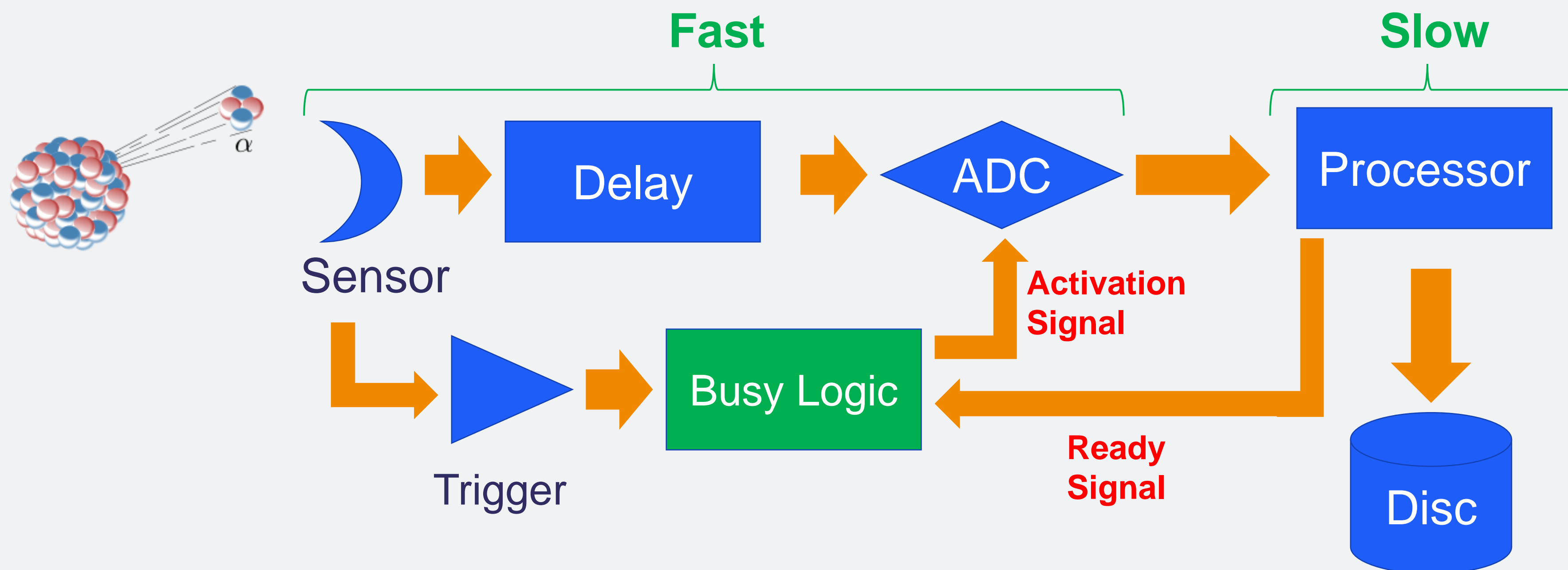
- What if process is unpredictable?
 - e.g. radioactive decay



Need to delay data to ADC to synchronise with trigger: **latency**

DAQ Basics

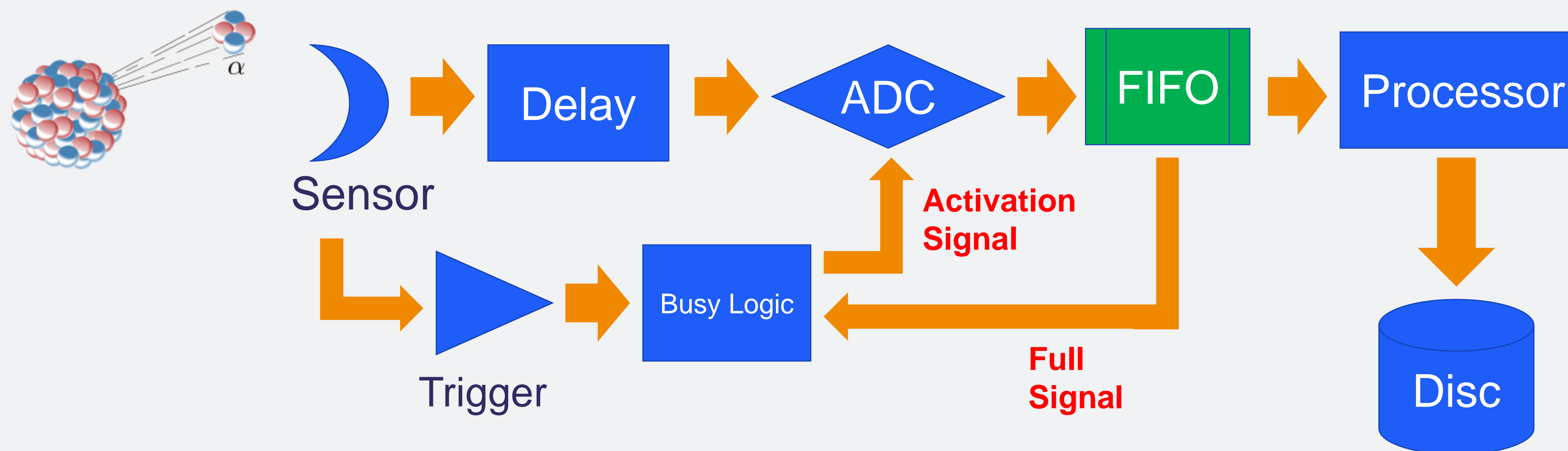
- Not whole story
 - What if new signal arrives while still processing old one?
 - Need mechanism to stop additional data from being collected while system is unavailable: **busy logic**



Time when system not available to new signals: **deadtime**
Dominated by slow component

DAQ Basics

- Additional improvement: ‘Derandomising’ buffer
 - ‘First-in First-out’ structure (FIFO)

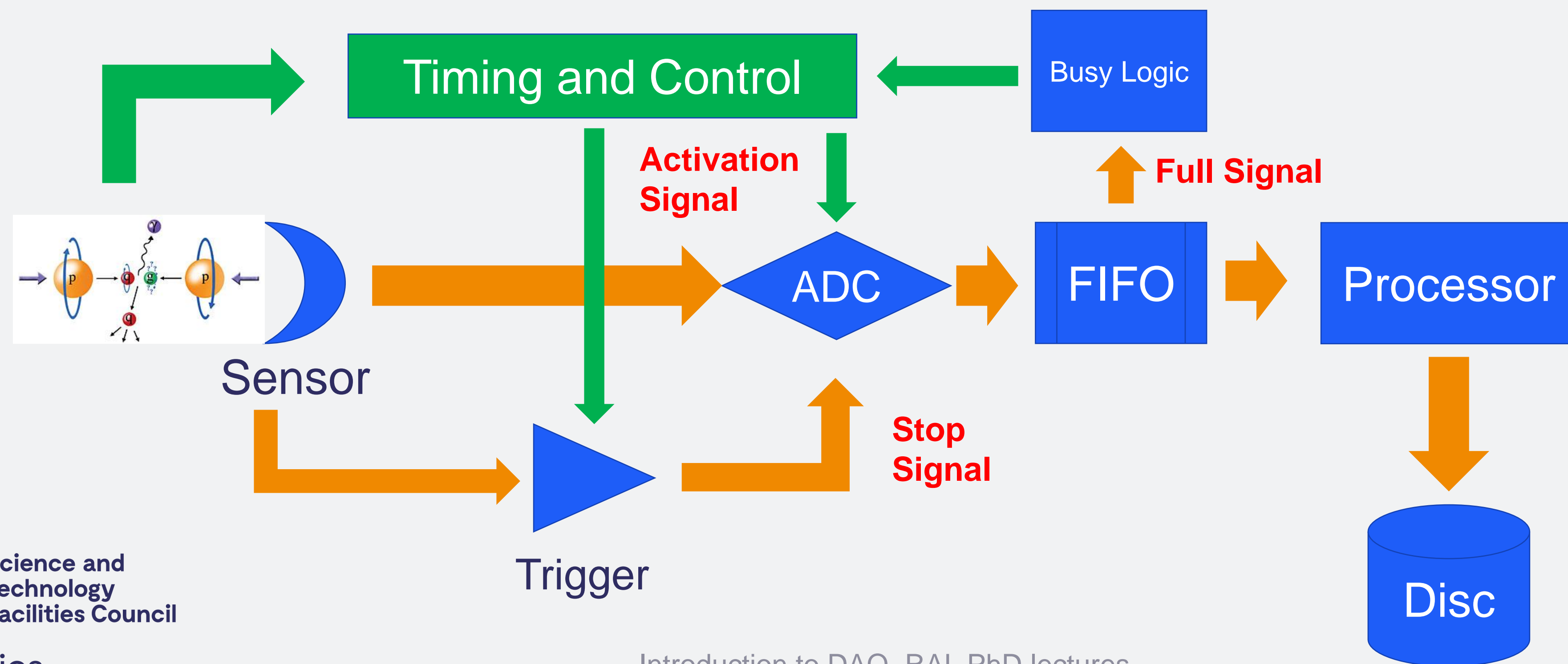


Decouple front-end (fast) from readout (slow)
No need for more expensive and complex readout

DAQ Basics

■ Collider Experiment:

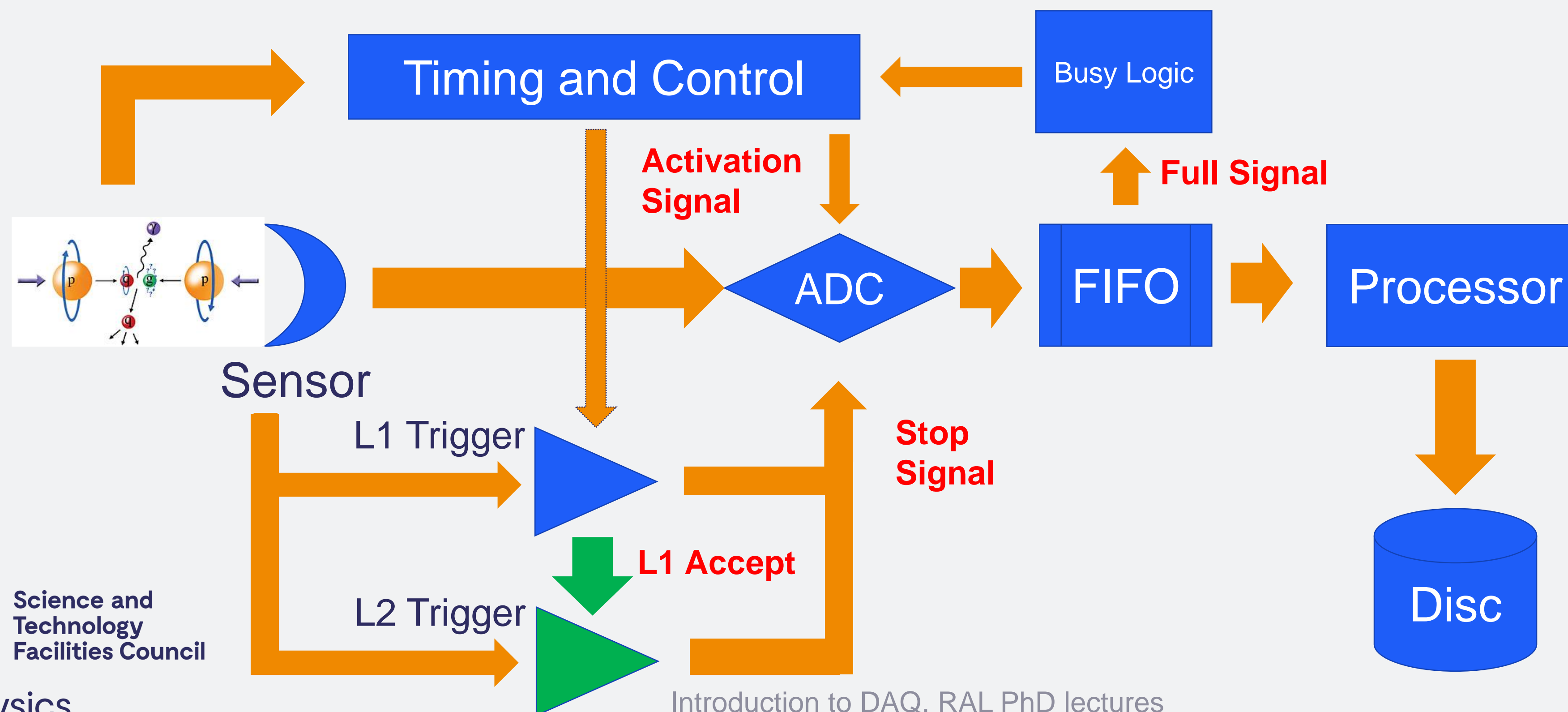
- We know beam crossing rate, can predict when interesting data might be available
 - Configure Trigger to operate synchronously to beam crossings
- (May) no longer need for initial delay, but still have deadtime from slowest components



DAQ Basics

■ Collider Experiment:

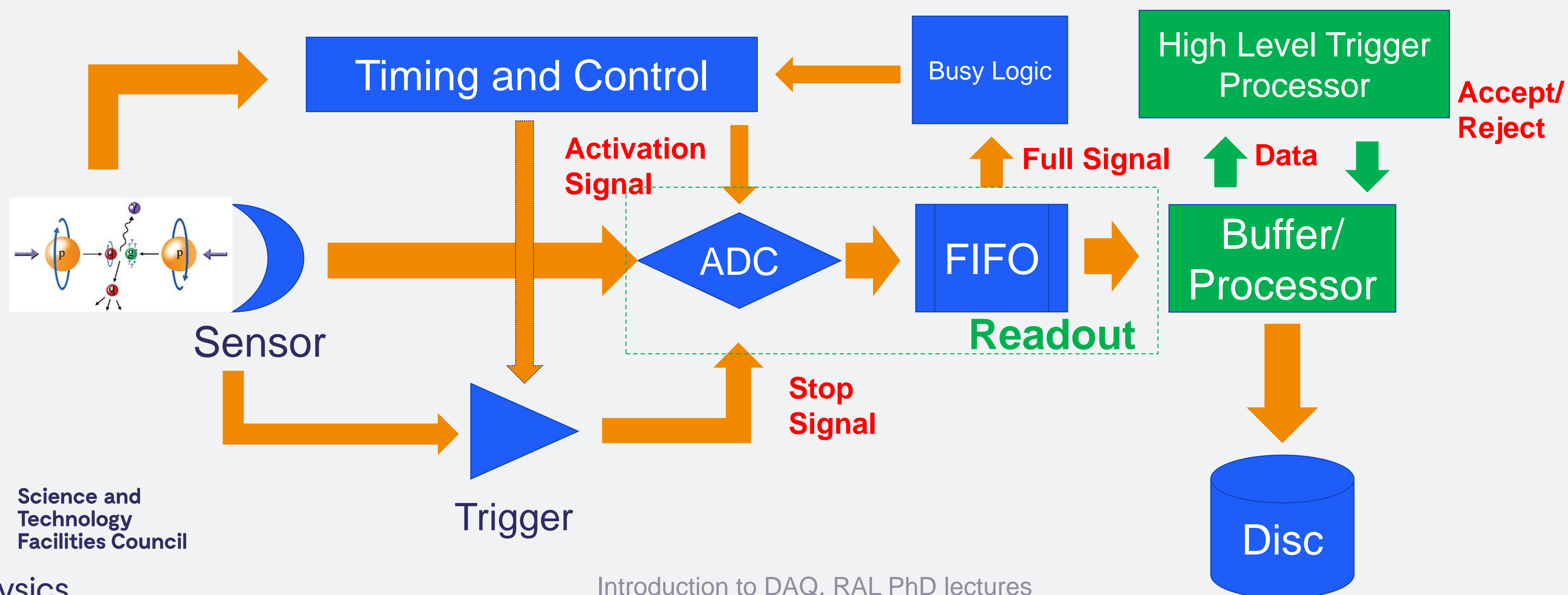
- No problem if trigger latency less than bunch crossing period
- Not guaranteed with more complex triggers – add more levels
 - L1 Trigger here eliminates large fraction of events with simple criteria, allowing L2 Trigger to perform somewhat more complex analysis on a smaller fraction of the events



DAQ Basics

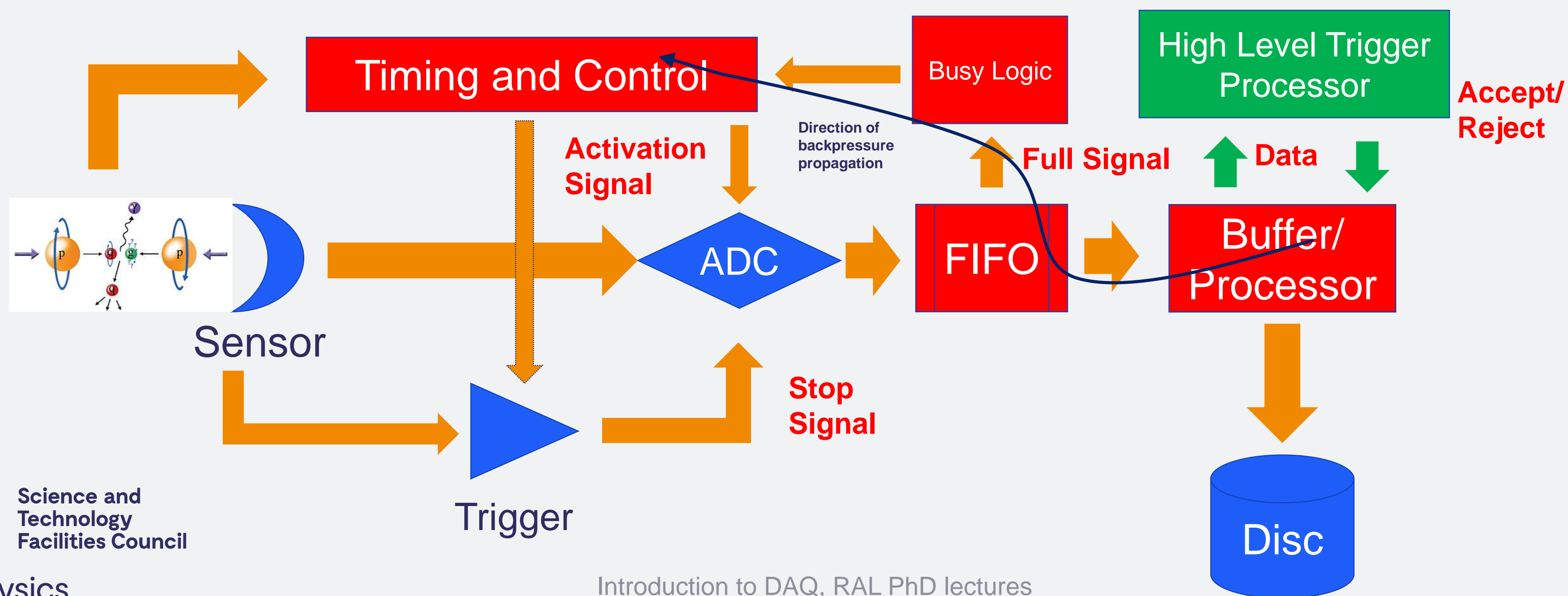
- Collider Experiment:

- Better results if you can do more complex calculations after readout stage – add a ‘high level’ trigger
- Readout now functions also as a buffer



DAQ Basics

- What if part of your system is struggling under the load?:
 - In this case, our processing/buffering component is saturated, generating **backpressure**
 - This causes components further back in the chain to also become saturated, propagating the backpressure until the busy logic steps in to stop the trigger
 - Experts often spend a lot of time working out where backpressure has arisen in the system

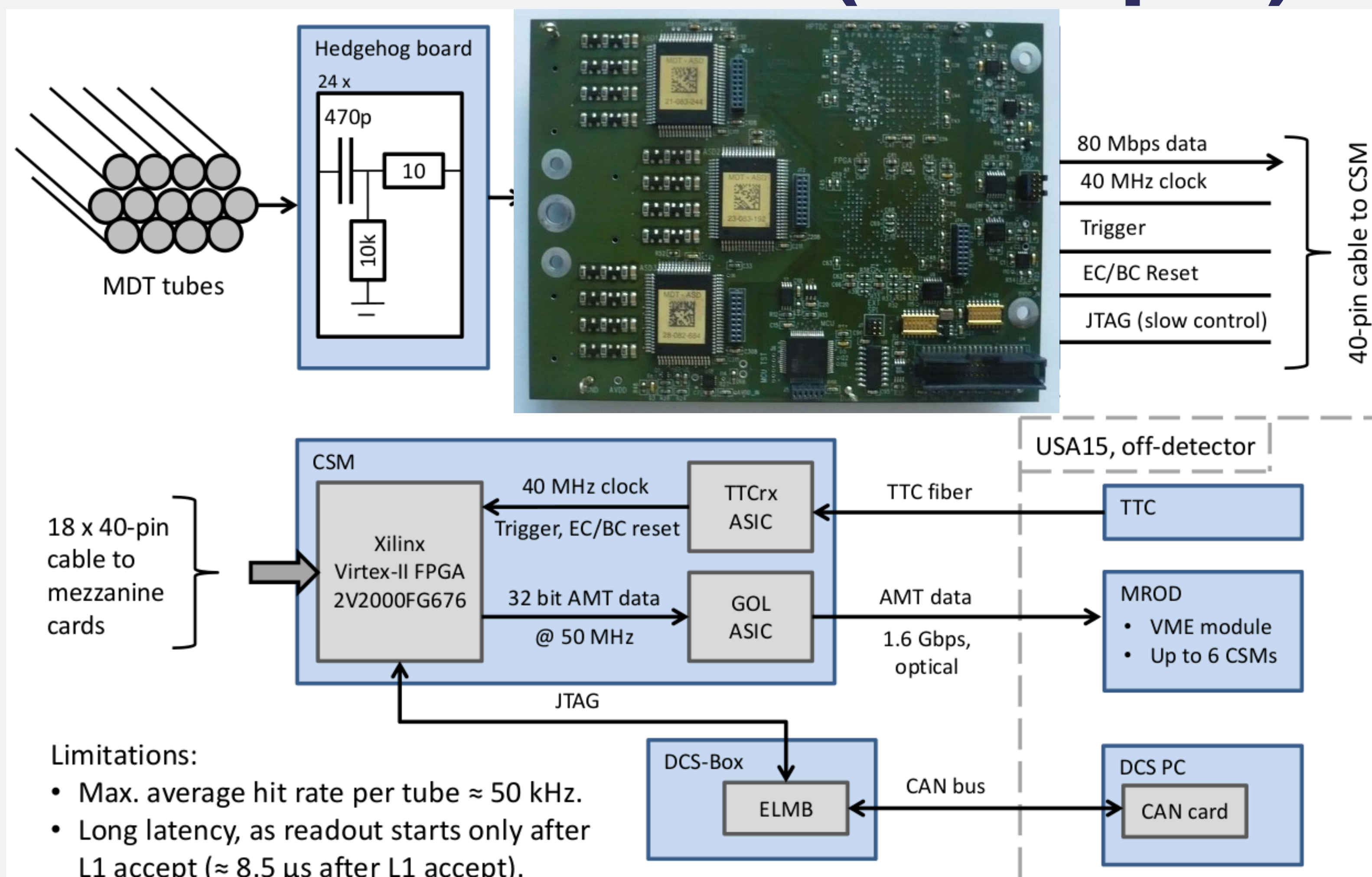


Outline

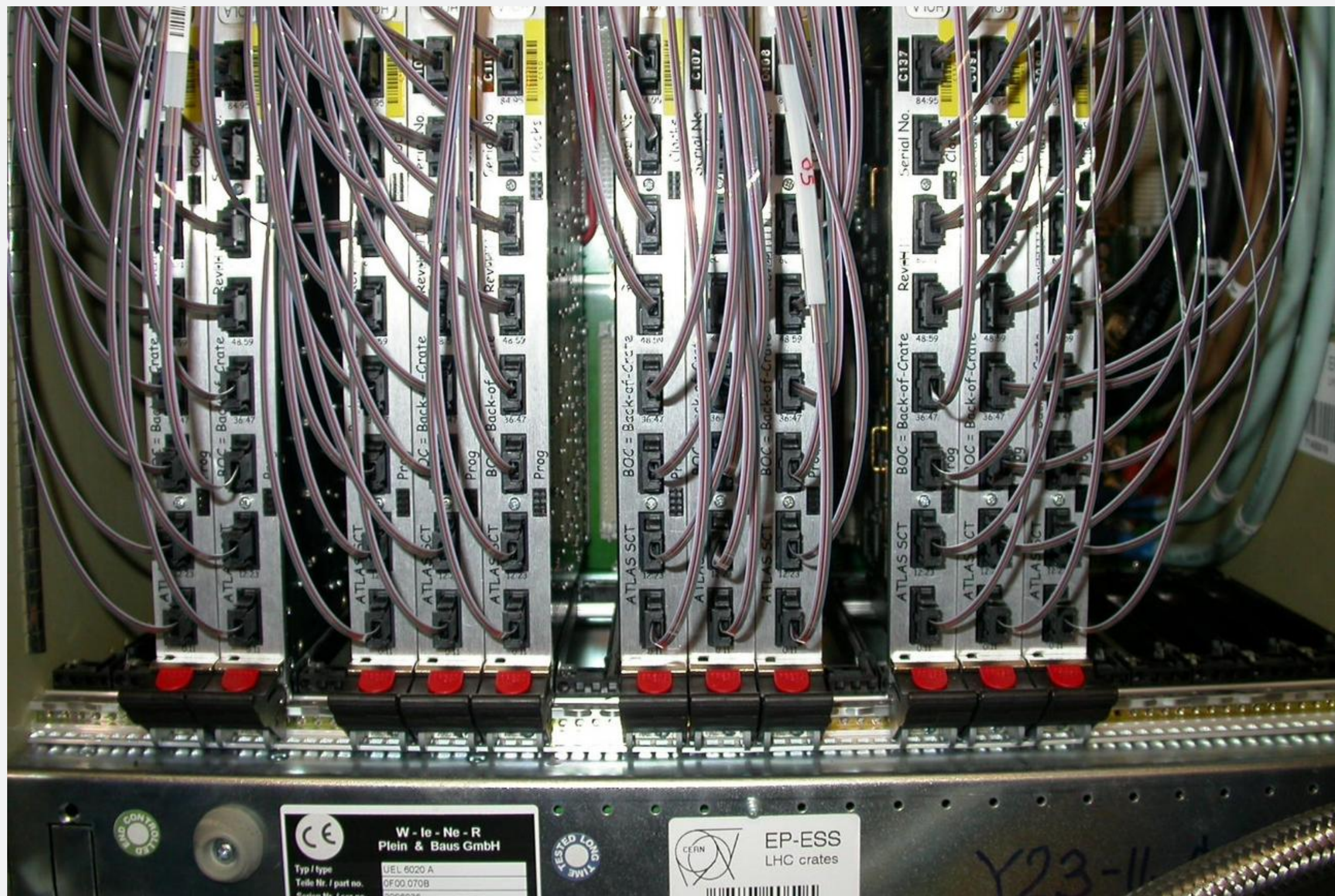
- *Introduction*
 - What is DAQ?
 - Example system architecture
- *Basic DAQ concepts*
 - Digitization, Latency
 - Deadtime, Busy, Backpressure
 - De-randomization
- *Scaling up*
 - Readout and Event Building
 - Buses vs Network
- *DAQ Challenges at the LHC and beyond*



Front End electronics (example)

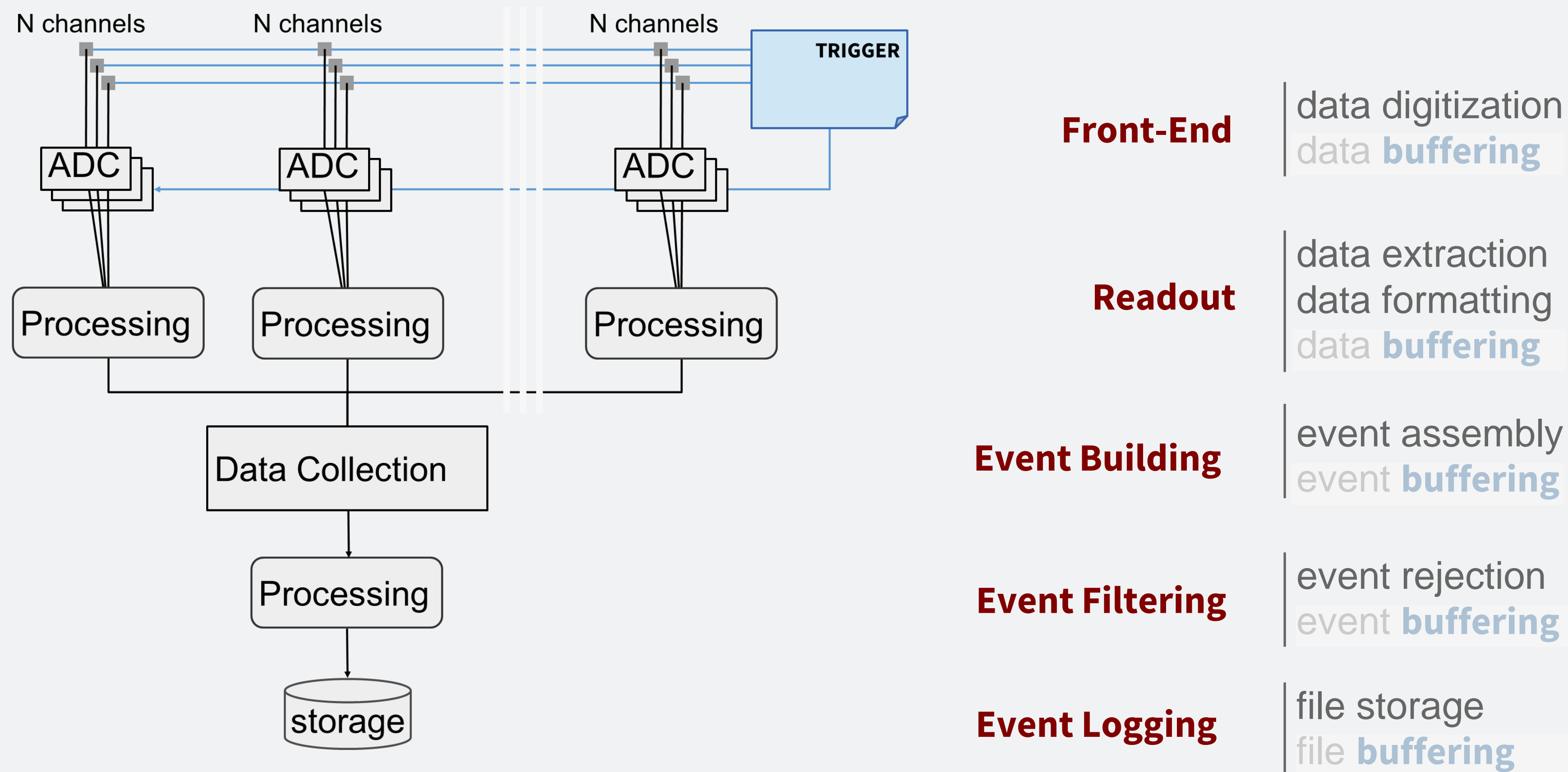


Readout Boards (Counting Room)



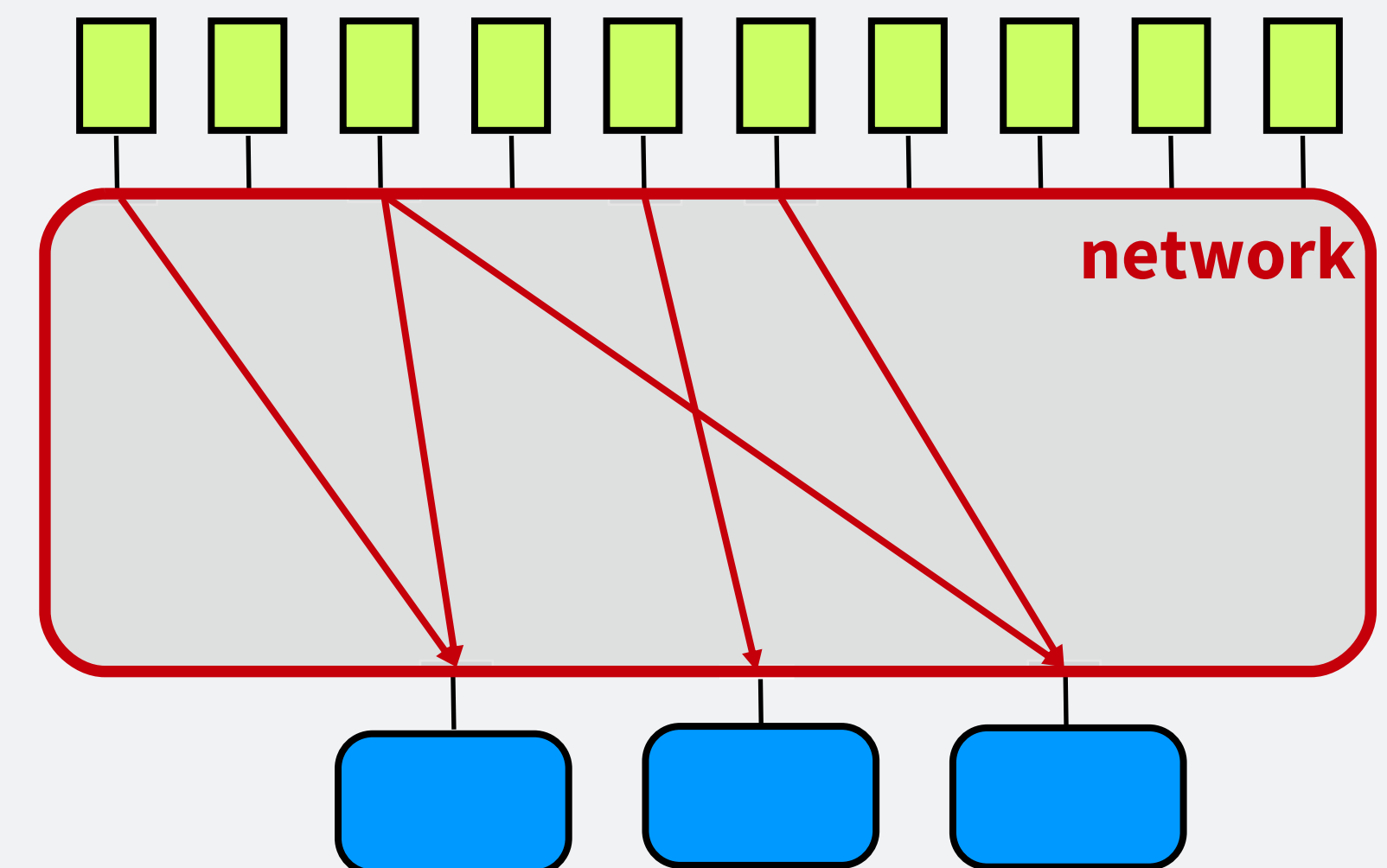
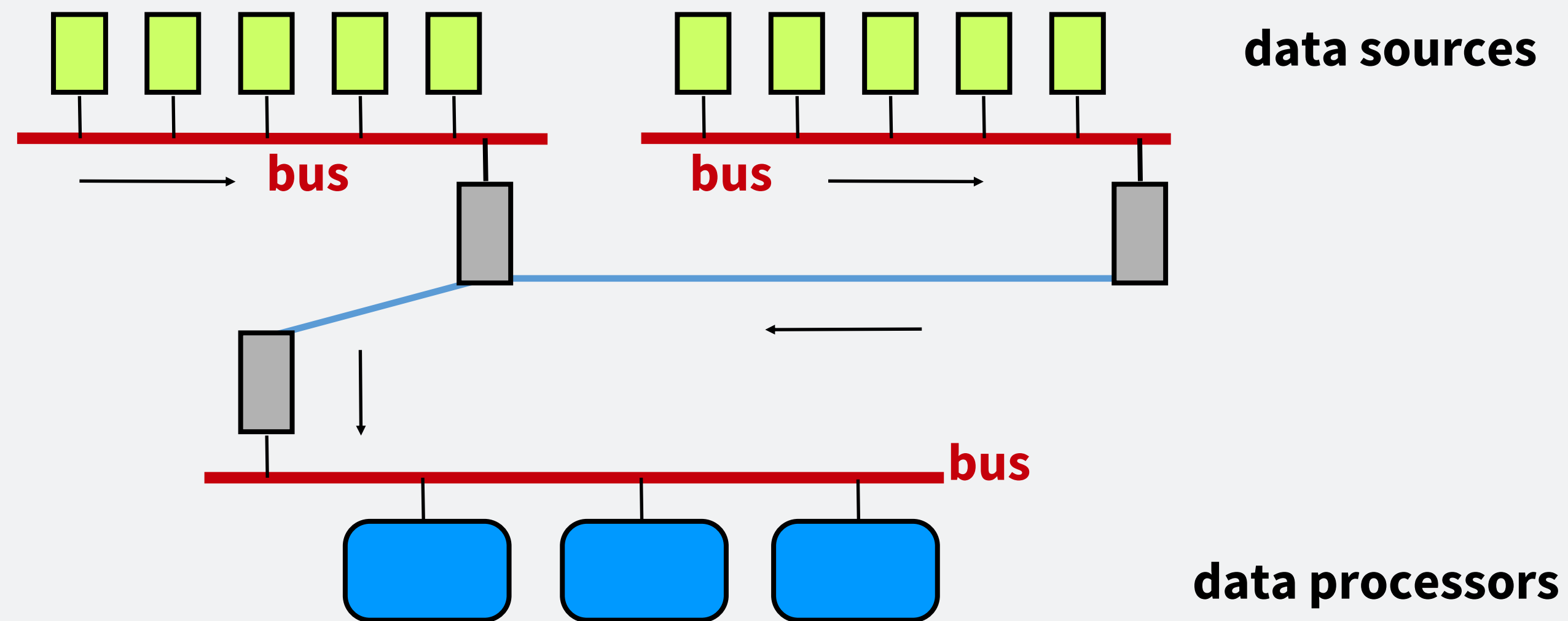
Scaling up the system

Adding more channels requires a hierarchical structure committed to data handling, aggregation and transport



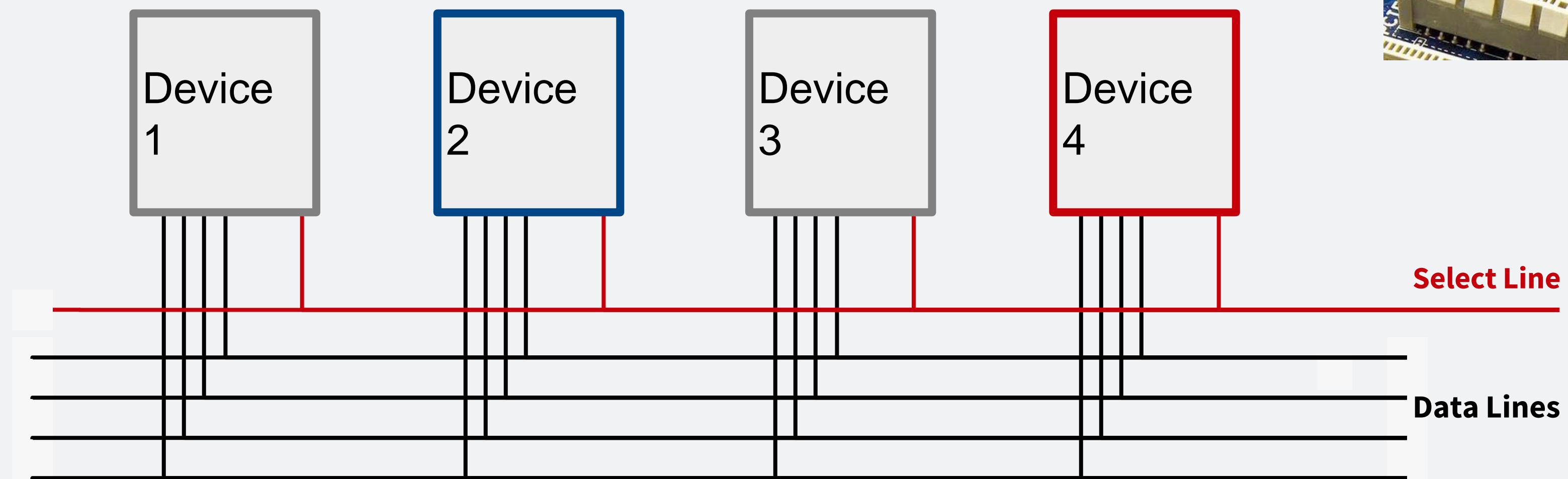
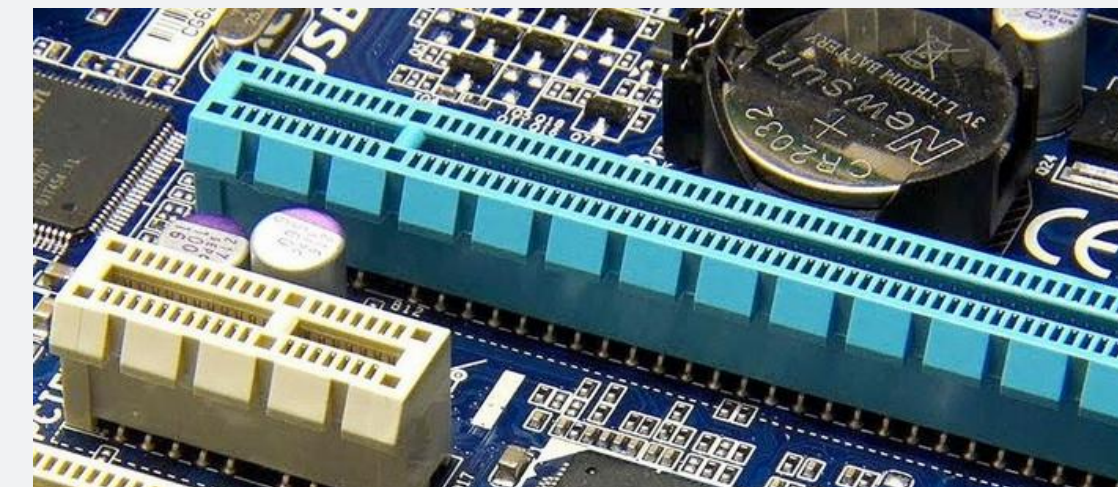
Readout Topology

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



Buses

- Devices connected via a **shared bus**
 - Bus → group of electrical lines
- Sharing implies **arbitration**
- Devices can be unique addresses on the bus
- E.g.: PCI, ATCA, SCSI, Parallel ATA, VME ...
 - local, external, crate, long distance, ...



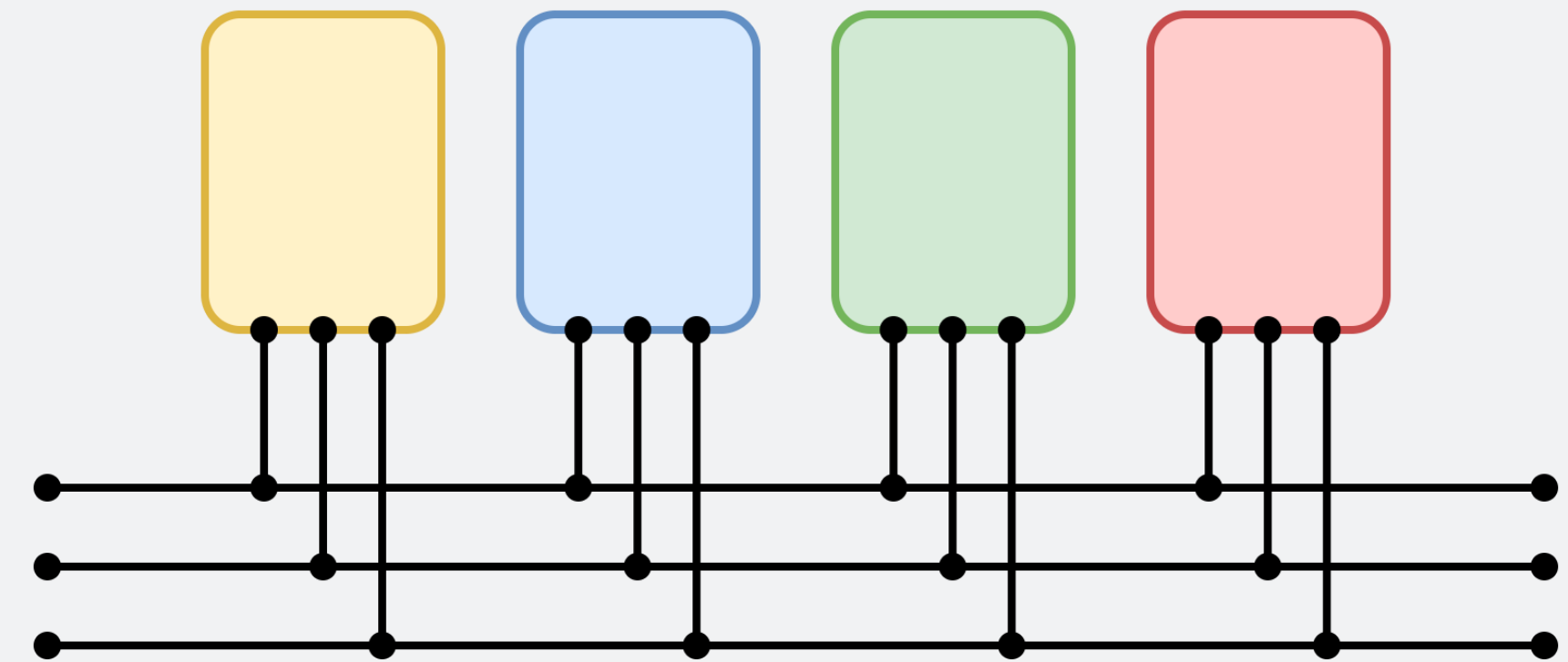
Bus facts

- **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, second order “effects” may limit the scalability of your system



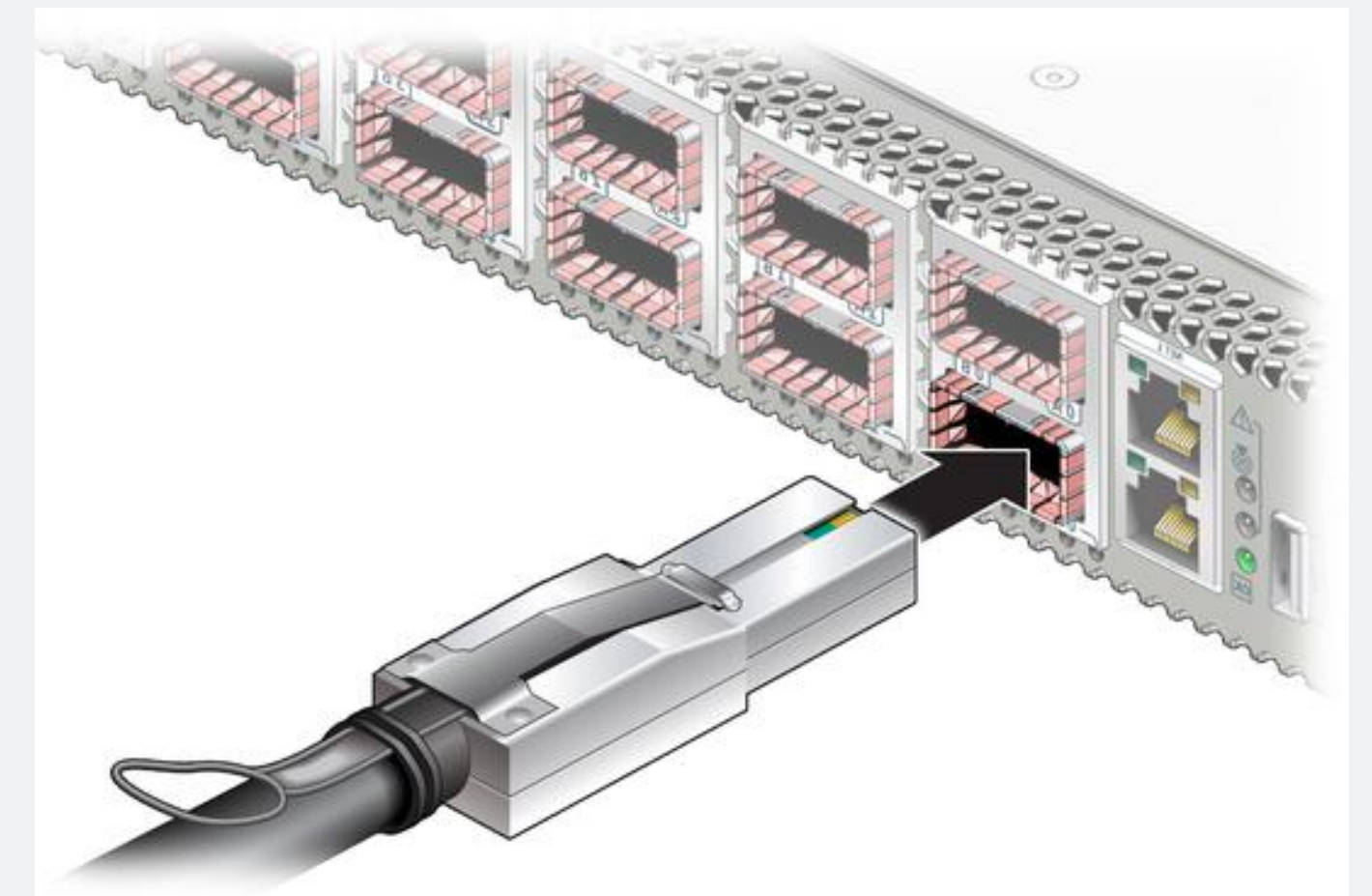
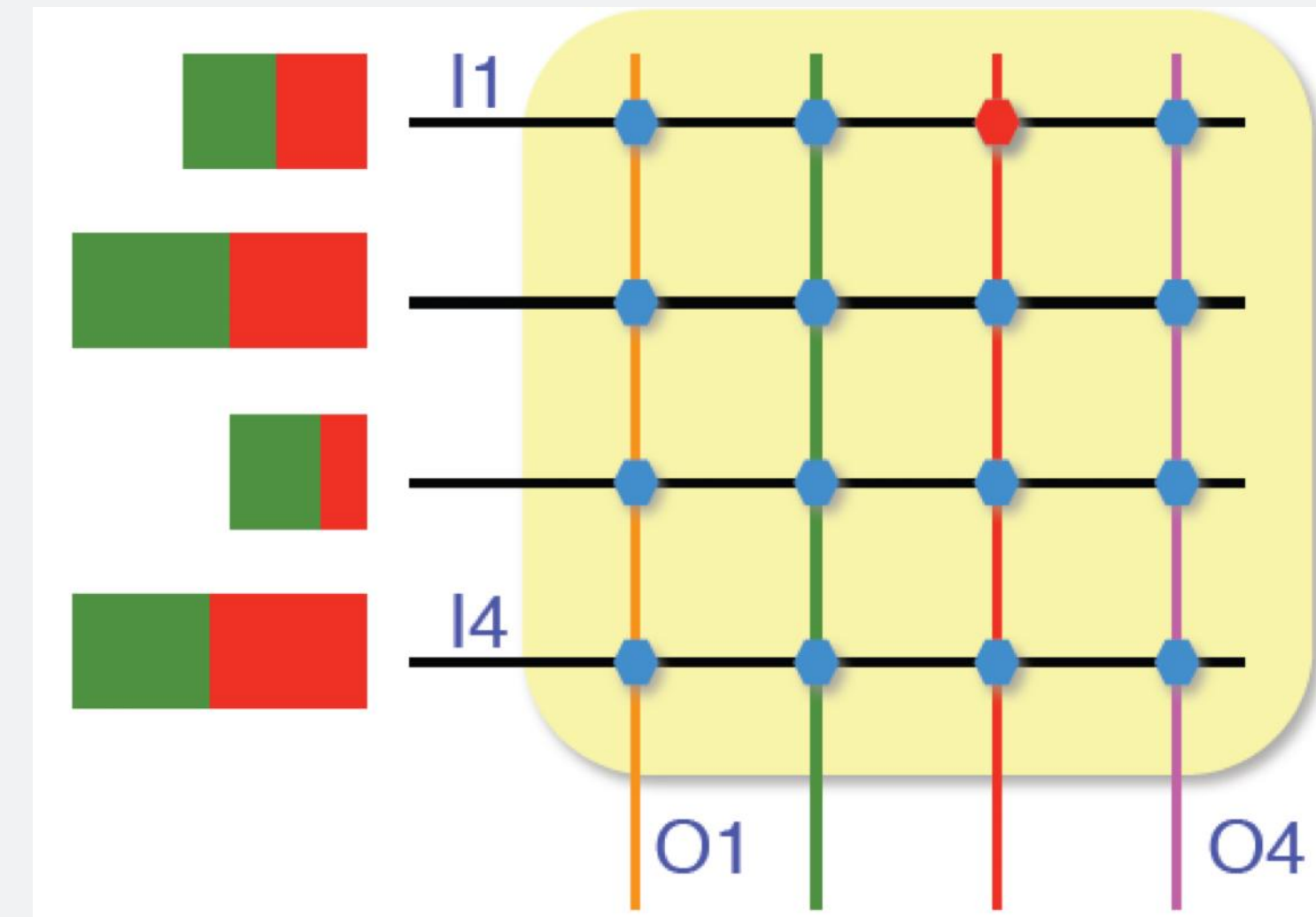
Bus facts



Some second order “effects”

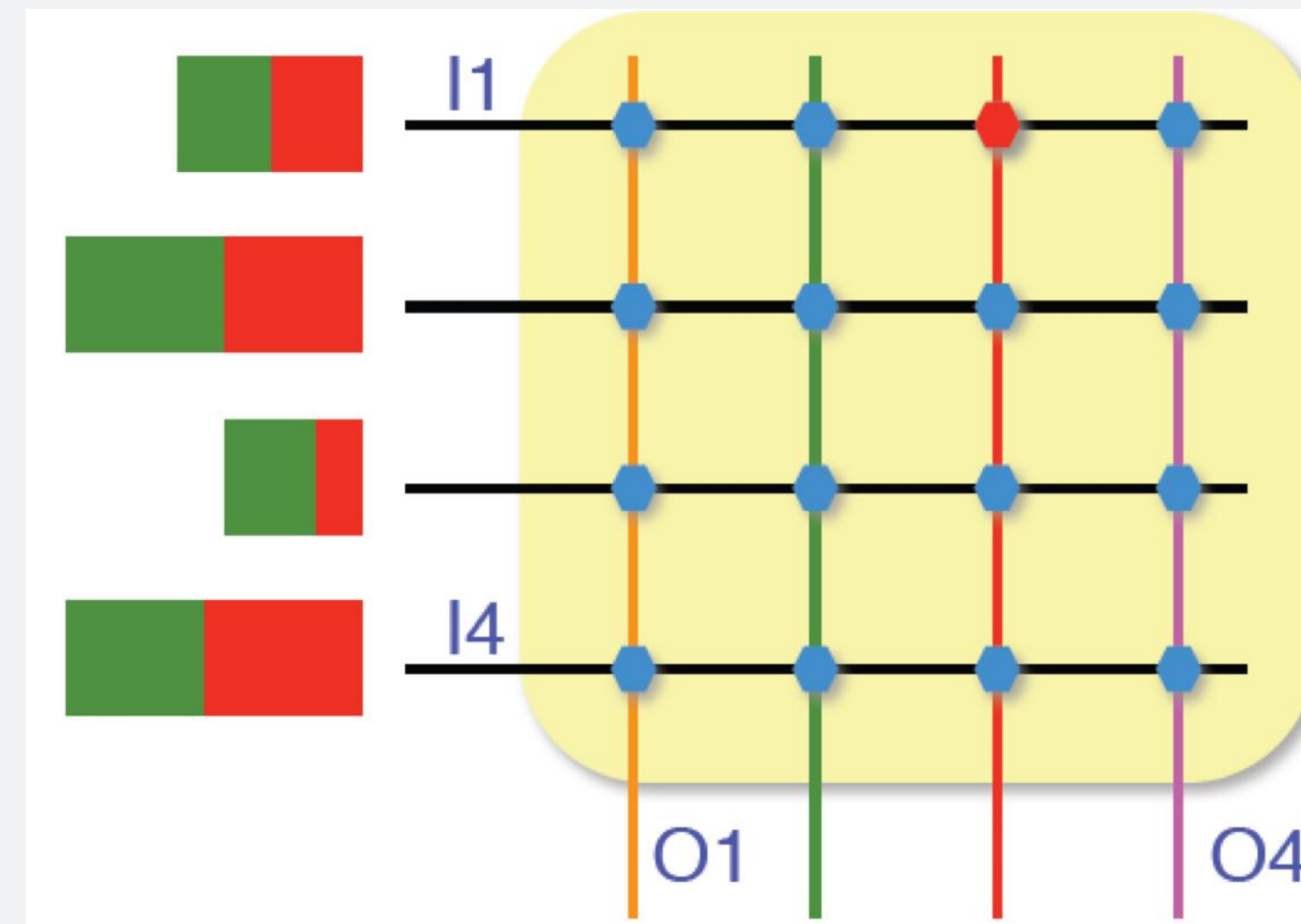
Networks

- All devices are equal (peers)
 - They **communicate directly** with each other via messages
 - No arbitration
 - Bandwidth guaranteed
 - Can be copper, optical, wireless
- Eg: Telephone, Ethernet, Infiniband, ...



Networks

- In switched networks, **switches** move messages between sources and destinations
 - Find the right path between endpoints
- **Congestion** (two messages with the same destination at the same time) handled through additional **buffering**
- Over time, experiments have updated designs to move the network layer closer to the detector, in place of bus-based systems
 - Technological advancement has made this possible
 - Easier to scale
 - Better access to COTS components



Networks

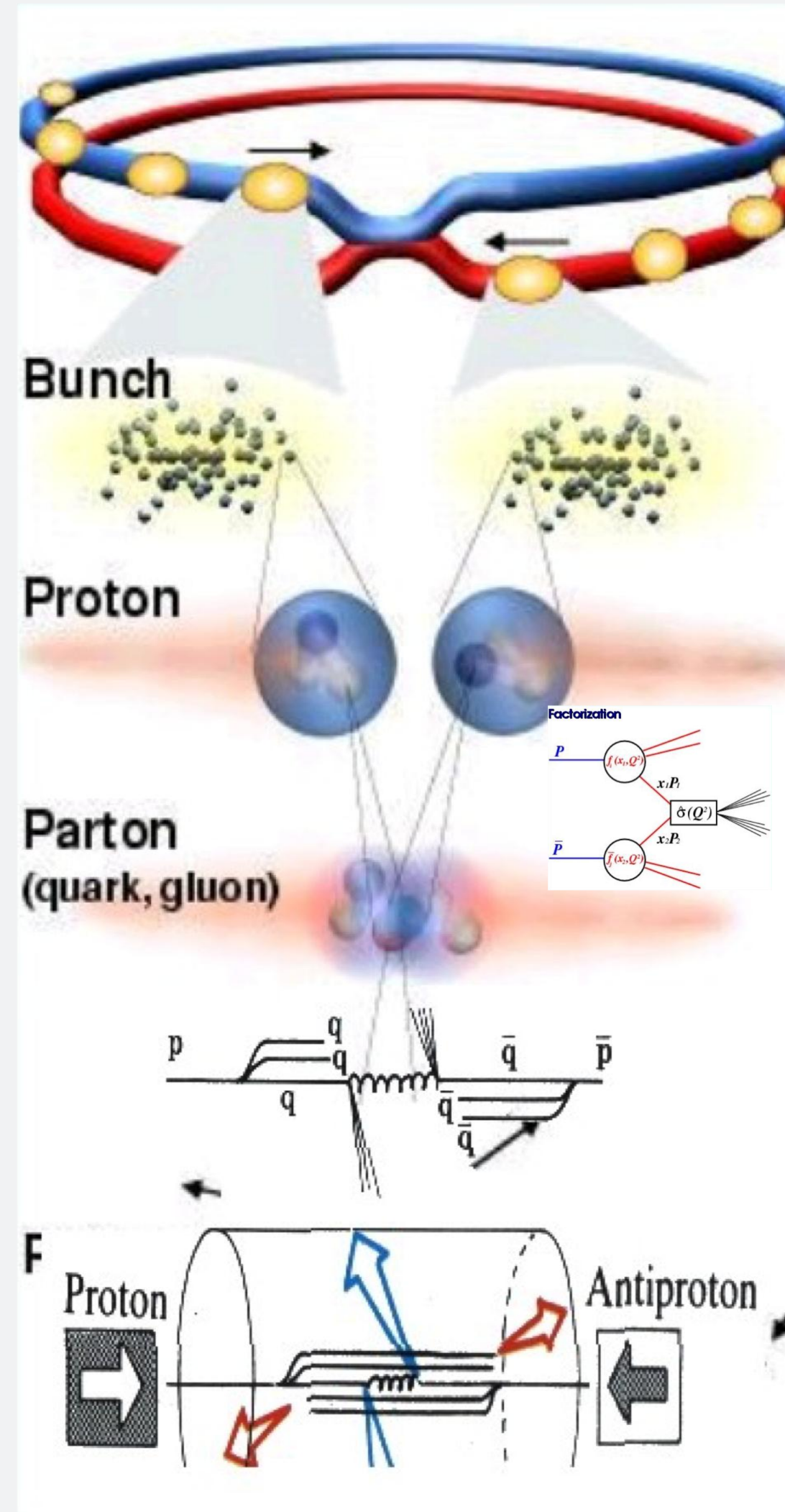


Cable management is still a thing.

Final stage - HLT Farm



LHC and its products



2835×2835
bunches
in the LHC ring

10^{11} protons / bunch

≤ 30 pp collisions
per bunch crossing
(BC)

N parton-parton
collisions / pp
collision

Complex final-states
in every parton-
parton collision.

Design
parameters

$E_{\text{cms}} = 14 \text{ TeV}$
 $L = 10^{34} / \text{cm}^2 \text{ s}$
BC clock = 40 MHz

$$R = \sigma_{in} \times L$$

- Interesting processes **extremely rare**, high Luminosity is essential
 - Close collisions in space and time
 - Large proton bunches (1.5×10^{11})
 - Fixed frequency: 40MHz (1/25ns)
- Protons are **composite particles**
 - abundant low energy interactions
- Few rare high-E events overwhelmed by abundant low-E environment

LHC Detector Challenges

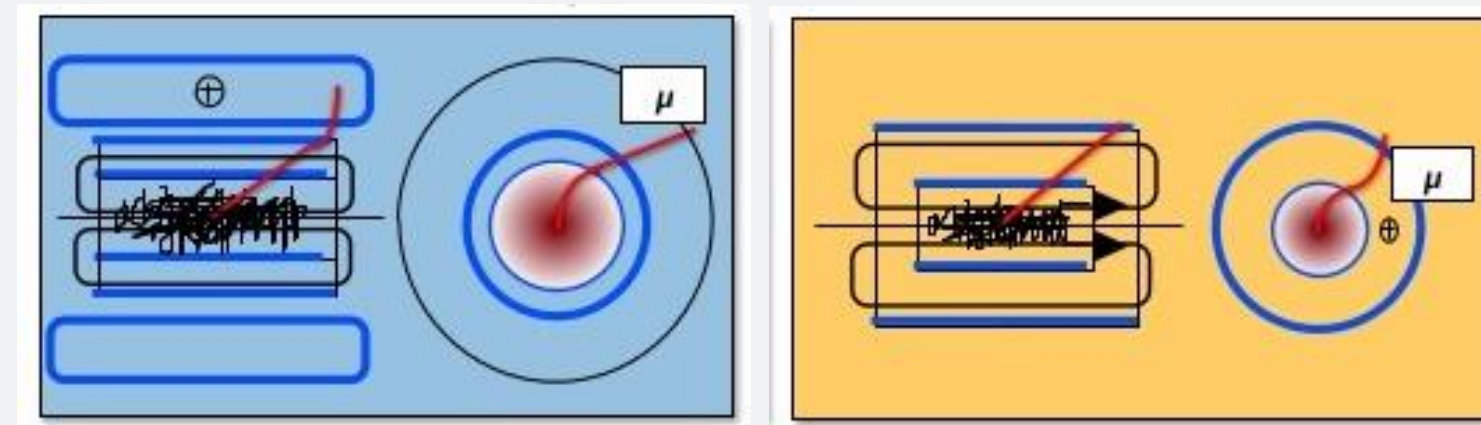
- Huge
 - $O(10^6-10^8)$ channels
 - ~1 MB event size for pp collisions
 - 50 MB for pb-pb collisions (Alice)
 - Need huge number of connections
- Fast and slow detectors
 - Some detector readout requires >25 ns and integrates more than one bunch crossing's worth of information
 - e.g. ATLAS LArg readout takes ~400 ns
- Online, what is lost is lost forever
 - Need to monitor selection - need very good control over all conditions



ATLAS & CMS design principles

- Same physics program

- Different magnetic field structure
 - **ATLAS**: 2 T solenoid + Toroids
 - **CMS**: strong 4 T solenoid

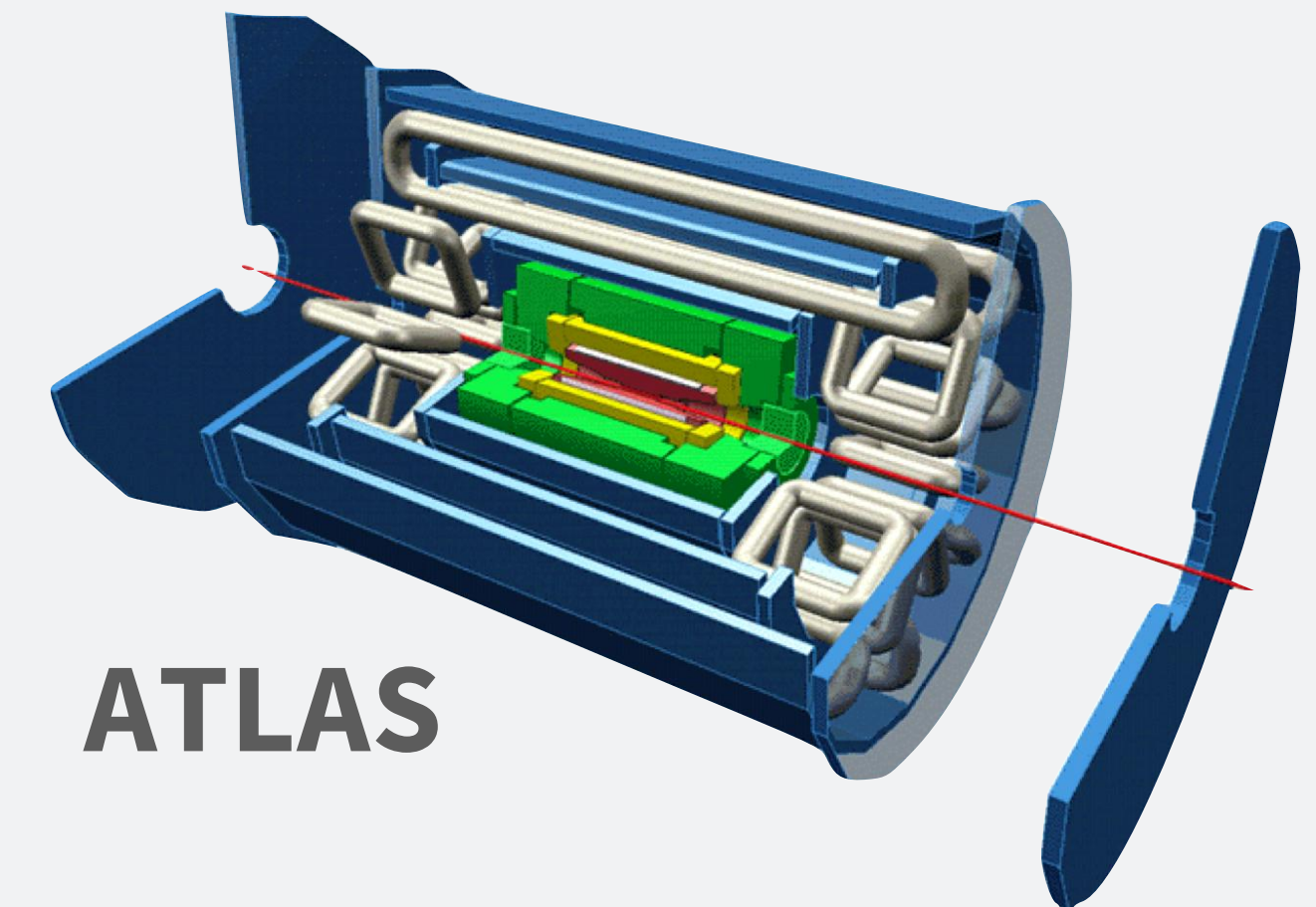


- Different DAQ architecture

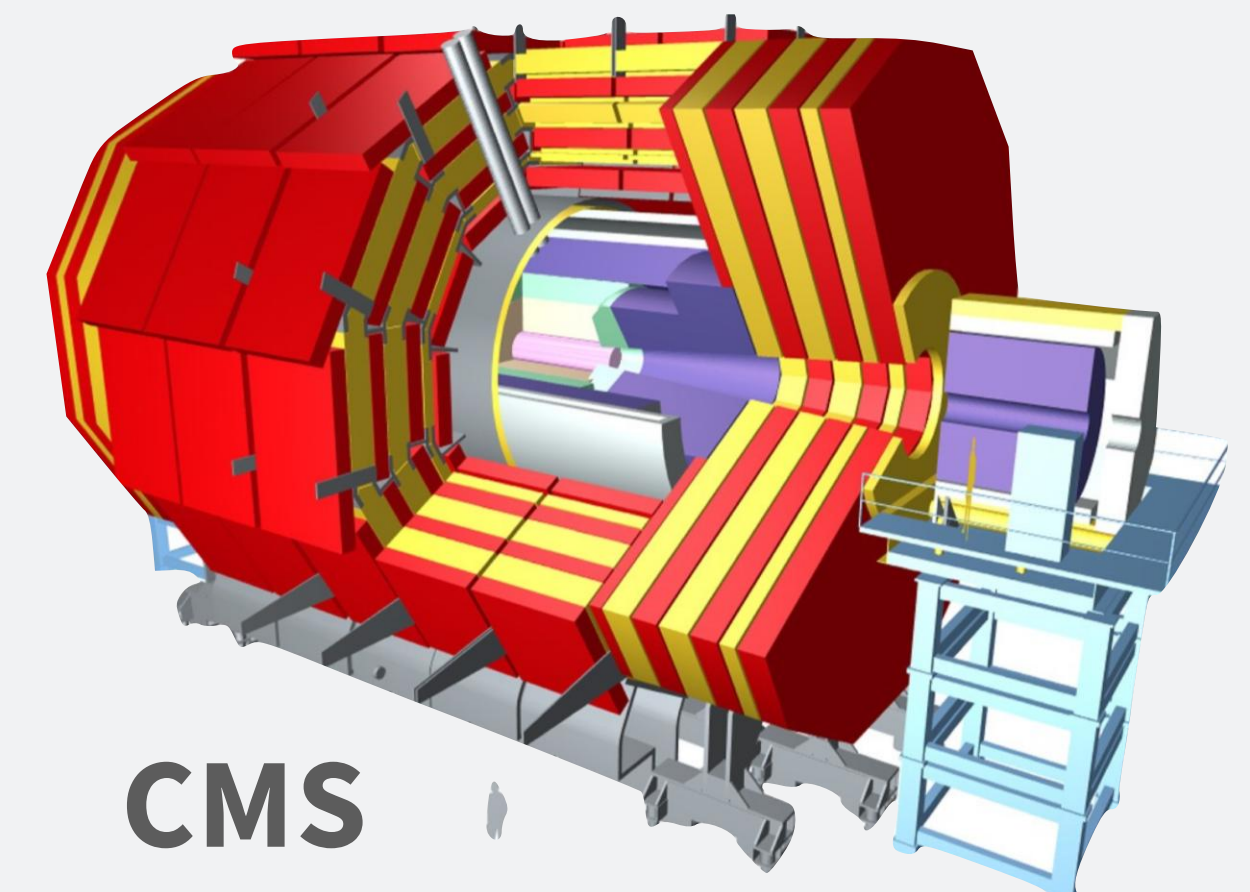
- **ATLAS**: minimise data flow bandwidth with multiple levels and regional readout
- **CMS**: large bandwidth, invest on commercial technologies for processing and communication

- Same data rates

- $\sim 1 \text{ MB} * 100 \text{ kHz} = \sim 100 \text{ GB/s}$ readout network

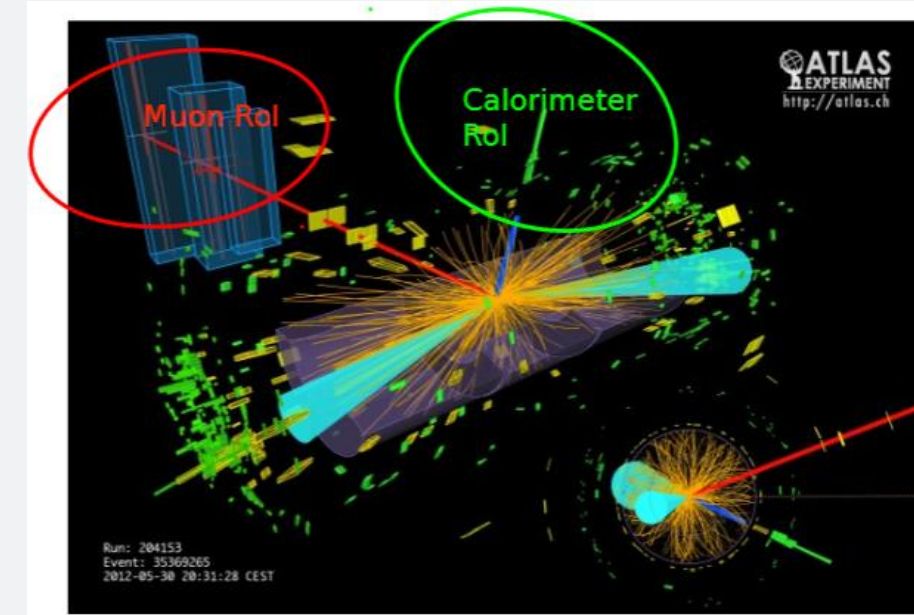


ATLAS

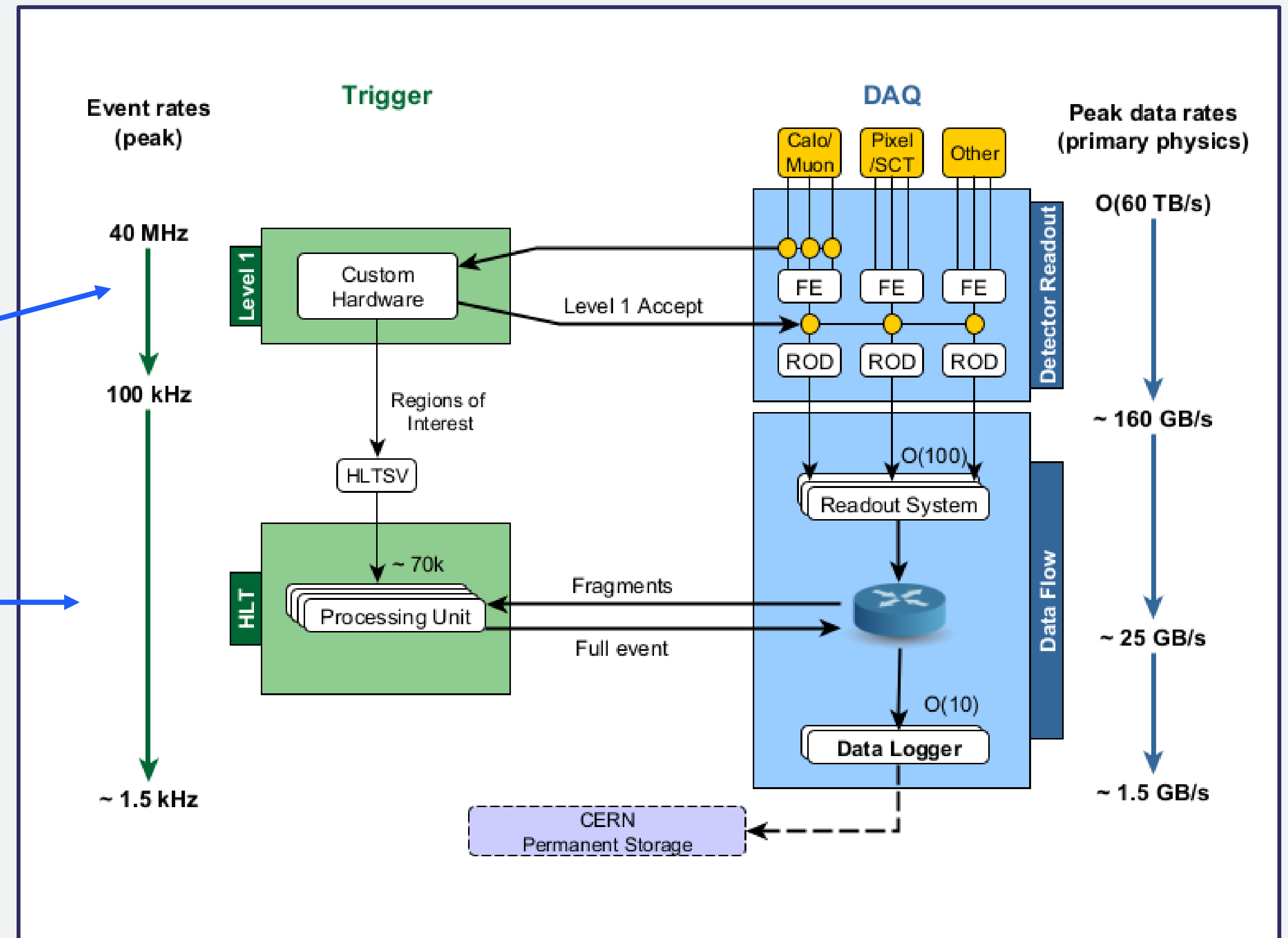


CMS

ATLAS TDAQ System in LHC Run 2



- Trigger decision distributed over two steps, each rejecting the vast majority of events
 - Seeding based on Region-of-Interest (RoI)
- L1 Trigger based on custom hardware (ASICs, FPGAs), with 2.5 μ s latency
 - Data buffered on FE electronics
- HLT based on software running on commodity servers, with approaching 1 s latency
 - Data buffered in readout system
 - HLT samples only needed portions of event



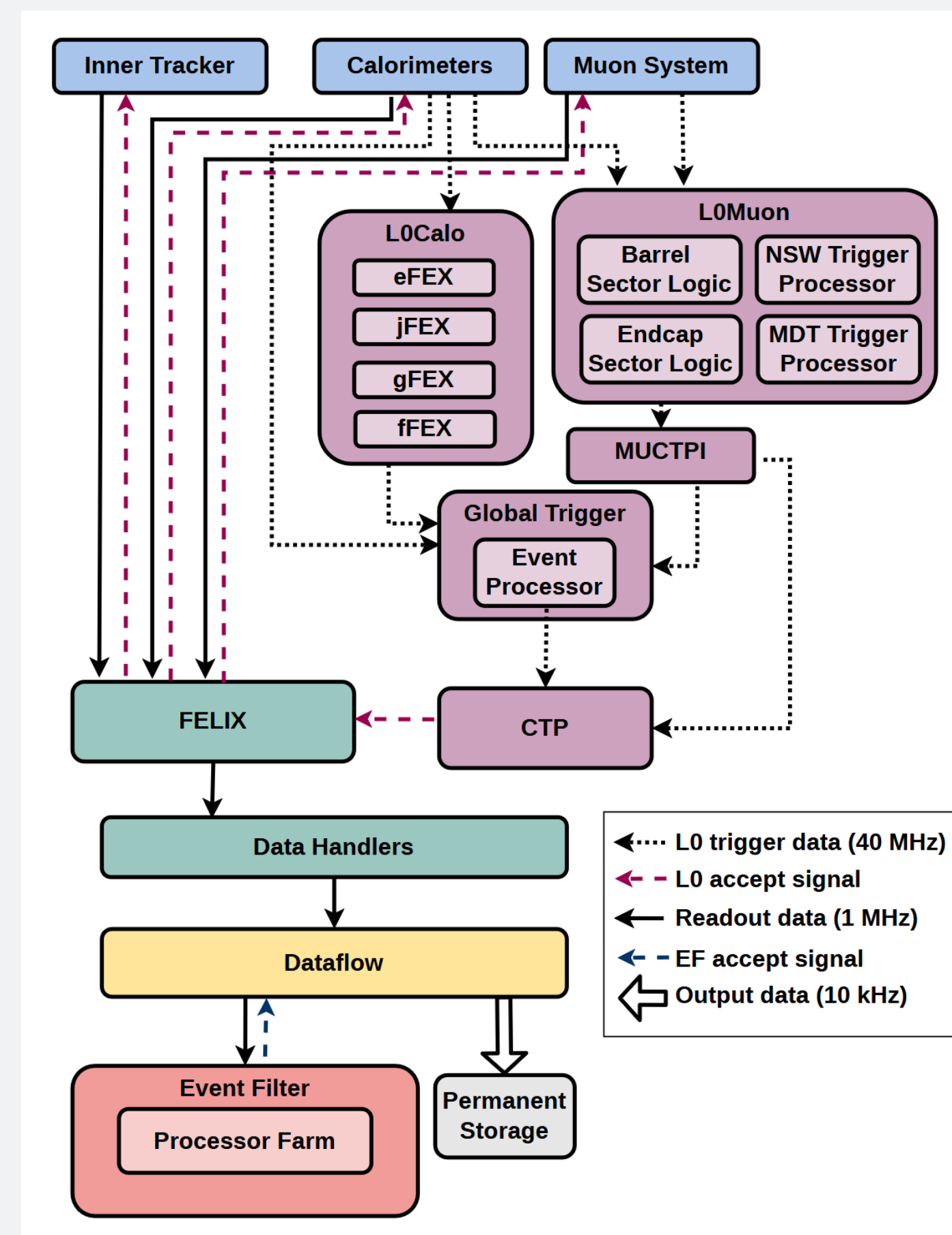
ATLAS TDAQ Upgrade for Run 4

- Still an RoI-based system
- L0 Trigger
 - Hardware trigger for low latency event selection
 - Operating at 1 MHz accept rate
 - 10 μ s latency budget
- DAQ
 - Readout system based new commodity server-based components (FELIX)
 - Serving data from custom detector links to commodity switched network
 - Already introduced for some systems in Run 3
 - Completely new Dataflow system to transport data to Event Filter and permanent storage
 - 4.6 TB/s data rate through readout and dataflow to event filter
 - All events fully built before Event Filters
 - Overhauled network and online software
- Event Filter
 - High level event selection in dedicated farm (largely equivalent to today's HLT)
 - Will feature GPU accelerators

DAQ

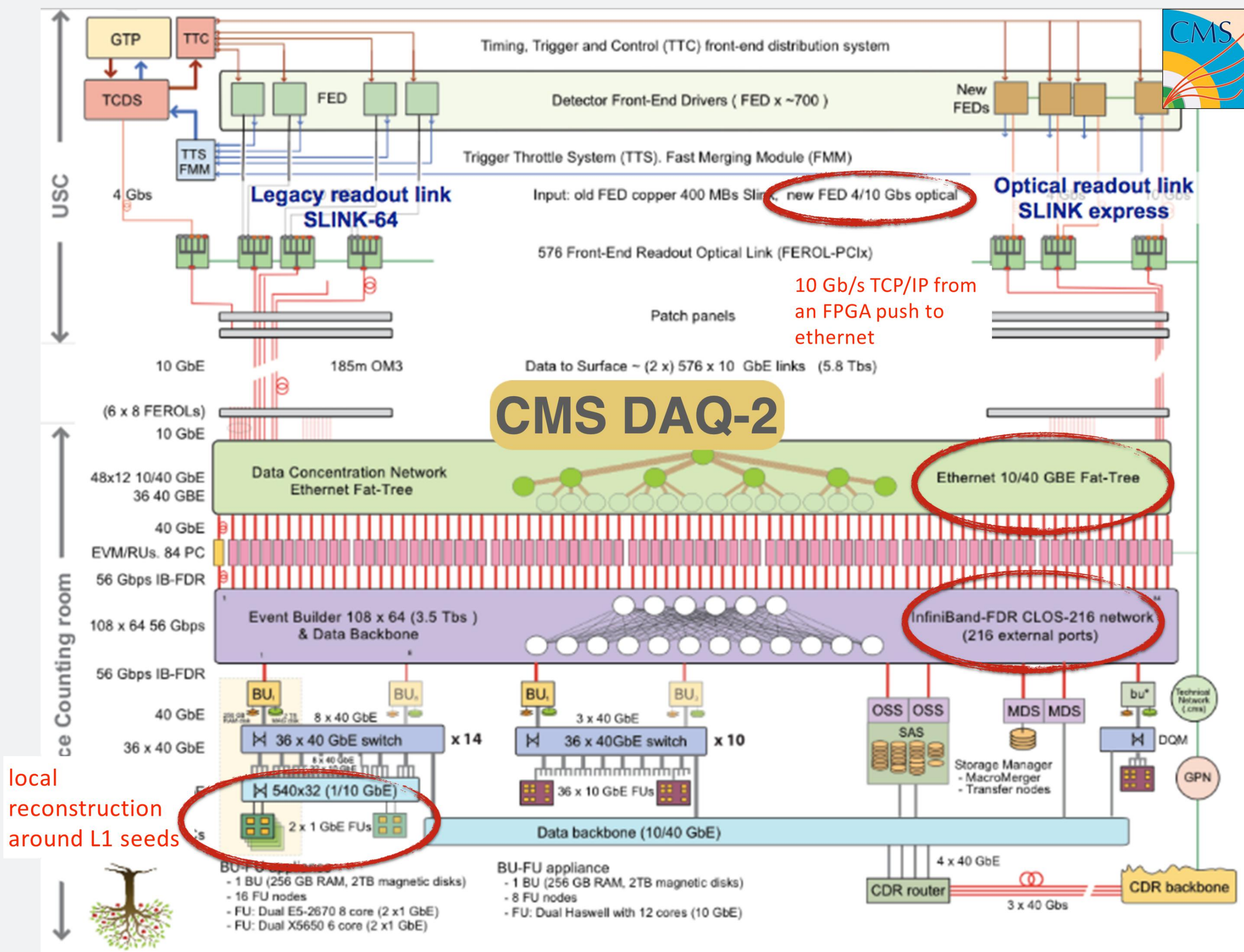
EF

L0



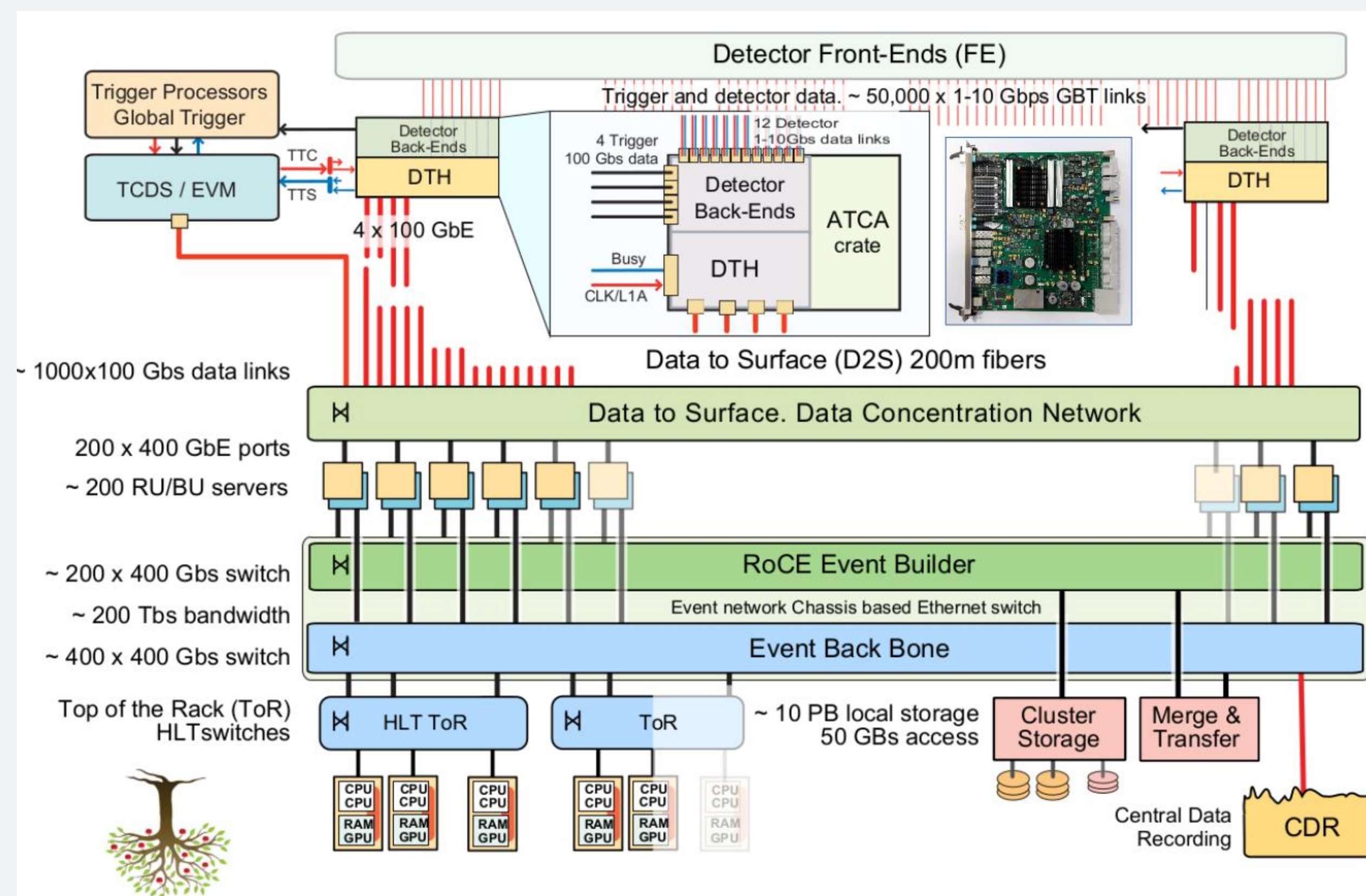
CMS DAQ System in LHC Run 2

- Two-stage Trigger System
 - L1 @ 100 kHz, 4 μ s latency
 - ASICs, FPGAs
 - HLT @ 1 kHz
 - Commodity Servers (incorporated GPUs for Run 3)
- Fully build all events by default
 - Requires large concentrator and event building network
- File-based HLT processing (full events)
 - DAQ and HLT can be completely decoupled



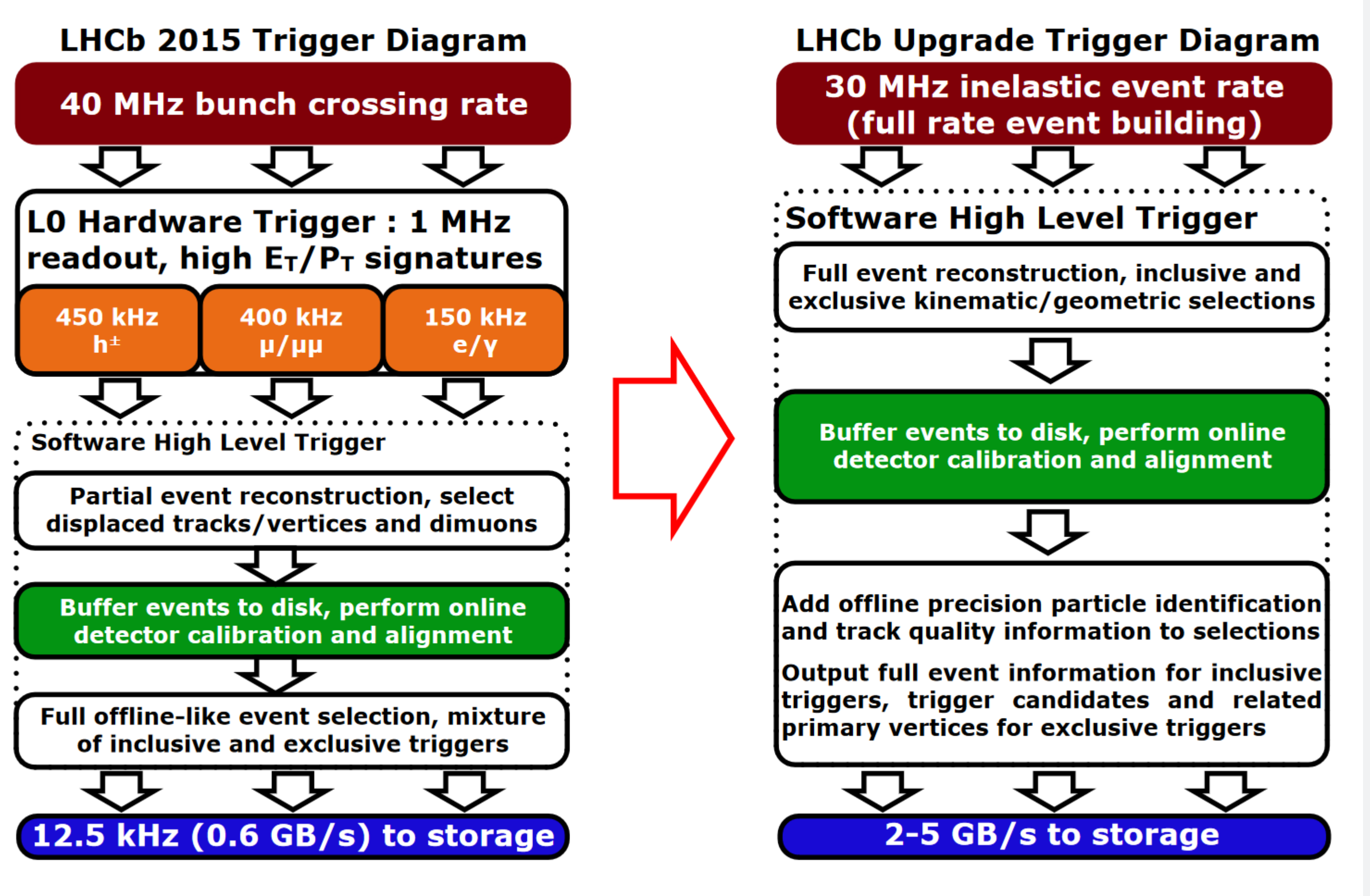
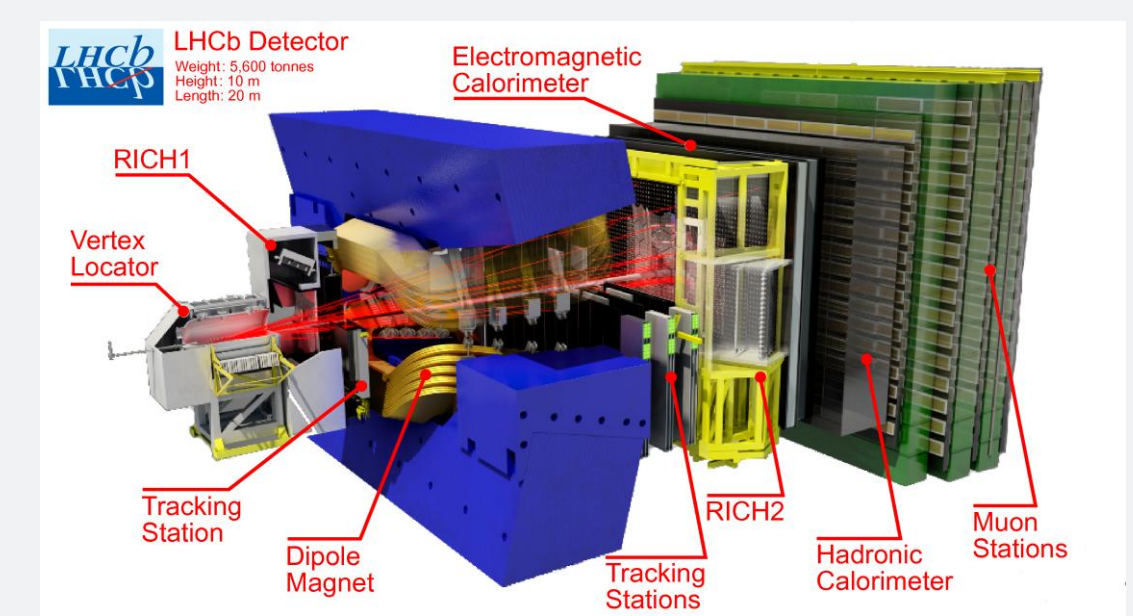
CMS DAQ Upgrade for Run 4

- L1 Trigger operating at 750 kHz, incorporating tracking
 - 12.5 μ s latency
- HLT operating at 7.5 kHz
- Dataflow throughput of 5.5 TB/s
 - Significantly higher network capacity needed
- Data-to-surface (D2S) boards bridging custom detector links to switched network



LHCb Trigger Upgrade

- From Run 3 (2022), LHCb has essentially done away with its hardware trigger
- Instead, all detector data are read out directly into High Level Trigger (software), which features GPU-based acceleration
- Possible due to (relatively) small event size (approx. 10% of CMS/ATLAS), but involves very complex reconstruction at high rates in HLT



ALICE Trigger

- Run 1 and 2 Trigger
- 3 hardware levels
- 1 software

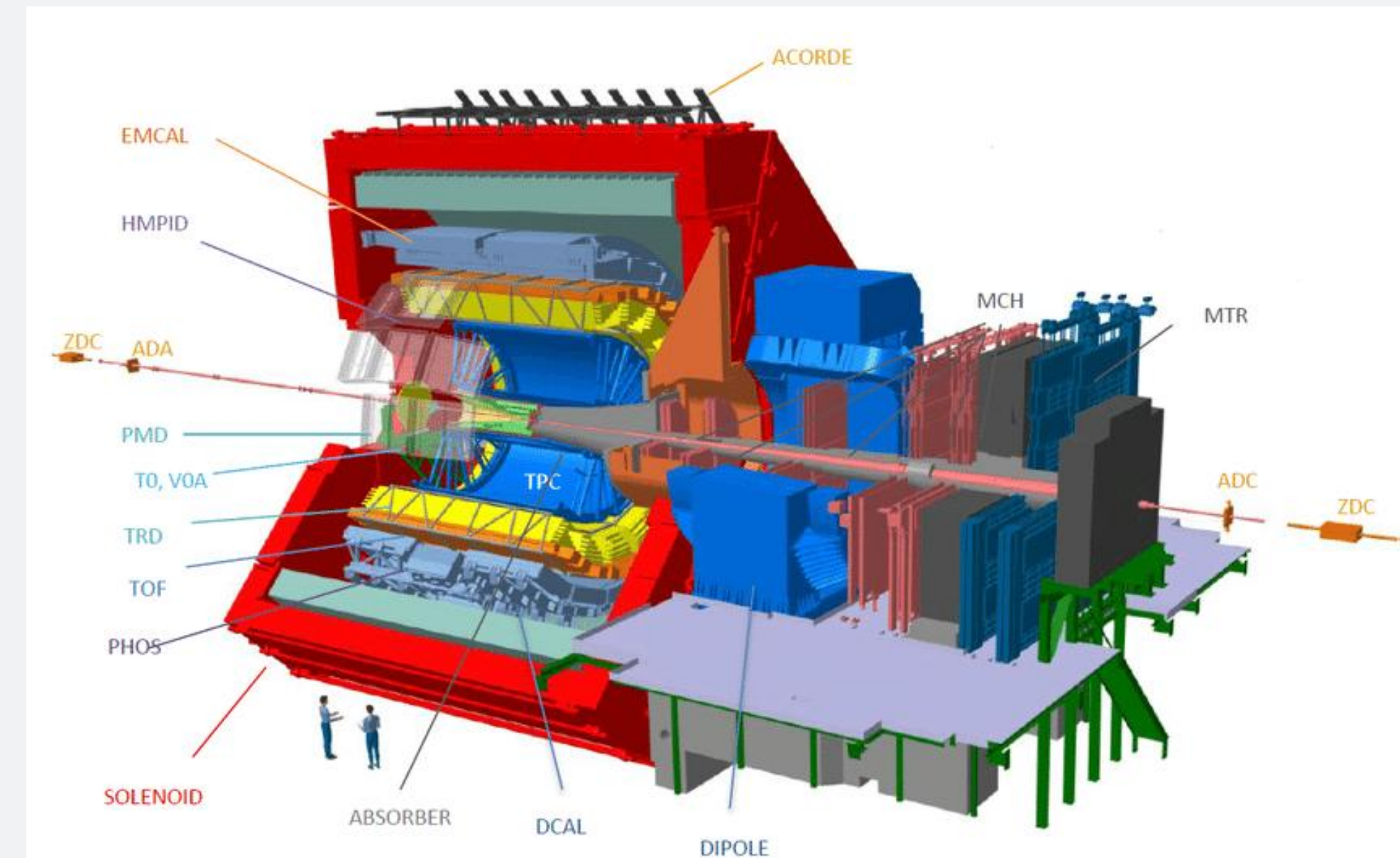
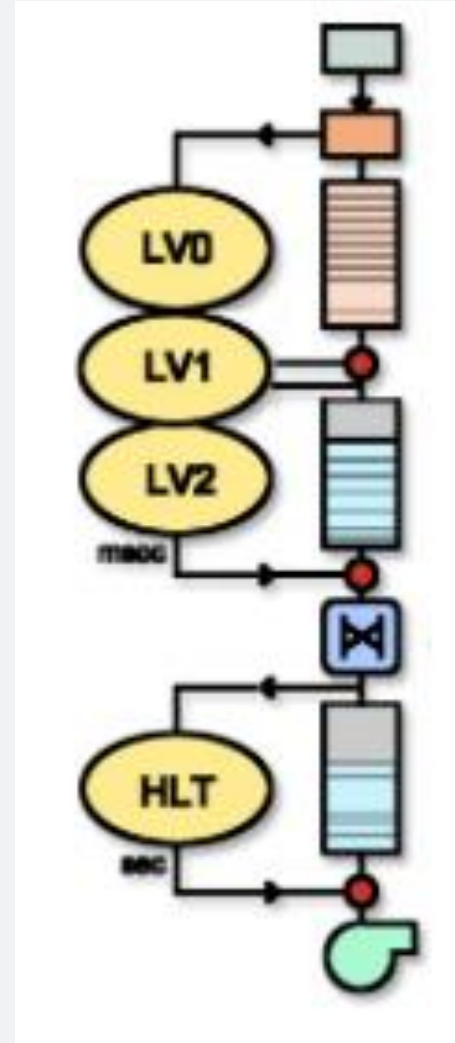
Latency

1.2 μ s

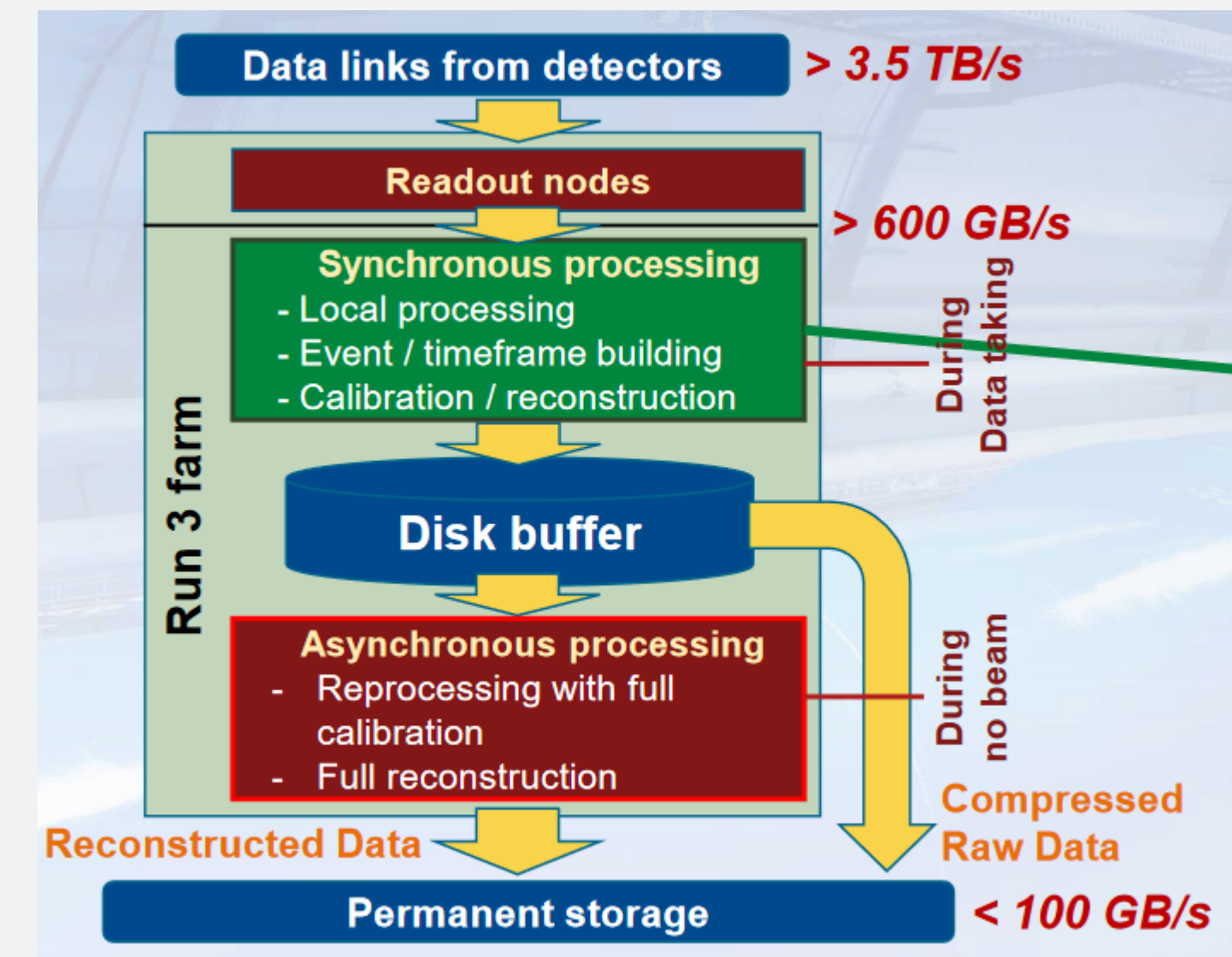
6.5 μ s

100 μ s

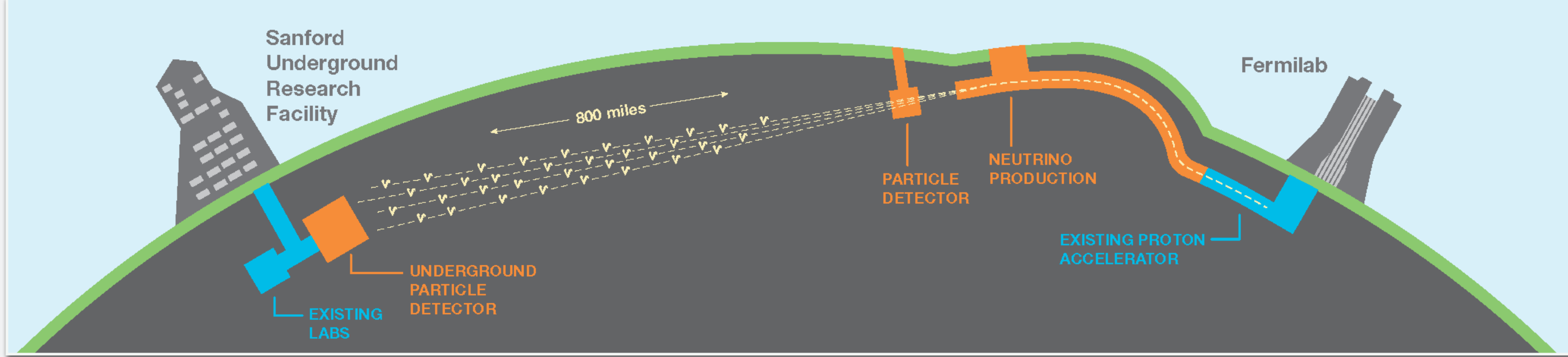
Software



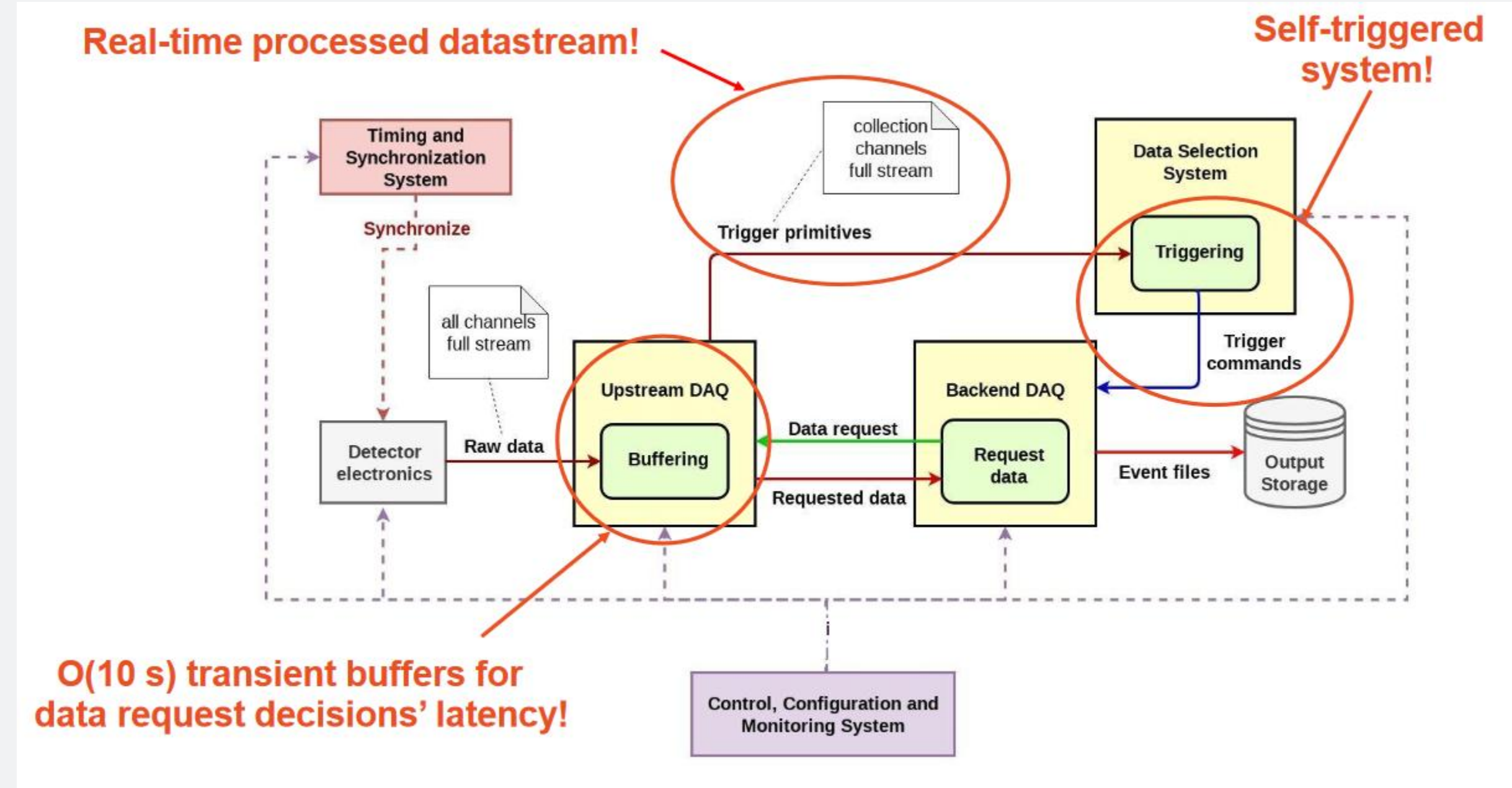
- In Run 3, moved to integrated online-offline 'O²' system with continuous readout
- Introduction of new Fast Interaction Trigger (FIT) minimum bias trigger with high precision timing and luminometry
- Making use of GPU-accelerated reconstruction



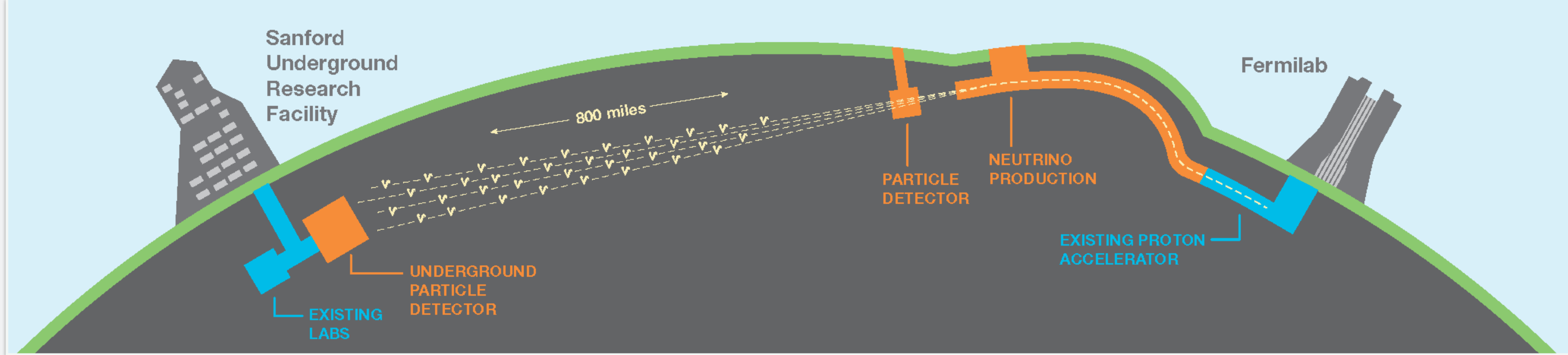
DUNE TDAQ System



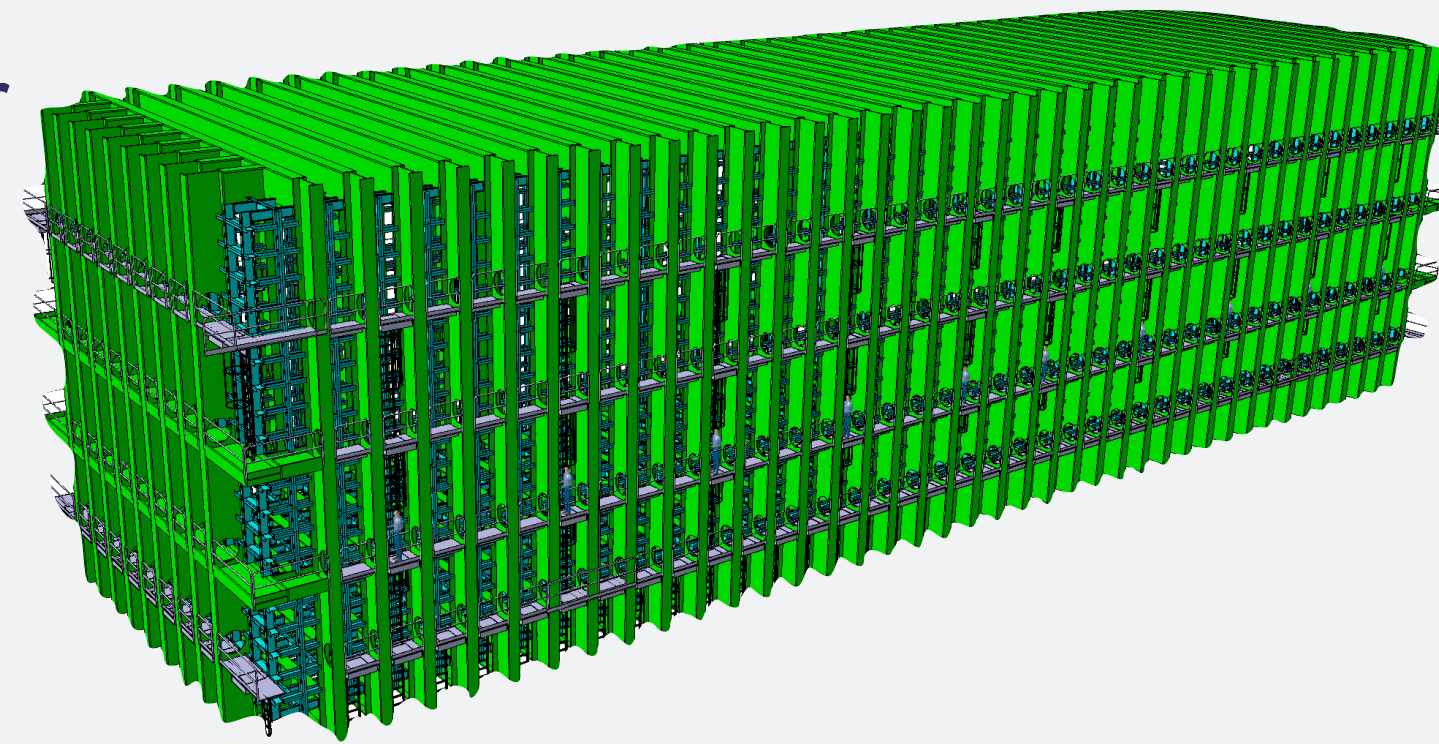
- Single trigger level with two modes of operation:
 - Interaction triggers (interesting localized activity somewhere in the detector)
 - Examples: beam triggers, cosmic rays and photon detection.
 - Supernova Neutrino Burst (SNB) triggers (sufficient activity in the detector to suggest a SNB)
 - All data are stored for 100 sec window including $O(10\text{ s})$ before the trigger signal.
 - External Trigger Interface (ETI) to pass messages to Supernova Early Warning System (SNEWS)



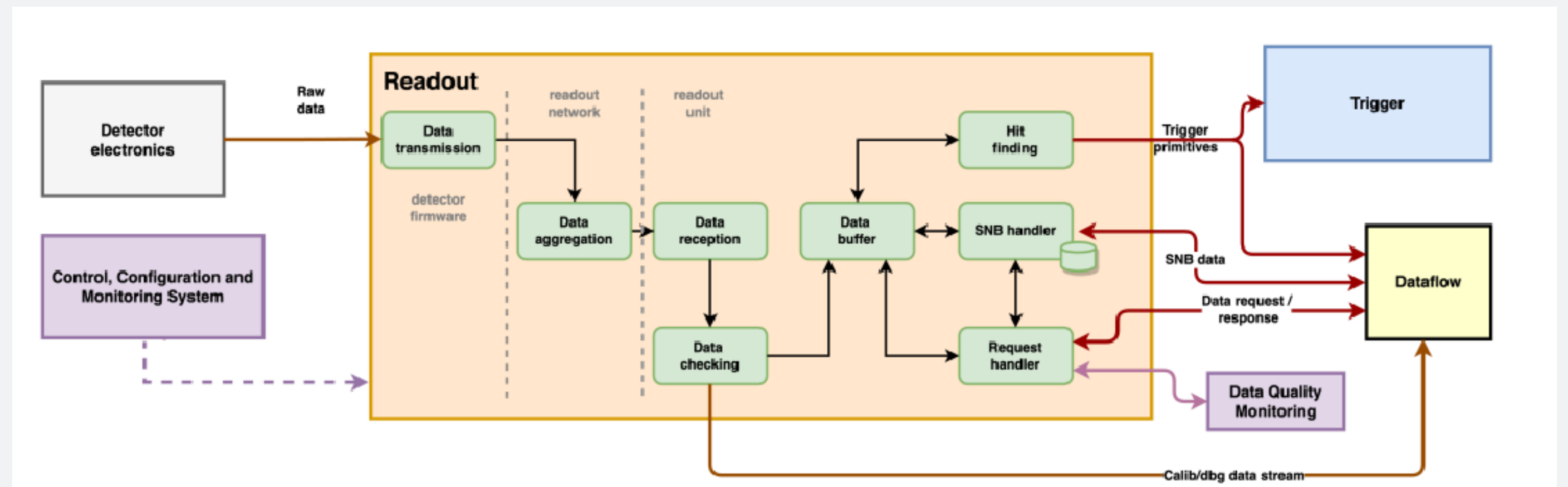
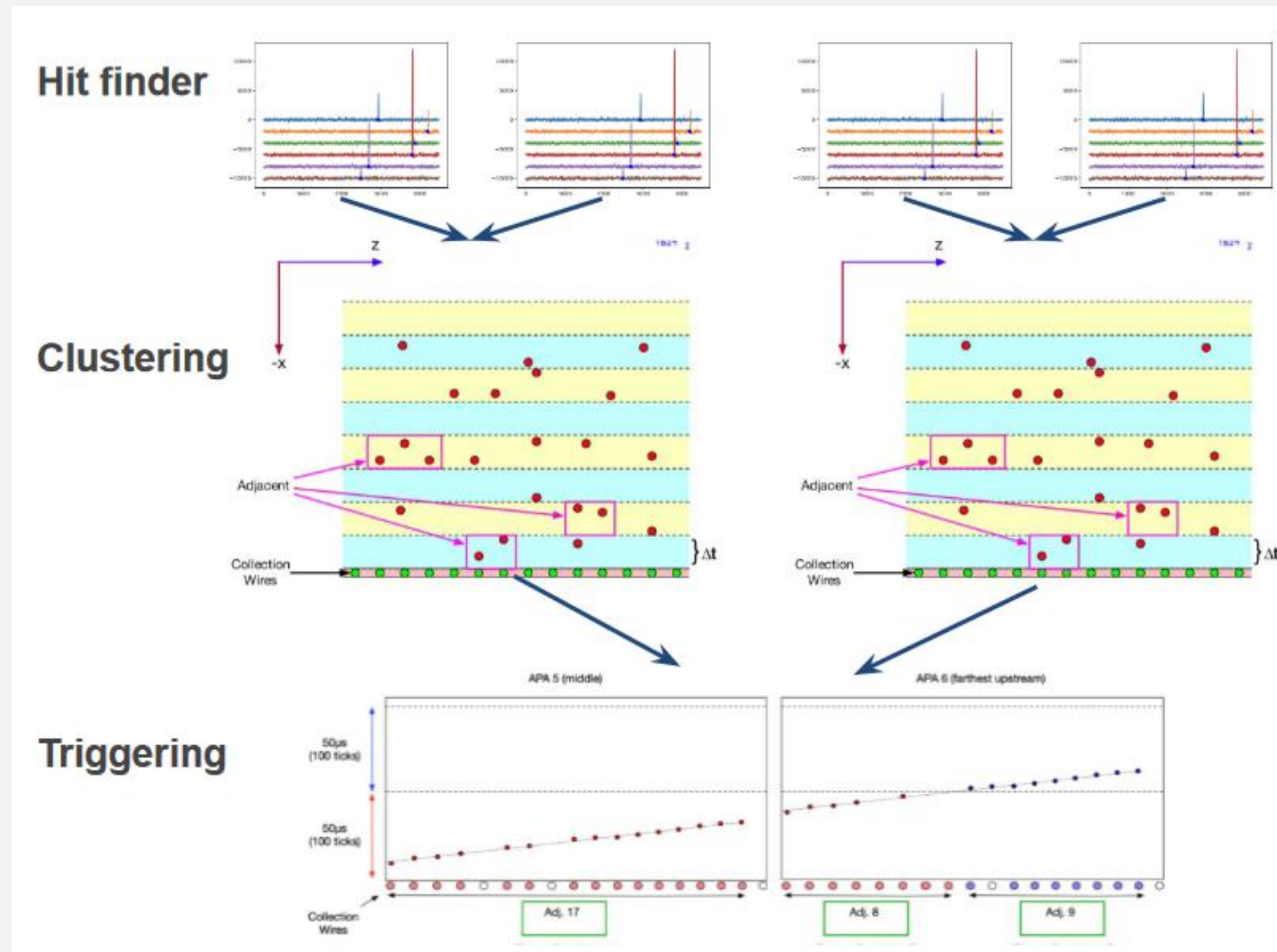
DUNE TDAQ System



- Software-based (SIMD) hit finding to produce Trigger Primitives, followed by clustering and selection



DUNE Far Detector comprised of four 17 kT Liquid Argon Time Projection Chamber (LARTPC) modules



Summary

- This lecture is just an introduction about data acquisition
 - DAQ (& Trigger) is a complex and fascinating topic, combining very different fields of expertise
- Covered the principles of a simple data acquisition system
 - Basic elements: trigger, derandomiser, FIFO, busy logic
 - Scaling to multi-channel, multi-layer systems
 - How data are transported
 - Bus versus network
- A (very) brief overview of LHC experiments + DUNE DAQ systems
 - Similar architectures, different optimisations driven by detector requirements



Science and
Technology
Facilities Council

Particle Physics

The background features a large, abstract geometric design. It includes a dark blue square on the right side, a blue triangle pointing downwards on the left side, and a series of blue lines forming a jagged, zig-zag pattern that separates the dark blue square from the orange background. The word 'Questions?' is written in white, bold, sans-serif font across the center of the image, overlapping the dark blue square and the orange background.

Questions?



Science and
Technology
Facilities Council

Particle Physics

Thank you

Search: STFC Particle Physics

 @PPD_STFC

 @STFC_Matters

 Science and Technology
Facilities Council