Imperial College London

THE ONGOING EVOLUTION OF TRIGGER ELECTRONICS AT CMS

IOP 2019 – Imperial College London

Andrew W. Rose, Imperial College London

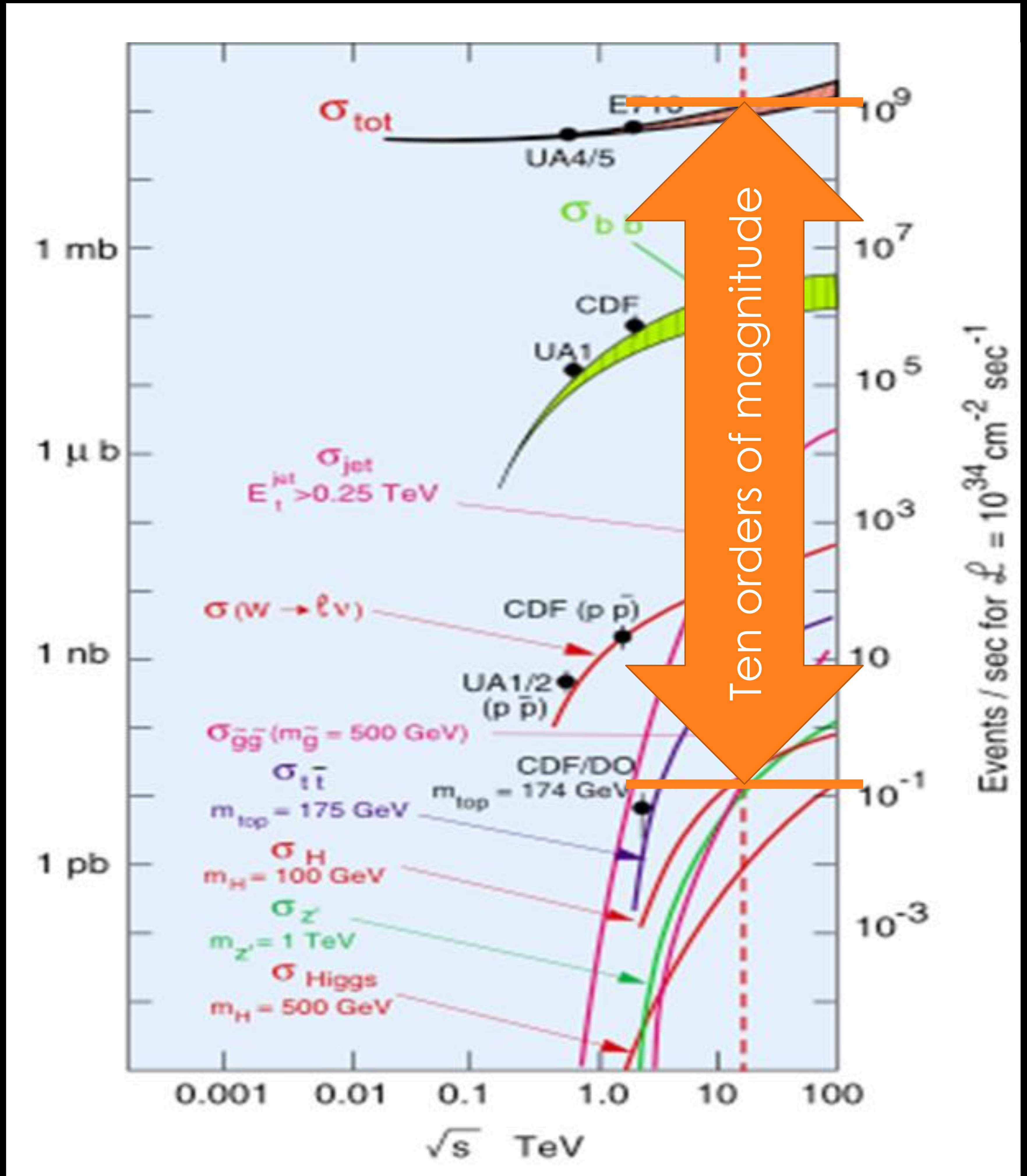
awr01@imperial.ac.uk

SCIENCE: THE BASICS

- Science is the art of knowing what to record, and when

SCIENCE: THE BASICS

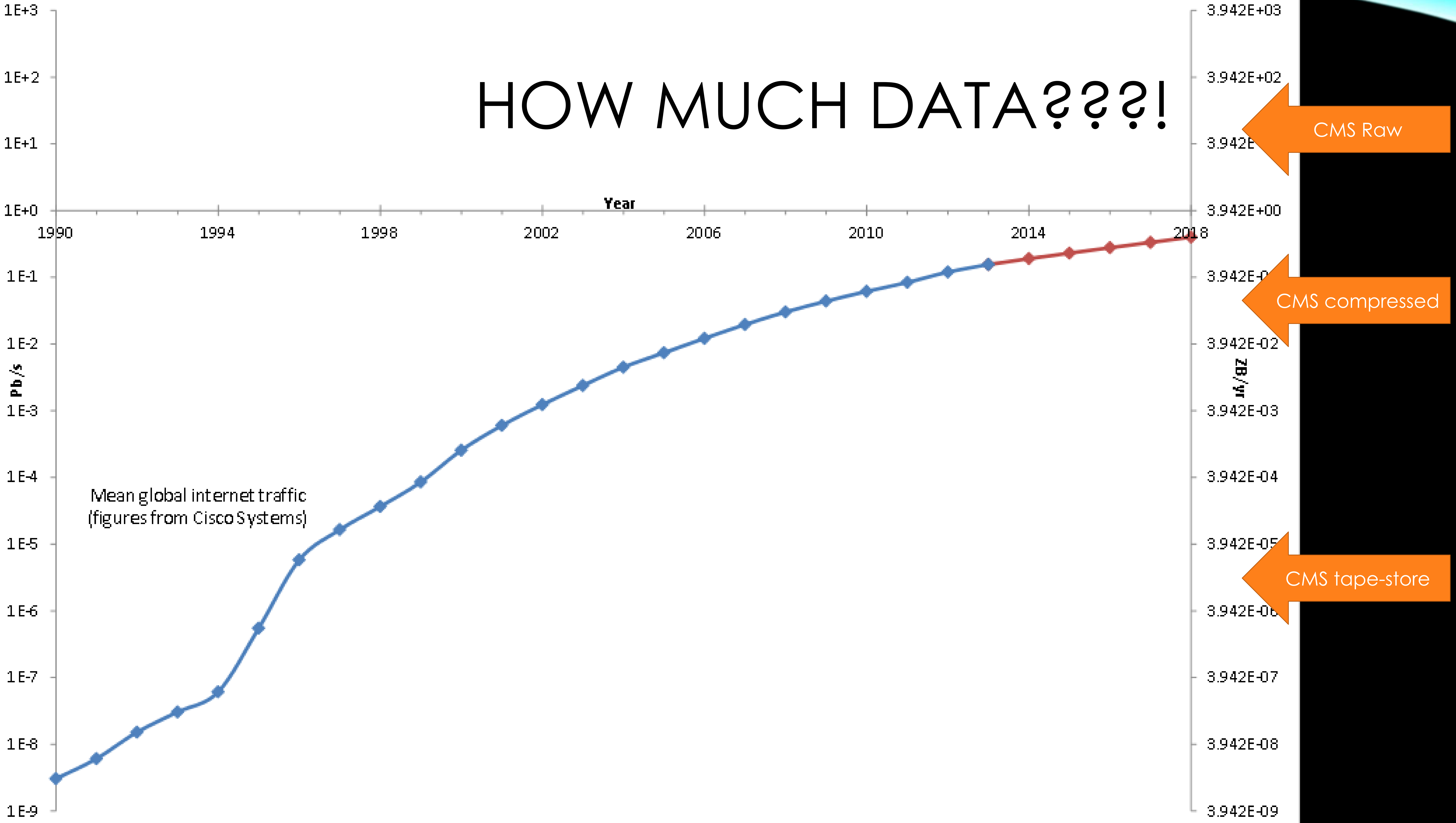
- Science is the art of knowing what to record, and when
- With CMS & ATLAS in “discovery mode”, we care about the Higgs Boson or rarer
  - Higgs Boson production is ten orders of magnitude below the total interaction rate
  - That is a needle in a haystack the same mass as the Empire State Building
- And we want statistics, a lot of statistics



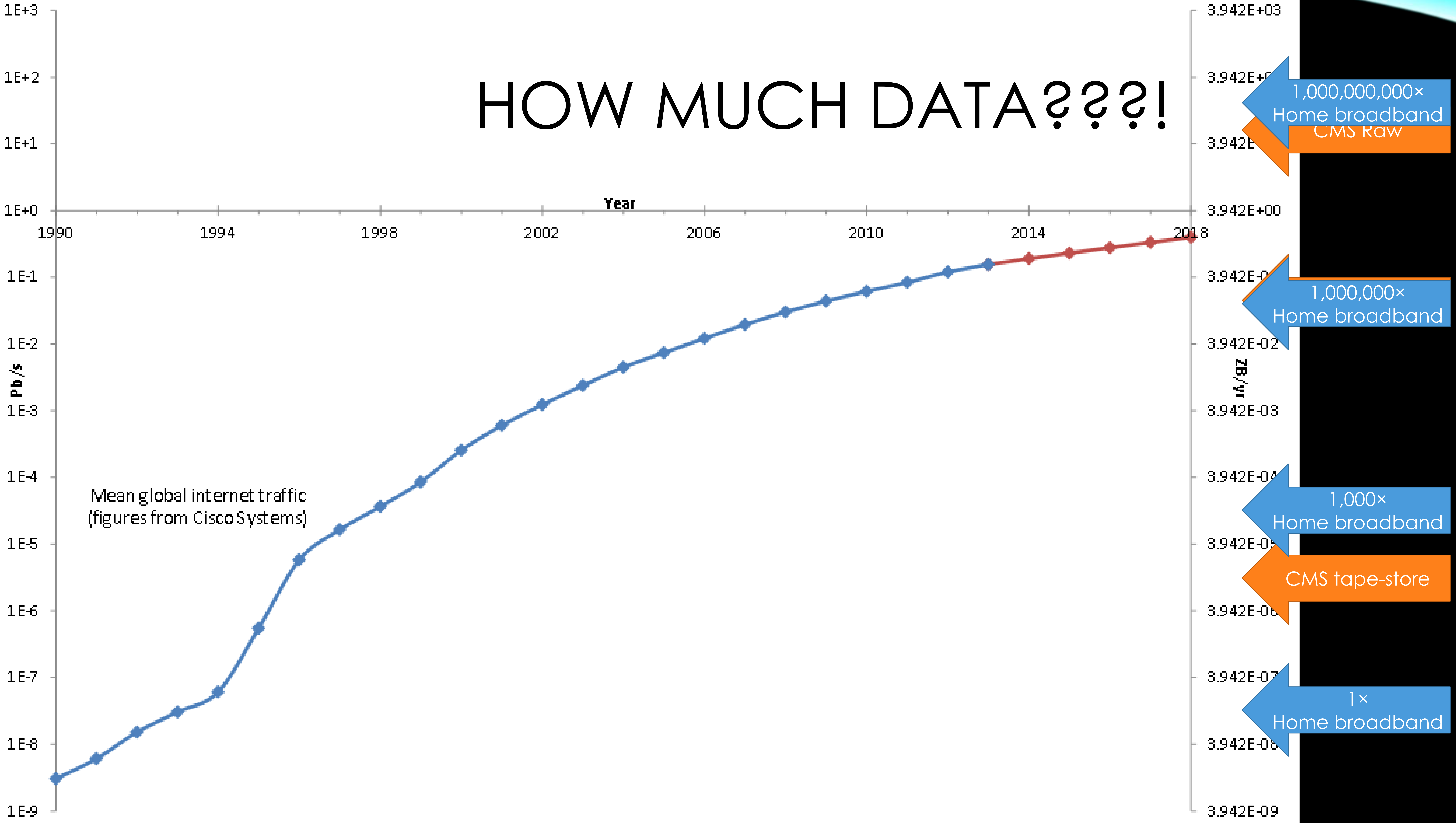
UNFORTUNATELY, STATISTICS REQUIRES DATA

- The LHC's 40MHz crossing rate and 10^{34} cm^{-2} s^{-1} luminosity was chosen to provide 1 billion interactions per second
- Unfortunately, 40MHz on a 70 million channel tracker produces the equivalent of 25Pbit/s of data
- And 10^{34} cm^{-2} s^{-1} luminosity produces ~25 times more background in your detector than signal, making selection tricky
  - And every time the LHC improves its performance, this gets worse

HOW MUCH DATA???!



HOW MUCH DATA???!

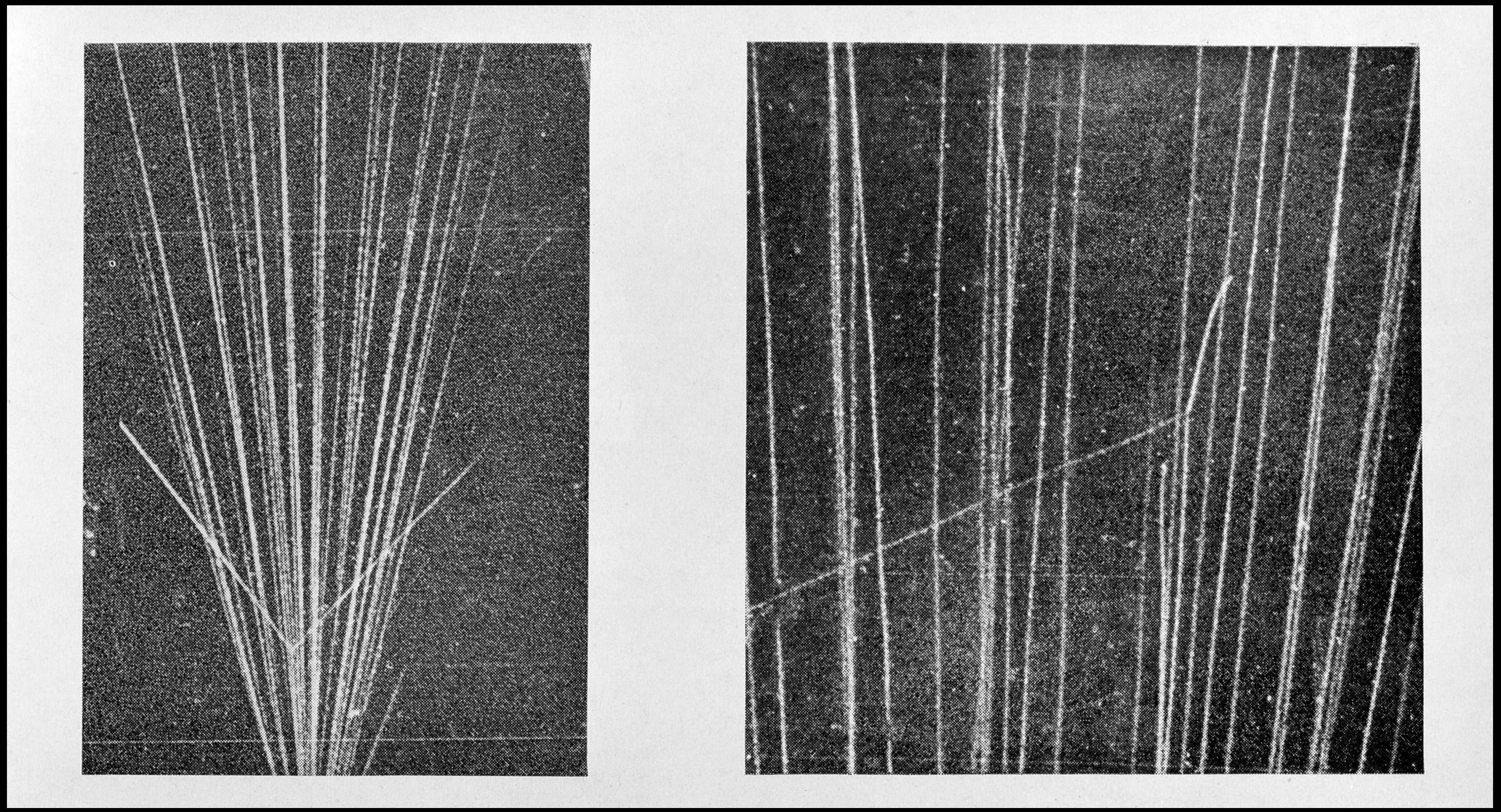


DATA-TAKING REQUIREMENTS

- Basic requirements for data-taking systems
  - Need high efficiency for selecting data for later analysis
  - Need large reduction of rate from unwanted high-rate processes
  - Needs to be robust for consistency of data
  - Must be highly flexible to react to changing conditions
  - Must be affordable

THE EARLIEST TRIGGER

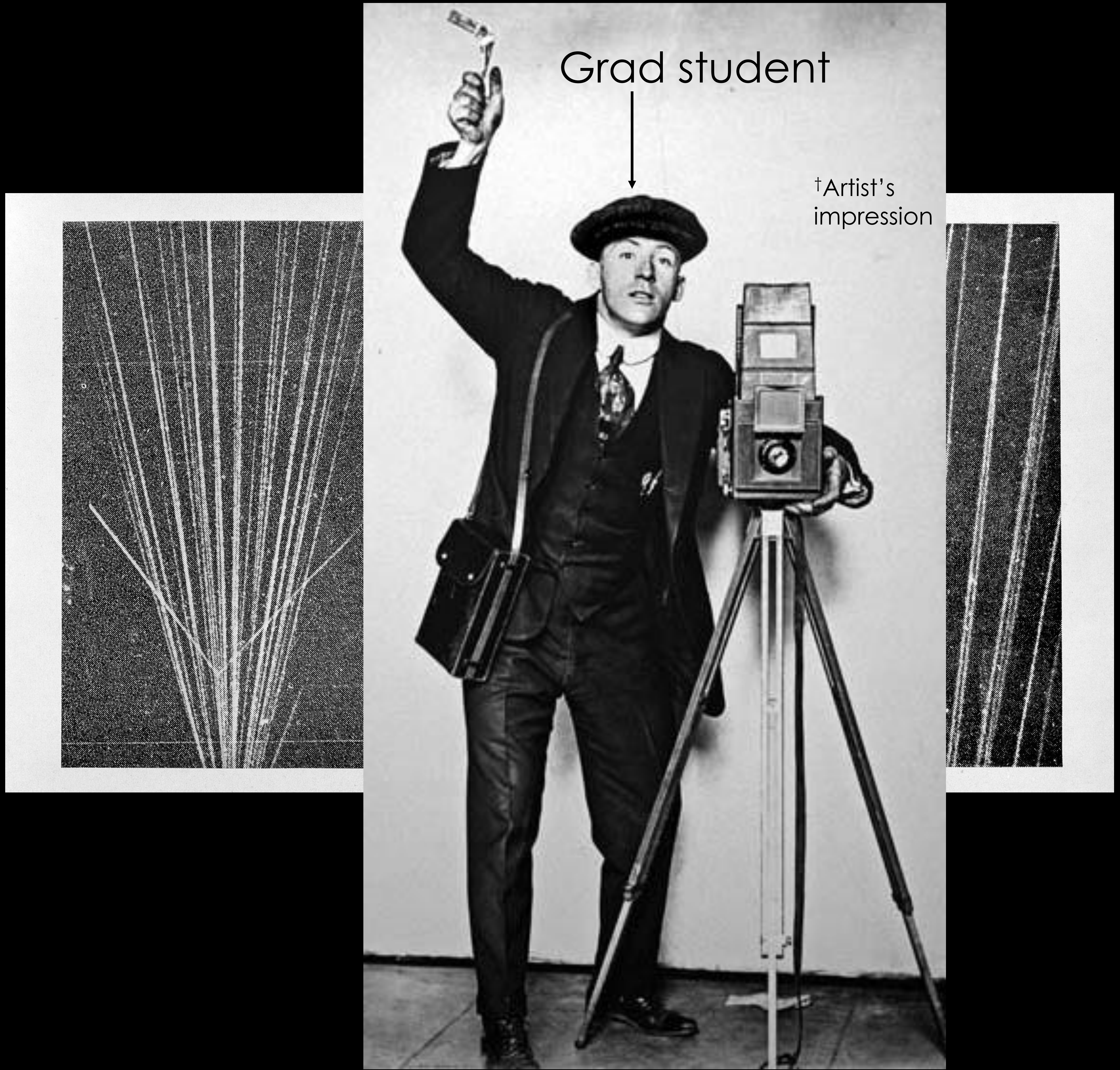
- Cloud-chamber images recorded on film
- Need some way to trigger the camera



THE EARLIEST TRIGGER

- Cloud-chamber images recorded on film
- Need some way to trigger the camera
- Not the best trigger system
  - High efficiency
  - Large rate reduction
  - Robust
  - Highly flexible

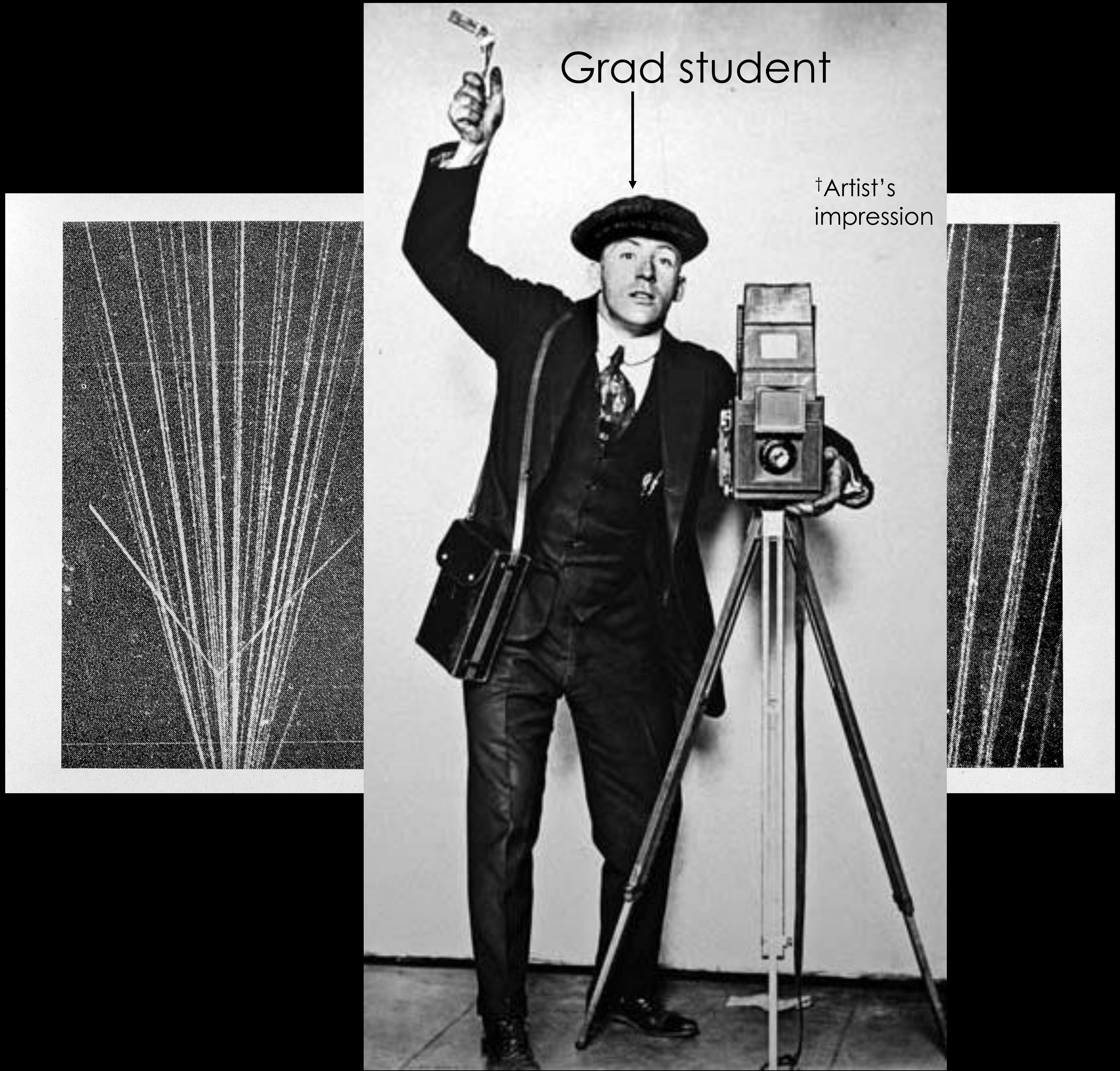
Depends on the student



THE EARLIEST TRIGGER

- Cloud-chamber images recorded on film
- Need some way to trigger the camera
- Not the best trigger system
  - High efficiency
  - Large rate reduction
  - Robust
  - Highly flexible
  - Affordable ✓

Depends on the student (High efficiency, Large rate reduction, Robust, Highly flexible)



THE EARLIEST TRIGGER

- Blackett pioneered a technique to trigger the camera of cloud chambers (and got the Nobel prize for this and other work)
- Just missed out on discovering the positron in 1932
- Stevenson and Street used this to confirm the discovery of the muon in 1937

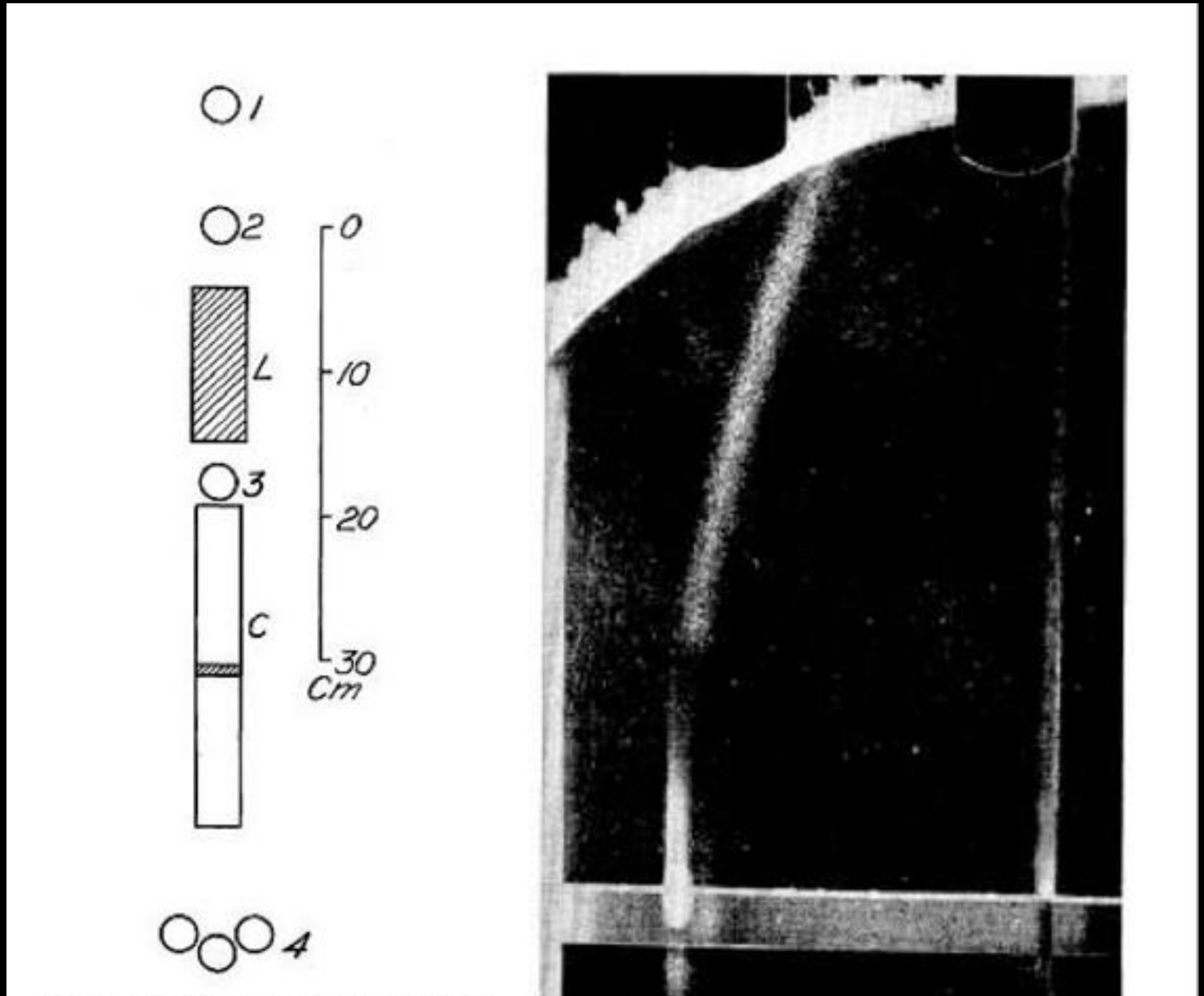
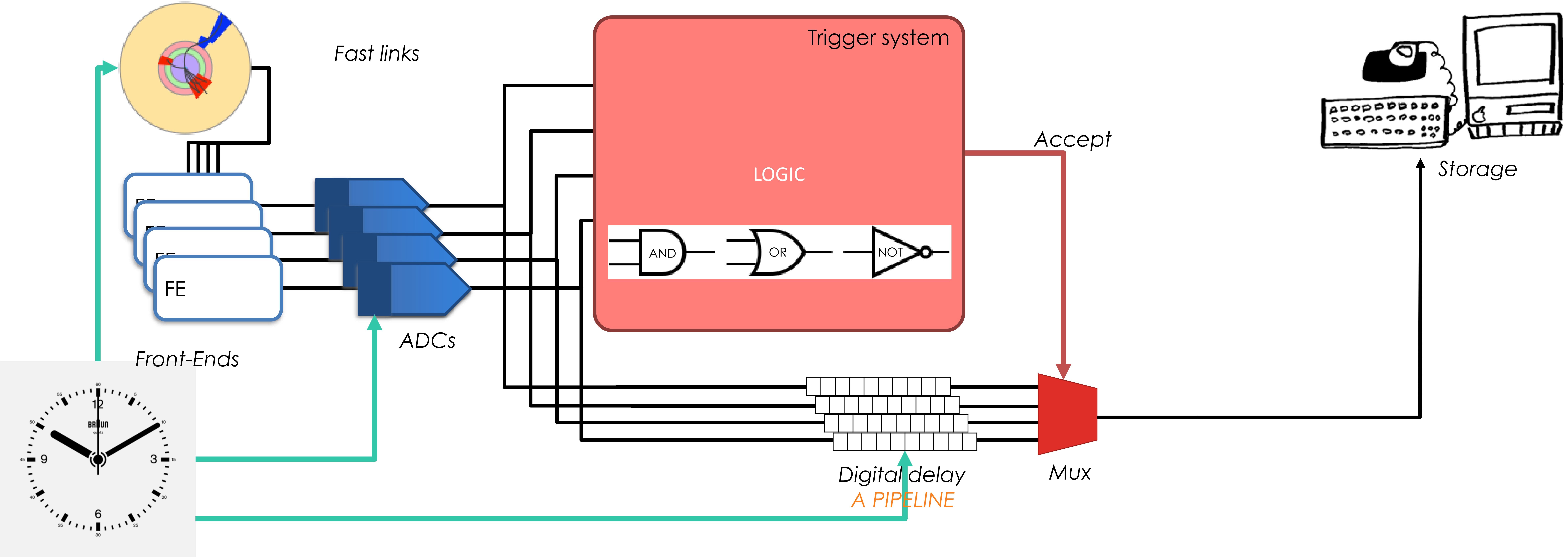
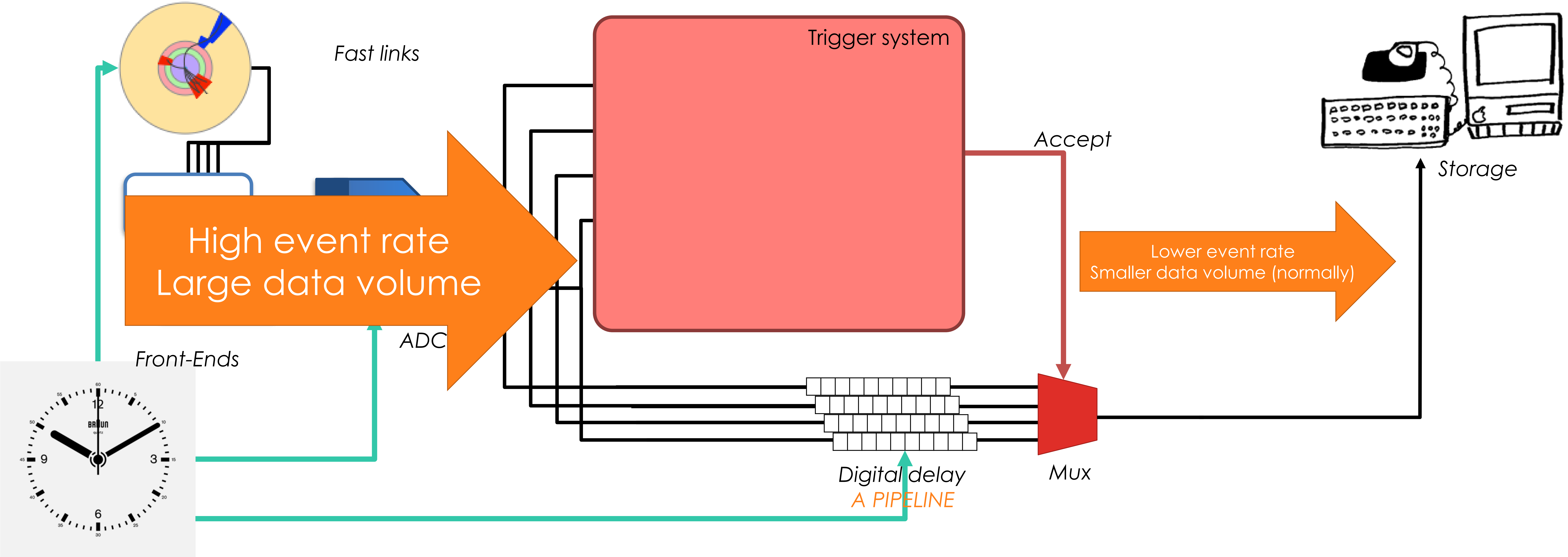


FIG. 1. Geometrical arrangement of apparatus.

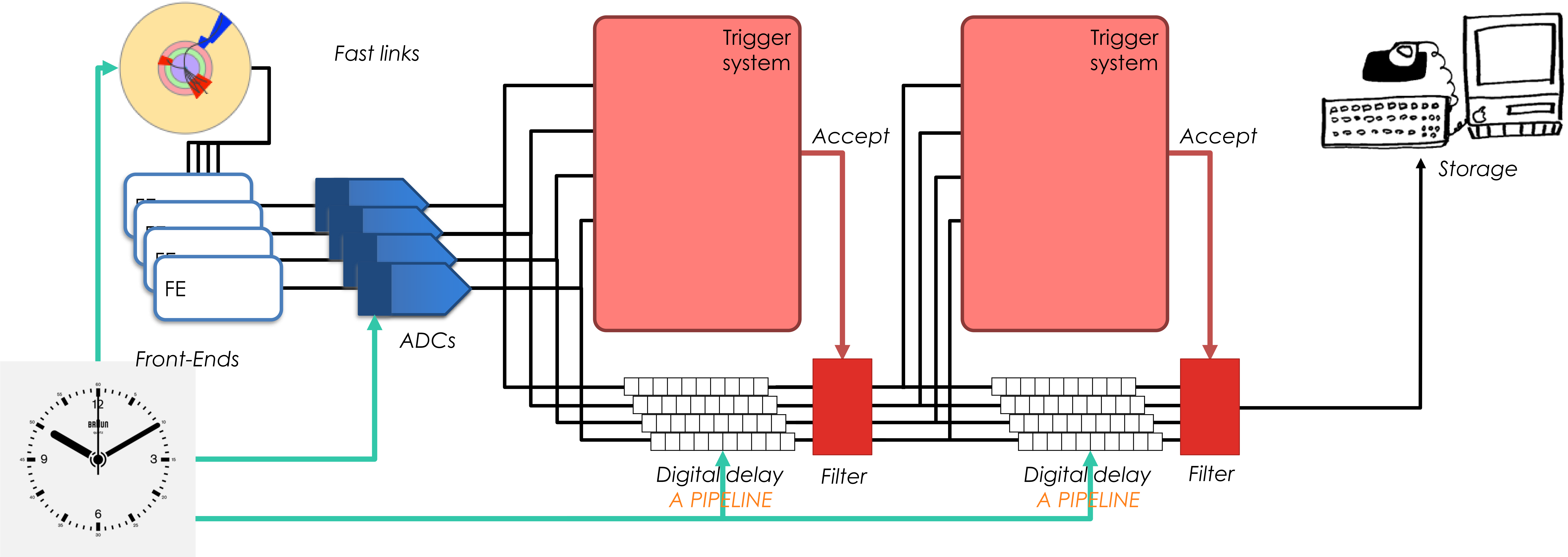
A SIMPLE MODERN TRIGGER SYSTEM



A SIMPLE MODERN TRIGGER SYSTEM

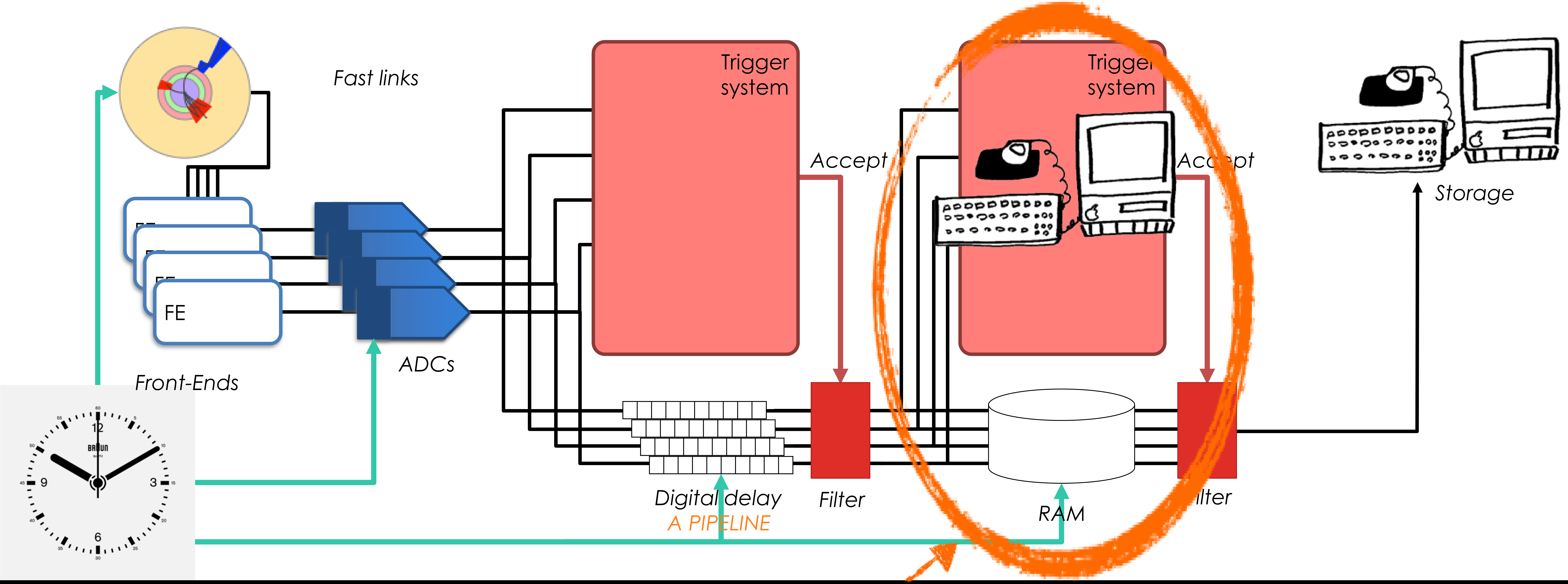


A SIMPLE MODERN TRIGGER SYSTEM



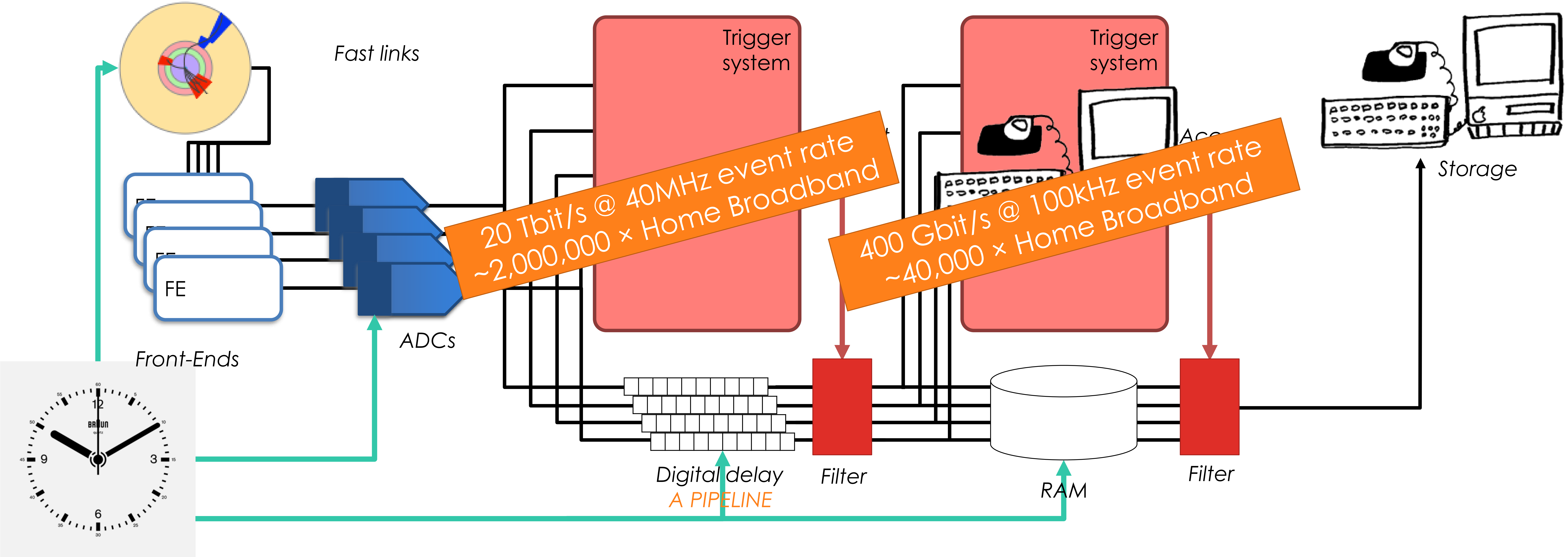
Each successive layer doing increasingly complex operations on ever decreasing event rates

A SIMPLE MODERN TRIGGER SYSTEM



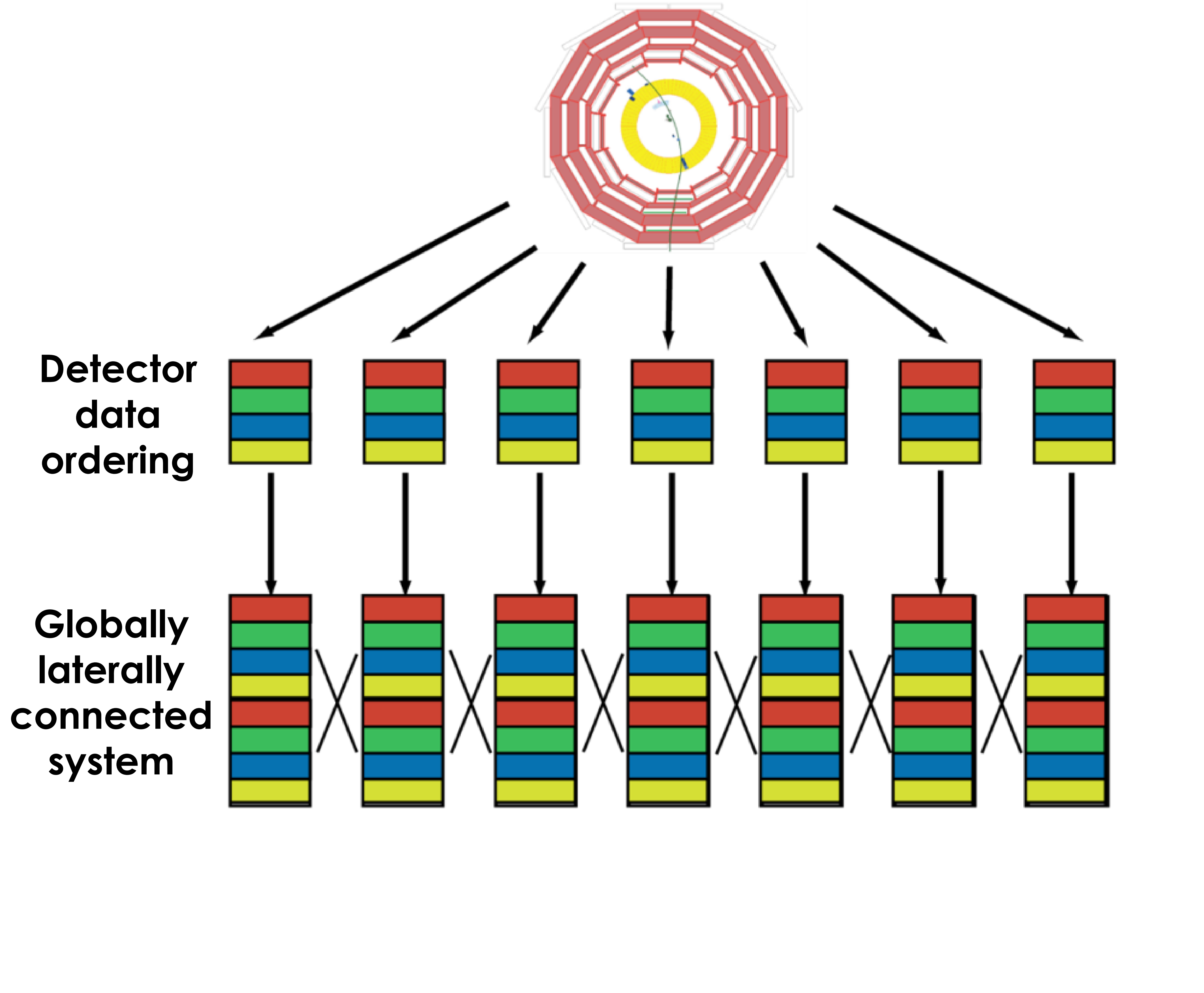
If your input rate is low enough…

OF COURSE, "LOW ENOUGH" IS RELATIVE



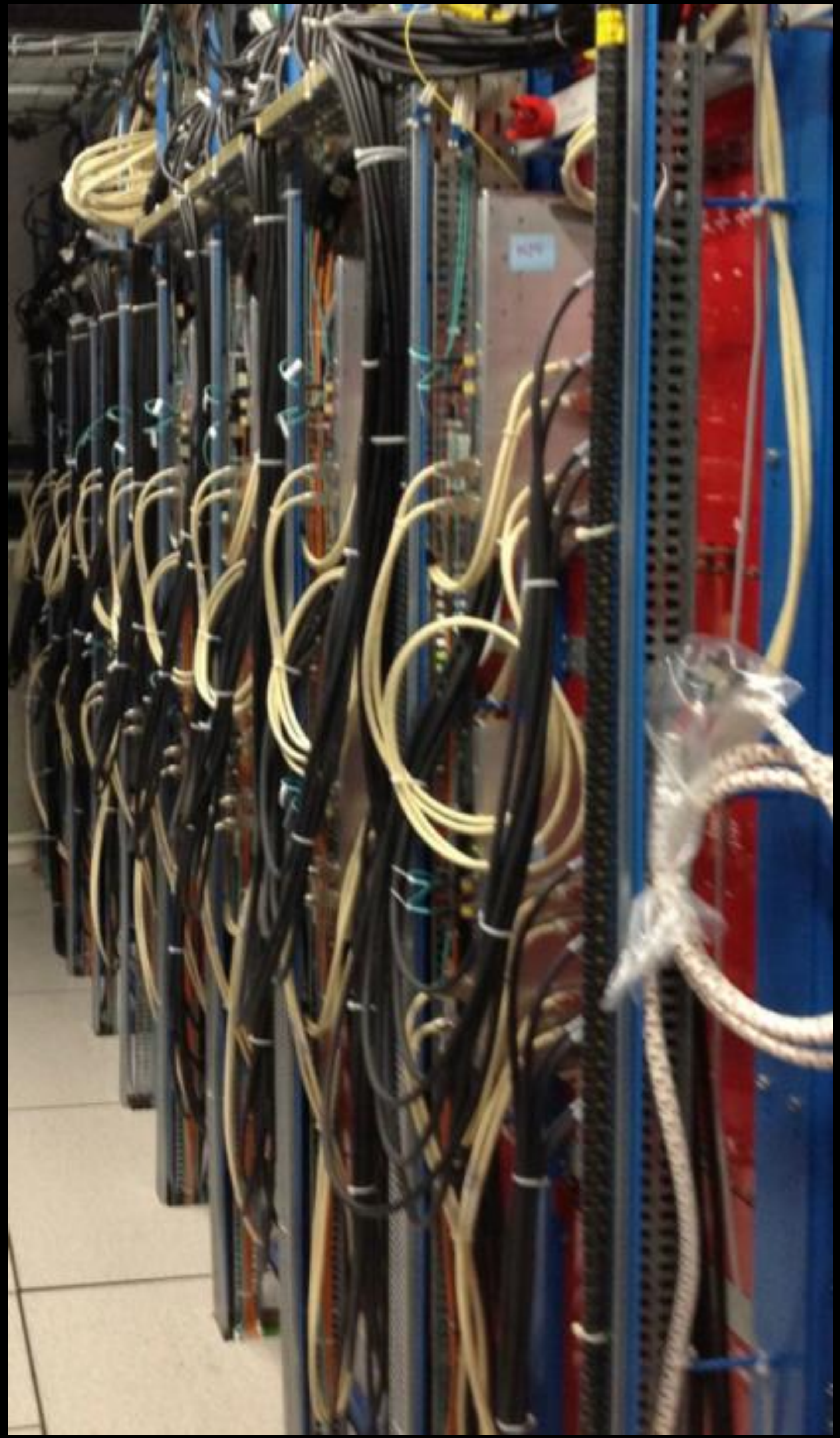
CMS: Runs 1 & 2

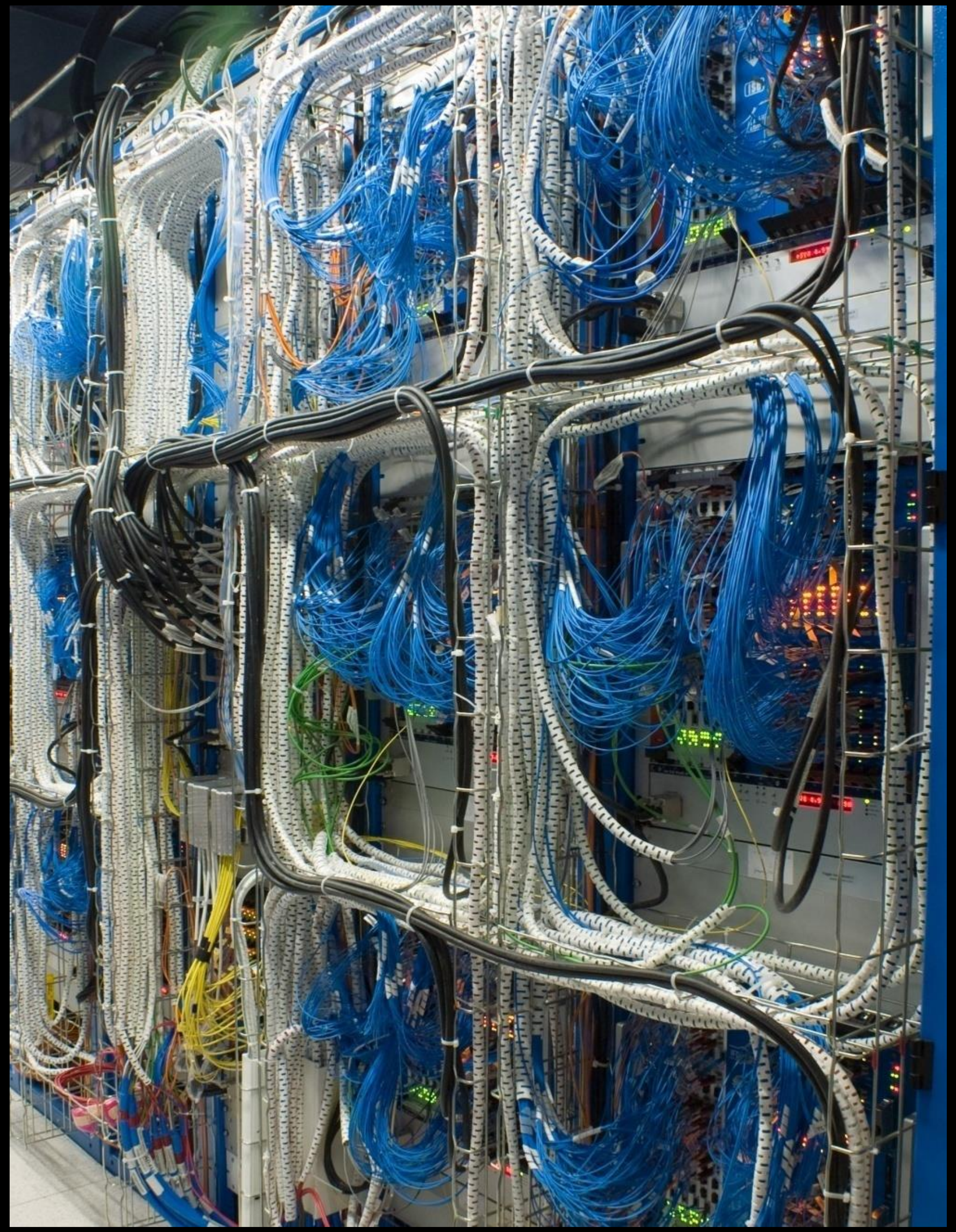
CONVENTIONAL ARCHITECTURE

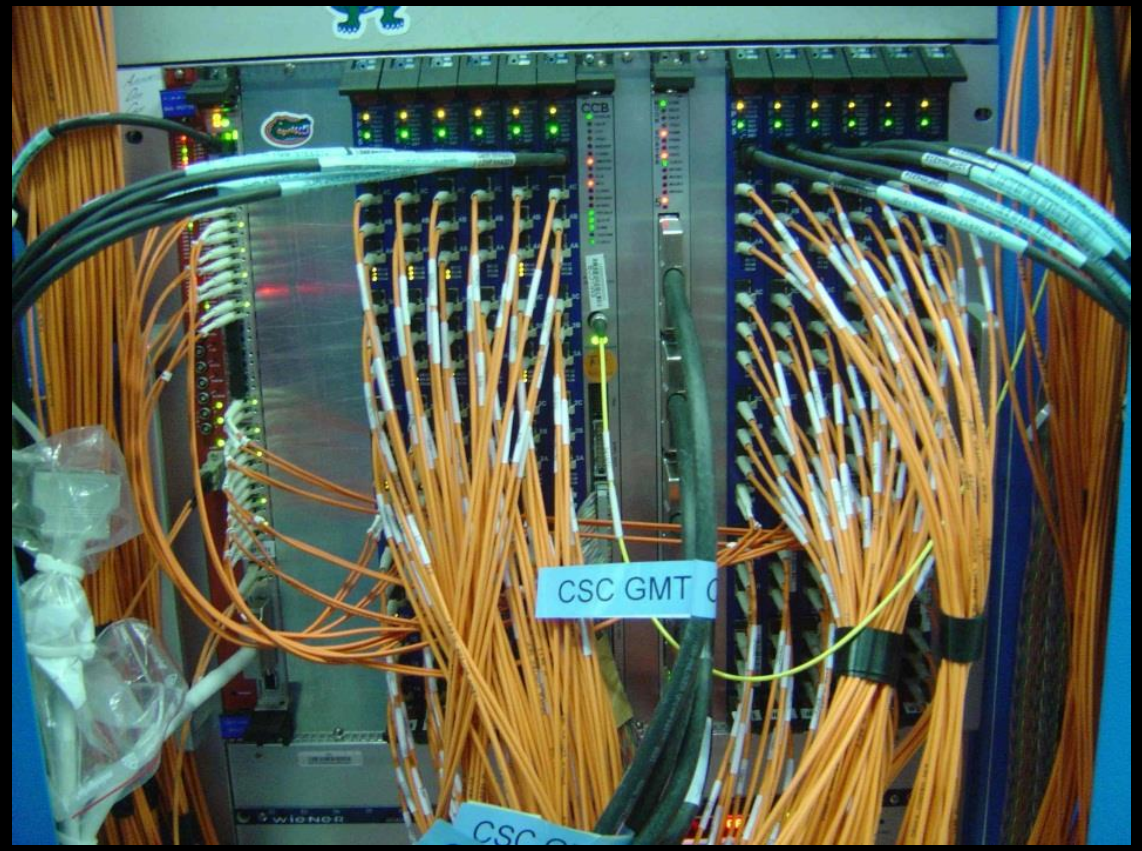


- Each subsystem is regionally segmented
  - Each region must talk to its neighbour
- Each region compresses, suppresses, summarizes or otherwise reduces its data and passes it on to the next level, which is less regionally segmented
- Repeat until you have a global “yes/no” answer

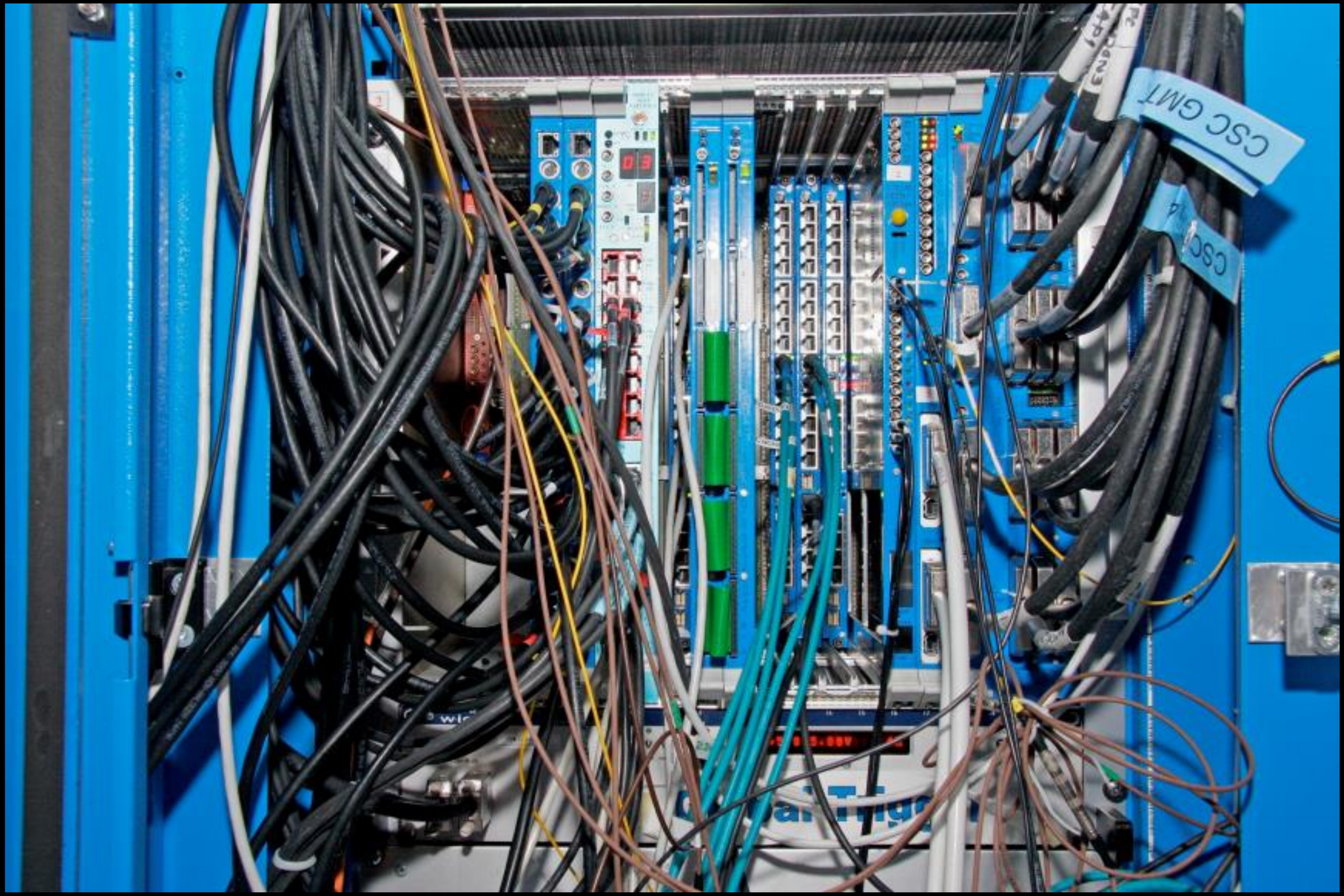
RUNS 0 & 1: HOW WE PROCESSED DATA







RUNS 0 & 1: HOW WE PROCESSED DATA



AN ASIDE: THE PROPHETIC WITTICISM

By 2015, every board in the [CMS] trigger will be identical and, after that, they will only get more similar

Andy Rose, 2010

WHY ALL THE DIFFERENT BOARDS?

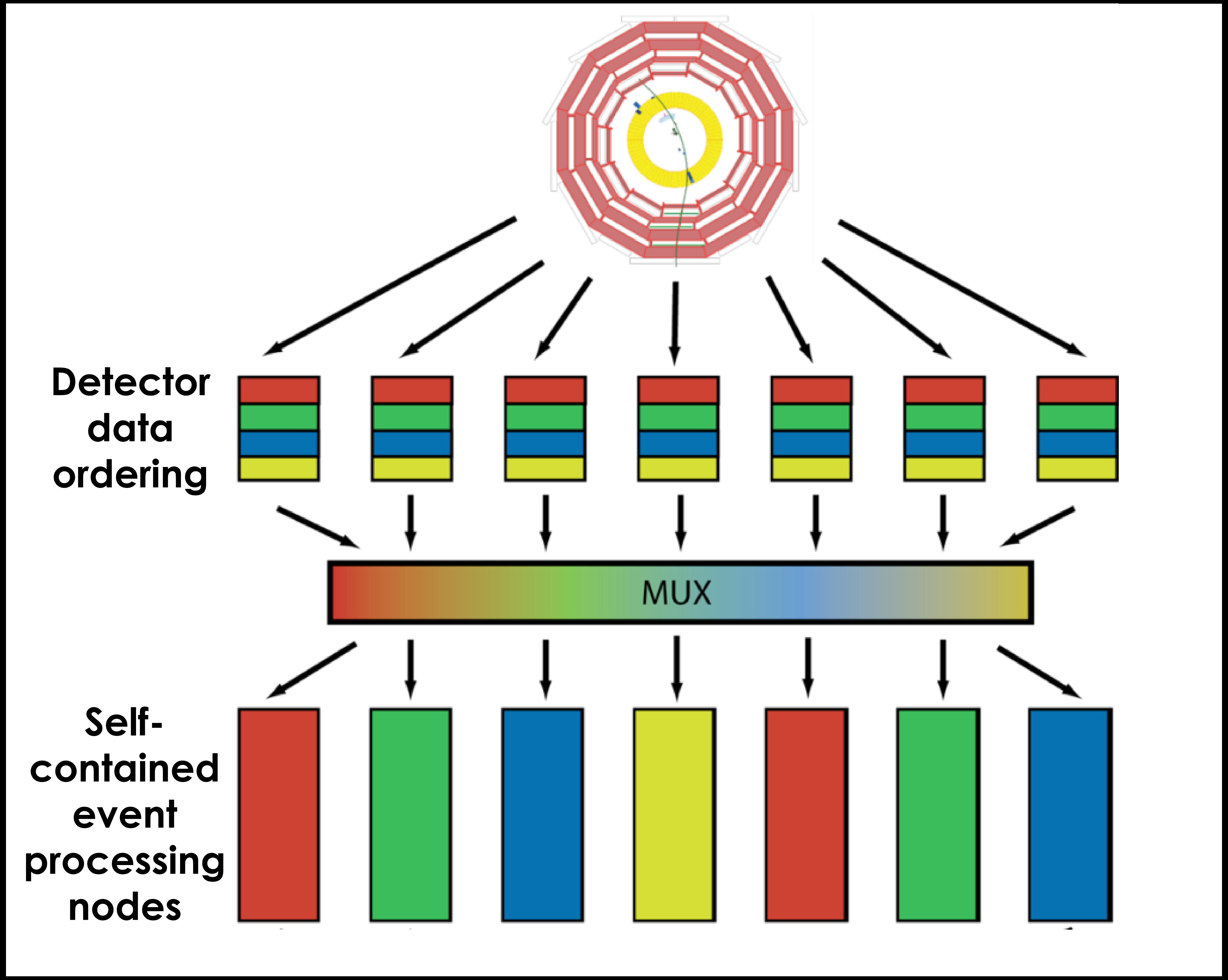
- Each board had to perform a very specific task
  - But there were many different board “flavours” with fully programmable logic
  - Why?

WHY ALL THE DIFFERENT BOARDS?

- Each board had to perform a very specific task
  - But there were many different board "flavours" with fully programmable logic
  - Why?
- The "meaning" of regional segmentation is different for each system and it is boundary handling which results in different board flavours

TIME-MULTIPLEXED ARCHITECTURE

- Buffer data and stream it out optimized for processing
- Spread processing over time
  - Stream-processing rather than combinatorial-logic
  - Maximise reuse of logic resources
  - Easiest for FPGA design tools to route and meet timing



WHY TIME-MULTIPLEX?

- Parallel processing is hard
  - Parallel processing 20Tb/s with a 1μs latency limit is really hard
- How you structure your data is the most important decision you will ever make

"The parallel approach to computing does require that some original thinking be done about numerical analysis and data management in order to secure efficient use. In an environment which has represented the absence of the need to think as the highest virtue this is a decided disadvantage."

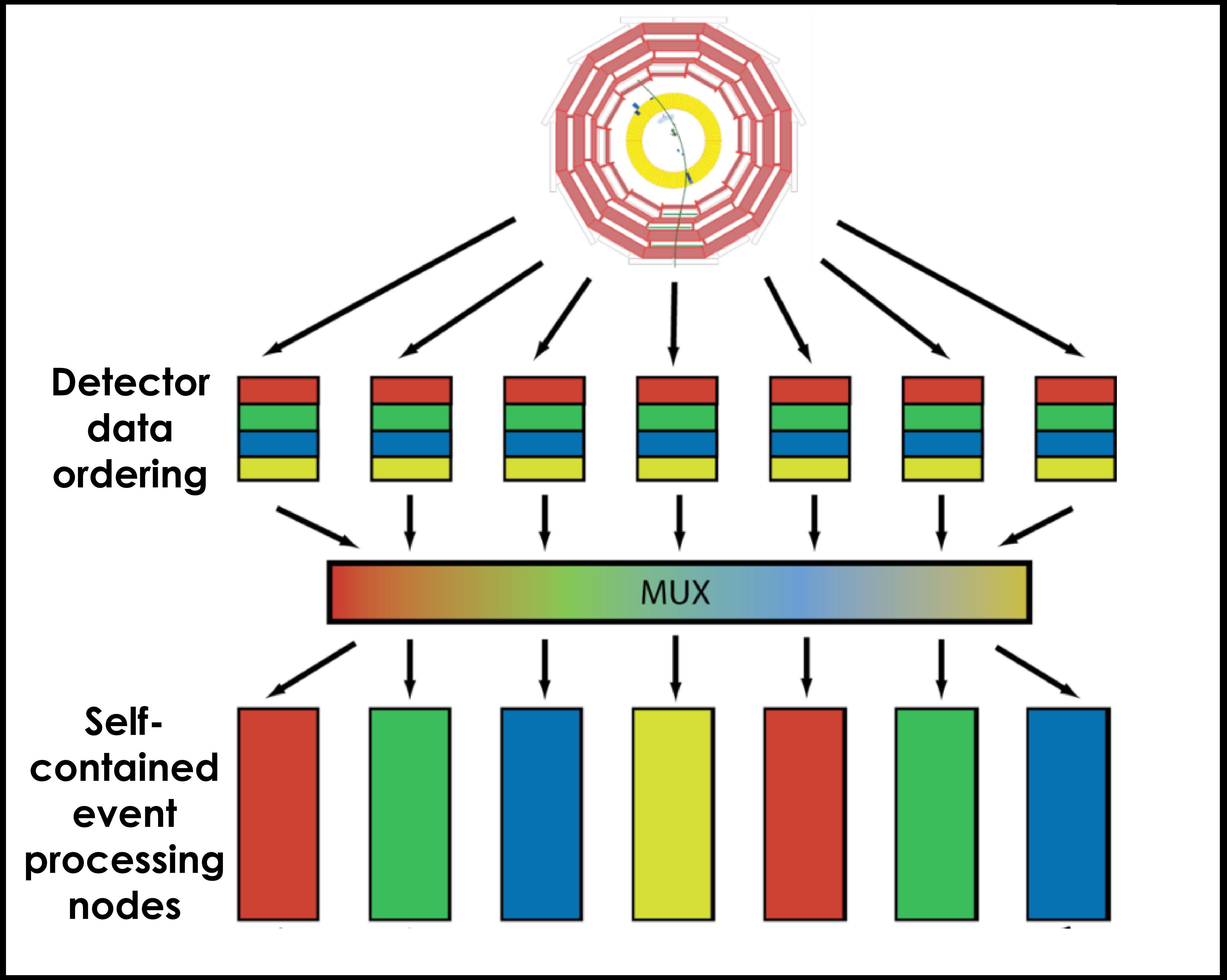
Daniel Slotnick, 1967

"I will, in fact, claim that the difference between a bad programmer and a good one is whether he considers his code or his data structures more important"

Linus Torvalds, 2006

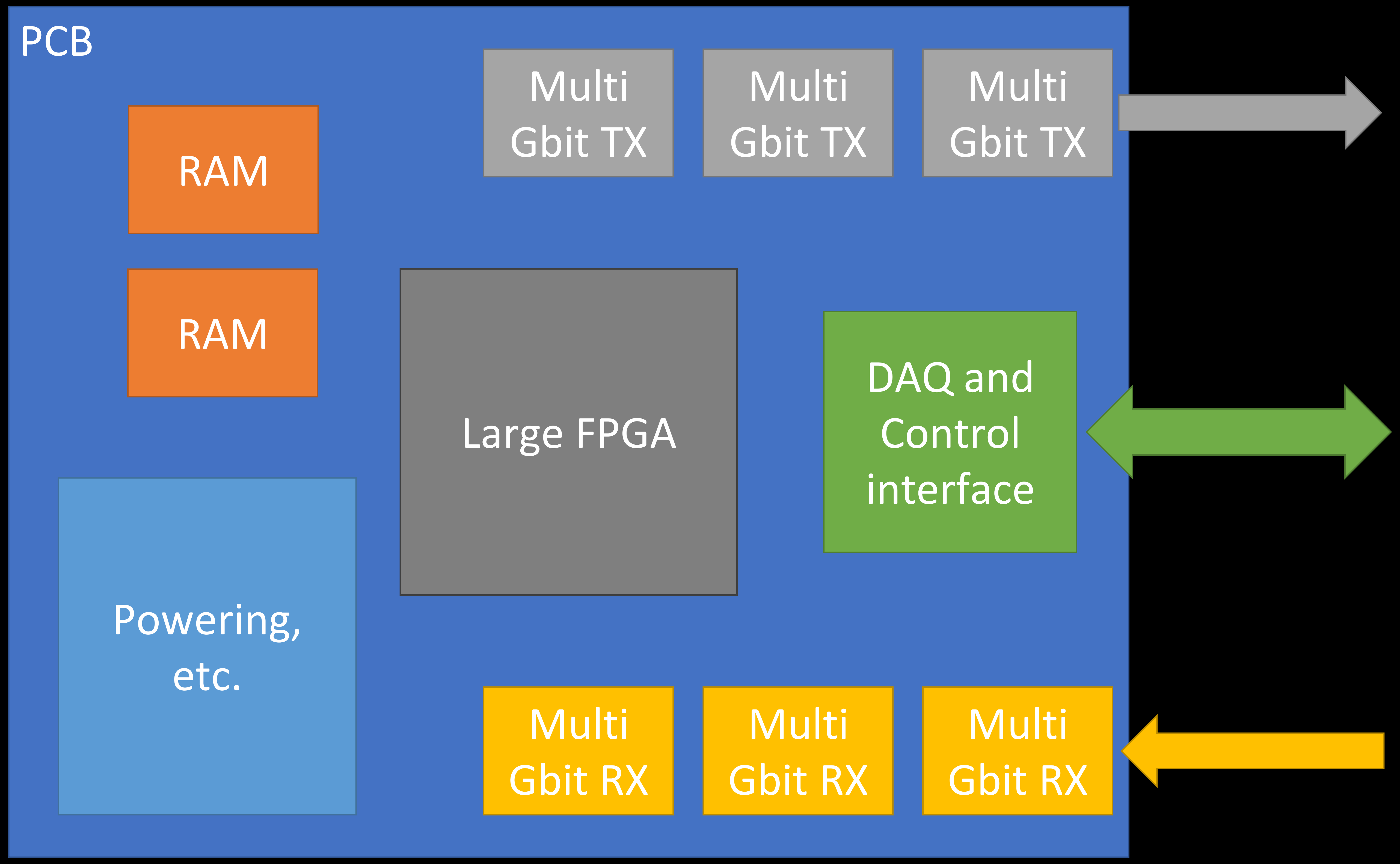
TIME-MULTIPLEXED ARCHITECTURE

- No longer required to throw away data at any given step
- No lateral sharing, fully flow-forward
  - Total equality of data
- Each board now
  - Receives parallel streams of data
  - Processes parallel streams of data
  - Outputs parallel streams of data

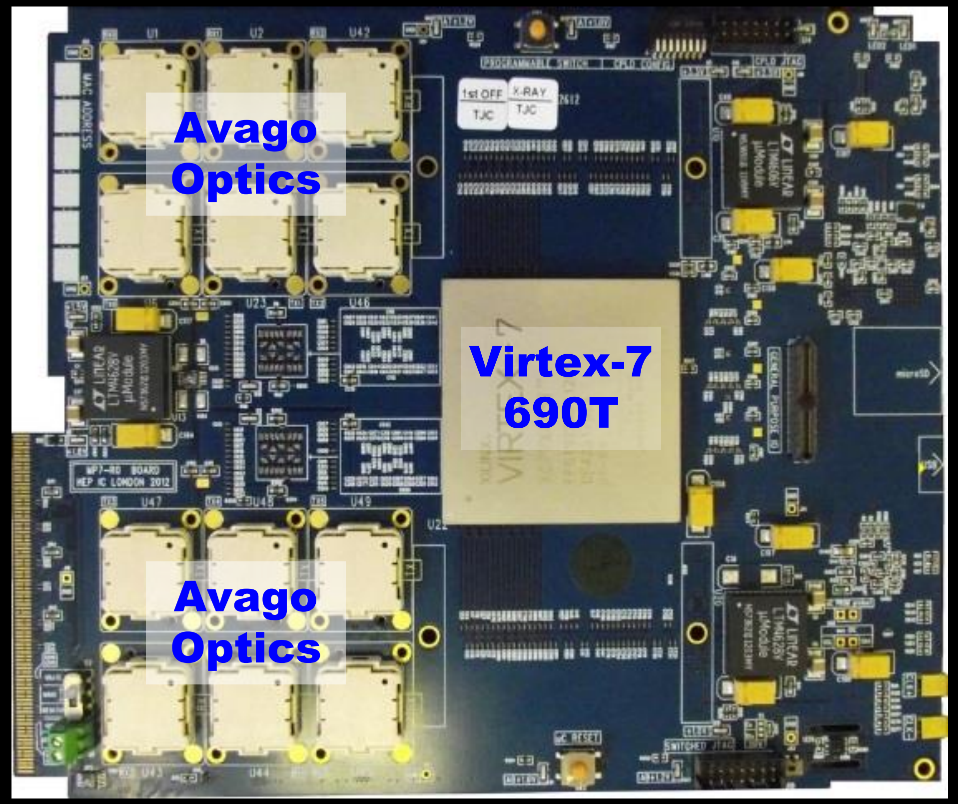


THE PROPHETIC WITTICISM

- And if each board now
  - Receives parallel streams of data
  - Processes parallel streams of data
  - Outputs parallel streams of data
- Coupled with industry trends
- The variation between boards disappears
  - Hence the prophecy

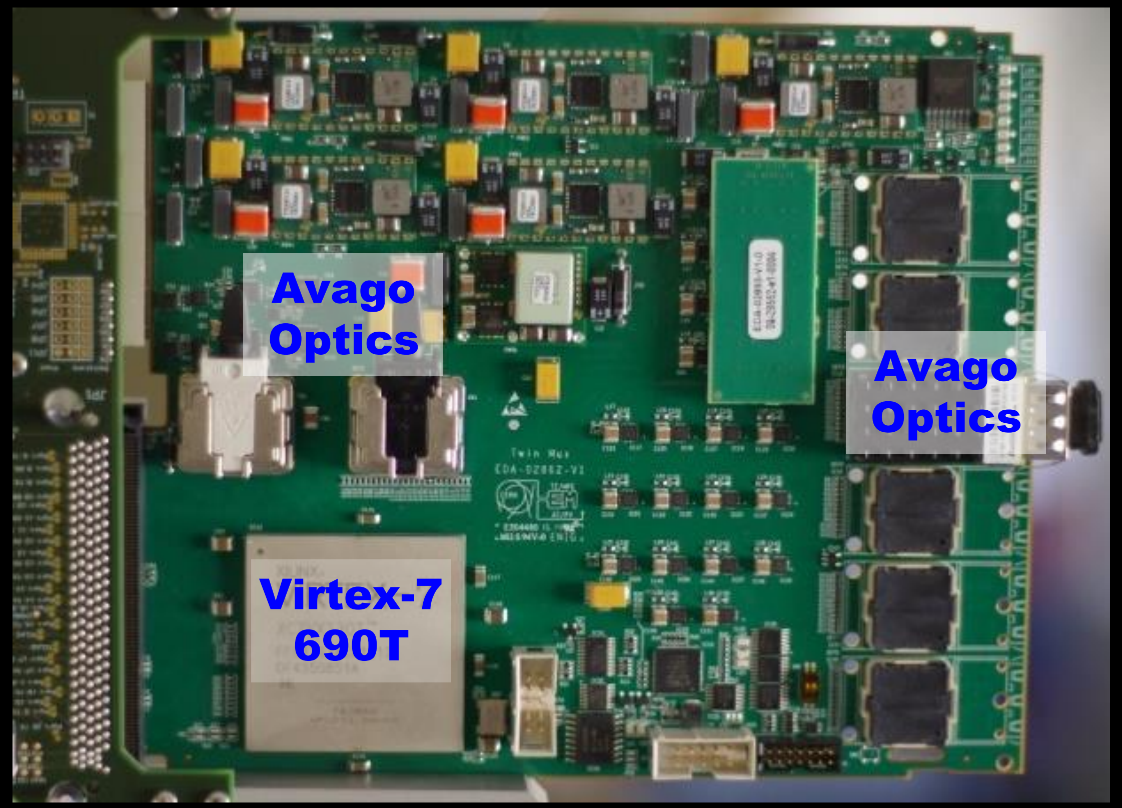


THE PROPHETIC WITTICISM: 2015 UPGRADES

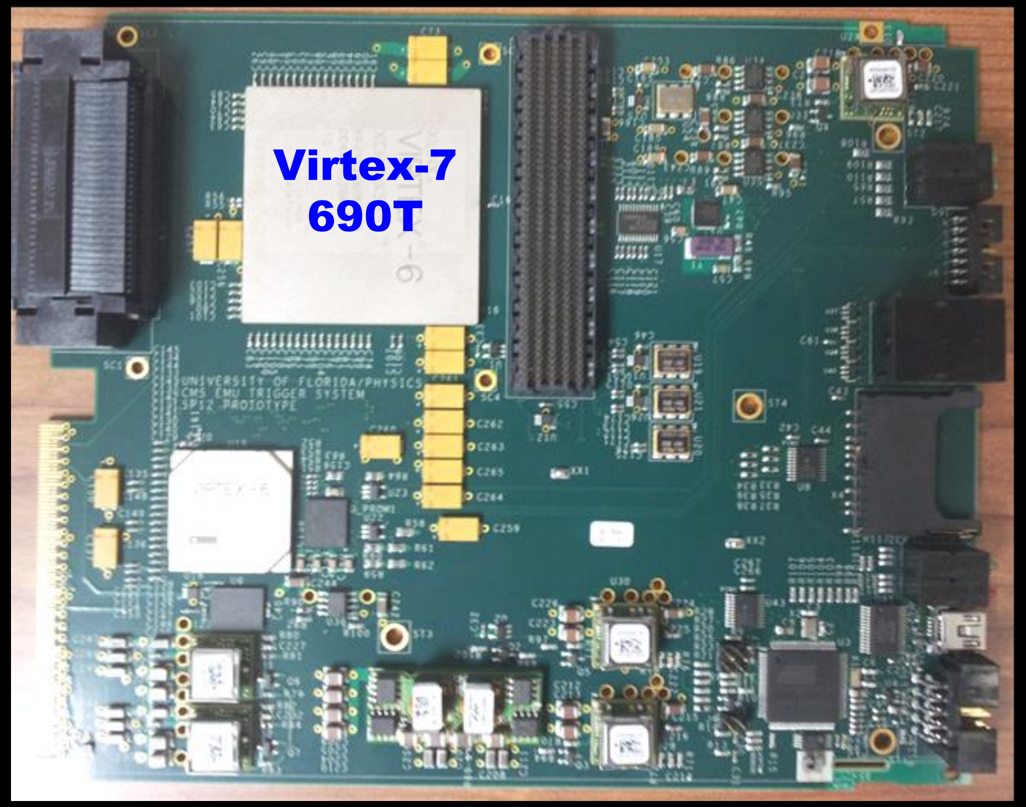


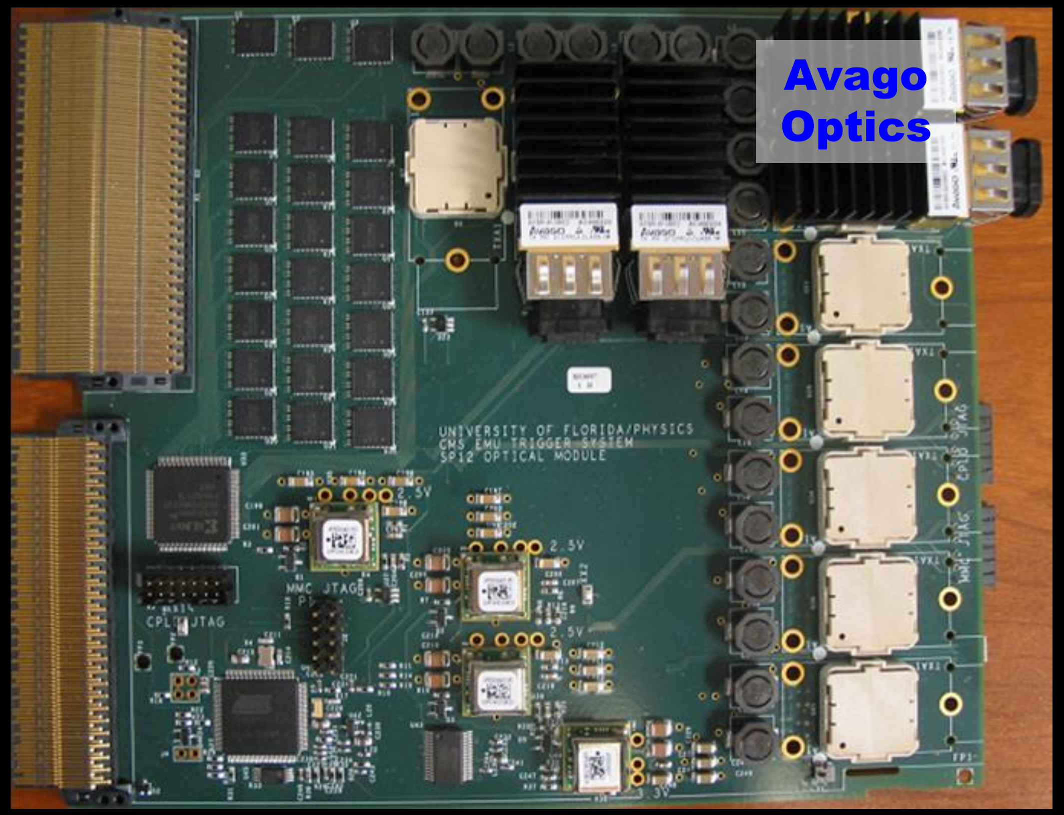
MP7

CTP7

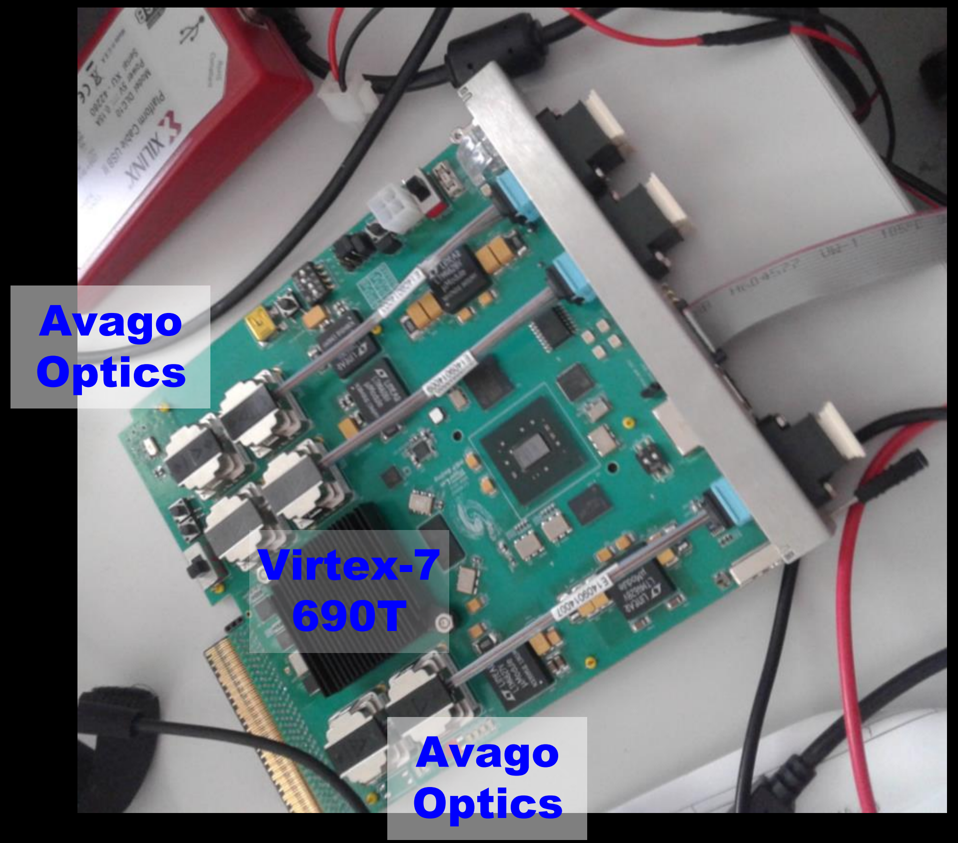


TwinMux



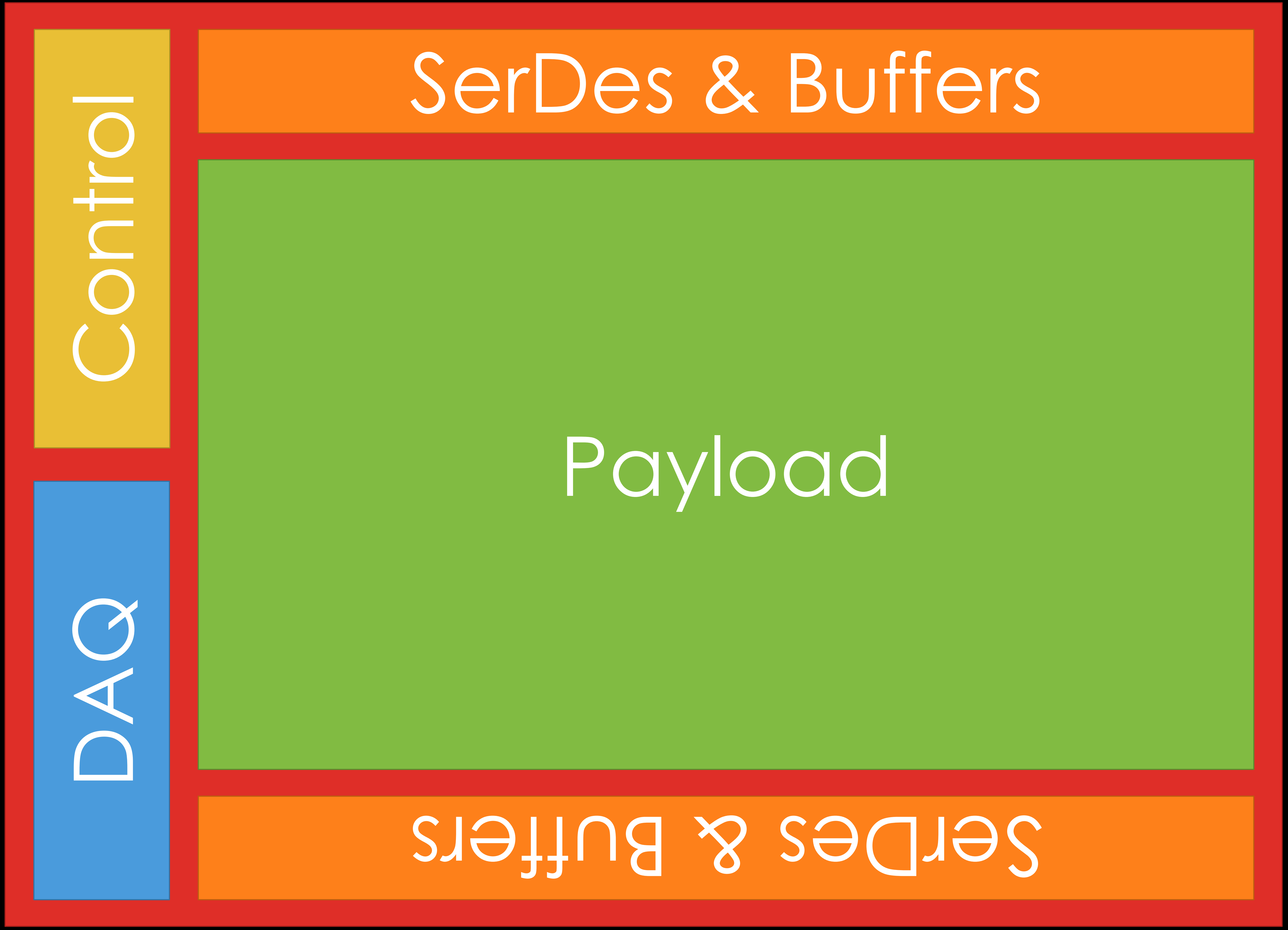


MTF7 (MTF6 shown)



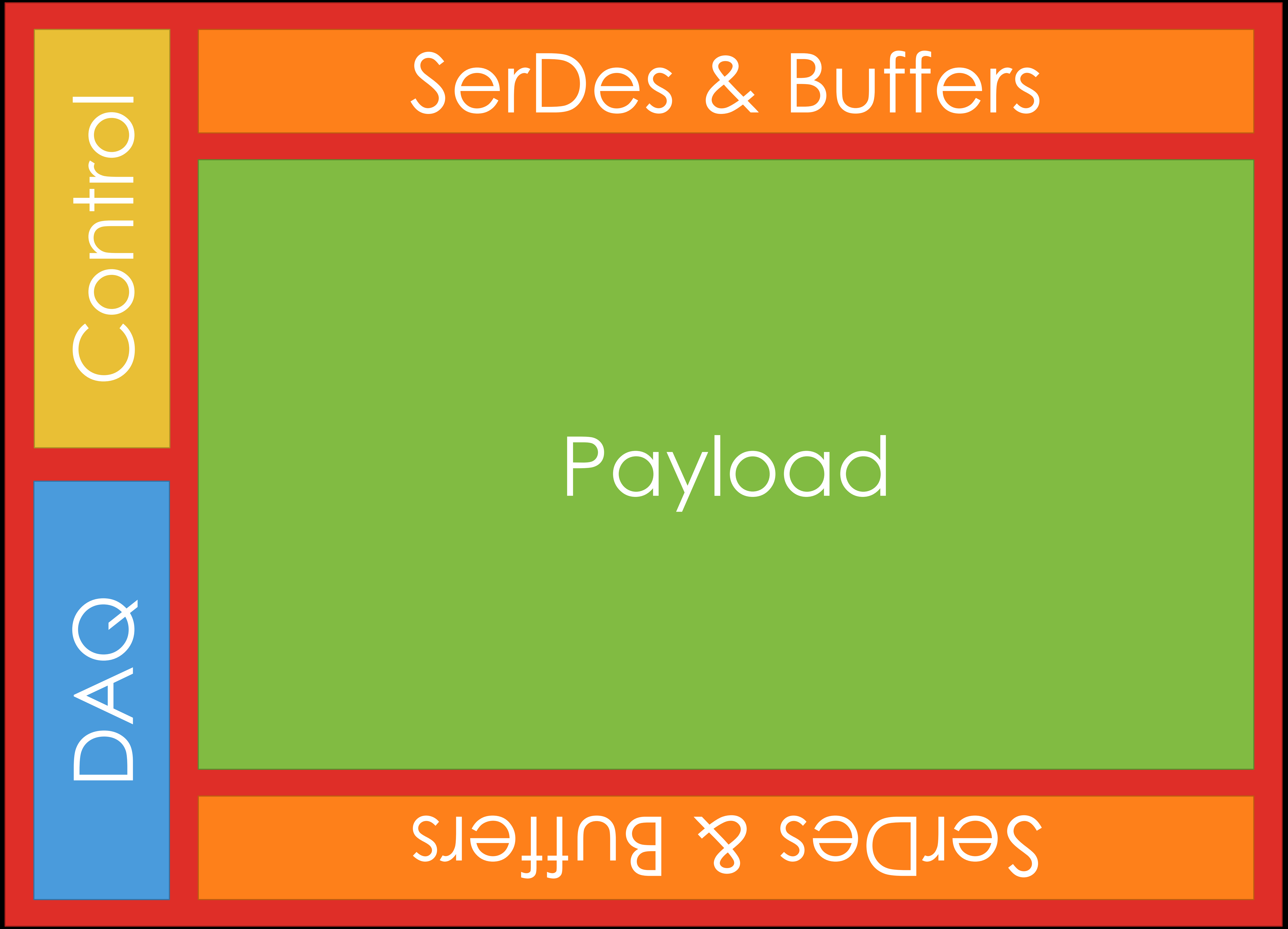
CPPF

AND THE SIMILARITY CONTINUES INTO THE CHIP



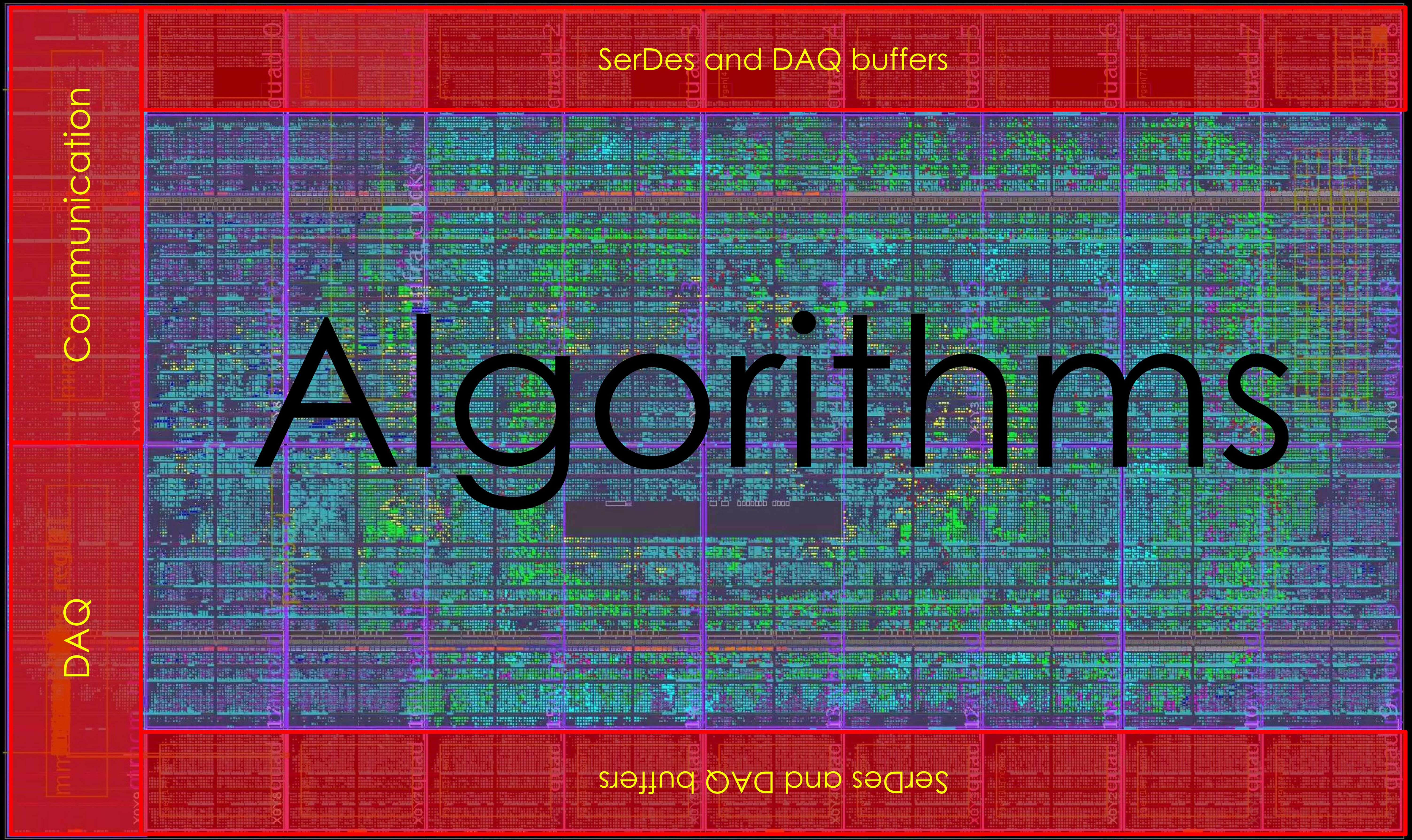
AND THE SIMILARITY CONTINUES INTO THE CHIP

- Can separate the payload from the infrastructure by standardizing the interfaces
  - A Firmware operating system!
- Requires a build-tool to allow anyone to build their design in "unfamiliar hardware"



AND THE SIMILARITY CONTINUES INTO THE CHIP

- Can separate the payload from the infrastructure by standardizing the interfaces
  - A Firmware operating system!
- Requires a build-tool to allow anyone to build their design in "unfamiliar hardware"
- User free to focus on the algorithms



2015 UPGRADE: CONCLUSION

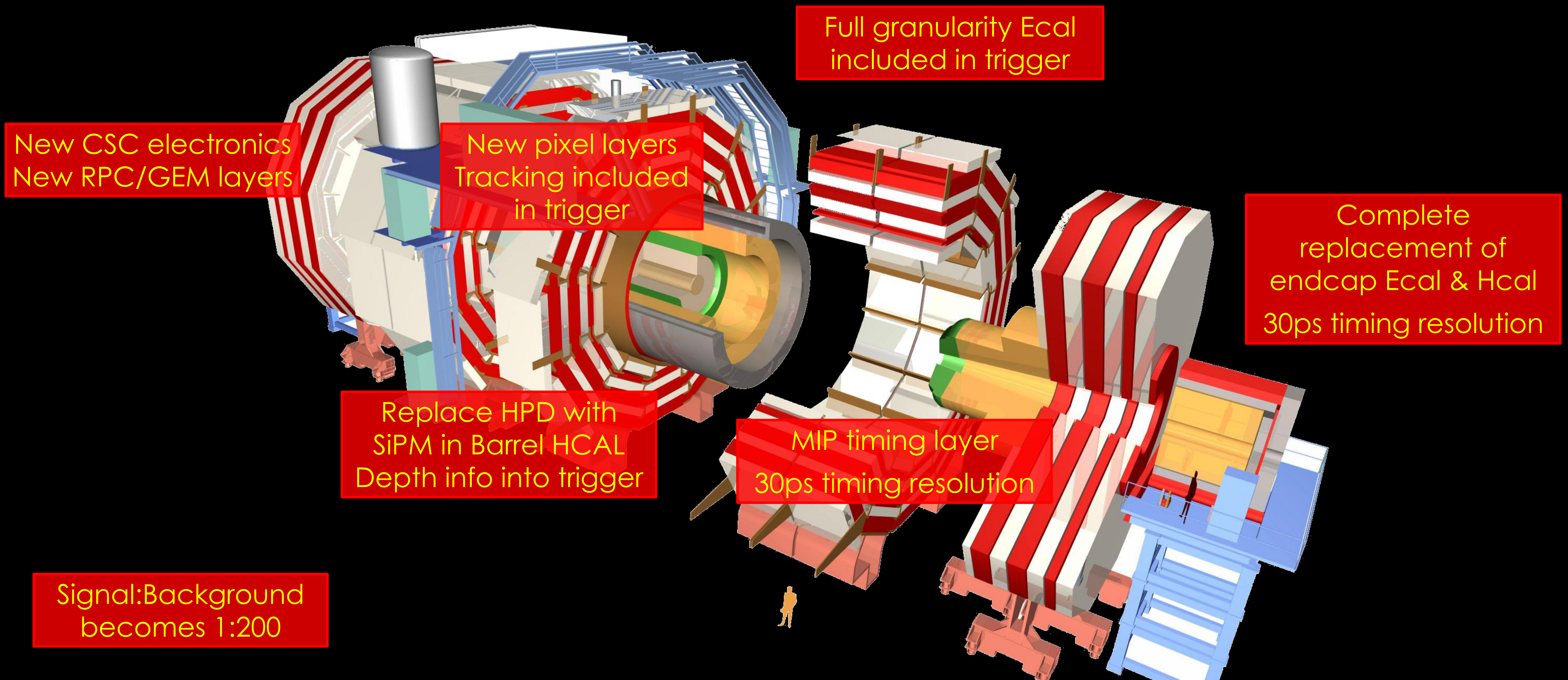
- The UK-originated concepts of
  - Time-multiplexing (& fully stream-oriented trigger processing)
  - Generic hardware
  - The "firmware operating system"

Have all been shown to provide enormous benefit and have gained traction in the CMS collaboration

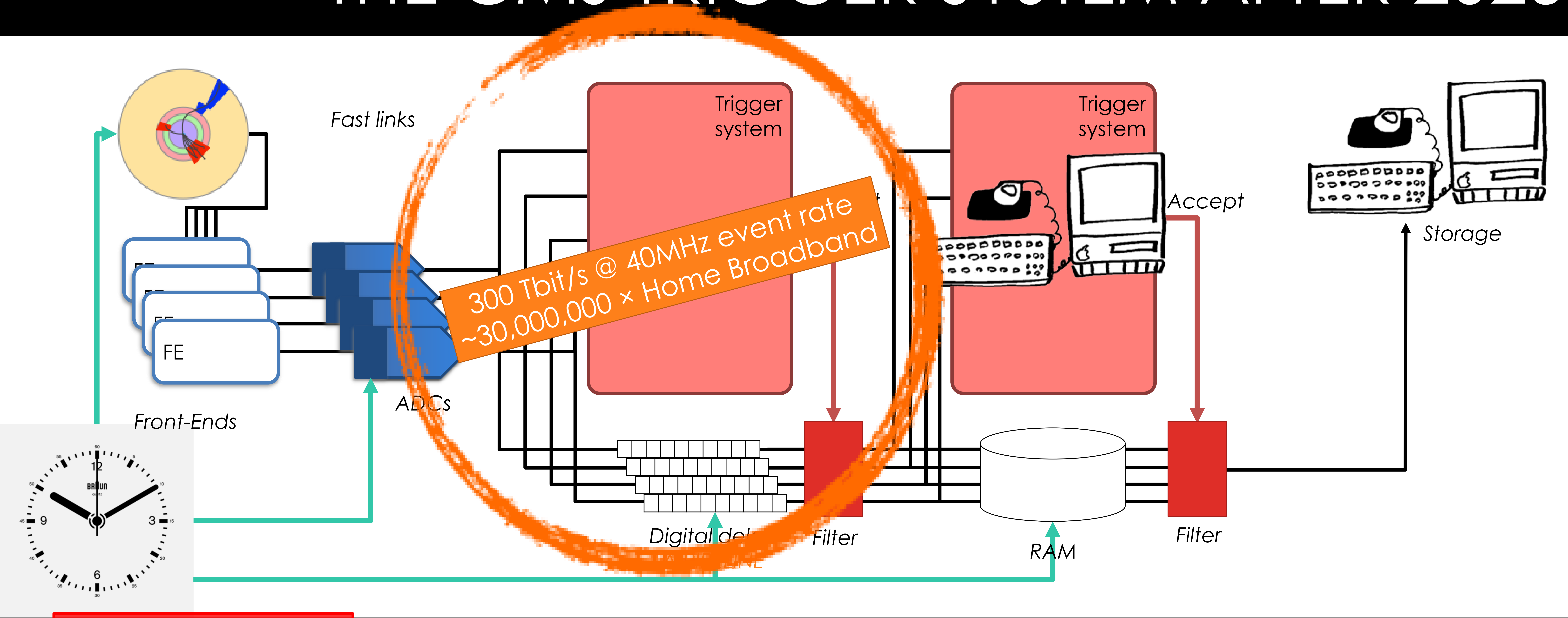
COMMON X-WARE: WHAT BENEFITS?

- Risk mitigated
  - More time/resources to validate
  - More users become experts – fewer critical people
- Maximal reuse of software & firmware (Minimal reinvention of the wheel)
  - More cost-effective
  - More time to spend on “physics” algorithms
- Common hardware is cheaper
  - Fewer production runs
  - Bulk buying (Bigger Pack – Better Value)

THE CMS DETECTOR AFTER 2025

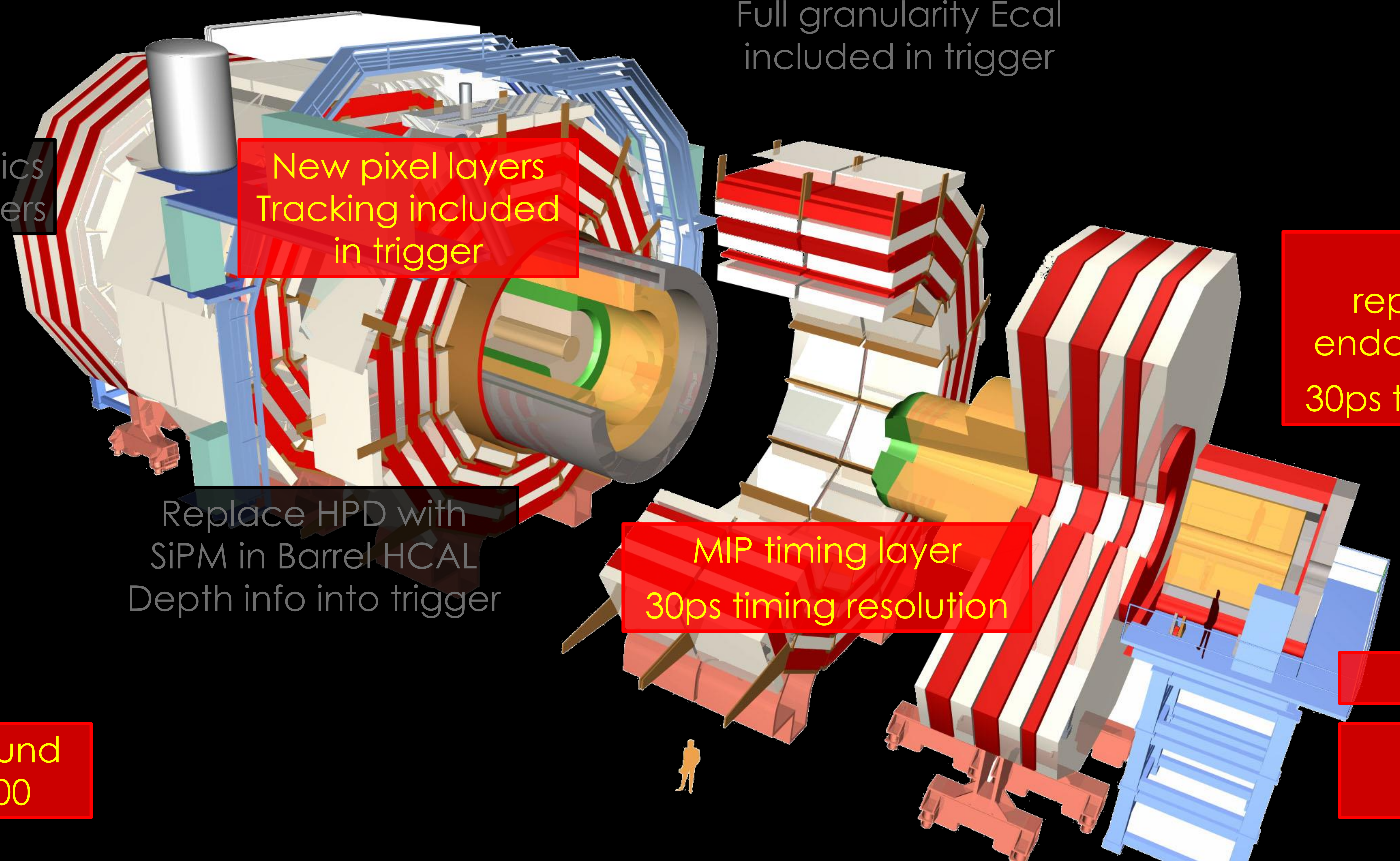


THE CMS TRIGGER SYSTEM AFTER 2025



Signal:Background becomes 1:200

THE CMS DETECTOR AFTER 2025: UK ROLES



THE CMS DETECTOR AFTER 2025: UK ROLES

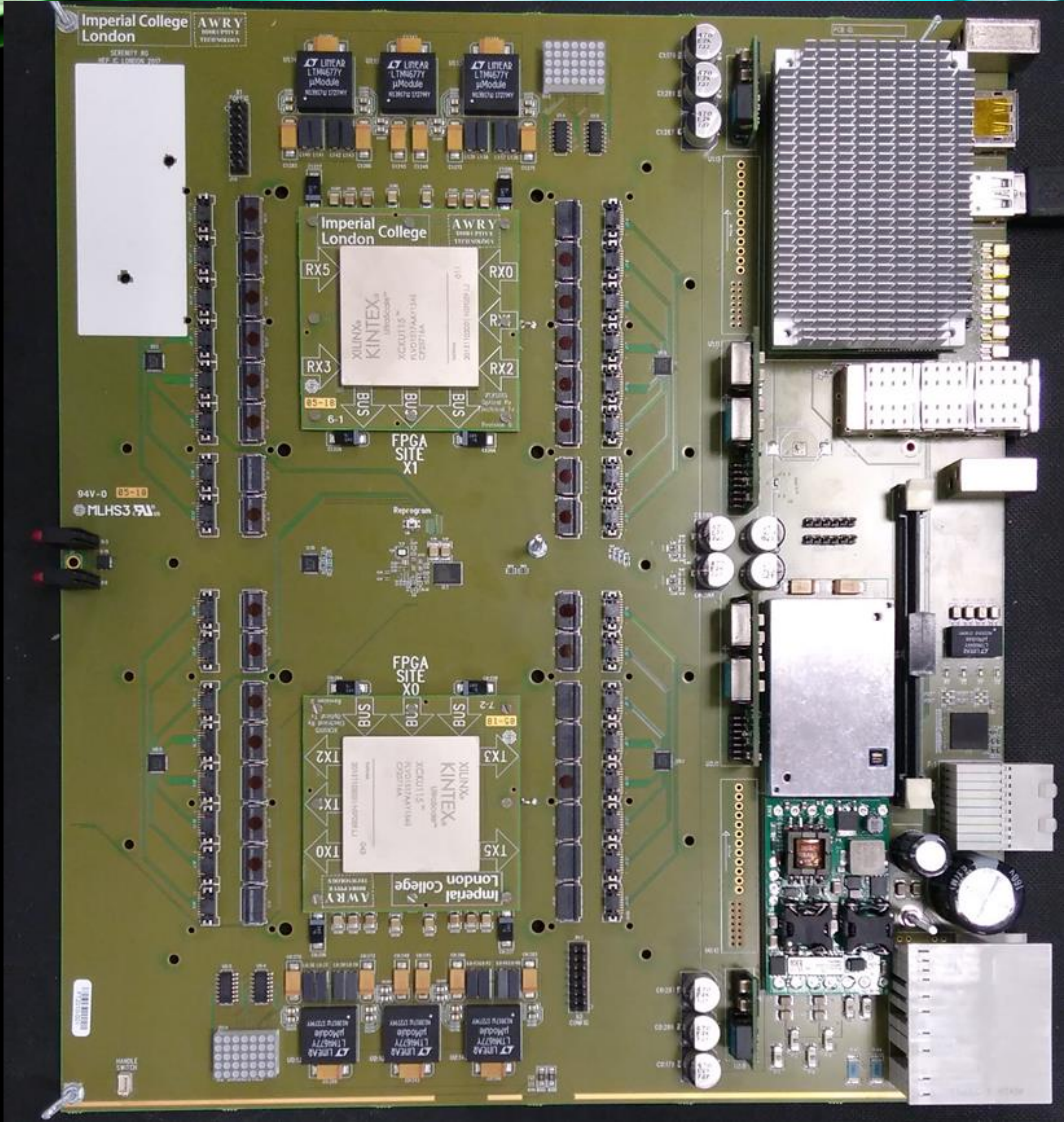
Oh… and we are only 4 institutes

Putting our money where our mouth is:

A list of projects like that requires taking the common X-ware concept to the extreme

COMMON HW: SERENITY

- ATCA Development Platform
- Generic optical stream processor
  - Up to 5.375 + 5.375 Tb/s Optical I/O
  - Nothing specific to any subsystem
    - In fact, nothing specific to CMS
- Carrier Card provides board services
- Daughter Cards host data-processing FPGAs



CONCLUSION

- Common Hardware, Software & Firmware has provided significant gains for 2015 upgrades of CMS
  - The UK has been pivotal in moving the CMS off-detector electronics away from highly customized hardware towards a common X-ware
  - These developments are also already providing benefits for other experiments
- The triggering challenges posed by the 2025 upgrades are extreme
  - Ad hoc approaches will fail, best-practice is not optional
  - The common X-ware allows for more effort to be focused on the physics

THANKS FOR LISTENING!

Any questions?

SPARES

THE PROPHETIC WITTICISM: 2019

The tyranny of the links

$$N_{bits} = \underbrace{N_{links}}_{\text{Restricted by available space}} \times \underbrace{N_{channels/link} \times Linespeed \times \varepsilon_{encoding}}_{\text{Restricted by technology available}} \times \underbrace{\tau_{transmission}}_{\text{Restricted by only feasible architecture}}$$

Question:

So, what can you do?

Answer:

Chose the best available technology you can and then throw out as much data as you can get away with until you satisfy your bit-limit

Imagine a “card” with 4 input links and up to 4 output links…

The conventional architecture

Can the tyranny be broken?

$$N_{bits} = \underbrace{N_{cables}}_{\text{Restricted by available space}} \times \underbrace{N_{channels/cable} \times Linespeed \times \varepsilon_{encoding}}_{\text{Restricted by technology available}} \times \underbrace{\tau_{transmission}}_{\text{Restricted by only feasible architecture}}$$

Restricted by available space

Restricted by technology available

~~Restricted by only feasible architecture~~

Maybe it's time to start questioning the received wisdom…

- Can we increase $\tau_{transmission}$?
- This requires a completely new architecture
- In order that every BX is processed, we require as many processing "nodes" as the number of bunch-crossings the data is transmitted over

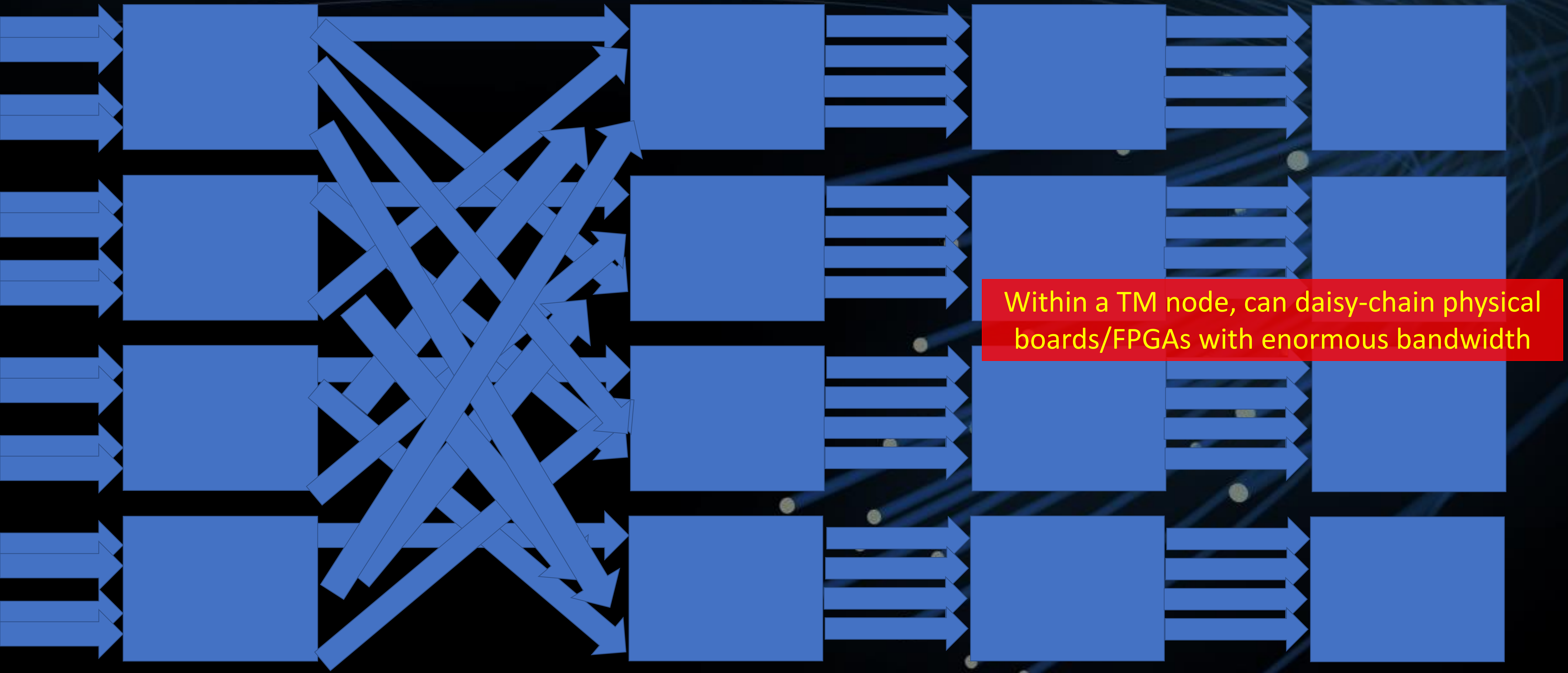
Imagine a "card" with 4 input links and up to 4 output links…

The time-multiplexed architecture

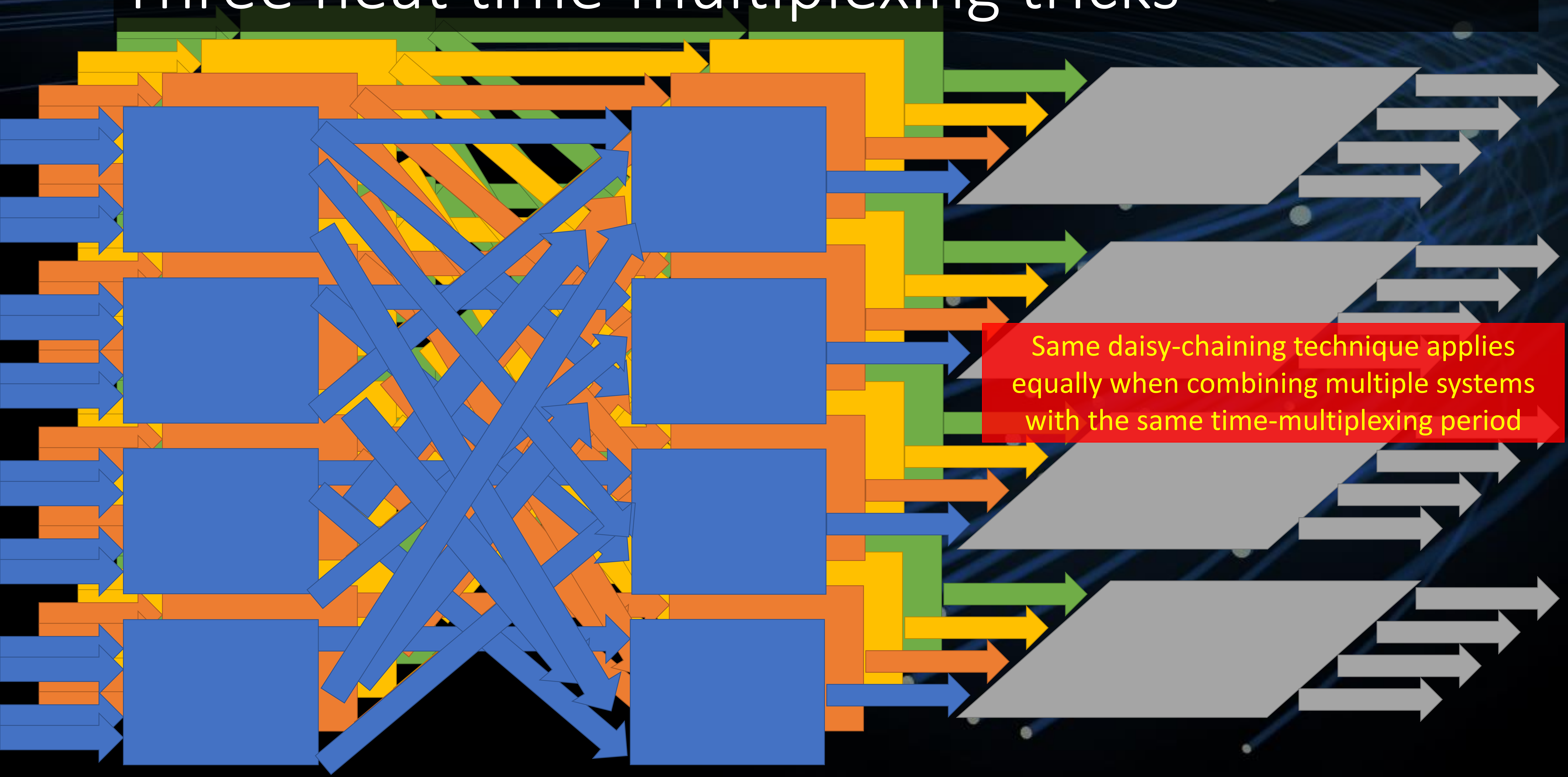
Well… this doesn't look like a great start – but we'll come back to it in a minute…

The time-multiplexed architecture

Three neat time-multiplexing tricks

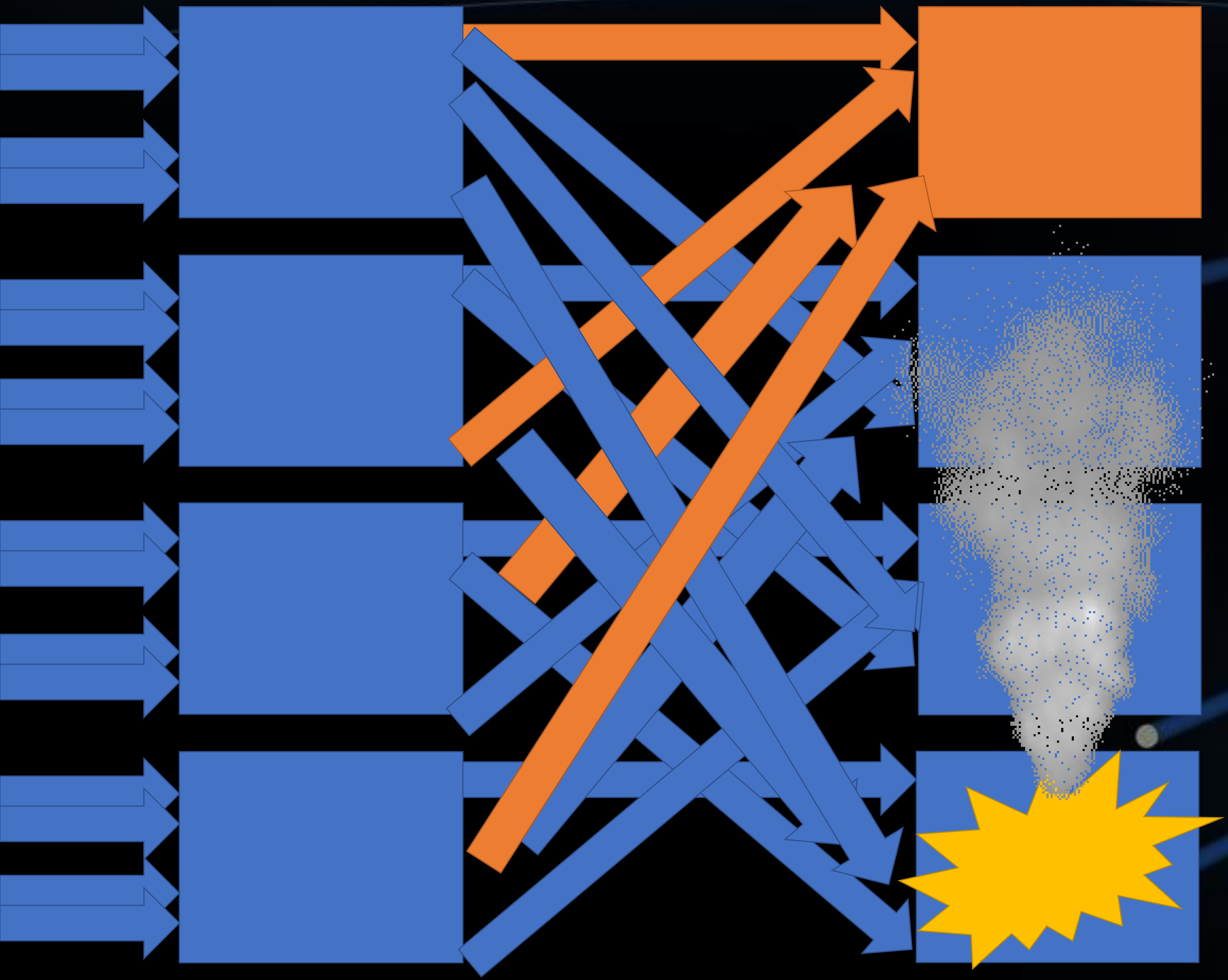


Three neat time-multiplexing tricks



Same daisy-chaining technique applies equally when combining multiple systems with the same time-multiplexing period

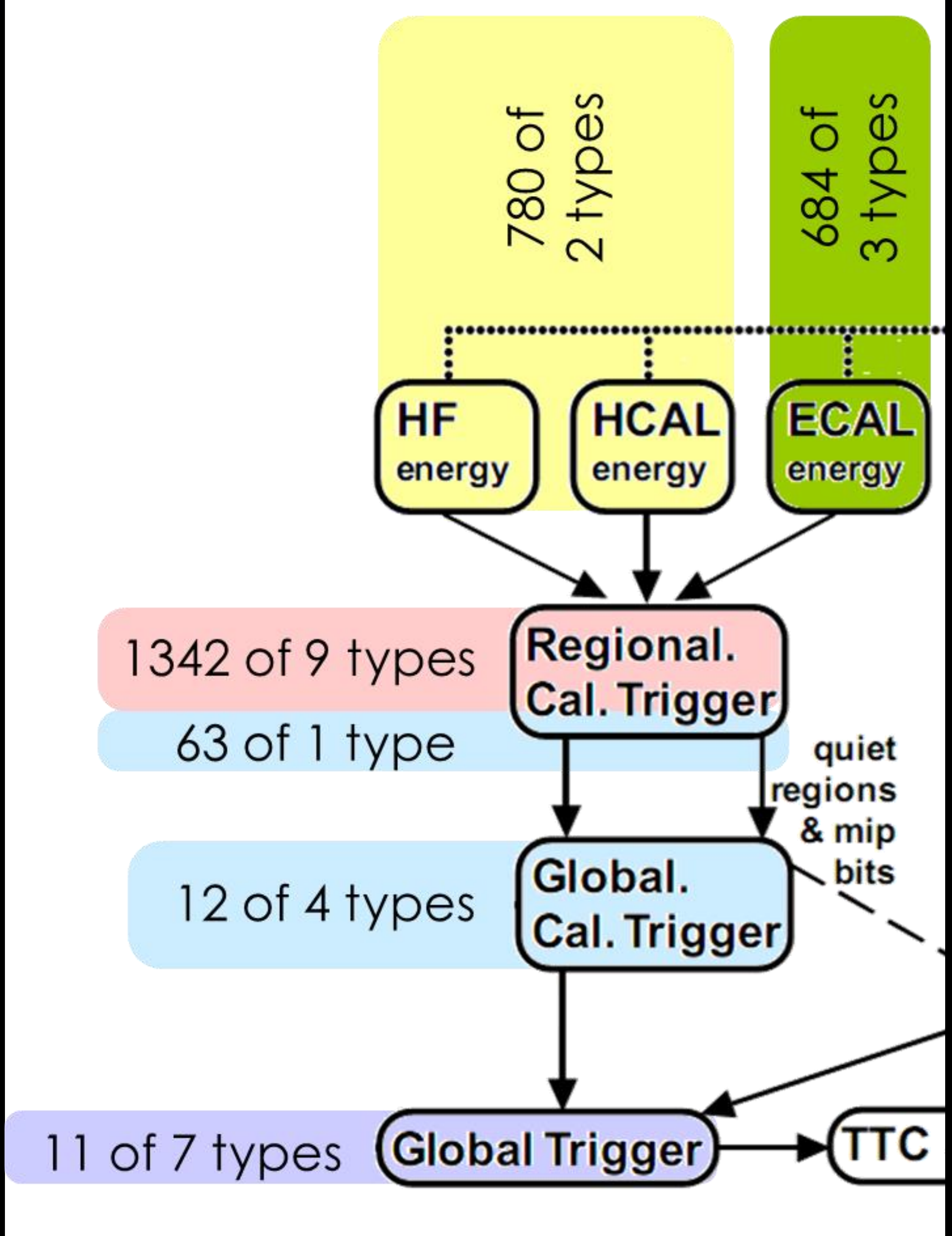
Three neat time-multiplexing tricks

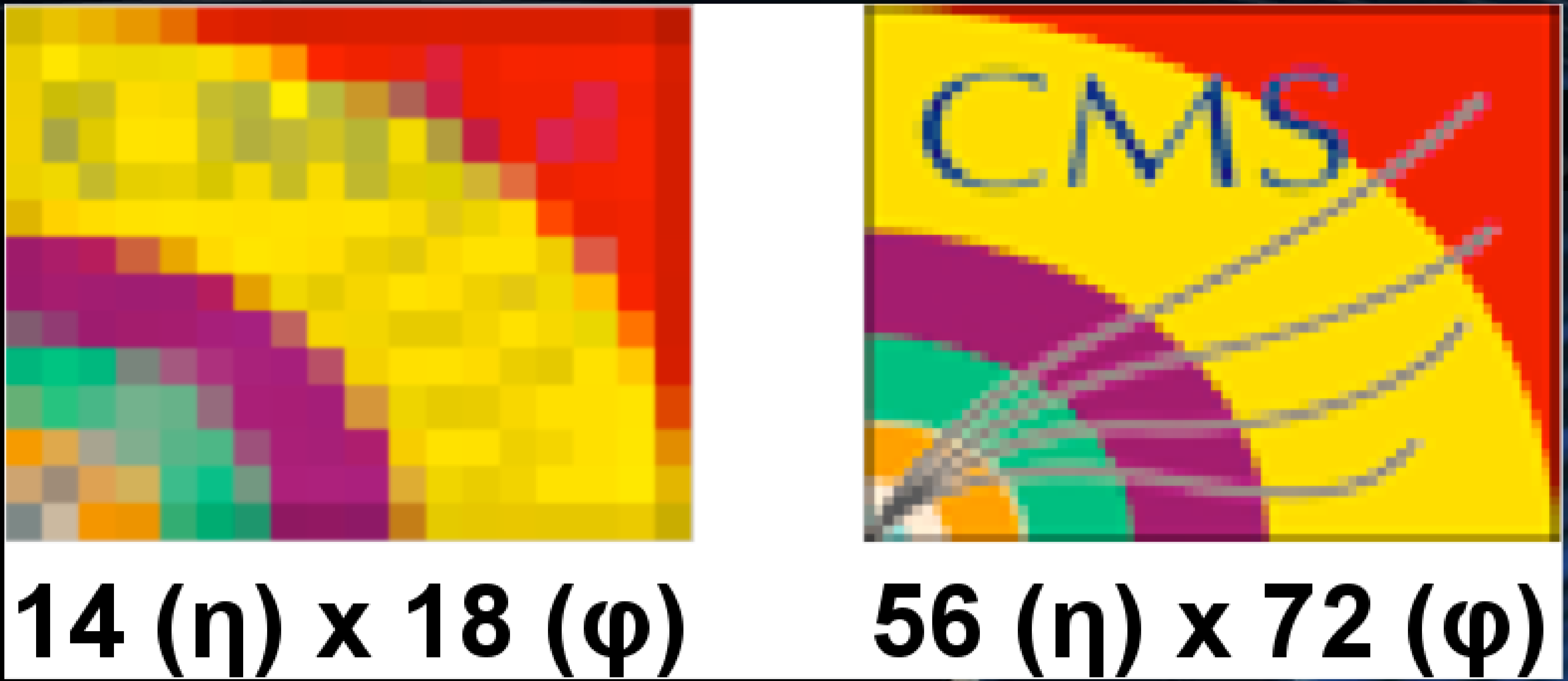


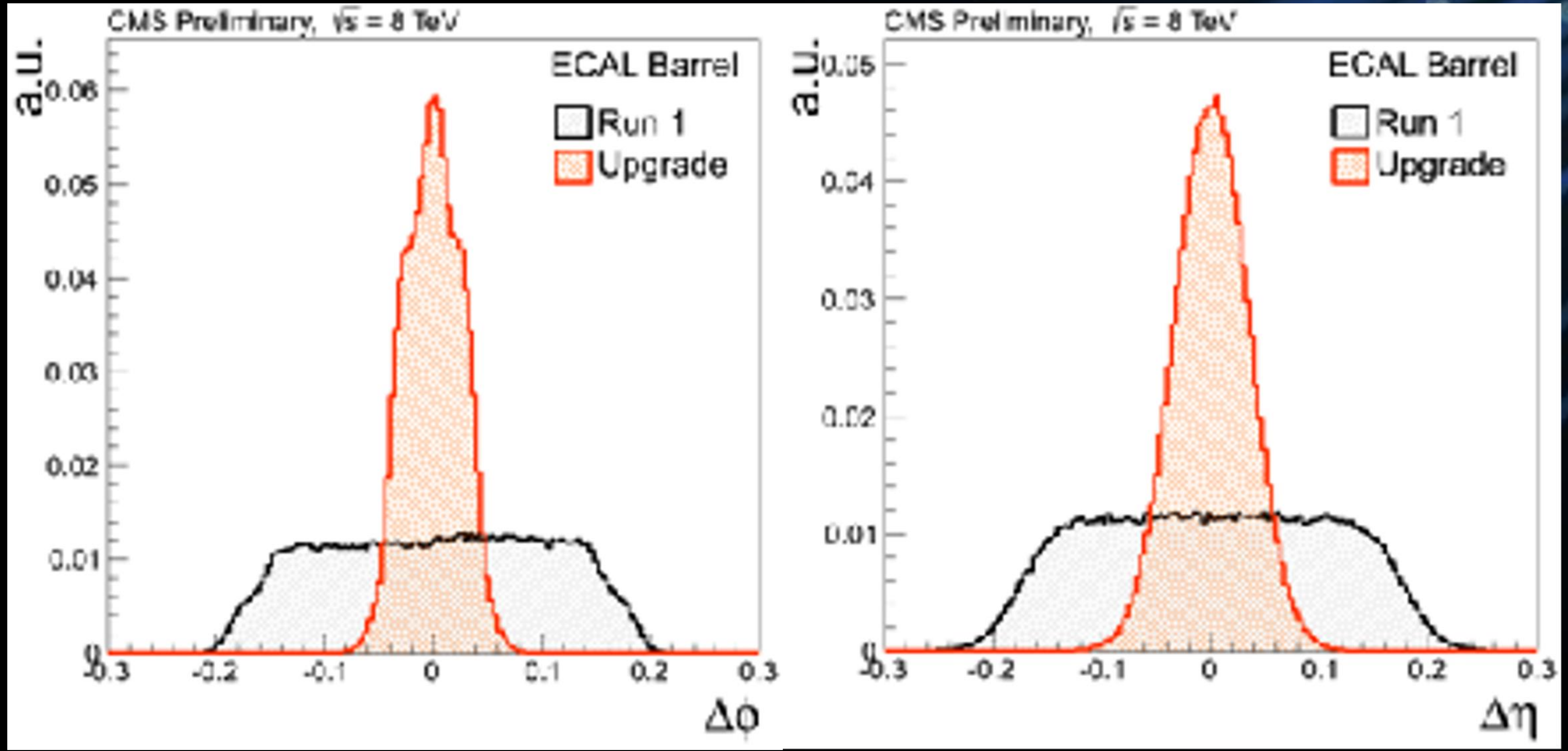
- Imagine a system with N nodes
- We lose a node due to hardware failure
  - In a conventional trigger losing a board means degrading (or losing!) every event!
  - With TM, we lose 1 BX in N – a (inadvertent) trigger prescale!
- But we have a final card up our sleeve!
- For an additional 1/Nth of the cost we can have a spare node which we can switch in at run-time
  - Efficiency restored!

Does TM actually work?

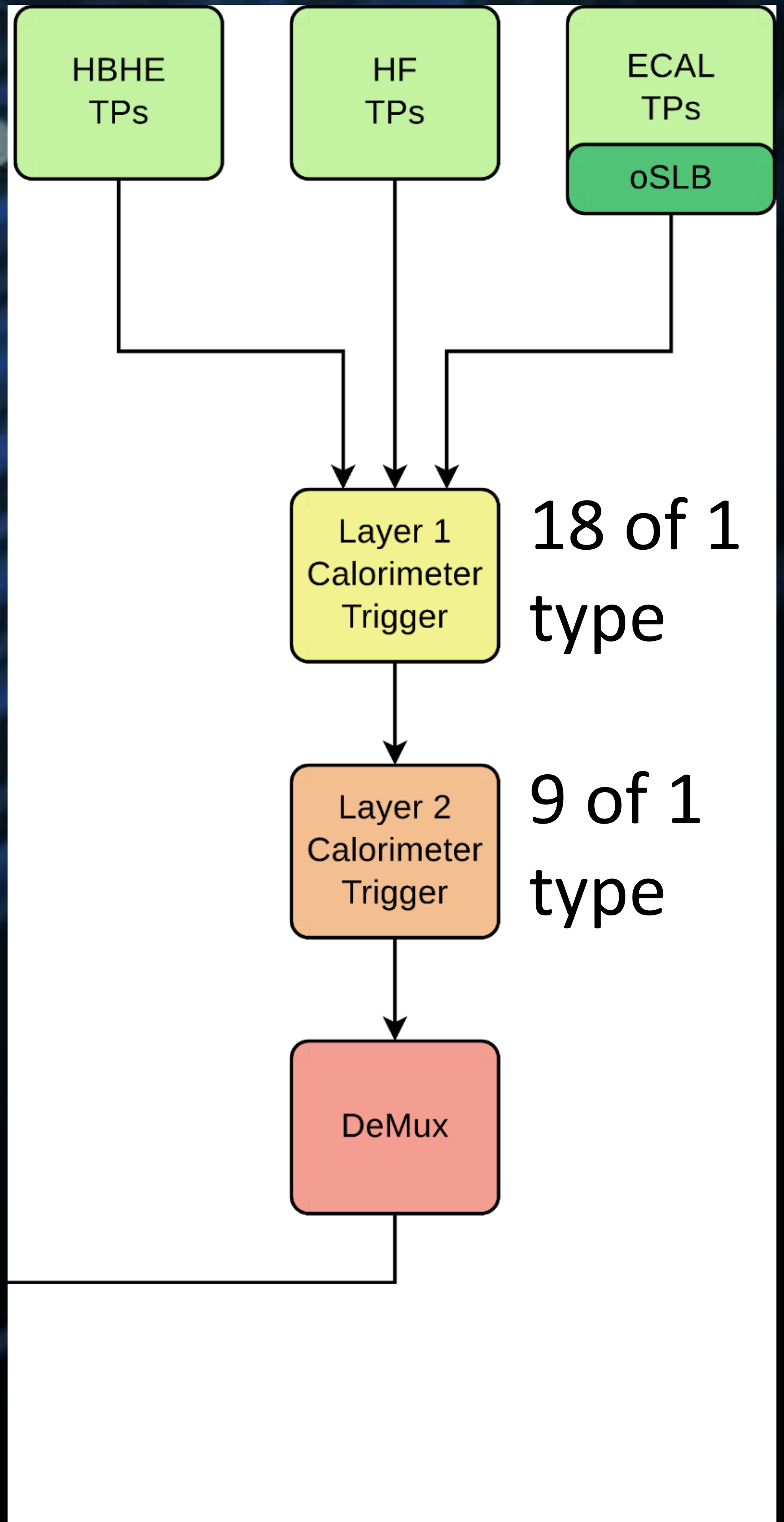
- Run 0





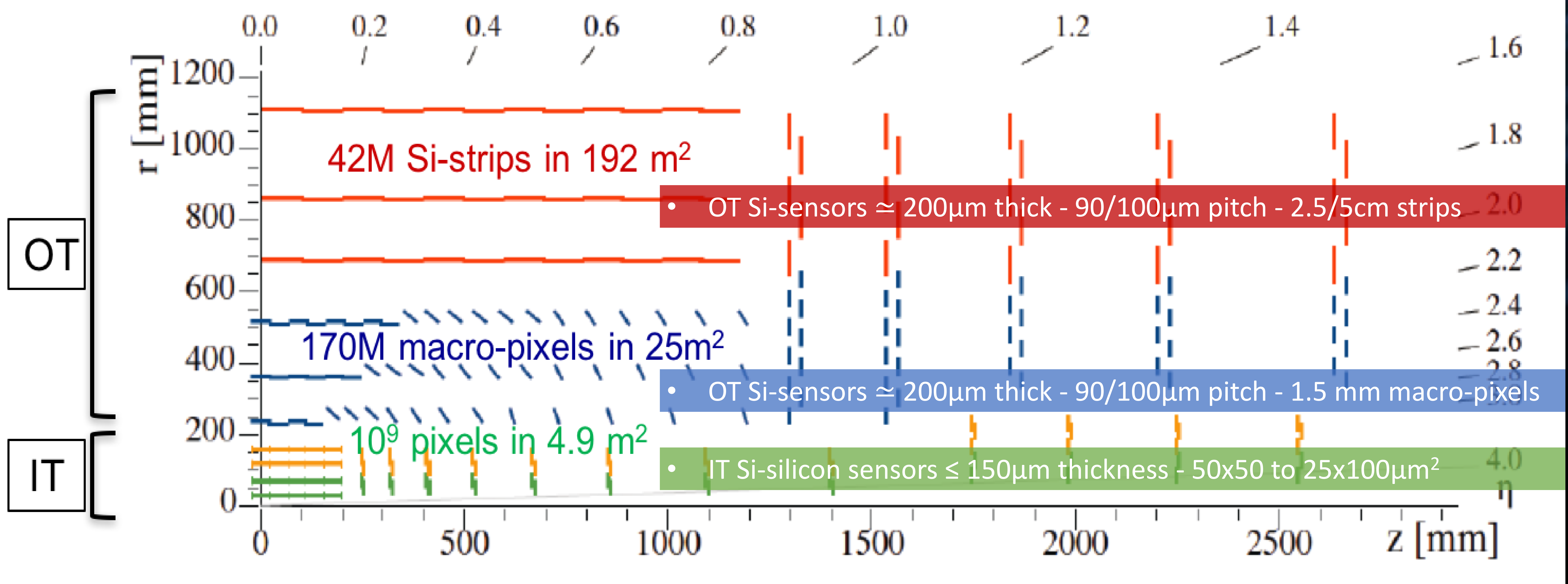


- Run 1



Tracker

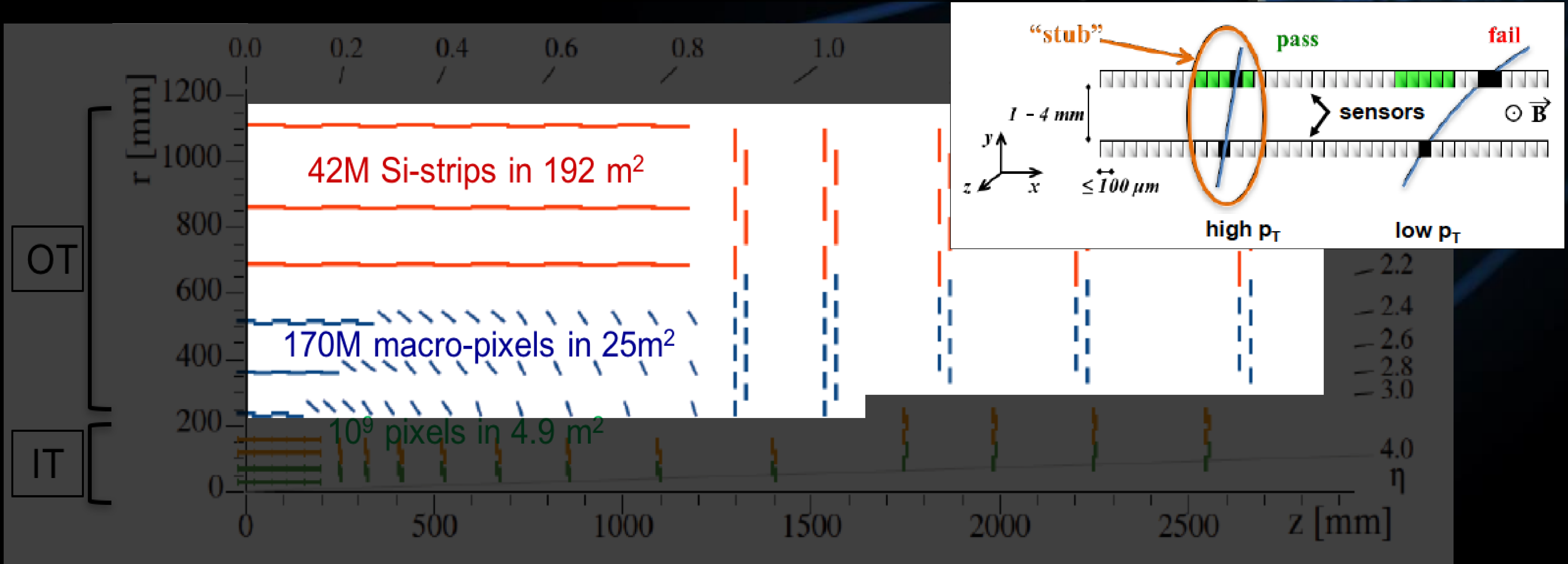
- Inner Tracker (pixel) design to extend coverage to $\eta \simeq 3.8$



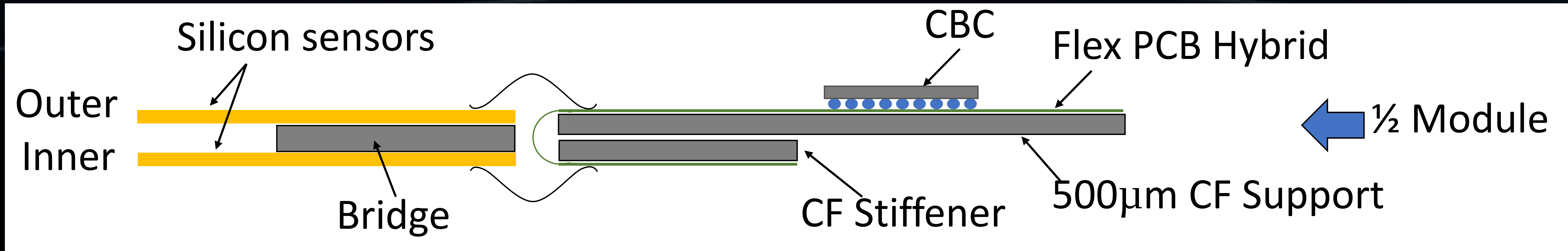
- OT Si-sensors $\simeq$ 200μm thick - 90/100μm pitch - 2.5/5cm strips
- OT Si-sensors $\simeq$ 200μm thick - 90/100μm pitch - 1.5 mm macro-pixels
- IT Si-silicon sensors ≤ 150μm thickness - 50x50 to 25x100μm^2

Tracker

- Inner Tracker (pixel) design to extend coverage to $\eta \simeq 3.8$
- Outer Tracker design driven by ability to provide tracks at 40 MHz to L1-trigger



Outer tracker 2S modules

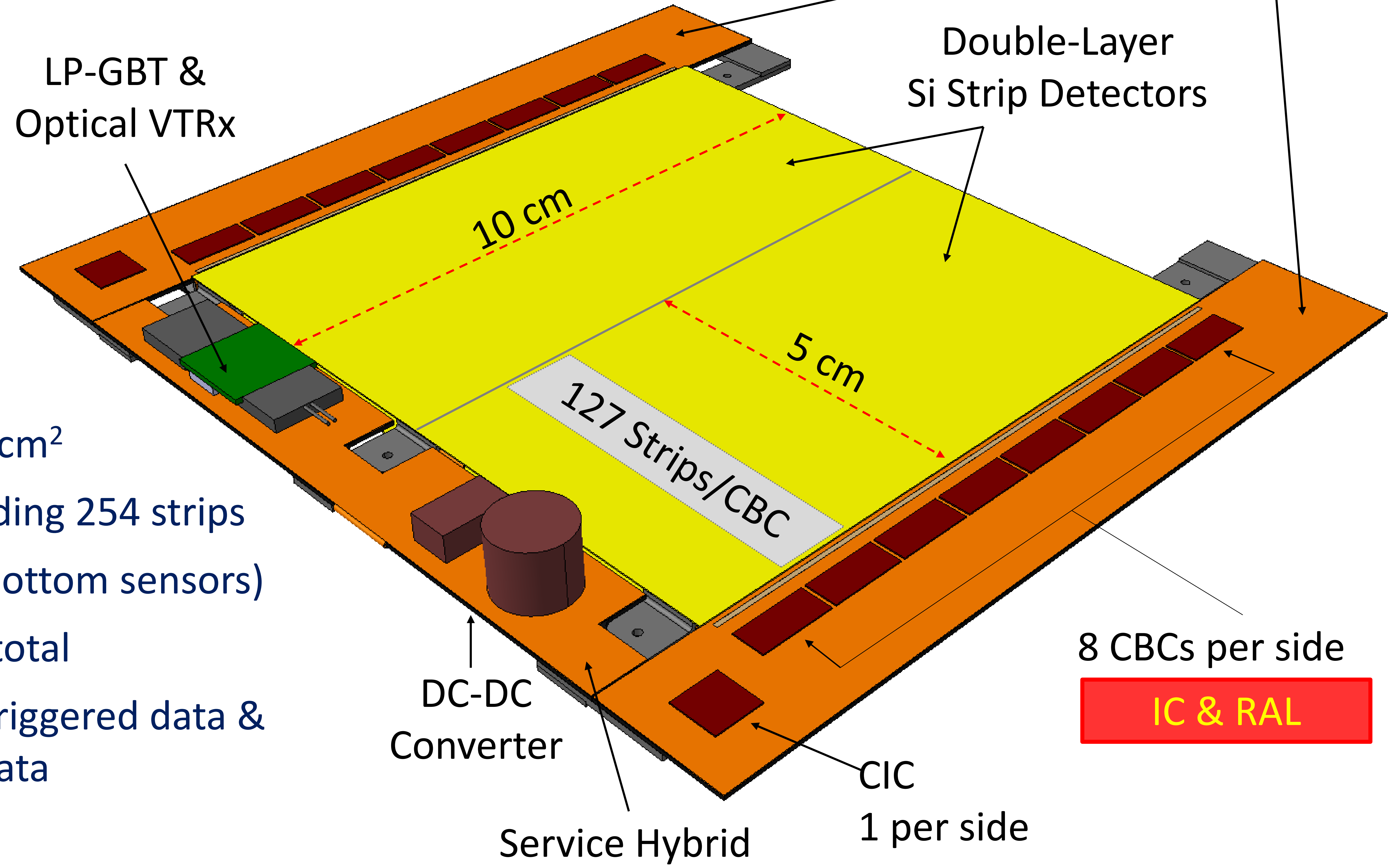


- 2S Modules: Two-strip double-layers
- ~10k modules
- 42M channels

~5,000 modules @ 10Gb/s
+ ~10,000 modules @ 5Gb/s
= 100Tb/s = 394 EB/yr

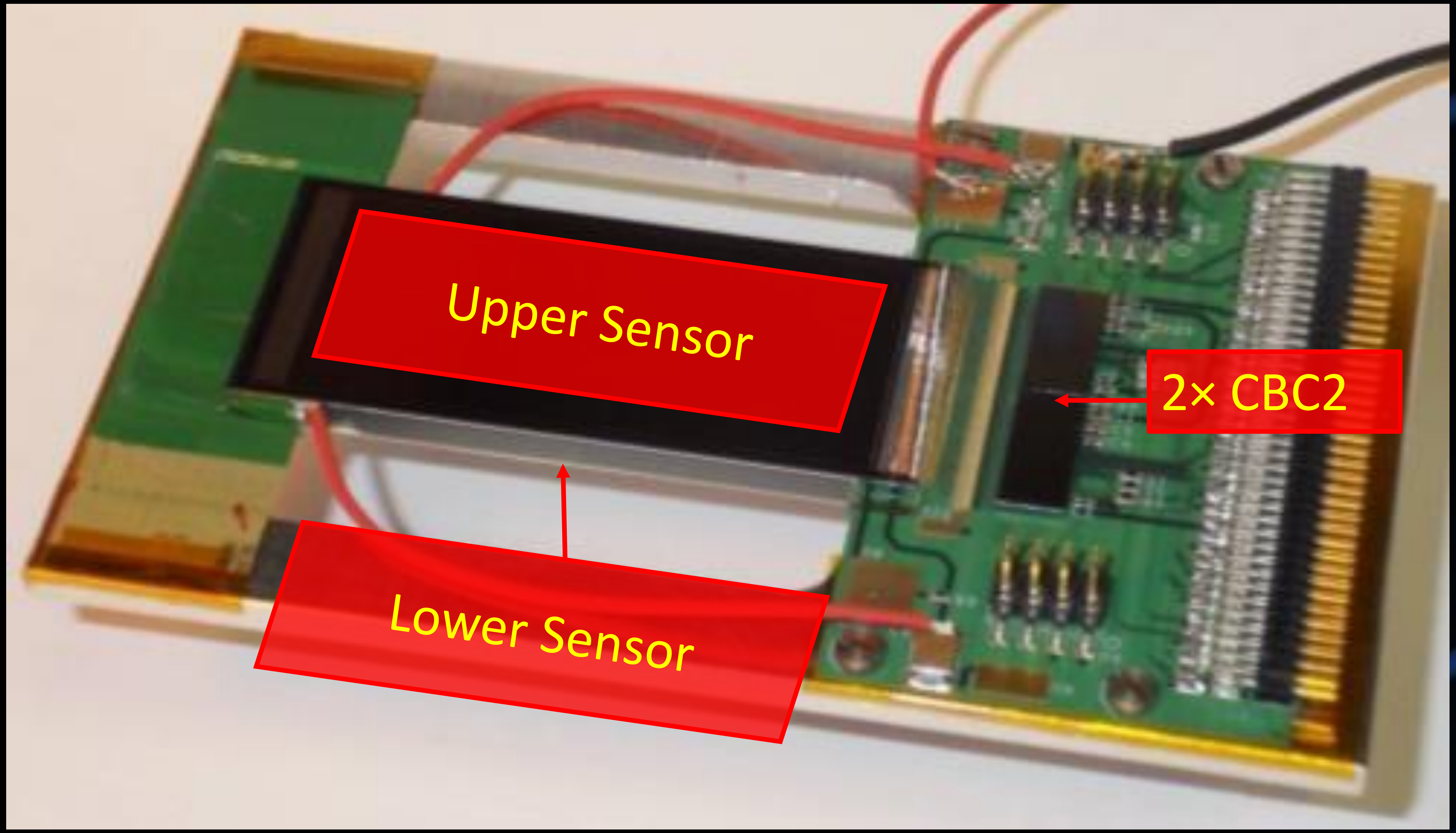
Each 2S Module:

- Sensor Area ~100 cm^2
- 16 CBCs, each reading 254 strips (127 from top & bottom sensors)
- 4064 Channels in total
- Readout both L1 triggered data & Primitive trigger data

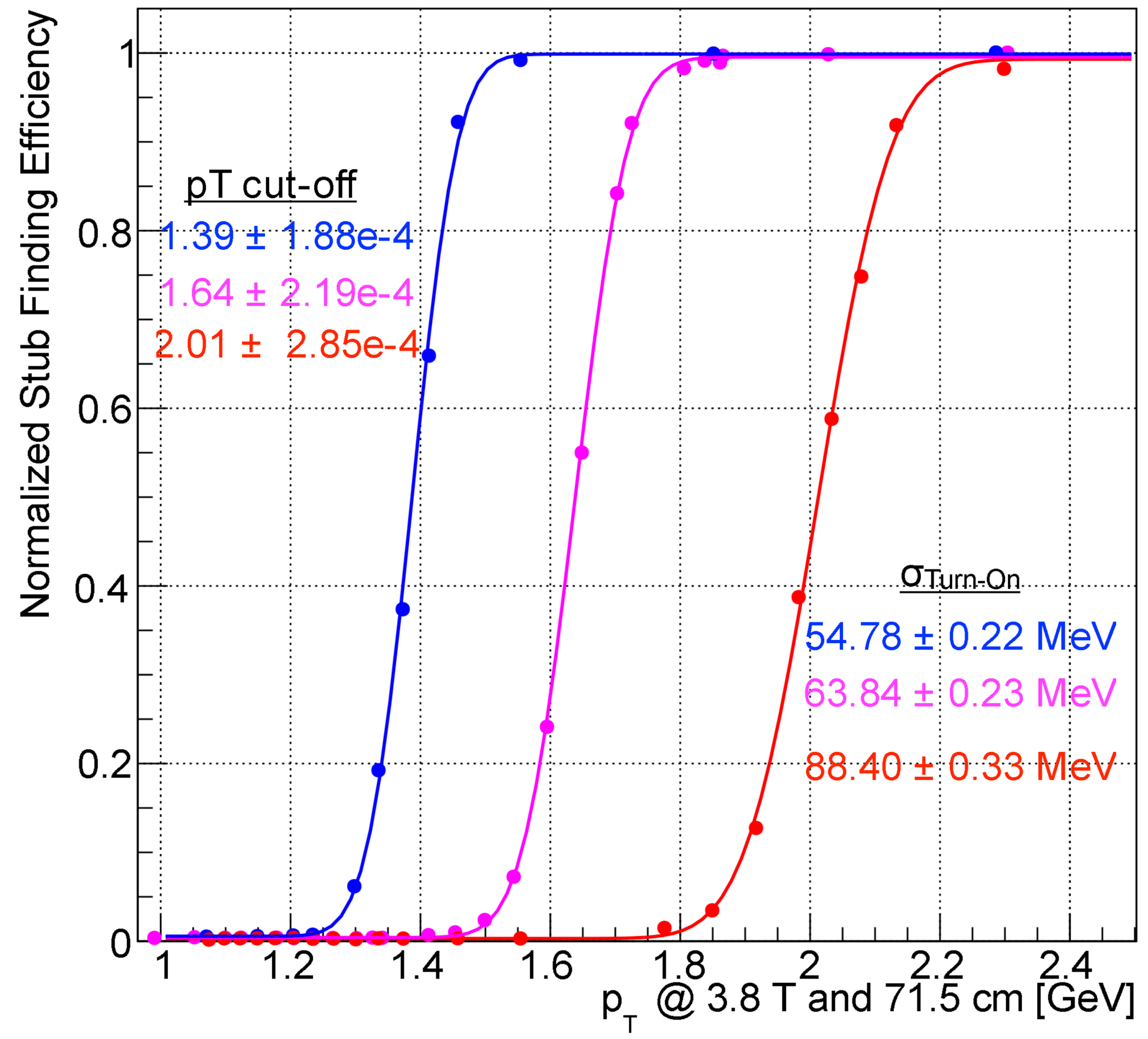


IC & RAL

Outer tracker 2S modules: Do they work?

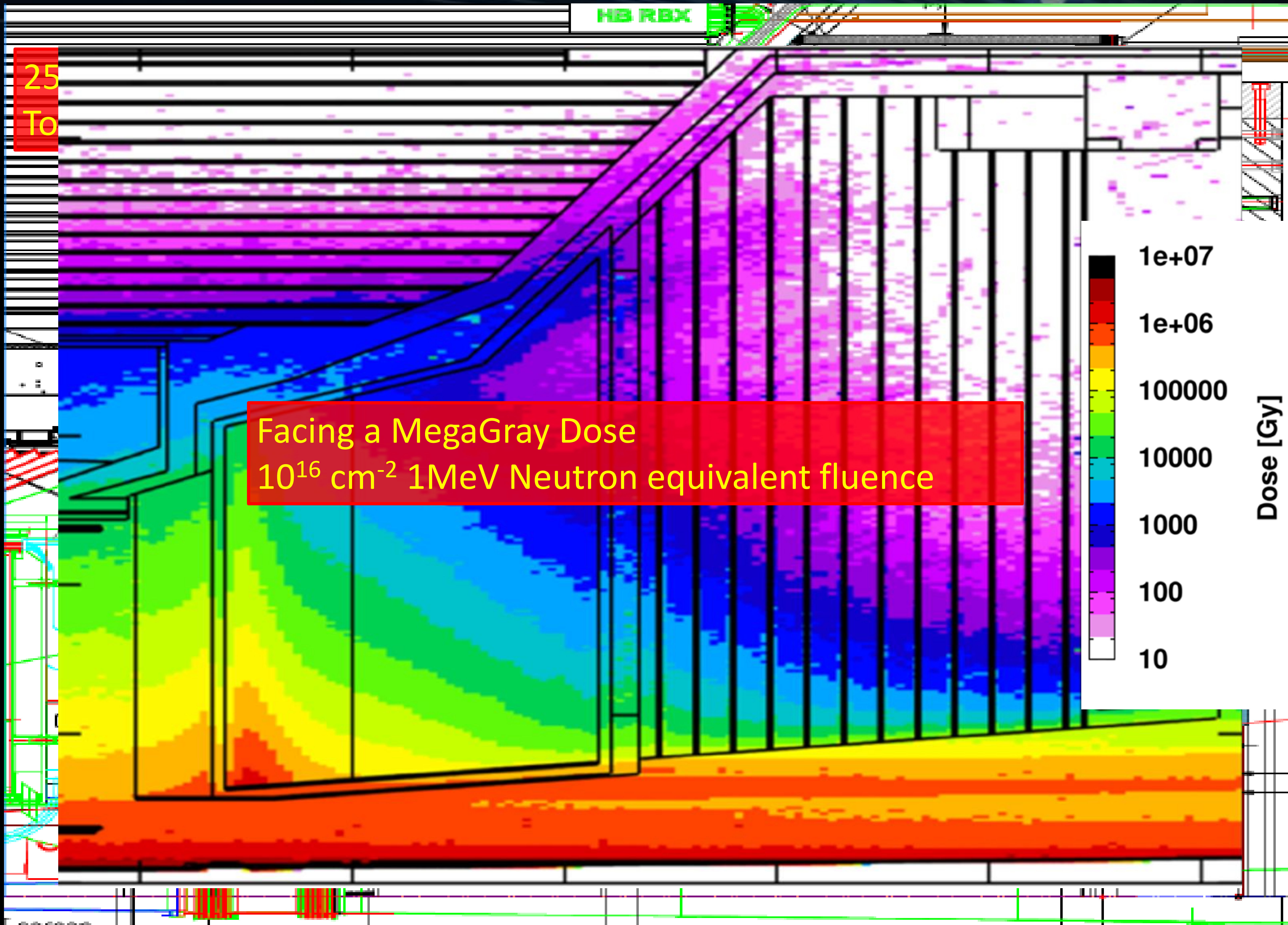


Stub turn-on curve for 2CBC mini-module at FNAL test-beam



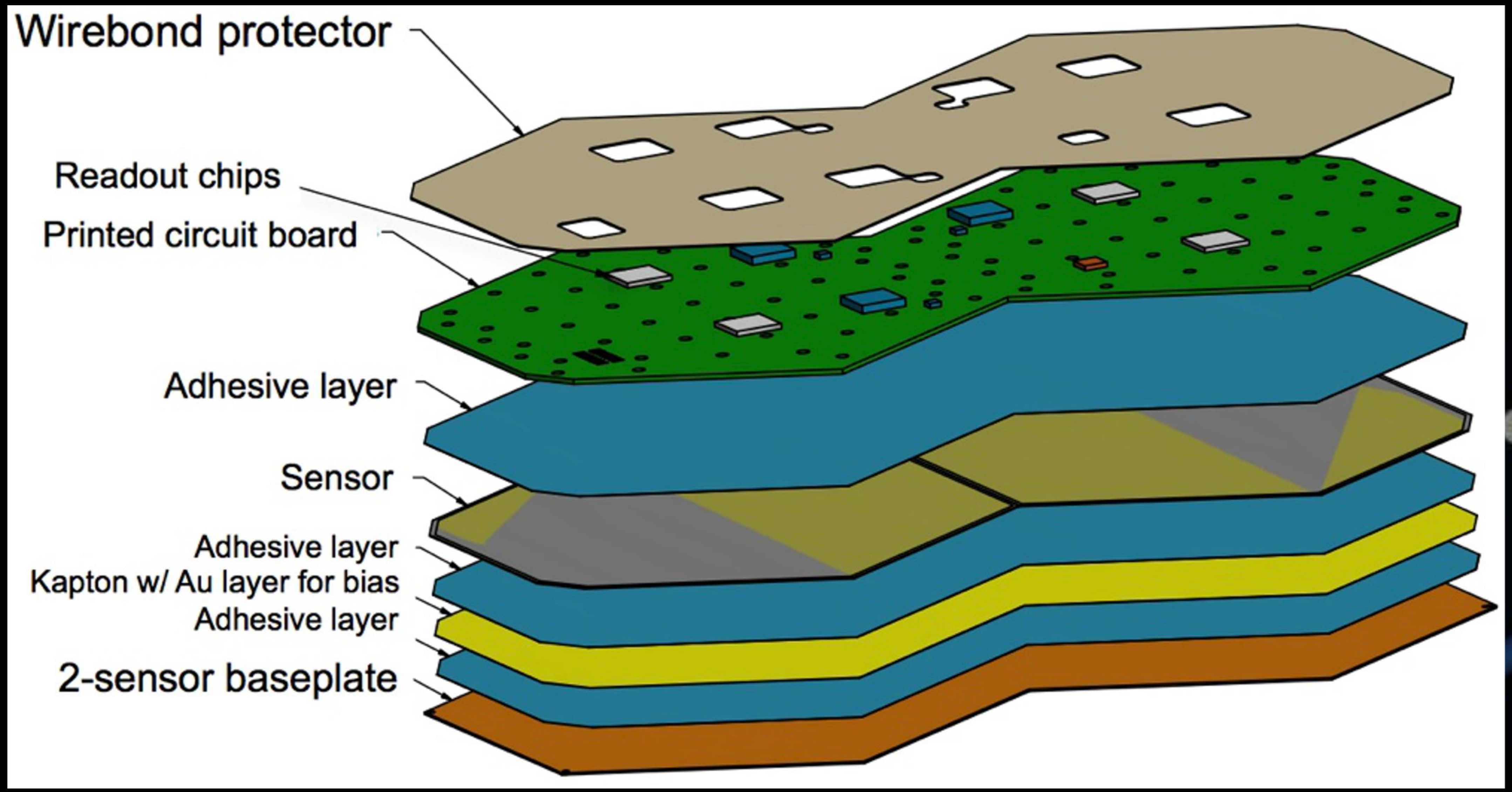
Calorimeter Endcap design

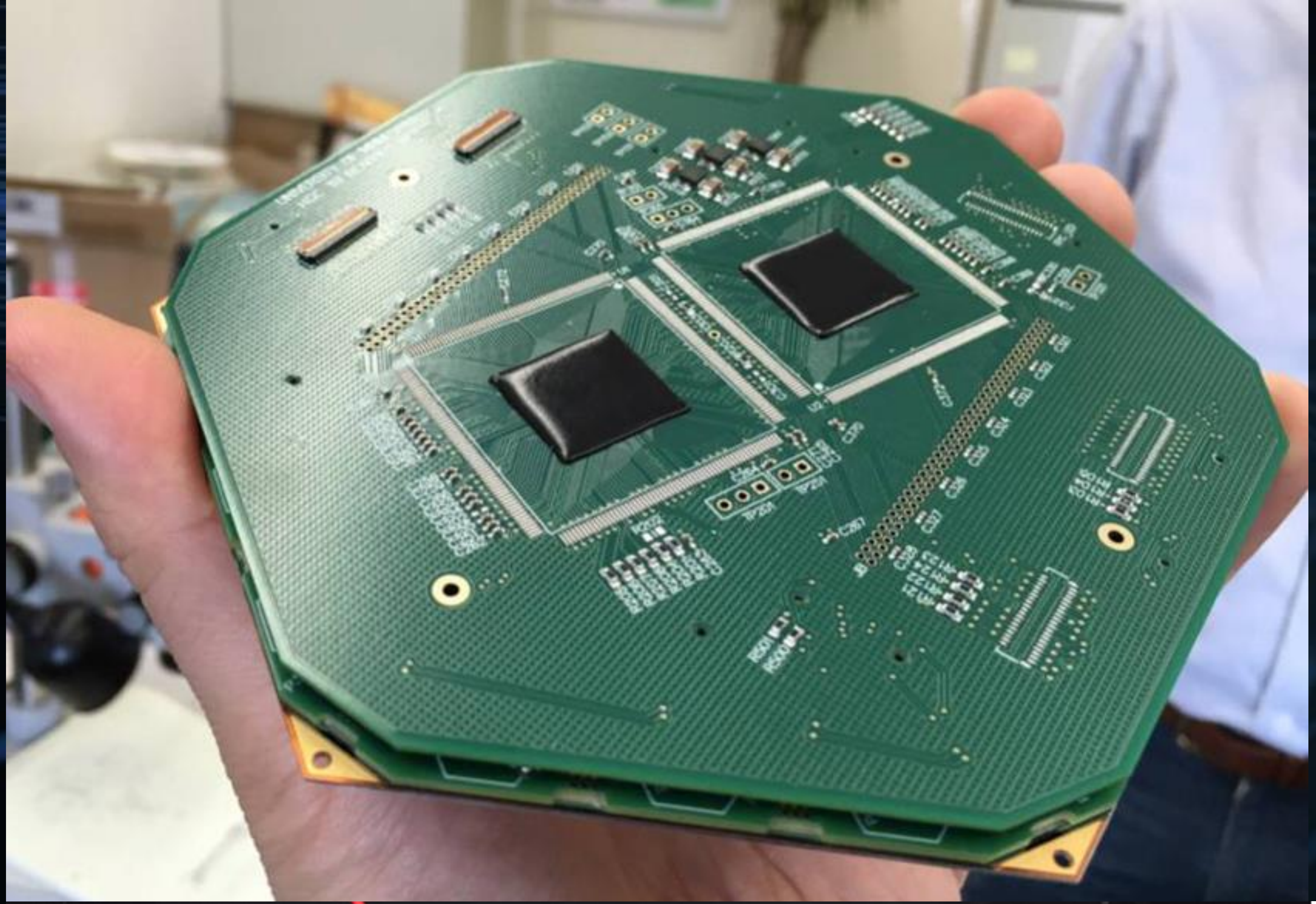
- 3D shower topology and time resolution of ~30ps
- Electromagnetic Endcap (EE)
  - 28 layers of Silicon sensors in W/Pb absorber (25 X_0, 1.7λ)
- Hadronic Endcap (EH)
  - 24 layers: 8 silicon + 16 silicon/scint. tiles at high/low η in stainless steel absorber (9λ)

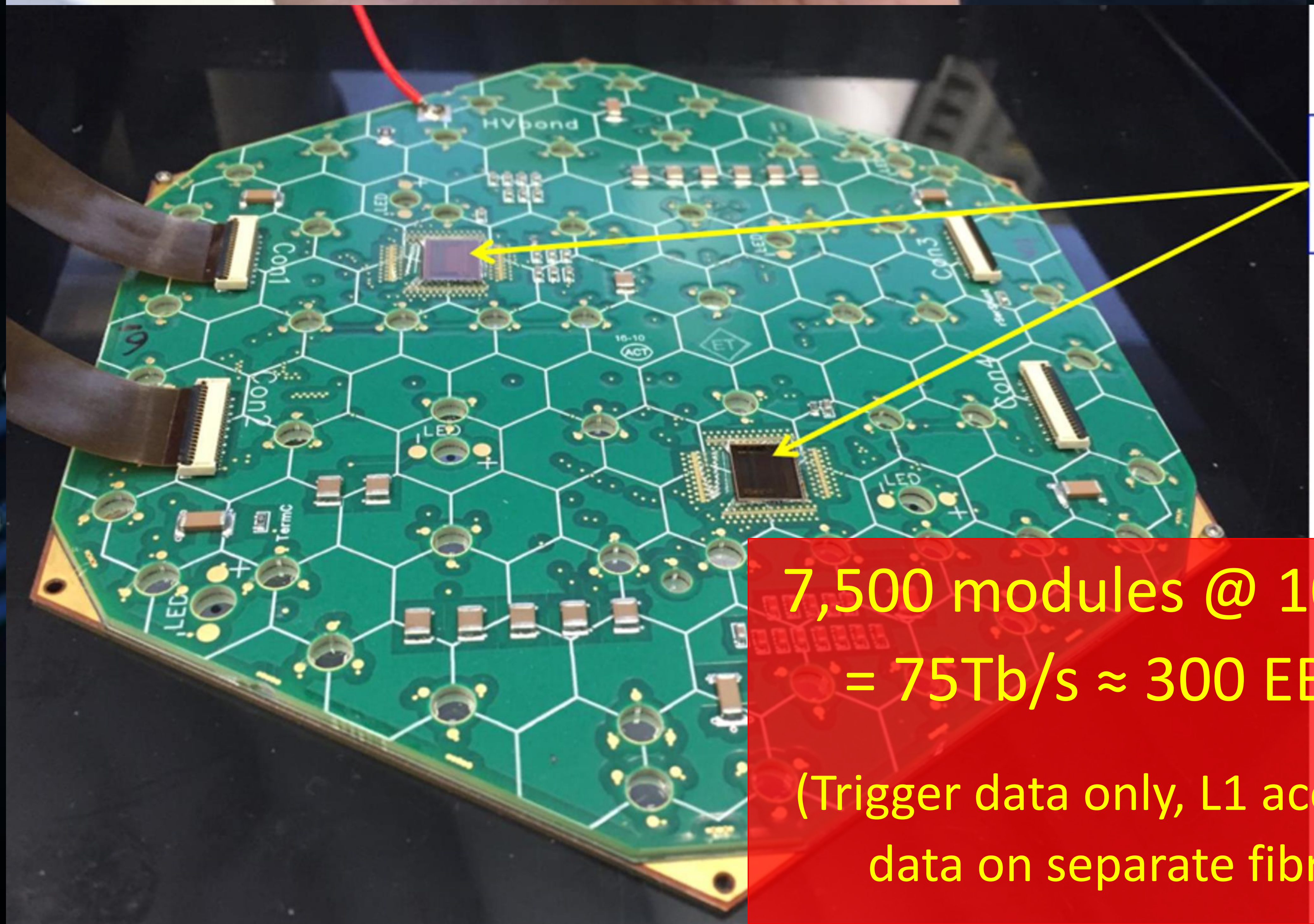


Calorimeter Endcap modules

- 593 m2 of silicon
- 6M ch, 0.5 or 1 cm2 cell-size
- 21,660 modules (8” or 2x6” sensors)
- 92,000 front-end ASICS



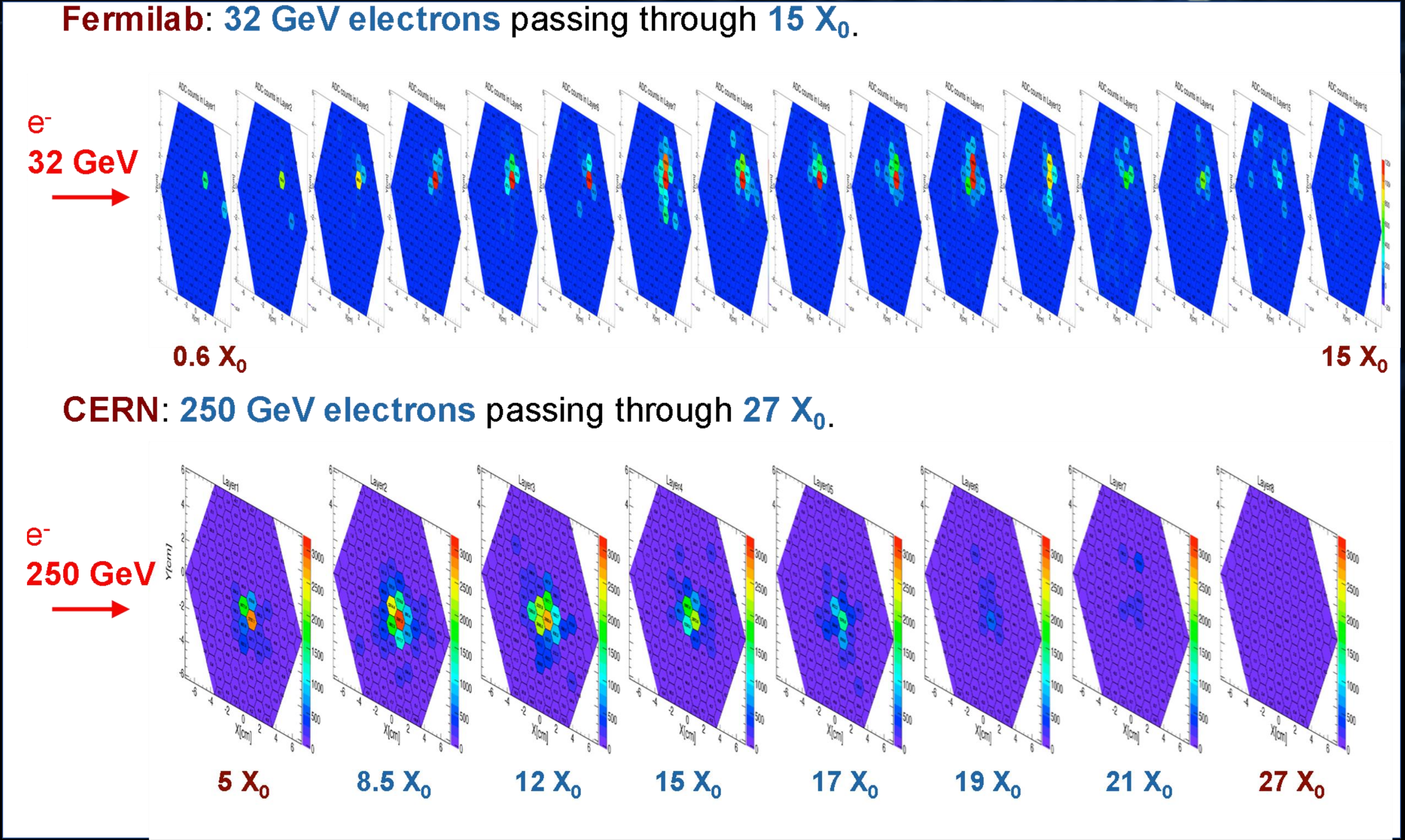




7,500 modules @ 10Gb/s = 75Tb/s ≈ 300 EB/yr

(Trigger data only, L1 accepted data on separate fibres)

Calorimeter Endcap modules: Do they work?

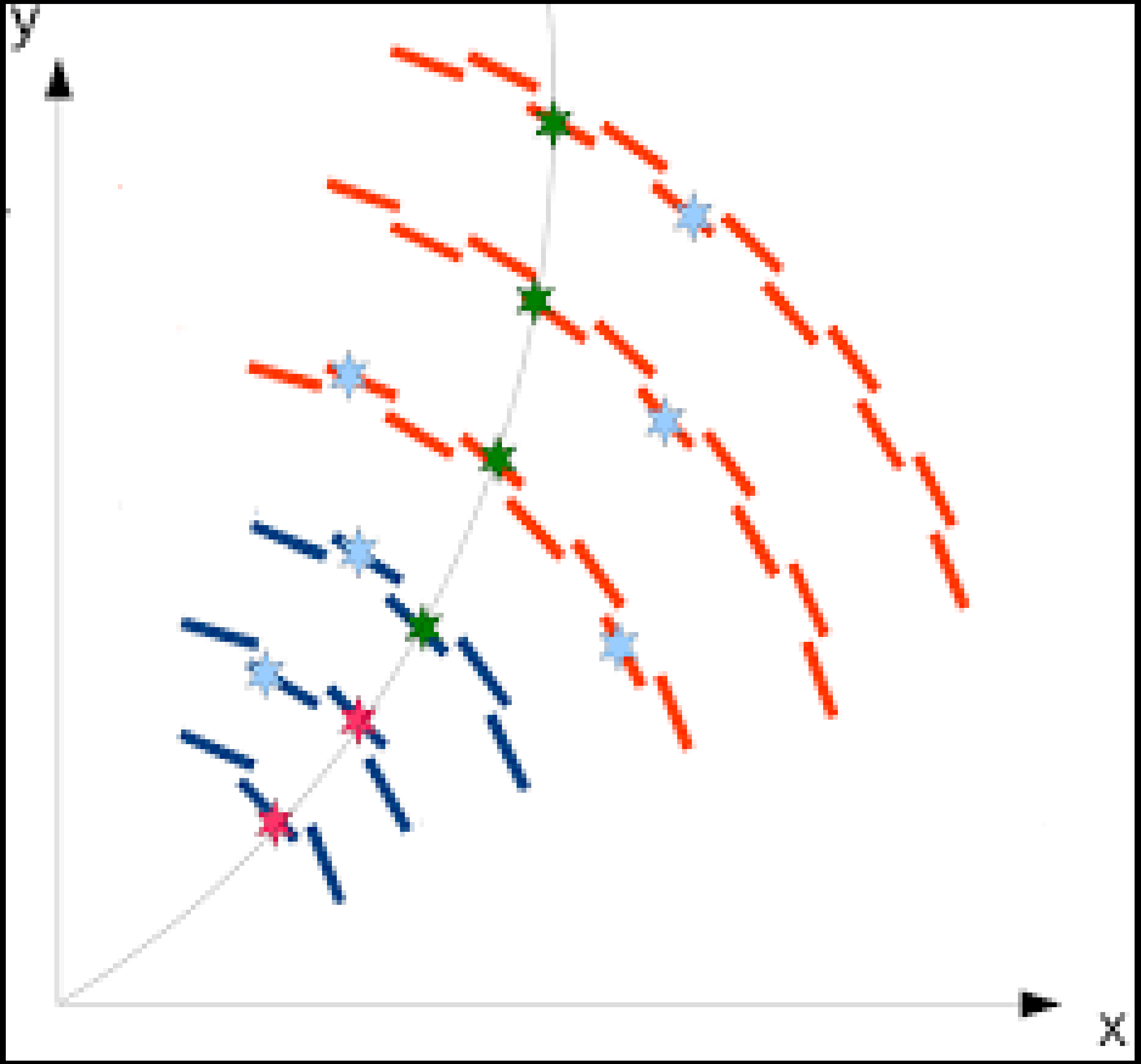


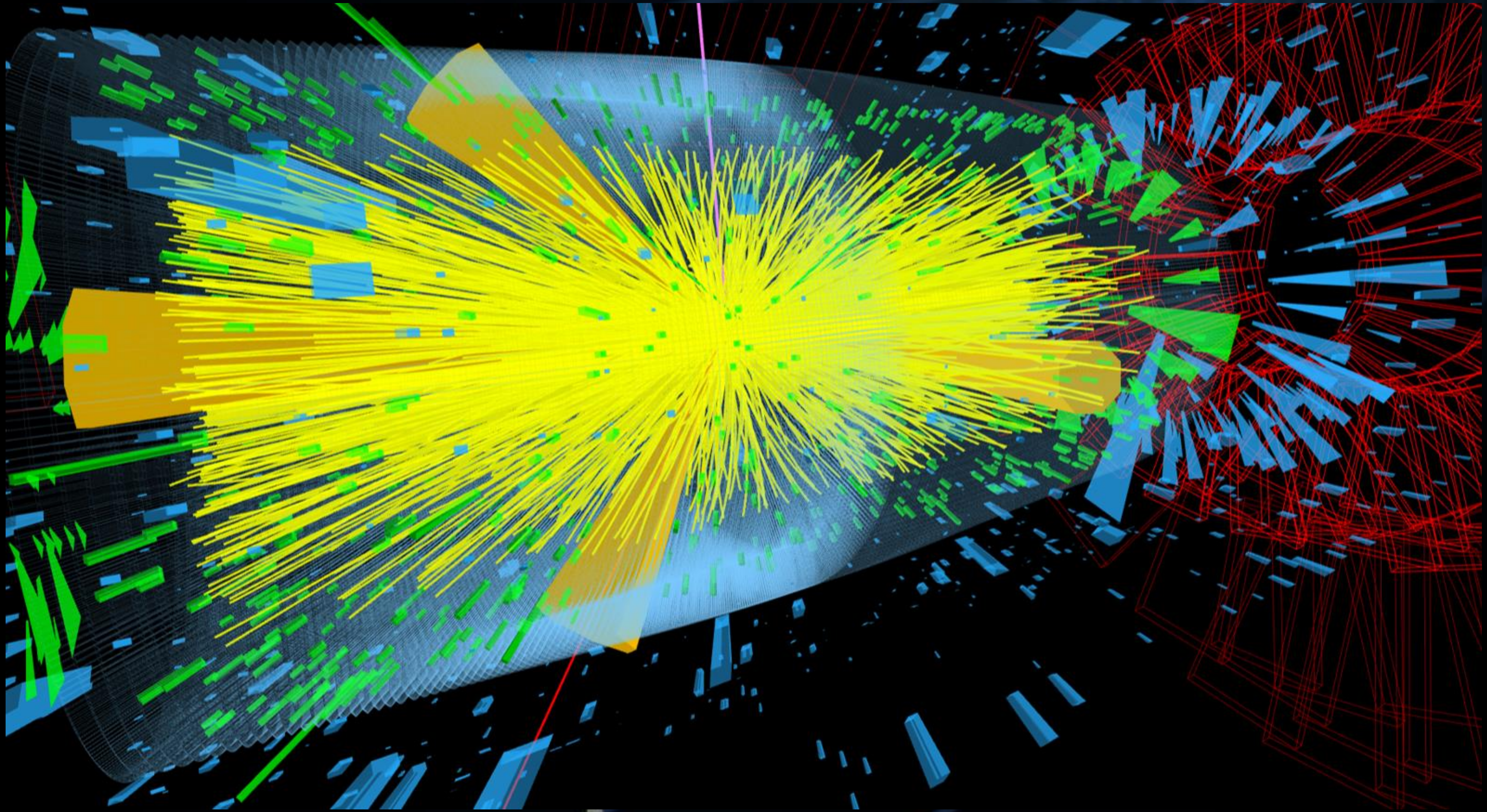
What are the firmware challenges at Phase-II?

- You mean, apart from the small matter of 300Tb/s of data?
- So much data it has to be zero-suppressed
  - No (or, at least, limited) geometric timing which can be utilized
  - Variable data-volume
    - Do you handle the worst case? Very inefficient
    - Do you handle the average? How do you handle overflows?
- We did such a good job at Phase-I, people have very high expectations…

What are the firmware challenges at Phase-II?

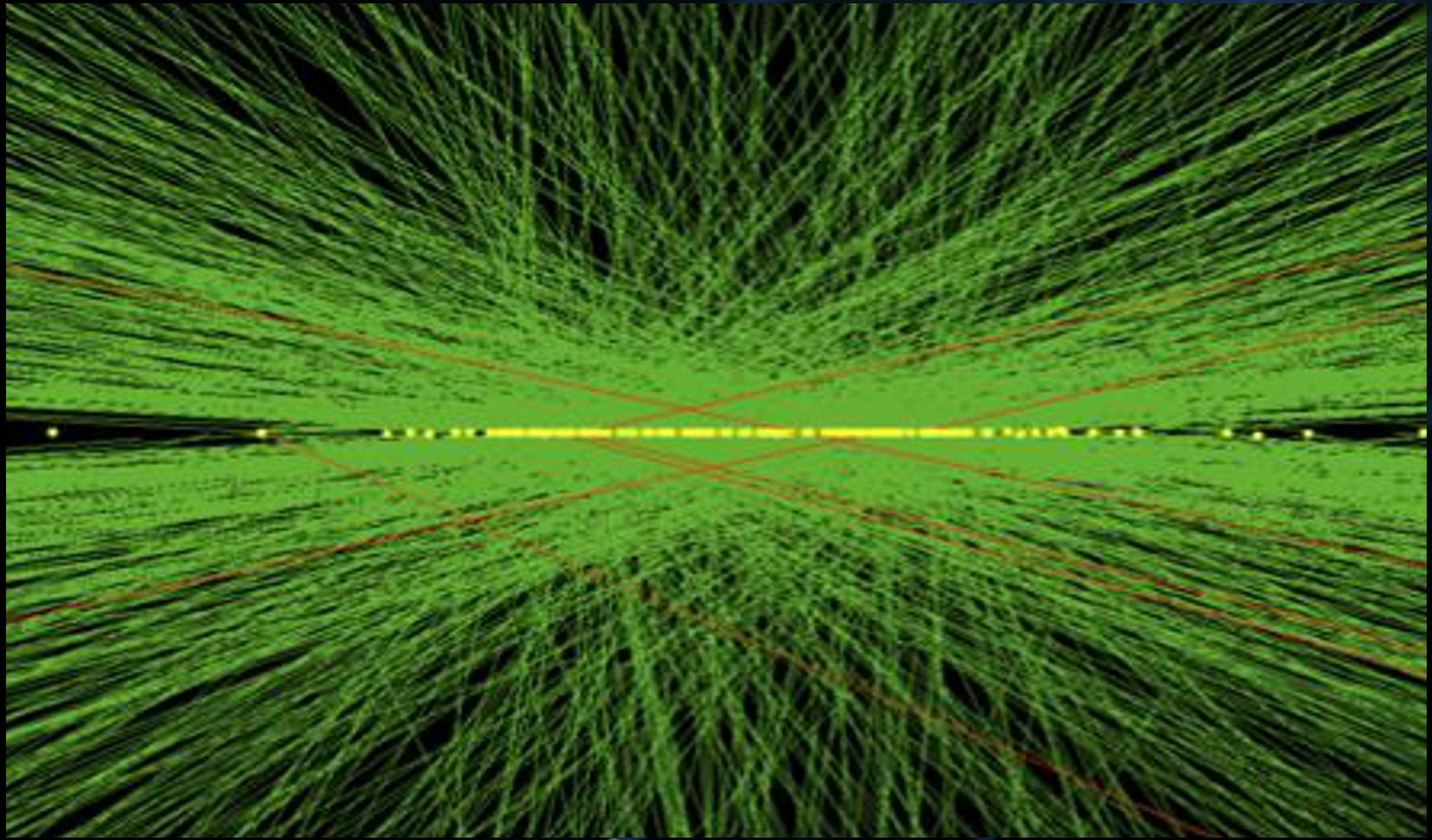
- Real-time track-finding and fitting





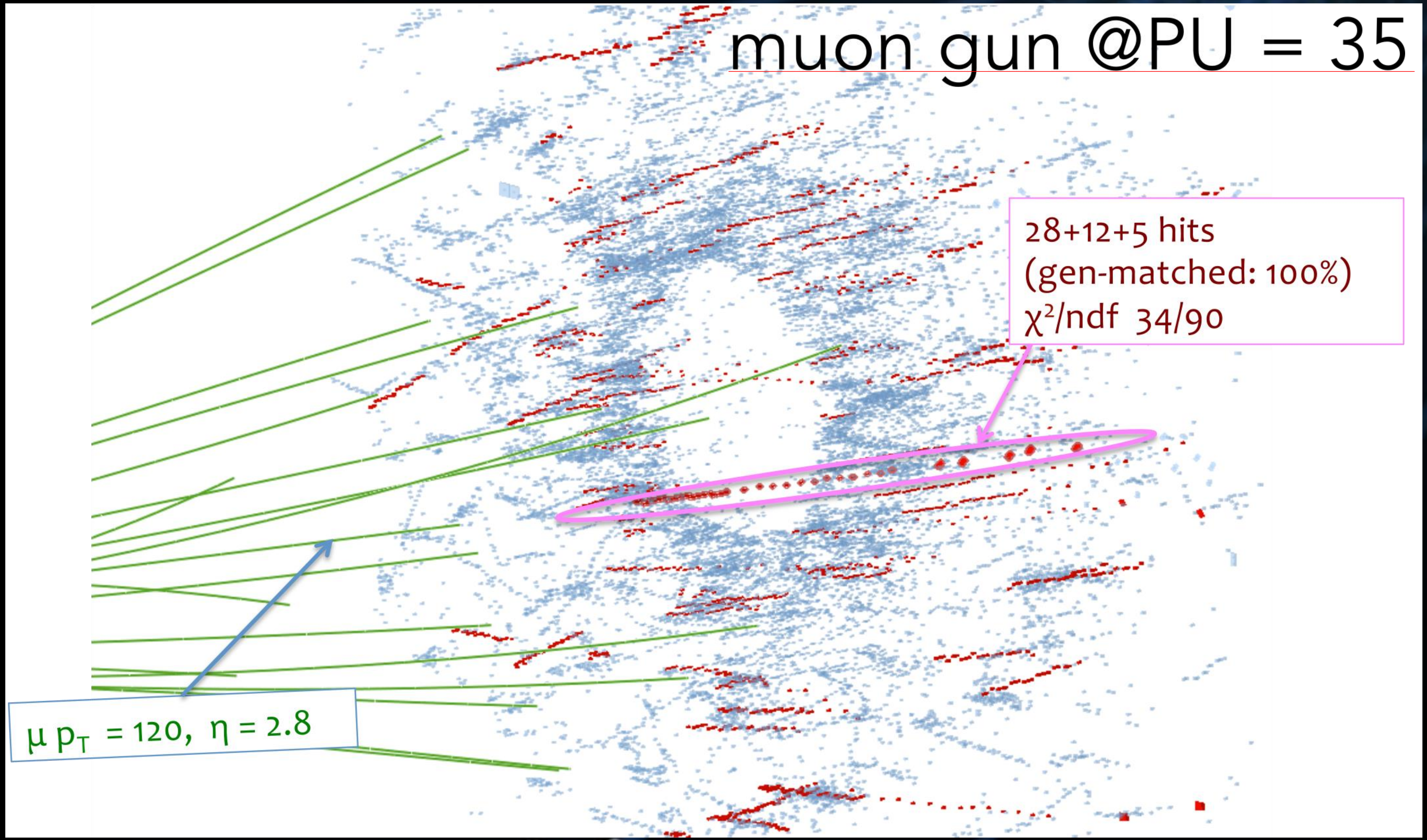
What are the firmware challenges at Phase-II?

- Real-time track-finding and fitting
- Real-time vertex-finding



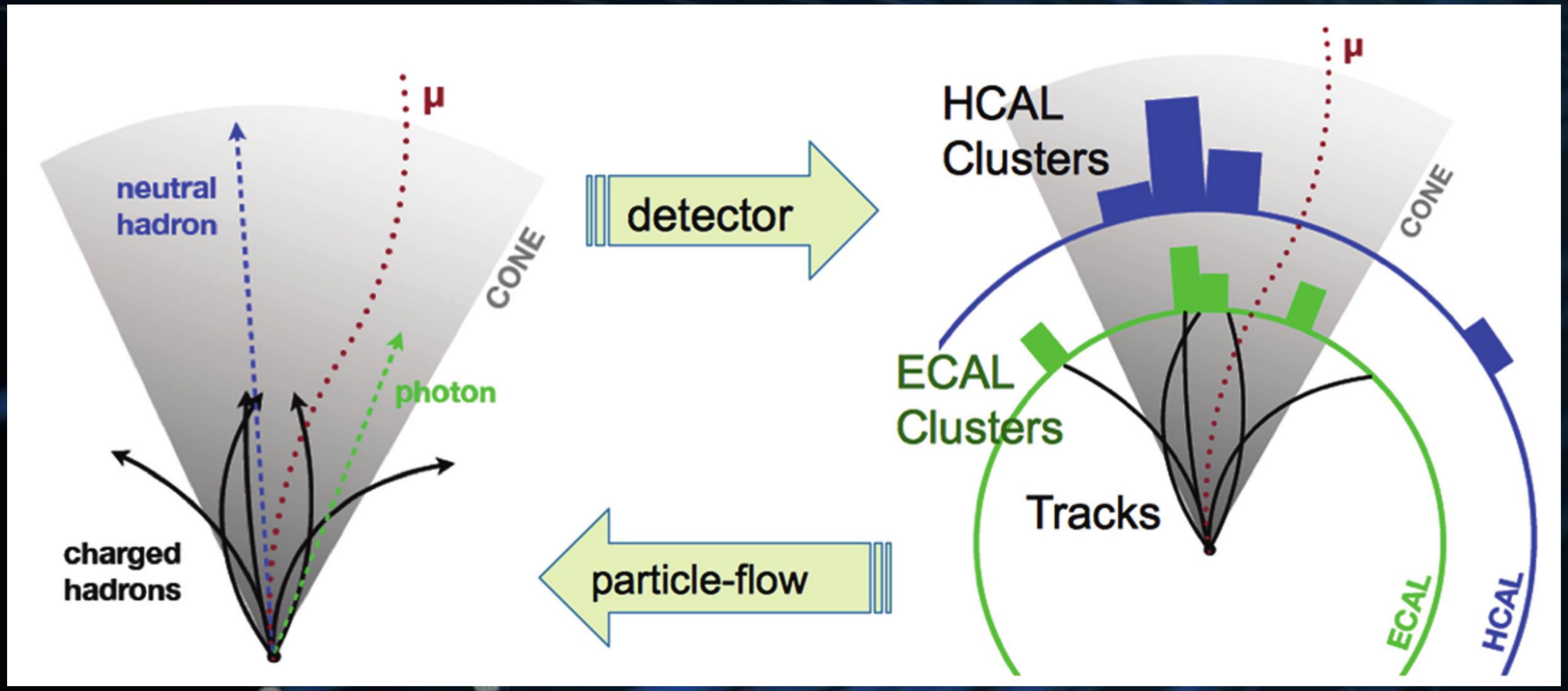
What are the firmware challenges at Phase-II?

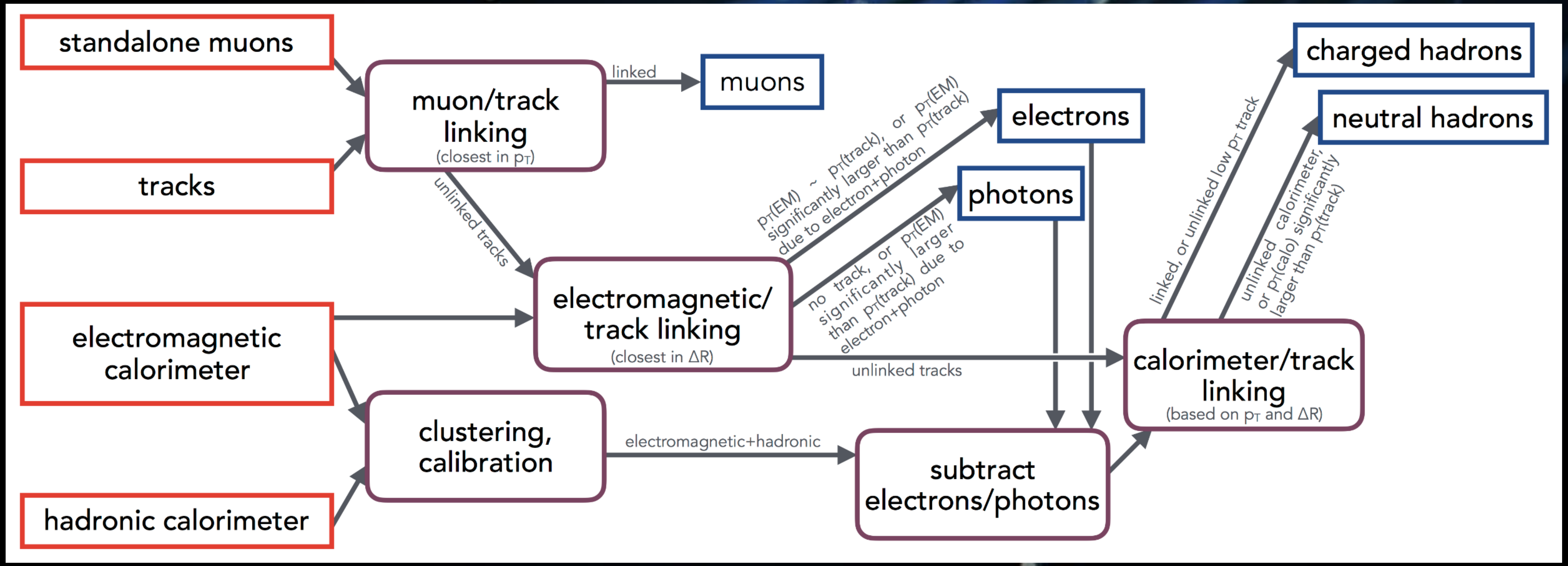
- Real-time track-finding and fitting
- Real-time vertex-finding
- 3D cluster-finding in endcap



What are the firmware challenges at Phase-II?

- Real-time track-finding and fitting
- Real-time vertex-finding
- 3D cluster-finding in endcap
- Particle-flow





What are the firmware challenges at Phase-II?

- Real-time track-finding and fitting
- Real-time vertex-finding
- 3D cluster-finding in endcap
- Particle-flow
- So, basically, they want event reconstruction

What are the firmware challenges at Phase-II?

- Real-time track-finding and fitting
- Real-time vertex-finding
- 3D cluster-finding in endcap
- Particle-flow
- So, basically, they want event reconstruction
  - in under 10µs

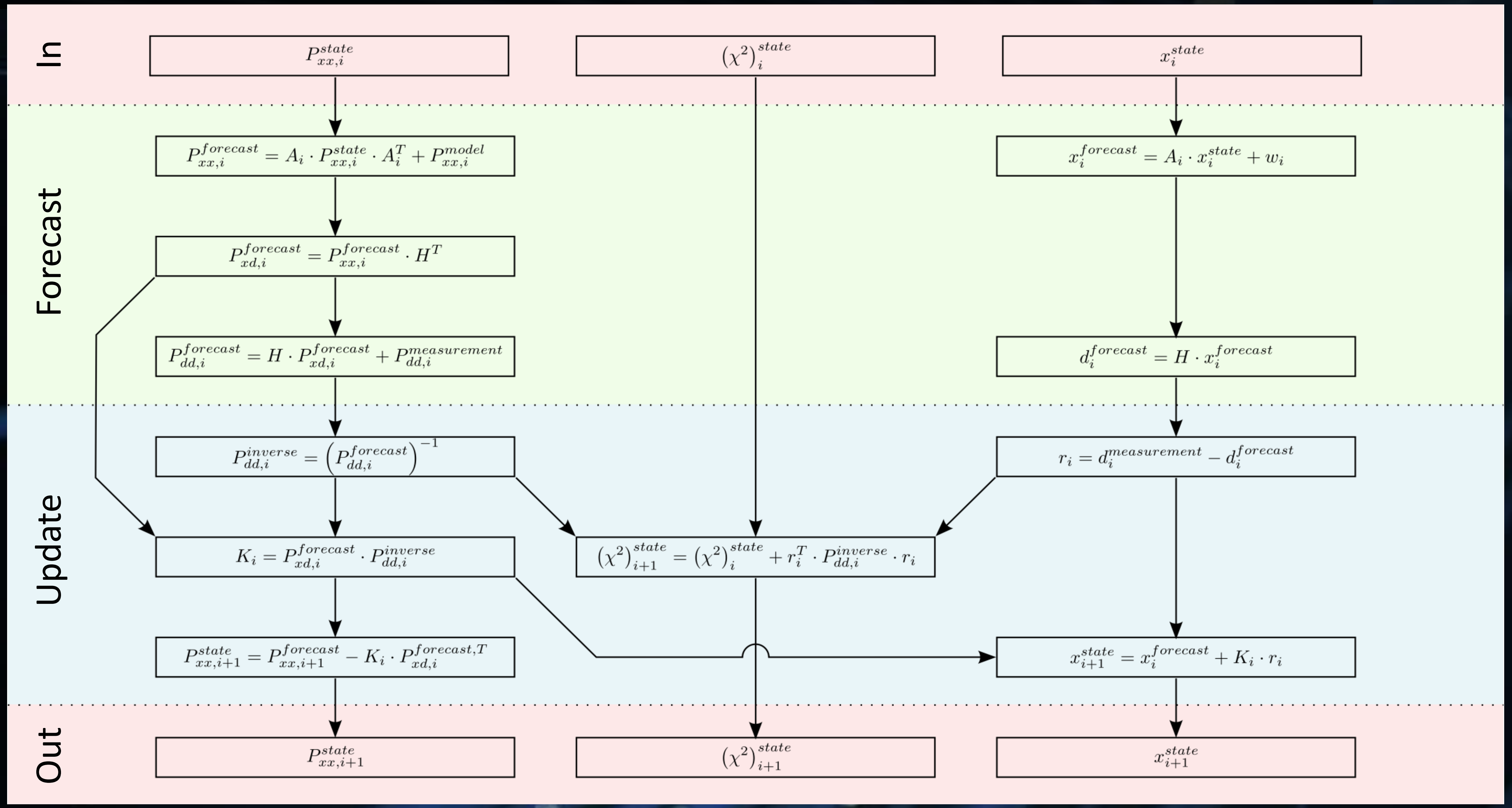
L1 cache reference/hit	1.5	ns				
Floating-point add/mult/FMA operation	1.5	ns				
L2 cache reference/hit	5	ns				
Branch mispredict	6	ns				
L3 cache hit (unshared cache line)	16	ns				
L3 cache hit (shared line in another core)	25	ns				
Mutex lock/unlock	25	ns				
L3 cache hit (modified in another core)	29	ns				
L3 cache hit (on a remote CPU socket)	40	ns				
QPI hop to a another CPU (time per hop)	40	ns				
64MB main memory reference (local CPU)	46	ns				
64MB main memory reference (remote CPU)	70	ns				
256MB main memory reference (local CPU)	75	ns				
256MB main memory reference (remote CPU)	120	ns				
Send 4KB over 100 Gbps HPC fabric	1,040	ns	1	us		
Compress 1KB with Google Snappy	3,000	ns	3	us		
Send 4KB over 10 Gbps ethernet	10,000	ns	10	us		
Write 4KB randomly to NVMe SSD	30,000	ns	30	us		
Transfer 1MB to/from NVLink GPU	30,000	ns	30	us		
Transfer 1MB to/from PCI-E GPU	80,000	ns	80	us		
Read 4KB randomly from NVMe SSD	120,000	ns	120	us		
Read 1MB sequentially from NVMe SSD	208,000	ns	208	us		
Write 4KB randomly to SATA SSD	500,000	ns	500	us		
Read 4KB randomly from SATA SSD	500,000	ns	500	us		
Round trip within same datacenter	500,000	ns	500	us		
Read 1MB sequentially from SATA SSD	1,818,000	ns	1,818	us	2	ms
Read 1MB sequentially from disk	5,000,000	ns	5,000	us	5	ms
Random Disk Access (seek+rotation)	10,000,000	ns	10,000	us	10	ms
Send packet CA->Netherlands->CA	150,000,000	ns	150,000	us	150	ms

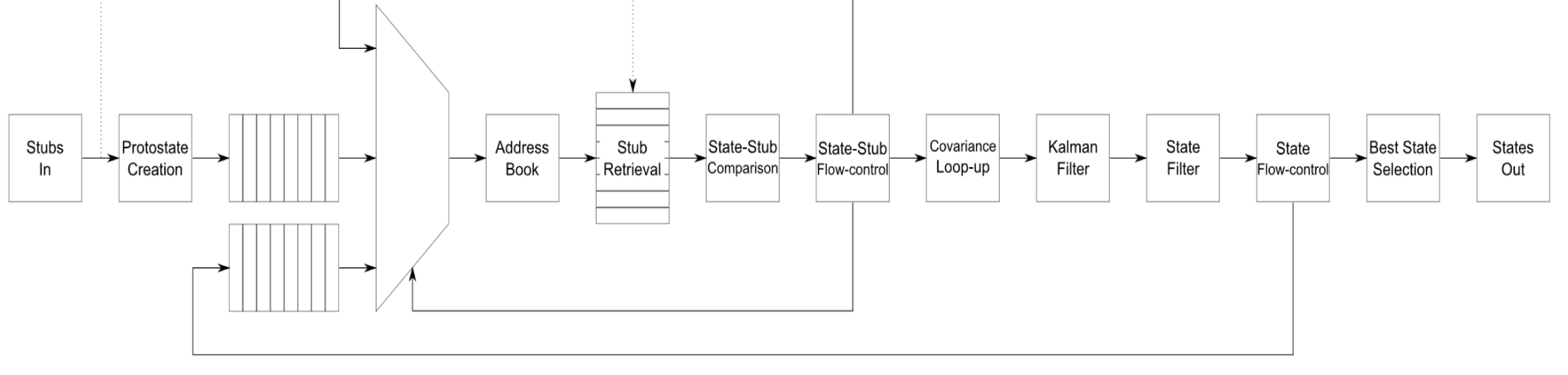
And we have stepped up to the plate!

- I used to give the following firmware advice to my students on preserving their sanity:
  - Avoid iterative algorithms
  - Avoid combinatorics
  - Make the data-flow deterministic
  - Division is hard, resource hungry and latency intensive
  - Floating-point is hard, resource hungry and latency intensive

And we have stepped up to the plate!

- For track fitter we wrote a full Kalman Filter
  - Which is iterative
  - Results in combinatorics
  - Is data-dependent (pseudo-deterministic)
  - Requires a division
  - We also played with floating-point but dropped it in the final version



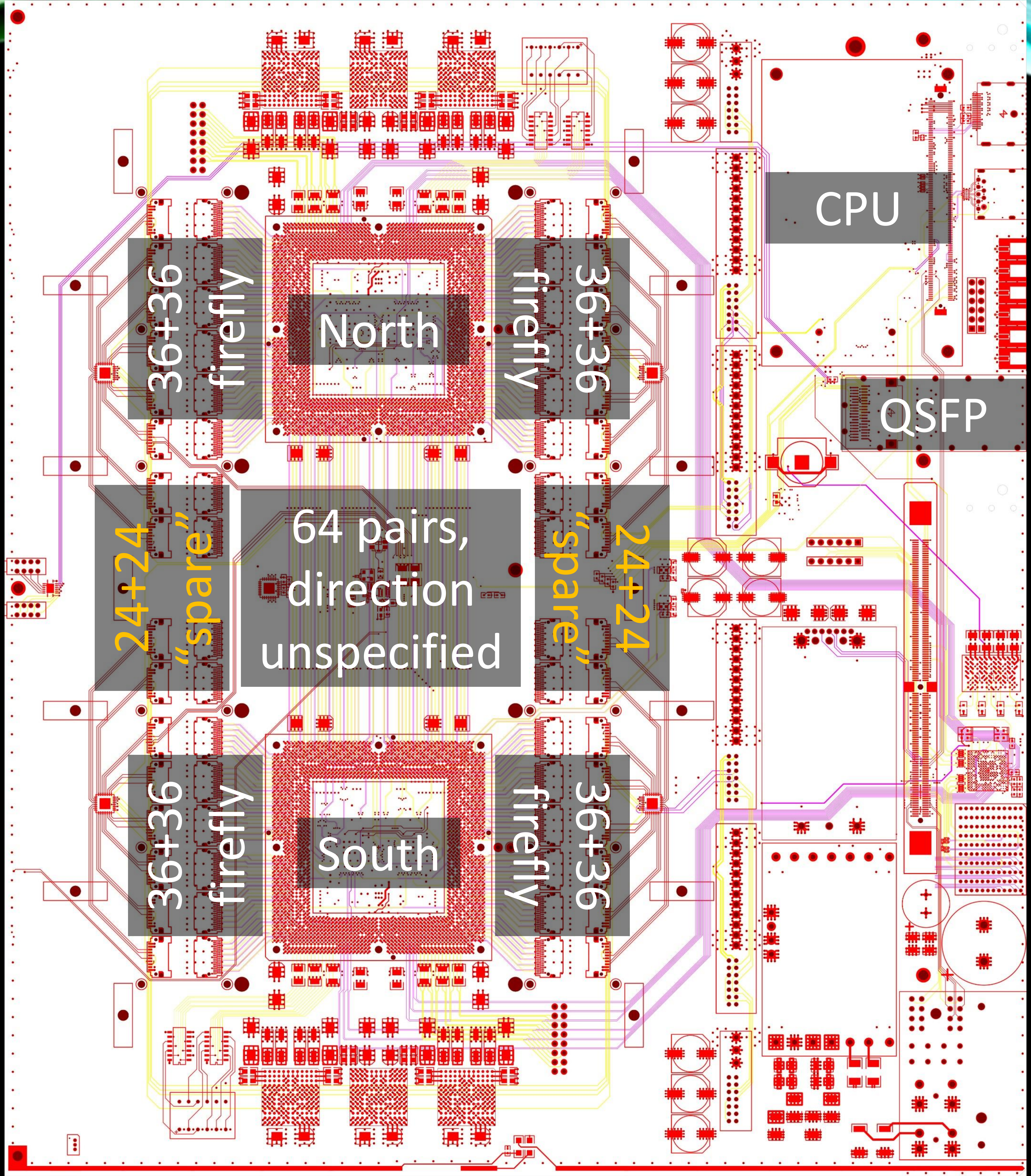


And we have stepped up to the plate!

- Fits a 4-point track in 1.5μs
- Matches offline floating-point resolution

COMMON HW: SERENITY

- Standard configuration. Provision for:
  - 2 × 72+72 links @ 28Gbps optical
    - 4 + 4 Tbps
  - 64 links @ 32Gbps between DCs
    - 2 Tbps
- Optional optical expansion:
  - 2 × 96+96 links @ 28Gbps optical
    - 5.375 + 5.375 Tbps
  - 16 links @ 32Gbps between DCs
    - 0.5 Tbps



COMMON HW: SERENITY

- Freedom
  - Choose your preferred family, package, generation, (vendor?)
  - Choose your balance of optical and electrical connectivity
- Reduces financial risk
  - Carrier (bulk of potential failure-modes) qualified before FPGAs (bulk of the cost) are fitted!