EIC readout proposal (Part 1)

Jan C. Bernauer

Massachusetts Institute of Technology

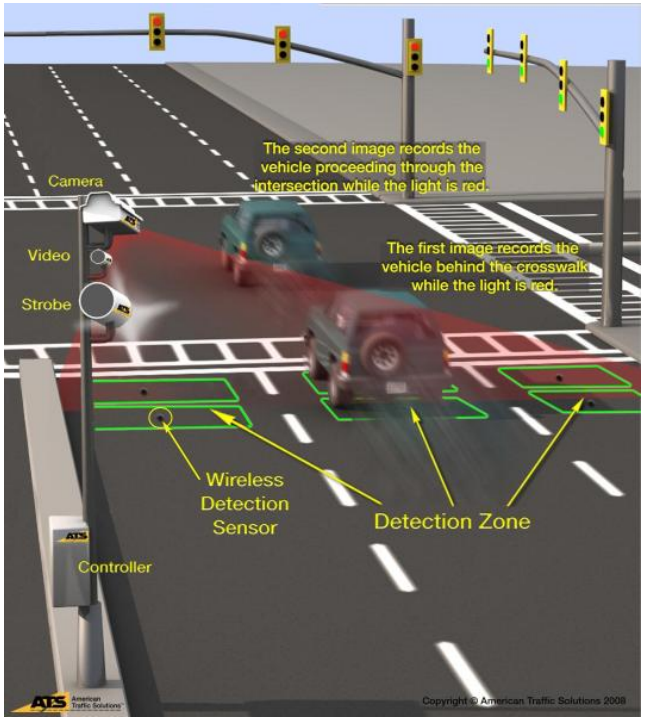
EIC Generic Detector R&D Advisory Committee Meeting

A typical experiment....

... works like a red light camera trap:

A typical experiment....

... works like a red light camera trap:



1. Trigger: Maybe something interesting happened.
   - Something went over the line when the light was red.

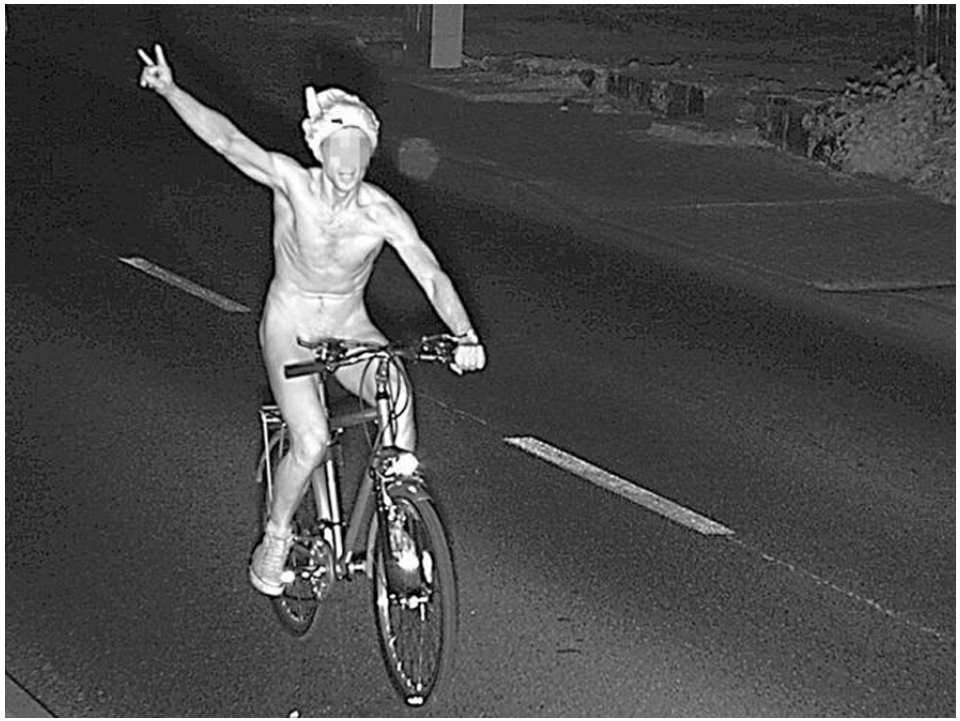
A typical experiment....

... works like a red light camera trap:

2. Data readout: We make a snapshot of everything.
   - But we might miss a second car because the camera isn't ready again.

A typical experiment....

... works like a red light camera trap:



3. Analysis: Sort out the (few) bad cases. Send tickets.
   - Ambulances and fire trucks

Looking for the black swan

EIC will have high luminosity:

- High rates for common events
- Detect rare events

Looking for the black swan

EIC will have high luminosity:

- High rates for common events
- Detect rare events, we might not be able to **trigger** on

Looking for the black swan

EIC will have high luminosity:

- High rates for common events
- Detect rare events, we might not be able to **trigger** on

Make a movie!

- Continously record *everything*.
- Cannot save everything
  $\Longrightarrow$(proto) analyse data on the fly and save:
  - salient information (high rate)
  - interesting events (rare events)

Looking for the black swan

EIC will have high luminosity:

- High rates for common events
- Detect rare events, we might not be able to **trigger** on

Make a movie!

- Continously record *everything*.
- Cannot save everything
  $\Longrightarrow$(proto) analyse data on the fly and save:
  - salient information (high rate)
  - interesting events (rare events)

Fast readout system

- continuous, trigger-less, high-rate capable
- scaling from small to large channel counts, data rates
- (close to) deadtime free
- solve *transposition problem*

The transposition problem

- High channel count and rates $\Longrightarrow$ highly parallel readout
- We have: all data for **one** channel for **all** times.

The transposition problem

- High channel count and rates $\Longrightarrow$ highly parallel readout
- We have: all data for **one** channel for **all** times.
- We need: all data for **all** channels for **one** time segment.

The transposition problem

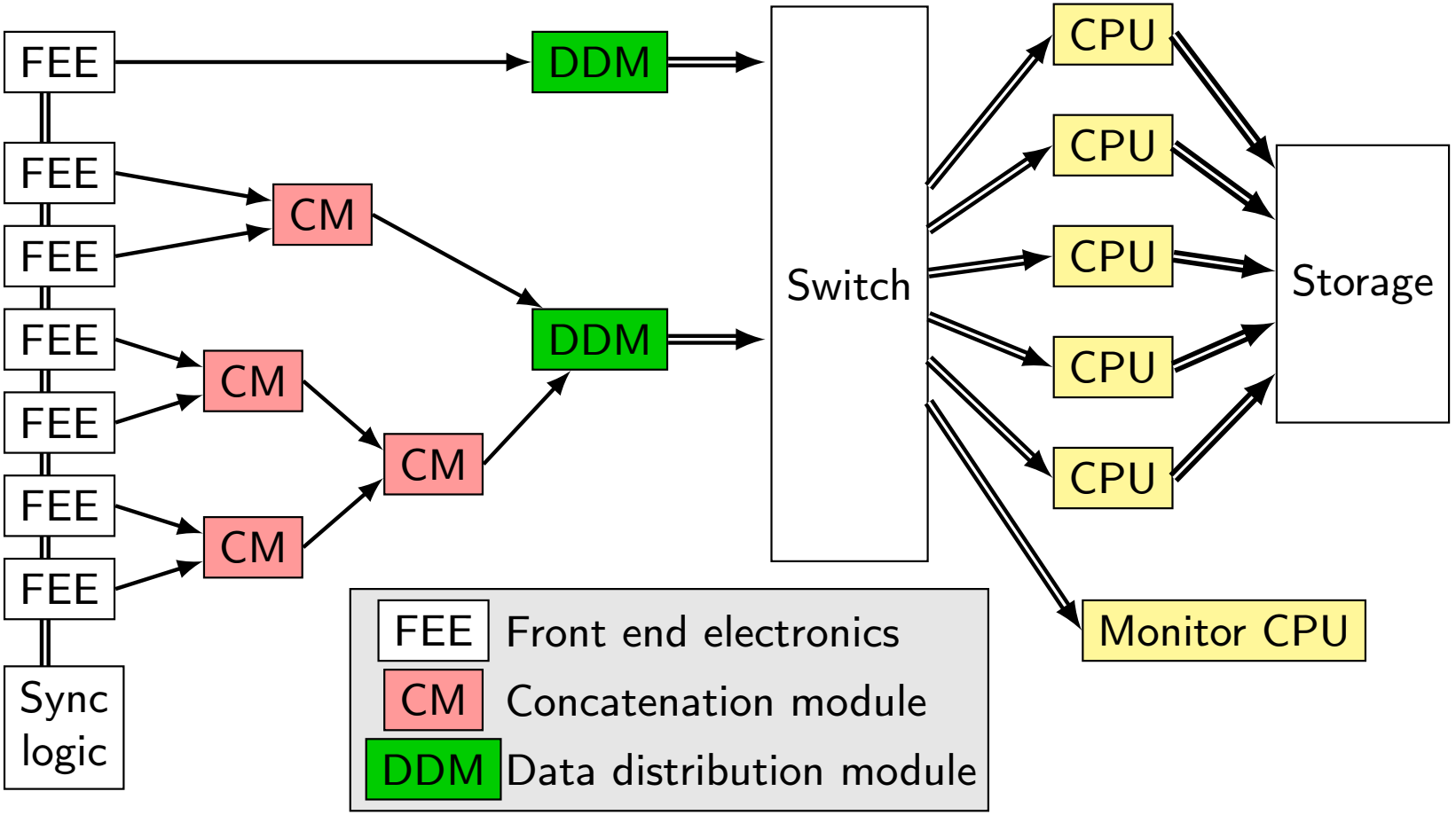
- High channel count and rates $\Longrightarrow$ highly parallel readout
- We have: all data for **one** channel for **all** times.
- We need: all data for **all** channels for **one** time segment.
- "Event building"
- Typical approach: Read in first, then redistribute.

The transposition problem

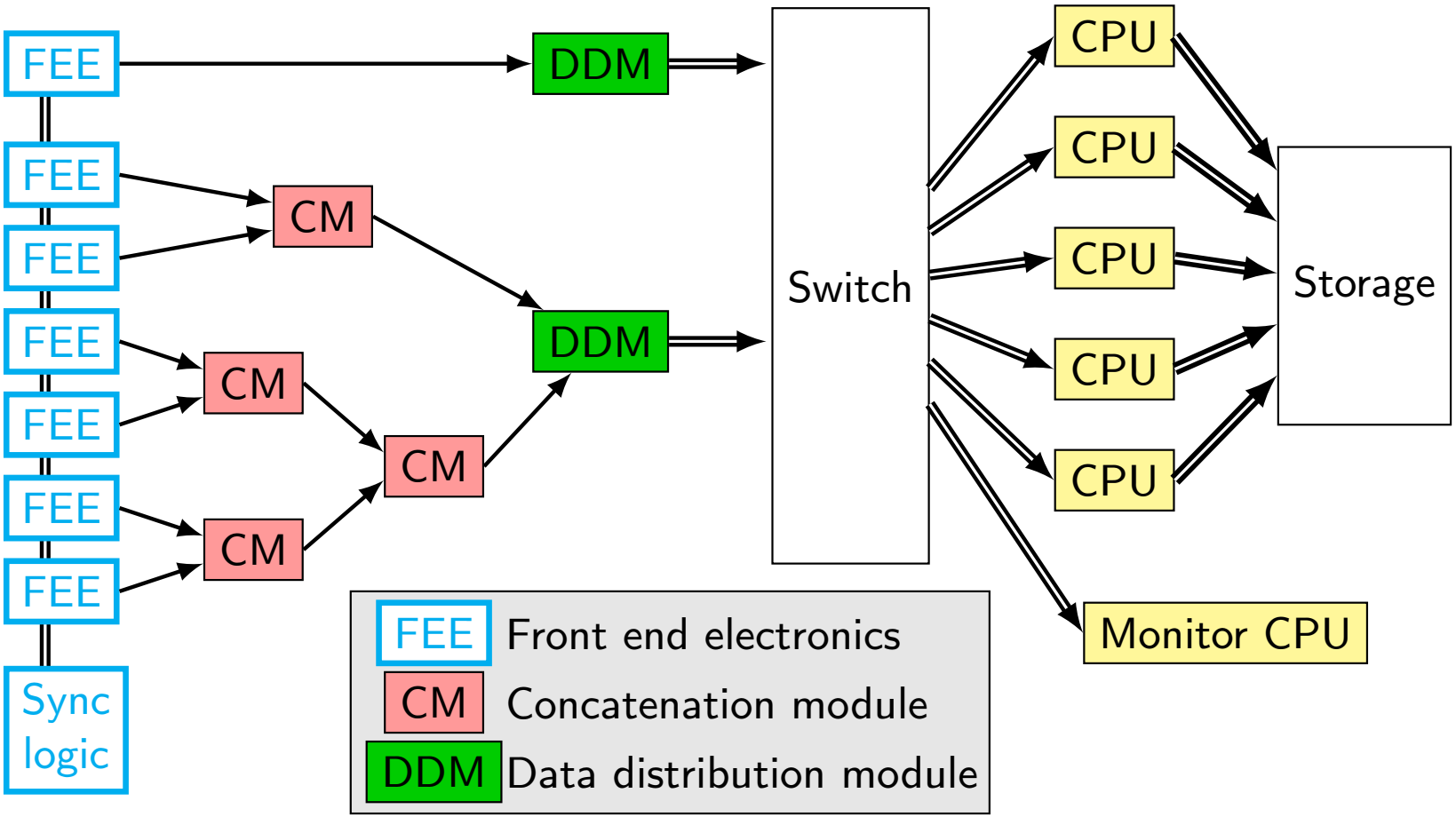
- High channel count and rates $\Longrightarrow$ highly parallel readout
- We have: all data for **one** channel for **all** times.
- We need: all data for **all** channels for **one** time segment.
- "Event building"
- Typical approach: Read in first, then redistribute.

Why not distribute in DAQ itself?

Overview

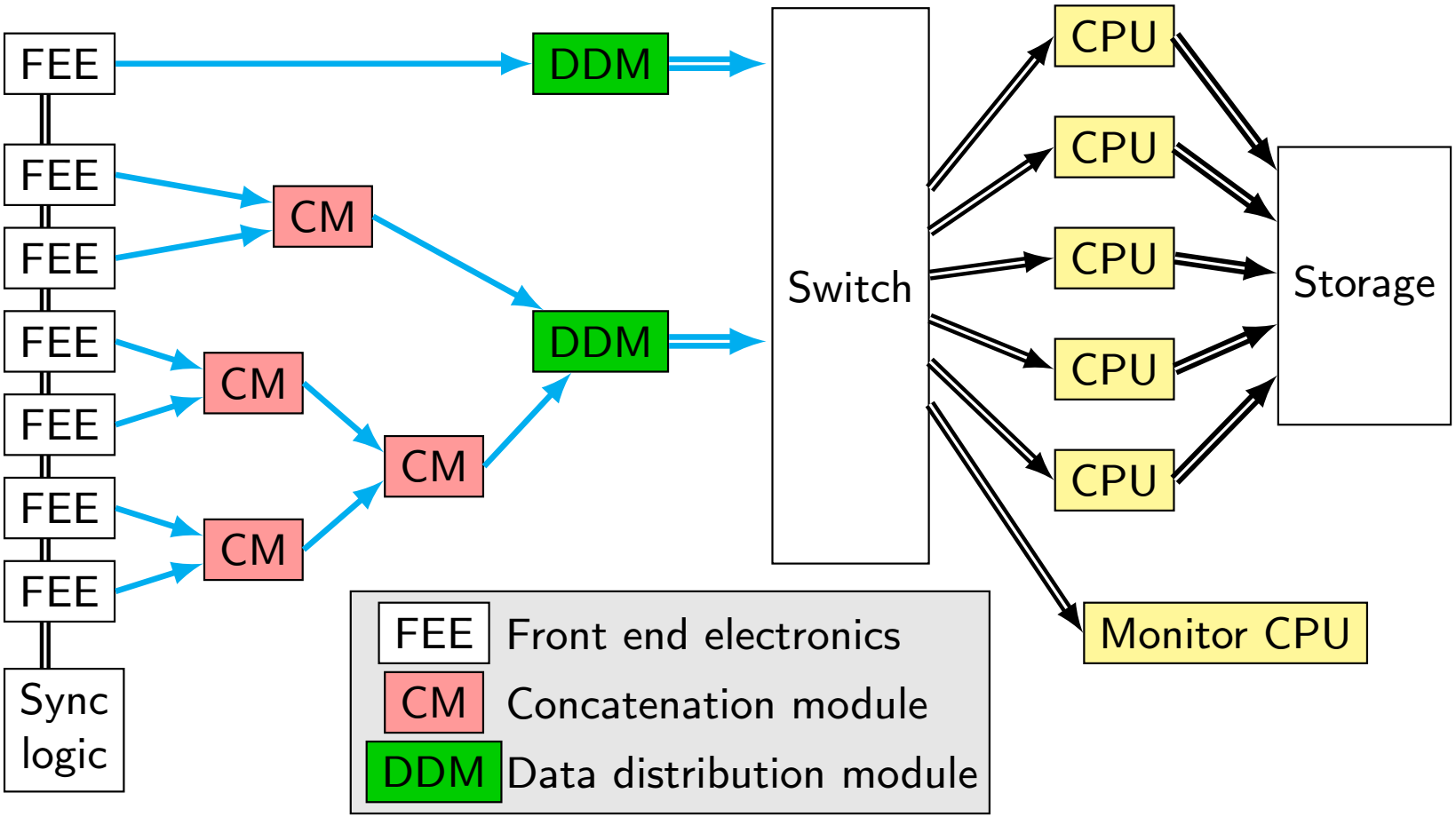

FEE
FEE
FEE
FEE
FEE
FEE
FEE
Sync logic
DDM
CM
DDM
CM
CM
CM
Switch
CPU
CPU
CPU
CPU
CPU
Storage
Monitor CPU
FEE Front end electronics
CM Concatenation module
DDM Data distribution module

Overview



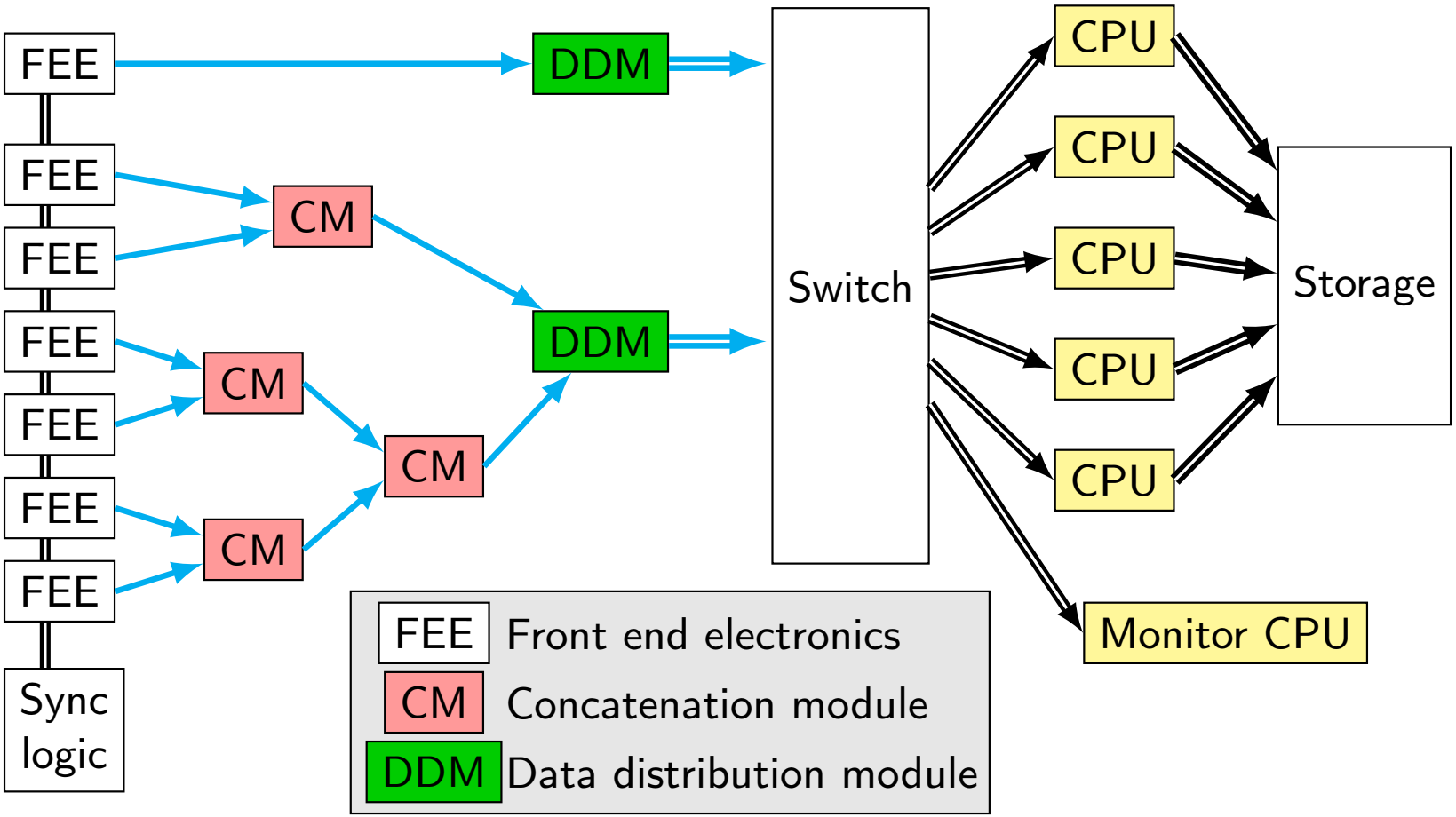
- Only front end electronics is detector specific

Overview



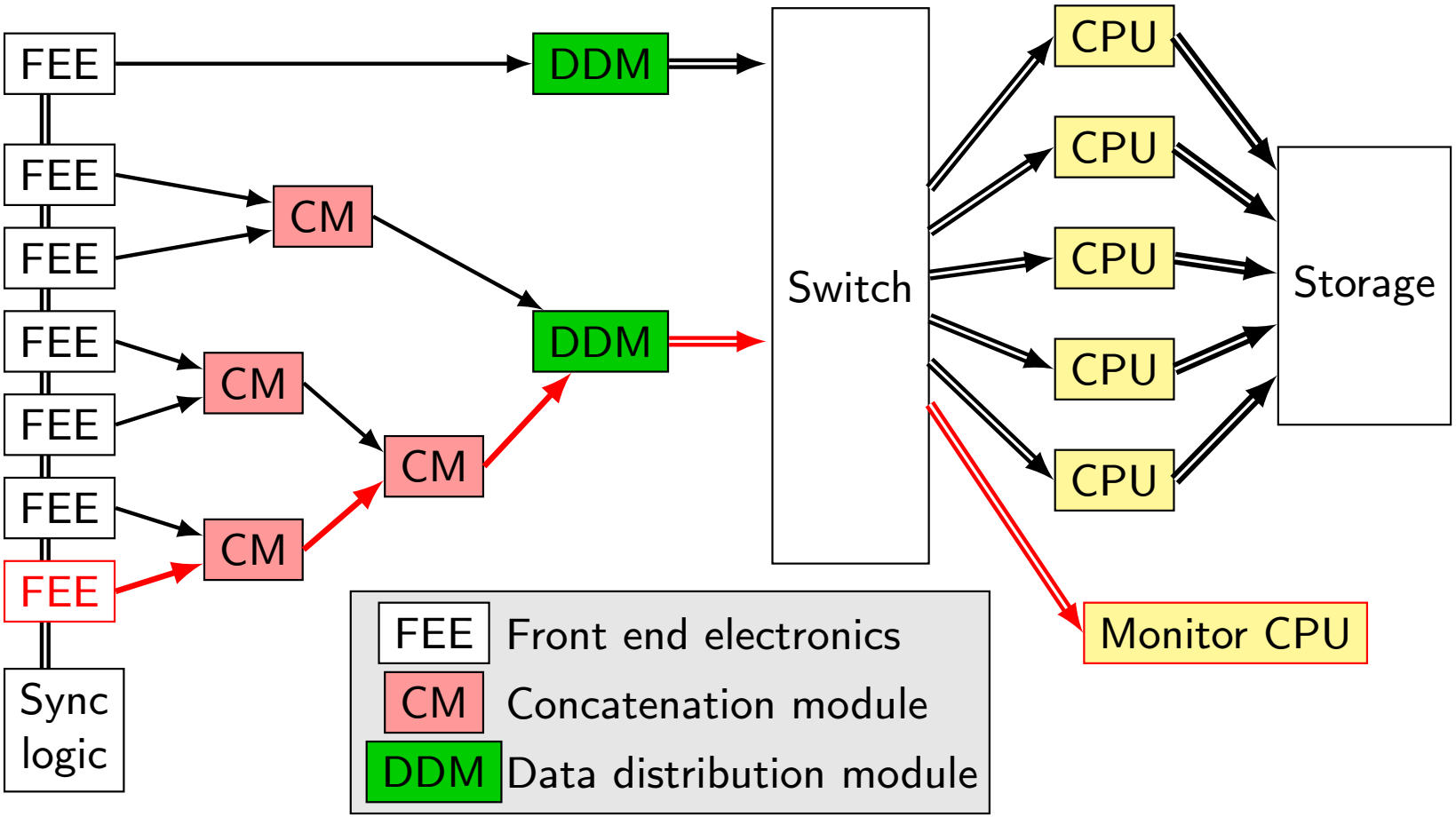
- Only front end electronics is detector specific
- Protocol is agnostic!

Overview



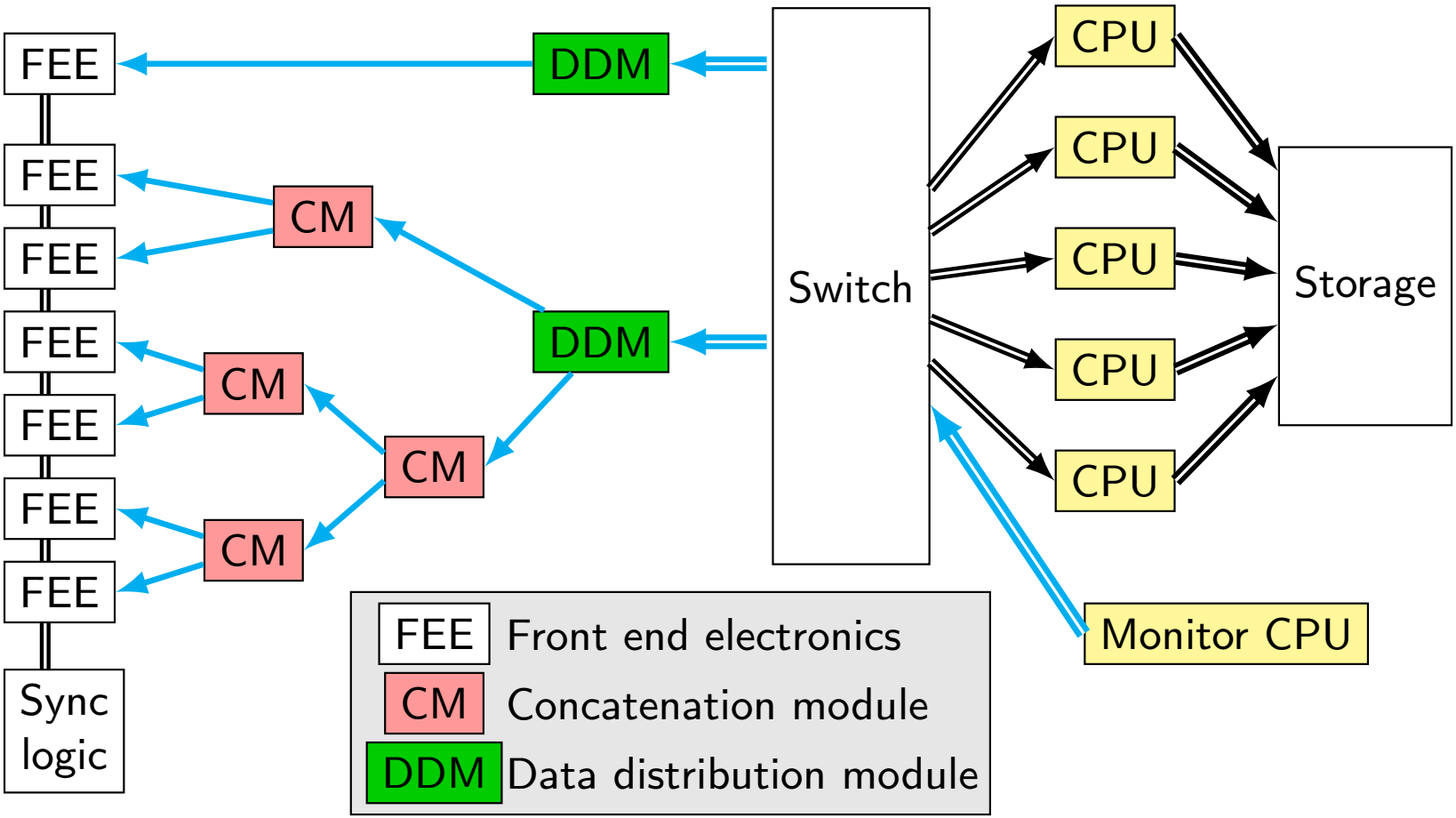
- Protocol must:
  - have low overhead

Overview



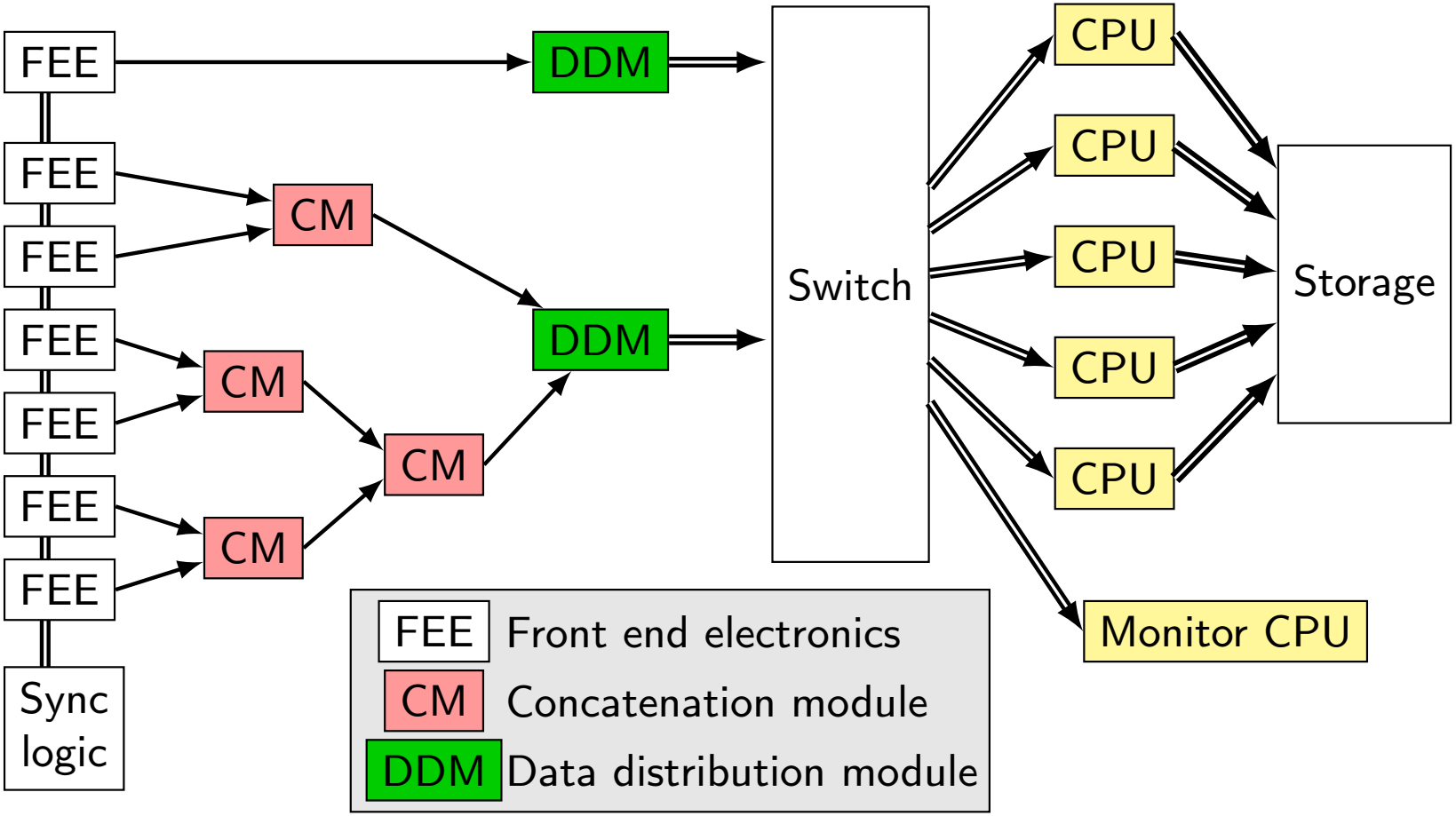
- Protocol must:
  - have low overhead
  - allow for out-of-band data (calibration, fault signaling, ...)

Overview



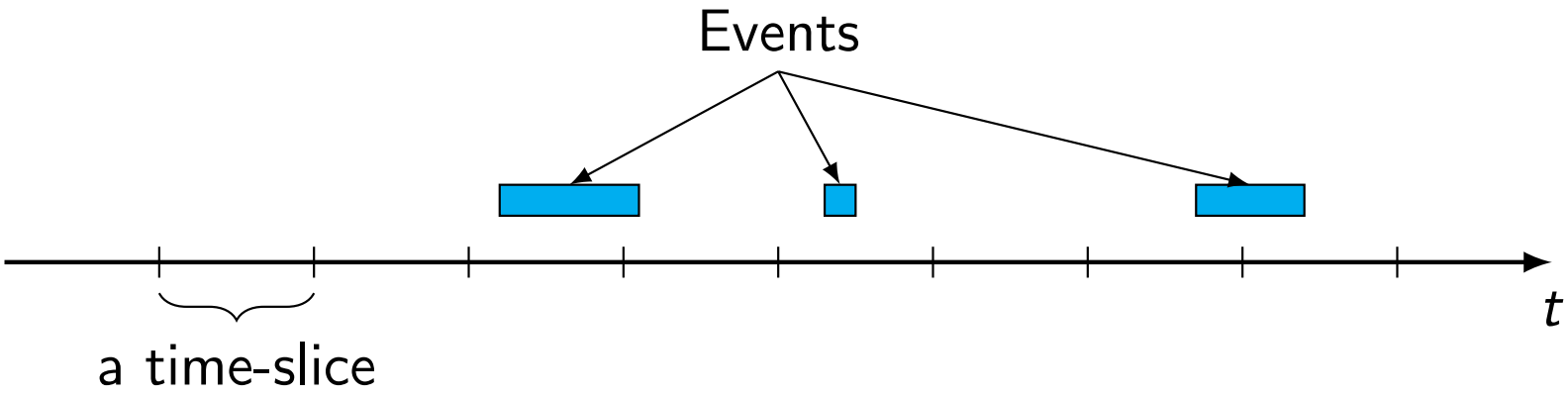
- Protocol must:
  - have low overhead
  - allow for out-of-band data (calibration, fault signaling, ...)
  - have a back channel (setup, calibration, ...)

Overview



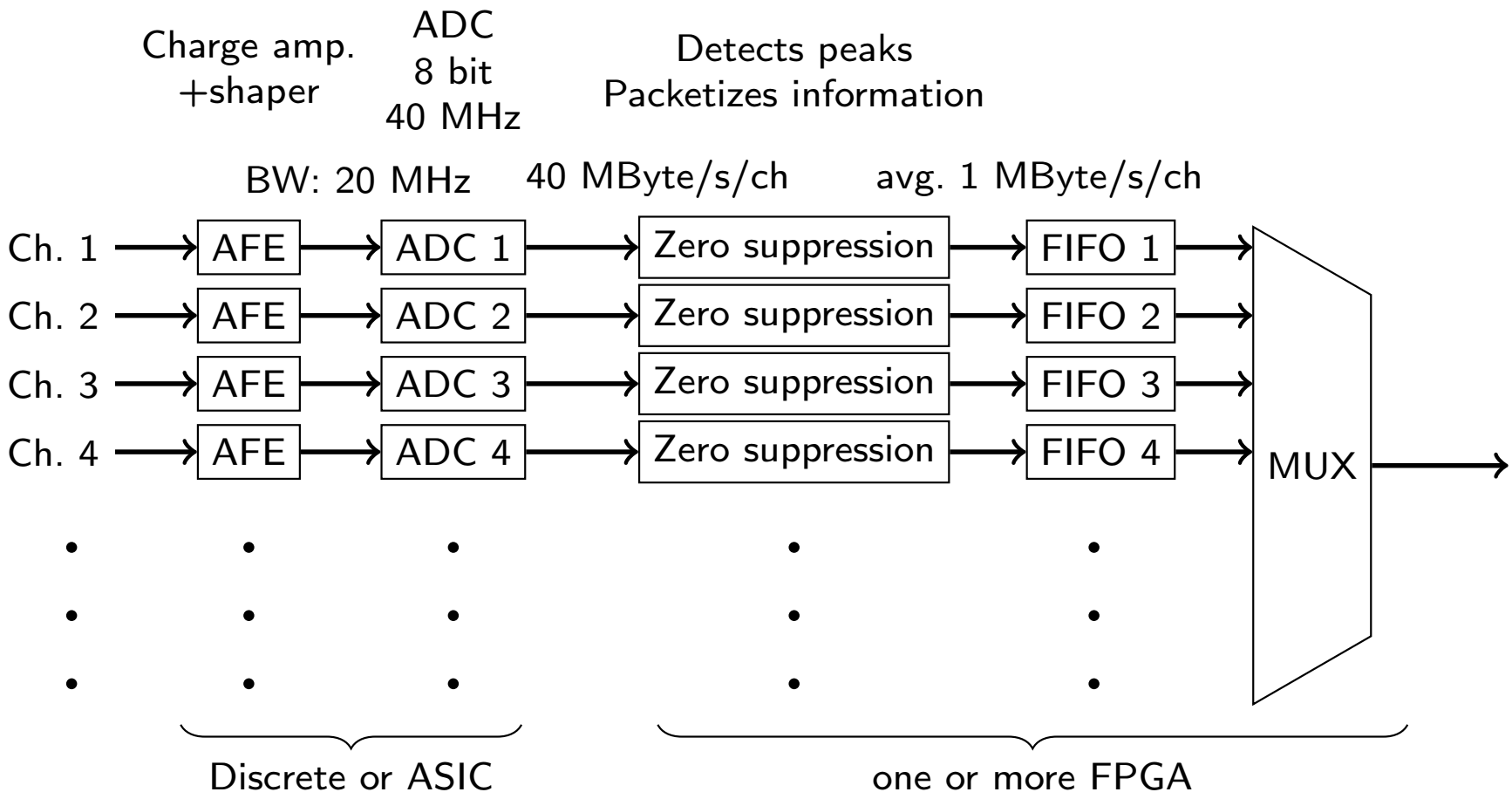
Need external input for further protocol requirements
$\Longrightarrow$Collaborate with detector groups, iterate design.

Time-slice



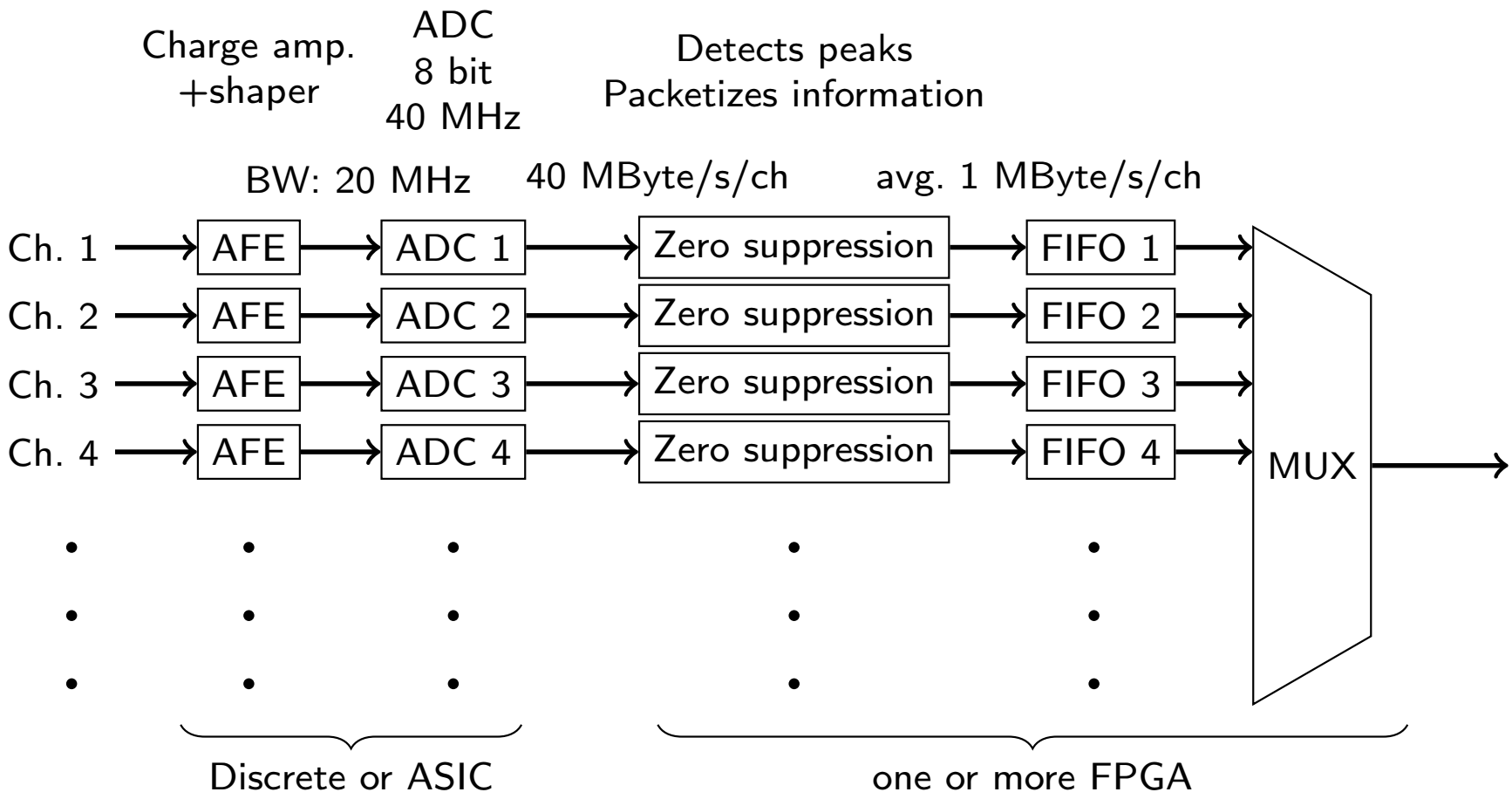
- A constant clock defines time-slices.
- Period is slightly larger than maximal event duration in detector.
- Each event can overlap two time-slices at most.

Front end electronics



- Highly detector dependent$\Longrightarrow$mock up with data generator modules, build baseline design (collaboration!)
- Can be more complex, e.g. CPU based

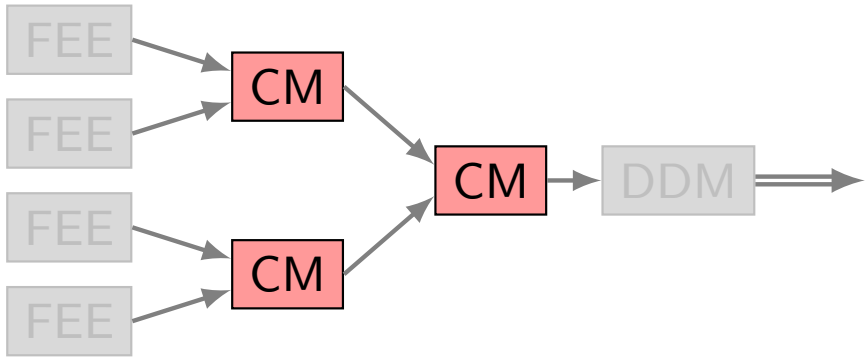
Front end electronics



Detector design $\Longleftrightarrow$ readout design

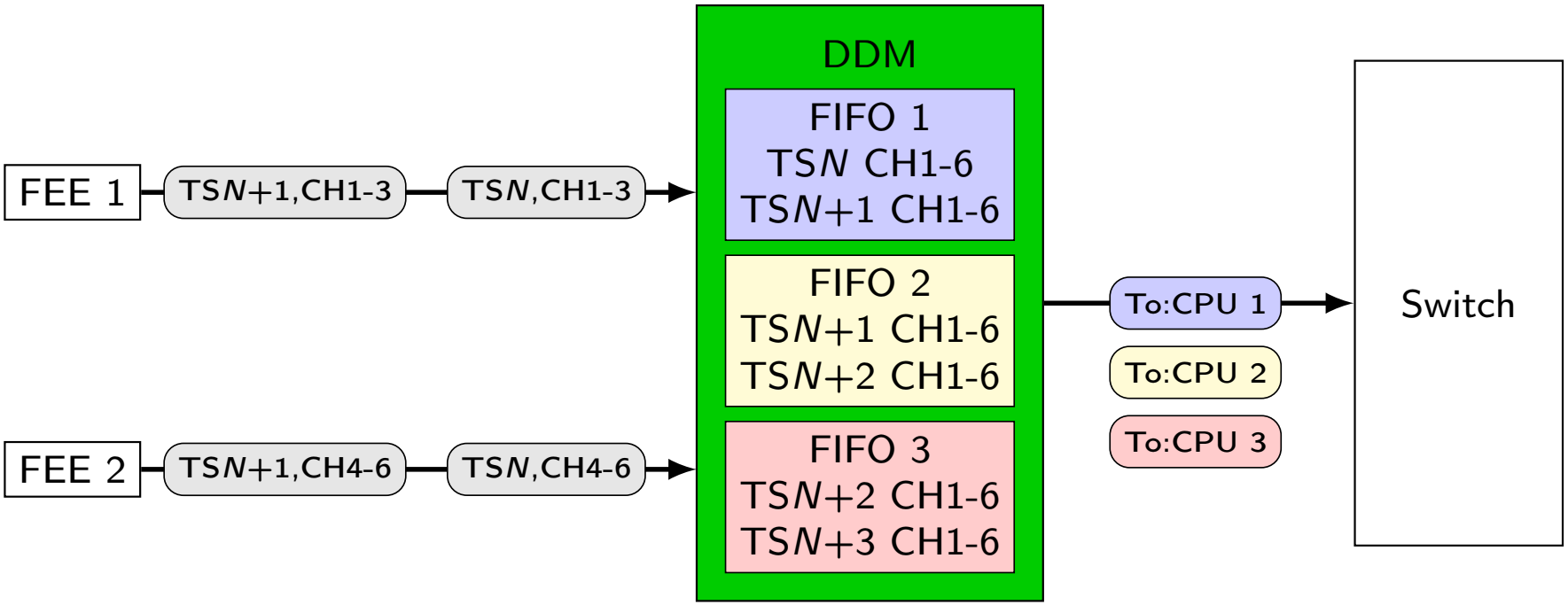
- No trigger detector needed
- Readout must handle detector specifics

Concatenation modules



- Single FEE might not saturate link speed $\Longrightarrow$ wasteful
- Aggregate streams
- Can be cascaded
- Could be "dumb" $\Longrightarrow$ use off-the-shelf aggregator ICs
- Could be "smart" $\Longrightarrow$ (small) FPGA to repackage data by time-slice

Data distribution modules



- Aggregate data by time slice into FIFO
- If a FIFO is full, send it to one analysis CPU
- Also handles out-of-band data
- Fast backbone switch does the actual heavy lifting

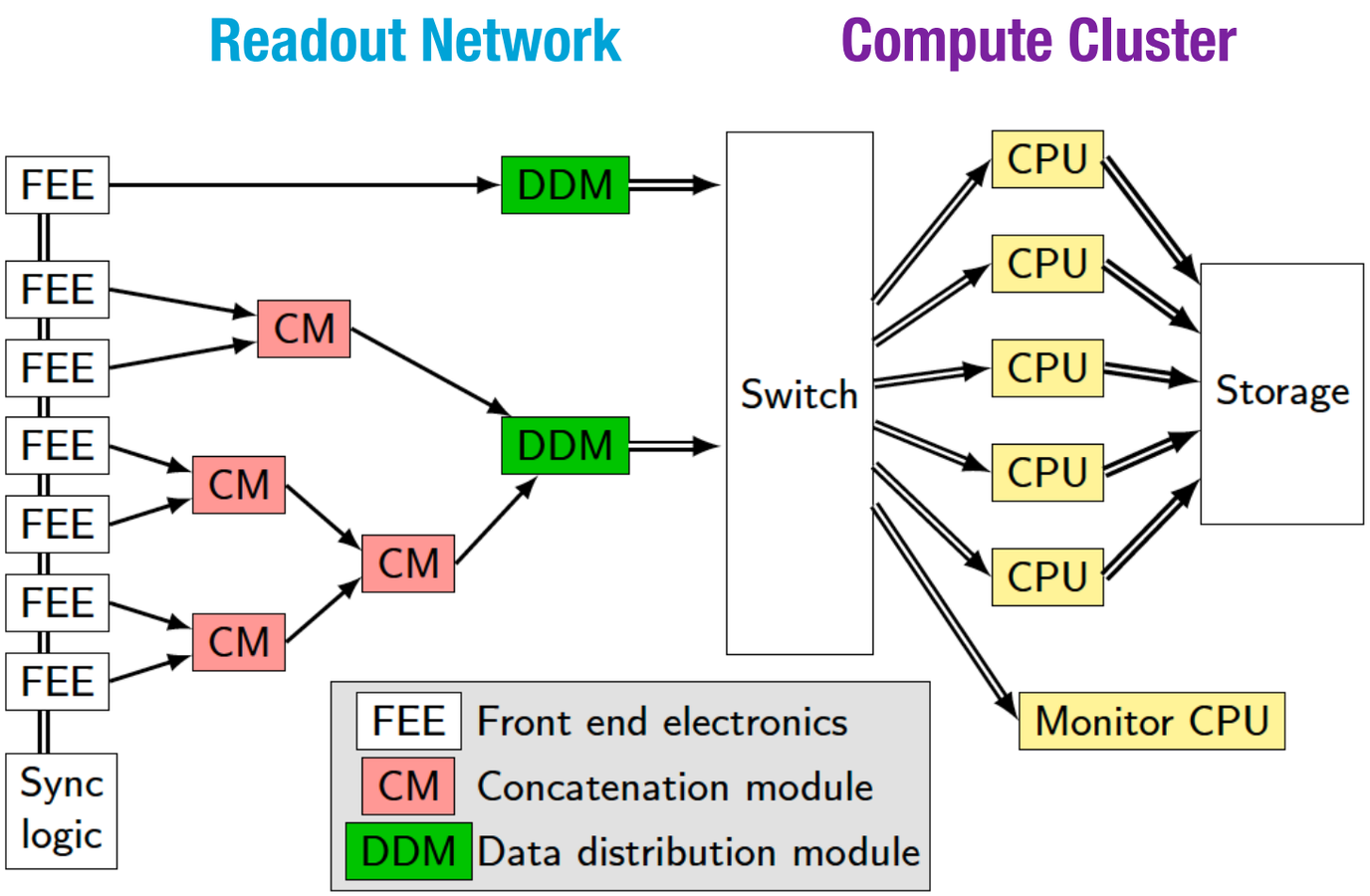
Deliverables

- Reference firmware (FPGA IP) for detector-specific front-end electronics
- Reference design + firmware for concatenation module
- Reference design + firmware for data distribution module
- Design *specifically tuned* for the requirements of EIC

Cost and Timeline

Year 1	Year 2	Year 3
Procurement of FPGA dev. boards, oscilloscope. Dev. of protocol. Start of DDM reference design. Design data generators (FEE).	Produce and test reference design DDM, design, produce and test CM.	Fully test reference design. Update / test design on new technology nodes.
Hardware: 114 k$ ('scope, dev. boards, workstation, dev. tools) Manpower: 81.5 k$ (0.4 FTE engineer)	Hardware: 103 k$ (Production ref. DDM, CM, dev. boards, ...) Manpower: 63 k$ (0.3 FTE)	Hardware: 37.6 k$ (updated FPGAs) Manpower: 43.4 k$ (0.2 FTE)
195.5 k$	**166 k$**	**81 k$**

Event processing on compute cluster

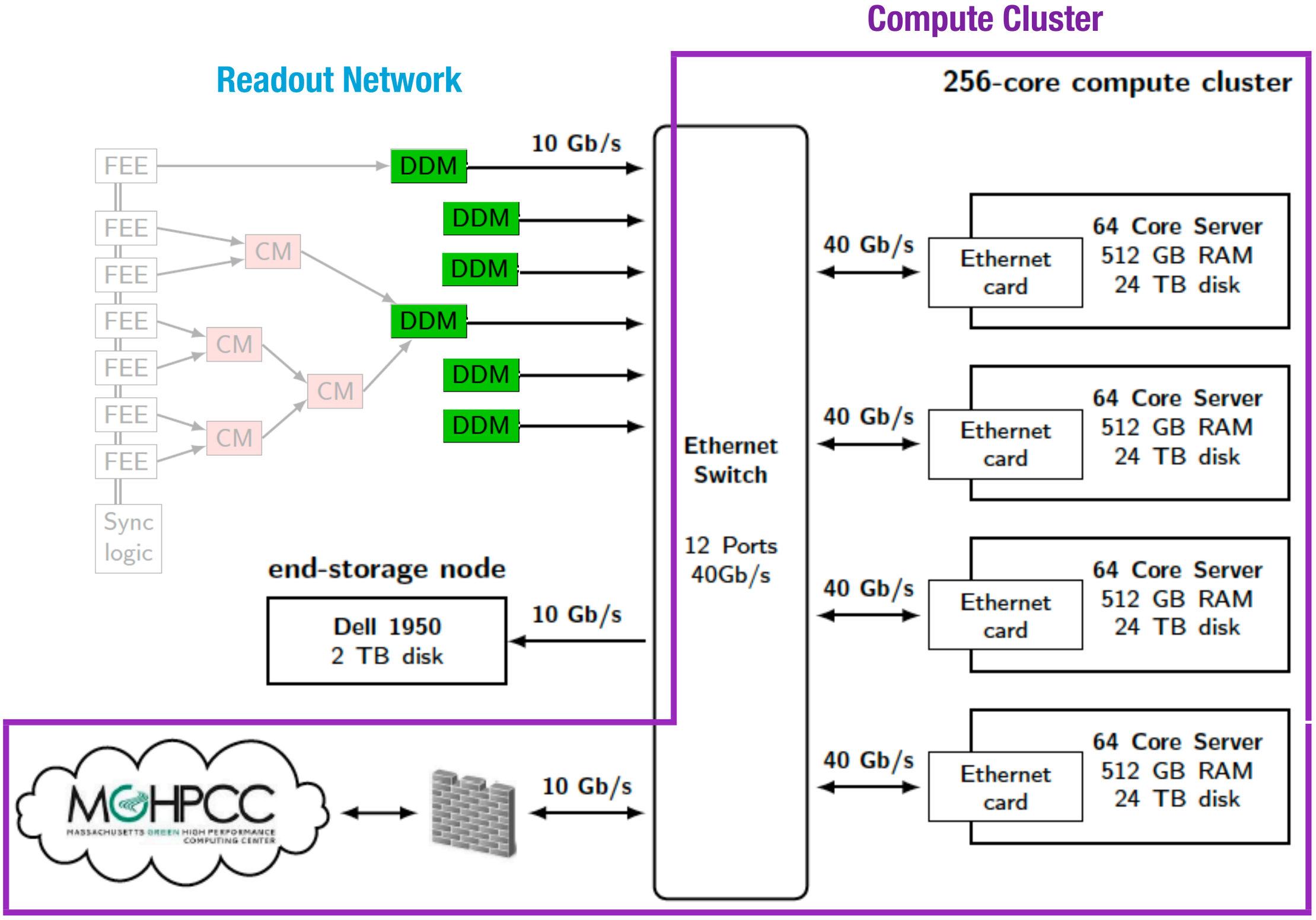


Two essential requirements for trigger-less readout:

1. Assemble complete events from many detectors in a high-rate environment ⇒ **Readout Network**

2. Efficiently classify those events "on the fly," reducing the data footprint while maintaing high trigger efficiency ⇒ **Compute Cluster**

Event processing on compute cluster

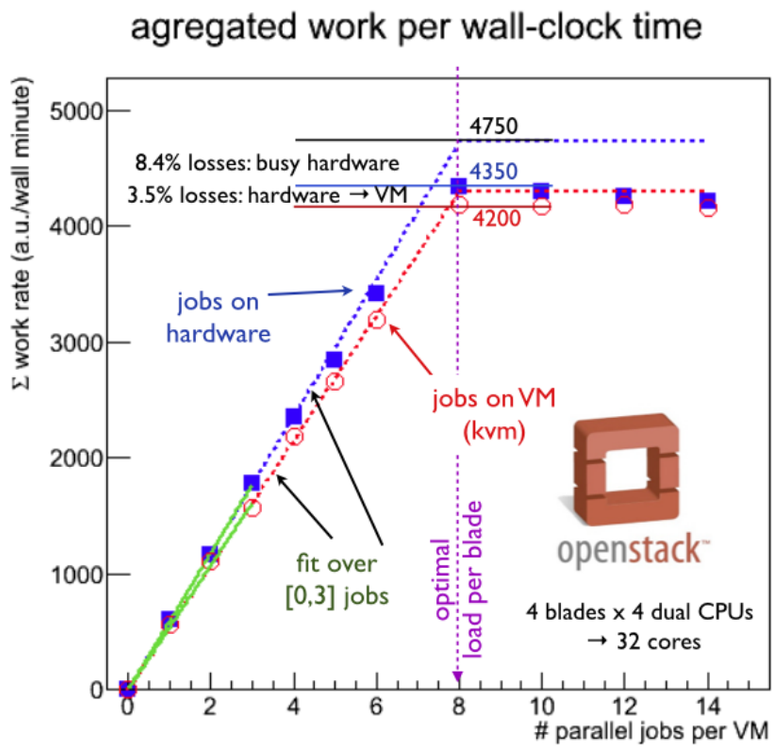


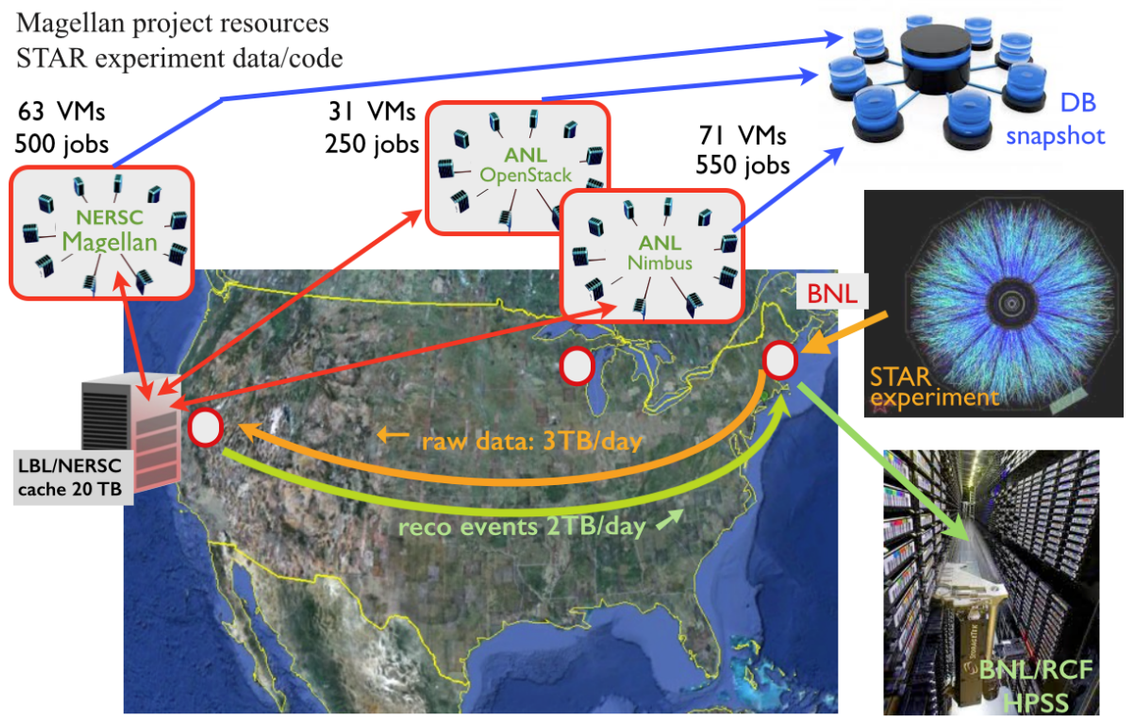
EIC considerations

- Considerable work done developing trigger-less readout proposals for LHC experiments in Run 3 (~2019), specifically ALICE and LHCb
- What's unique at an EIC?
  - Much lower multiplicity than LHC environment emphasizes the need for full event information in event selection
  - Versatility in beam species (ep/eA), energy, and luminosity to accommodate the diverse physics program
  - Requires flexibility in event selection algorithm complexity with varying physics priorities, etc.
- This proposal uses virtualization to encapsulate the event processing algorithm, allowing "on demand" scaling to meet the physics needs
- Utilize a local cloud (eg. the JLab/BNL offline computing cluster) to provide for scalable complexity of the event classification requirements with minimal capital investment in the online computer cluster

Virtualization benefits & experience

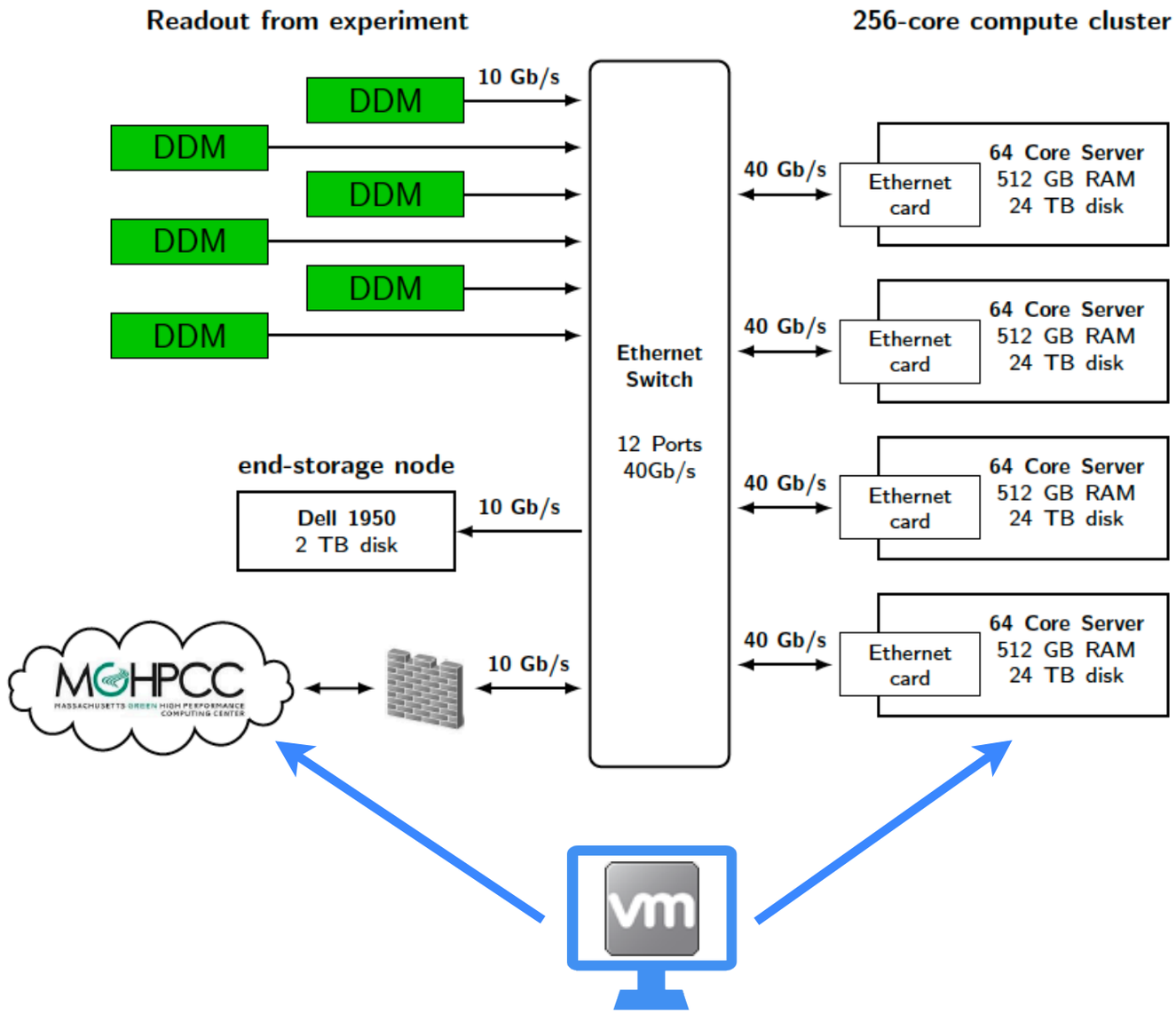
- Virtual Machine (VM) is a software implementation of a physical computer
- Benefits
  - Remove dependence on hardware and OS of physical node
  - Natural integration of online and offline compute clusters through cloud capabilities
  - Net benefits compensate small (~4%) CPU losses
- Experience
  - MIT-Reuse virtual cluster ♻
  - Real-time STAR data production on the Cloud
  - Multi-site GlueX production through FutureGrid project





Deliverables: compute cluster

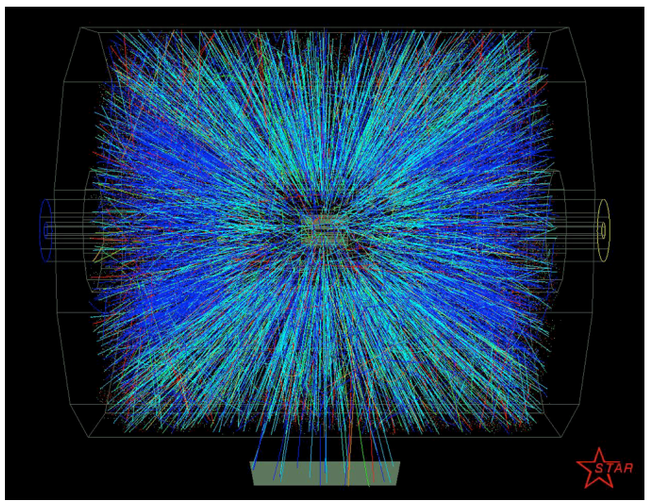
- Setup virtual cluster on local compute nodes
- Develop and vet virtual machine (VM) running locally and on remote cloud (Holyoke/Amazon EC2)



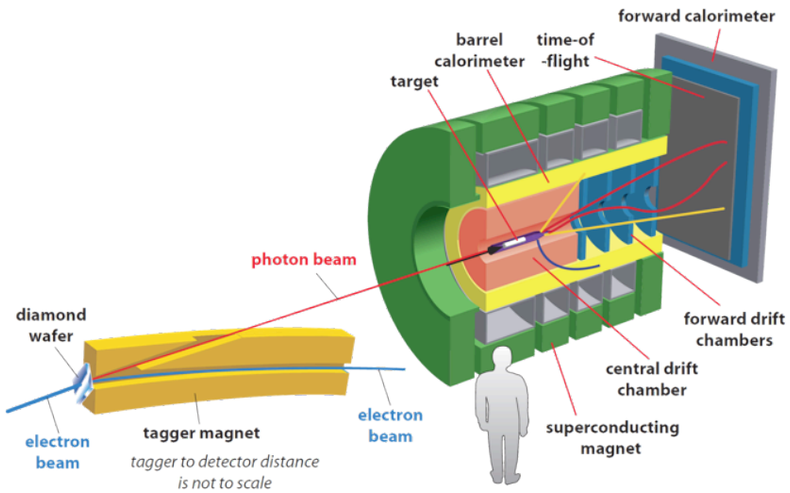
Deliverables: compute cluster

- Setup virtual cluster on local compute nodes
- Develop and vet virtual machine (VM) running locally and on remote cloud (Holyoke/Amazon EC2)
- Develop and test software on local compute node VMs to assemble sub-events from DDMs into full events at high rate
- Implement event classifier algorithms on VMs

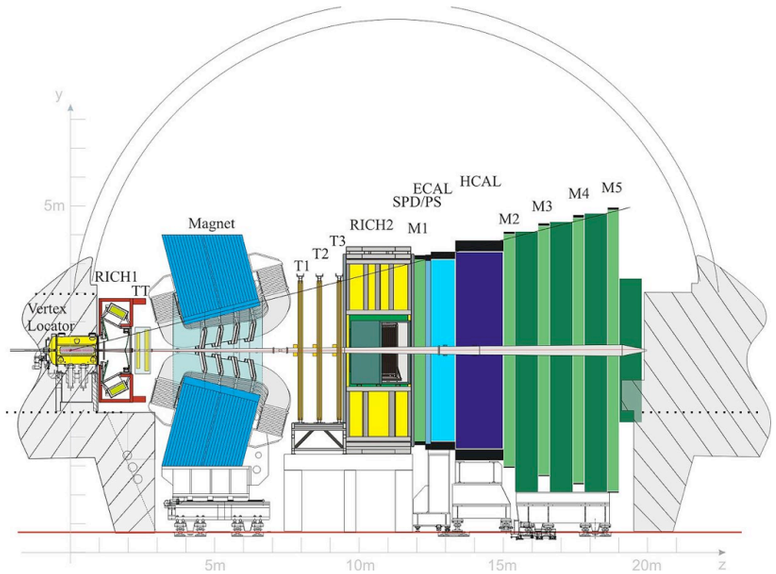
STAR



GlueX citations experiment

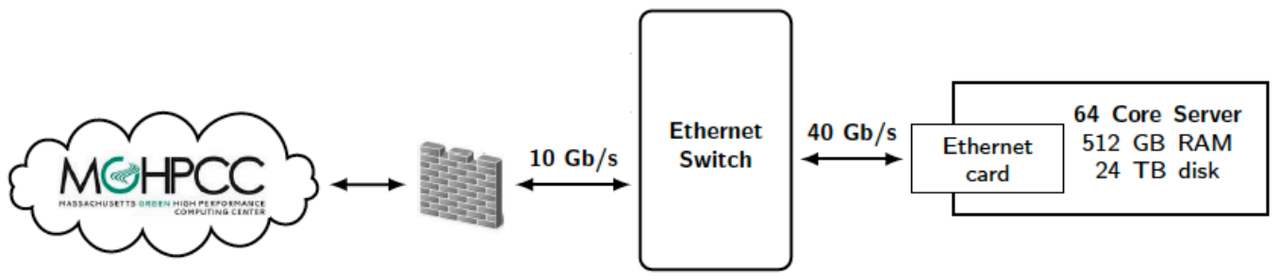


LHCb



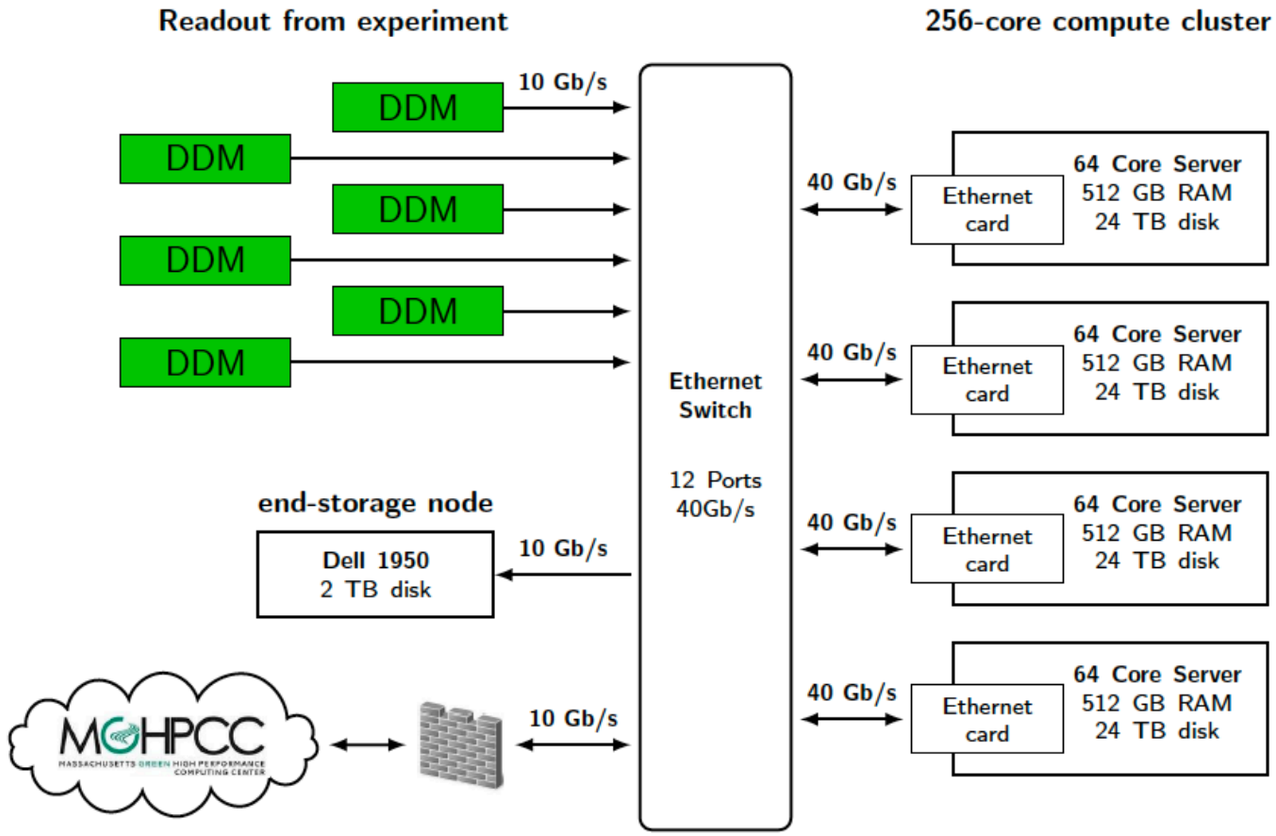
Deliverables: compute cluster

- Setup virtual cluster on local compute nodes
- Develop and vet virtual machine (VM) running locally and on remote cloud (Holyoke/Amazon EC2)
- Develop and test software on local compute node VMs to assemble sub-events from DDMs into full events at high rate
- Implement event classifier algorithms on VMs
- Develop and test mechanism for processing event rates beyond the capacity of the local compute cluster:
  - Offload event processing CPU load to remote compute cluster (eg. Holyoke) running identical VM and event processing



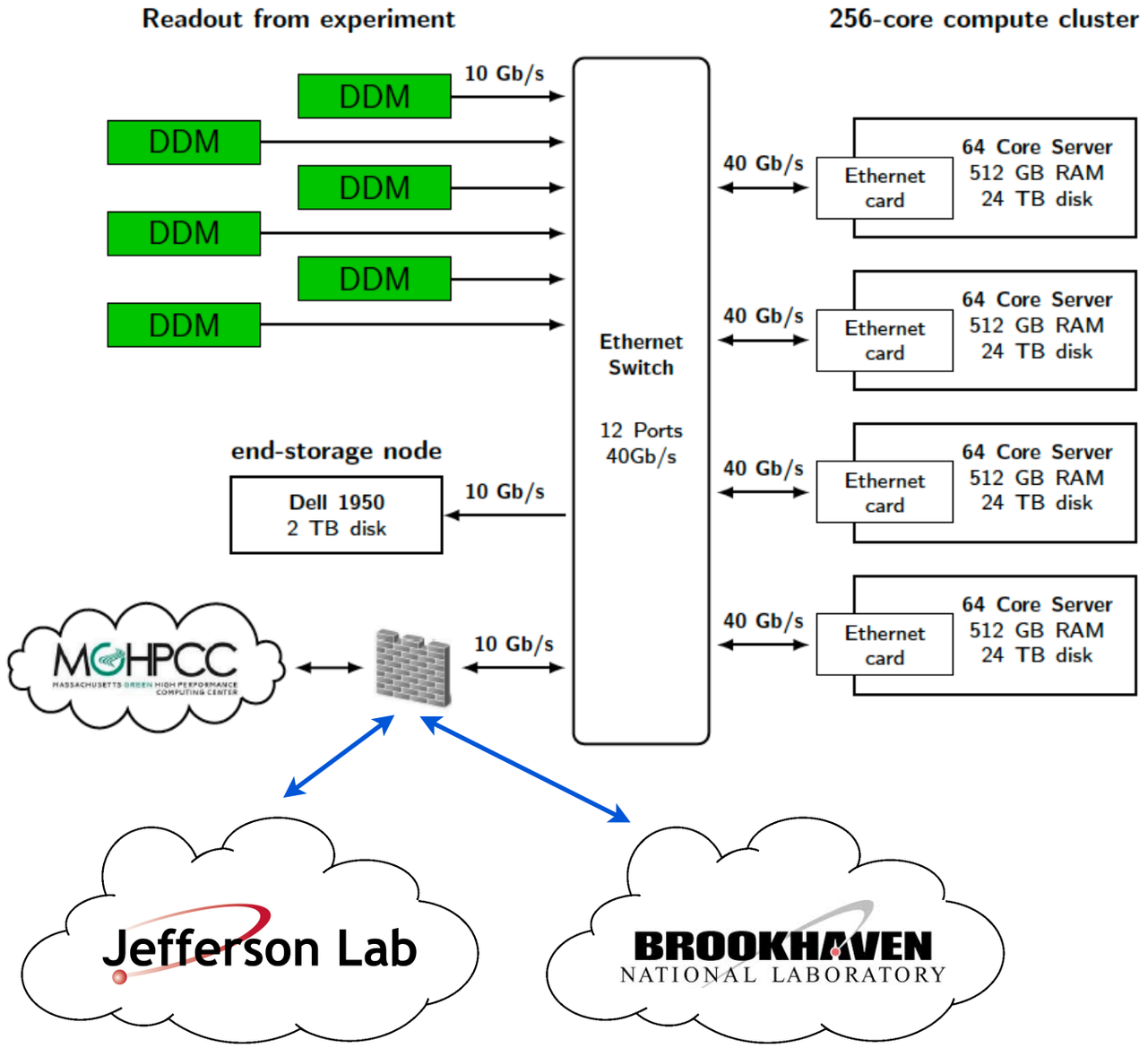
Research Clouds (Holyoke/MGHPCC)

- This proposal: utilize research cloud available in Holyoke, MA



Research Clouds (Holyoke/MGHPCC)

- This proposal: utilize research cloud available in Holyoke, MA
- For full scale EIC readout, realistic to use local “Tier 0” resources at the host lab
- Future work: implementation with offline computing farms at JLab/BNL
- Develop expertise within the EIC user community



Cost and Timeline: Compute Cluster

Year 1	Year 2	Year 3
Incorporate compute node in virtual cluster and vet VM on local and remote clouds.	Develop event building with mock data from DDMs. Establish high-speed transfer to VM at Holyoke.	Evaluate full chain of event processing. Demonstrate offloading of peak event processing to remote cloud.
Hardware & Services: 35.5 k$ Manpower & Travel: 25.1 k$ (0.2 FTE CS student)	H & S: 48.3 k$ M & T: 25.8 k$ (0.2 FTE)	H & S: 27.5 k$ M & T: 26.5 k$ (0.2 FTE)
60.6 k$	**74.1 k$**	**54 k$**

Why now?

- Incorporation in ongoing detector development:
  - *The Committee notes that trigger concept(s) for EIC detectors should be developed, as they will have a significant impact on the detector design* -EIC R&D Committee reports
- Develop expertise in EIC user community to:
  - Design, deploy, and operate readout system for trigger-less DAQ
  - Develop and employ scalable event-classification solution
- Opportunities for real-world tests in upcoming experiments:
  - DarkLight at JLab (Phase-I recently funded by NSF MRI) provides a very high rate experiment to test the entire readout system
  - GlueX could implement it's software-based High-Level Trigger through a virtual compute cluster similar to what's proposed here

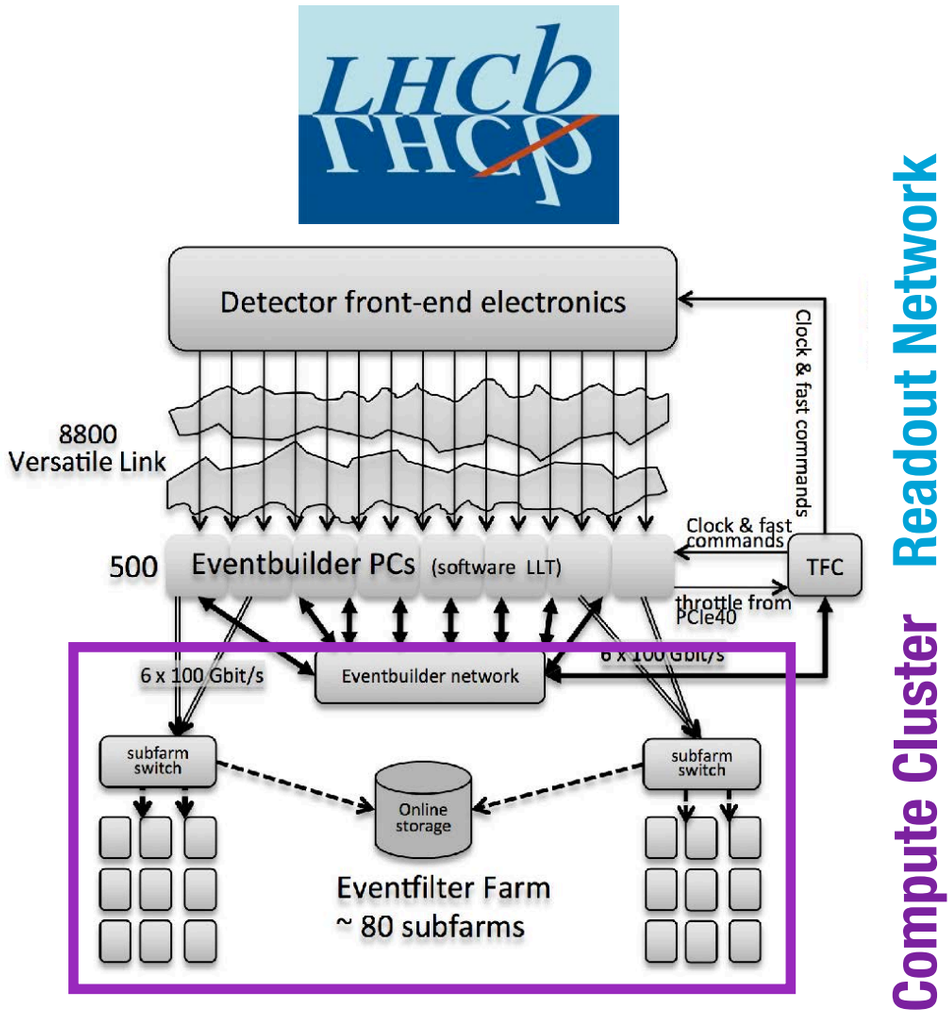
Summary

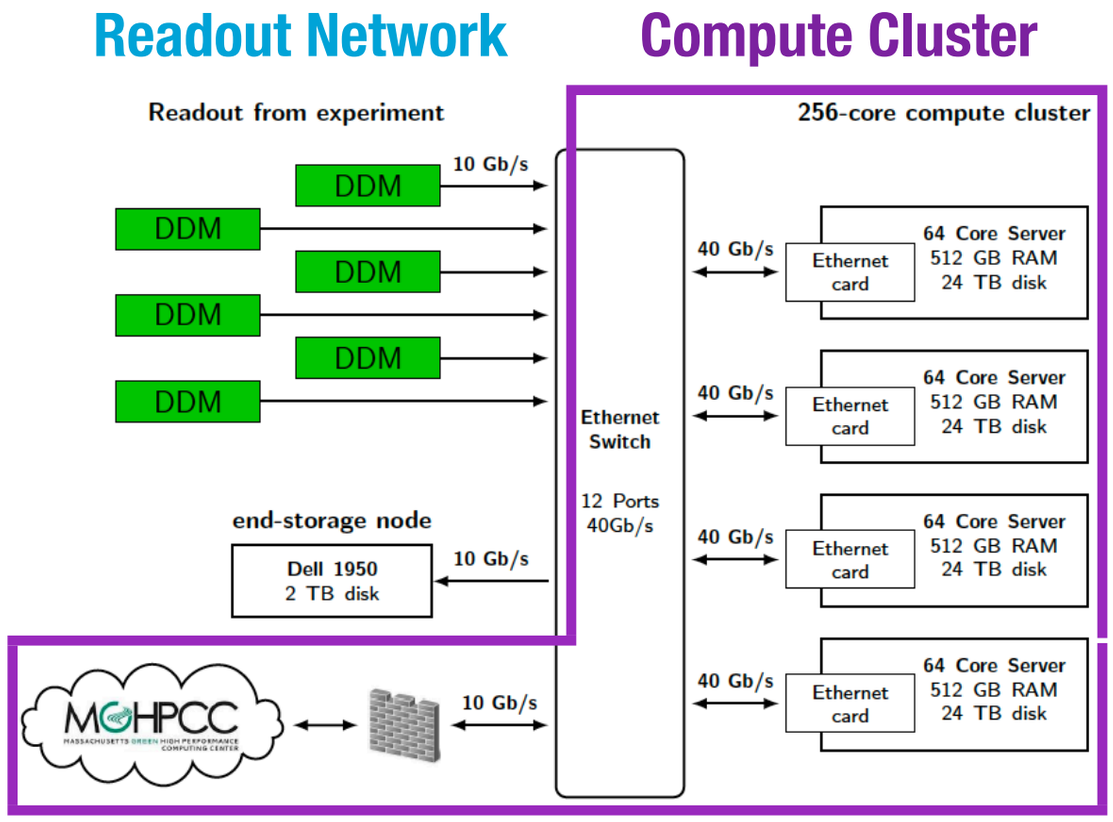
- We propose to develop a scalable, trigger-less readout system for data acquisition in EIC experiments, requiring:
  - A high-bandwidth readout network to assemble complete events in a high-luminosity environment
  - Efficient real-time event classification, which is flexible enough to be optimized for the diverse physics program
- Parallel development of readout system and detector designs allows the requirements for each to be optimized and establishes the technology/expertise in the community

	Year 1	Year 2	Year 3
Readout Network	195.5 k$	166 k$	81 k$
Compute Cluster	60.6 k$	74.1 k$	54 k$
Total	**256.1 k$**	**240.1 k$**	**135.0 k$**

Backup

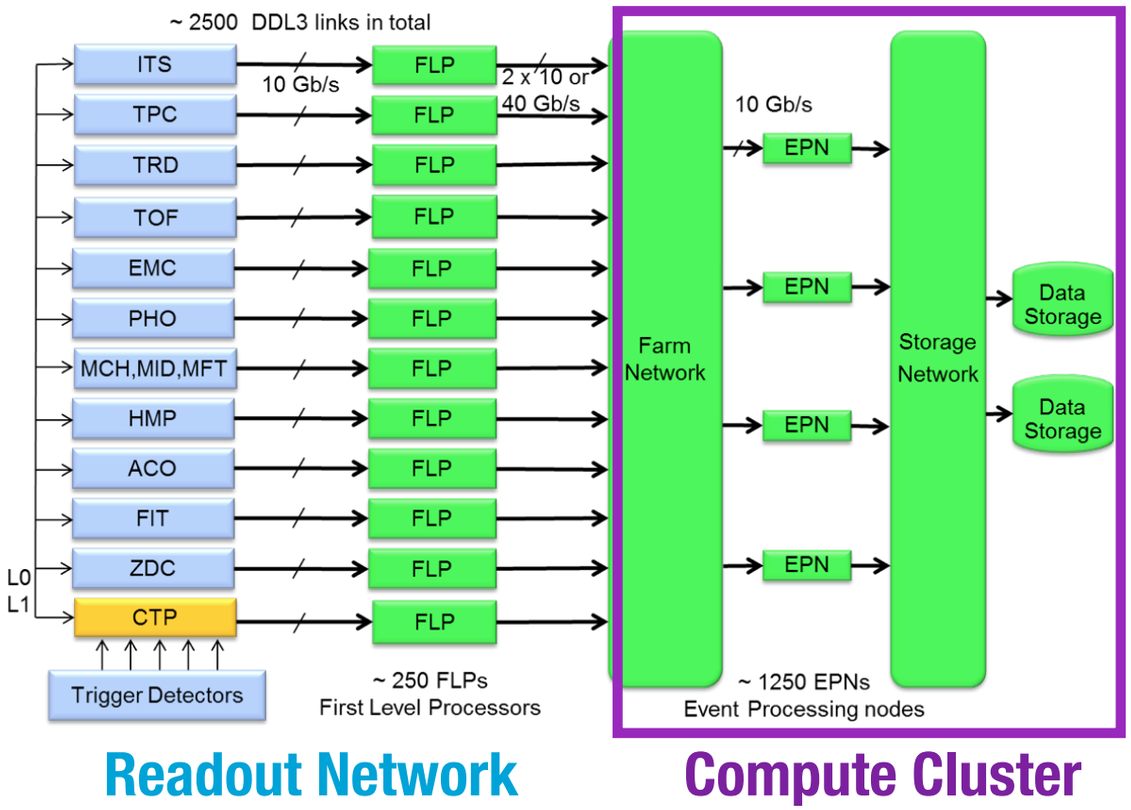
LHCb comparison

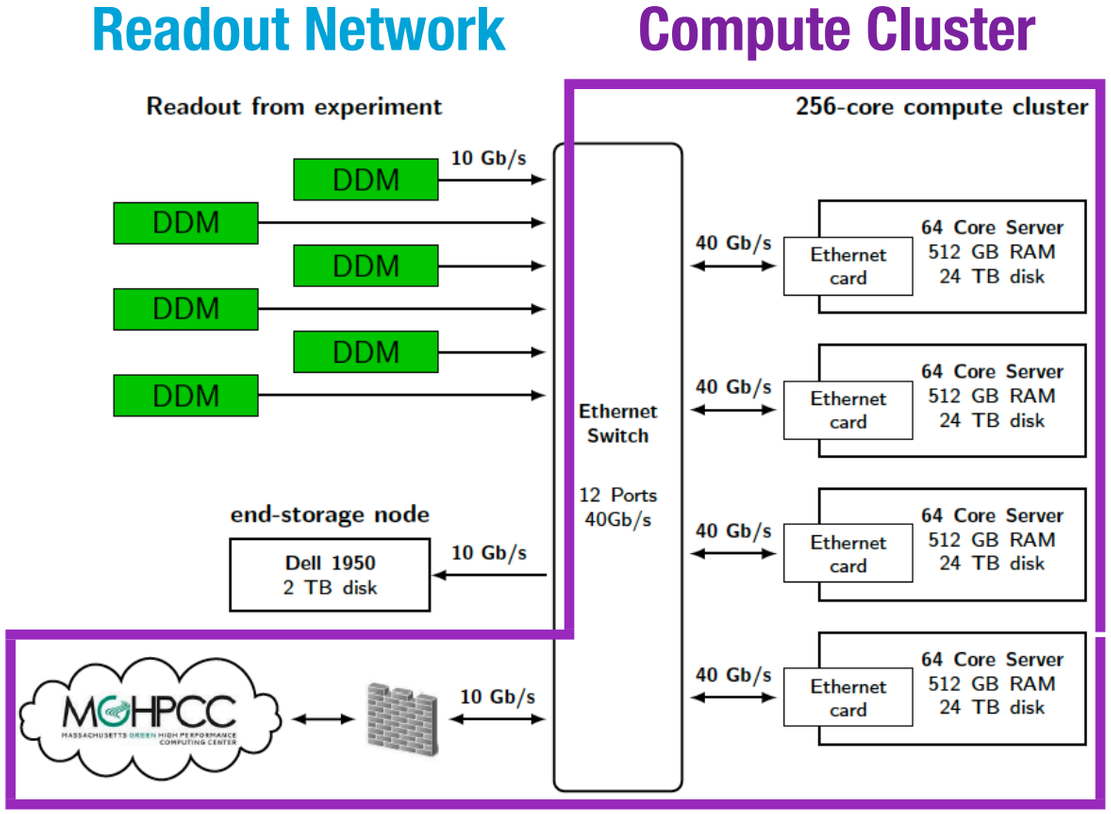




- LHCb plans for ~1000 event filter nodes in it's "online" farm
- Our proposal provides a natural extension to cloud resources potentially in the host lab's central computing resources

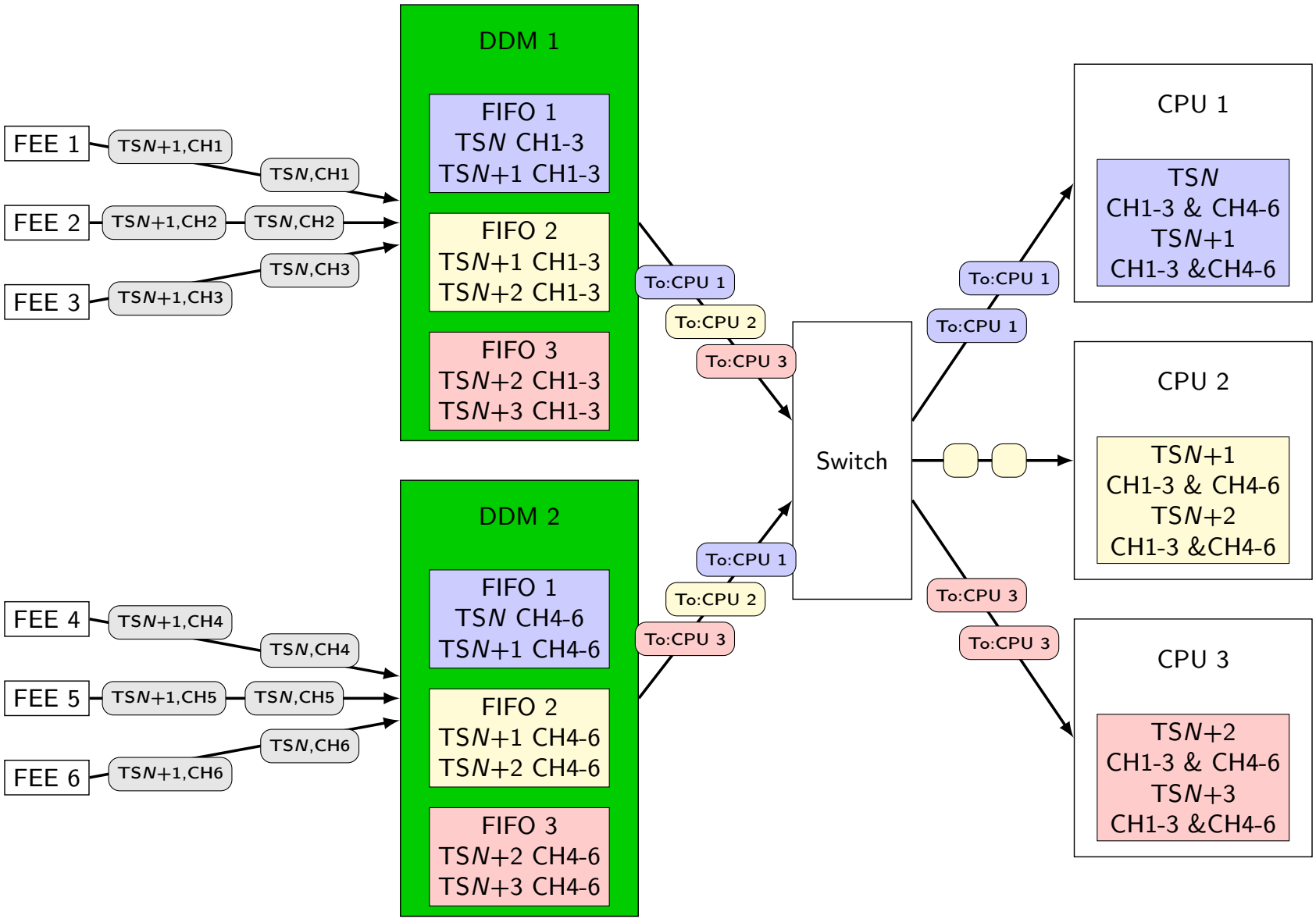
ALICE comparison



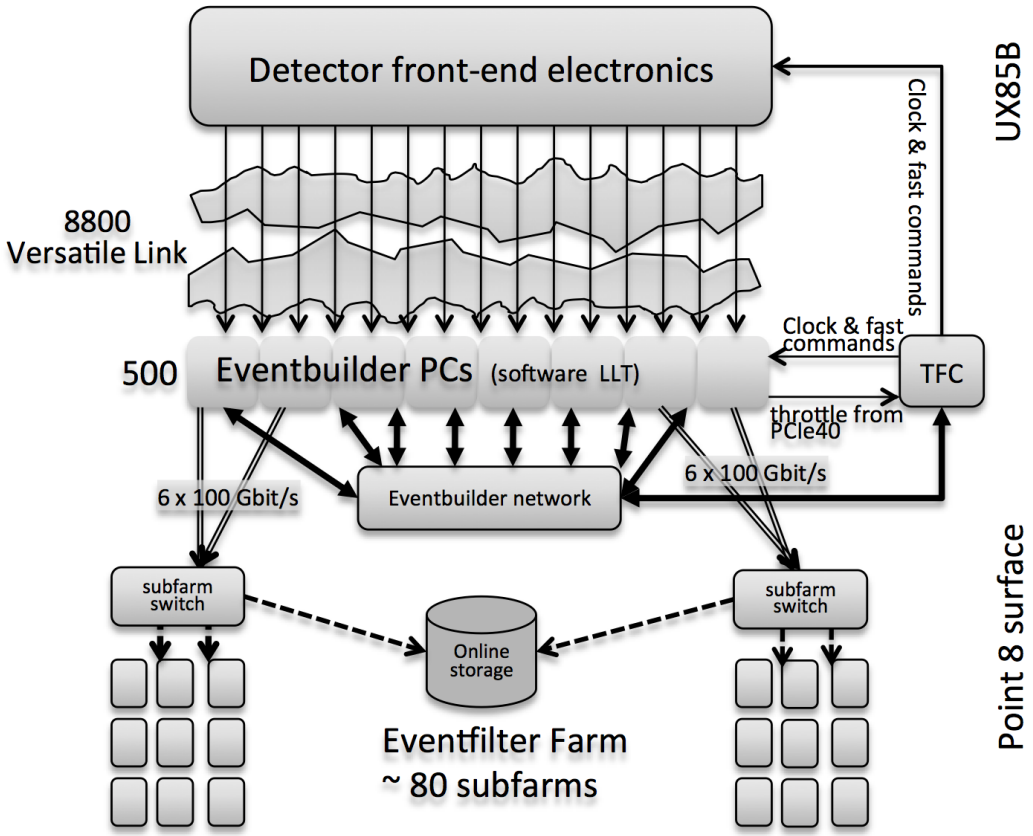


- Sub-event pre-processing provided on the ALICE FLP, not necessarily beneficial at an EIC where full event reconstruction is most relevant
- Similar plans as LHCb for many “online” event processing nodes

Data flow overview

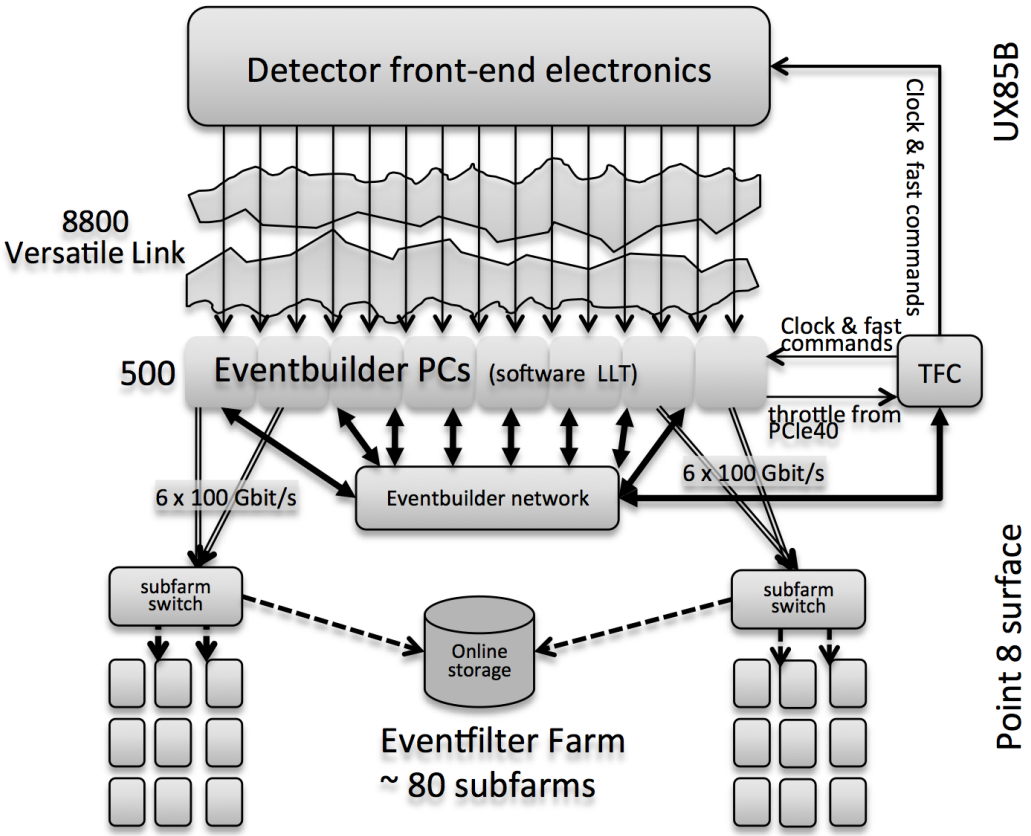


Comparison with LHCb



- Superficially similar
- But data has to go through CPU/memory bus twice!
- Hardware complexity similar:
  - Zero suppression same
  - Data aggregation for versatile links $\sim$ CM
  - FPGA based versatile link receivers $\sim$ DDM

Comparison with LHCb



- Superficially similar
- But data has to go through CPU/memory bus twice!
- Hardware complexity similar:
  - Zero suppression same
  - Data aggregation for versatile links $\sim$ CM
  - FPGA based versatile link receivers $\sim$ DDM
- Hardware design specific for CERN requirements
- Need experience in EIC community!

AN EXPERIMENT TO SEARCH FOR THE DECAY $\mu \to eee$