Future trends in DAQ
SoLID DAQ

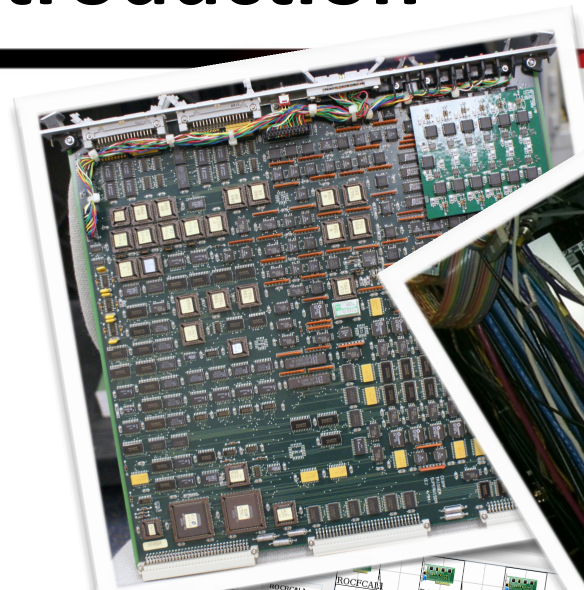
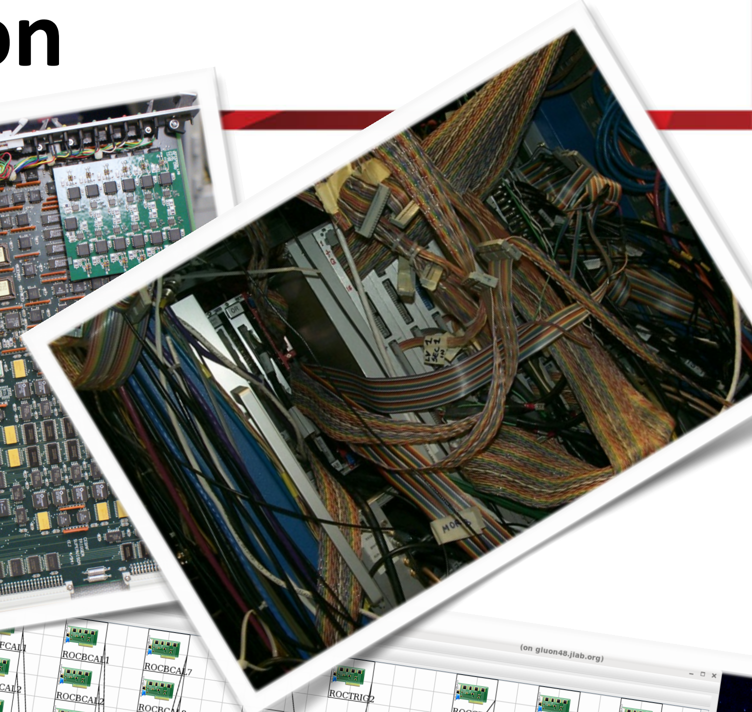
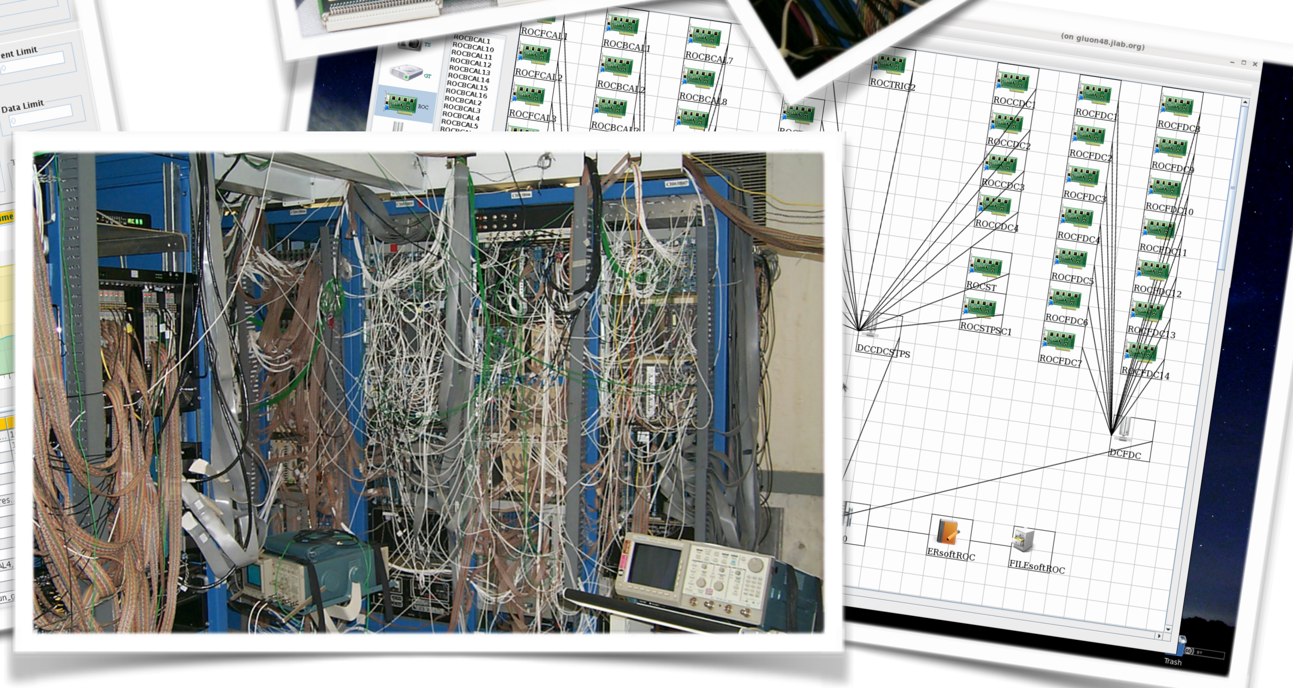
Graham Heyes

Steve Wood

Sept 20th 2016

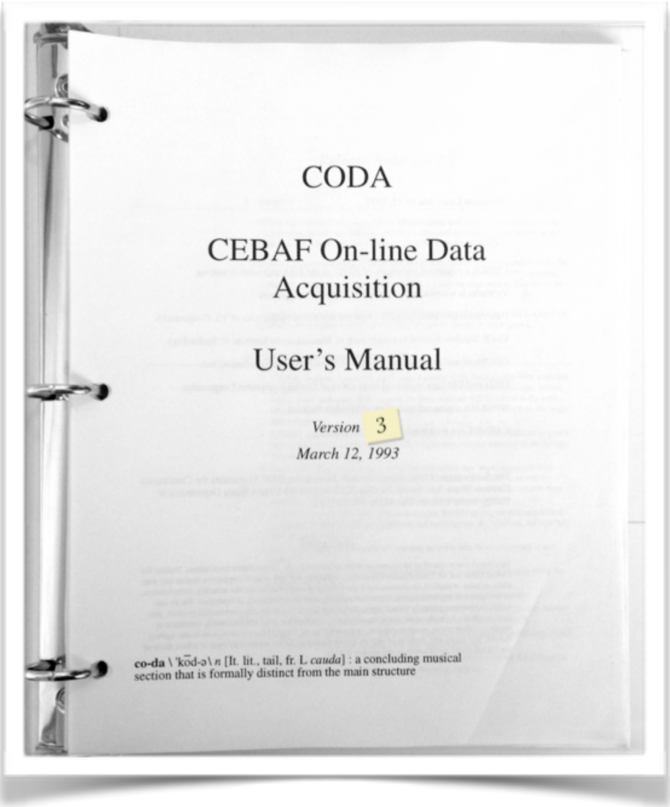
Introduction

- How JLab does things now.
- Challenges.
- Lessons learned.
- Trends and ideas.
- Concluding remarks.

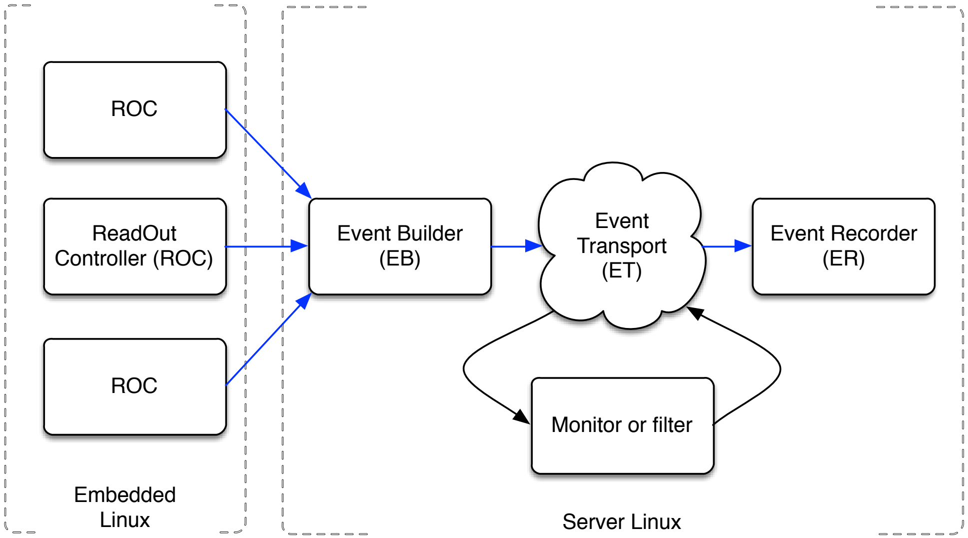
DAQ at JLab

- CEBAF Online Data Acquisition
  - Specification of recommended COTS hardware.
  - Suite of custom hardware.
  - Software:
    - Hardware drivers.
    - Embeddable Linux OS.
    - Readout software.
    - Event transport - ET.
    - Event building and storage.
    - Data format - EVIO.
    - Experiment control - AFECS.



CODA, core concept

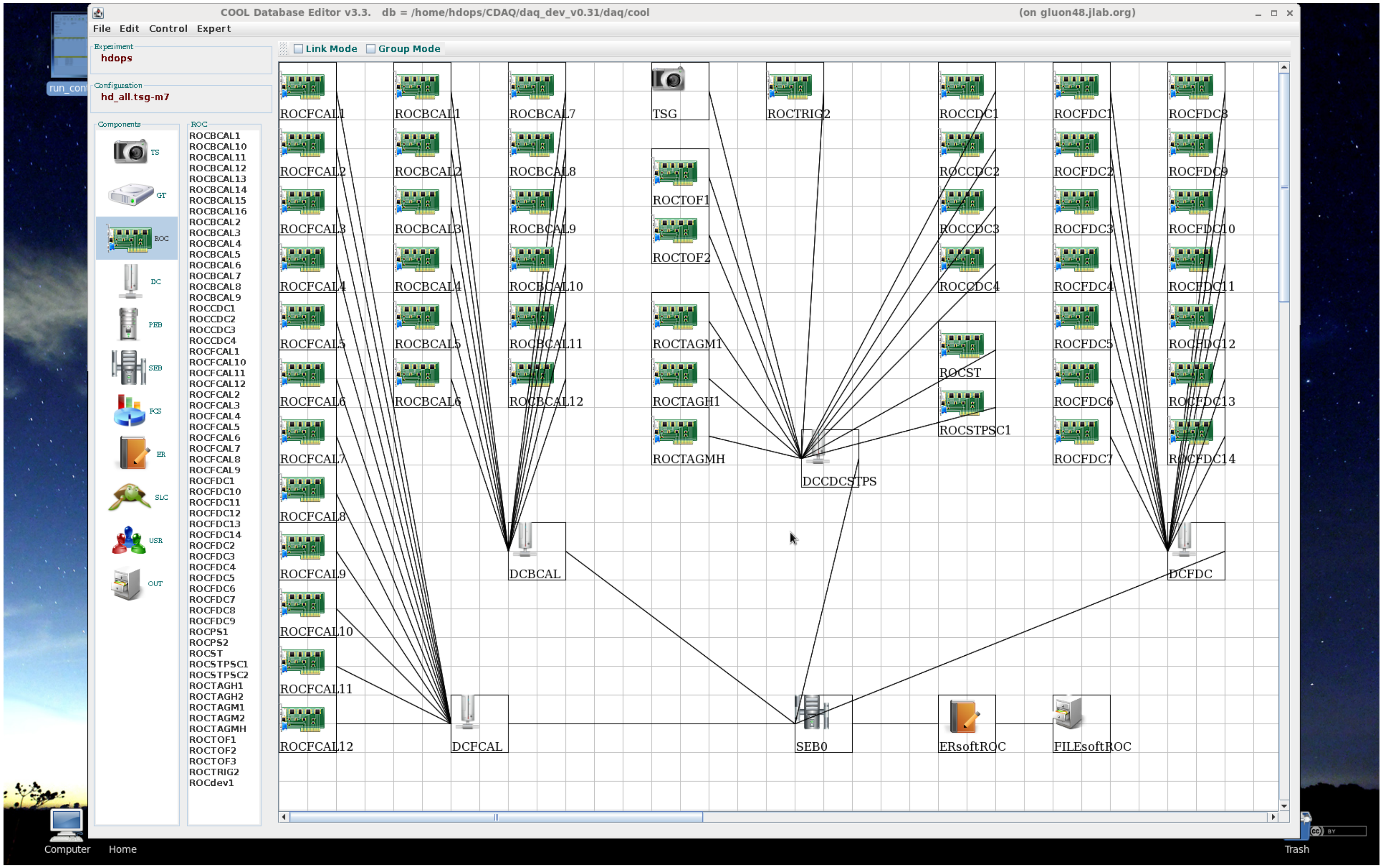
- Modular software, CODA components, with common core functionality.
- ReadOut Controller (ROC) running on embedded Linux in VME.
  - Receives trigger
  - Formats raw data
  - Sends data to EB over network.
- EB builds events from multiple ROCs.
- ET - “buffer manager”
  - Hooks for monitor & filter.
- Event Recorder - writes to disk.



Scalability

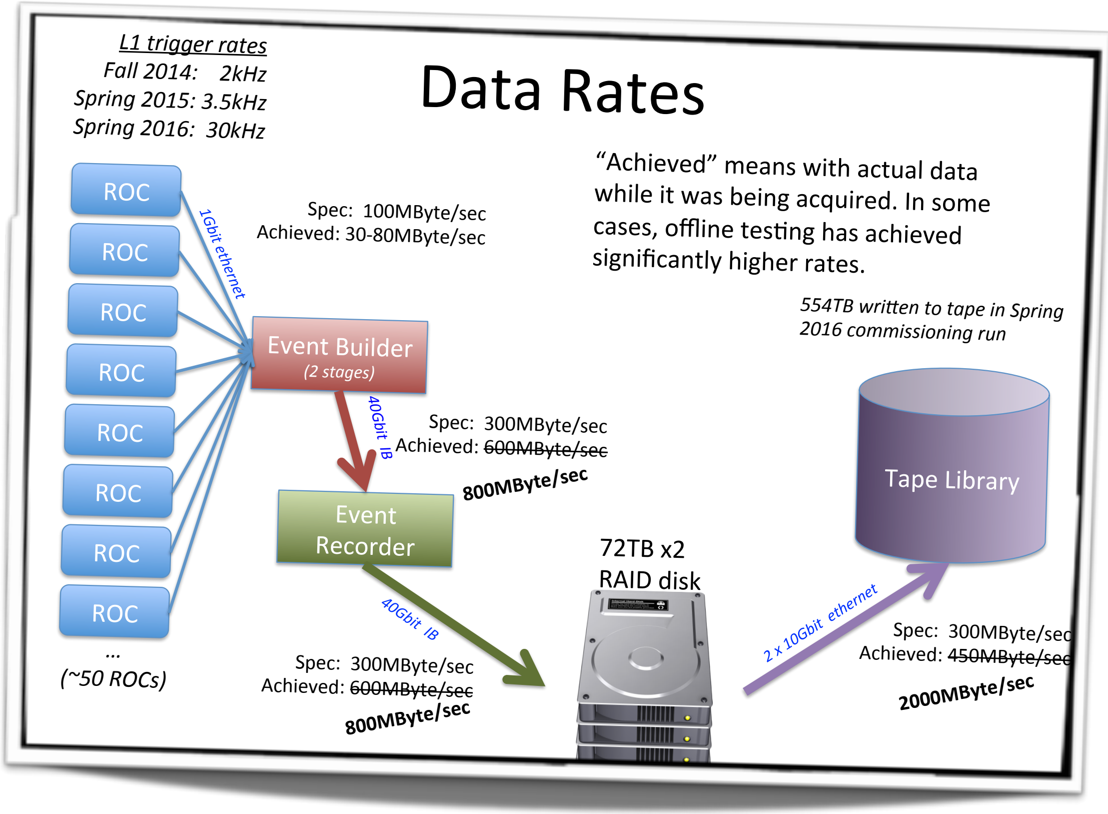
GLUEX configuration:

50 ROCS, 4 Data Concentrators, 1 Secondary EB, 1 Event Recorder.



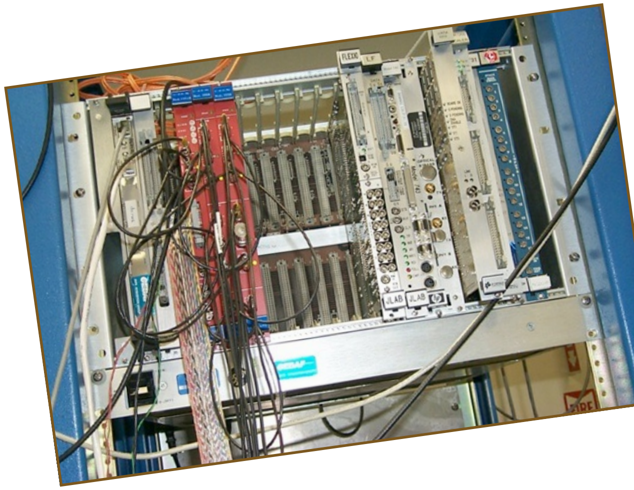
Real world performance

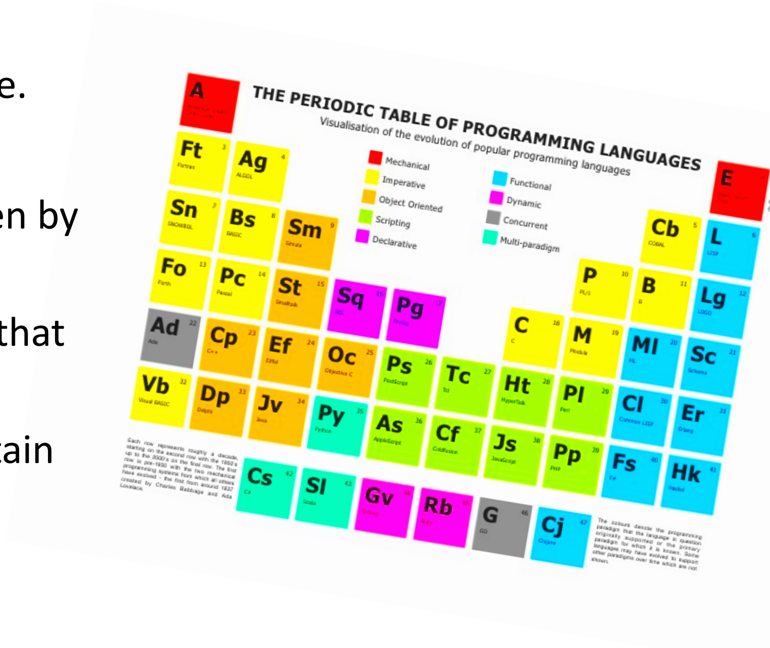
- Spring 2016, low luminosity, 0.8×10^7 γ/s GlueX ran at 800 Mbyte/s !!
  - Remember the design goal was 300 Mbyte/s
- Wondering what rate for 5×10^7 γ/s will be? Maybe 9 GByte/s instead of the expected 3 GByte/s?
- Fall 2016 test the L3 trigger.
- Fall 2018 high luminosity.



Current and future challenges

- VME and busses in general
  - Vendors of ENP DAQ hardware dwindling.
  - Future VME may not be any faster than it is now and may even be slower.
  - Move to high speed serial fabrics rather than traditional parallel backplanes.
    - VXS works well for us but we are one of only a few adopters.
    - Intend testing MicroTCA in 2017.
    - Any solution requires custom FPGA based hardware.
- Software
  - Languages and techniques are constantly evolving, driven by industry. Need to be adaptable to new ideas.
  - Much of CODA “back end” is written in Java, where will that be in ten years?
    - Take care not to jump on a bandwagon with uncertain future.





Challenges continued…

- Experiments are moving towards very loose triggers, or no trigger at all.
  - G byte/s rates at the ROC level.
    - Increased use of firmware instead of software.
      - Physicists write software, firmware harder to change.
    - Data has to be tagged with a timestamp from a high speed global clock.
    - Bandwidth constraints on possible architectures.
  - Event building at these rates becomes a hard problem again.
    - Can't pass all the data through a small number of computers.
  - Storage is currently a bottleneck
    - Heavy reliance on online data filtering and compression.
      - Large compute clusters online with high bandwidth network.
    - Media costs are reasonable but have to run many drives in parallel to store gigabytes per second.
    - Novel storage solutions? Maybe but can't see anything replacing tape soon.

Challenges continued…

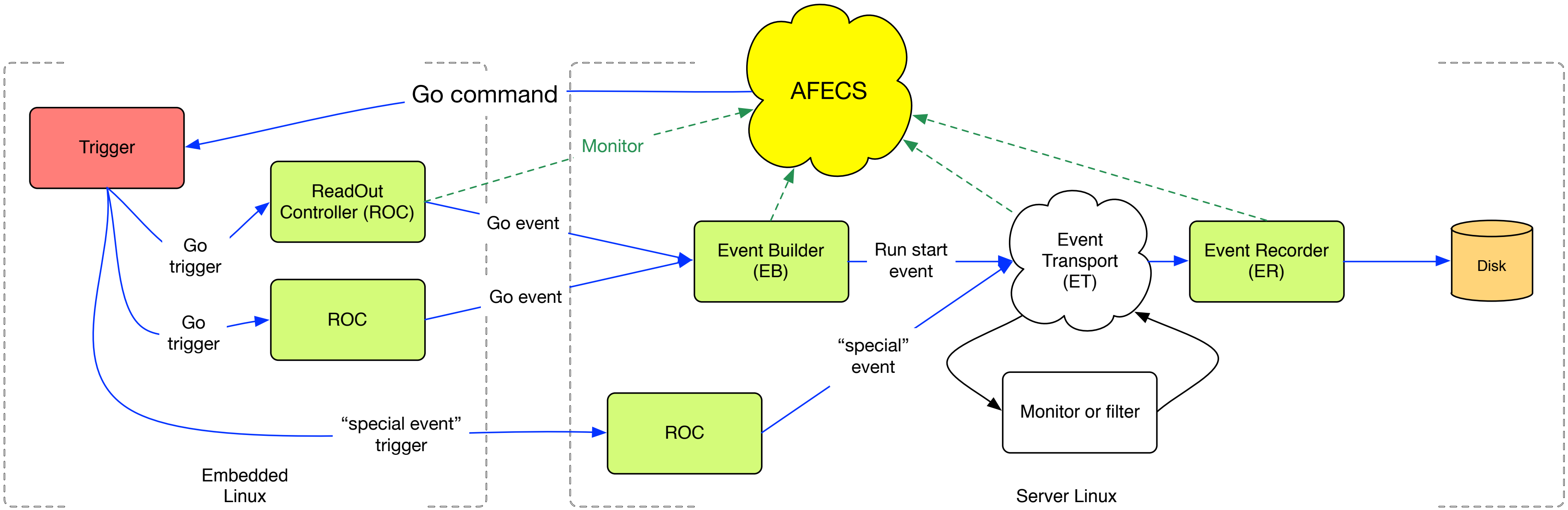
- Experiment complexity is increasing.
  - Configuration of a system the scale of GLUEX requires care to avoid errors that impact data taking.
  - AFECS is a good step forward compared with the previous CODA run control but its inherent complexity makes debugging an issue.
    - When it works it works very well but when it doesn't it can be hard to find out why.
  - Transitions that used to be instantaneous take time.
    - User's find it harder to understand what is going on.
  - In distributed systems symptoms can be disconnected from causes.
    - Unpredicted interactions and behaviors.

Future - Configuration

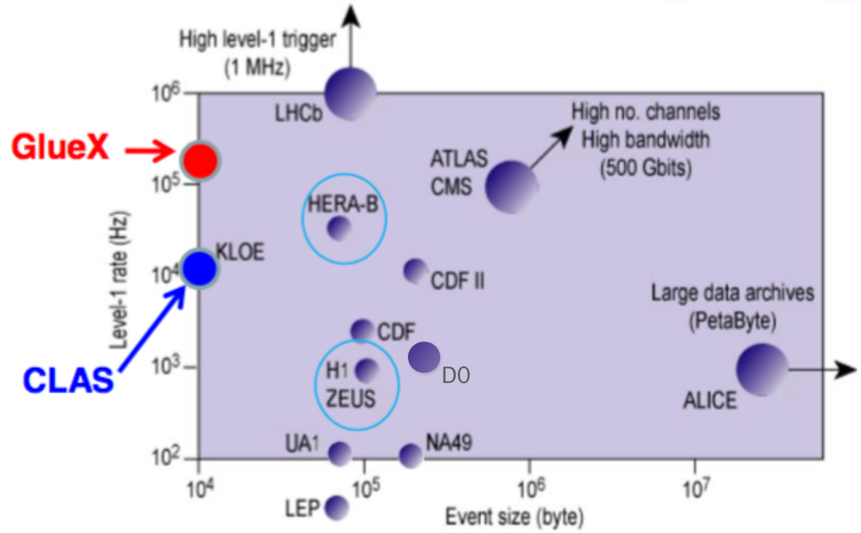
- The control transitions "configure" and "download" were conceived when the cost of starting a new process was high, particularly with the VxWorks OS in use in the 1990's and early 2000's.
  - The idea was to allow rapid reconfiguration of the DAQ.
  - Download caused running program to dynamically unload a plugin "readout list" and load a new one for a different run type.
  - In reality, in an abundance of caution, most users now completely shut down the DAQ and restart everything when they change run configuration.
- In the new system AFECS would start all the DAQ components with configuration parameters passed once at startup, this would replace the current "download" command.
- A configuration change would cause a complete shutdown ensuring a clean restart.

Future - flow control

- A solution to data flow related issues is to move to a **data driven** model.
  - Make CODA components, as viewed from outside, stateless.
    - AFECS then only cares that they are running and ready to take data.
  - The trigger hardware already generates “special” trigger types that ROCs use to generate “marker” events at the start and end of runs.
  - Components would respond to marker events instead of AFECS commands.



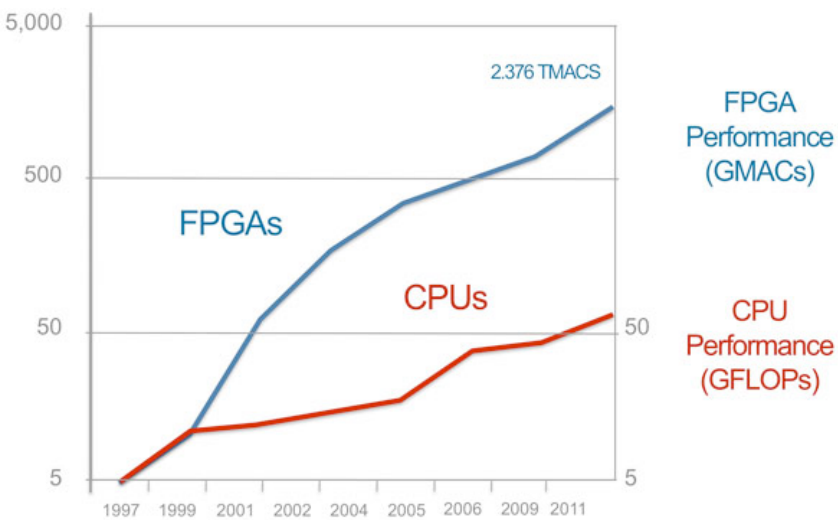
Future - rates

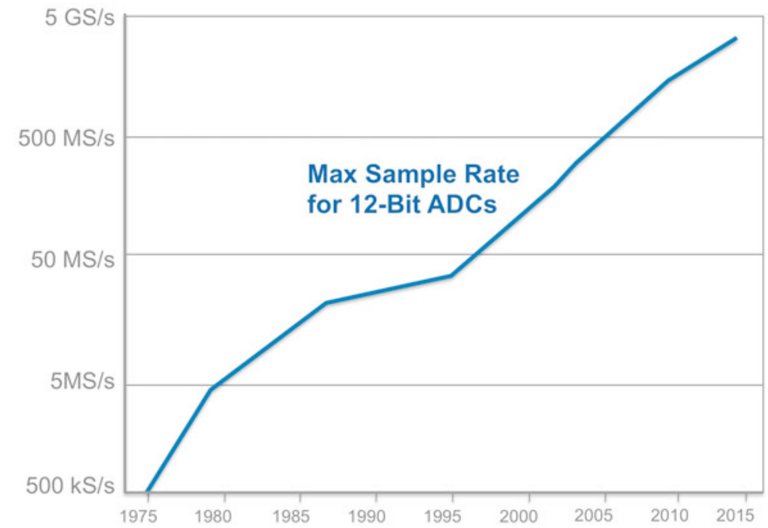


- Look at historical trigger and data rates.
- At JLab
  - mid 1990's CLAS, 2 kHz and 10-15 MB/s
  - mid 2000's - 20 kHz and 50 MB/s
  - mid 2010's
    - HPS, 50 kHz and 100 MB/s
    - GlueX
      - 100 kHz, 300 MB/s to disk.
      - (Last run 35 kHz 800 MB/s)
- FRIB - odd assortment of experiments with varying rates
  - LZ Dark matter search 1400 MB/s
  - GRETA 4000 channel gamma detector with 120 MB/s per channel. (2025 timescale)
- RHIC PHENIX 5kHz 600 MB/s
- RHIC STAR - Max rate 2.1 GB/s average 1.6 GB/s
- SoLID ~ 30 GB/s front end.
- Looking at the historical trends the highest trigger rate experiments increase rate by a factor of 10 every 10 years.

Trends in trigger and electronics

- FPGA performance is increasing faster than CPU performance. Why? There is a delay between when technology is developed and when it becomes affordable for use in custom electronics. So there is room for growth over the next ten years.

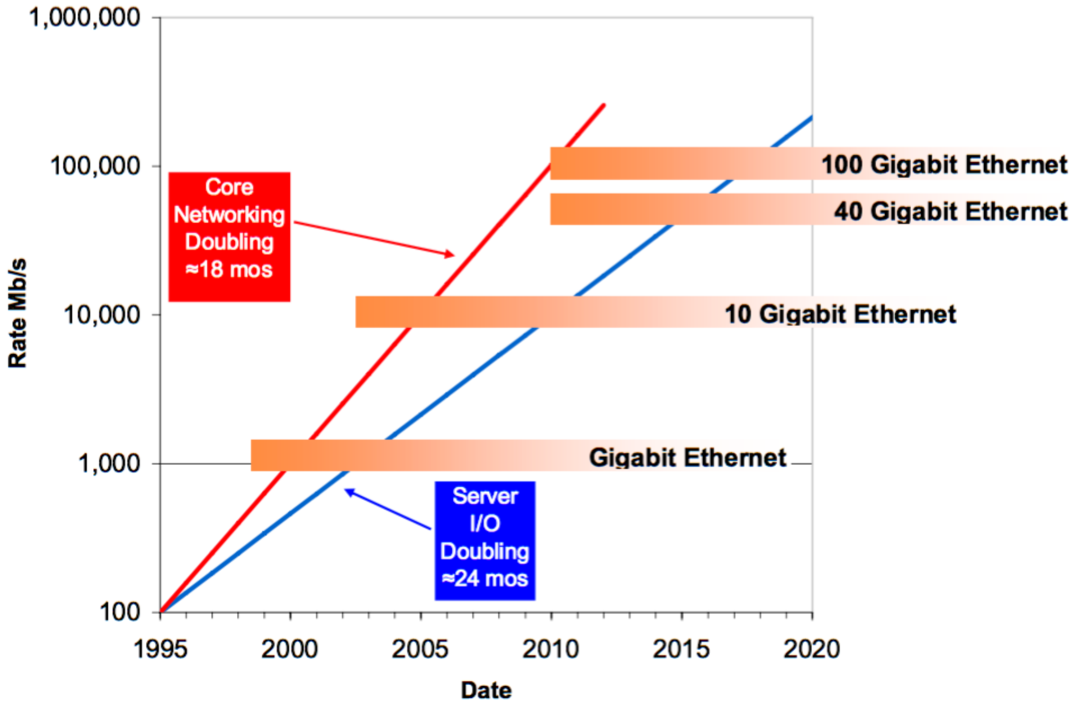




- Current trend is to push some functionality currently performed in software running on embedded processors into firmware on custom electronics. This will probably continue.

Trends in data transport

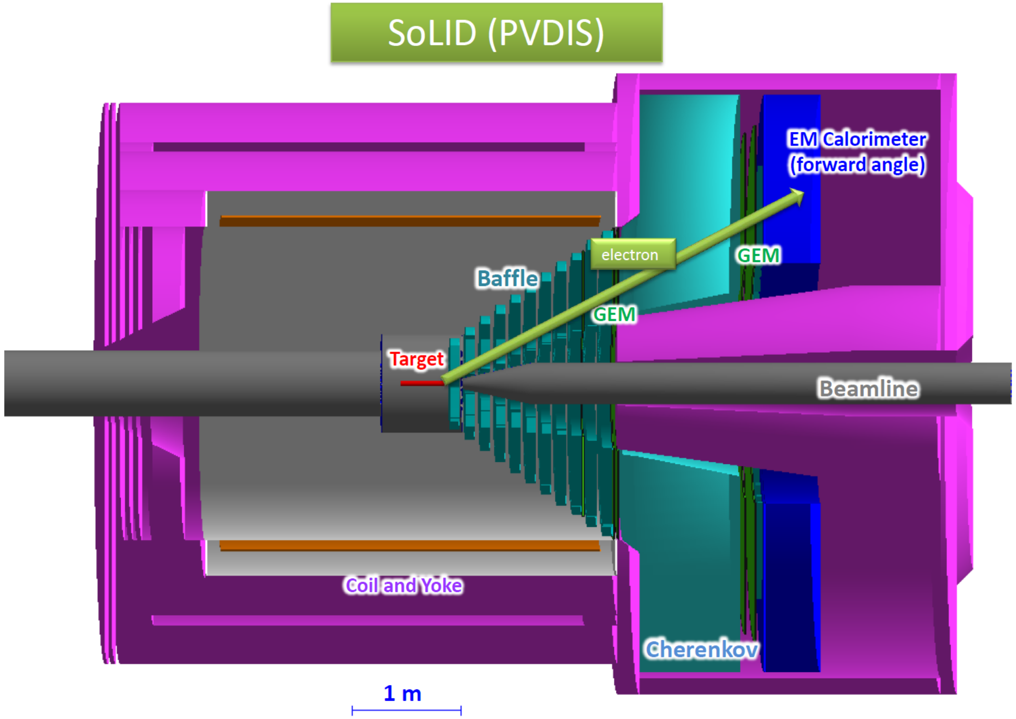
- Network speed trend 2x every 18 months.
- Server IO trend 2x every 24 months.
- Network technology is shown as a horizontal bar. It is introduced at the left of the bar and becomes cheaper.
  - 10 Gb/s appeared in mid 2000's but we could only afford it in any quantity maybe 2010.
  - Interestingly this was just the time that server I/O caught up.
- Takeaway fact - if you upgrade early it will be costly and there will be no computer fast enough to put on the end of the link. Fast links start out as switch to switch links where cost is less of an issue.



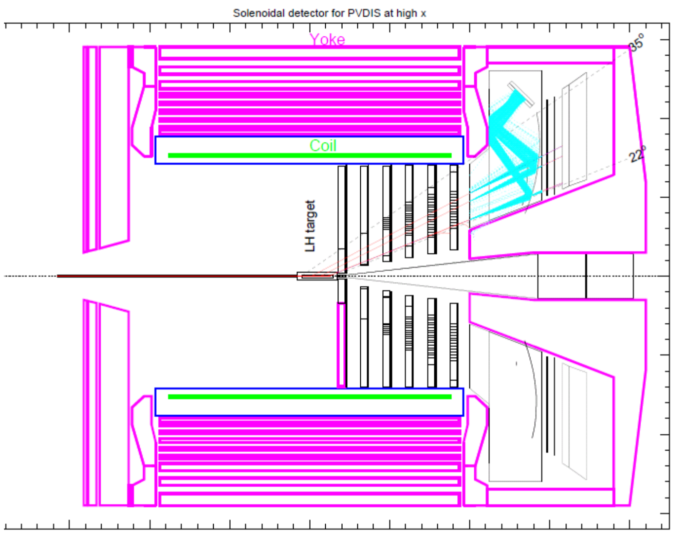
IEEE 802.3 Higher Speed Study Group - TUTORIAL

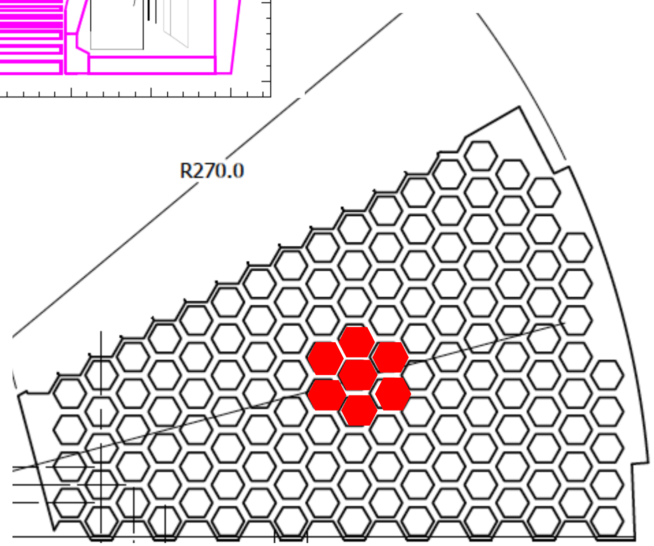
Future experiments, JLab - SoLID

- SoLID is an experiment proposed for installation hall-A at JLab.
- The detector has two configurations. In the PVDIS configuration electrons are scattered of a fixed target at high luminosity.
- The detector is split radially into 30 sectors, 500 kHz trigger rate and 30 GB/s data rate from whole detector.



PVDIS in SOLID





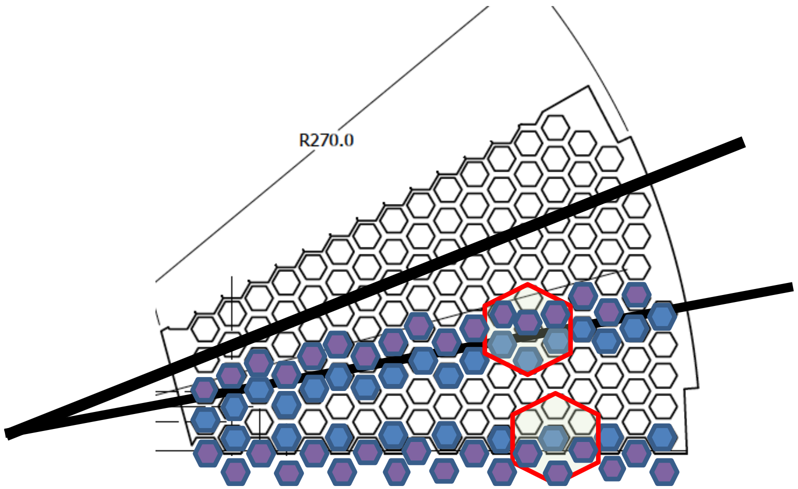
Parity violating p(e,e')
Rate: ~ 500 kHz

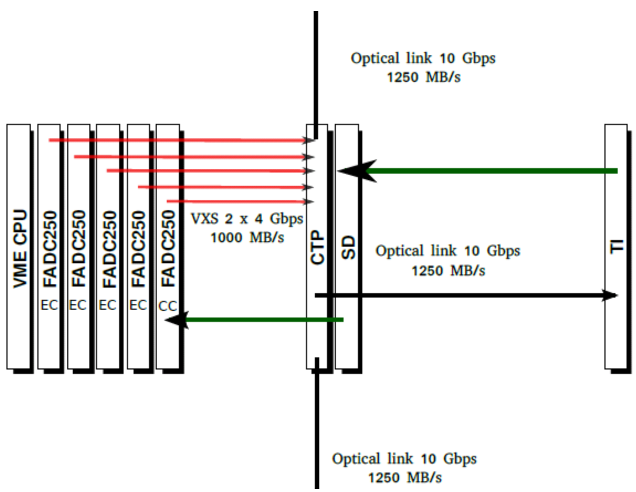
DAQ
- GEM planes for tracking
- Calorimeter+Cerenkov
  - PID and trigger

PVDIS configuration
- Segmented by 30 sets of curved baffles design to accept desired momentum range and block neutrals and positive particles

Trigger
- Energy sum of 7 calorimeter blocks (like HPS)

PVDIS - Segmented DAQ





Strategy

- Segment into 30 independent DAQs (following baffle segmentation)
- ~20 kHz / DAQ
- GEMs also follow segmentation
- 1 FADC crate + GEM ROC per DAQ

>50% of clusters require calorimeter blocks from adjacent segment

CTP or VTP receives streaming trigger data from FADCs. Makes energy sum in firmware.

VTP also receives data from adjacent VTPs for calorimeter blocks in adjacent segments

PVDIS: 30 coupled DAQs

VTP to VTP communication solves trigger problem for overlapping shower clusters.

But DAQ does not have access to full ADC (calorimeter + Cerenkov) data for proper offline analysis of edge events.

Possible solutions:

1. Trigger adjacent DAQs. Always or if adjacent DAQ trigger info used in trigger.
   Max rate: 60 kHz
   Need to not trigger adjacent segment GEMS to avoid GEM DAQ rate limits
   Will need to rebuild full events offline using timestamps.

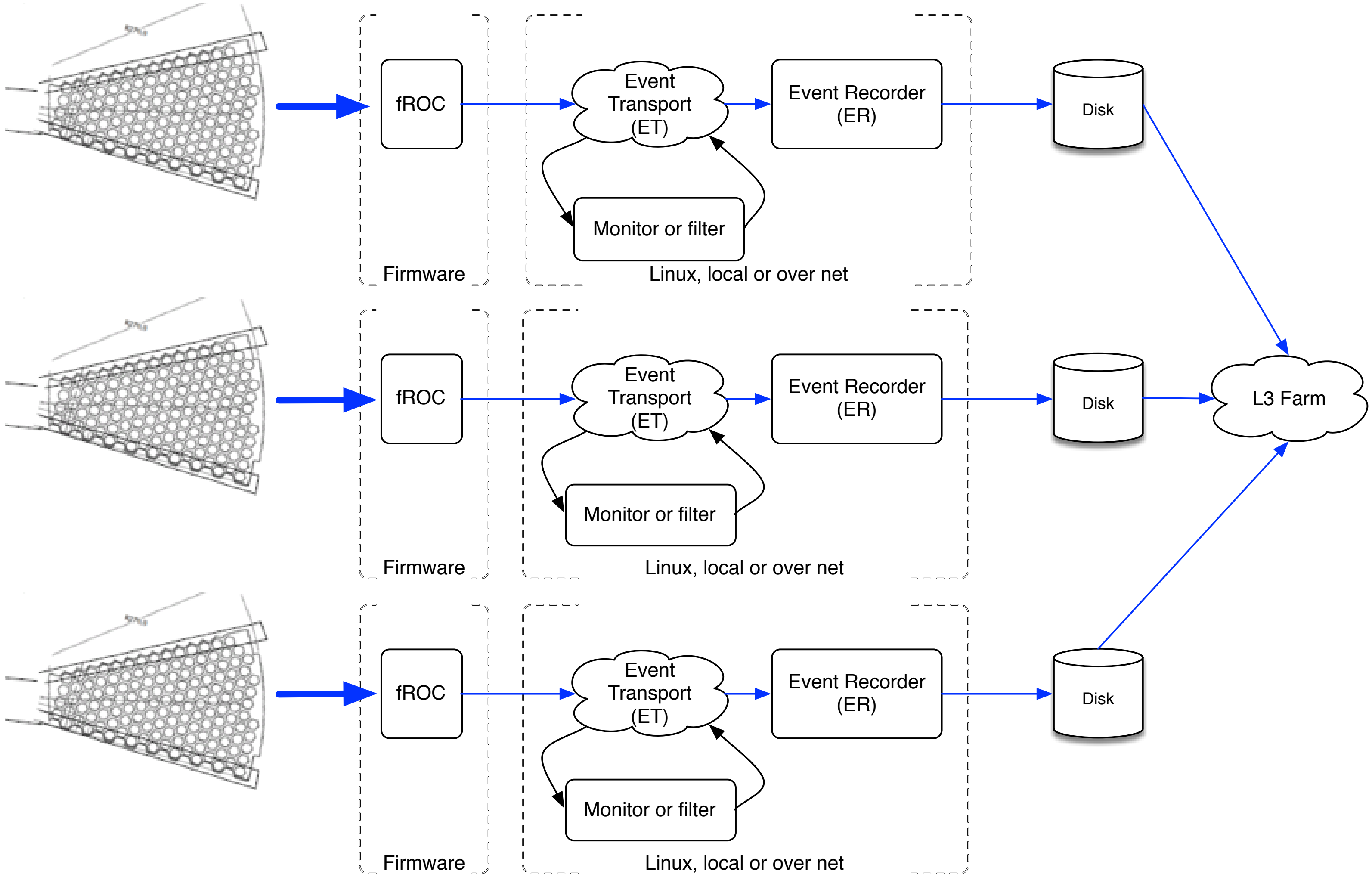
2. VTP requests data from required ADC channels from adjacent DAQs over intercrate optical links.
   Triggering ADCs without generating an event
   Complicated logical/firmware
   Latencies and deadtimes?

3. Revisit segmented DAQ concept. Run DAQ at 500 kHz? (not really serious about this)
   Heavily sparseify readout.
   Keep GEM segmentation and sparseify GEM triggers.

Need to set up a 2 or 3 crate test-stand to try out possibilities.

SoLID PVDIS DAQ

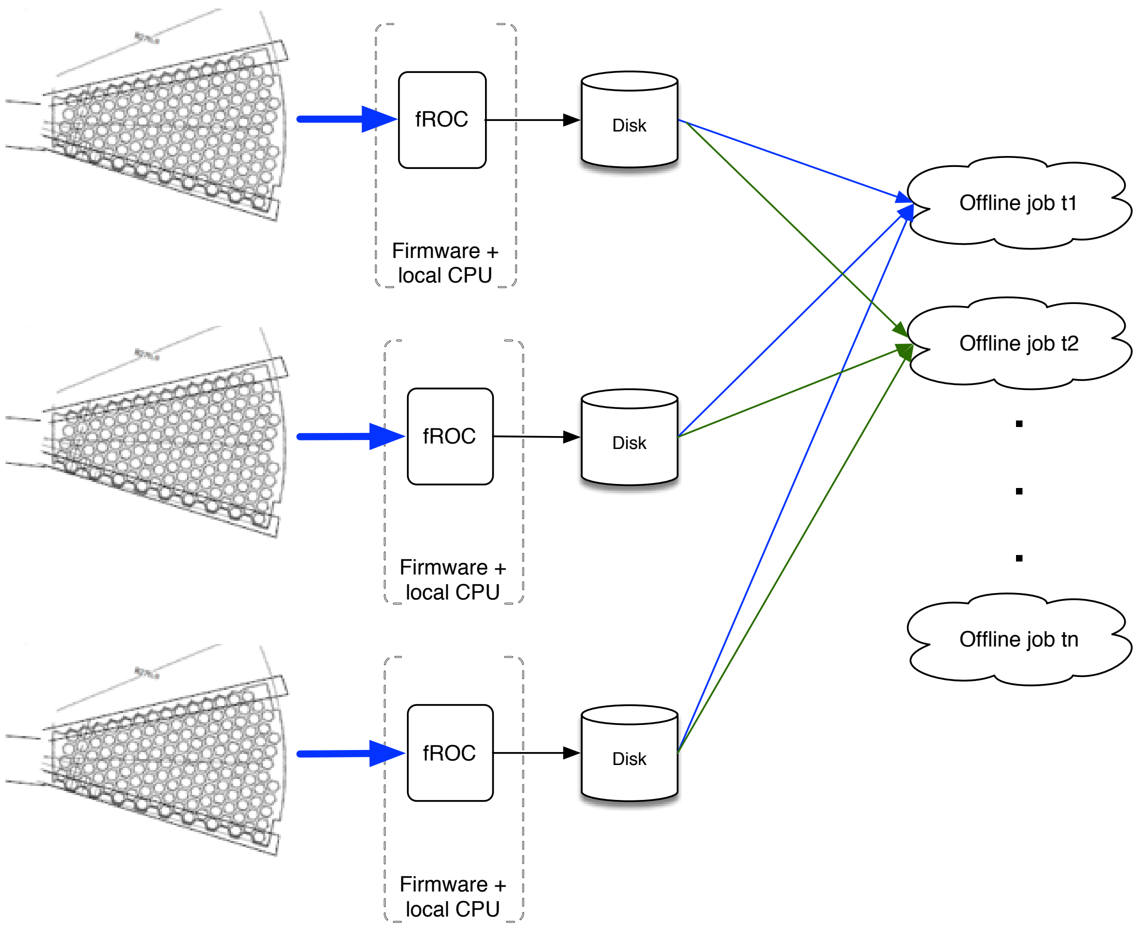
- ROC implemented in firmware on custom electronics, no EB needed so stream data to ET system.



repeat 27 more times for 30 streams…

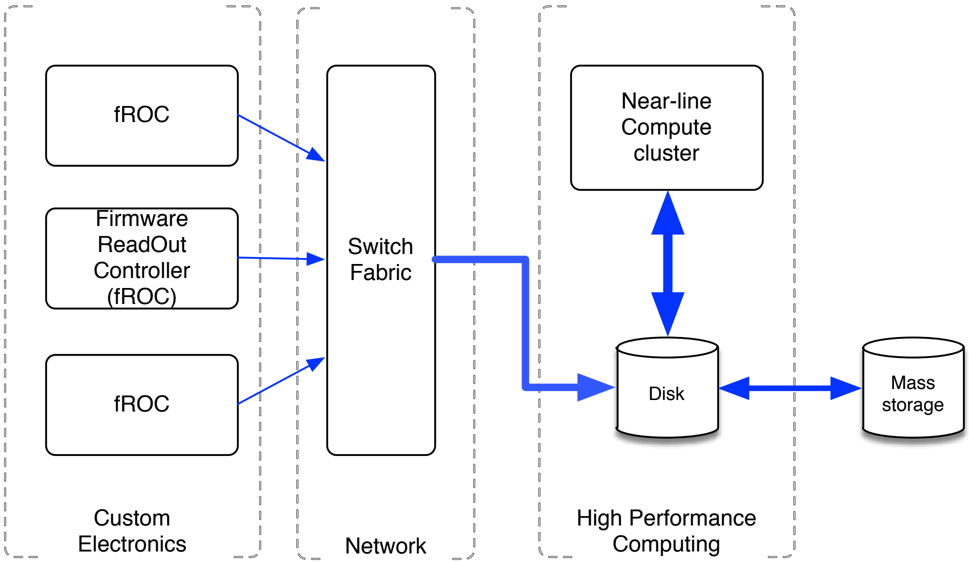
SoLID PVDIS DAQ, future tech. alternative

- If disk speed advances enough stream data directly to disk.
- Run L3 and other filters and monitors semi-offline, i.e. soon after data taking but not online.
- Much easier tuning of L3.
- Simpler DAQ design.
- Refine ideas for post SoLID DAQ systems.



Streaming DAQ, EIC, maybe SoLID

- Digitize and zero suppress in custom electronics.
- Firmware ROCs are very simple, trigger sends "start" event and they read out until the trigger sends a "stop"event.
- Stream the time-stamped data through a network directly to temporary storage.
- High performance compute system processes the data "near-line" implementing a software trigger.
  - Several different triggers in parallel? Safe debug of trigger.
- Data surviving trigger or output from online processing migrates to long term storage freeing space for raw data.
- Simpler architecture = more stable DAQ
- needs affordable versions of :
  - High bandwidth network storage ✔
  - High bandwidth network ✔
  - Time stamped streaming ADCs ✔
  - Terra scale computing ✔
  - Firmware ROC ✘ ✔
  - DAQ software ✘ ✔
  - Software to organize time-stamped data ✘



Concluding remarks

- Often the limiting factors in DAQ design are available technology and budget. It is not surprising that trigger and data rates follow an exponential trend given the "Moore's law" type exponential trends that technologies have been following.
- **What matters is not when a technology appears but when it becomes affordable**. It takes time for a technology to become affordable enough for someone to use it in DAQ.
- If current technology trends hold then in the five year timescale much simpler DAQ architectures will be possible that have significant advantages.
  - Ease and flexibility of implementation.
  - Stability.
  - Cost.
  - Speed.
  - Accessibility - physicists are used to writing software in an offline environment. Let's move what was traditionally an online interaction with data into the same arena.
- Are researcher limiting their vision of future experiments because of assumptions based on current technology?