Streaming Readout – the Jlab perspective

Streamin'? you got to be dreamin'

Graham Heyes

Wednesday May 13th 2020

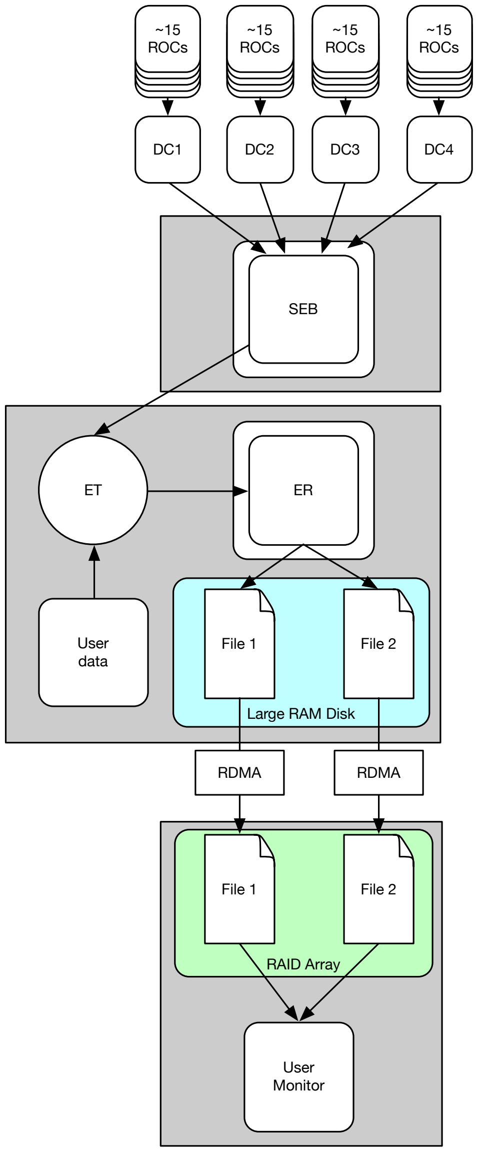
What can you say in 15 minutes?

- Reorganization

- Origins
- Compromise
- Evolution
- Modes
- Practicalities
- Concepts
- Comparisons
- Path forward

Origins

- For many years, the design of data acquisition (DAQ) systems has assumed that:
  - *The data rate from a large detector is impossible to acquire with an affordable data acquisition system without a trigger to reduce event rates.*
  - *Even if the untriggered data rate could be acquired it would be impossible to store.*
  - *Even if it could be stored the full dataset would represent a data volume that would require impractically large computing resources to process.*
- Based on these assumptions a trigger-based readout system, CODA, was designed and implemented as the foundation for DAQ systems at Jlab.
  - CODA has existed for 25 years.
  - Many lessons have been learned from its operation and evolution.
  - CODA, as it exists is not a good fit to projected requirements for future experiments.
- It is time to design a new DAQ framework for the future.



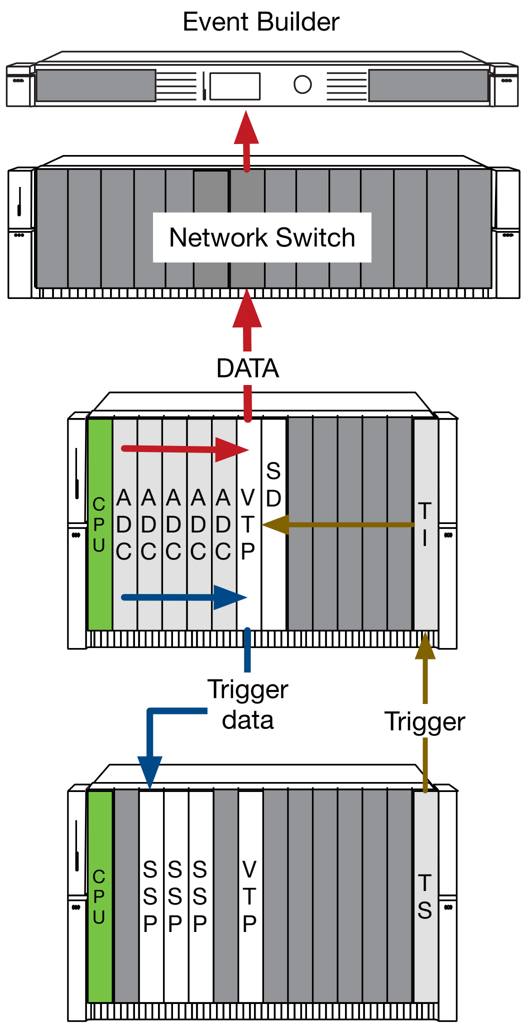
Compromise

- Assuming that it is impossible to acquire all of the data demands data reduction at source, before acquisition.
  - Zero suppression.
  - Gated readout – L1 trigger.
  - Physics based selection – L1/L2 trigger.
  - Limit data rate by discarding information.

- This has some well-known intrinsic limitations for certain experiments.
  - It can carry bias to low-energy particles,
  - It does not deal well with event-pileup.
  - It is not ideal for complex, general-purpose detectors.

- Compromises to limit the data rate inevitably lose information.
  - Detector channels are often summarized by single numbers rather than full wave forms.

- In the case of the current generation of JLab experiments the trigger has become almost as complex as the rest of the DAQ and duplicates effort.



Evolution

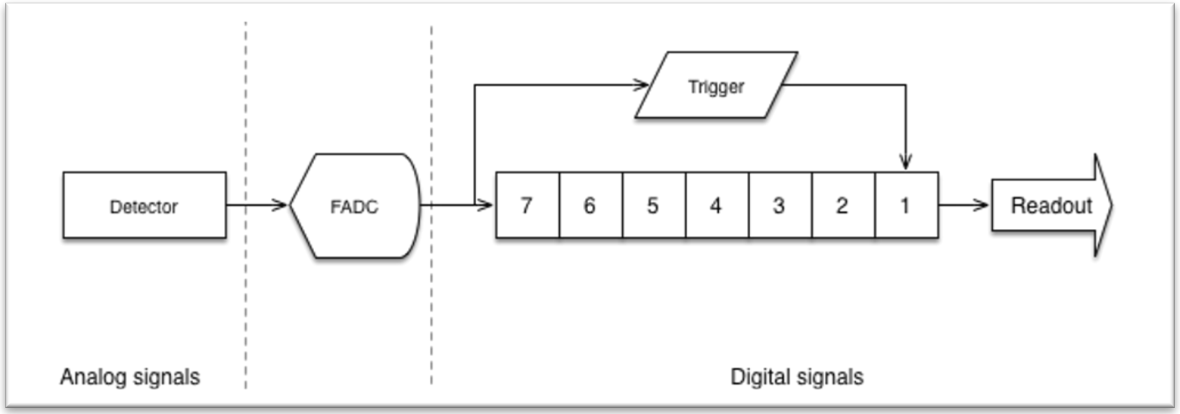
- One of the biggest changes in the last 25 years is the ubiquitous presence of computing in 21st century life.
  - Cost has gone down.
  - Processor performance and bandwidth has consistently improved.
  - Storage is cheaper and faster every year.

- It can be argued that, at least for the nuclear physics experiments envisaged over the next couple of decades, the assumptions that held for most of the last 25 years are no longer valid.
  - Extrapolating forward the evolution of both experiment and technology it is likely that these assumptions may never be valid again.

- To take advantage of these changes the first step is to look at alternative modes of readout.

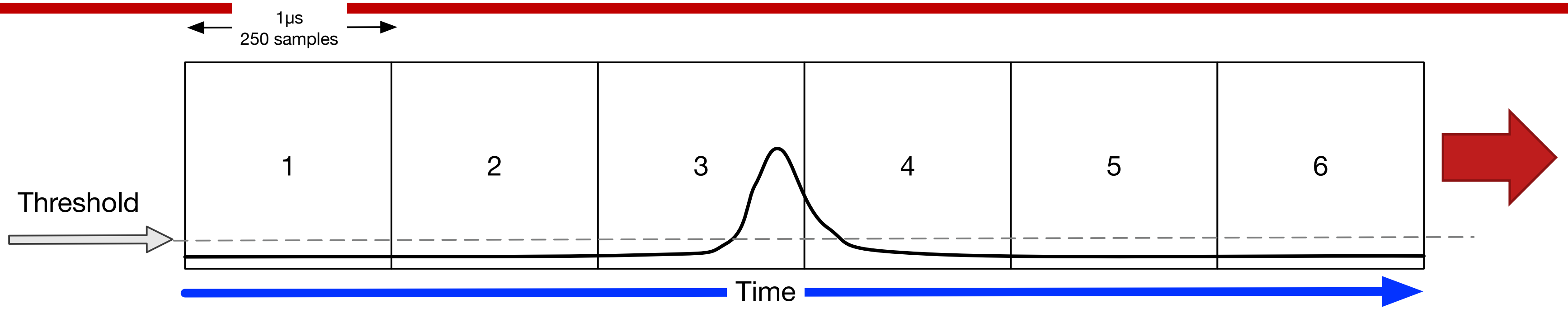
Modes

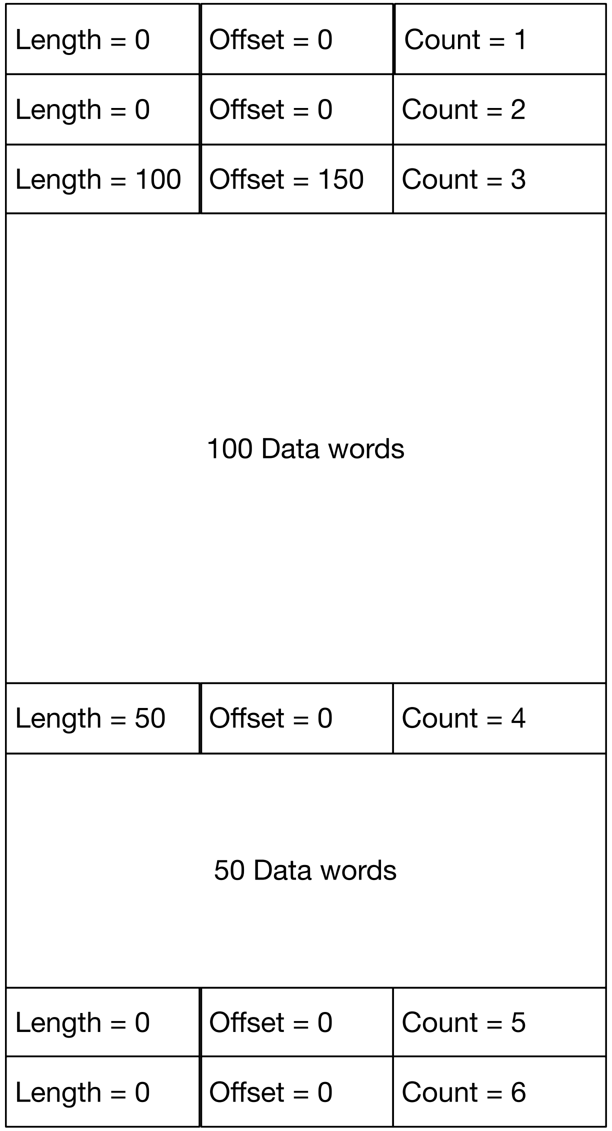
- **Triggered pipeline DAQ** is the mode currently used by CODA at JLab.
  - Analog signals are digitized, and the result held in a memory.
  - A trigger system reads some or all of this data, decides whether the data merits collection or should be discarded, and initiates full readout via the main DAQ data path.



- **Free running DAQ** is an alternative mode of operation of the same hardware.
  - Allow the digitizer to freely run and clock data into the memory continuously.
  - Continuously read the memory on the electronics so that it never overflows.
    - Fill memory then read large blocks which is faster than word by word.
  - A single channel of a 250 MHz flash ADC, if each sample was encoded as 4 bytes, would generate over 1 Gbyte/s of data.
  - Summed over a large detector the data rate would be beyond the capability of any DAQ system that could be affordably implemented.
  - Only practical for very simple detectors.

Modes continued





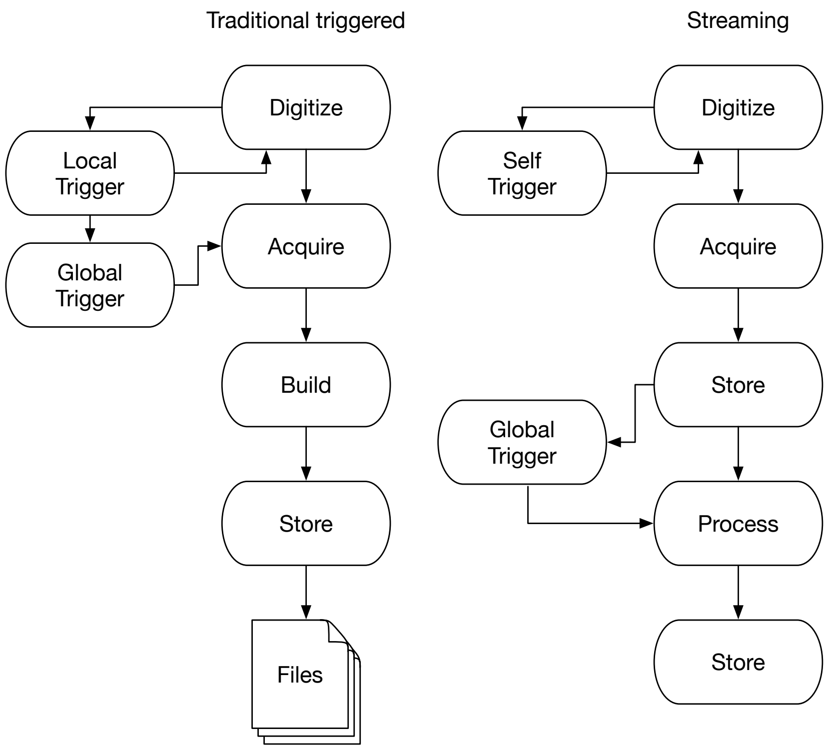
- In the example an ADC clocks a 32-bit sample every 4 nS into a memory.
- 250 samples represents 1µs of time - call this a “frame”.

- The memory can be read as a sequence of six data frames each starting with a length and offset.
  - Frames 1, 2, 5, and 6 contain no data above threshold. So, length 0, offset 0.
  - Frame 3 has length 100, offset 150.
  - Frame 4 has length 50, offset 0.
  - **If length and offset are encoded into a single 32-bit word this simple format encodes the useful data from 6 µS in only 156 words.**

- Encoding data as a continuous sequence of data blocks, each representing a period of time, is ***streaming readout***.
  - There is always a data block even if it just indicates that there is no data.
  - Fixed rate of variable length blocks.
  - If needed these marker blocks can be condensed to indicate longer periods with no data.

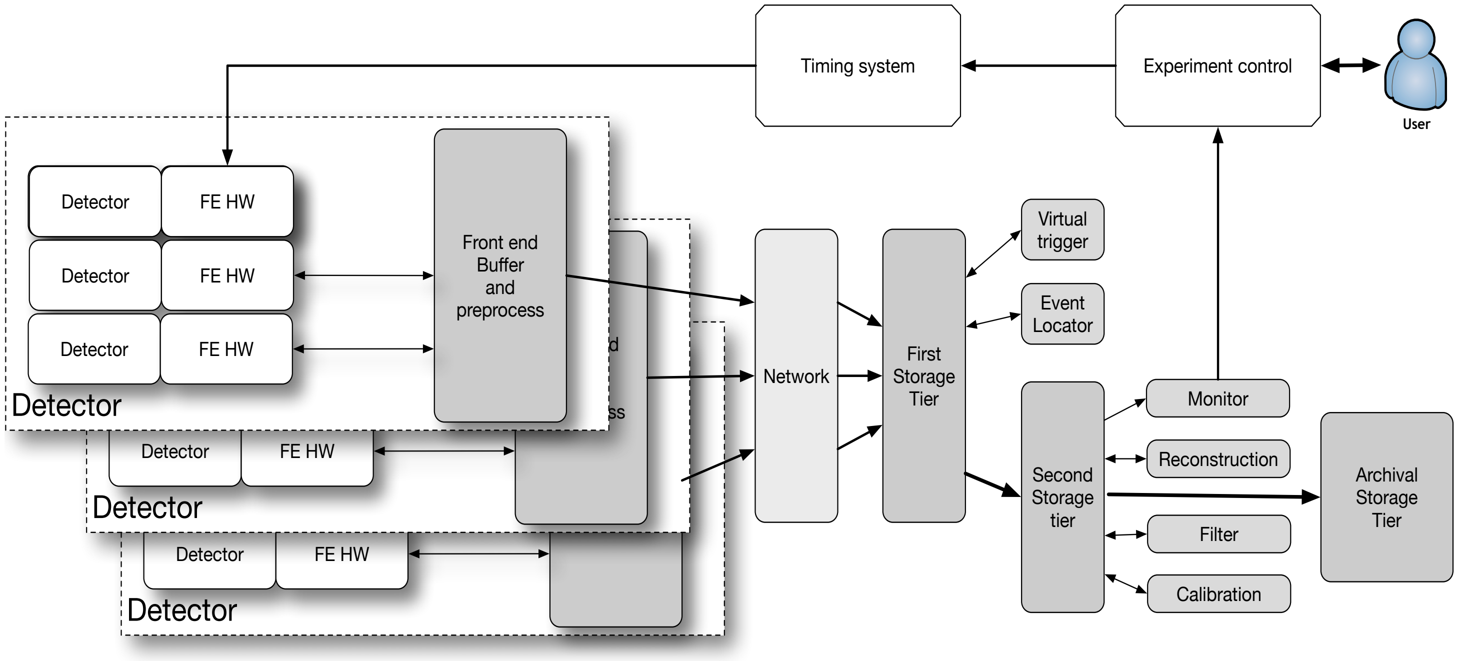
Practicalities

- Streaming mode acquires less data than free running but more than traditional triggered.
- Streaming mode can still take advantage of zero suppression and fast triggers that open gates during which the streaming data is acquired.
  - Steal an analogy off Martin, “streaming is like a video, you’re only interested in the part where something happens.”
- Effectively, compared with pipeline readout, the global trigger has been moved from the start of readout to the end, it is calculated after the data is acquired and is in temporary storage.
  - Much more flexible.
  - Removes randomness – deterministic.
  - Can be implemented in software – parallelism
  - Does not require hardware modification or reconfiguration between run types.



Jefferson Lab

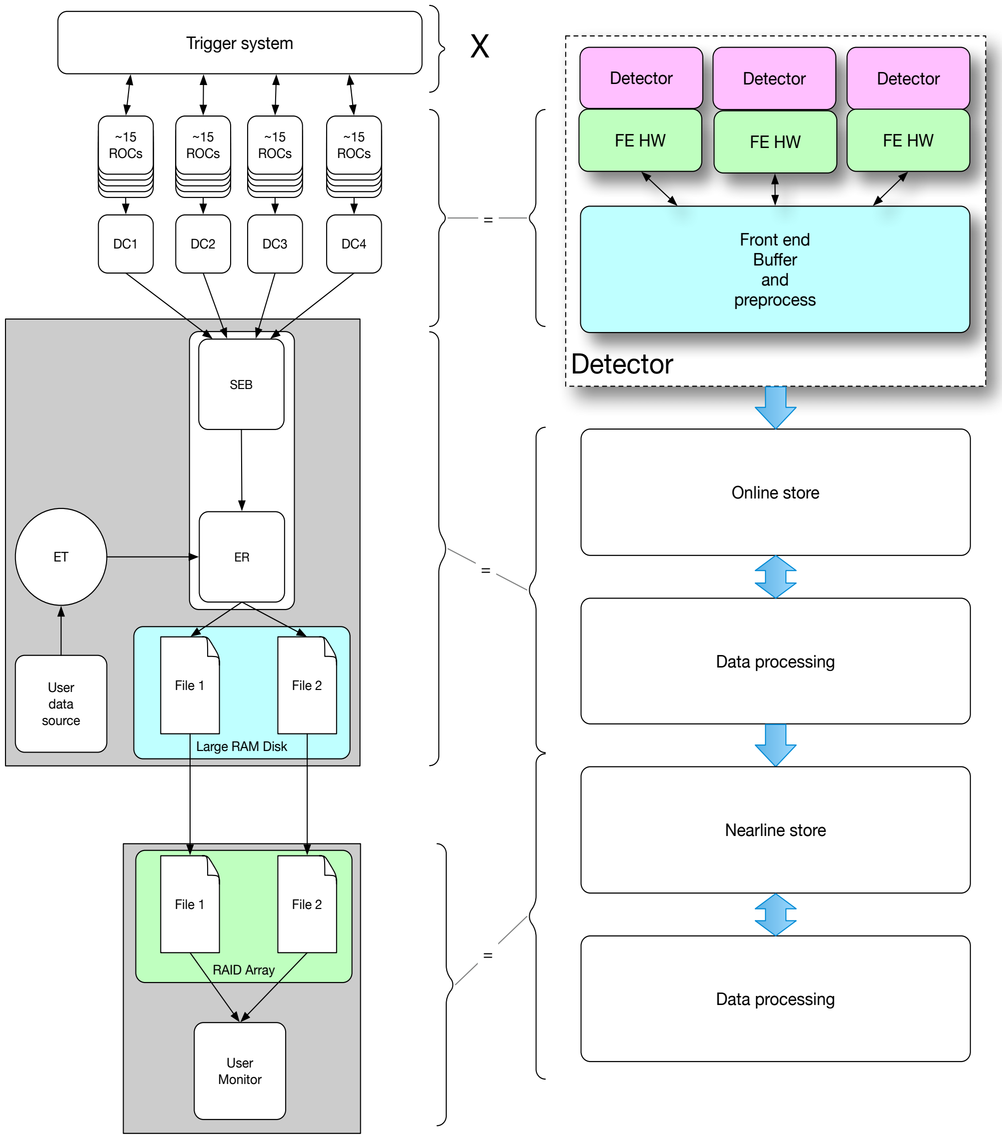
Concepts



- The diagram is one concept for a streaming system – there are many –.
  - The detector is instrumented with Front End Hardware close to the detector.
  - Fiber links stream the data to hardware that merges and preprocesses, Felix at BNL, VTP at Jlab.
  - The first storage tier is a memory based system storing data indexed by detector, channel, and time.
  - Data is held here just long enough for the virtual trigger to identify time regions corresponding to events.
  - From there data is copied to longer term storage (hours rather than minutes) for further processing.

Comparisons

- There is often the argument that this is a radical shift (risky) or that the cost will be prohibitive.
- In the diagram the left side is the GLUEX DAQ and the right a rough concept of a streaming DAQ to do the same job.
- Subsystems with equivalent function are highlighted by the parentheses.
- In each case, except one, the hardware used in the existing GLUEX DAQ has an equivalent which is similar in cost and performance, but configured differently, and with different software.
- The one exception is the global trigger hardware which does not exist in the streaming system.
- Great! Let's switch GLUEX to streaming mode!
  - Not so easy for an existing system.
  - Remember it is configured differently
  - The different software, isn't written yet.



Path forward

- Several critical components are required to implement a practical system :
  - *A stream-oriented timing system.*
  - *A standard low-level stream data format.*
  - *Front end electronics that outputs time-based streams of data in the standard format.*
  - *Efficient and robust streaming data transport.*
  - *A stream oriented random-access data store and associated data access tools.*
  - *A framework for data flow tasks such as event location and virtual trigger.*
  - *A framework for processing tasks such as calibration, monitoring, reconstruction, etc.*

Hardware

Software

- Several groups at Jlab are looking at these areas.
- Chris' Fast Electronics and DAQ group are working on the first three items.
- The newly formed EPSCI group in Scientific Computing are looking at the last four.
- CLAS12 investigating converting a major detector to streaming mode.
- Markus' LDRD is looking at online calibration and analysis of streaming data.
- All in all, a lot going on.