protoDUNE Data Reduction: Various Issues

Brett Viren, Xin Qian

Physics Department

BROOKHAVEN
NATIONAL LABORATORY

DCCG
2017 Apr 18

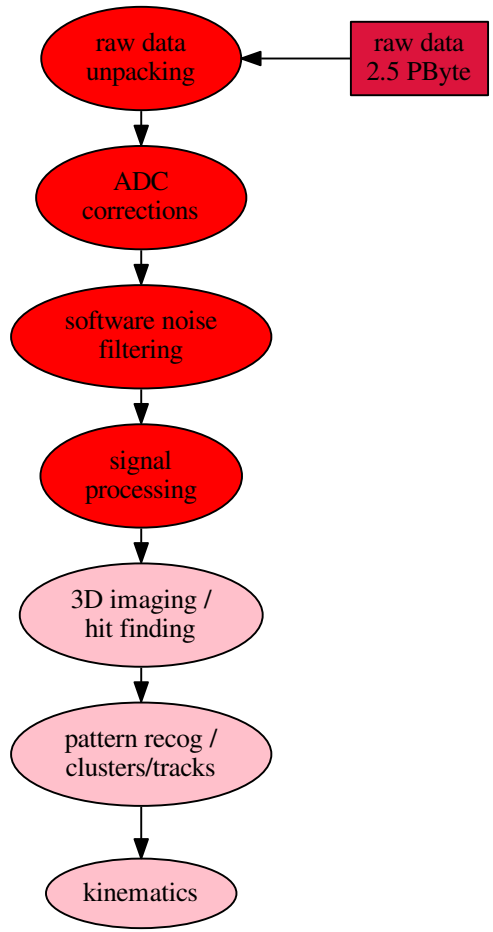
Potential protoDUNE Processing Chains

"Monolithic"

- Conceptually, offline processing consist of a chain of high-level algorithms applied to **2.5 PBytes** of input.
- Competing implementations could imply some parallel branches (eg, two tacking algorithms).
- Rough estimate: 30 min/trig (whole chain), 52M triggers with 6M good beam triggers.
- If use beam trig: 12 beam trig per 5 GByte file, 6 hours per job.

20% CPU red parts (mostly sig.proc.)

80% CPU pink parts (reconstruction)

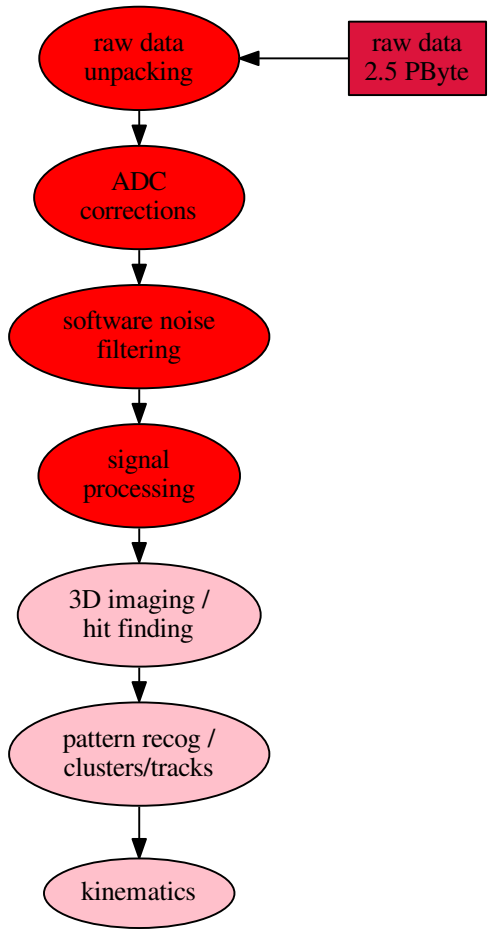


Potential protoDUNE Processing Chains

We need red parts no matter what!

- Unpacking relies on `artdaq::Fragment` and raw DAL from DAQ group.
- ADC stuck code and non-linearity require oversampling and potentially large calibration database to correct.
- Software noise filtering requires early detector debugging by experts, discrete frequency filters, time-based pathology detection.
- Signal processing requires field and electronics response functions, 2D deconvolution.

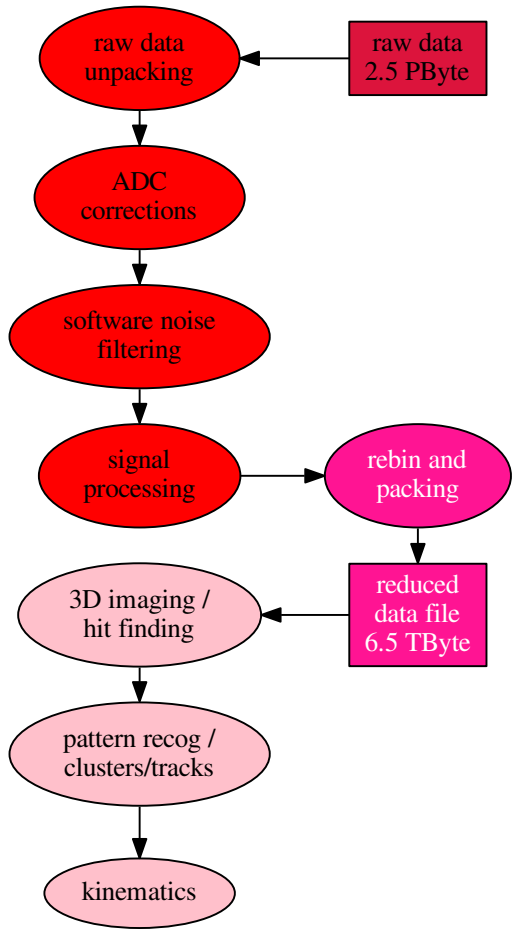
Much expertise gained from 35t + MB.



Potential protoDUNE Processing Chains

"Two Stage"

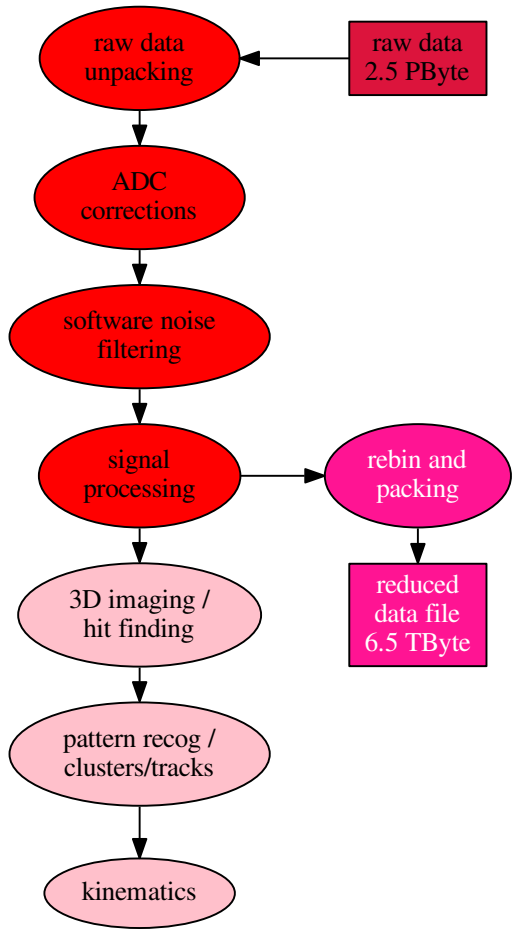
- After signal processing, there is an opportunity to write out a vastly reduced intermediate result (**2.5 PBytes** $\rightarrow$ **6.5 TBytes**).
- Of course options exist: could skip rebin, write **17 TBytes**.
- Either way, by construction, still contains **all signal information** (more on this).
- The chain could be broken into two jobs here, connected by this small intermediate file set.
- First stage, with beam trigger: 1.2 hours per job.



Potential protoDUNE Processing Chains

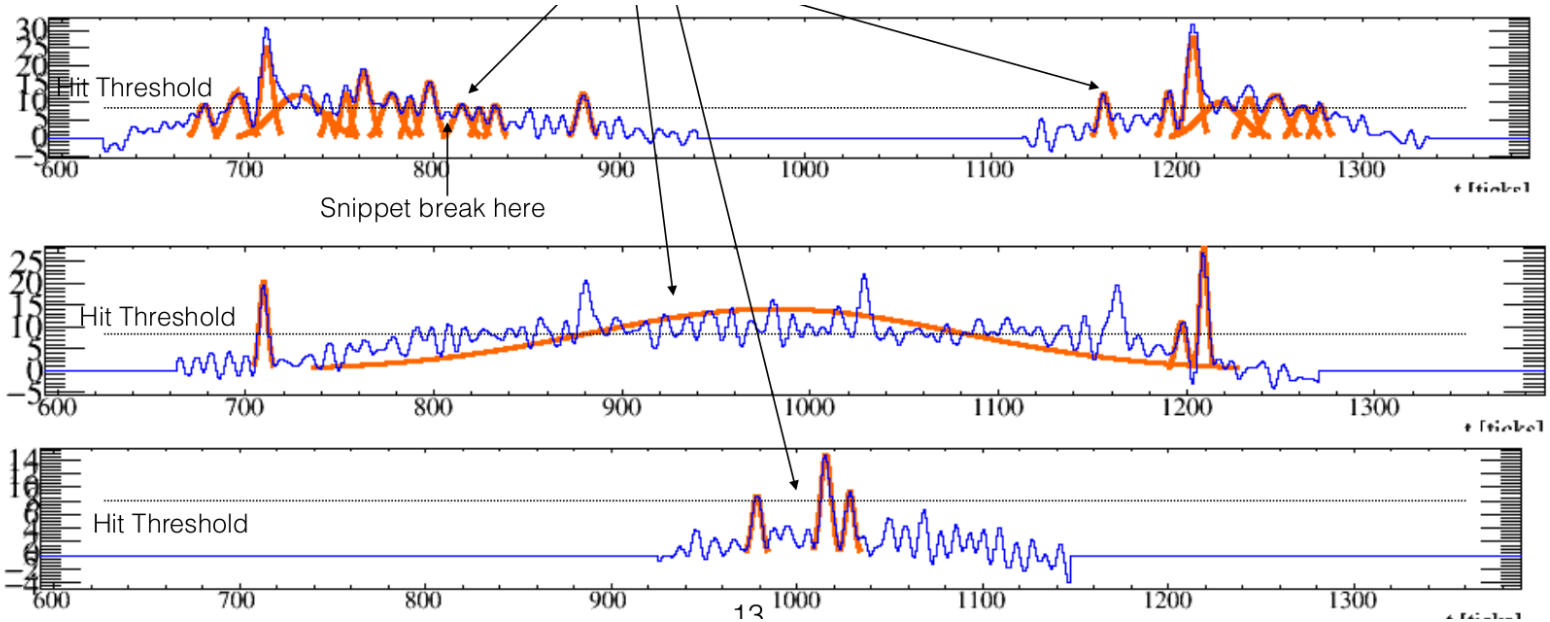
"Side Effect"

- Of course there are yet more options.
- The "Monolithic" process could output the reduced data as a side effect.
- Imaging would "see" oversampled signals and, in principle, allowed to have all the (deconvoluted) noise too.
- Same 6 hours per job/file as "Monolithic" assuming can do beam trigger selection.



Signal Processing Output

What's your mental model of signal processing output?



("Gauss Hit Finder Proposed Changes", Tracy Usher, Feb 2, 2016)
(for illustration, ignore orange)

- Deconvolution with response functions makes signals unipoloar.
- **Signal ROIs** select regions consistent with signal plus some padding into the nearby noise. Portions outside signal-ROIs contain no signal information, **by construction**.
- Oversampled at 2MHz.

Before SigProc: Oversampling Is Needed

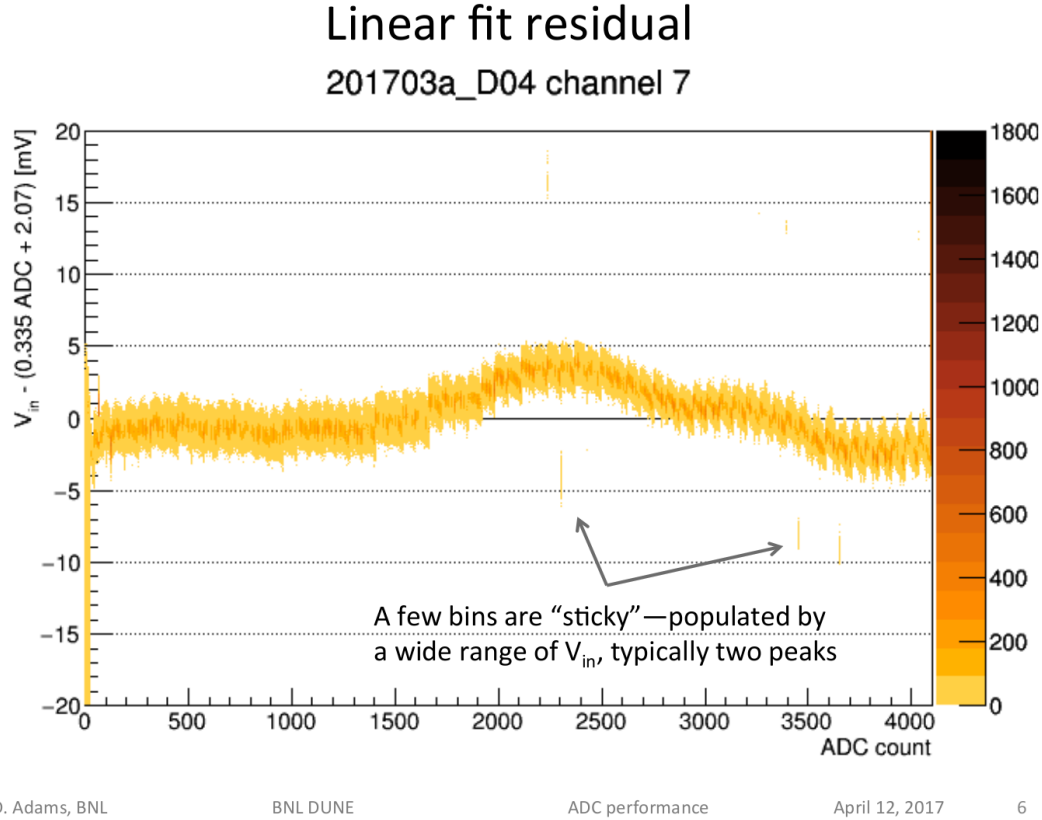
To mitigate ADC "sticky bins" one imagines some algorithm like:

```
if ADC[i] in sticky_list:
  ADC[i] = interpolate(i, ADC)
```

The `sticky_list` is per channel but **hopefully** not a function of time.

Because of oversampling, interpolation should be okay if it's across an isolated sample.

Acceptable? Needs some study?



(D. Adams CE Testing eg as presented to DUNE CE task force)

After SigProc: Oversampling Is Not Needed

Because of shaping....

Nyquist-Shannon sampling theorem if there is **no power** above a frequency f_{edge} then sampling at a frequency $f_{sample} = 2f_{edge}$ obtains **all available information** about a waveform.

Corollary whatever power is above f_{edge}, when sampled at $2f_{edge}$ gets **aliased** into lower frequencies causing **unwanted artifacts** in the waveform.

Q: What determines f_{edge}? **A:** The "cut off" of a low-pass filter.

Q: What low pass filters (aka smearing) exist in our data?

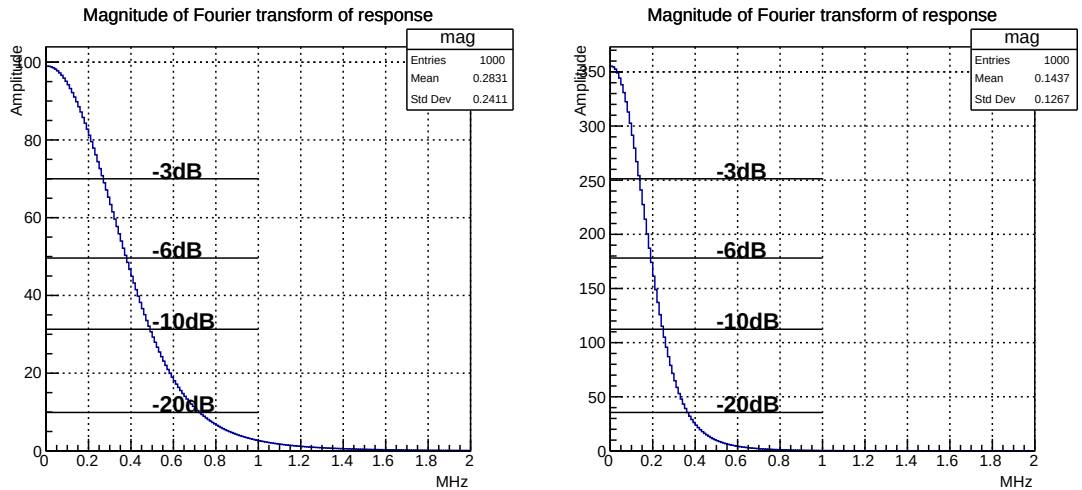
- The explicit anti-aliasing filter in FE electronics.
- The Wiener filter in used signal processing to maximize sig/noise.
- Variation in field response across one wire region.
- Long-range induction effects.
- Diffusion of drifting electron in the LAr.

Q: In an analog, real world what is the **proper cut-off**?

A: Look at a few and measure oversampling as:

$$M = \frac{f_{sample}}{2 \cdot f_{edge}}, \text{ want } M > 1.0 \text{ with minimal aliasing.}$$

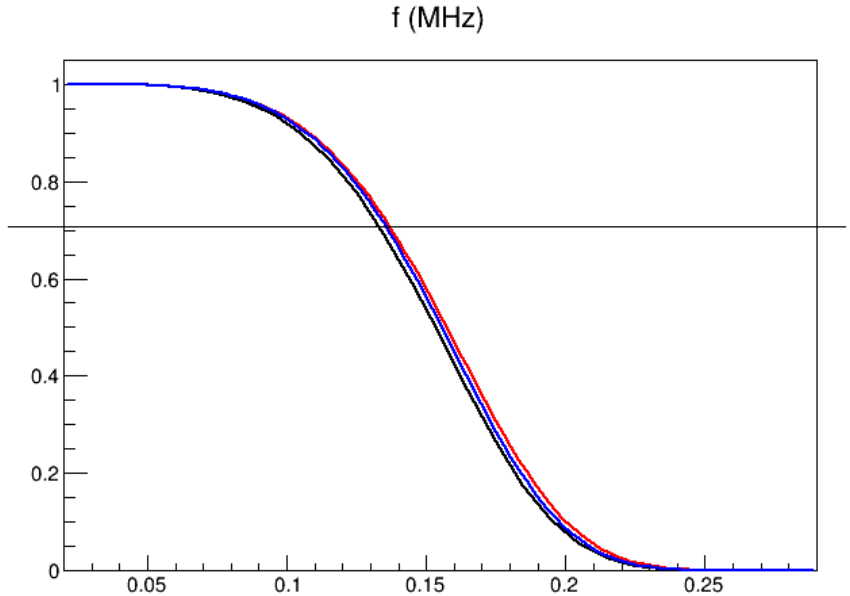
Anti-Aliasing Filter in FE Electronics



Fourier amplitude of electronics response
for 1μs (left) and 2μs (right) peaking times.

- **AA filter has long tails**: 1μs filter still has $\lesssim$5% amplitude at the Nyquist frequency (for $f_{sample} = 2$ MHz) $\implies$ one reason for using 2μs filter.
- Industry rule of thumb: $f_{edge} \equiv 2 \cdot f_{3dB} = 0.30$ MHz, (for 2μs filter), M$=$ 1.11 for $f_{sample} = \frac{2}{3}$MHz
- Another eg: at -20dB (10%) the 2μs filter has $f_{edge} = 0.35$ MHz and has: M$=$ 1.43 at $f_{sample} = 1$MHz, M$=$ 0.95 at $f_{sample} = \frac{2}{3}$MHz

SigProc filter to maximize signal/noise



Current MicroBooNE filter. Line is at -3dB.
Ideal shape is function of total smearing in data.

- A filter like this is required for "non crazy" deconvolution results.
- **Much sharper cutoff than FE AA filter**, nothing above 0.25 MHz.
- At $f_{edge} = 2 \cdot f_{3db} = 0.26$ MHz, M=1.3 at $f_{sample} = \frac{2}{3}$MHz.
- At $f_{edge} = f_{20dB} = 0.20$ MHz, M=1.25 at $f_{sample} = 0.5$ MHz.
- At $f_{edge} = f_{cutoff} = 0.25$ MHz, M=1.00 at $f_{sample} = 0.5$ MHz.

Even at rebin-4, post-sigproc data satisfies Nyquist-Shannon.

Reminder of the some resource numbers

DocDB 2089 proposal.

- `rebin(1,2,3,4)` $\Longrightarrow$ (150×, 290×, 415×, 540×) reduction
  - assumes 16 bit packing and "level 9" ROOT compression
  - only TPC data
- red parts need 2700 cores **IF run in "keep up" mode**.
  - Cores, not nodes.
  - 2700 core × 60 days = 4M core-hours.
    - Pertinent number if we do delayed processing (2017 we have 27M)
  - Is there any concern about LArSoft RAM footprint?
    - assumed to spread 2700 jobs out on 200 nodes with 20 core/node.
    - or, if delayed processing, does 4M CPU-hours double to 8M (make 2CPU/job request just for the extra RAM)?
    - RAM is less a problem for dedicated data reduction job than for a full reconstruction job.