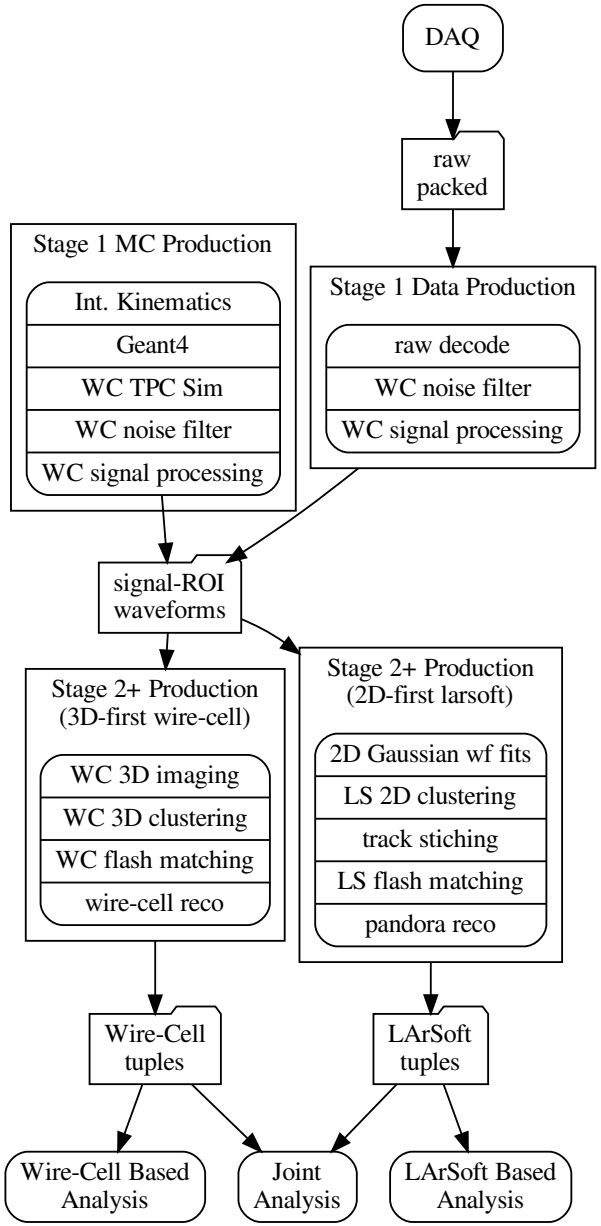
A View of DUNE Production Software

BV

August 10, 2020

High-level View of DUNE FD TPC Processing

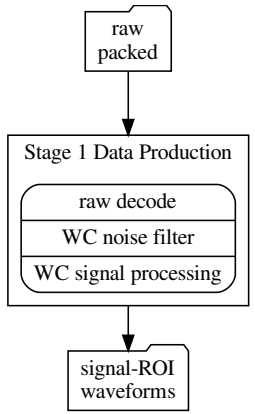


- Stage 1: noise filter + signal processing.
  - Fairly well defined stage.
  - Noise filtered, deconvolved signal regions of interest (ROI), zero suppressed,
    - ★ $\approx 10^3$ data volume reduction.
- Stage 2+: reconstruction
  - WC and LS branches take fundamentally different approach
  - Some comparisons between later stage products possible.
  - WC 3D $\rightarrow$ LS 2D pollination possible (eg flash matching)
- Caveats:
  - Conceptual view, ignoring many details.
  - Light sim and processing not depicted
    - ★ Less computation than TPC
    - ★ WCP has flash matching.
  - ND proc is unknown creature (to me)

Stage 1 Data Production Issues - Data Size

DUNE FD 10kt "event" readout is **large** (rule of thumb: 2 GB / CPU core)

- 10kt-5ms: 6 GB packed, 15 GB unpacked in memory
- SNB 10kt-100s is 20,000 times larger.
- APA-5ms: 40 MB packed, 100 MB in memory, very reasonable.



Per-APA Stage 1 processing is desired and required

- DAQ will likely produce per-APA files
- No Stage 1 processing requires multi-APA info.
  - NF/SP operates per-APA basis in any case.
- SNB needs yet more fragmentation
  - Slice 100s into $\approx$ 5-10 ms per-APA chunks, must reuse $\approx$ 100 μs to avoid edge-effect
- Residual care needed:
  - ROOT overhead in *art* / LArSoft adds memory pressure, would like to avoid entirely.
  - WCT NF/SP "light weight" enough for per-APA 5-10 ms. (PDSP proc is okay)
  - WCT can operate in multi-threaded mode to better utilize memory on 3+ cores.

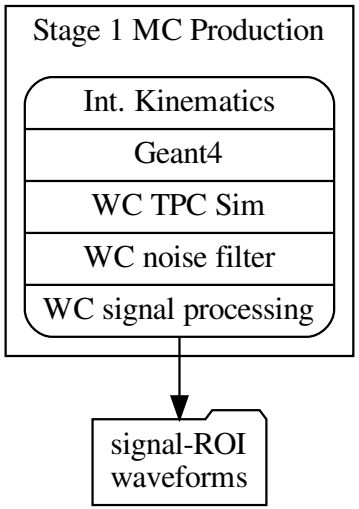
Stage 1 Data Production Issues - Multi-processing Issues

- Multi-threading CPU
  - WCT has *pipelined, multi-threaded* execution mode
  - Means multiple "events" in-flight at once
  - Higher total memory but lower per-thread usage
  - Best exploited if Stage 1 data processing runs as pure-WCT job
    - ★ N-1 threads wasted in non-WCT, ST'ed job portion
- Hybrid CPU/GPU production environment becoming **required**
  - WCT SP has new DNN ROI (Haiwang) ML inference
    - ★ $\approx$ 20s / APA / 5ms on CPU $\rightarrow$ 200ms on GPU
  - More GPU ussage to come:
    - ★ CCE-PPS (CSI+Haiwang) attacking WCT sim/SP hot spots with GPU
- GPU is "too" fast
  - Need $\mathcal{O}(100)$ CPU cores to avoid idle GPUs with SP tasks.
- WCT developing multi-job/multi-host shared use of GPU.

Stage 1 MC Production Issues

WCT relies on LArSoft for int./Geant4 interface

- Relatively low-CPU compared to NF/SP.
- Many small "energy depos" produce unwanted overhead when transferred to WCT via *art* data products.
- LS also handles light sim, **WCT has nothing here**.
- Possibly develop direct Geant4-WCT integration.
  - EDEP library good candidate
    - ★ (McGrew, CAPTAIN and DUNE ND)
  - WCT-native Geant4 not out of the question.



Scaling WCT sim to 150-APAs? Probably OK.

- PDSP 6 APA posed no substantial problem.
- ST: no known issues.
- MT: known TBB "N-fanout" limitation, work-arounds exist.
- Easy feature: avoid producing APAs with only noise+^{39}Ar
- Still, may expect scaling issues, we just need to try it!

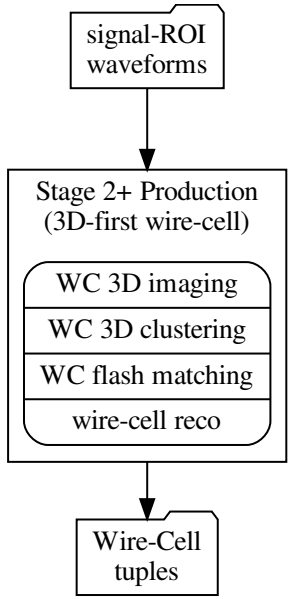
Stage 2+ 3D-first reconstruction - Prototype vs Toolkit

Wire-Cell **Prototype** (WCP)

- Focus on functionality, features, fast development.
- WCP code is **working prototype**
  - All late-stage Wire-Cell®™© reco for MicroBooNE.

Wire-Cell **Toolkit** (WCT)

- Focus on functionality, features, configurability, performance, generality, maintenance
- Proven ideas in WCP are *ported* to WCT
  - Typically completely rewritten based on understanding the algorithms
  - Sometimes, better algorithms used instead
  - Coded to fit WCT patterns, standards, conventions, etc.
  - **Very** labor intensive, but result is high quality



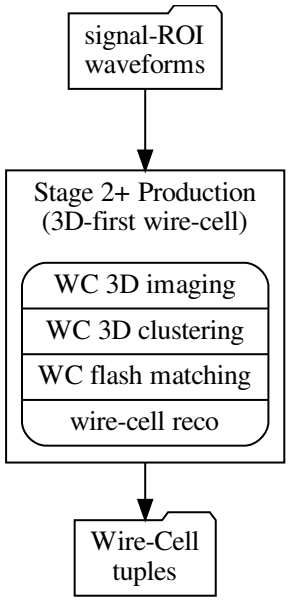
Stage 2+ 3D-first reconstruction - MicroBooNE → DUNE

MicroBooNE uses WCT + WCP

- WCT NF/SP via *art* module
- WCP hooked in via *art* modules + exchange files.

DUNE **needs** WCP algorithms (in WCT!)

- Now: WCT only has initial 3D imaging stages
  - "tiling" and "charge solving"
- World-class LArTPC reco algs resulting from a huge investment by EDG uBooNE effort must not be squandered on "just" uBooNE.



Substantial effort needed to DUNE'ify and WCT'ify

- Prior wild guess is 5 FTE-year of expert physicists/developer.
  - Do actual porting, optimization, re-inventing
  - Physics validation is needed and additional.
- Now, maybe higher, more WCP development since then.
- Possible to make "short cuts" in porting, but may not save much effort in near term, likely more effort in long term

Some WCT Future Directions

DUNE offline framework task force

- Led by Paul (+A. Normal, FNAL), deliver requirements document.
- IMO: WCT could be "it" but politics will likely trump technical.
- WCT does now and will run in whatever (reasonable) framework we end up with.

WCT is developing "distributed I/O"

- Connect multiple sources/sinks of data to multiple algorithms on multiple threads.
- Distribute "live" data between processes / hosts.
- Target big DUNE's problems:
  - "Large event" problem: spread data across many threads, reduce memory/core, keep "event coherency" avoid bookkeeping worries.
  - CPU/GPU heterogeneity: share GPU among many threads/procs/hosts.
- CPU/GPU heterogeneity being also attacked with "local" code solution
  - Haiwang+CSI / CCE-PPS using "Kokkos" library shim over h/w

My “vision” for us and DUNE production software

- Shore up the investment in WCP algorithm development for MicroBooNE by porting into WCT while optimizing, generalizing and reinventing.
- Apply WCT distributed I/O to “large event” and “CPU/GPU heterogeneity” problems.
- Continue to strengthen collaboration with CSI on GPU-accel and ML, both our current initiatives and pursue new ones.