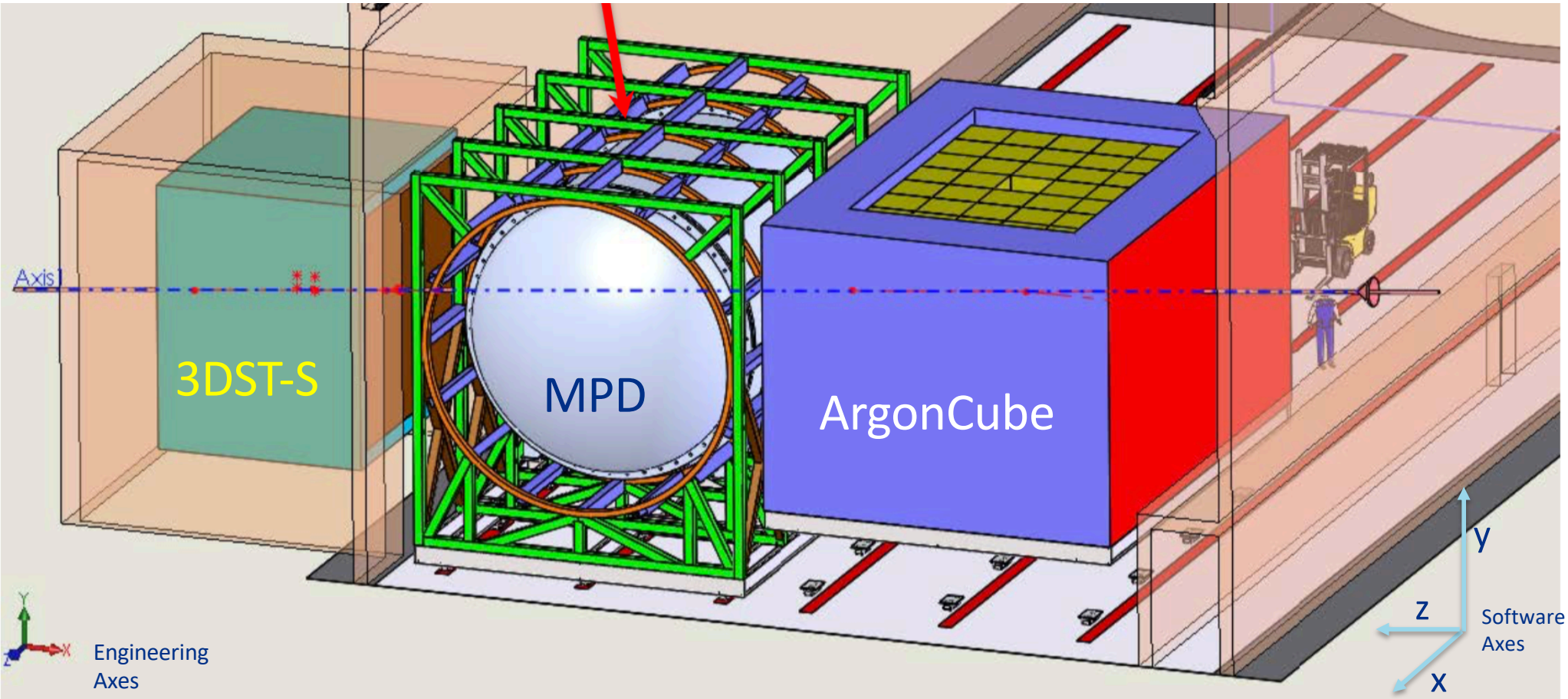
GArSoft Status – Simulation and Reconstruction Software for the DUNE MPD ND

Tom Junk, Leo Bellantoni, Eldwan Brianne, Thomas Campbell, Gavin Davies

DUNE ND Software Integration Workshop

July 24, 2019

The DUNE Near Detector Complex



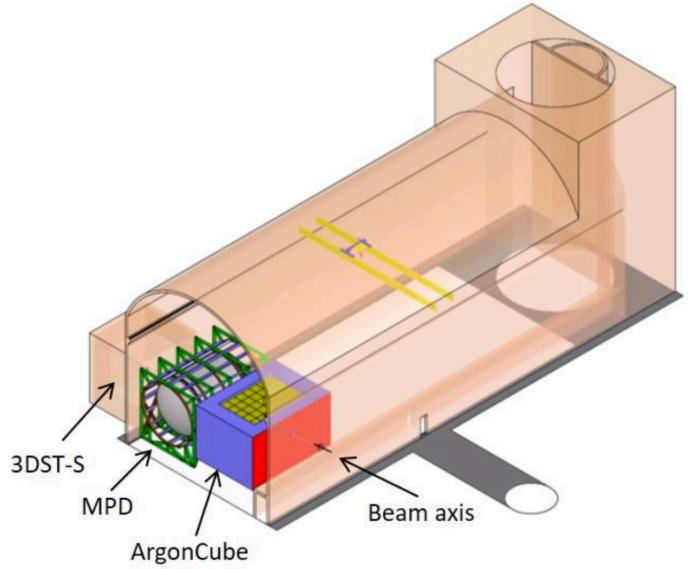
ArgonCube: Pixel-based LArTPC, unmagnetized (150 Tons)

MPD: "Multi-Purpose Detector": High-Pressure Gas TPC, solenoid, ECAL, muon stack

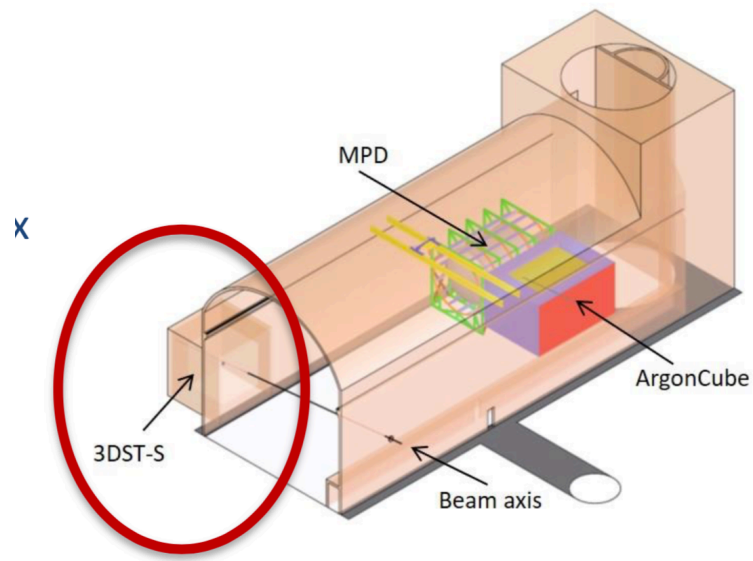
3DST-S Plastic scintillator, gas TPC, magnet, and ECAL

DUNE ND Prism Hall

MPD and ArgonCube plan to move up to 35m off axis.

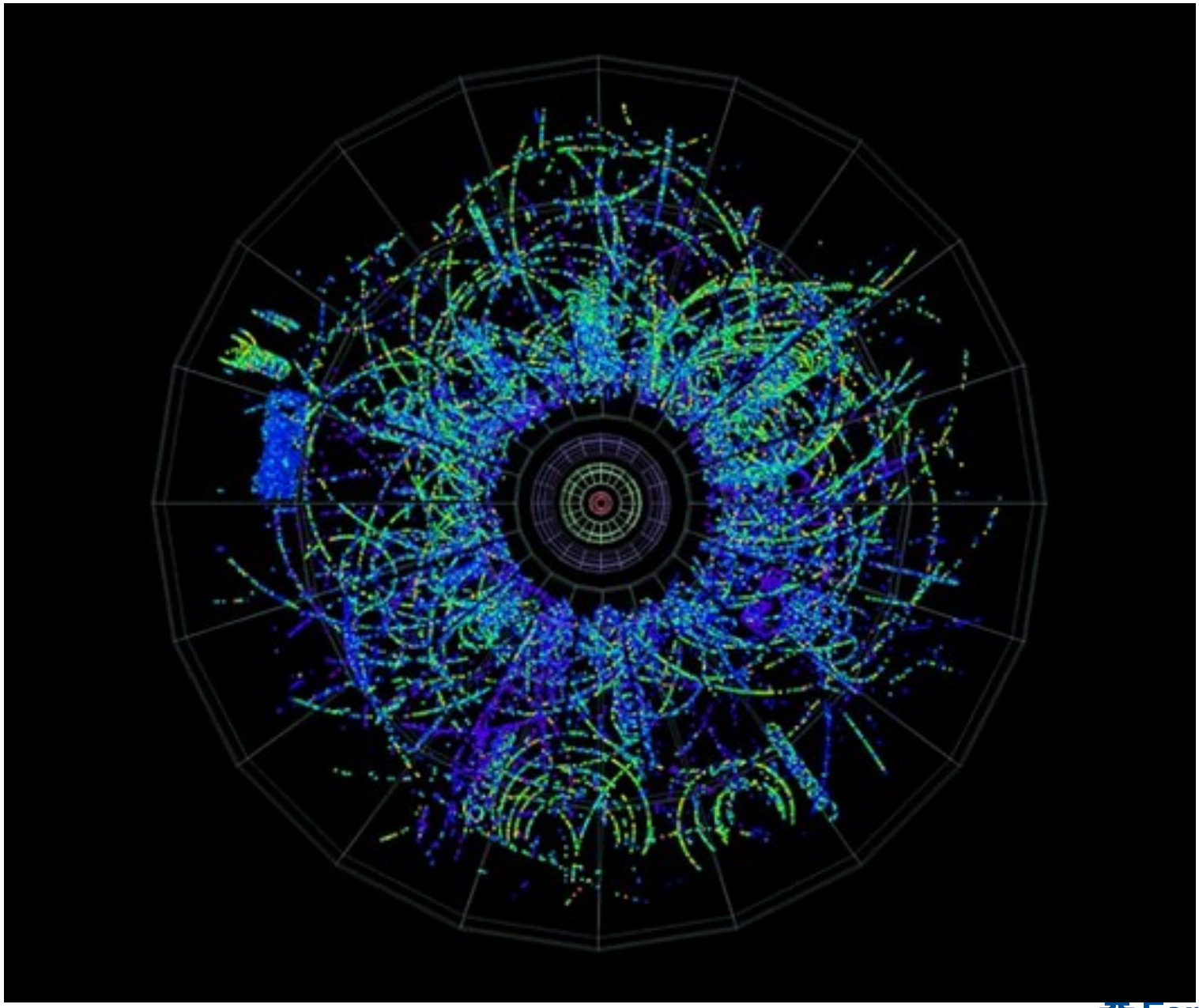


3DST-S stays on axis in an alcove



One of the first p-Pb collision events in ALICE (raw data)

The hits are all there!



We Need our Own Software

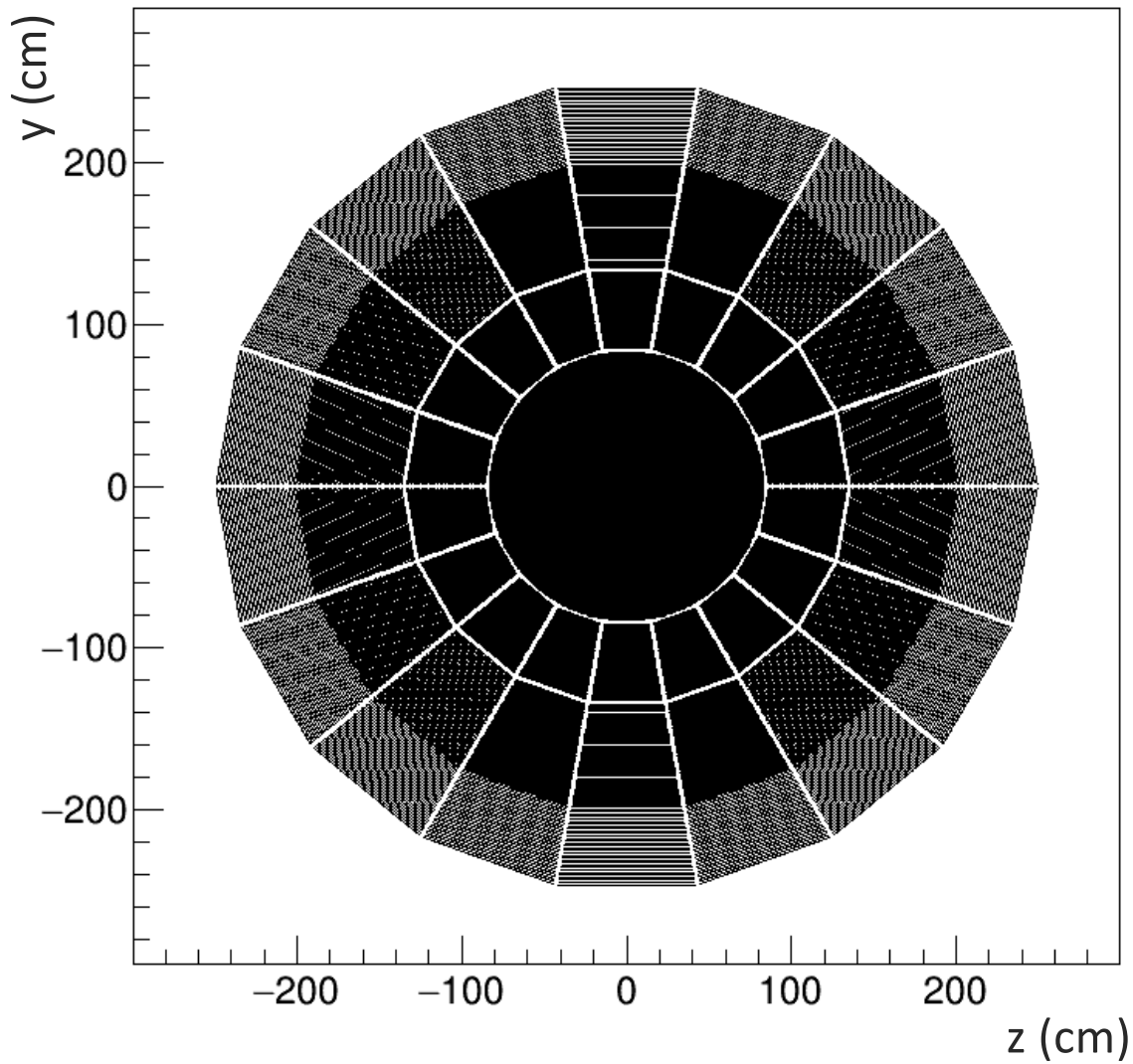
- ALICE software is great
  - GEANT4 (*and* GEANT3!)
  - Calibrated with data, used for physics
  - Includes lots of subtle effects
  - Efficient, compact
- But it doesn't do what we need
  - We need 10 bar Ar (c.f. 1 bar mostly Ne)
  - Hole in the middle needs to be filled
  - We need to reconstruct low-momentum tracks
  - We need to reconstruct tracks pointing in arbitrary directions, starting anywhere in the chamber
  - We need to integrate it with the LAr Pixel detector, the ECAL, 3DST, and muon catchers.

Ways Forwards (as of 1.3 Yr ago)

1. **Fork ALICE software and develop it as our own**
   - Great starting point
   - Terminology and techniques differ from FD (and ND) (e.g. doesn't use *art*). It's just based on ROOT
   - May find some things targeted at ALICE physics baked in very deep in the code.
   - More tricky to use different framework from the Far Detector, or the Pixel LArDetector and 3DST-S
   - I/O structure mixed in with calculation
2. **Modify LArSoft**
   - PixLAr ran into computer resource issues. LArSoft likes to save raw::RawDigits for example
   - LArSoft's G4 interface is hard to modify away from the rectangular prism (Difficult to add the CPA frames to the ProtoDUNE geometry for example as they cut small notches in the active volume)
   - Data products aren't appropriate. Tracks e.g. are helices and not just arbitrary hit collections.
   - Need a fully-simulated ECAL
3. **Build Our Own**
   - STT did this – HISOFT (T. Alion et al). Repo: dunefgt
   - Lots of work
   - Reco algs would have to be all our own anyway, even in options 1 and 2
   - Brian Rebel already went down this path: GArSoft, which is based on ***art* and nutools**

Filling the Hole

- 18 Sectors of IROC and OROC channels now using ALICE nominal geometry
- Rectangular array of pixels in a disk in the center
- Pixel size: 6mm x 6mm c.f. 4 mm x 7.5mm for inner pad rows.
- Total channels per side is now 339068.
- Total on both sides: 678136
- About 18% of channels are in the disk.



MPD: Reconstruction: GArSoft

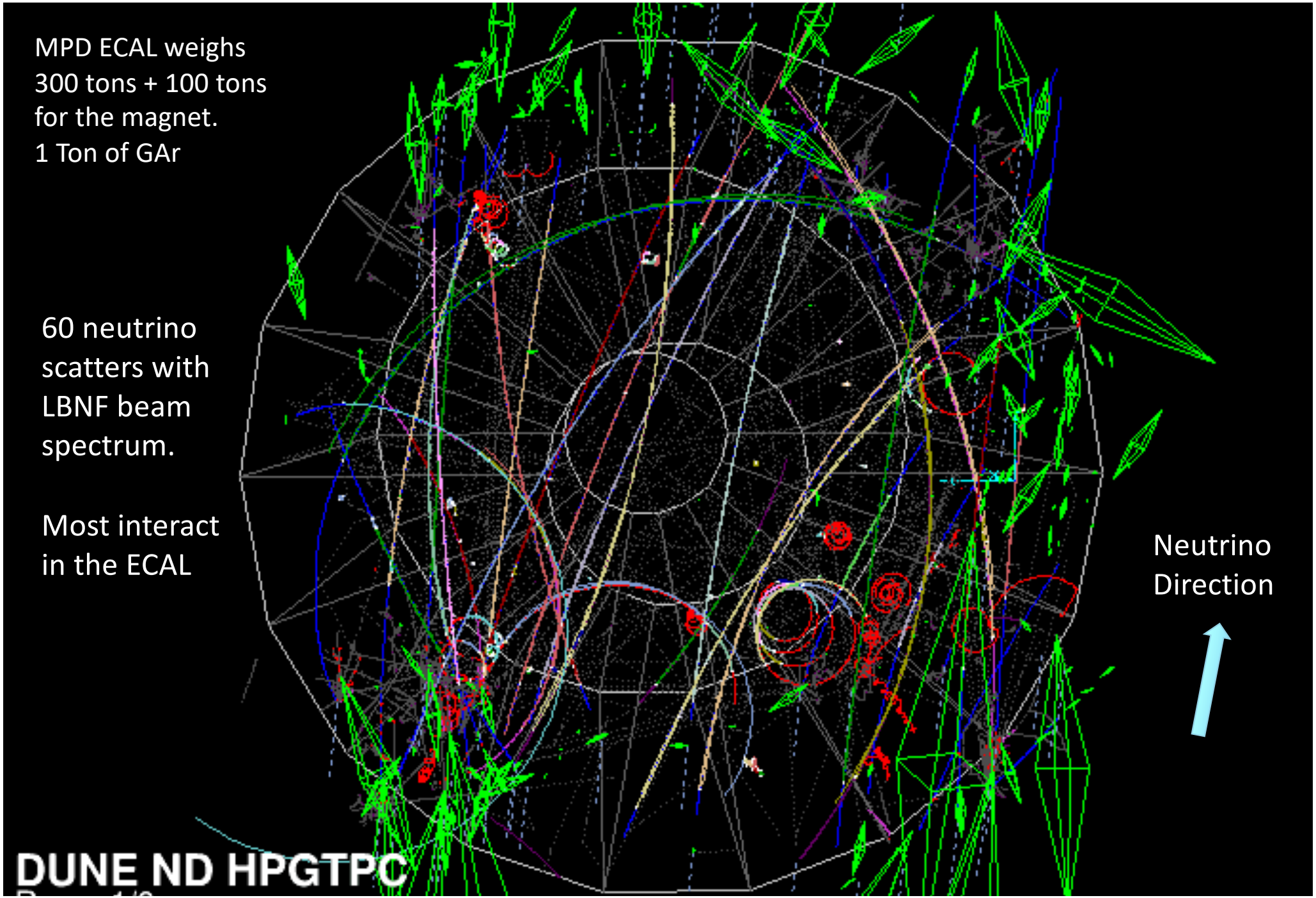
Implemented

- Event Generation
- Detector Geometry
- Particle Interactions & Energy Deposits
- Drift and Diffusion
- Digitization
- Hit finding and clustering
- Pattern recognition
- Track fitting
- ECAL Digitization
- ECAL Reconstruction
- Ionization-Based Particle ID
  - Initial version exists – needs work

To do (to some degree optimization)

- TPC Field Response and Electronics Response
- Optimize pattern recognition in difficult cases
- Optimize track fit
- Very short tracks in crowded environments will require innovative algorithms
  - Deep learning methods being studied now
- Vertexing
  - Preliminary vertex-finding algorithm written and tested
- ECAL
  - Cluster-Track matching
  - Full energy reconstruction (only visible energy for now)
  - .....

Slide from Alan Bross


MPD ECAL weighs
300 tons + 100 tons
for the magnet.
1 Ton of GAr
60 neutrino
scatters with
LBNF beam
spectrum.
Most interact
in the ECAL
Neutrino
Direction
DUNE ND HPGTPC

GArSoft Beginning to End

DUNE-Doc 13933

- Generators
  - Particle Gun
  - GENIE
  - CRY (to do: CORSIKA)
  - Radiologicals (not yet tested)
- Simulation:
  - GEANT4 (uniform B assumed at the moment)
  - GArG4 module patterned on LArG4
  - Outputs energy deposits (recently added to LArSoft)
  - Also outputs channel waveforms: gar::raw::RawDigits, which are zero-suppressed. No noise. Small event sizes.

GArSoft TPC Simulation

- Drift model is a copy of LArSoft's
  - Nearest channel geometry – charge diverts over cracks and inward from edges. To do – deaden the cracks and edges to simulate cover electrodes.
  - No Lorentz angle or space charge yet. Uniform E and B
  - Longitudinal and Transverse Diffusion simulated numerically by sampling from Gaussian distributions
  - Drift velocity and diffusion input parameters from Magboltz
  - No induction field response on the pads – charge "collects"
  - This may overestimate response to tracks that point straight at pads
  - Less of an issue than with wires since tracks have to point along E rather than being in the plane containing E and a wire (a la MicroBooNE or DUNE FD)
  - Still a concern as low-energy electrons spiral along B (and E)

GArSoft Track Reco First Steps

- Hit finding
  - gar::rec::Hit
  - hits belong on one and only one pad. Necessary for BackTracker to work, as it makes this assumption (carried over from LArSoft)
  - zero-suppression threshold is a "hit finder".
  - Hit finder module refines hits – if a waveform drops below 20% of its peak it'll start a new hit.

- Hits → TPC Clusters (also done by ALICE)
  - gar::rec::TPCCluster
  - Nearby hits grouped together
  - Charge-weighted centroid and RMS calculated

Vector Hit Finding ("tracklets")

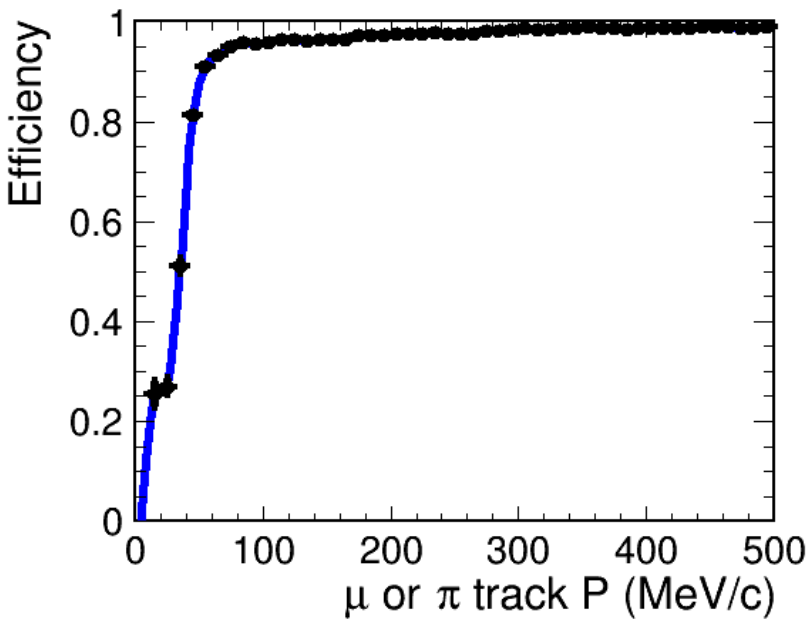
- gar::rec::VecHit
- Line segments fit in 3D
- Maximum length 20 cm (adjustable)
- TPCClusters added to line segment candidates if they are close to them
- Line segments started by close pairs of TPCClusters
- Two passes performed – remove highest chisquare hits and attempt to reassign them to other Vector Hits
- Vector hit contamination near the primary vertex is an issue ("hit stealing")
- Tendency to follow pad-row geometry in places (to fix)

Pattern Recognition & Fitting

- Pattern Recognition: grouping vector Hits into track candidates
  - Output is gar::rec::Track data products and associations with TPCClusters and VecHits
  - Matching in 2D (circles) and 3D (constant dip angle) of nearby vector hits
  - Currently has loose cuts so it's highly efficient, but it does break electron curlers up (cuts too tight), and it stitches together two legs of a conversion or a V
- Track fit: Kalman Filter
  - Output is a second set of gar::rec::Track data products and associations with TPCClusters
  - Fit is performed twice, once from either end of the track (vertex is not yet known)
  - Energy Loss and Scattering → track parameters are different on the two ends
- Tracking takes less than 5 secs of CPU/event

Tracking Performance: $\pi^{\pm}$ and $\mu^{\pm}$

Estimated using Leo B's sample of ν_{μ} events with the optimized LBNF FHC spectrum



Charged pion and muon tracking efficiency

Electrons are similar, but including them produces a kink at 20 MeV (bigger than the one that's there).

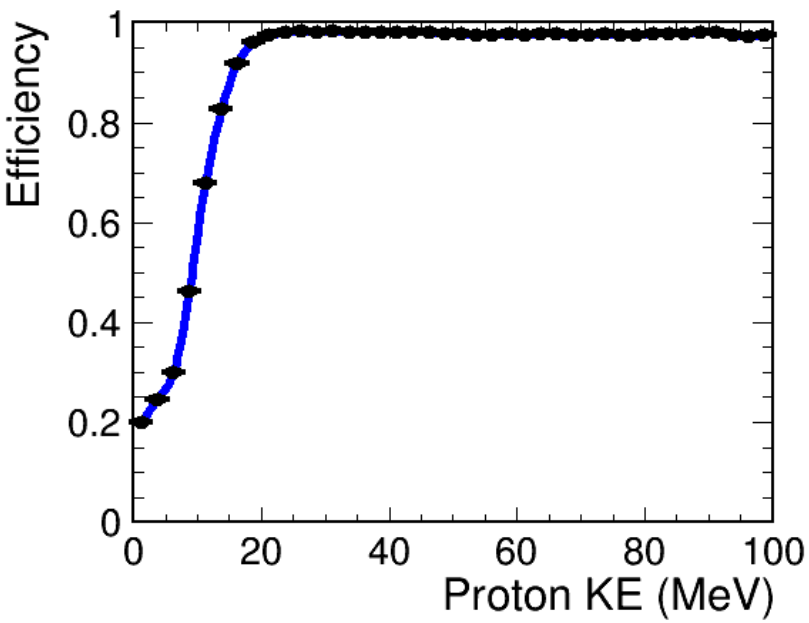
Low-energy electrons curl around – only partial efficiency for them

Low-energy pions and muons stop – have a track length cut of 20 TPC Clusters

Protons with P<150 MeV have very little KE and thus stop quickly – plot their efficiency vs. KE

Tracking Performance: Protons

Estimated using Leo B's sample of ν_μ events with the optimized LBNF FHC spectrum



Very short track efficiency overestimated near a dense primary vertex due to combinatorics – fake matches.

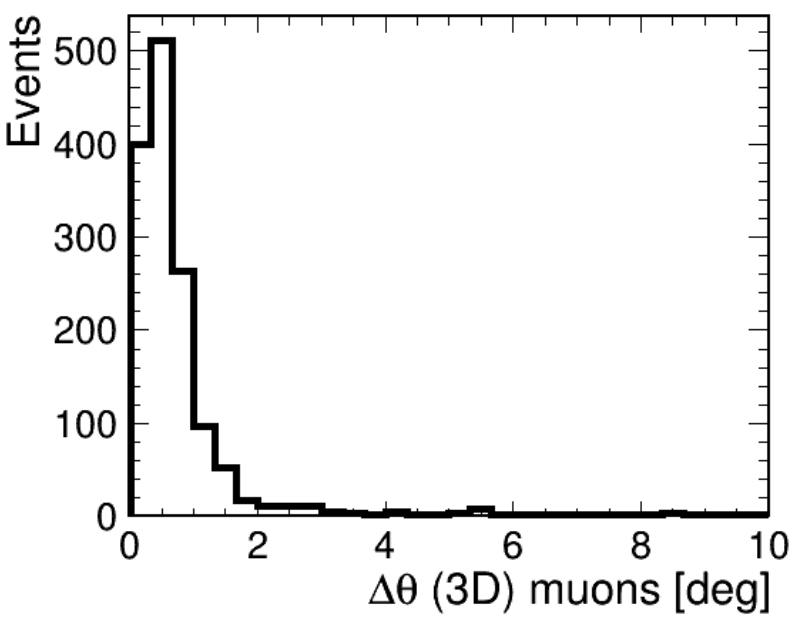
Efficiency should go to zero at KE=0.

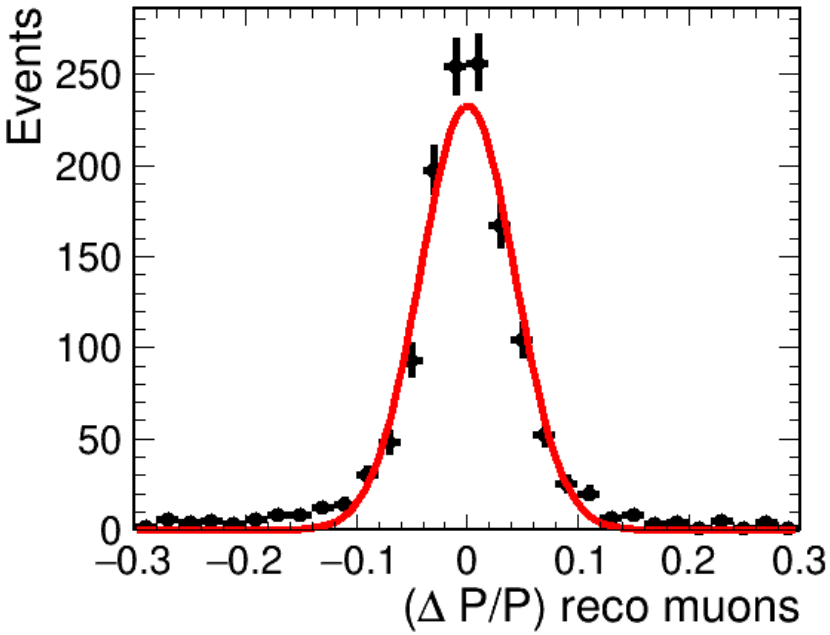
Work in Progress – Optimizations will improve this

Fermilab

Tracking Performance: Muon Angles and Momenta

Work in Progress – Optimizations will improve these

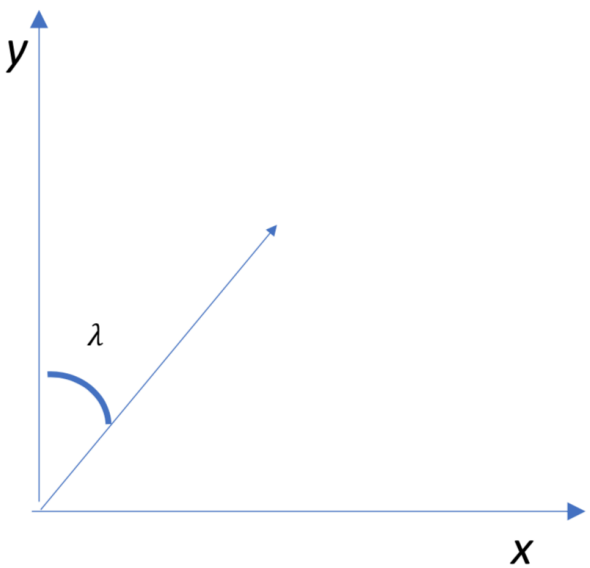


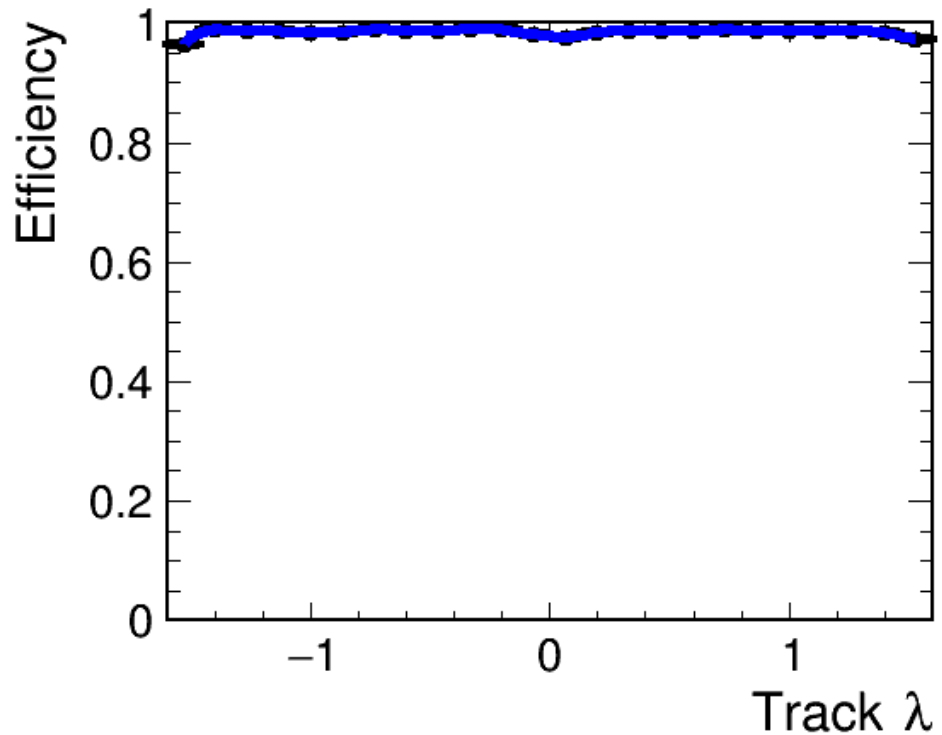


~1 Degree angular resolution, and ~4.2% momentum resolution
At 0.4 T B field. Should be 3.6% resolution at 0.5 T.

Fermilab

Tracking Performance: 4π Coverage



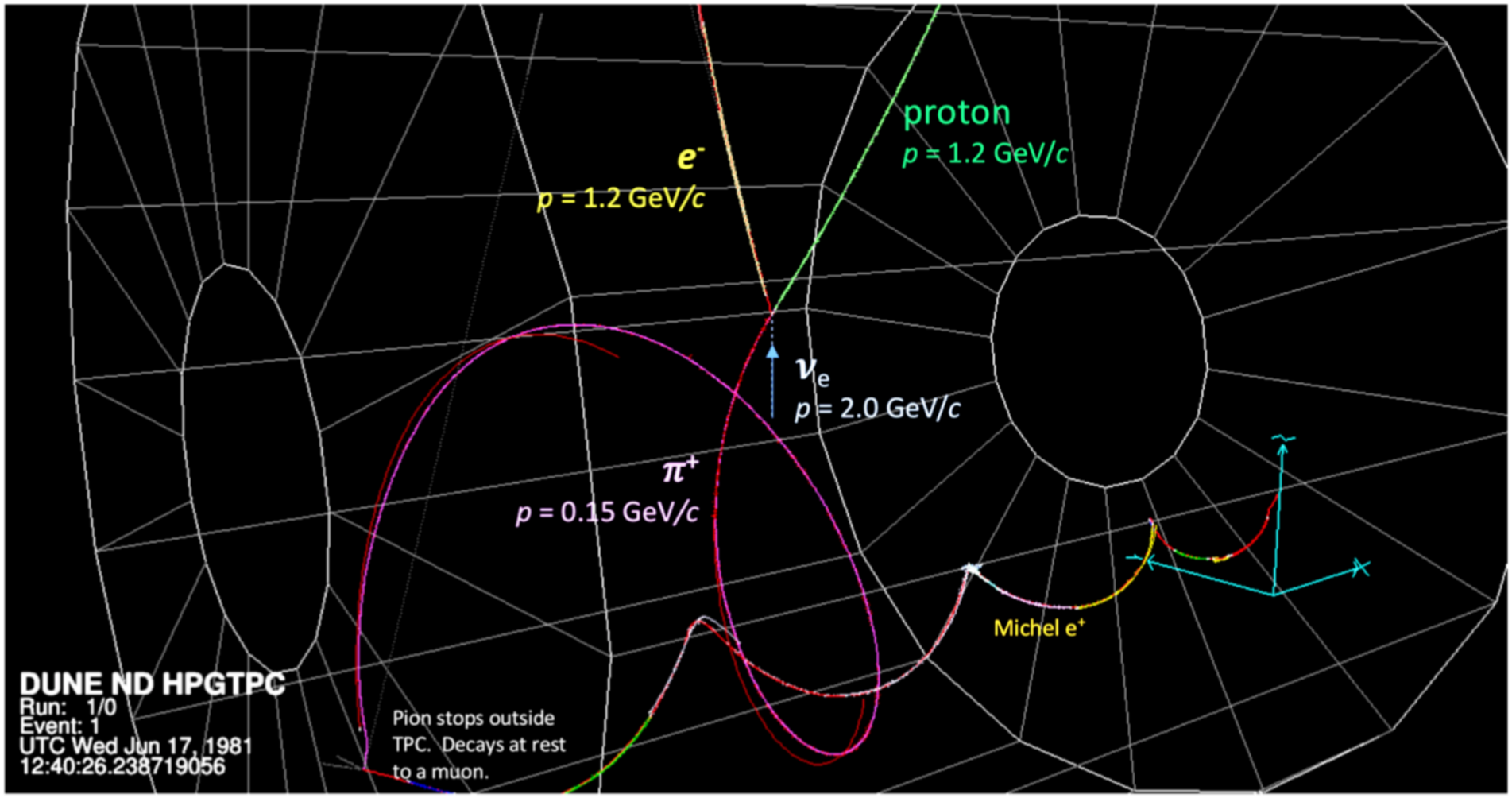


All tracks with momentum > 200 MeV/c (protons are inefficient for momenta below 150 MeV/c)

n.b. Charge modeling on the pads is naive – induced signals will be less for trains of charge arriving on the same pad over lengths of time

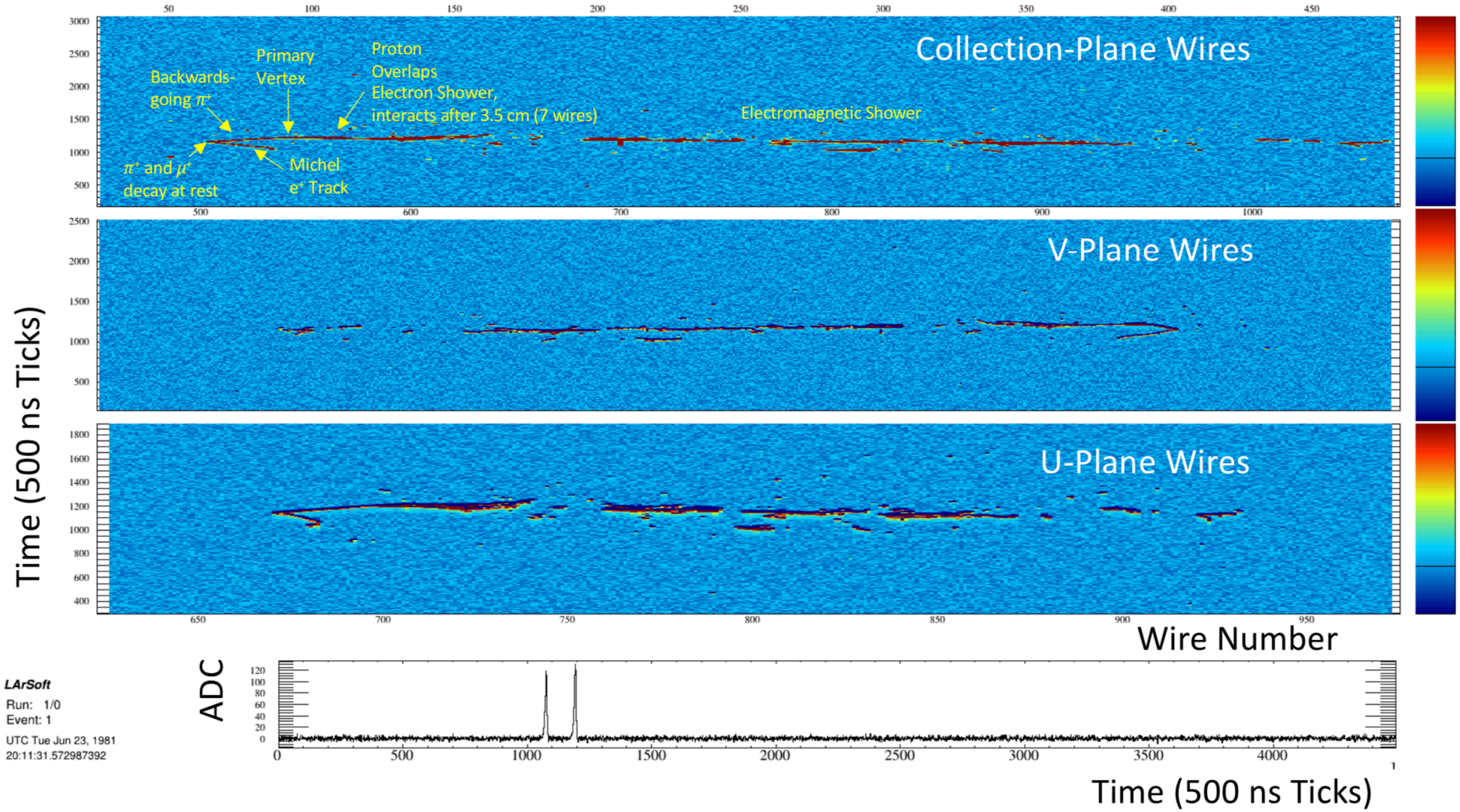
A Simulated and Reconstructed ν_e Charged Current Event in the HPGTPC

ν_e + Ar → e- + π+ + p + n



Neutron with p = 0.23 GeV/c at the P.V. not shown

The Same Event in LAr

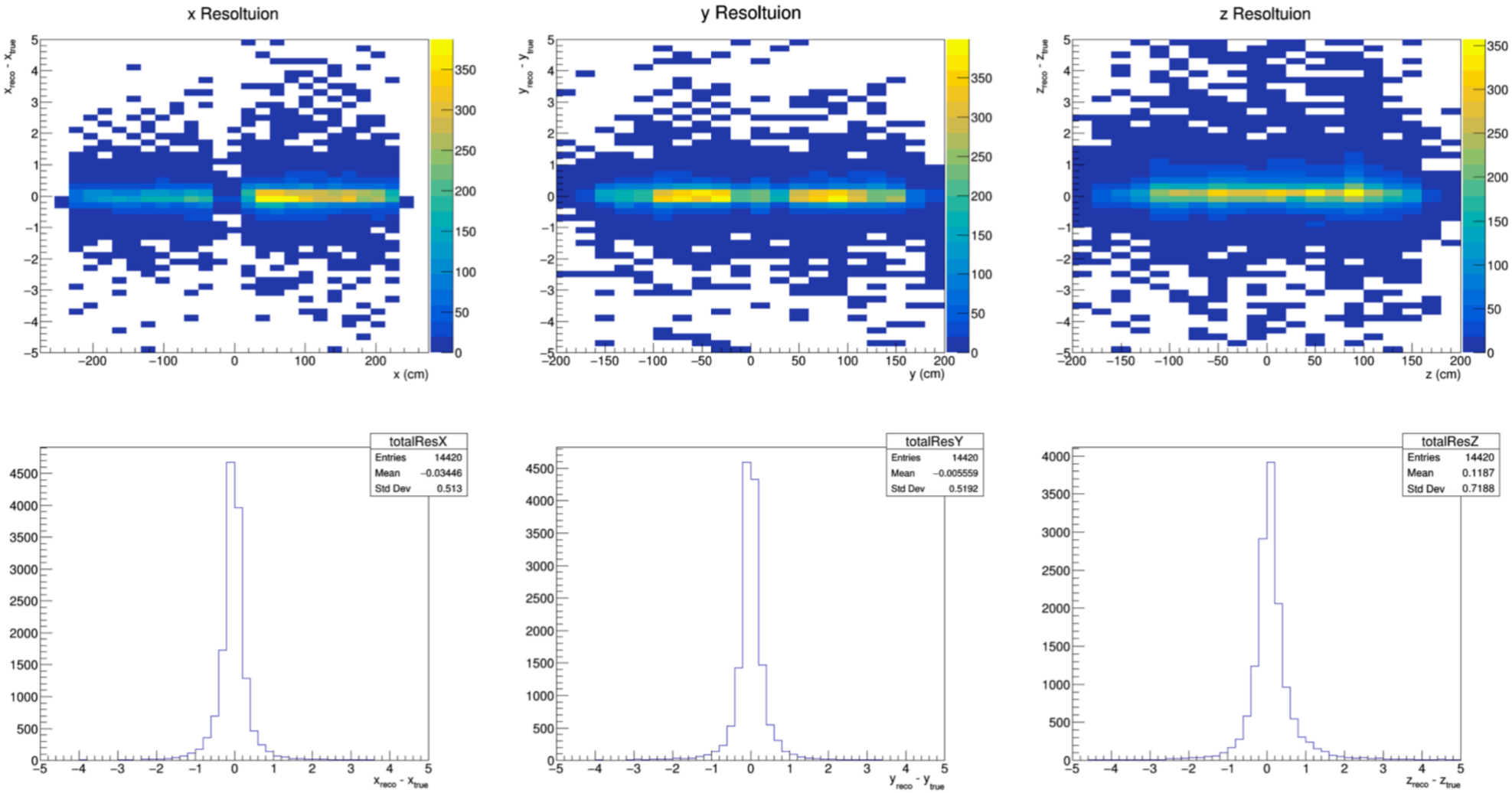


CVN selects it as a ν_e CC. Nick and Tingjun's Ereco = 1.914 GeV (2.0 was true)

Reconstructing Vertices

- Initial vertex reco algorithm requires at least two tracks with nearby endpoints. Extrapolated helices used to fit vertex
- Some neutral-current events have no tracks at all
- Some charged-current events have just one track that starts in the middle of the detector
  - We can declare bare track endpoints as vertices but do not do so at the moment
  - Neutrals entering from outside the TPC can scatter and make fake vertices
  - Photon conversions make their own vertices
- Scattering of throughgoing tracks makes fake vertices with 2 tracks

Vertex Resolution



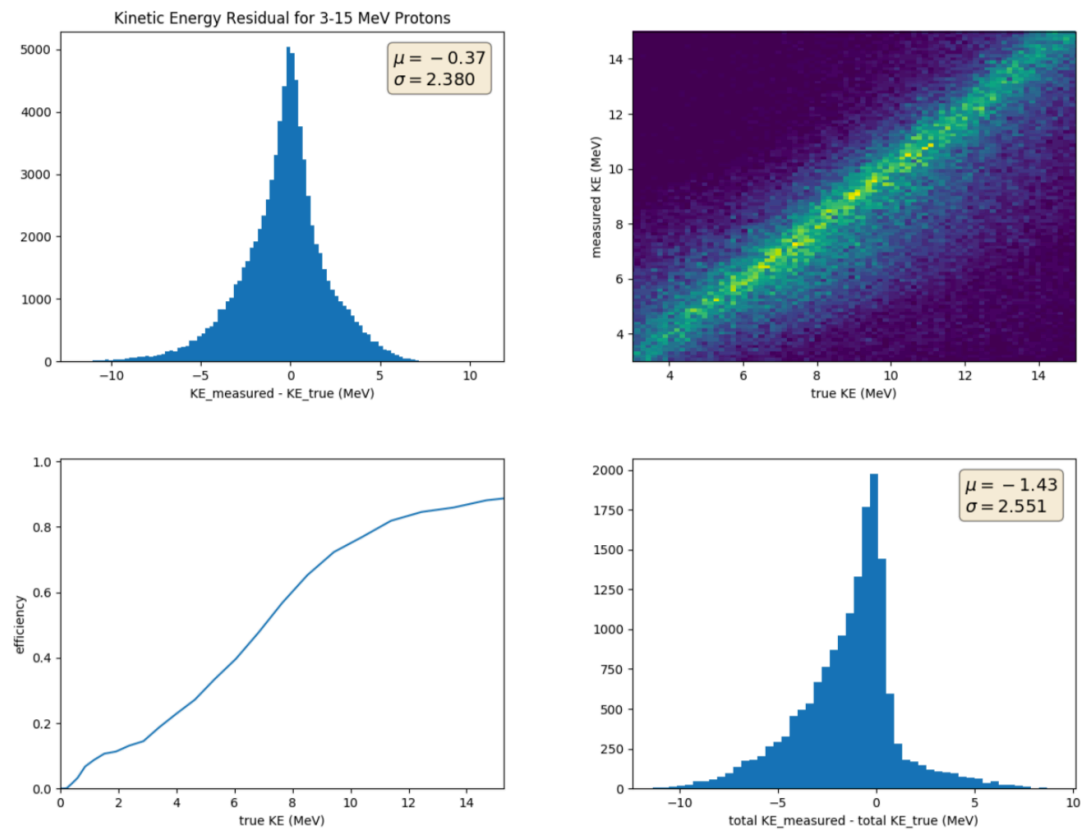
Thomas Campbell

ν_μCC Events with beam pointing along Z

Finding Short Tracks Near the Primary Vertex

- Thomas Campbell's work
- RANSAC line finding + Neural Network for p/pi separation and energy estimation
- Energy is dominantly from range for short tracks. Longer tracks use curvature. We need an algorithm that uses both optimally

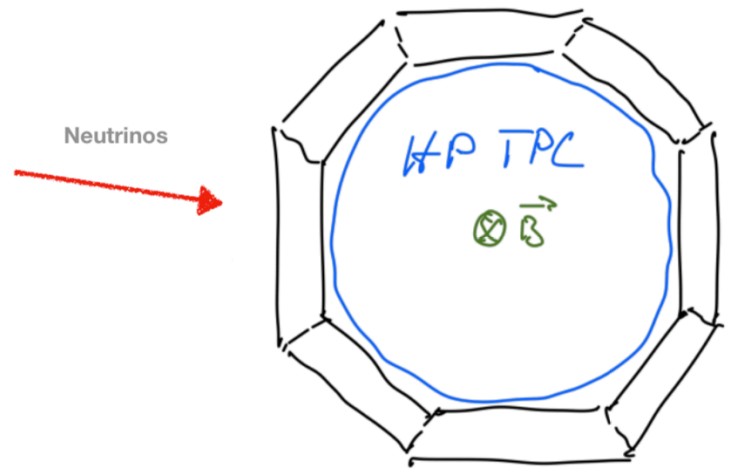
Results

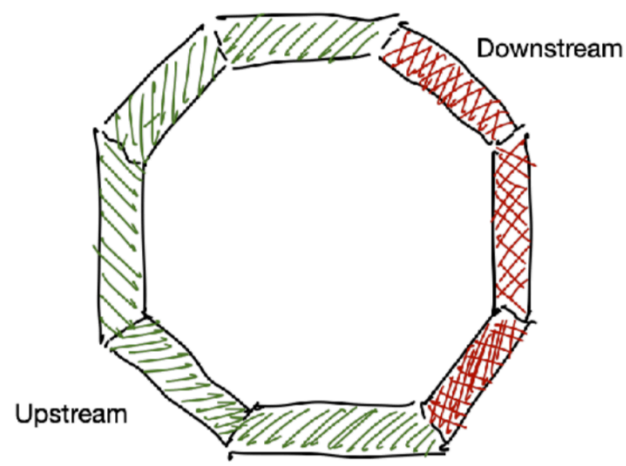


The ECAL concept.

Possible geometry design

- Global layout driven by the HPgTPC size
  - Radius ~2.7 m, Length ~5.5 m
- Need to fit the cylindrical geometry of the HP vessel and planar geometry of a granular calorimeter ➠ octagonal geometry
- Overall dimensions: Octogon side length ~ 2-3 m
  - Barrel surface ~ 150 m^2
  - Endcap surface ~ 70 m^2
- Due to the nature of the experiment ➠ granularity/depth/resolution does not need to be the same everywhere
  - Two regions: downstream and upstream
- Possible variable longitudinal segmentation
  - Thin layers in front (good EM res), thick layer in the back for containment



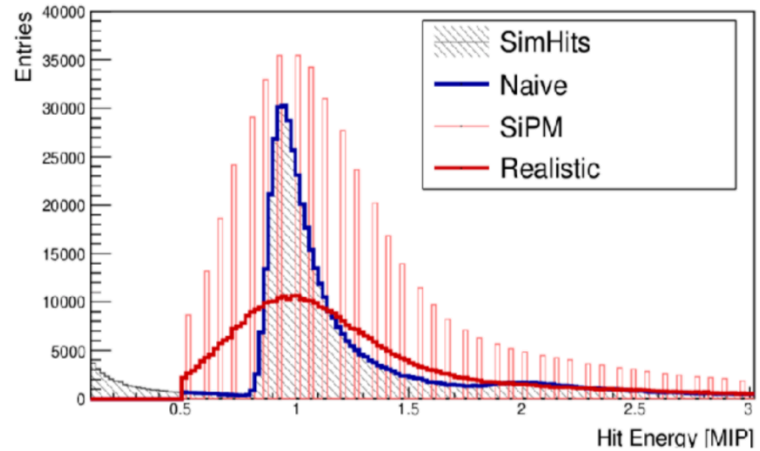


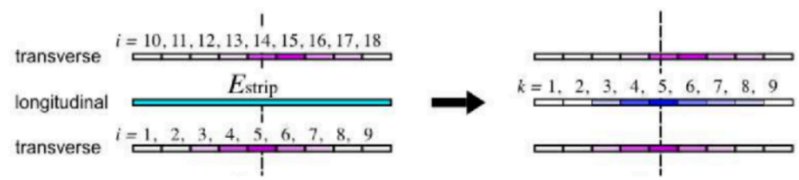
Slide from Eldwan Brianne, TPC Mini-Workshop, 12 July 2019
https://indico.cern.ch/event/827540/

Software: ECAL Reconstruction.

Work in progress

- Developed in parallel to the HPgTPC reconstruction (see Tom's talk) and integrated into GArSoft
- SiPM digitisation ➠ Saturation + SiPM binomial smearing as used in CALICE
- Hit reconstruction
  - Tiles straight forward (threshold check - still need proper electronic response)
  - Cross-strips
    - ➠ strip-splitting algorithm (1405.4456v2) in development
    - ➠ use timing to reconstruct hit along the strip (250 ps ~ 7 cm)
- Clustering algorithm
  - Simple NN algorithm
- Integration issues
  - Unified software framework for the ND complex ➠ task force starting



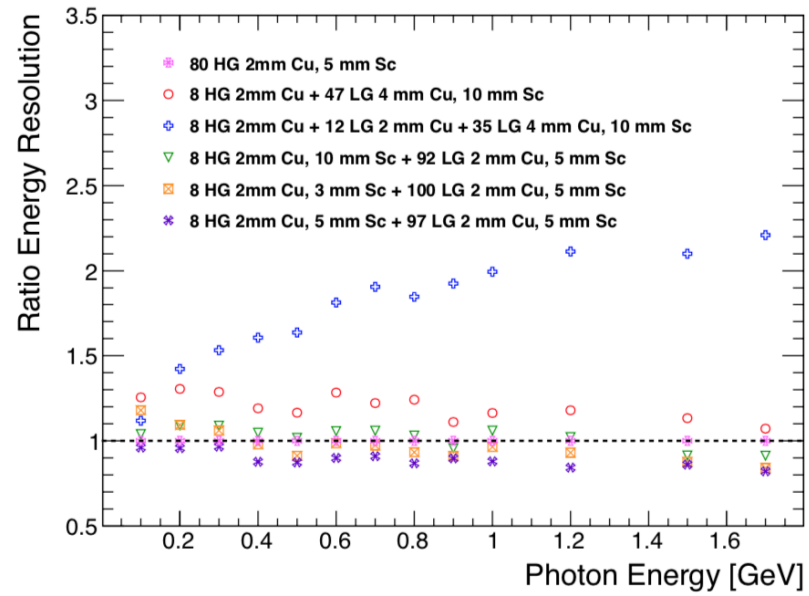


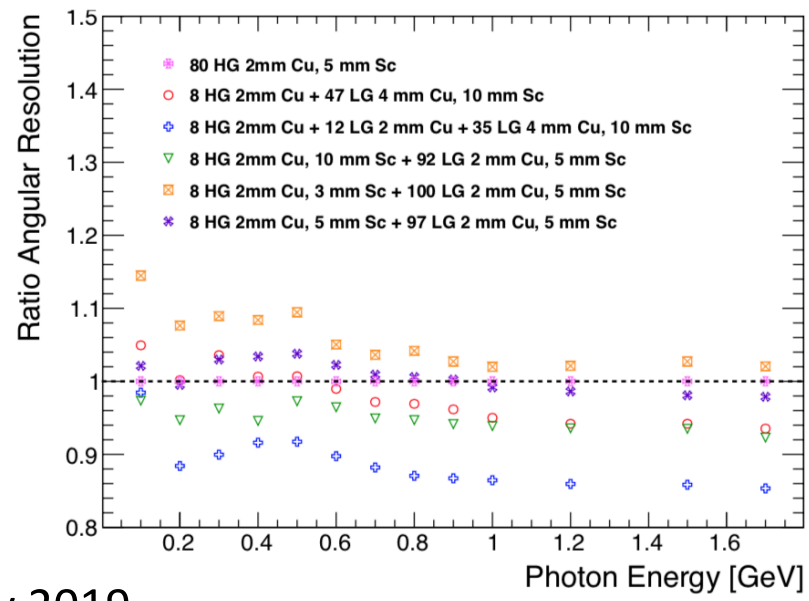
Slide from Eldwan Brianne, TPC Mini-Workshop, 12 July 2019
https://indico.cern.ch/event/827540/

Simulation studies.

Full comparison

- A lot of models have been compared! (~12 models)
- **To take away**
  - Angular resolution dominated by front layers → granularity in the back layers does not matter much ➠ *strips can be used*
  - Thinner absorber with *small Molière radius* in the front is preferred for angular resolution
  - *Shower containment* is important for high energies ➠ more layers or thicker absorber in the back
  - Thicker scintillator in the front helps in the angular resolution
  - *Pressure vessel thickness* needs to be kept below 1 X_0 to keep energy resolution below $6\%/\sqrt{E}$ ➠ ECAL can be put outside the PV





Slide from Eldwan Brianne, TPC Mini-Workshop, 12 July 2019
https://indico.cern.ch/event/827540/

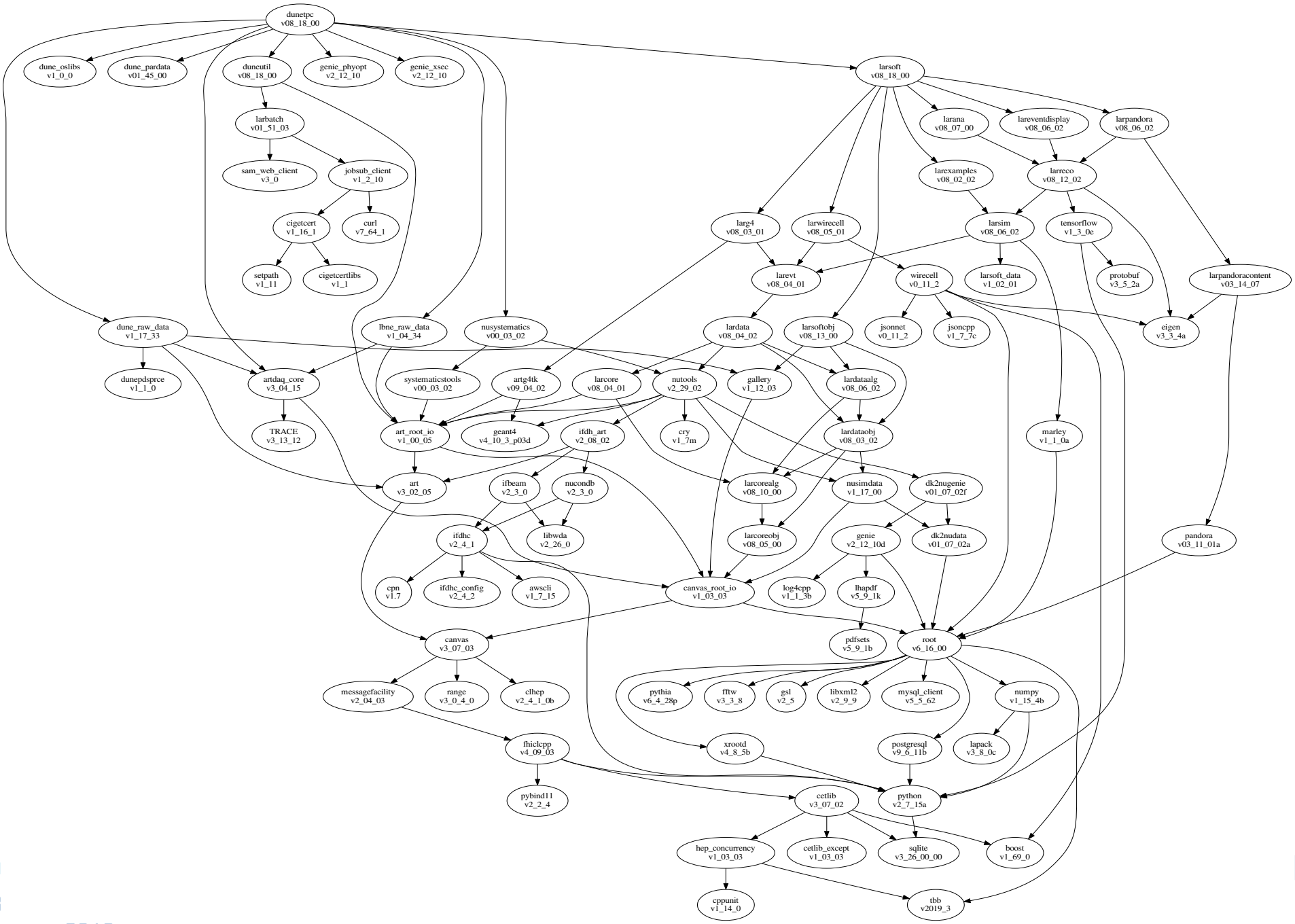
Software: To do list.

Open-ended list

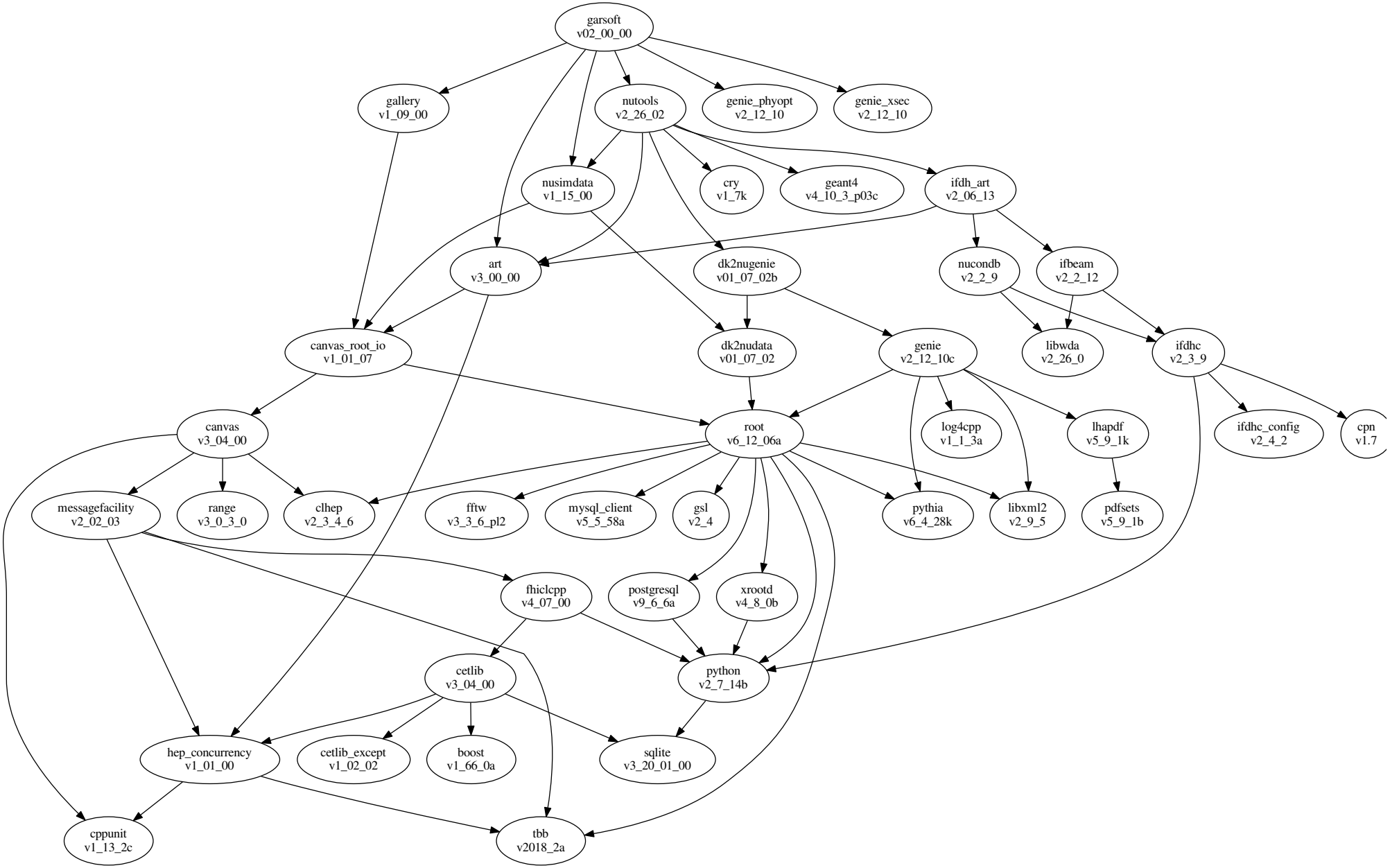
- ECAL specific
  - More realistic electronic response
  - Improvement of clustering and possible Pandora integration
  - SSA to be finalised
  - Association TPC tracks to ECAL clusters
  - Implement particle identification / π^0 reconstruction / neutron reconstruction techniques
  - Full energy reconstruction (only visible energy so far)
- General
  - Full event reconstruction
- **Ideas and help are welcomed!**

Slide from Eldwan Brianne, TPC Mini-Workshop, 12 July 2019
https://indico.cern.ch/event/827540/

The dunetpc Dependency Tree (v08_18_00)



The GArSoft Dependency Tree (depends on art, nutools)



Integration: Easy Issues First

Running detector-specific simulation and reconstruction are all independent pieces – modules work on independent data.

- channel response
- data output from sim job and readin in reco job
- noise filtering
- deconvolution
- TPC clusters and hit-finding
- tracking
- shower reco
- calorimetry

- Some modules and services may duplicate names with those in LArSoft. Can fix those easily.

Integration: Harder Issues

- Unified GEANT4 simulation
  - Current modules: LArG4 and GArG4. Consume MCTruth data products, make sim::SimChannel and energy deposits
  - particles produced in LAr -> GAr -- one can imagine running LArG4 first and then piping particles that come out of the LAr as MCTruth for GArG4, which gets run second.
  - Particles produced in GAr traveling back into LAr. Our CDR-Lite Executive summary mentions that backwards-going cosmic rays are an important calibration source for the LAr
    - Either need to iterate this, or run a unified GEANT4 step
- Unifying the GEANT4 step means having a single geometry description GDML file (or files), and calling GEANT4 once to follow particles back and forth.
- Energy deposit data products look like the new LArG4's way of doing things.
- Data products have different names but that's okay

Integration: Event Display

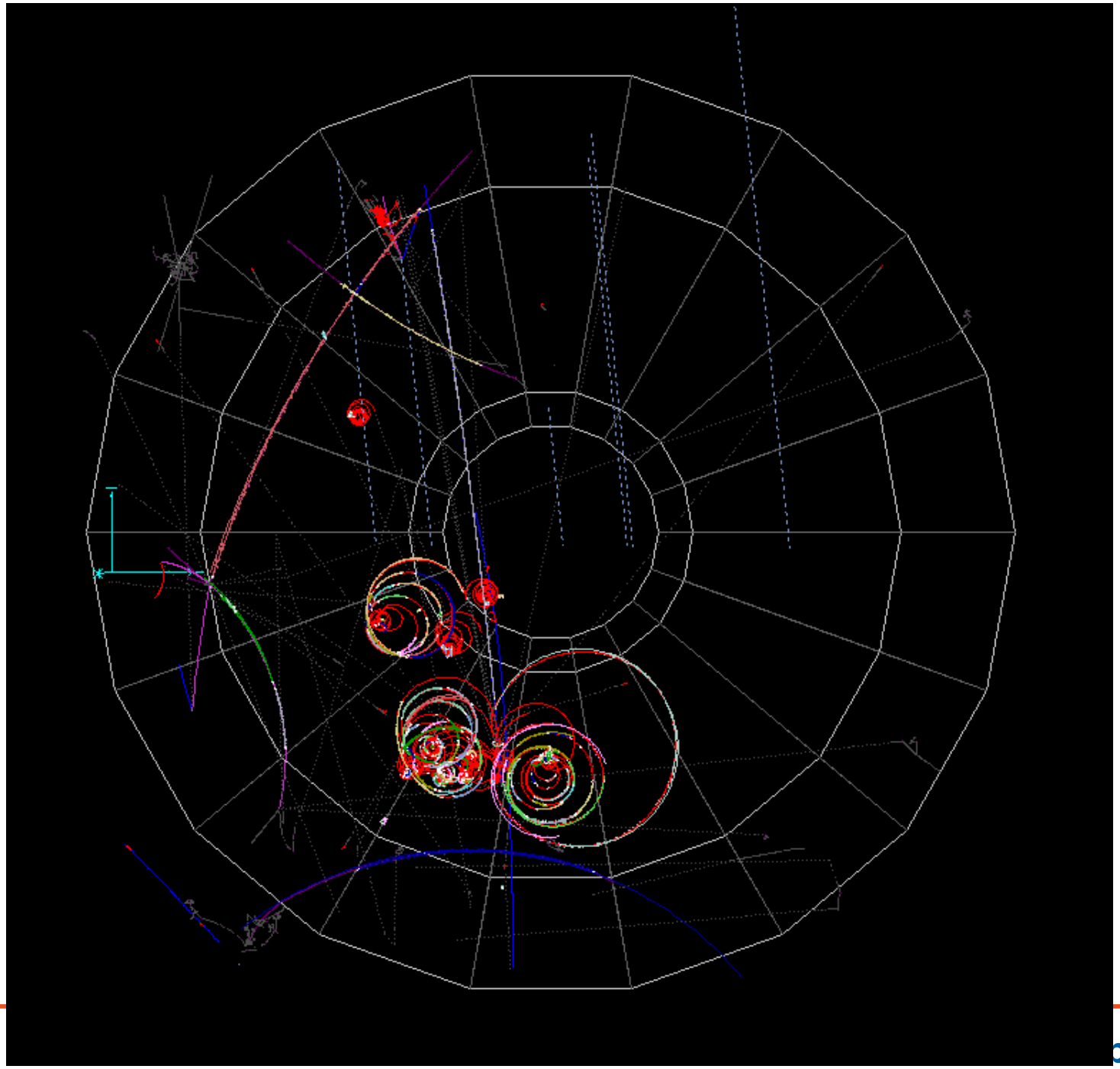
- The three-detector ND Complex will have particles exiting one detector and possibly going into the other two.
- Visualizing the events will be useful in developing (traditional) reconstruction and track-matching algorithms
- Currently we are working independently
- MINERvA/MINOS solution involves exchanging data files as far as I know – not nearly as well integrated.

Summary

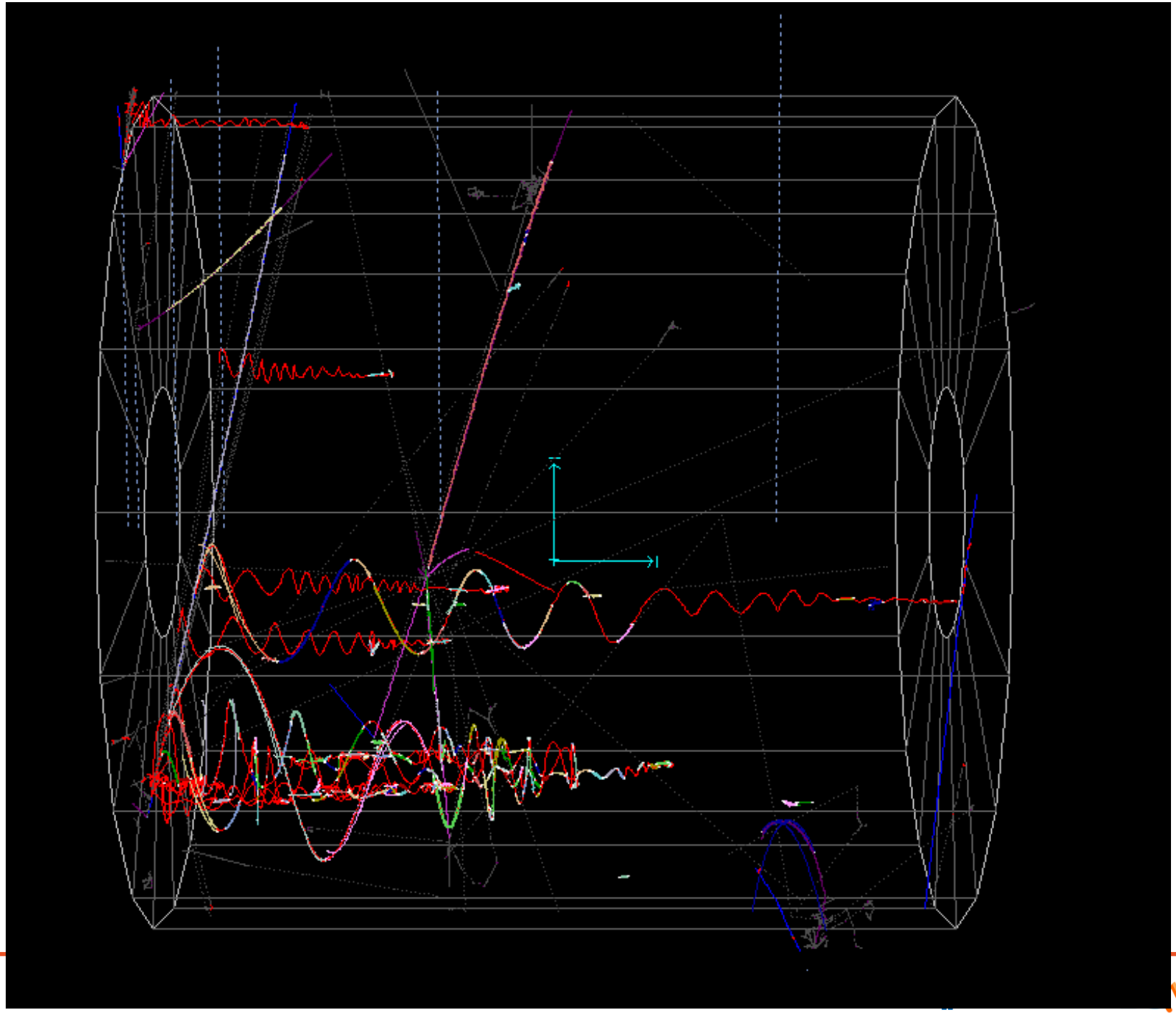
- Much progress has been made
- Lots of to-do items
  - First priority is to make the pad response function realistic.
    - Many of the algorithms and tuning parameters' optimal values depend on efficiency and resolution of underlying pulses.
    - Noise is not in the simulation yet.
  - Hit clustering, pattern recognition, track fitting, vertex finding and fitting and track extrapolation need optimization. We have initial tries at all of these but performance needs to be improved.
  - Finding short tracks in crowded environments is a challenging problem.
  - To do: Use in analyses
    - Demonstrate we can select muon and electron neutrino CC events with low backgrounds and high efficiency
    - Count and measure low-energy tracks at the primary vertex
  - Integration with ArgonCube and 3DST-S is a priority, though it is easier to make progress in a factorized environment.

A 10-interaction event with conversions end view

No ecal in this display



A 10-interaction event with conversions side view



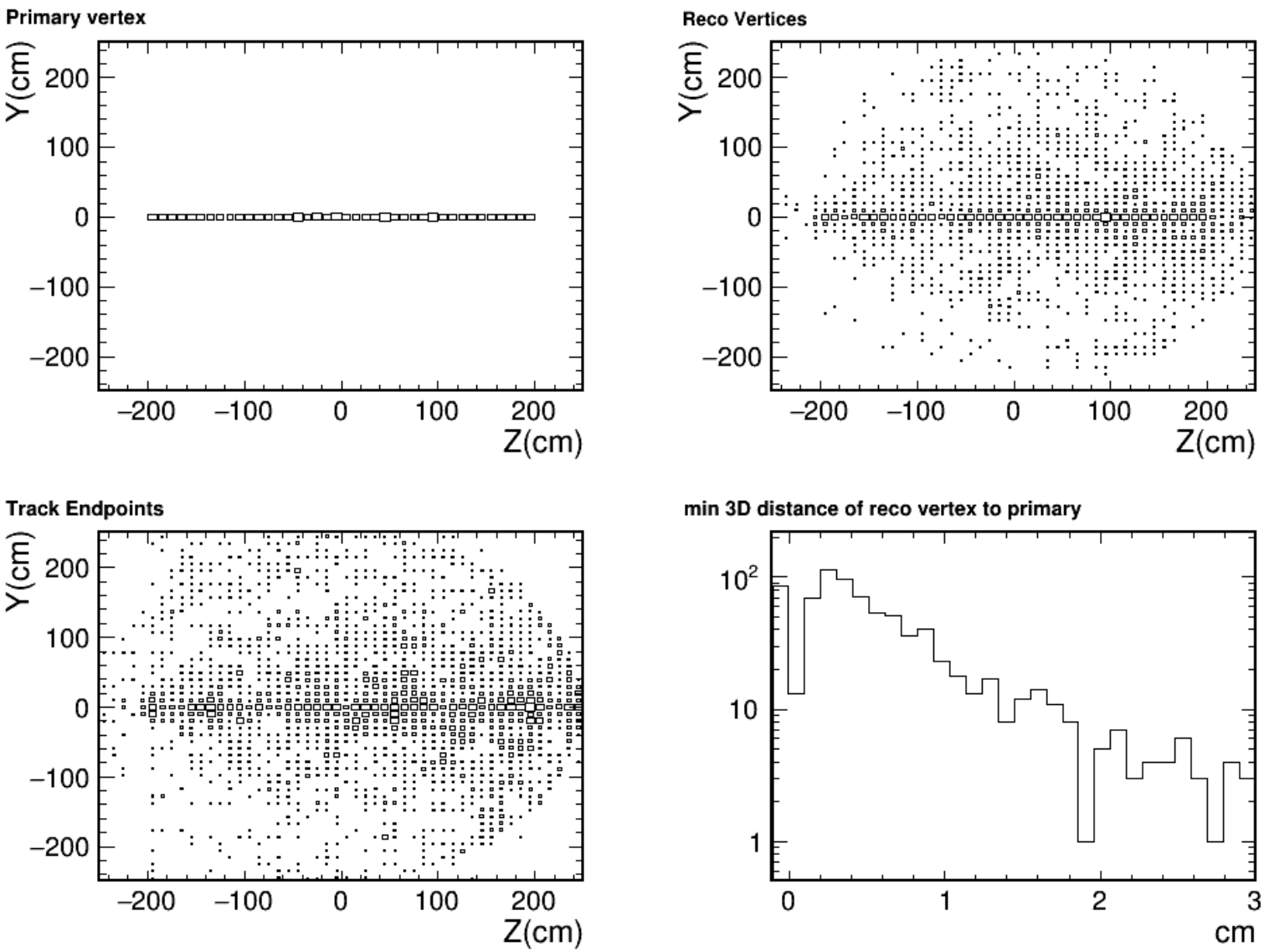
Some current timing numbers

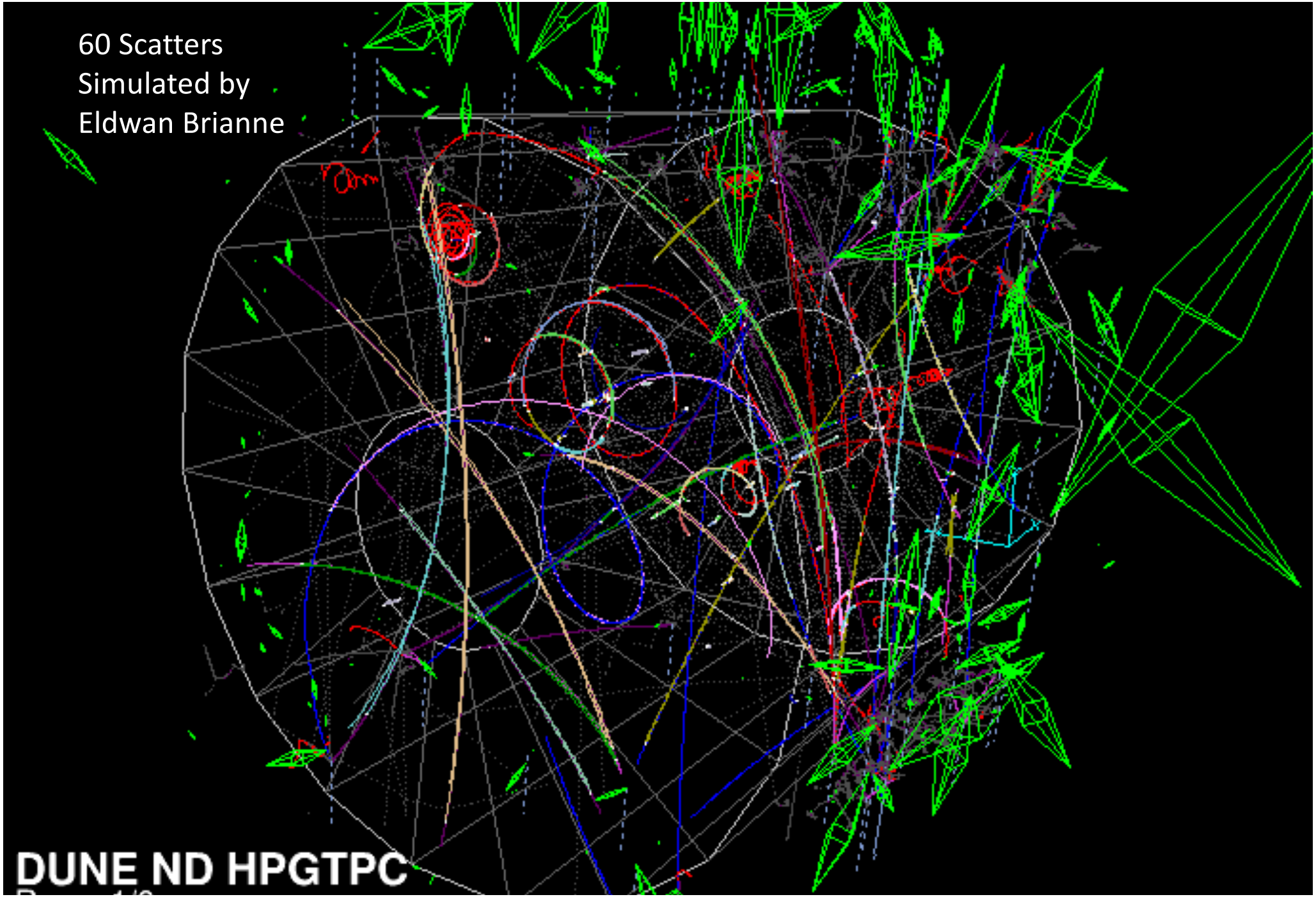
- 60-interaction events (300 averaged) in ECAL+TPC
- 3.2 GHz Sandy Bridge desktop with local disk
- GENIE+G4: 14 secs/event
- Detector Simulation: 14 sec/event 8 TPC, 6 ECAL
- Reco: 7 sec/event

- output size: 2.6 MB/event (ROOT compressed, sim+reco data products)

Extras

Vertex Performance




60 Scatters
Simulated by
Eldwan Brianne
DUNE ND HPGTPC