SBN software overview

G. Cerati (FNAL)
WireCell Summit
Apr. 11, 2024

Outline

- Overview of data processing in SBN
- Software challenges for the experiments
- WireCell usage and areas for improvement

- Notes:
  - trying to avoid too much overlap with other talks, leads to a technical talk
  - bias towards ICARUS due to personal involvement and coverage at this meeting
  - but requirement of SBN joint analyses imply that most methods will have to be shared

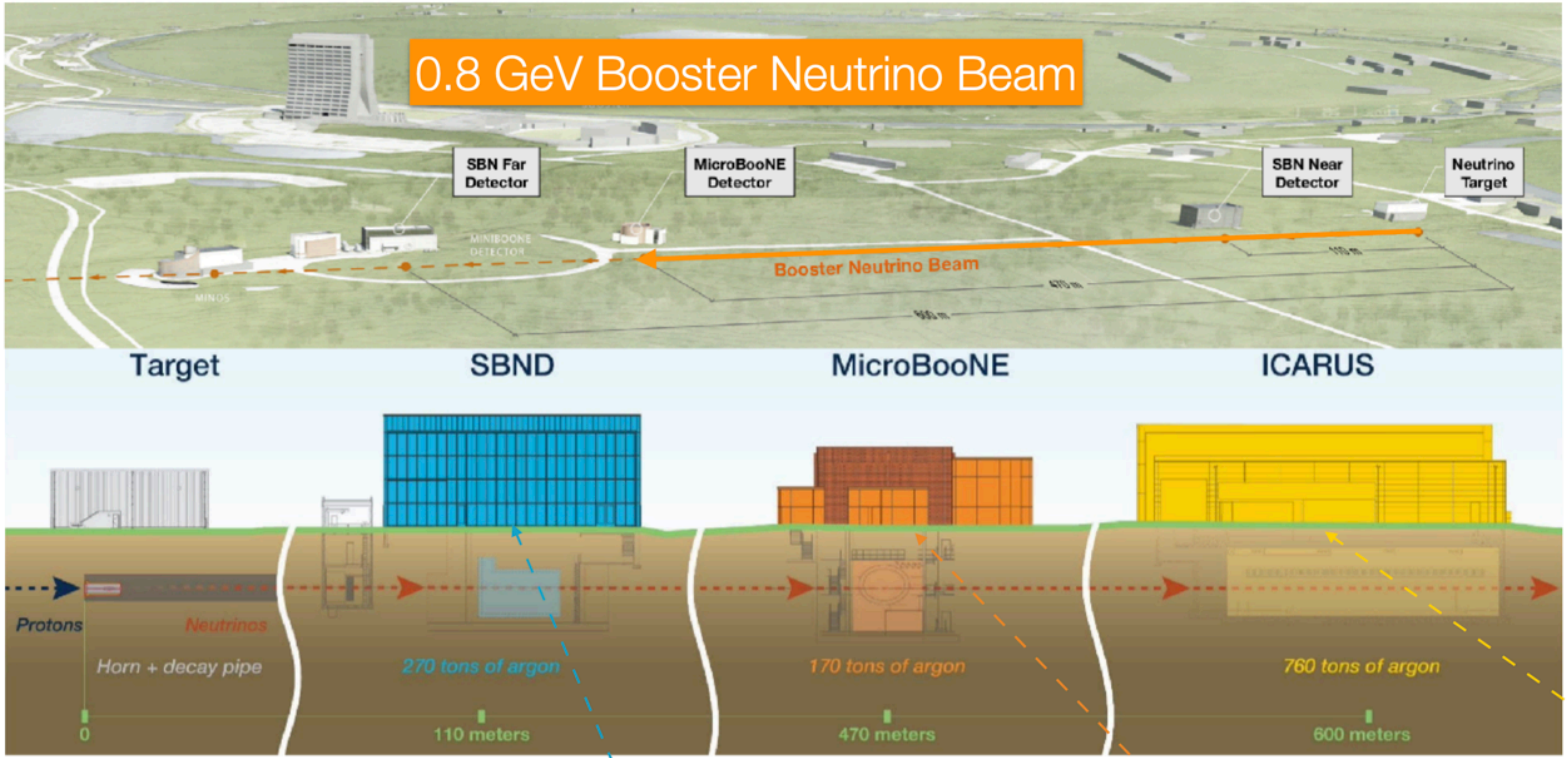
slide credit: A. Campani

The Short Baseline Neutrino (SBN) program

Precision search for **1 eV mass scale sterile ν** to confirm/rule out previous anomalies from past experiments

Sensitive searches for ν_μ **disappearance**, ν_e **appearance**

ICARUS exposed also to **NuMI beam** (6 degrees off axis)



High statistics measurement of **ν-Argon** cross sections for **DUNE**

Search for Beyond Standard Model (**BSM**) physics

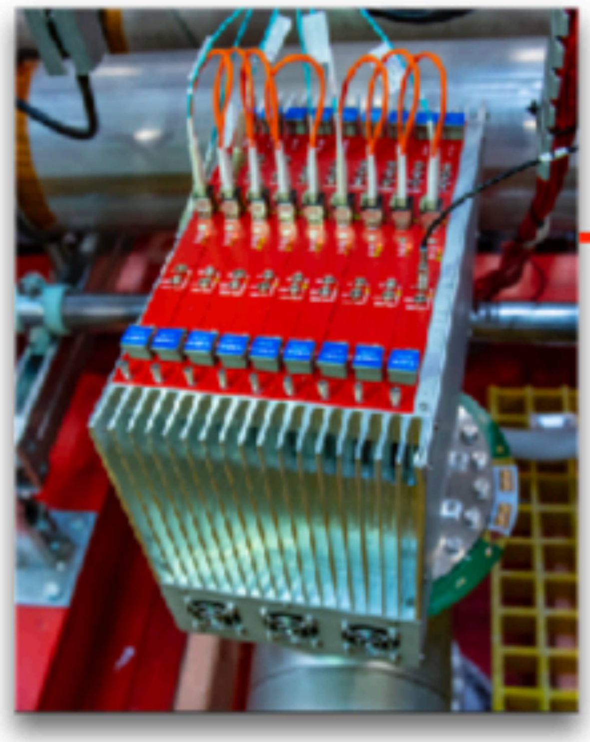
Same detector technology to reduce **systematics** and increase sensitivity

SBND
Near detector

MicroBooNE

ICARUS
Far detector

Event reconstruction in LAr TPCs: ICARUS reconstruction chain



Data

Decoding → **Deconvolution** → **ROI Finder** → **Gauss hits**

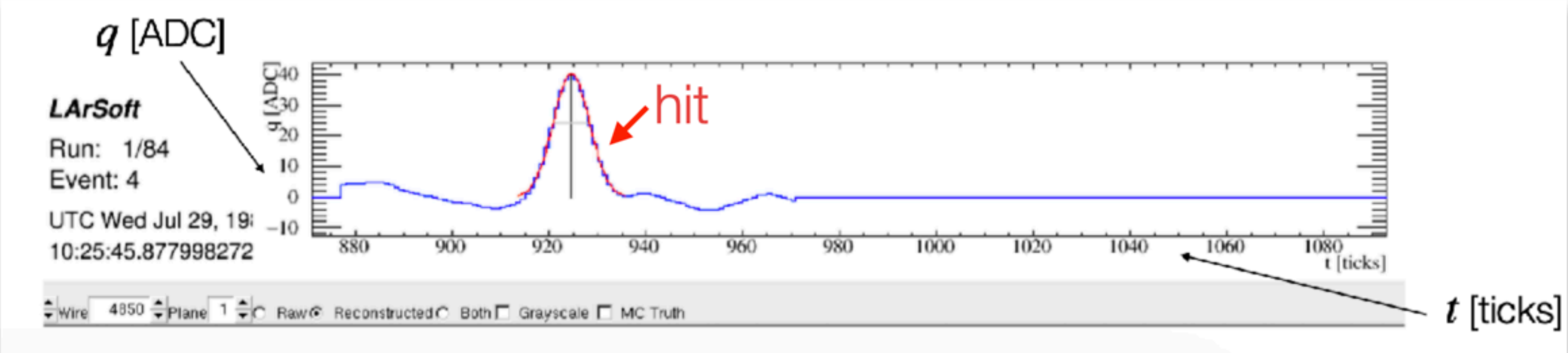
Decoding: Unpack the data and turn it into a **raw waveform**

Deconvolution:
- Removal of **coherent noise**
- **Deconvolution** to remove the $\vec{E}$ distortions and electronics shaping effects on wire signals

ROI Finder: Threshold-based algorithm to identify regions containing ***hits***, i.e. segments of waveforms corresponding to signal.

Gauss hits: Fit each signal hit with **Gaussians:** the area is proportional to n_{e^-} drift electrons that generated that.

Example of deconvolved signal (charge vs time) on a single wire plane after ROI finding and Gaussian fit

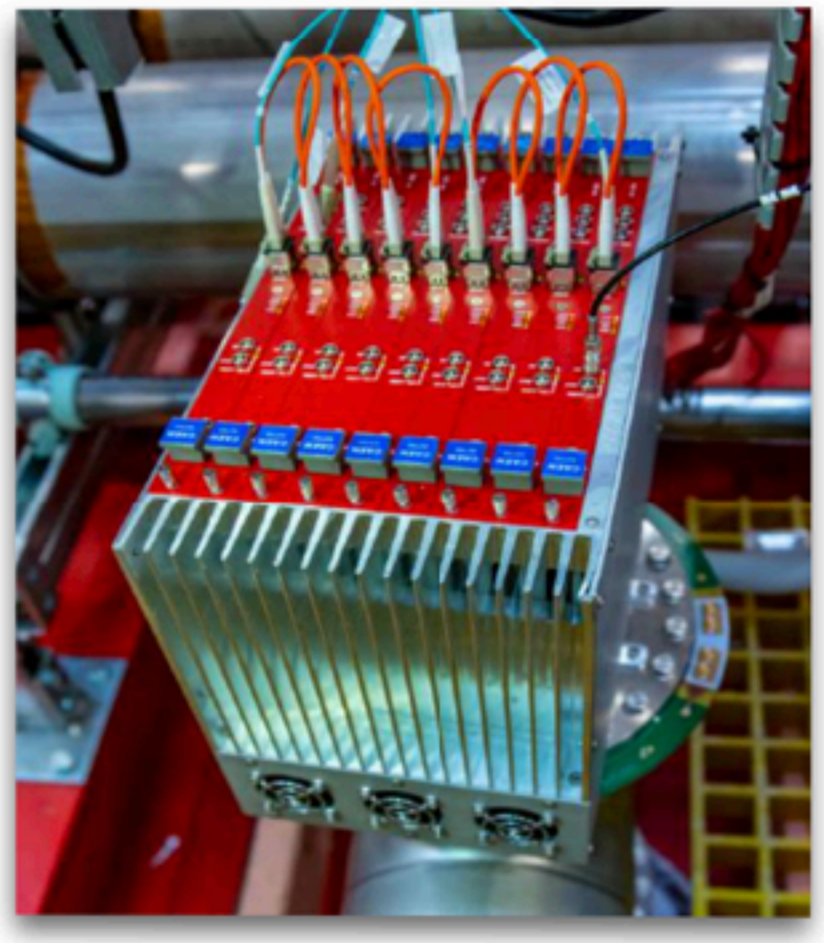


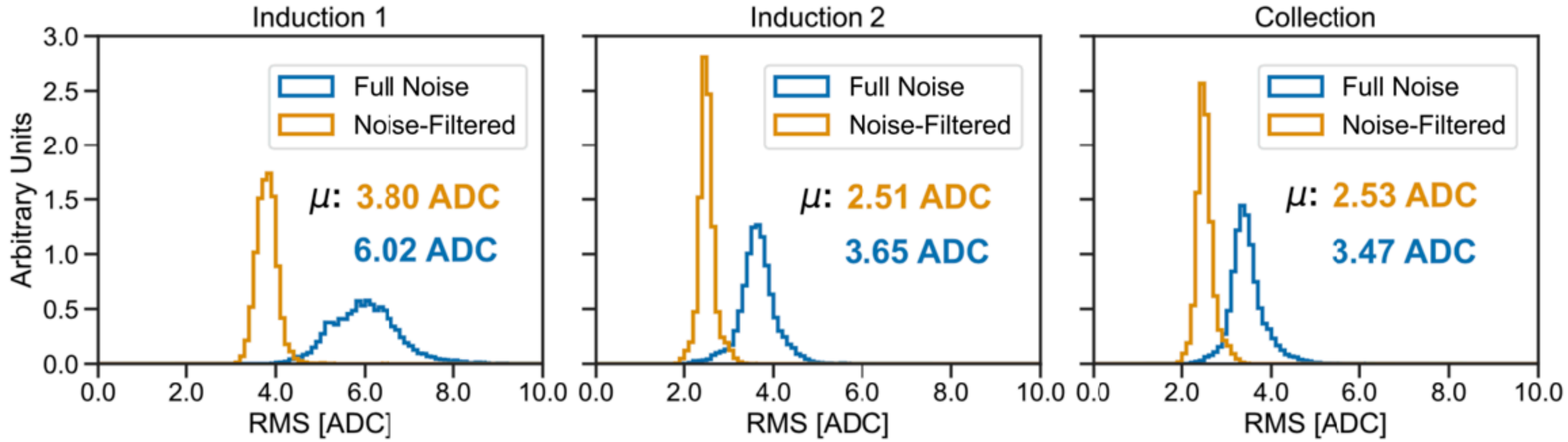
slide credit: A. Campani

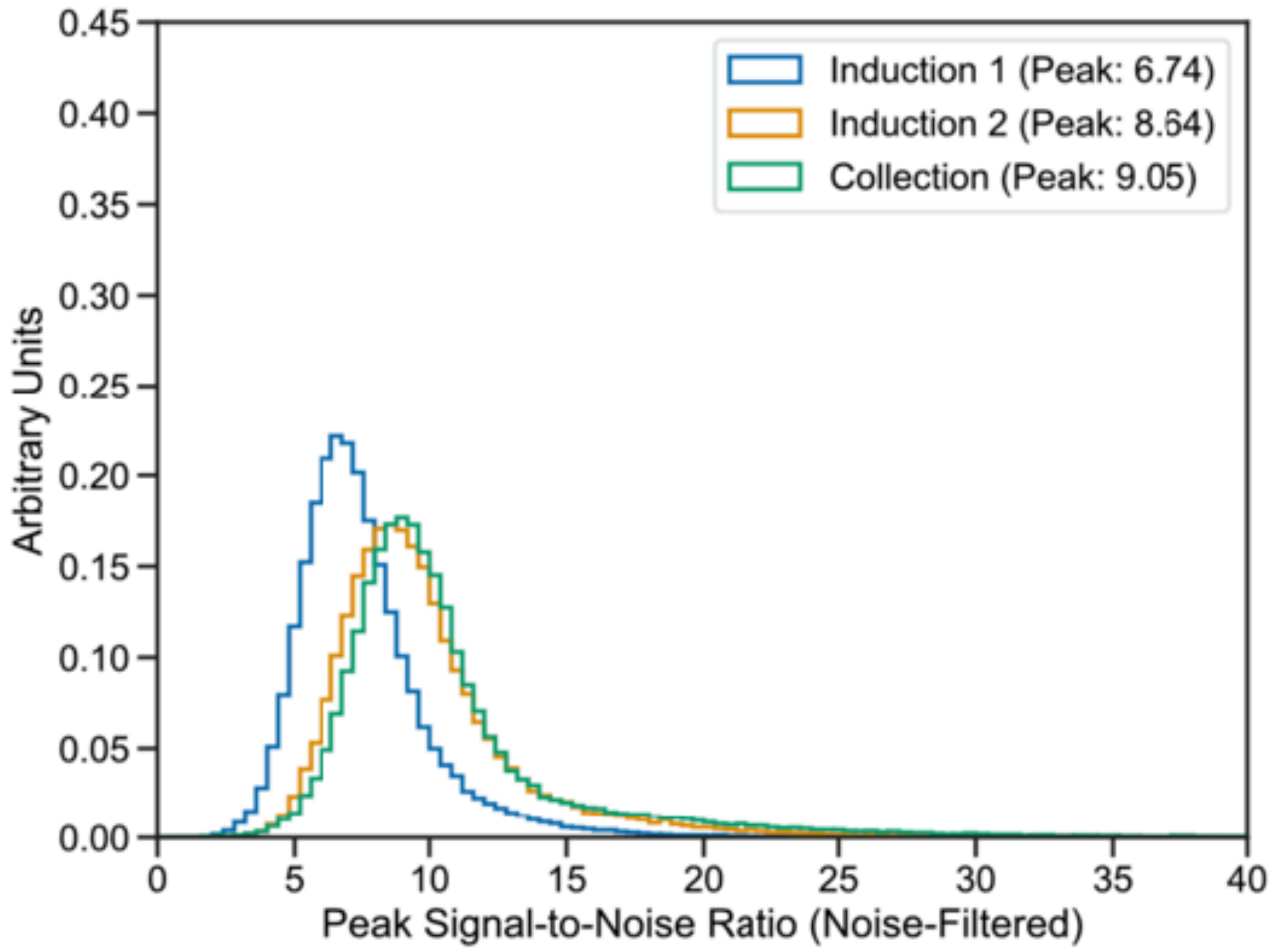
The TPC and upgrades of the electronics readout

- New, higher performance TPC readout electronics compliant with **higher data rates** at shallow depths at FNAL compared to LNGS
- Same modularity/architecture, but integration on a special custom crate: more **compact layout** with analog & digital components in the same board
- Anomalous coherent **noise** inside the 64 channels board found upon detector activation attributed to the ancillary cryogenics instrumentation reduced after several interventions

Eur. Phys. J. C 83:467 (2023)







2D Deconvolution Workflow

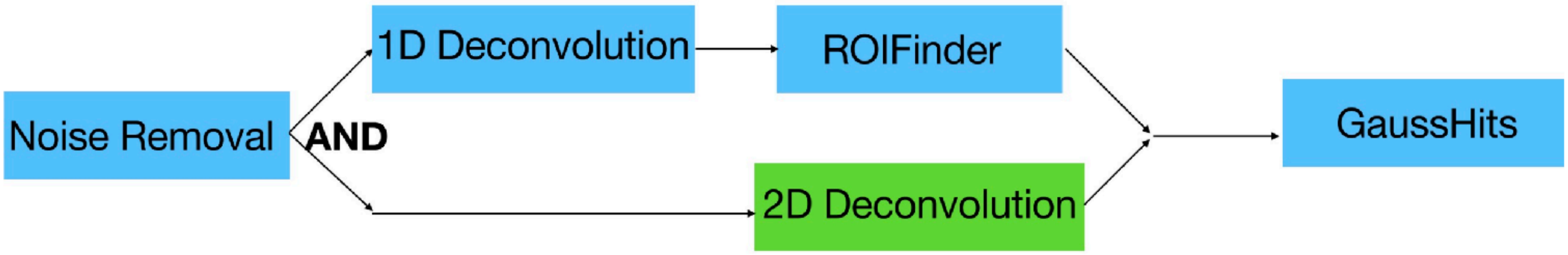
slide credit: M. Mooney

2D Signal Processing

S. Martynenko

More info here

- Blue - current ICARUS SP chain
- Green - WireCell module



- ♦ Previously running ROI finding on 1D deconvolution output
- ♦ Now testing **2D deconvolution** (Wire-Cell) for charge estimation, but still using 1D deconvolution for ROI finding
  - Wire-Cell ROI finding does not work at ICARUS given relatively low S/N ratio compared to other LArTPC neutrino experiments

Event reconstruction in LAr TPCs: ICARUS reconstruction chain

To reduce noise, keep only hits that are **consistent** with 3D points using 2 wire planes combinations.

Gauss hits

Cluster 3D

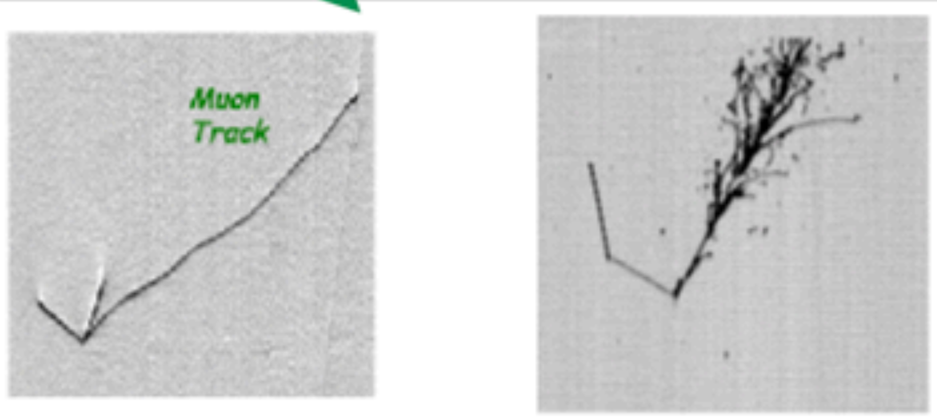
After **Cluster 3D** the reconstruction chain splits in **two parallel paths.**

Pandora-based reconstruction
Downstream reconstruction is based on cluster, *slice* (groups of tracks related to the same particle interaction) and pattern recognition.

Pandora reco

ML-reco

ML-based reconstruction
The full reconstruction chain is based on Machine Learning (ML) techniques



Analysis

Track and shower reconstruction, calorimetry

Signal processing: foreseen change from 1D to 2D deconvolution

- Wire signals are a convolution of **electric field** and **electronics responses**:

$$M(t) = \int_{-\infty}^{+\infty} R(t,t') \cdot S(t')dt$$

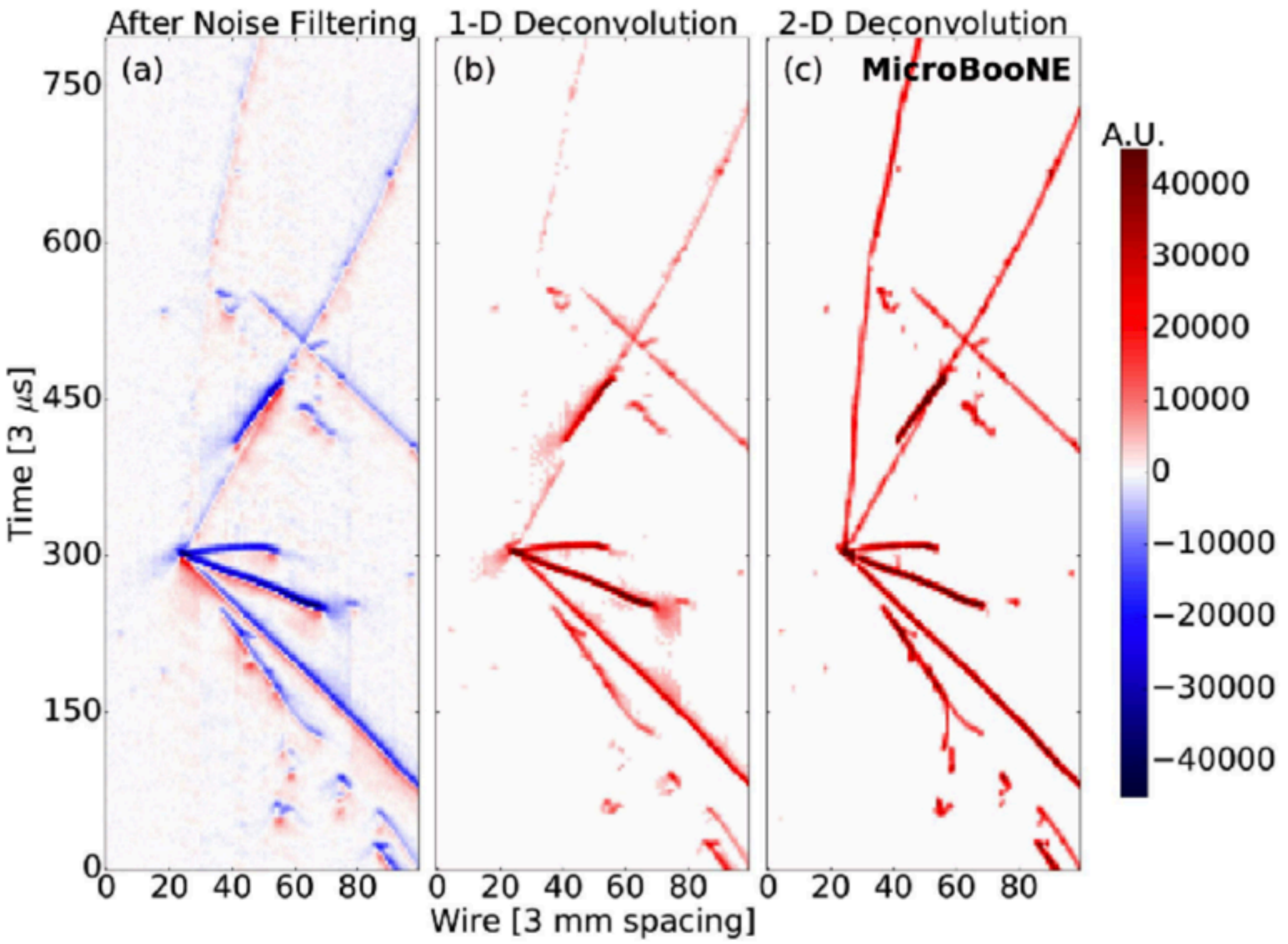
Measured signal | Response function | Original wire signal

- Original wire signal extracted with **1D deconvolution** after applying a filter for noise

- **2D deconvolution** to account for induced charge effects, i.e. charge drifting in nearby wire regions

- improvement of the charge resolution

- higher ε on hits reconstruction for specific track classes



Example of charge recovery from MicroBooNE
Adams, C. et al. 2018 *JINST* 13 P07006 (ArXv link)

Analysis Workflow

slide credit: J. Mueller

Creation of "joint" CAF files with *sbn_ml_cafmaker* gives us access to:

- Event weights (for systematics) (SBN)
- POT information (SBN)
- Event-by-event blinding decision (ICARUS for now)

Link to Github

Colorado State University

Computing challenges for the SBN experiments

- **Processing and storing of large data sets**
- Production of detector systematic samples
- Simulation of cosmic ray background and noise => data overlays

Release integration and CI monitoring

slides credit: S. Seo

ICARUS CI Validation Scheme

❑ <u>Short term:</u>
- **Set of root macros** from each WG produce validation plots for a small set of samples (based on **calibration ntuples** and **flat cafs**).

- Each WG needs to review and **<u>sign off on a first version of the plots</u>**, which becomes a **<u>reference</u>**.
- Towards the next production release we'll periodically produce new validations and WG conveners are required to verify changes are as expected (especially for plots with large chi2 wrt reference).
- Plots are automatically uploaded to the **CI dashboard**, and those with large chi2 are highlighted.
- **<u>New production</u>** releases will need **<u>sign off from all WGs</u>** before starting production.

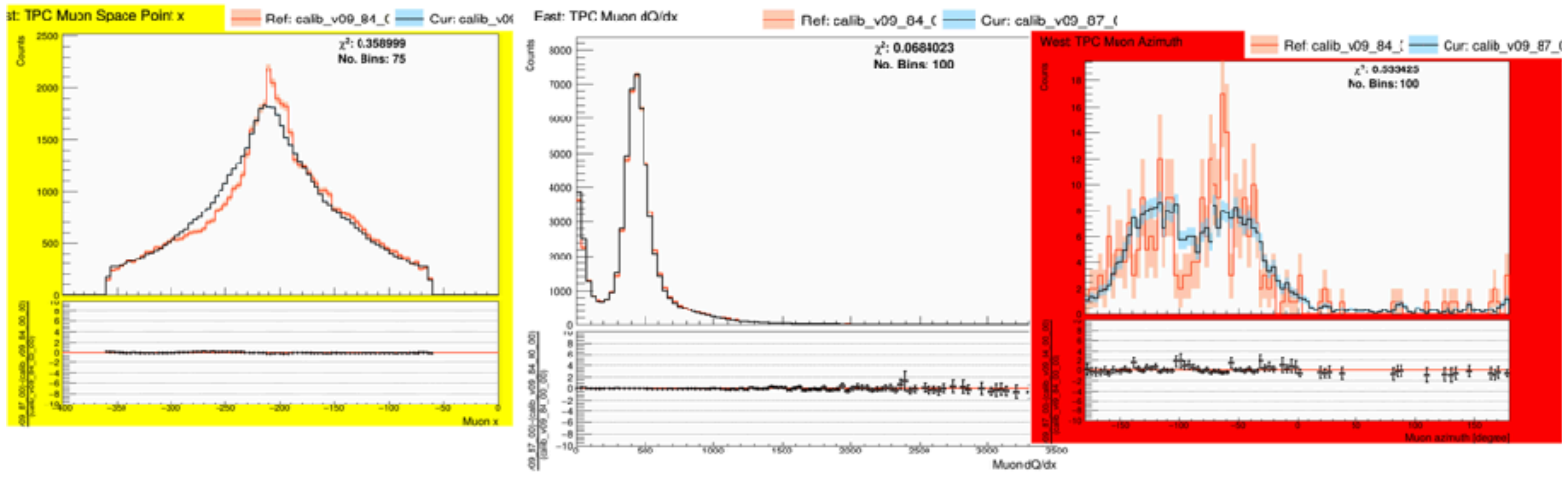
❑ <u>Longer term:</u>
- **Validation process** needs to be integrated into **release management**, both at single PR level (small stats tests) and at weekly integration level (larger scale tests).

- **Sign off from WG conveners with plots** showing chi2 above threshold is required **before moving on**.
- Eventually we could also monitor the full content of calibration ntuples and flat cafs.
- Procedures need to be shared with SBND.

6

Calibration WG

Red. : v09_84_00_01
Blue : v09_87_00



More plots: V09_87_00

14

Fermilab

Resource usage for production jobs with latest release

- icaruscode v09_88_00_02 was cut last week
  - candidate production release pending validation

- BNB+cosmics (5 events), legacy G4 and 2D deconvolution:

```
- Gen    :   3 s/evt CPU time,  1 GB memory, 0.1 MB/evt file
- G4     : 311 s/evt CPU time,  6 GB memory, 260 MB/evt file
- Detsim: 456 s/evt CPU time,  4 GB memory, 440 MB/evt file
- Stage0: 355 s/evt CPU time, 10 GB memory, 340 MB/evt file
- Stage1: 170 s/evt CPU time,  2 GB memory, 380 MB/evt file
```

- Notes:
  - tested on icarusbuild01 physical node
  - refactored G4 requires less time and smaller file size, but still to be validated

Detsim and Stage0 breakdown

Detsim is ~100% WireCell

TimeTracker printout (sec)	Min	Avg	Max	Median	RMS	nEvts
Full event	418.649	455.628	488.356	459.307	23.0995	5
source:RootInput(read)	0.00098451	0.0038618	0.00592583	0.00366134	0.00170015	5
simulate:rns:RandomNumberSaver	4.6139e-05	0.000121756	0.000376716	6.0868e-05	0.000127742	5
simulate:opdaq:SimPMTIcarus	14.6035	15.1688	15.9368	15.3005	0.497188	5
simulate:daq:WireCellToolkit	374.4	411.078	443.24	414.458	22.6778	5
simulate:crtdaq:icaruscode/CRT/CRTDetSim	15.972	16.8439	17.8458	16.8808	0.617255	5
[art]:TriggerResults:TriggerResultInserter	1.9221e-05	3.08422e-05	6.8452e-05	2.1616e-05	1.88673e-05	5
end_path:rootoutput:RootOutput	8.71e-06	1.38044e-05	2.9997e-05	9.439e-06	8.17957e-06	5
end_path:rootoutput:RootOutput(write)	11.7909	12.532	13.2584	12.801	0.605279	5

Stage0 is ~50% WireCell

TimeTracker printout (sec)	Min	Avg	Max	Median	RMS	nEvts
Full event	346.634	354.651	365.464	351.326	6.70443	5
source:RootInput(read)	0.00114783	0.0374612	0.0725585	0.0388746	0.0226184	5
path:pmtfixedthr:DiscriminatePMTwaveforms	0.80448	0.934237	1.16341	0.934323	0.128679	5
path:pmtlvdsgates:LVDSgates	0.0104037	0.0117642	0.0135947	0.0117642	0.00110583	5
path:pmttriggerwindows:SlidingWindowTrigger	0.0273565	0.0321254	0.0379105	0.0318242	0.00364924	5
path:triggersimgates:FixBeamGateInfo	0.000888898	0.00155083	0.00320941	0.00112763	0.000845746	5
path:emuTrigger:TriggerSimulationOnGates	0.00931932	0.0112574	0.0154496	0.0107704	0.00221715	5
path:pmtbaselines:PMTWaveformBaselinesFromChannelData	0.344556	0.39808	0.508983	0.3609	0.0607188	5
path:ophit:OpHitFinder	5.59812	6.42422	7.97744	6.30579	0.859528	5
path:mcophit:ICARUSMCOpHit	1.75225	2.23183	2.73487	2.25809	0.394396	5
path:opflashCryoE:ICARUSFlashFinder	0.198695	0.266363	0.358825	0.256309	0.051937	5
path:opflashCryoW:ICARUSFlashFinder	0.177023	0.19732	0.232935	0.195373	0.0190908	5
path:MCDecodeTPCROI:MCDecoderICARUSTPCwROI	91.825	92.6351	92.9652	92.7813	0.411964	5
path:decon1droi:Decon1DROI	17.4283	17.7859	18.6501	17.6366	0.439892	5
path:roifinder1d:ROIFinder	16.0557	16.3237	16.8052	16.253	0.27917	5
path:decon2droiEE:WireCellToolkit	35.9042	43.4819	51.0719	40.2551	6.31704	5
path:decon2droiEW:WireCellToolkit	39.305	42.2052	50.1131	40.5877	4.02346	5
path:decon2droiWE:WireCellToolkit	37.7498	39.7816	45.7896	38.4309	3.01824	5
path:decon2droiWW:WireCellToolkit	36.686	38.1463	39.7439	37.4488	1.22709	5
path:roifinder2d:ROIFinder	3.45142	3.49348	3.52392	3.51652	0.0321835	5
path:gaushit1dTPCEW:GausHitFinder	1.22029	1.71985	2.18699	1.8382	0.417721	5
path:gaushit1dTPCEE:GausHitFinder	0.984533	1.38633	1.98289	1.08841	0.426475	5
path:gaushit1dTPCWW:GausHitFinder	1.06748	1.50929	2.36499	1.37333	0.44524	5
path:gaushit1dTPCWE:GausHitFinder	0.989294	1.32403	1.97224	1.22426	0.338215	5
path:gaushit2dTPCEW:GausHitFinder	1.44967	2.22485	3.02078	2.49695	0.622572	5
path:gaushit2dTPCEE:GausHitFinder	1.33179	1.92695	2.75476	1.61618	0.592712	5
path:gaushit2dTPCWW:GausHitFinder	1.42169	1.98152	2.99014	1.85006	0.531423	5
path:gaushit2dTPCWE:GausHitFinder	1.26185	1.69087	2.58148	1.51325	0.45974	5
path:purityana0:ICARUSPurityDQM	0.00282278	0.00315642	0.00441728	0.00285393	0.000630546	5
path:purityana1:ICARUSPurityDQM	0.00239621	0.0024053	0.00242492	0.0024036	1.03349e-05	5
path:crthit:icaruscode/CRT/CRTSimHitProducer	1.62154	1.91707	2.47159	1.83766	0.289565	5
path:crttrack:icaruscode/CRT/CRTTrackProducer	0.0138548	0.0181775	0.021415	0.0187934	0.00246118	5
path:crtpmt:icaruscode/CRT/CRTPMTMatchingProducer	0.0125697	0.0137383	0.014816	0.0142923	0.000951218	5
[art]:TriggerResults:TriggerResultInserter	1.8285e-05	3.48246e-05	9.1819e-05	2.0839e-05	2.85463e-05	5
end_path:rootOutput:RootOutput	2.6218e-05	8.55438e-05	0.000306219	3.0281e-05	0.00011037	5
end_path:rootOutput:RootOutput(write)	28.692	34.5176	39.9898	36.512	4.86025	5

Running the jobs

see also Barnali's talk this morning — here focus on the other side of the coin

Context: multithreading for production jobs

- art and larsoft provide multithreading capabilities through TBB library
  - art multithreading can process concurrently data across events or within the same event

- Grid allocations have total available memory split by CPU cores
- Grid jobs often need slots with large memory, thus getting multiple cores
- Production jobs are however running single-threaded, thus use only one core

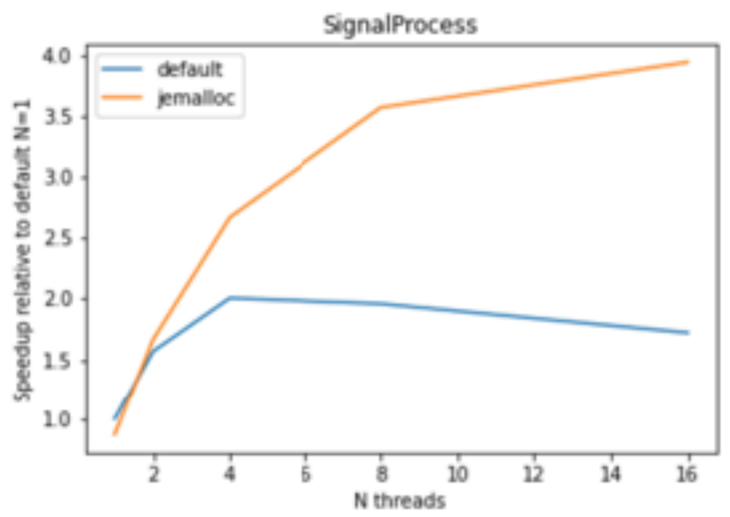
- We can achieve significant processing speedups if we are able to exploit multithreading and increase our core utilization efficiency
  - multithreading within the event doesn't need to load more event data, can exploit unused cores given the same memory allocation
  - target for production jobs is to have efficient multithreading at moderate thread counts

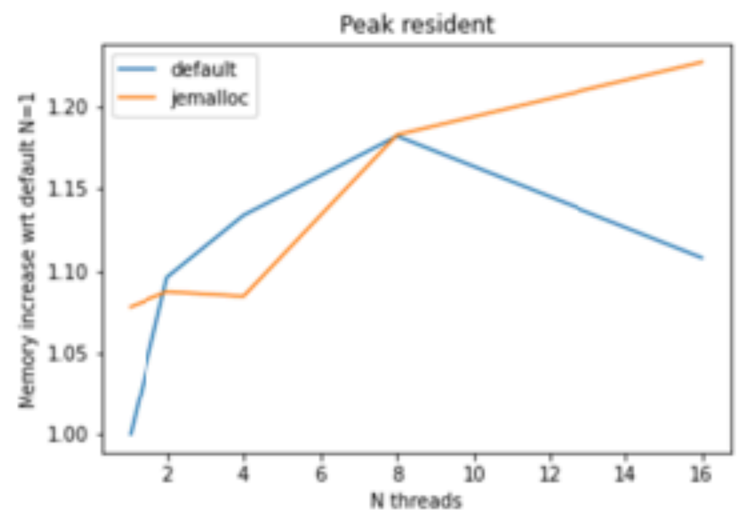
Scaling results

Out of the box, not necessarily optimized/tuned.

- Tested on icarusbuild02, without other ongoing jobs
  - not a production environment
- Can achieve up to 4x speedup for the 4 modules that are multithreaded
- Full stage0 processing speedup limited by other time consuming modules
  - but some of them may be low hanging fruits for speedups
- Memory increase is overall small, as expected

CPU time speedup





memory usage increase

https://indico.fnal.gov/event/57914/contributions/260764/attachments/164436/218111/scidac-parallal-larsoft-mt-wrkshp.pdf

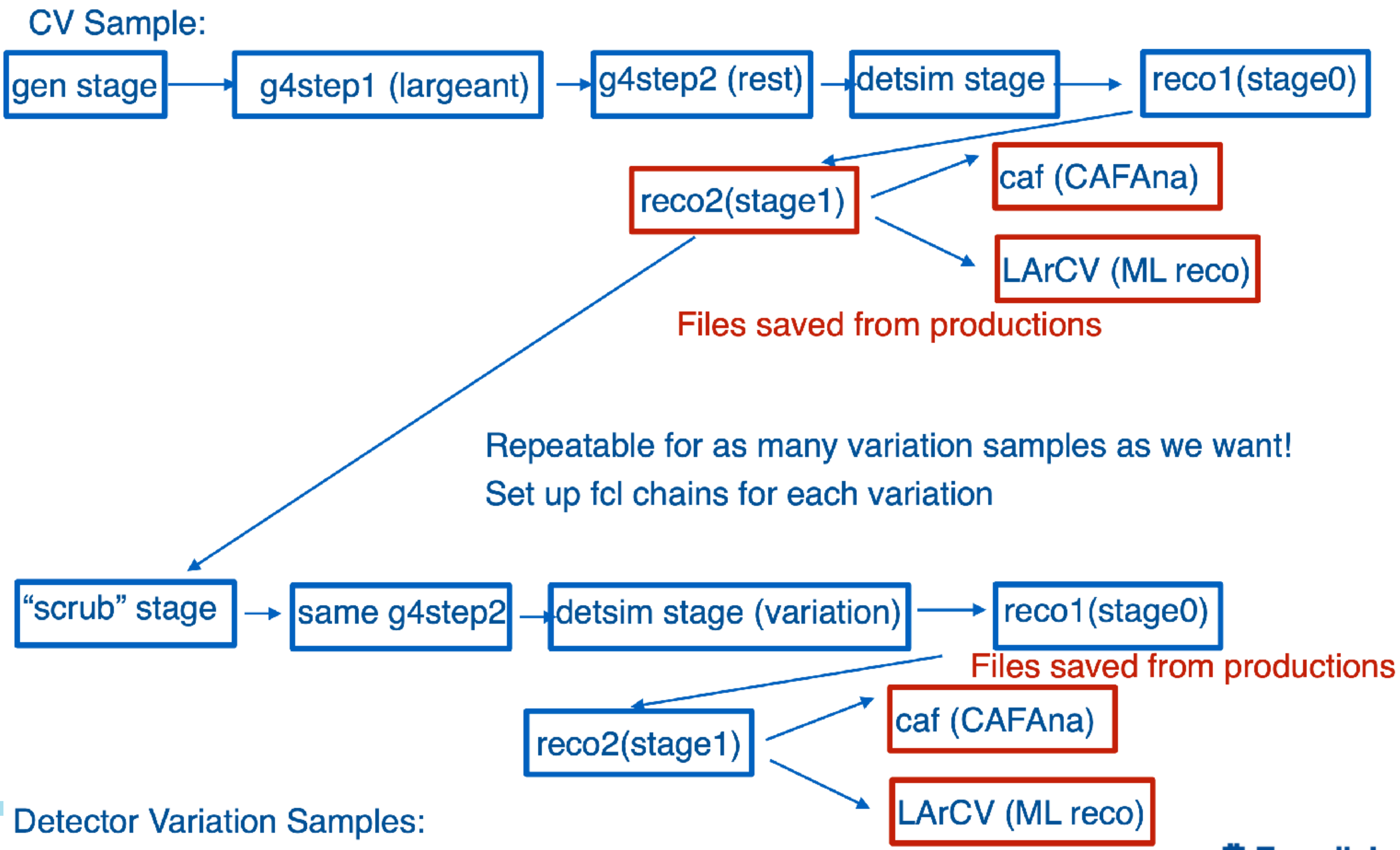
"1D" signal processing chain fully multithreaded in production: additional cores requested because of memory usage do not sit idle.

Computing challenges for the SBN experiments

- Processing and storing of large data sets
- **Production of detector systematic samples**
- Simulation of cosmic ray background and noise => data overlays

slide credit: J. Zettlemoyer

Diagram of the wall of text



Fermilab

MC sets to evaluate uncertainty

slide credit: I. Safa

- How many unique variation samples are needed to quantify a single detector systematic at the 1, 2, and 3 sigma levels?
- Assuming we only evaluate at integer sigma levels (+1, -2 etc) then the number of sets needed are shown on the right.
- Knowing this, what is the minimum amount of statistics **per variation sample** required to reach our target?

Fit to width

Number of MC variation sets	1 sigma	2 sigma	3 sigma
1 Parameter Model	2	4	6
2 Parameter Model	8	24	48

results — summary

- For an SBN oscillation analysis with only **three** detector systematics, assuming **1** is a **single parameter** and **2** have **2 parameters** (recombination and diffusion for eg.).
- We use the numbers per variation sample, and the multipliers from Slide 3 **assuming we only ever evaluate +/- 1 sigma**

Best case scenario for 10 bins:

Total Events:	**17M** SBND	—	**31M** ICARUS
Total CPU Hours:	**650k** SBND	—	**11.8M** ICARUS
Total Disk Space:	**5.5TB** SBND	—	**22TB** ICARUS

*flatcaf only

results — summary

- For an SBN oscillation analysis with only **three** detector systematics, assuming **1** is a **single parameter** and **2** have **2 parameters** (recombination and diffusion for eg.).

- We use the numbers per variation sample, and the multipliers from Slide 3 **assuming we also want to evaluate +/- 2 sigma variations**

Best case scenario for 10 bins:

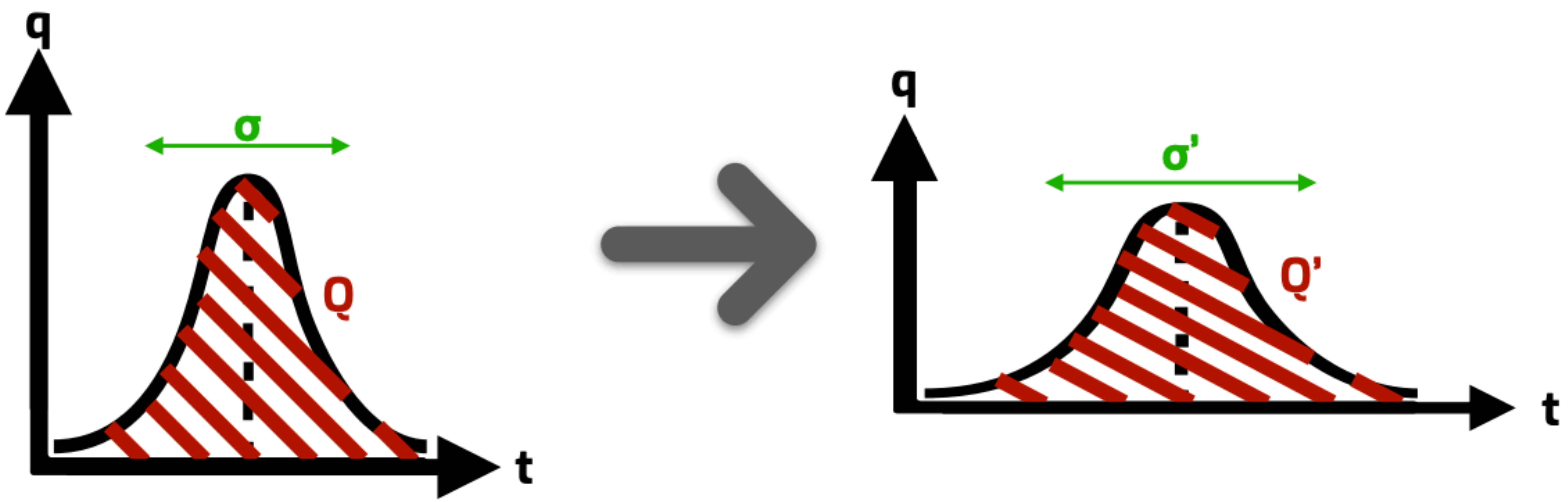
Total Events:	**48M** SBND	—	**90M** ICARUS
Total CPU Hours:	**1.9M** SBND	—	**34M** ICARUS
Total Disk Space:	**15.8TB** SBND	—	**63TB** ICARUS

*flatcaf only

Another possibility moving forward: The WireMod tool

slide credit: J. Zettlemoyer

- WireMod is a tool initial developed by MicroBooNE designed to directly shift the TPC signals
- The idea is to directly modify the MC deconvolved wire signals to make them look more like the data
- The tool maps the original track integral and width to new values based on the location of the hit and track direction
- Current implementation assumes deconvolved signals are Gaussian, but is possibly configurable within the tool



Eur. Phys. J. C 82, 454 (2022)

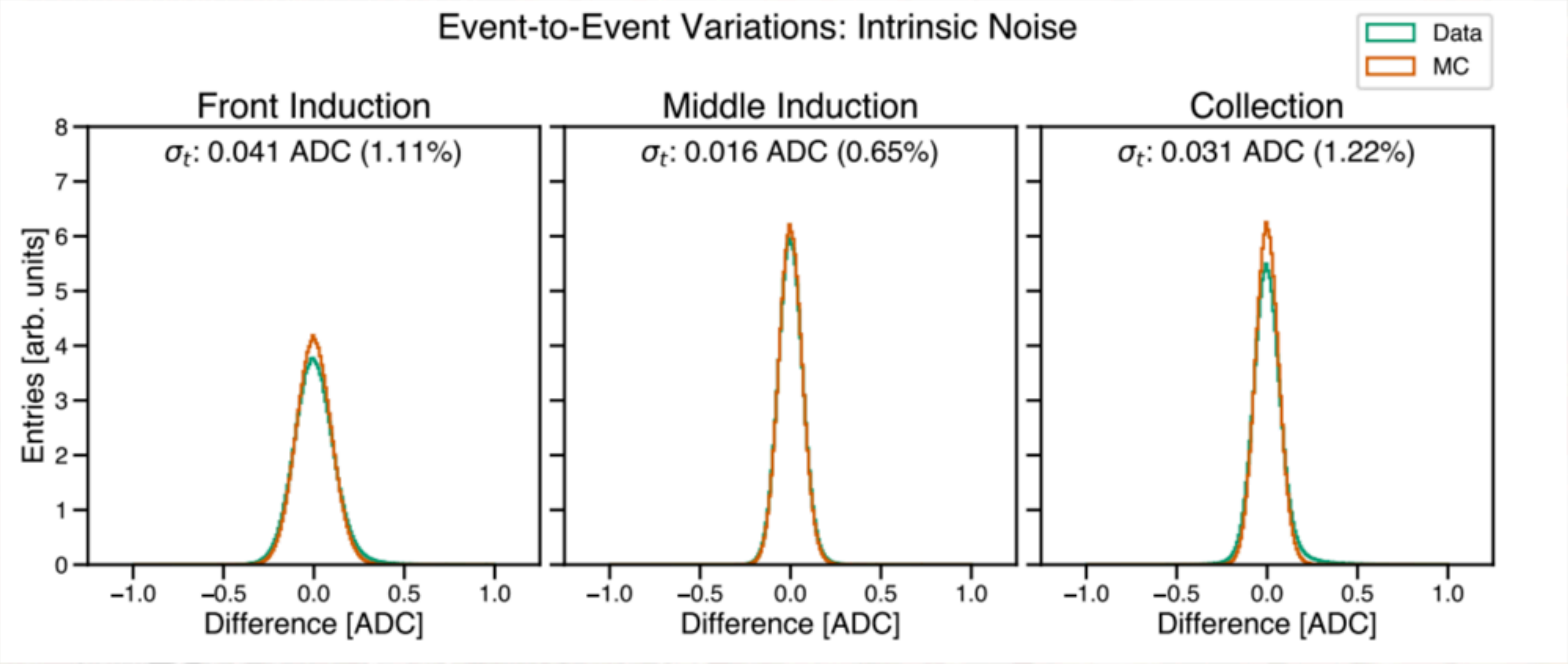
Fermilab

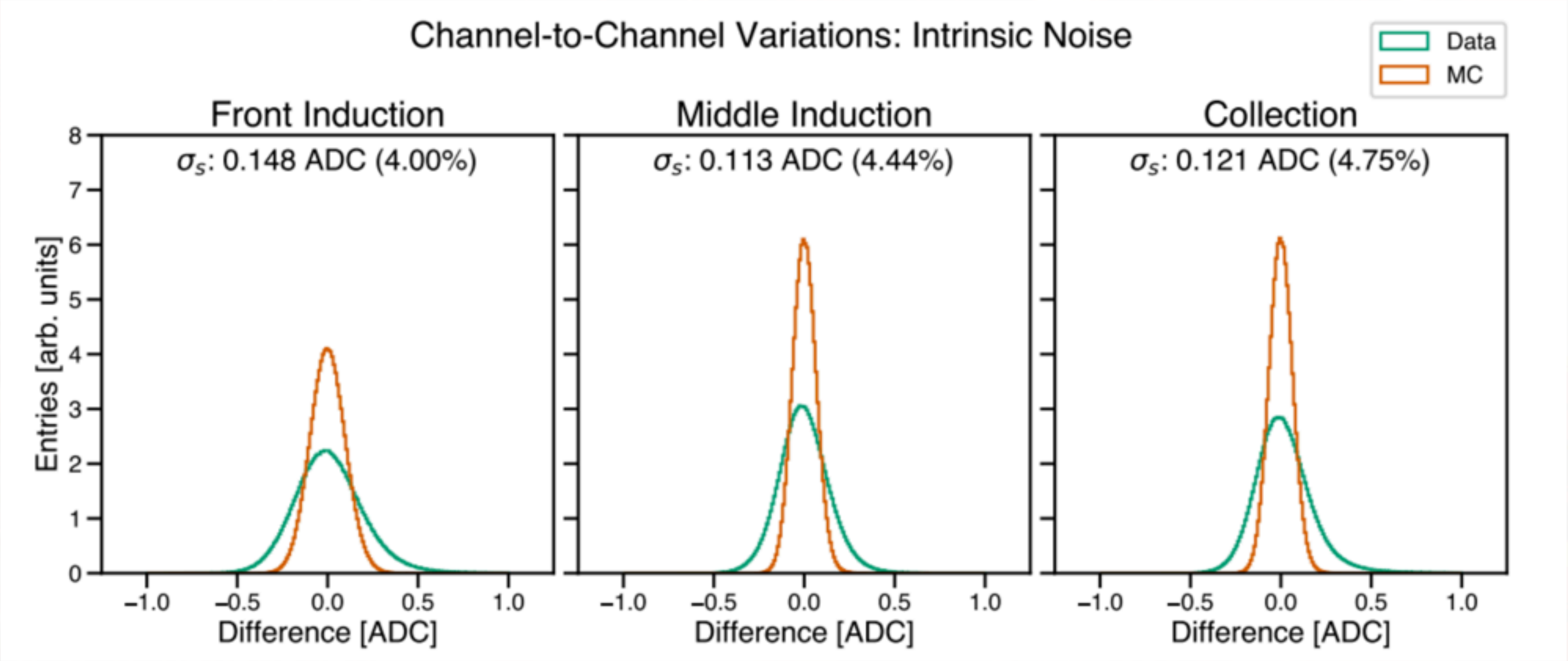
Computing challenges for the SBN experiments

- Processing and storing of large data sets
- Production of detector systematic samples
- **Simulation of cosmic ray background and noise => data overlays**

Simulating TPC Noise

- Understanding noise within the TPC is important for spatial resolution and calorimetric reconstruction
  - Lots of effort to categorize (coherent vs. intrinsic) noise within the detector
- A noise model is made using detector data to be applied in the simulation
  - Has some limitations modeling channel-to-channel spatial variations

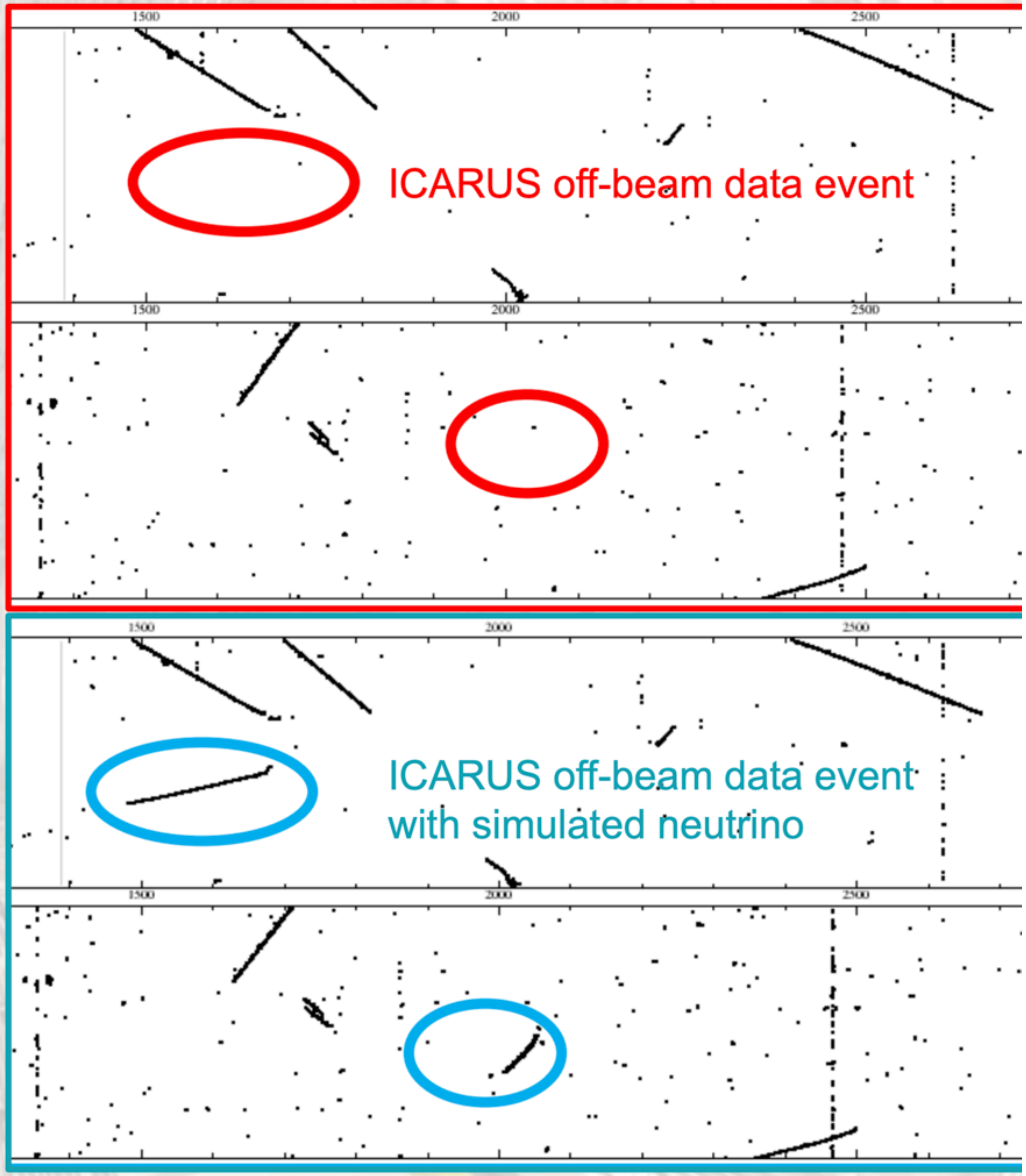




slide credit: A. Heggestuen

Motivation for overlays

- ✓ Don't have to rely on Monte Carlo cosmic generators to model our cosmic backgrounds
  - – CORSIKA, FLUKA or other cosmic ray generators may have different composition or flux compared to reality
- ✓ Reduced dependance on detector noise simulation
- ✓ Reduced computing time spent simulating cosmic backgrounds
- ✓ Get radiological or other backgrounds that aren't modeled for free



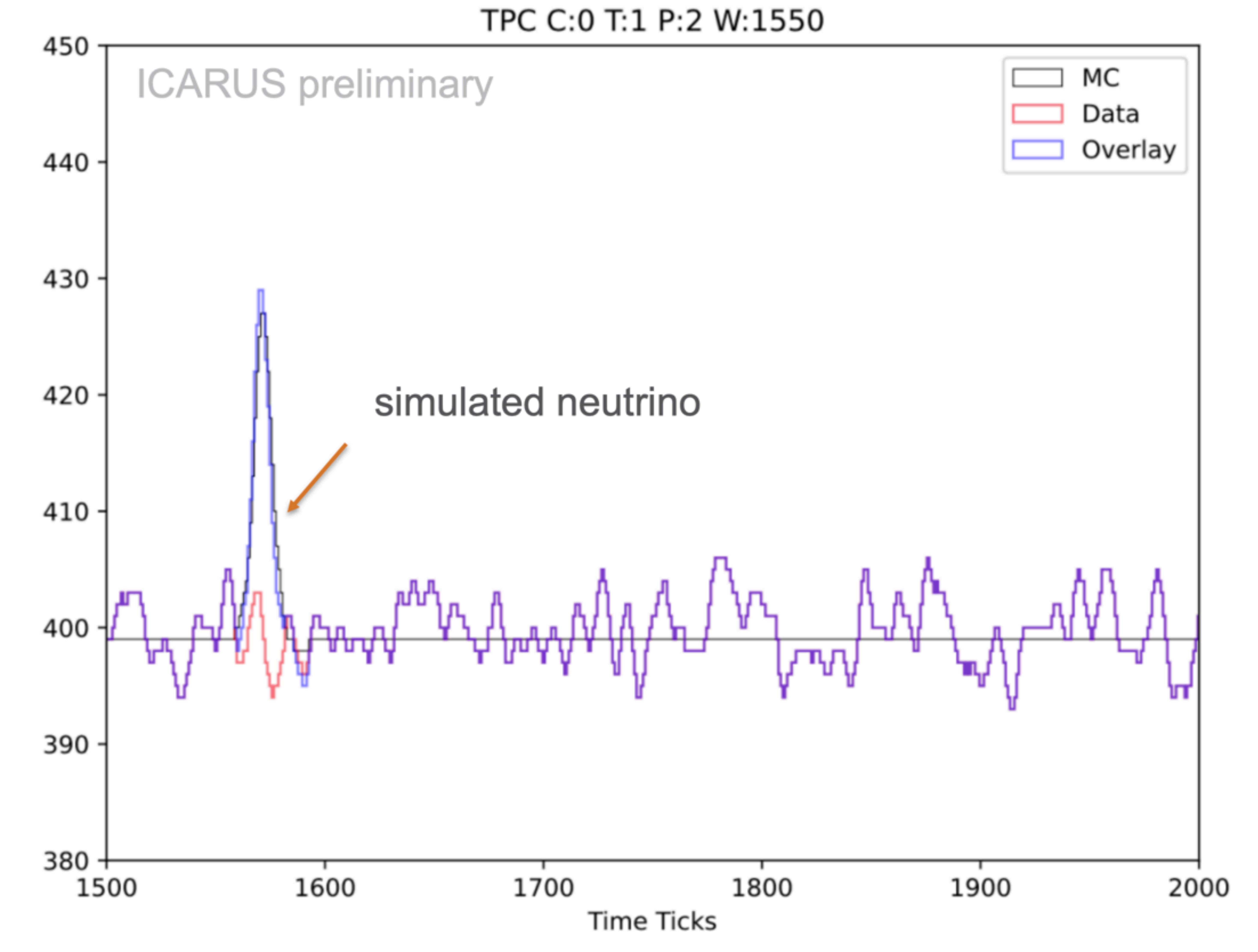
slide credit: A. Heggestuen

Recipe for making cosmic overlays

1. Add simulated + data signals
   – Overlay raw waveforms/signals from each subsystem together
2. Calibrating simulation to match data
   – Dead channels, readout board dead time, simulating any needed detector effects
3. Synchronizing time in simulation and data
   – Applying any offsets, making sure times line up
4. Producing overlay files
   – Create an overlay production chain to mix simulation and data together, requires additional steps to our normal data processing

Adding TPC waveforms

- Here we see a raw waveform from a single TPC wire in off-beam data overlayed with simulated neutrino
- See that the raw TPC signals are correctly being merged from data and simulation
- *Work is ongoing*



WireCell usage and areas for improvement

- The following is a list of ideas/needs of improved usage of WireCell tools in SBN. Apologies if any of them sounds trivial to experts!
  - Improved WireCell 2D ROI finding for warm electronics detector?
    - interesting discussions yesterday, the perspective of using AI/ML for ROI finding is attractive
  - Jsonnet configuration import&replacement for detector variations?
    - may be a limitation of the fhicl-Jsonnet interplay?
  - Input/ideas for detector systematics?
    - non-reweightable, huge production efforts
  - Optimization of WireCell modules?
    - Turning on multi-threading and other speedups?
    - Further reduce memory footprint?
      - Are already limited by I/O reads/writes? 2D deconvolution memory usage was already reduced by ~2x. thanks to Haiwang and the WireCell team for the prompt help!

Conclusions

- Processing of large data sets in SBN presents significant challenges
- Work is ongoing to bring the software to a steady state for full SBN results
- WireCell is integral part of the SBN software and we look forward to continue working together towards the improvements needed to reach the SBN goals