



DUNE Feedback to HEP-CCE

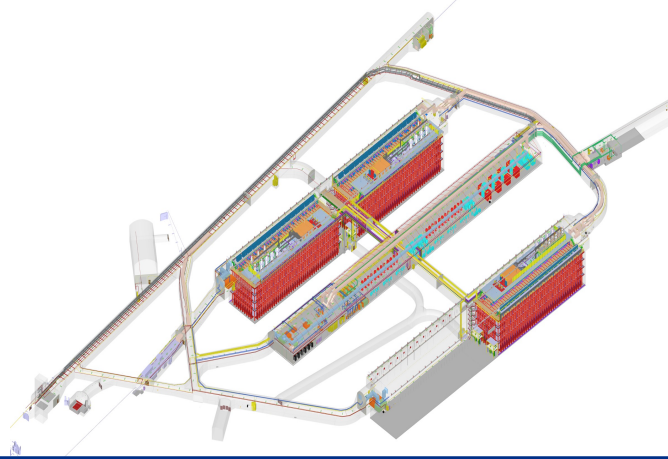
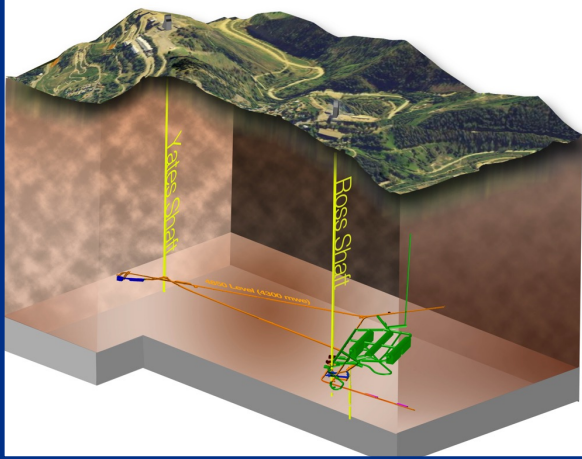
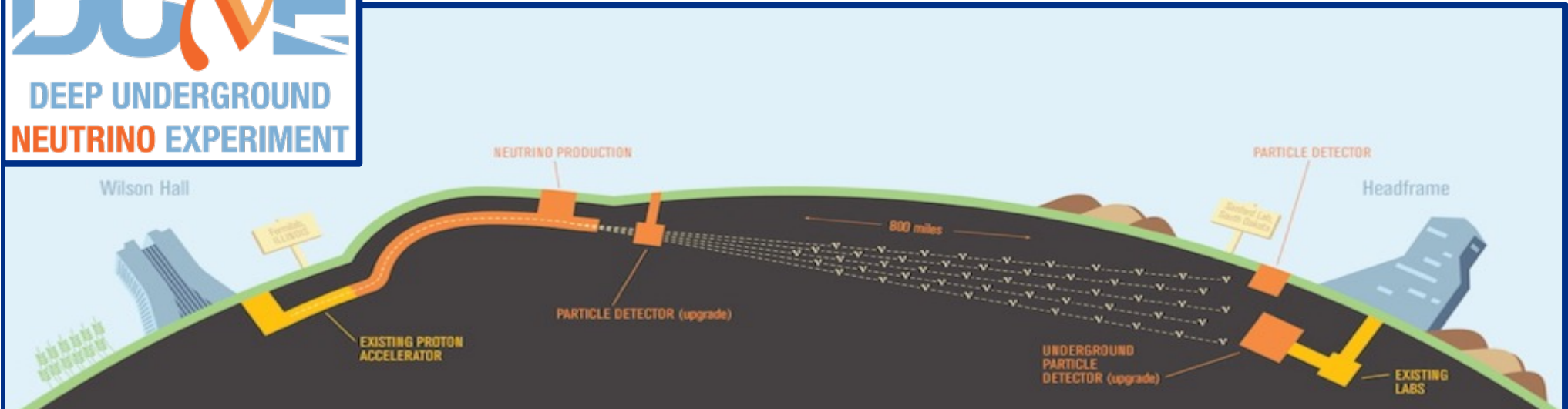
Andrew Norman
HEP-CCE All Hands Meeting
April 2023

In partnership with:



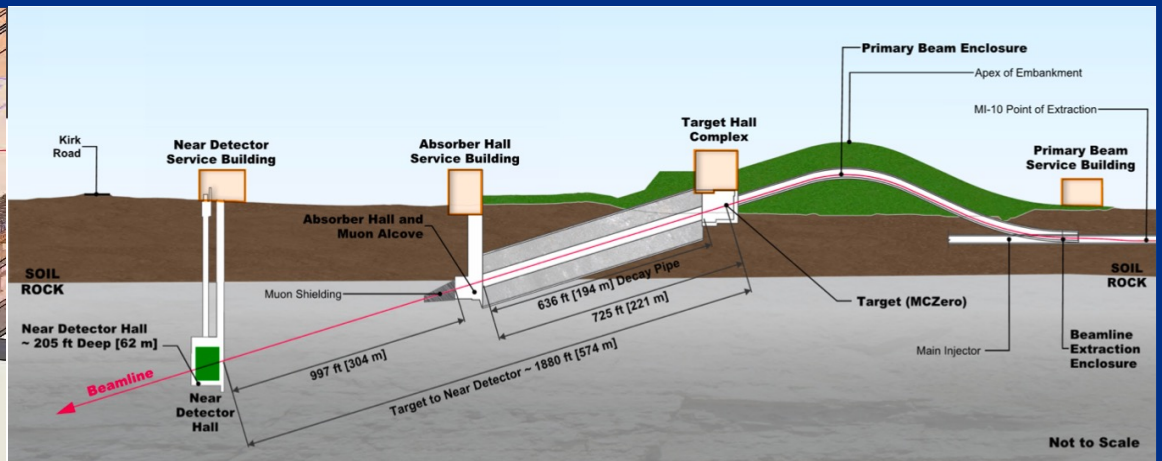
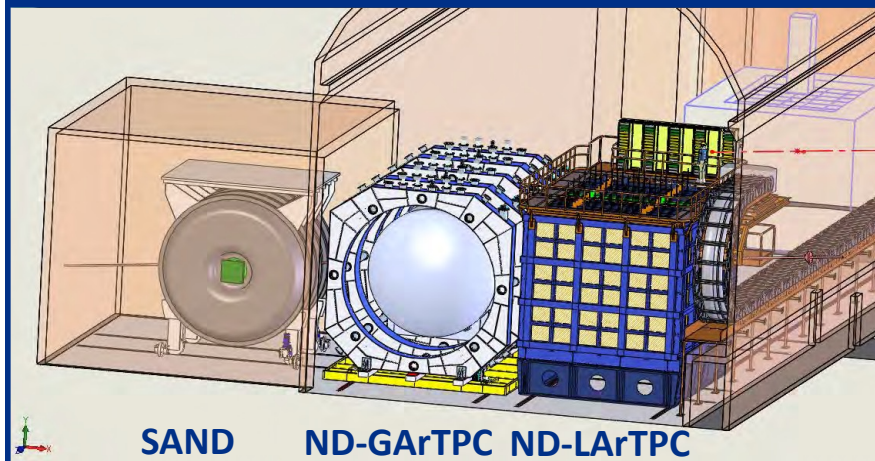
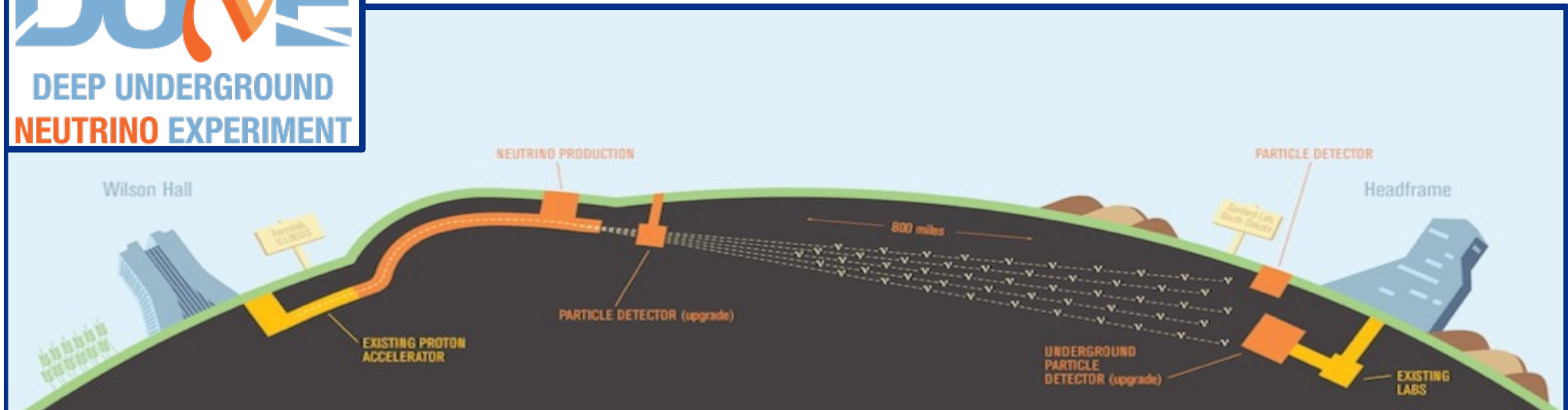


Far Detector Complex





Near Detector Complex



DUNE is...

- DUNE is a **combination** of:
 - Massive LArTPC modules located at the far site (Sanford Underground Lab) in South Dakota
 - Combination of multiple technologies of high rate/granularity detectors at near site cavern (FNAL, Batavia IL).which may slide on rails (DUNE-Prism)
 - A mega-Watt+ neutrino beam (FNAL)
- Each of the components brings with it computing challenges
 - Far Det: Low rate/Large event size (~6 Gb/trigger)
 - Near Det: Multi-detector sync, fine Granularity, high occupancy, moderate trigger rate
 - Beam: High interaction pileup

Mike Kirby says....

I asked Mike, “What do we need to do?
(prioritized, near term, concerning portability)

....and Mike said “



- michel electron finder that goes through SONIC to be adaptable to hardware on the inference server
- taking the 2x2 ML reco or simulation algorithms and making them portable
- FD low-energy trigger algorithms being portable
- low-energy reco algorithms being portable for doing fast reco on all available accelerator resources for SN reco
- signal processing/data prep (basically everything up through hit finding and ROI)
- “

Which means...

1. There are core physics tasks which want ML solutions, but we aren't sure how to make this portable
2. We have an entire detector complex that we haven't considered for acceleration and portability
3. We need to be able to reproduce our trigger and small signal physics (off detector)
4. Near-realtime supernova detection/analysis probably requires large LCF resource pools or aggregating across LCFs
5. The initial (raw) data processing is highly accelerable, we are making progress, need more

Answering HEP-CCE Questions


DUNE Computing CDR


- There is a well defined, public, conceptual design report
 - FERMILAB-DESIGN-2022-01 (<https://inspirehep.net/literature/2171912>)
 - This sets forth the major designs and implications for the computing model
- There are 5 top level computing challenges called out, 4 of which are directly impacted by HEP-CCE work and the future CCE roadmap:
 - Large memory footprints (and data representations)
 - Data storage and processing on heterogeneous resources
 - Integration of machine learning
 - Efficient/sustainable use of heterogeneous resources

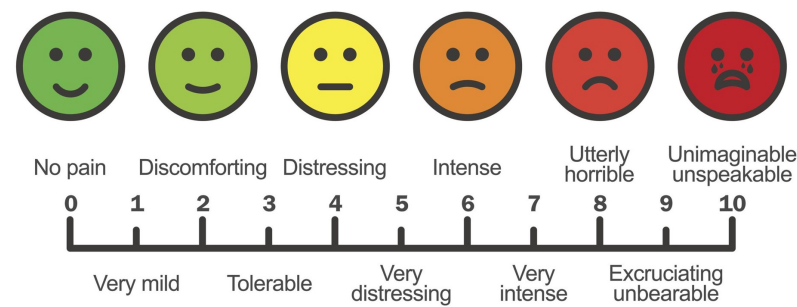
Pain Points (1)

- Data representation & In core memory footprint

- Today this is 

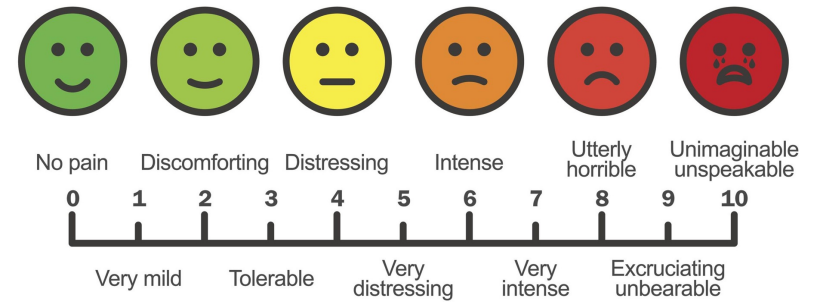
- If we can segment/split/subset/page/etc... our readout data this most likely becomes 

- If we need high fidelity for AI/ML this probably ends up  due to complexity and bandwidths needed for offloading to accelerators



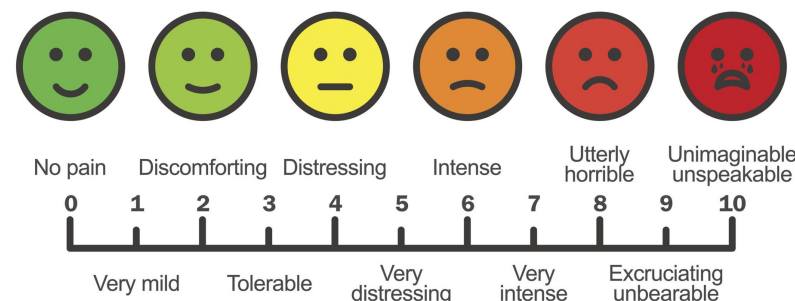
Pain Points (2)

- Data storage/caching w/ complex workflows (*especially with HPC*)
 - Today this is borderline non-existent
 - We do almost all data management by hand
 - Rucio based management is in its infancy
 - Integration with the workflow engines requires significant R&D
 - R&D + HPC center adaptations are required to get this to the point where it is at least -- but can fulfill the core science needs of the experiment
 - Data representation & Storage R&D could radically change this
 - Data life cycle + organization for DUNE may naturally want to look different than past neutrino experiments



Pain Points (3)

- Efficient use of accelerators and effect on workflow topology & scheduling
- Wirecell work is showing that using GPUs efficiently for early stage processing is 😊
- SONIC/Triton style inference server tests are also showing good results in terms of accelerating portions of workflows 😊
- Problem is how to do this at scale given the mismatches between event loop processing and accelerator offloads (in an HPC environment in particular)
- True co-scheduling is 😞 and it's not completely clear how much it buys us in performance and efficiency without major changes to workflow management.
 - Needs real investigation and R&D



Role of HPC for DUNE

- HPC is logical choice for initial stage data processing
 - Provides GPU acceleration (at scale)
 - Provides memory footprint (and transfer bandwidth)
 - Has realistic storage caches and incoming bandwidth capacity to match DUNE data volume/rates

- After initial data reduction/reconstruction makes less sense

- “Physics” level reco representations are small.
Traditional HTC/grid computing can handle the low occupancy
far detector after ROI processing etc...



Role of accelerated ML
training and inference
unclear

- HPC (analysis facility style) may be the ideal fit for analysis level processing of common [monolithic] data sets
- HPC is the ONLY place to perform full parameter fits due to compute budget
- HPC is the ONLY place that could handle near-realtime burst computations (supernova)

Heterogeneity

- IF we are intending to use HPC for initial processing we need to be portable across what different LCFs choose for their accelerator architecture
 - Wirecell work already shows this is possible. ✓
 - PPS testing/porting needs to be expanded into other portions of the code base
 - Determine generalized approach which will work/be performant for most DUNE processing
 - Or segment off portions and adopt specialized recipes 😞
- Alternative is to lock into a specific LCF's technology choice 😞
 - This may be ok for a specialized near-realtime processing pipeline (supernova)
 - Want to remain flexible for more generalized processing

Accelerated Algorithms

- R&D Priorities remain:
 - Initial stages of raw data filtering, DSP and hit finding/clustering (a.k.a. Wirecell toolkits)
 - Higher level track-finding (with strong bias towards ML techniques)
 - Calibration workflows [this is an odd piece that may or may not be at issue]
- These stages provide the major data reduction which make the DUNE problem very tractable
- Later stage ML/AI classification
 - It's not clear how much of the “raw-ish” information get's reintroduced
 - IO bandwidth challenge w.r.t. feeding models*

Data Representations

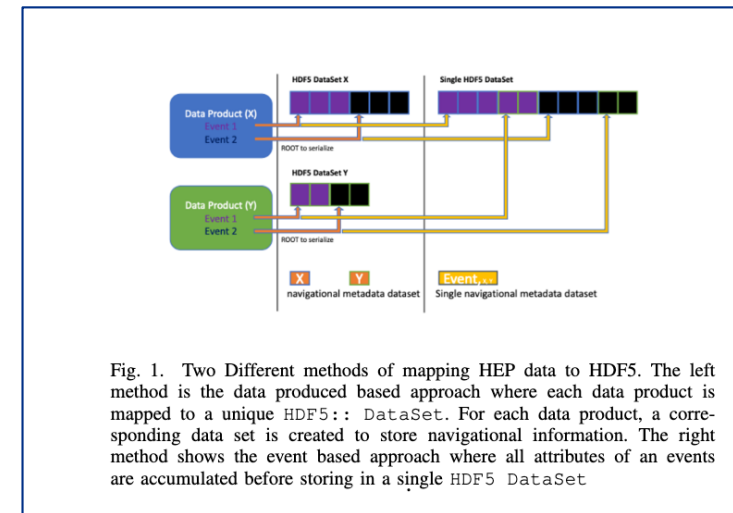
- Statement:
DUNE will use multiple data representation
- Currently we support ROOT and HDF5
 - HDF5 is an ingest format from the DAQ
 - HDF5 is an is an output format for analysis NTuples
 - ROOT serves as the mainline processing chain format
- We have a mandate to explore other data formats
 - **RNTuple**
- Motivation to understand other data representations or translation tools
 - Driven by ML toolkits

Data Representation

- Compression is a topic in DUNE
- Baseline is that we need loss-less compression at the detector levels due to the way charge is shared across wires/planes and ROIs are formed
 - Need to preserve sensitivity to small energy depositions (~few MeV scale interactions)
- Compression on reconstructed quantities is interesting
 - Not clear which types of compression we benefit from the most
 - Not clear how much gain (small number of events)

Work towards Data Representation

- HEP-CCE developed a test framework for parallel I/O (specifically HDF5)
 - This allows us to take a HEP/DUNE data model and evaluate it
 - We need more of this type of investigation and expansion into other data representations (RNTuple etc...)
- HEP-CCE studied various data model designs, and optimized and determined scaling
 - We need more of this but for other formats and run on other machines (i.e. beyond CORI and early Perlmutter)
- GPU Friendly data model
 - Work in progress (that needs to continue). Goal is to develop [automated] translation and offload to GPU which can handle the C++ structured data “flattened” for GPU
 - We need more of this. This is the glue we need for DSP and potentially simplifies AI/ML workflows



Other things to know

- There are different types of core analyses
 - Oscillations, Cross sections, Searches
- The oscillation analyses require significant “fitting” after main processing and selection
 - Compute for the fits [in current neutrino experiments] dominates all prior compute in the chain
 - Example: 3-Flavor fits (2021 NOvA) took
 - 54M core hours on HPC (fits)
 - ~26M core hours on grid (2-years) Data processing, reco, calibration, signal select
 - Scales w/ sensitivity and complexity systematics (need MORE for DUNE)
- Accelerated/Portable fitters are a priority

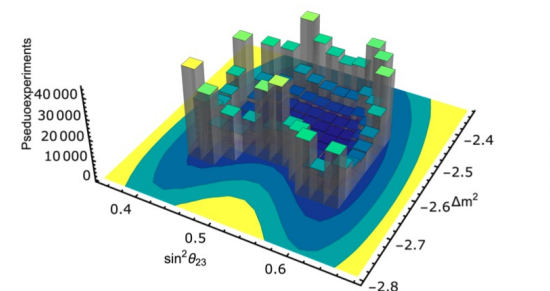


FIG. 2: Example of construction of an allowed region in the relevant parameter space and the pseudoexperiments generated for each section of that space. Further details on how the number of pseudoexperiments are determined are found in Section IV B 2.

Asides

Neutrino Data Filtering & Histogramming

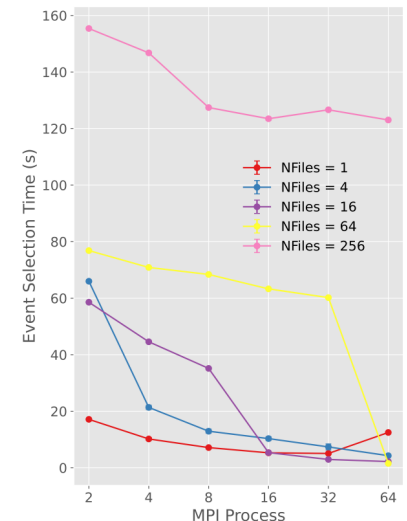
	Grid Computing (current)	HPC (Future)
Language	C++	Python
Data Format	Nested Tuples/ROOT [across $O(100k)$ files]	Multi-Index Tables HDF5
Data Processing	Event Loop	Columnar
Scalability	Limited by # of discrete files	Strong Scaling (demonstrated to $O(10^6)$ ranks)

HDF5 File Format Support

- ROOT serialization is the default in our current frameworks, but has limitations in both scalability and access speed.
- DUNE and NOvA have both added support for HDF5 data formats (and this looks to be the future)
- **Advantage is organizational & parallel I/O access**
- **Effectively allows columnar analysis patterns w/ strong scaling behavior**
- **Ideal for end stage datasets and dataset hosting**

NOvA HDF5

- Easy to use Python interface
- Leverages highly optimized MPI implementations at HPC
- Industry buy-in
- HDF5 chunked IO enables us to scale file size with little overhead
- Demonstrate filter-time scaling and **flexibility in job configuration**
- Monolithic file provides most benefits
 - Consistent 10x speedup
 - 11x compression
 - Self-contained
 - Cheaply concatenated – 3TB using only 180 CPU hours (Cori-Haswell)



SC22 | Dallas, TX | hpc accelerates.

D. Doyle | Analyzing NOvA Neutrino Data with the Perlmutter Supercomputer

11/16/2022

13

Pythonic Analysis

- Move to HDF5 datasets also allow for pythonic analysis tool suites to be leveraged.
- PandAna is a next-generation CAFAna prototype
 - Combines Pandas dataframes with data parallel HDF5 access and MPI compute/reduction distribution
 - Includes similar accounting and re-weighting/re-normalization functionality for neutrino analysis
 - Exhibits near perfect strong scaling on realistic neutrino cross section analysis

PandAna and Llama

- Empower high energy physics analyzers with HPC
- Written in python for ease-of-use
- Implicit parallelism

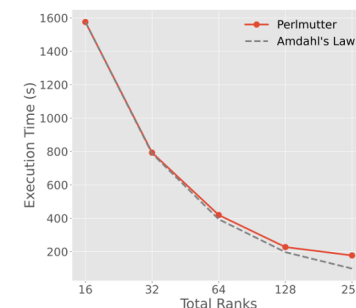
PandAna:

- Data filtering and transformation with pandas
- Intelligent distribution of table rows among ranks

Llama:

- Histogramming package built upon boost_histogram
- Lazy aggregation of results based on arithmetic properties of operations

Datasets	Beam Spills (million)	Size (GB)
Dataset 0	104	432
Dataset 1	25	74
Dataset 2	25	74
Dataset 3	25	76



Workflow achieves near-perfect scaling on Perlmutter with realistic analysis!

Turnaround in minutes enables interactive physics



SC22 | Dallas, TX | hpc accelerates.

D. Doyle | Analyzing NOvA Neutrino Data with the Perlmutter Supercomputer

11/16/2022

14

