Argonne NATIONAL LABORATORY

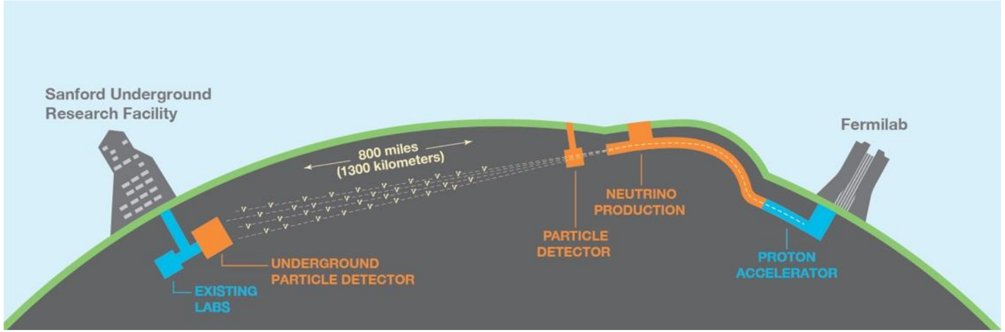
DUNE PRODUCTIONS USING ANL RESOURCES

ALEENA RAFIQUE
Assistant Physicist
HEP Division
Argonne National Laboratory

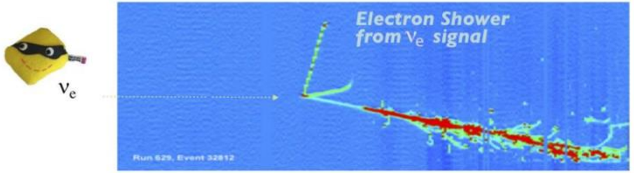
12/19/2024

DUNE

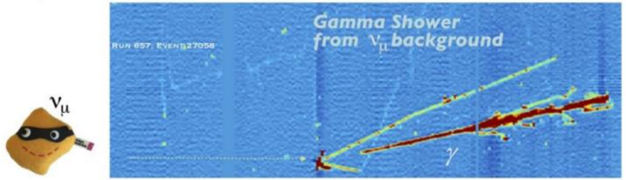
- Deep Underground Neutrino Experiment (DUNE) consists of a near detector (ND) at Fermilab, a far detector (FD) at SURF, South Dakota, and a beam travelling 800 miles underground from near to far detector.
- Primary goals of DUNE
  - Neutrino oscillations (including CP-violation measurement), supernova detections etc
- DUNE uses LArTPC detectors that produce large data records based upon drift time and mm resolution
- Detector begins data-taking in 2029. Prototypes are collecting data already.

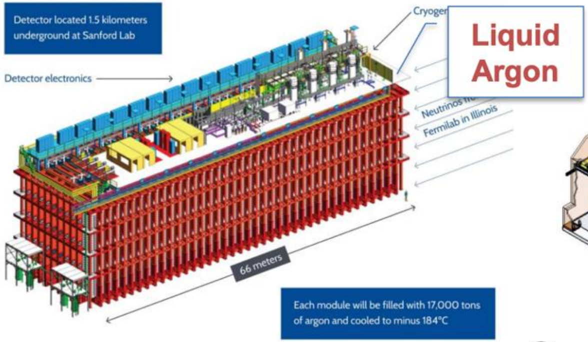


DUNE COMPUTING NEEDS

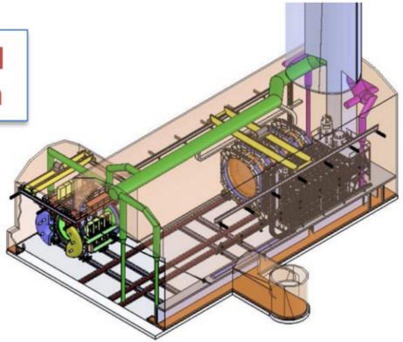


Am I a ν_μ or a ν_e





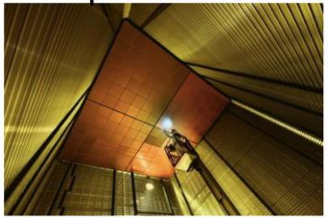
4 Far detectors



On and off axis near detectors

np04 test

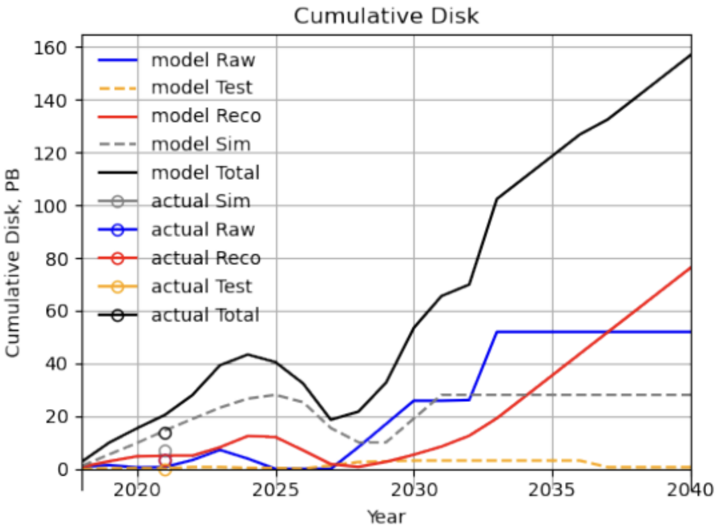
np02 test



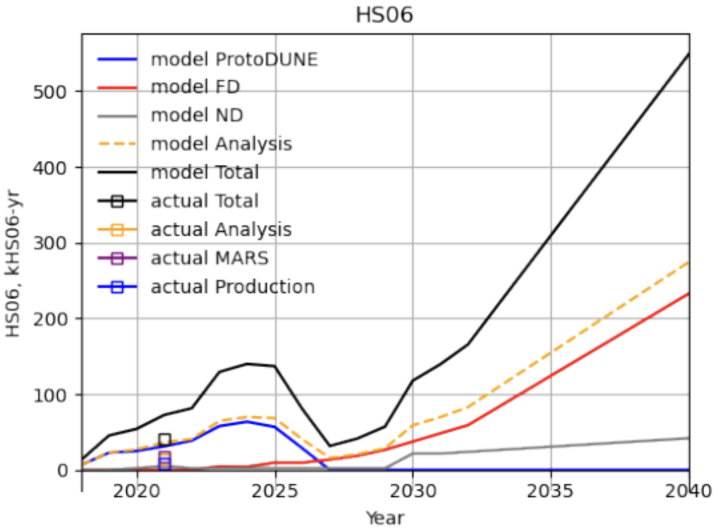
Each 10-20 m^2 TPC readout plane has ~2500 channels and reads out ~6000 14-bit samples.
Each far detector module has ~150 planes

DUNE PROJECTED COMPUTING NEEDS

DUNE needs are at ~10% of CMS



Estimated size of various disk samples in PB for DUNE. The points show actual use in 2021 which was lower than planned due to delays in distributing second copies of samples to remote sites.



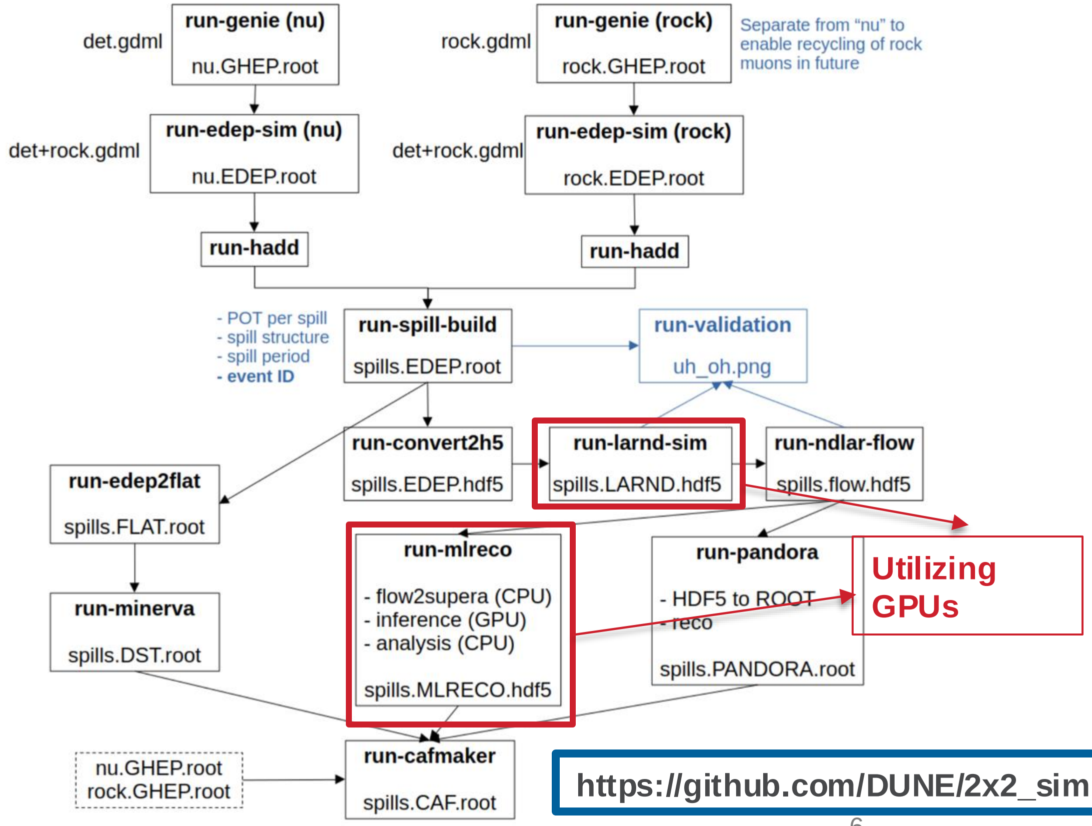
Estimated CPU needs for various samples. The units are kHS06-years (2020 vintage CPUs are ~11 HS06 [81]) assuming 70% efficiency. CPU utilization in 2021 was lower than the model due to the absence of a yearly reconstruction pass.

DUNE computing CDR: https://arxiv.org/pdf/2210.15665

ALCF FOR DUNE

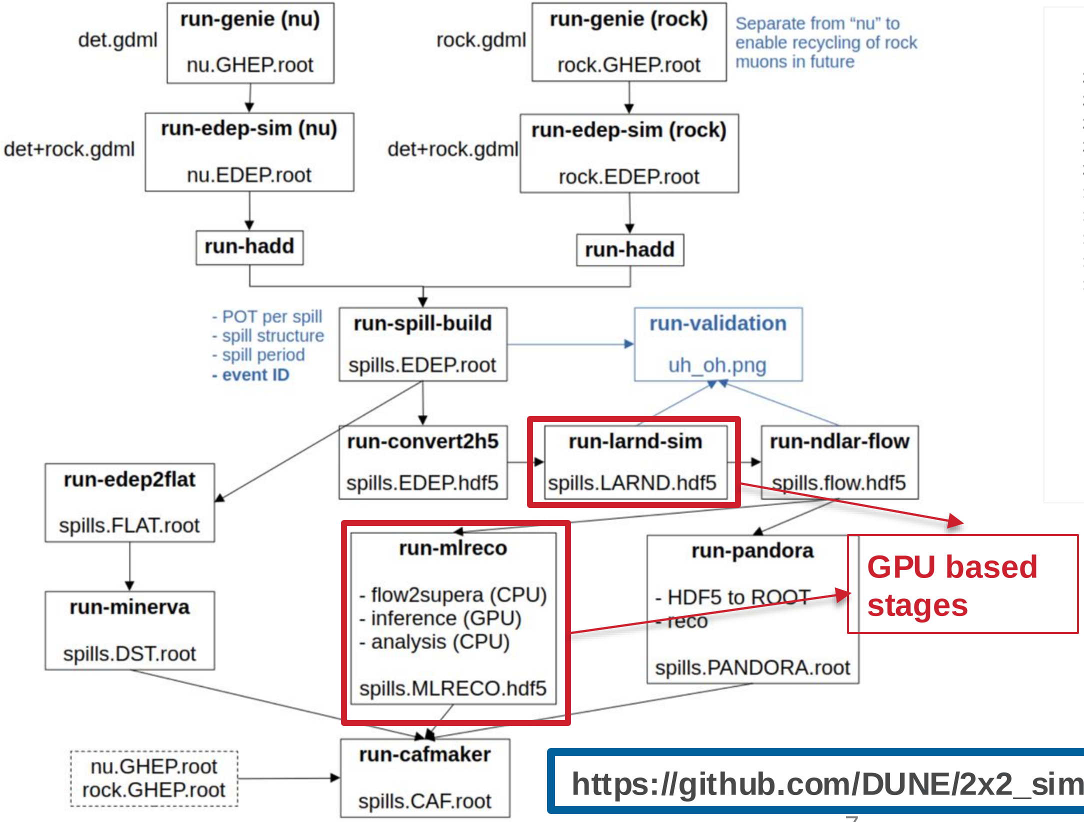
- Started the efforts in 2023 under an internal ANL-award with an intention to have long-term ANL involvement in DUNE computing using future machines (Aurora and beyond)
- Started with ND production R&D for prototype ND-LAr 2x2 situated at Fermilab
- We started R&D with LCRC Bebop and Swing, moved to Theta/ThetaGPU and now using Polaris for the official productions (with allocation via DD award)
  - The productions are being developed and run officially at NERSC Perlmutter
- With successful production run at ANL, we stepped into official production of systematic samples that are being used by all collaborators.
- Started official productions using ANL resources since June 2024.

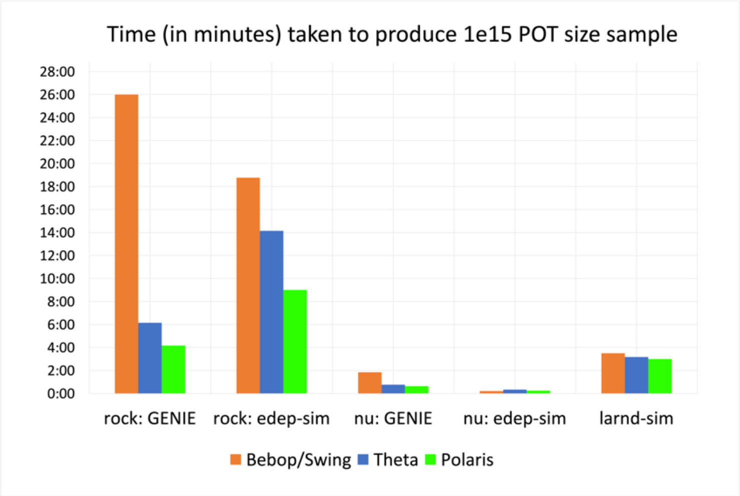
ND PRODUCTION CHAIN OVERVIEW



https://github.com/DUNE/2x2_sim

ND PRODUCTION CHAIN OVERVIEW

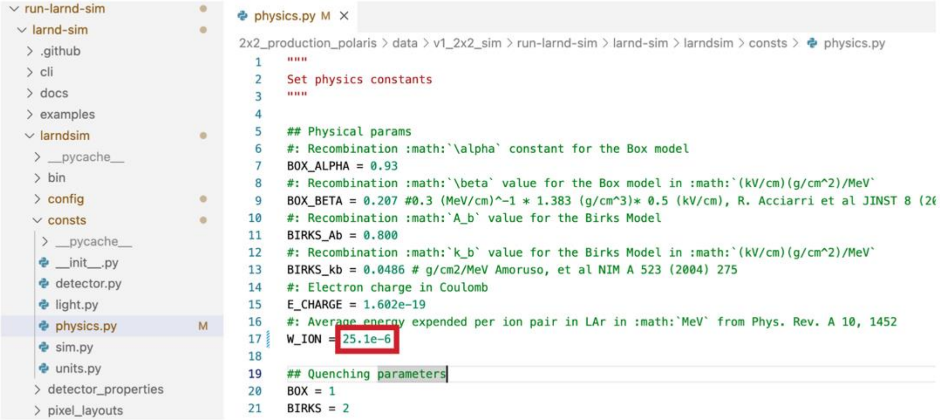




https://github.com/DUNE/2x2_sim

SYSTEMATICALLY VARIED SAMPLE PRODUCTION

- Installation:
  - larndsim stage:
    - Carry out other installation steps until cloning the larndsim repo using GitHub
    - change the parameter value manually in const/physics.py of larndsim
    - built the code in the larndsim venv of 2x2_sim (larnd.venv), so that when larndsim code is sourced for the runs it points to the changed code.
  - ndlar flow and mlreco stages: installation as usual with the script provided
- Transfer the output files to Fermilab and NERSC machines



SETTING UP 2X2_SIM FOR VARIED SAMPLES

yaml config files for each step with the environment variables that need the be set up for the jobs

Larnd-sim stage

```yaml
base_envs:
  - name: 'MiniRun6_1E18_RHC_Birks_k_1sig_down_Systematics.larnd'
    env:
      ARCUBE_OUTDIR_BASE: '/grand/ALCF_for_DUNE/users/aleena/systematics/MiniRun6_1E18_RHC/MiniRun6_1E18_RHC_Birks_k_1sig_down_Systematics/out'
      ARCUBE_LOGDIR_BASE: '/grand/ALCF_for_DUNE/users/aleena/systematics/MiniRun6_1E18_RHC/MiniRun6_1E18_RHC_Birks_k_1sig_down_Systematics/log'
      ARCUBE_CONTAINER: 'sim2x2_ndlar011.sif'
      ARCUBE_CONVERT2H5_NAME: 'MiniRun5_1E19_RHC.convert2h5'
      ARCUBE_LARNDSIM_CONFIG: '2x2_old_response'
      ARCUBE_DIR: '/home/aleena/2x2_sim_111424/2x2_sim'
      ARCUBE_CONTAINER_DIR: '/grand/ALCF_for_DUNE/2x2_sim_data/containers'
      ARCUBE_USE_LOCAL_PRODUCT: '1'
      ARCUBE_OUT_NAME: 'MiniRun6_Systematics_Birks_k_1sig_down.larnd'
      ARCUBE_RUNTIME:  'NONE'
```

MLreco spine stage

```yaml
base_envs:
  - name: 'MiniRun6_1E18_RHC_Birks_k_1sig_down_Systematics_stage2.spine'
    env:
      ARCUBE_RUNTIME: 'SINGULARITY'
      ARCUBE_CONTAINER: 'larcv2_ub20.04-cuda11.6-pytorch1.13-larndsim.sif'
      ARCUBE_OUTDIR_BASE: '/grand/ALCF_for_DUNE/users/aleena/systematics/MiniRun6_1E18_RHC/MiniRun6_1E18_RHC_Birks_k_1sig_down_Systematics/out'
      ARCUBE_LOGDIR_BASE: '/grand/ALCF_for_DUNE/users/aleena/systematics/MiniRun6_1E18_RHC/MiniRun6_1E18_RHC_Birks_k_1sig_down_Systematics/log'
      #SINGULARITY_BIND: $ARCUBE_DIR,/grand/ALCF_for_DUNE:/grand/ALCF_for_DUNE
      ARCUBE_IN_NAME: 'MiniRun6_Systematics_Birks_k_1sig_down.flow2supera'
      ARCUBE_DIR: '/home/aleena/2x2_sim_111424/2x2_sim'
      ARCUBE_CONTAINER_DIR: '/grand/ALCF_for_DUNE/2x2_sim_data/containers'
      ARCUBE_USE_LOCAL_PRODUCT: '1'
      ARCUBE_OUT_NAME: 'MiniRun6_Systematics_Birks_k_1sig_down.spine'
```

SYSTEMATIC PRODUCTION PROJECT NEEDS

- We have been tasked to produce ~30 1E18 protons-on-target samples initially
- Each of these samples take ~200 node hours (without the event generation stages)
- Each of these samples take ~0.5 TB storage space including outputs from various stages
- Currently DD allocation with 6000 node hours and 2TB storage space
  - Planning to apply for INCITE in the future

WORKFLOWS

- Project name: ALCF_for_DUNE, Machine: ALCF Polaris
- Workflow manager used is Balsam (https://docs.alcf.anl.gov/polaris/workflows/balsam/)
- Balsam workflow code and scripts maintained at https://github.com/DUNE/ALCF_for_DUNE/tree/main/Balsam/2x2_production_polaris
- Workflow referred from fireworks4dune at NERSC
- Balsam script prepared creates individual jobs for each ARCUBE_INDEX (0 to 128 for 1E18 POT) for each step (larnd-sim, ndlarflow, flow2supera, spine) by sourcing the yaml environment configs shown earlier and setting ARCUBE_INDEX. The production chain is divided into two stages 1: (larnd-sim, ndlarflow), 2: (flow2supera, spine)
- ndlarflow configured to wait for parent larnd-sim job with the same ARCUBE_INDEX, similar spine waits for parent flow2supera job to finish before running.
- The script then dispatches these individual jobs to ALCF queue by packaging together 32 ARCUBE_INDEX jobs together (with wall time estimate = 60 minutes and requesting 32 nodes), therefore, a total of 4 resulting jobs on the ALCF queue.
- Each compute node runs one larnd job at a time, as we are seeing that GPU memory exceeds and job fails when 2 or more larnd job run at the same time on a compute node
- larnd-sim error: cupy.cuda.memory.OutOfMemoryError occurs for ~25% larnd-sim jobs.

PRODUCTION REPORT

- Detailed report at: https://github.com/DUNE/ALCF_for_DUNE/blob/main/Balsam/2x2_production_polaris/usage_summary
- Recent node hours consumption: ~200 hours per sample
- Job status based on ARCUBE_INDEX:

Stage	Total job run	Completed jobs	Failed jobs
run-larnd-sim	128	95	33
run-ndlar-flow	95	95	0
run-flow2supera	95	95	0
run-spine	95	95	0
run-caf	95	0	95

- CAF stage gives error: “Failed to initialize loader socket”
  - Uses cvmfsexec package, only get error when submit jobs to the prod queue
  - Working on resolving the issue

SUMMARY

- We are producing official systematic sample production for DUNE-ND 2x2
- Need to work on resolving the errors for failed jobs
- Need to find a solution for CAF stage to work on Polaris in order to complete the simulation chain
- Available balance node hours on Polaris for the project: 3,608.7 node hours
- The next varied sample is currently in the queue for running
- Will apply for a larger allocation on Polaris in the near future. Also, plan is to perform R&D for running DUNE jobs on Aurora.

BACKUP SLIDES

AWARD ALLOCATIONS ON ALCF FOR DUNE

- LCRC
  - Quarterly allocations. Most people perform R&D then pivot to ALCF
  - **Bebop**: ~100k core hours since May, 2023. Renewed multiple times successfully
  - **Swing**: 500 gpu hours for R&D.
- ALCF:
  - Renews every 6 months under Director Discretionary award allocation
  - ~~**Theta**: 32k node hours~~
  - **ThetaGPU**: 200 node hours for R&D
  - **Polaris**: current 6k node hours.
- Plan is to submit an INCITE proposal (1-3 year long, renewable) on Aurora to get the continued support for DUNE

Physics → Computing Challenges

Fine segmentation needed for electron-photon discrimination:

Sub-cm-level segmentation over very large volumes drives the number of channels and data volumes.
4 x 400K channels x 6000 14-bit samples

Low-energy (< 10 MeV) thresholds for astrophysical neutrinos:

The need to optimize the low-energy threshold drives our need to carefully record waveforms with minimal processing and thus drastically increases the raw data volume.

Precise energy calibrations:

The large volumes, complex E field configurations, liquid motion, and potential variations in electron lifetime and drift velocity make it necessary to have large calibration data samples that span the full FD detector volumes. Large cosmic ray and artificial calibration samples will dominate the total data volumes from the FD.

Supernovae:

A supernova neutrino burst candidate will generate 320 TB of (uncompressed) data across the first two modules, resulting in thousands of data files produced over a 100 s period. Supernova physics drives the need for fast data transmission from the FD to computing facilities The drastically different time scale of SNB physics also places requirements on the software framework.

Near Detector integration:

Integration of disparate detector technologies into a coherent whole. New pixel LArTPC geometry.

Analysis and parameter extraction:

Neutrino interaction samples are generally simpler than event records at colliders. However, final parameter extraction using large numbers of nuisance parameters (and or ML) is still a computationally intense problem and will likely require efficient utilization of HPC resources.

DEEP UNDERGROUND NEUTRINO EXPERIMENT

https://indico.cern.ch/event/1203733/contributions/5110141/attachments/2542055/4376827/DUNE_Computing-2022-11-07.pdf