(proto)DUNE computing update

Ken Herner (with material from Andrew Norman and Heidi Schellman)
WLCG GDB
17 October 2018
Reminder: Plan is to put a future huge LAr detector “DUNE” in the Homestake Gold Mine
Underground Laboratory

DUNE Far Detector site

- Sanford Underground Research Facility (SURF), South Dakota
- Four caverns on 4850 level (~ 1 mile underground)
Full DUNE is 4 of these – filled with liquid Argon

**External Dimensions:** 19.1m x 18.0m x 66.0m
protoDUNE: the here and now
protoDUNE: Inside EHN1

dual-phase

single-phase
The computing problem

3 hrs of raw data transfers from CERN->FNAL 5 Oct

- ProtoDUNE can write up to 2-3 GB/sec of data when running at 25 Hz beam.
- Expect 2-5 PB of raw data from the 6 weeks of running this fall + more from additional cosmic running + ProtoDUNE-DP
- Big DUNE will write 10-30 PB/year from the far detectors and probably more than that from the near detector.
- Instantaneous rates are already >= CMS/ATLAS.
- Integrated rates in 2026 will be where ATLAS/CMS are now
**What is the production group doing?**

Two main efforts right now: protodDUNE data keepup processing (first priority) and MC production

Keepup: run raw data decoding and hit reconstruction (not tracking yet; will be added soon).

**protoDUNE SP by the numbers**

*as of 17 Oct 2018*

- Raw data: 1298 TB
- Raw “physics” data: 471 TB
- Raw physics data fraction processed: 73.6% (complete through Oct. 10)
- Total reconstruction output size: 186 TB
Raw data

Raw ADC for run 4696 event 103 TPC plane 0z (APA 3: US-RaS)
protoDUNE SP Keepup Processing

Processing all kinds of runs (test, noise, physics)

Able to get lots of jobs running, but reconstruction takes a while for physics runs

DUNE uses the larsoft suite (http://larsoft.org/) for reconstruction and analysis. Common art-based framework for LArTPC detectors (e.g. Microboone, ICARUS). art is a CMSSW fork.
Factor of ~2 difference in CPU speed between oldest and newest FNAL WNs.

Run time in hours by site.
Memory usage (bins of 500MB)
No tracking yet
POMS is a FNAL-designed workflow management system; evaluating it and others for long-term needs

Initial keepup submissions are 3500MB memory and 36h run time
Production keepup processing - POMS setup

Automated Recovery job launches

Recovery jobs submitted with 48h run time and 5000 MB mem

Initial submissions are 250 jobs at a time every 30 minutes (tunable)

Current status: https://pomsgpvm01.fnal.gov/poms/campaign_stage_info?campaign_stage_id=1775
Keepup outputs

Keepup runs the raw data decoder and reconstruction through hit finding (including photon detector bits)
Saves outputs to separate data tiers (decoded-raw and hit-reconstructed) in SAM (not using Rucio yet; highly likely in the future)
Analyzers can start from these files for tracking studies; no need to run expensive hit-finding many times
Do not save the raw data branches in outputs (saves a factor of ~2 in space)
Going Global with User Jobs

- International collaborators often bring additional computing resources to bear
- Users want to be able to seamlessly run at all sites with unified submission command.
- Following OSG setup process makes it easy to have sites around the globe communicate with a common interface, with a variety of job management systems underneath.
- Integration times as short as 1-2 weeks; all accessible via standard submission tools. **Record set-up time is just 2 hours!**
- DUNE requires a few CMVFS repositories: dune.openesciencegrid.org, fermilab.openesciencegrid.org, larsoft.openesciencegrid.org
- Local batch system (HTCondor, CREAM, ARC, etc.) does not really matter. **Use glideinWMS pilots (same as CMS) to run the user jobs**
Keepup dataflow

- **EHN1**
  - Worker Nodes
    - inbound: xrootd
    - outbound: gridftp
  - Managed by F-FTS

- **CERN EOS**
  - RAW DATA
  - CERN only

- **FNAL**
  - dCache/Enstore
  - Other sites
  - RAW DATA
  - KEEPUP OUTPUT

Keepup launches via POMS on a crontab. Launches occur after raw data show up in SAM. SAM metadata for outputs declared within the job.
protoDUNE Cross Labs Monitoring

- protoDUNE detectors now running
  - Significant computing infrastructure at both CERN and FNAL (buffer disk, long-term storage, reconstruction and analysis resources)
- Developed a unified monitoring system to show data from both laboratories side-by-side:
  - Covers data transfer rates, storage health and usage, job submission and status
Central MC Production

Also running MC production (most recent campaign is MCC11)
Use POMS to set up infrastructure
Run samples once centrally; distribute outputs to entire collaboration

MCC11 jobs only, July 23 to October 8 (right: no FNAL, no CERN)
Taking the next steps

Low-hanging fruit in the short term:

1) Increase job distribution: move all sites to multicore setup (tracking can use multiple cores), higher memory limits and longer pilot maximum run times. (DP MC may come in at some point too).

2) Integrate site SEs starting with UK sites:
   a) stream inputs locally (or from CERN)
   b) copy outputs locally, set up some FTS back to FNAL

Longer term:
Develop computing consortium
Evaluate technologies and make long-term choices (data management, workflow management, software, etc.)
Adapt to changing infrastructure landscape
Future production dataflow?

Worker Nodes
inbound: xrootd
outbound: ?

EHN1

Site SEs

CERN EOS

FNAL
dCache/Enstore

RAW DATA
KEEPUP OUTPUT

Not all paths may be here for all workflows: determined by replication/retention policies

All SE to SE copies managed by some form of FTS
Long-term: The Consortium Plan

- Forming a DUNE consortium of interested parties to work on DUNE strategy, assemble resources and make technical decisions
- Where it works, collaborate with the broader HEP and Astrophysics communities on common tools and infrastructure
  - Grid jobs
  - Data management (Rucio!)
  - Authentication!
  - New coding methodologies
The Goal(s)

- Record, reconstruct and analyze the ProtoDUNE data ASAP
- Prepare a CDR for full DUNE (x100 in size) which starts taking data in 2026
- Work with other experiments SKA/LHC/HyperK/LSST/SBND to have robust, global shared strategies
Current status of Consortium

- Many institutions are already contributing resources!!!!
- Setting up a directory of institutional contacts and resources
- Setting up a formal structure for technical decision-making
- Setting up a formal structure for resource contributions
  - Hardware (cores/GB)
  - Operations (people to keep it all running)
Three-fold way

Facilities
- National infrastructure
- Funding agencies

Technical
- DUNE standards
- Common tools
- New architectures

Operations
- Lab support
- Common costs
- Collaborators

Many lessons to be learned from WLCG
Long term: software

Frameworks, reconstruction/tracking techniques, data management
Deep learning will play a critical role (already does) in particle ID and analysis

Works to separate out individual tracks and showers.
Allows better energy estimators
Need combo GPUs/CPUs at large scales
Long term: Infrastructure

DUNE will need resources of all types (HTC CPU, GPU, HPC)

Clear mandate from US DOE to make use of HPC and LCF machines

Art v3 is a step in the right direction (multiple events in separate threads)

Now is the time to re-think how we do things to fully leverage new resources

DUNE is still something of a green field in this regard!
Summary

Production group has been very busy the past few months with both protoDUNE data and MC processing.

Short-term modifications to improve throughput include modifications to the gWMS factory entries for some sites and integration of site storage elements

Currently running with beam (next week and one more week after that) and then another big MC push expected for the TDR in spring 2019. Plenty of work to do in the next 6 months!

WLCG offers many lessons on how to do international computing at scale
Production MC Workflow Submission Example

Test workflow for SBND

- Each stage of the workflow takes a SAM dataset input with the output files of the previous stage. Often run gen+g4 together, or even gen+g4+detsim
- Stages can have different job resource requirements
- Each stage starts automatically when the previous one is finished
- Usually requesting 2 cores for reco stage
Auxiliary File Delivery via OSG's StashCache

Example of the problem: several neutrino experiments use O(100 MB) "flux files" for simulating neutrinos in rock, etc.

- Several files are used per simulation job
- Total input is few GBs
- Not well-suited to CVMFS for several reasons (each job uses random subset; can thrash worker node cache)
- Transferring all the time sub-optimal (plus no caching at all that way)

StashCache service developed and widely used by OSG to solve just such a problem:

- Files replicated to regional caches, streamed via transparent xrootd connection from closest cache (geolIP) if not cached locally (small cache on the machine itself)
- Can be layered on top of a dedicated CVMFS repository
- User sees a consistent POSIX file path across all sites, no need to worry about "how" the file gets there

Four FNAL experiments are using the service so far:

FIFE assists experiments with setting up and populating area.
StashCache (2)

More info:

https://opensciencegrid.org/docs/data/stashcache/overview/
Off-line Production Shifter Dashboards

- Customizable dashboards for each experiment
- Display information experiments deem relevant to shifters in a single place
- Could drill down to more details
Efficient User Code Distribution

• Until January 2018 analysers were able to access their Bluearc code dev area
  – Was NFS mounted on all FermiGrid nodes.
• After Bluearc has been unmounted analyzers have started to use dCache
  scratch area to upload tarballs and stage them from dCache to worker nodes.
  It turned out to be a bad idea:
  – too much hammering on single pools when many jobs of the same type start
  – can completely block access to files owned by other users on a given pool
• Interim solution: stage tarballs in “resilient” dCache area (files replicated 20x
  on different pools):
  – Advantages: keeps the system running, doesn’t block users, more efficient.
  – Disadvantages: tarballs use 20x space; need careful cleanup policies
• Longer-term solution: looking into automated distribution via CVMFS.
  – Maybe should wait for CVMFS release that allows parallel publishing (should
    be released by the end of 2018)
Data management: FIFE SAM and FTS

SAM originally developed for CDF and D0; many FNAL experiments now using it
• A File metadata/provenance catalog
• A File replica catalog (data need not be at Fermilab)
• Allows metadata query-based “dataset” creation
• An optimized file delivery system (command-line, C++, Python APIs available)
• PostgreSQL backend; client communication via http: eliminates need to worry about opening ports for communication with server in nearly all cases
– Heretofore mostly followed the "send the data to the jobs" model.
  Enhancements underway to also make allowances for the "jobs to the data" school of thought
Simplifying I/O with IFDH

• File I/O is a complex problem (Best place to read? What protocol? Best place to send output?)
  • FIFE cannot force opportunistic sites to install specific protocols
• Intensity Frontier Data Handling client developed as common wrapper around standard data movement tools; shield user from site-specific requirements and choosing transfer protocols
• Nearly a drop-in replacement for cp, rm, etc., but also extensive features to interface with SAM (can use ifdh to pull in files associated with SAM project, etc.)
• Supports a wide variety of protocols (including xrootd); automatically chooses best protocol depending on host machine, source location, and destination (can override if desired)
  – Backend behavior can be changed or new protocols added in completely transparent ways
  – Special logic for automatically parsing Fermilab dCache and CERN EOS paths; shield user from having to know proper URIs
Full workflow management

- Now combining job submission, data management, databases, and monitoring tools into complete workflow management system
  - **Production Operations Management Service (POMS)**
- Can specify user-designed “campaigns” via GUI describing job dependencies, automatic resubmission of failed jobs, complete monitoring and progress tracking in DB
  - Visible in standard job monitoring tools

- Usable for production-level running and user analysis
- REST API for data I/O
- Command line tools for needed operations
- Supports POMS launching jobs, or experimenters launching jobs and using POMS only for tracking