ProtoDUNE Computing Items

Brett Viren and Maxim Potekhin

Physics Department

BROOKHAVEN
NATIONAL LABORATORY

DUNE Collaboration
21 May 2016

Outline

Documents

Data Rates

SP & DP Differences and Symmetries

Technology Choices

New Ideas / Discussion

Documents

DocDB 914 **DUNE Computing Model**, Recently updated.

→ Needs more input, particularly from DP.

- GitHub project page

DocDB 1209 **Requirements for Data Management System**.

DocDB 1212 **ProtoDUNE Data Management System**, final conceptual / early technical design.

protoDUNE TDR git **ProtoDUNE TDR**. In preparation, Computing and Software section (15 pages).

Outline

Documents

Data Rates

SP & DP Differences and Symmetries

Technology Choices

New Ideas / Discussion

Initial SP Data Scenarios

As in DocDB 1086v4, lots of caveats:

- Three scenarios, "low", "high" and "Goldilocks".
- 5M triggers with 100% efficiency and purity.
- No accounting for dedicated cosmic-μ or other non-beam running.
- No zero-suppression (ZS) with 35t-like compression ($2\times$)
- low/Goldi: short readout time (2ms) and not all APAs.

$\rightarrow$ **total data: 75 TB - 1.2 PB**

Some takeaways.

- Even "low" requires parallel I/O or fast disks.
  - **DAQ must be designed to handle parallel writes!**
- "High" finishes 5M events in 2 days, (no accounting for beam reconfiguration time included here).

Improved SP Scenarios

As in DocDB 1086v5, the changes:

- Assume MicroBooNE noise levels or better: 5$\times$ compression.
- Take one cosmic-μ trigger per beam trigger (out of spill).
- 5ms readout time for all scenarios.

$\rightarrow$ total data: 230 TB - 345 TB.

Still uncertain:

- What efficiency and purity can we expect from beam veto?
- How much laser data?
- Improve on 1-cosmic-μ trigger per 1-beam trigger estimate?
- Is 10Hz trigger right? If faster, still only 5M triggers?

Better engagement with and input from **Measurement and Calibration groups** needed.

Contingency for SP Scenarios

What if[1]:

1. make up for delays by raising 10Hz→50Hz trigger rate?
   - But still 5M total triggers: parallel-I/O and buffer size limits.
2. veto is 50% pure so run longer to get 10M triggers?
   - Problem pushed downstream to CERN EOS/CASTOR and FNAL ENSTORE.
3. we see 35t noise levels instead of μB's?
   - Compression goes from $5\times \rightarrow 2\times$, need extra disk I/O and buffer size.

Overall worse cases:

volume 2&3 $\Rightarrow 10\times \rightarrow$ 3.3 PB total raw data.

parallel I/O 1&3 $\Rightarrow 25\times \rightarrow$ 65 parallel writes, sustained

buffer 1&3 $\Rightarrow 25\times \rightarrow$ 850 TB 3-day disk buffer.

[1] Numbers are non-exhaustive, wild-ass guesses.

DP Data Estimates

Numbers from Dario Autiero:

- $\mathcal{O}$(100M) beam triggers + cosmic ray + tests.
- 10$\times$ compression via Huffman coding, no ZS.
- 15 MByte/trigger

$\rightarrow$ 2.4 PB total

= 3.8 Gbps sustained for two months.

Outline

Documents

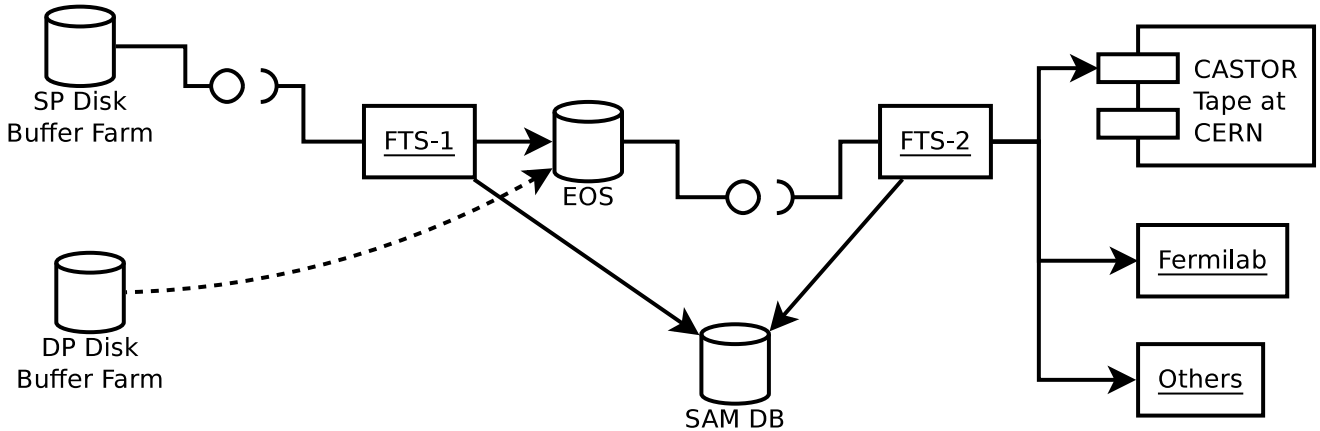
Data Rates

SP & DP Differences and Symmetries

Technology Choices

New Ideas / Discussion

SP & DP strategies: differences



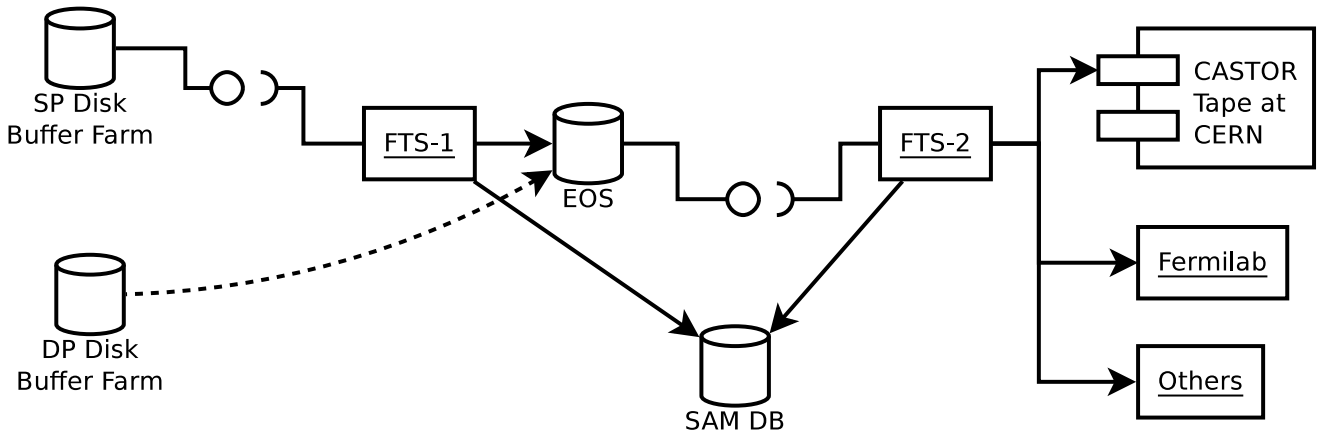
SP:

- **Transfer all raw data** to FNAL for processing.
  - CERN→FNAL bandwidth $>$ 10$\times$ SP→CERN bandwidth.
  - Nominal throughput requires relatively modest disk buffer.
- Promptly process small fraction of data at CERN for QA/QC.
- Use **F-FTS** for buffer→EOS transfer.

DP:

- **Promptly reconstruct all raw data** right next to detector.
  - Requires substantial storage and CPU farm.
- The DP buffer→EOS transfer method is as yet **needs clarity**.

SP & DP strategies: symmetries



Both detectors will:

- **transfer all raw data** to CERN and FNAL tape.
- use **F-FTS** for CERN→FNAL transfers.
- use same **file catalog** (F-FTS ⇒ SAM)

Both detectors might share a common:

- disk buffer technology (if not hardware).
- production workflow management system.

Outline

Documents

Data Rates

SP & DP Differences and Symmetries

Technology Choices

New Ideas / Discussion

Fermilab File Transfer System

Selected F-FTS for Data Management System

- Requirements document
- HSF technical forum query.
- Responses from LHCb/DIRAC and Fermilab/F-FTS.
- Reviewed documents and engaged experts in discussion.
- F-FTS pursued due to existing use in neutrinos @ FNAL, existing and further planned integration with CERN computing.
- Detailed technical questioning over email $\rightarrow$ all positive responses.
- In-person + phone meeting at FNAL with experts[2] and nailed down initial design and document.
- Final conceptual design.

To do: tests between FNAL, CERN, BNL/RACF, LBNL/NERSC

[2] Many thanks to Andrew Norman, Robert Illingworth, Oliver Gutsche

Detector Storage Buffers

- CERN requires 3 day buffer living close to detector.
- DP/SP require different scales, both require parallel I/O.
- Jacques Marteau's R&D results on BeeGFS and XRootD, note expected soon.
- LBNL interested in setting up early server at CERN which can serve as initial test bed for transferring from buffer via F-FTS.
- BNL will evaluate XRootD to receive data for for SP buffer.

$\rightarrow$ **But wait, some breaking new ideas coming up.**

Other Technological Decisions / Tasks

- Monitoring for data transfer and data QA/QC:
  - "Offline Data Monitor" (LBNL/Daya Bay)
- Workflow management system for production processing, exploitation of CPU outside of FNAL+CERN (EU Grid, OSG)
- Data federation for fast and efficient access to data by all collaborating institutions.
- Various offline related items on Giles Barr's recent list to `dune-fd-daq@fnal.gov`.

Outline

Documents

Data Rates

SP & DP Differences and Symmetries

Technology Choices

New Ideas / Discussion

Let's Unify DAQ End Nodes with Buffer Nodes

Hallway discussion with Giles:

- XRootD provides a nice conceptual separation between distributed artDAQ and buffer farm.
- **But** it makes the system more complicated
  - DAQ has to learn about XRootD
  - DAQ is already distributed, then why XRootD for dist. I/O?
    - (well, the "interleave problem" but see next)
- So, let's put all this on each buffer node:
  - artDAQ endpoint output module
  - F-FTS ingest dropbox
  - Some JBOD disks (not RAID)

The "Interleave Problem"

- Users want sequential triggers for convenience.
- Distributed DAQ builds events asynchronously.
- A DAQ event builder node does not receive sequential triggers.
- For DAQ to provide sequential triggers requires it to concentrate all stream into one node with $\sim$1 file's worth of data $\times$ N output streams.
- This limits scalability, increases cost.

The "Interleave Opportunity"

Let's manage our raw data like this:

1. Let DAQ write interleaved events.
2. Write to local disk buffer which also has F-FTS inbox.
3. Save this to permanent archive.
4. Preprocess this data to:
    1. Read N interleaved streams.
    2. Deinterlace.
    3. Apply software noise removal.
    4. Apply detector response deconvolution.
    5. Apply thresholding
    6. Write 1 stream
5. Production reco processing.

The point is to "fix" interleaving and also produce much smaller input files to the production processing.