Computing Model

Amir Farbin (UTA)

Tom Junk (FNAL), Maxim Potekhin (BNL), Craig Tull (LBNL), Brett Viren (BNL)

Status/Plans

- Great deal of work done in support of DUNE CDR and protoDUNE LOI.
    - Snapshot of best understanding of computing model will probably be necessary for future reviews and docs…
- Tom and I asked Maxim to help establish a computing model group. I'll "co-convene" for now to help out Maxim and become more immersed in the details.
    - We should think of an appropriate charge, but among the goals is to commission/track studies and collect results to provide best resource estimates for making decisions and documents.
    - Planning short informal/working weekly meeting (tentatively Tuesday 1 pm) to discuss progress, planning, issues. Everyone is welcome…
    - As for many groups in S&C, the conveners are the primary contributors:
        - So we'll start with periodic updates in for example the "Offline Infrastructure" (working name?) WG.
        - Expect such efforts to gradually evolve towards bigger groups with regular formal WG meetings.
- As far as I know (which isn't much), the primary contributors to the Computing Model have been:
    - Tom Junk (FNAL), Maxim Potekhin (BNL), Craig Tull (LBNL), Brett Viren (BNL)
        - Reliant on input from various sources through out collaboration…
    - Please write me and Maxim and remind of us any work/thinking you may have done or plan to do in this area.

Computing Model

- A lot of overlap and coupling to DAQ/Trigger… should think if/what are the boundaries between DAQ and computing.
    - Perhaps this group can take charge of collecting the info necessary for making DAQ and computing decisions.
- I can identify at least three (hierarchical?) levels of “detail”
    - *Resource Estimates*- Storage, Processing, Network requirements for various scenarios given particular assumptions.
    - *Model Design*- Conceptualization of the data processing flow(s).
        - physics/calibration data/work-flows, including streams, data tiers, databases, and mapping to physical resources… Include simulation…
        - consider contingencies (e.g. 2x noise) and policies (number of replicas, data access policies, …)
        - e.g. hypothetical scenario: 35t Raw data will be filtered near-line into streams, periodically reconstructed at FNAL, and reco summary distributed to 3 sites for easy access.
    - *Model Implementation*- mapping the design to hardware, software, services, etc…
- Great deal of uncertainty at each level: inputs, evolving model, and evolving technology.
- Exercise must be repeated for each detector (35t, protoDUNE, DUNE)

Resource Estimates Inputs

- *Detector*: N wires/channels, sample rate, bytes/sample, drift time, drifts/readout, electronic noise…
- *Backgrounds*: Cosmics, Ar-39, …
  - Thresholds… e.g. for SN triggering.
  - Effectiveness of Zero Suppression, Huffman compression, online cosmic or Ar-39 rejection.
- *Signal*: Size of tracks, clusters, …
- *Reconstruction*: # of chains (e.g. different reco for different physics or calibrations), rate, memory, …
- *Data tiers*: RAW, Reconstruction output, Analysis input/output.
- Brett has a package where some of these parameters are collected and compiled into relevant estimates for tables.

Resource Estimate Scenarios

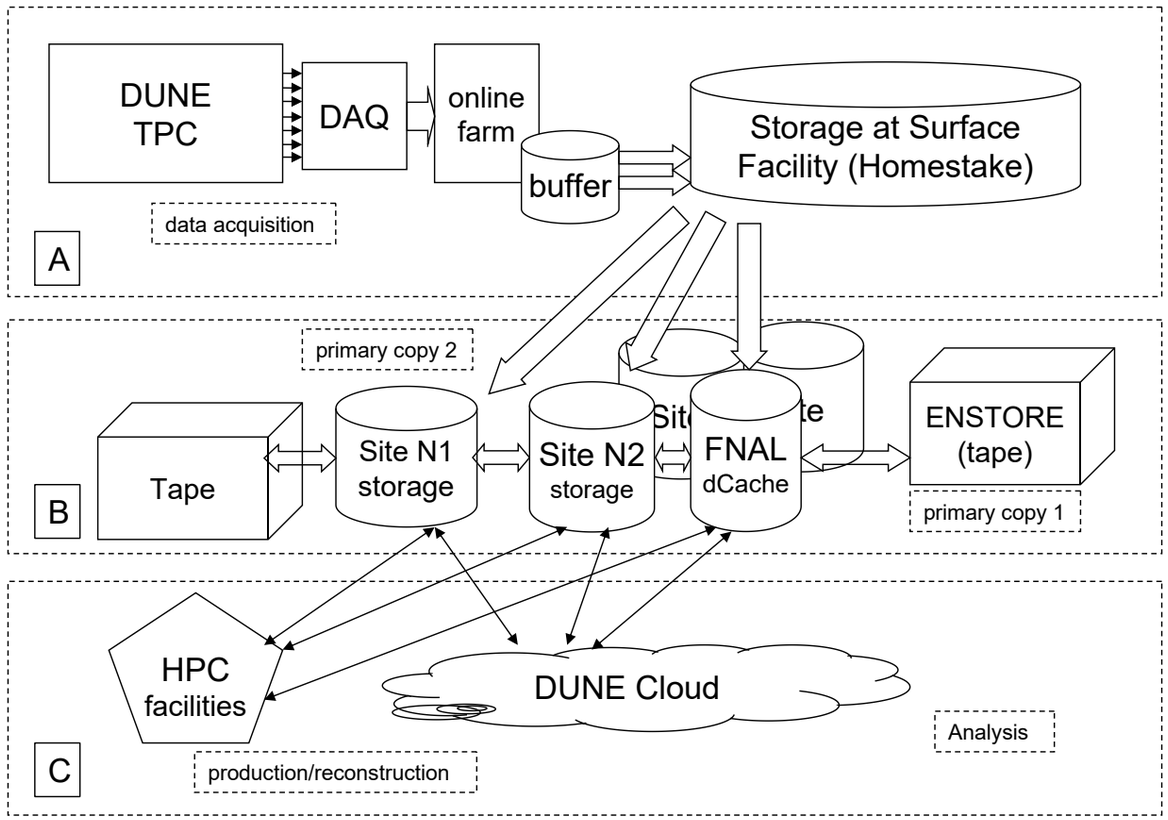
- I suggest we define some “benchmark” scenarios… for example:
  - *Full Data*: (100s of exabytes/year) Continuously read detector without any suppression….
    - What is the ceiling? Help understand the scale of the problem and motivate the design of the DAQ/computing model.
  - *Continuous Zero-suppressed*: (50 pb/year) Continuously read detector, store ZS/compressed only.
    - Represents an upper-bound realistic scenario?
  - *Triggered*: Rely on DAQ/Trigger for optimal data-reduction.
    - No suppression during beam-spill.
    - Identify Supernovae, proton decay, … for “optimal” storage otherwise.
- For each detector (35t, protoDUNE, WA105 (?), ND, DUNE).

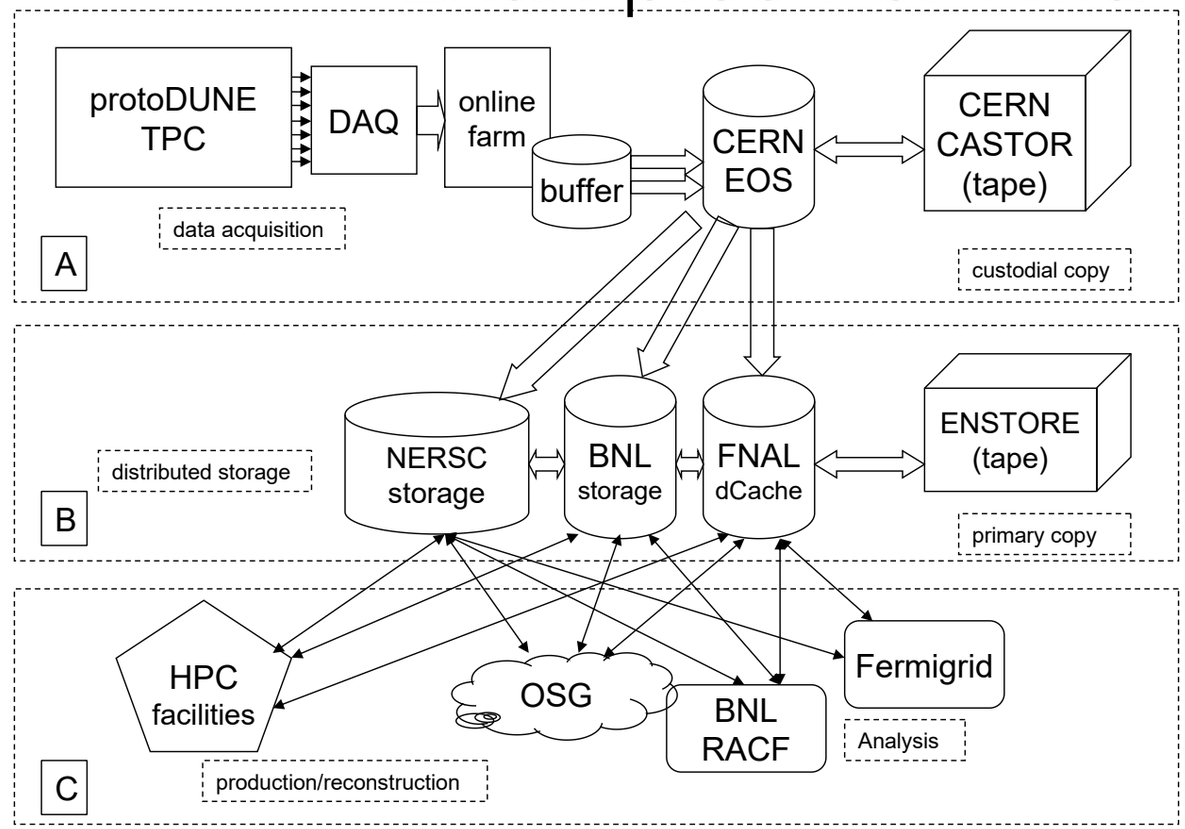
Model Design

- For each detector, we need to identify
    - calibration/study/physics tasks,
    - estimate requirements,
    - map the workflow, and design a model.
- Probably makes sense to identify small set of difficult and benchmark tasks which would also address less demanding tasks.
- May also define certain requirements… e.g. N replicas of data, access to full raw for some % of data, …
- Model ~ data reductions/steps.
    - My impression is that some analyses may require re-reconstruction in their analysis iterations…
    - Analysis Task example: **Stream** *-(raw)->* **central 1st pass reco** *-(reco-obj+raw)->* Filtering -> **2nd pass reco** *-(reco-obj)->* **n'tupler** *-(summary TTree)->* **plots**.

Model Implementation

Examples from Maxim





- For each detector, we need to workout how sites/hardware/software/services implement to proposed model.
    - Serves as a high-level blueprint for the software/computing.
    - May also define operation/production groups/activities.
- May require comparing technologies (e.g. WMS), formal review of competing options, and an official decision process.

Summary

- Maxim and I will be attempting to evolve Computing Model activity towards a working group.
  - Collect info, identify unknowns, commission studies.
  - Provide tool for turning input estimates/assumptions into resource estimates.
  - Define scenarios. Collect requirements.
  - Propose models… work out implementation.
  - Provides input to decision making process and reviews.