Comments/Needs for CMS event reconstruction

Slava Krutelyov (UCSD)
Reconstruction WG, HSF CWP Workshop
Jan 24, 2017

— Knowing the envelope
— Features to follow

Knowing the envelope

- **Resource side**
  - **We are projecting to needs for running in 7 years or more. Projections for technology gains (better HS06 of GB memory or PB storage per $) will be more precise a few years from now. We should be aware of this realism coming.**
  - **Memory/core envelope can restrict workflows (e.g KNL is ~0.3 GB/core)**
- **Physics side**
  - **Trigger rate can be lower**
  - **Track reconstruction can be restricted in momentum or vertex(s) of interest**
  - **How much MC is enough**
  - **Settle on expensive part that can't be redone on the fly (reduce chance of rereco)**
  - **Knowing which parts of pileup not to reconstruct**
- **Planned reconstruction workflows should have a baseline and an affordable configurations to converge on a workable solution. Define affordable.**
  - **An important component here is to keep a "fair share" in algorithm/product cost.**
- **Detector hardware side: recall that detectors are not built yet**
  - **Can the hardware be adjusted still to make reconstruction faster**

Features to follow

- **Costly features**
  - **Tracking: (this has historically been ~50% of CMS reconstruction time)**
    - Large enough task that can even benefit from specialized hardware and software implementation
    - Faster versions of algorithms most beneficial
- **New detector capabilities**
  - **High granularity calorimeter**
    - Number of channel and fairly high occupancies can lead to costly algorithms
  - **Timing layer(s) time measurement in tracks can reduce impact of (in-time) pileup**
- **Machine learning**
  - **Training: not explicitly a part of reconstruction, see other WG**
  - **Classification applications: adiabatic inclusion appears to be most practical**
    - Robust and less ambitious to be kept at lower level
    - Make sure most useful discriminants are available in (mini)AOD outputs
  - **Pattern recognition: adiabatic inclusion may be difficult for lower level objects**
    - Replacing one type of tracking with another can take a O(year)
- **Multi-threaded algorithm evolution**
  - **Fine-grained parallelism (at event level)**
  - **Improve memory use efficiency (reentrant; less cloning; event data fragmentation)**