LHCb computing model

David Hutchcroft for the LHCb group

Note numbers are generally order of magnitude estimates only

LHCb data files

- Simulation using GEANT4 to .sim files and Digitisation to .digi files
  - Currently takes about 10-20s to simulate an event
  - This includes the 2 previous and 2 next bunch crossings for some sub-detectors and on average 0.3 other pp collisions
- Reconstruction to .dst format from either .digi or raw data
  - About 2s per event
- Physics analysis will run on .dsts

My understanding of LHCb plan

- Data taken by detector
  - Processed at the pit for the trigger and online monitoring
  - LHCb runs a two stage trigger:
    - L0 looks for muons, calorimeter clusters, electrons and pileup, input is 40MHz output is up to 1MHz
    - HLT (high level trigger) takes 1MHz input and looks for specific channels
      - Reconstructs just enough to reject most events
      - Uses normal reconstruction software
  - After about 24 hours will be reprocessed at a Tier-1
    - Better calibration constants etc.
    - This year always CERN

Data available outside CERN

- After initial reprocessing have data suitable for physics analysis
  - Copied to Tier 1 centres
  - Have tracks and clusters with PID DLLs associated
  - DaVinci (analysis code) takes ~40ms per event
    - Used to generate ntuples for analysis
  - Data is "stripped" i.e. split into streams with specific triggers in each stream
  - Stripping is both event tag collections (pointers into main file) and copies of events

LHCb official analysis plan

- Central requests for MC
  - A dedicated processing and MC generation team managing generation and file distribution
- Jobs run via ganga which is a python front end to DIRAC which is the LHCb written wrapper on the grid
  - Can run job at remote sites, local batch system or on local machine
  - Can find the location of data and send jobs to sites with data
  - I get 80-90% success with ganga on the grid at the moment

Requirements at Liverpool: Disk

- Enough disk space to store the stripped data for each analysis
  - Each dst event is 150k for MC and 120k for data
- About 15TB sufficient for a years data for a normal (10Hz) rate stream
  - Tara will want one for W/Z production and I will want one for CP violation
  - MC data will probably be significantly smaller in numbers of events

Requirements: Batch Processors

- Enough computing to re-run on the events as required
  - Assuming a 10Hz stripped stream at 10^7 s per year at 40ms per event : 1k CPU hours to process all of the data from a complete year of running
  - Note single 1GHz Xenon CPU assumed
    - For example 16 3GHz cores would process the data in 24 hours
- Enough to generate 1M event MC samples in a week would be around 20 CPUs [approx 0.15TB of data]

Processors: “near” interactive

- Toy models: CP analysis depend on generating and fitting toy events to validate CP fits
- BaBar experience says need to run 500 toy fits “interactively”
  - Was taking about 20mins on IN2P3 interactive machines per job (approx 1GHz machines)
    - Recent review of a note causes a rerun of 7k jobs of toy fits
  - Needed to be quick turn around as the output of each job was used to decide the parameters of the next job to run
- So enough machines either interactive or dedicated to queues < 8hour queues

Parallelising processing

- LHCb (and ATLAS) run GAUDI which can be python driven
- Gaudi architect (Pere Mato) presented how to automatically use all cores in a machine in the same job
  https://twiki.cern.ch/twiki/bin/view/LCG/MultiCoreRD
- The code is at proof of concept stage
  - Can dispatch events to each core to process
  - Has a fraction of a fellow from September
- Was pointed out this is almost the same as running n copies of the program (saves common memory only) also with 8 cores get about x6 the speed of one core
- IO limits often become the bottle necks
- Better for simulation perhaps