LHC@HPC

Possible HEP-Experiment application in CoE

15-April-2014

Pete Clarke (Edinburgh, LHCb)
Andy Washbrook (Edinburgh, ATLAS)
Input from Mauro Mondrian (INFN, LHCb)

Motivation

- HEP-EX is definitely "Physics at Extreme Scales"
- Embarrassingly parallel event based problem
- No intrinsic HPC needed in the past – all computing done on massive HTC clusters (400,000 cores)
- But in the future: Many core commodity devices and possible convergence of HPC/HTC data centres means we **have** to learn to use many core-devices efficiently
- Also, there are some intrinsic areas where ~ 100s of cores can definitely help..
- In context of this CoE: to discuss whether "HEP-EX" brings a useful application area ?

I will show a few very simple slides to outline where HEP-EX has fingers into HPC

Then Andy Washbrook will show you a concrete example of some work

Everything is based upon individual "events" from a collision

Typical computing tasks are:

- The initial event trigger
- Event based data reconstruction
- Monte Carlo simulation
- Physics event classification (pattern recognition?)
  - MVA (BDT, NN..)
- Parameter estimation using maximum likelihood fitting
  - actually minimisation of – Log(joint likelihood)
- Very large pseudo experiment generation (100,000 experiments)
  - Statistical coverage checking a-la Feldman-Cousins
  - Systematic error determination
  - Exploring parameter space

Trigger

- Collision rate 40 MHz → Data recoding rate is in kHz
  - Real time trigger discards things which are “definitively not interesting”
  - This has to process at the collision rate
  - Typically done on very large clusters
- Today there is lots of interest in all experiments in using large GPU like machines
  - Primarily for cost effectiveness
  - Also reduce latency
- We think this is a good potential application

Event based reconstruction

- Event based data reconstruction and "skimming"
  - Reconstruct pulse heights ..etc.. into "hits", "energies"
    - → Create simple physics objects "tracks", "energy clusters"
      - → Create physics objects "particles", "jets"
  - Pass through data set to make initial selection of streams of interesting event categories

- What is scope for parallelisation:
  - This is all intrinsically embarrassingly parallel - each event takes ~ seconds, we just need to do it for millions of events
  - Atomic elements like track finding
    - → typically people look at GPUs
  - Course level Parallelisation of the reconstruction framework itself
    - → In ATLAS/LHCb it is called Athena/Gaudi
    - → Can run disjoint components in parallel

- Our guess is that this is not the most fertile ground for an application
  - But lots of work is going on into the course parallelisation

Monte Carlo Simulation: Geant

- The majority of LHC CPU time is spent on MC generation
  - You need a lot of CPU time to generate only a few seconds of beam time !
  - Almost exclusively uses Geant for detailed simulation
  - Fast simulations are also used when detail is not needed
  - Libraries of components taken from data are also used.
  - ATLAS have a sophisticated framework (ISF) which allows mixing and matching of fast/ detailed/library simulation according to the task.

- What is need/scope for parallelisation of Geant

  - I cant speak for Geant collaboration, but im sure there is lots of scope – see experts

- Very repetitive minimisation process
  - Miminise { Sum_over_events [- Log (Likelihood(data ; parameters)] }
  - The clue is in the sum: large number of events + complex calculation in likelihood means that fits can take more than a day to run.
  - This makes it impossible to easily “try things out” and do multiple fits

- What is scope for parallelisation:
  - Just need to split the Sum_over_events into N subsections
  - Lots of work on GPUs (“Goofit”)
  - Lots of work on Xeon-Phi

- This is a perfect case where parallelisation on N cores reduces time to insight by 1/N.
  - Relatively small and well defined sector (this could be good or bad ?)

Possible interested groups

- It there is mutual interest (in this CoE) then we are talking about a small HEP-EX (HEP-Experiment) interest
  - Maybe 3-6 groups ?
  - Edinburgh HEP-EX, INFN: Mauro Morandin in particular
  - We know of others – but havn't suggested anything yet.

Some interest statements from Mauro:

1. How does one build scientific applications appropriate for use in a global computing system with heterogeneous hardware? Which parallel programming models and technologies will permit maximum flexibility and minimum vendor lock-in without comprising performance?
2. How can one properly profile and optimize complex applications running on heterogenous architectures using standard multi-platform tools ?
3. Can we make applications "performance aware" so that they adapt themselves and their workloads to maximally use a given hardware resource?
4. Can we make applications "power aware" so that they optimize their power use, in particular taking into account the differing power use of heterogeneous hardware resources?
5. More generally how does an application do "resource scheduling" for a given hardware resource to satisfy throughput and power use constraints or objectives?
6. Can we make higher level scheduling decisions on a distributed E-infrastructure based on information gathered from the performance-aware and power-aware applications and from information related to site power costs ?
7. To what extent can better performance or reduced power use be facilitated by better information exchange between the software applications themselves, the facility/computing-center cluster scheduling and the high level global task scheduling?

Examples

Andy Washbrook : HPC conversion work in ATLAS