#patatrack

Andrea Bocci, Elena Corni,
Felice Pantaleo, Marco Rovere

port CA tracking algo to CUDA

- Tracking takes ~30% of the CPU time during the online reconstruction of proton-proton collisions at CMS
- Porting the Cellular Automaton tracking algorithm to GPUs would
  - let us run the CA on a larger fraction of the events
  - free up CPU resource for other algorithms

Prior Profile

- hltPixelTracks
  - reconstruct the tracks in the central part of the detector
  - original algo (cpu): 30% time
  - CA algo (cpu): 20% time
  - optimised CA (cpu): 17% time

Evolution and Strategy

- port the CA algorithm to CUDA
- including simple GPU
  - memory management
- no changes to the strategy

Results and Final Profile

- code compiles ...
-
- ... still debugging the kernels !

What problems you encountered

- heavy use of C++ features (templates, hierarchy, pointers, dynamic-sized arrays, ...)
-
- templates make nvcc unhappy
-
- keep CPU and GPU pointers well separated

Wishlist

- nvcc support for gcc 6.x
- better diagnostic
- cpu_pointer<T> vs gpu_pointer<T>
- ???