GPU vs Geant4 validation plots using monoenergetic electrons

Philippe Canal, Guilherme Lima, Soon Yung Jun

Fermilab

Vector Prototype Workshop
January 23rd, 2014

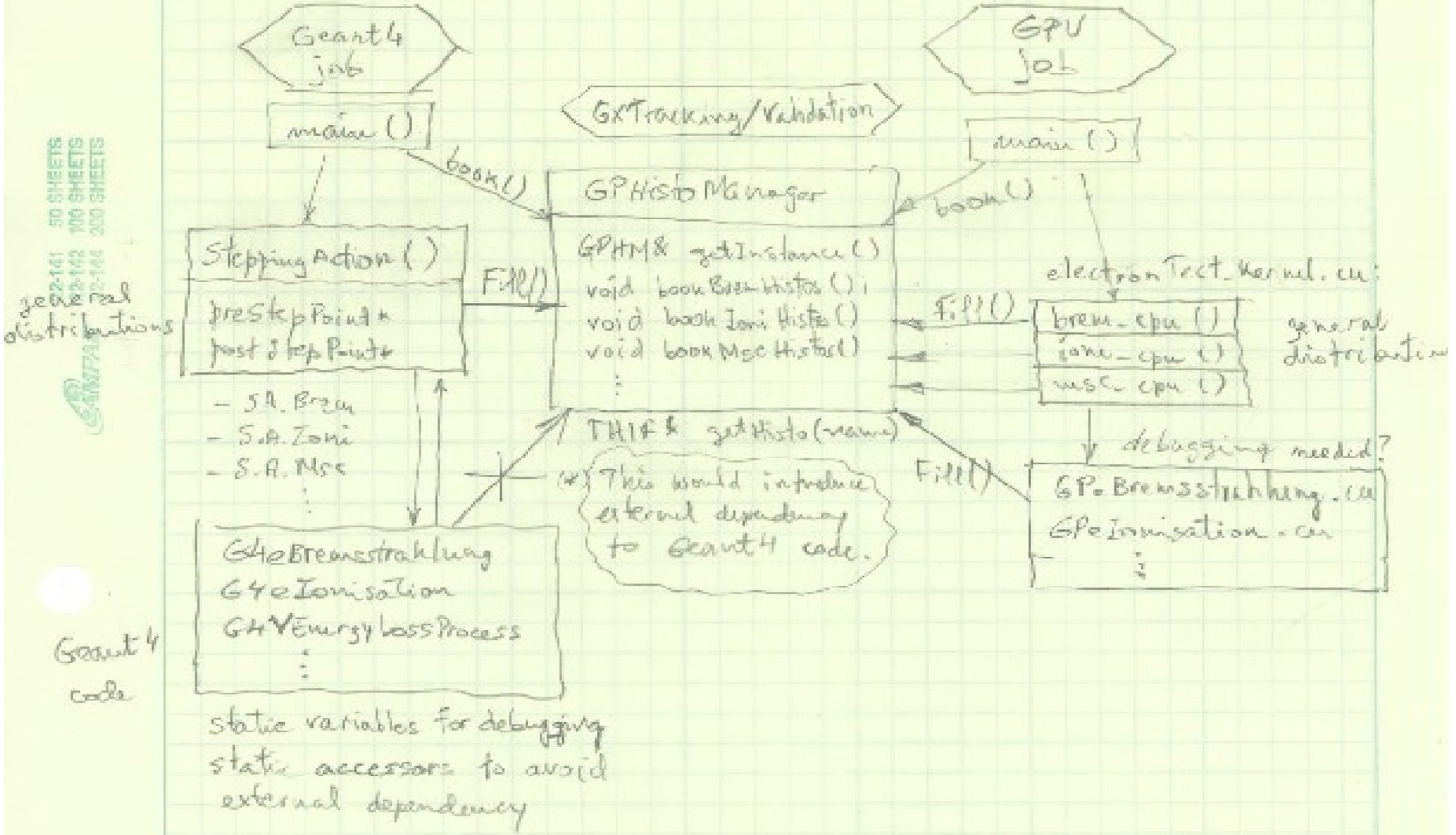
Development of GPU validation infrastructure

- Histogram management class: GPHistoManager
  - A singleton class, which can be called from anywhere
  - Sets of histograms booked for each physics process to be tested
    - Single place to change histogram parameters (easily overlaid)
    - Default booking in case a histogram is not booked explicitely
    - Python macros for histogram comparisons

- Latest work (late December & early January)
  - Build automation (Cmake) and configuration switches

    -DGPUDEBUG=OFF -DGPUNONRANDOM=OFF -DGPUPLOTS=ON
  - Switches can also be setup from file CMakeLists.txt
  - Validation library and two validation binaries built together

Class diagram for GPHistoManager



Comparing GPU prototype's physics with vanilla Geant4

- Simple EM physics processes implemented
  - *Bremsstrahlung*
  - *Ionization*
  - *Multiple scattering*
  - *Compton scattering*
  - *Photo-electric effect*
  - *Pair production*
- Started with *Bremsstrahlung* (simplest process): compare a few relevant distributions between GPU prototype and standard Geant4
  - Angular distributions
  - Energy spectrum
  - Step lengths
  - Energy loss in a step

If any distribution looks bad, other variables may be histogrammed to identify what is causing the discrepancies.

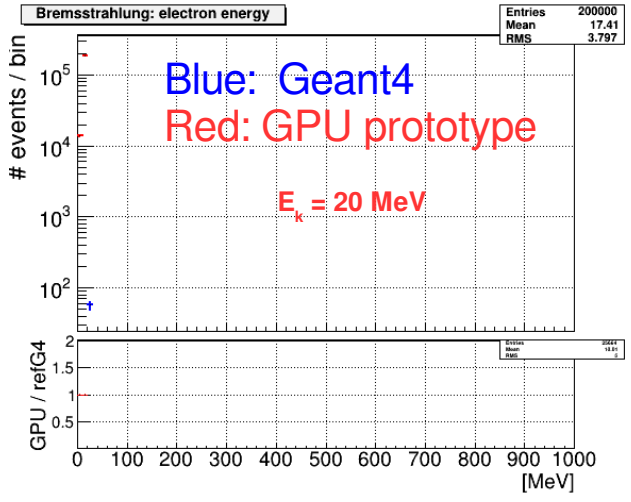
EM physics validation in the GPUs

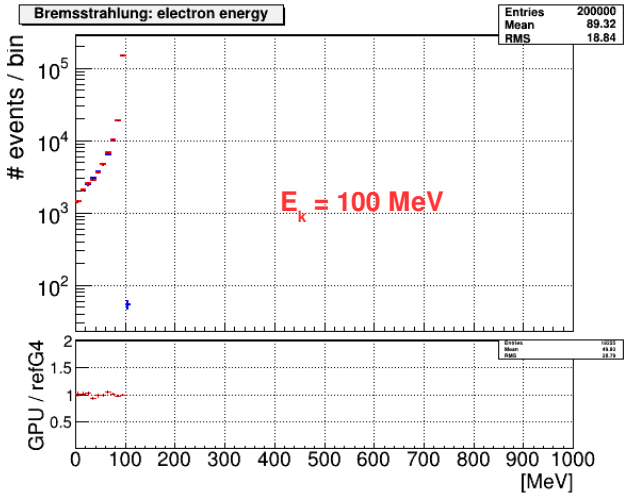
- Reminder: comparisons between GPU and standard Geant4 distributions in controlled environment
  - recently upgraded to Geant4.9.6.p02
  - Some processes turned off in Geant4 if not implemented in GPU
  - equivalent jobs: same geometry (very simple), no magField, same materials (properties compared in details). Physics list enables just one process at a time, single-step propagation of primaries only (secondaries immediately killed),
- Recent activities focused on automation: improved scripts to change verbosity and fix (non-)random number generators for maximum reproducibility
- Next slides: comparing *bremsstrahlung* plots using monoenergetic electrons, as suggested in a previous meeting

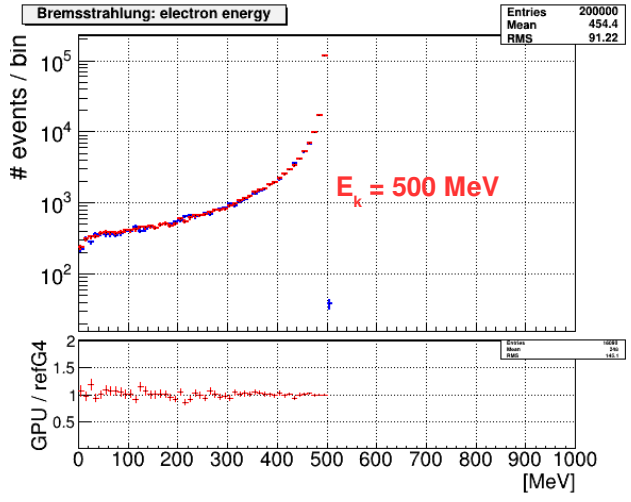
Bremsstrahlung Validation

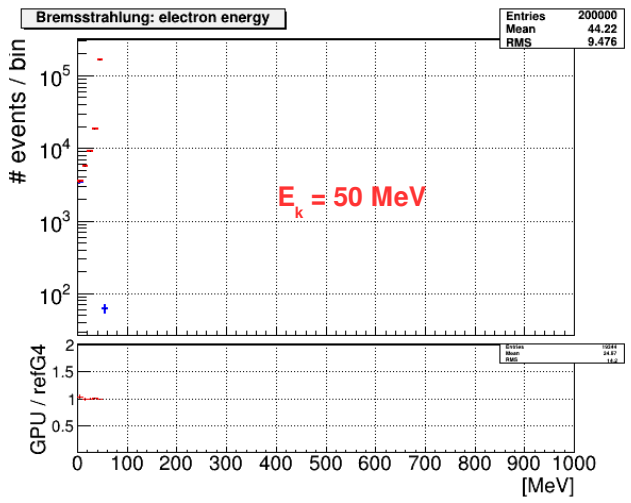
making sure that our GPU vs. Geant4 performance comparisons are fair

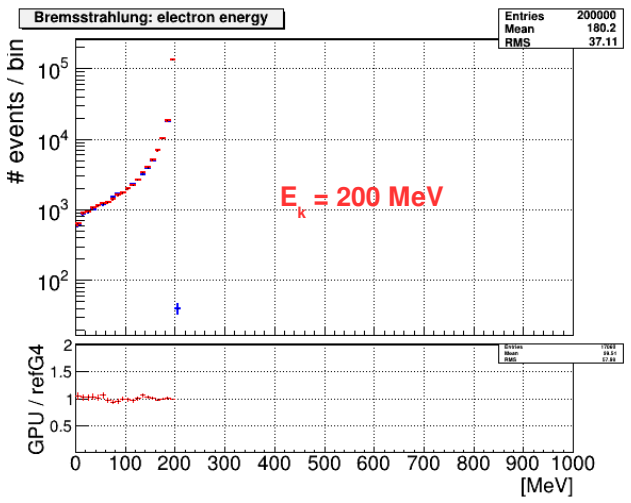
Bremsstrahlung: surviving electron's energy distribution

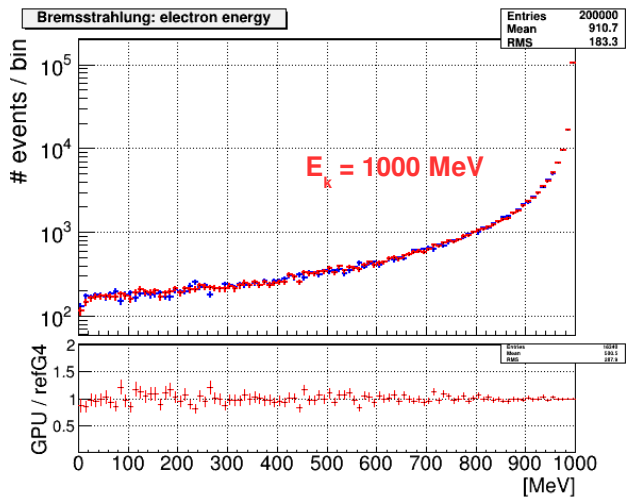




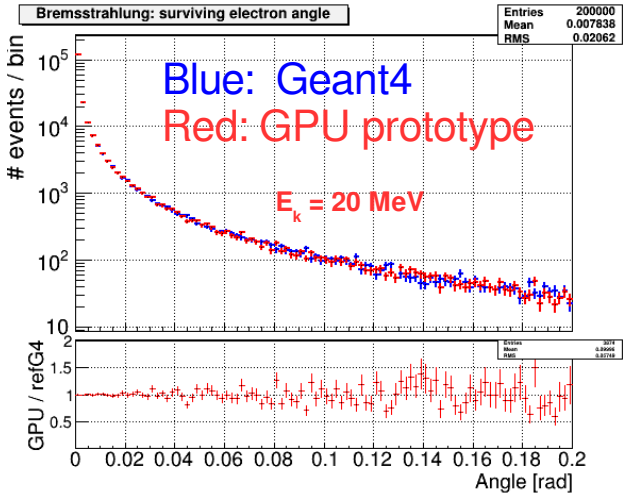


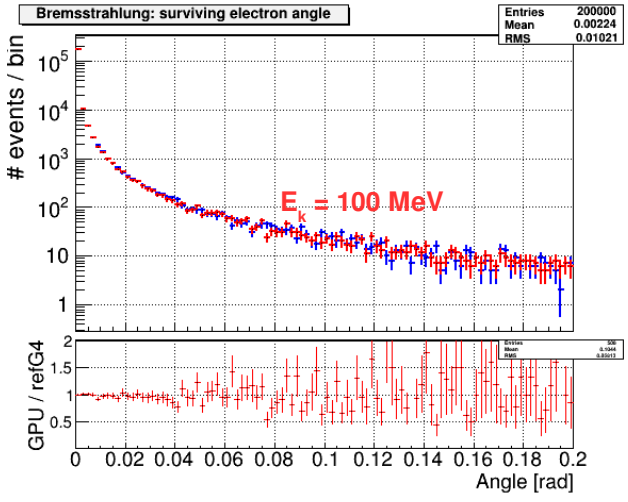


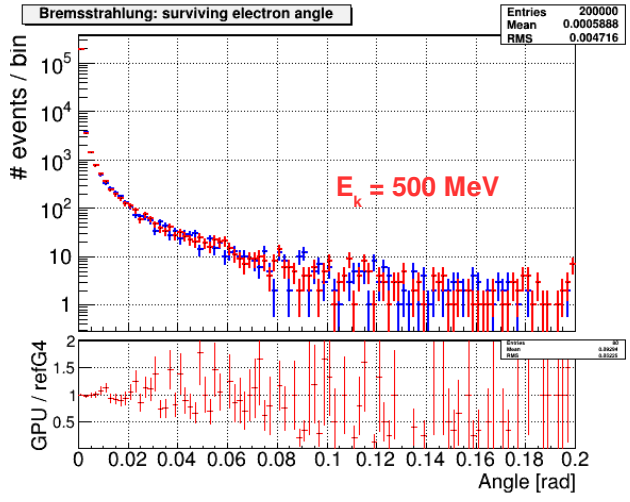


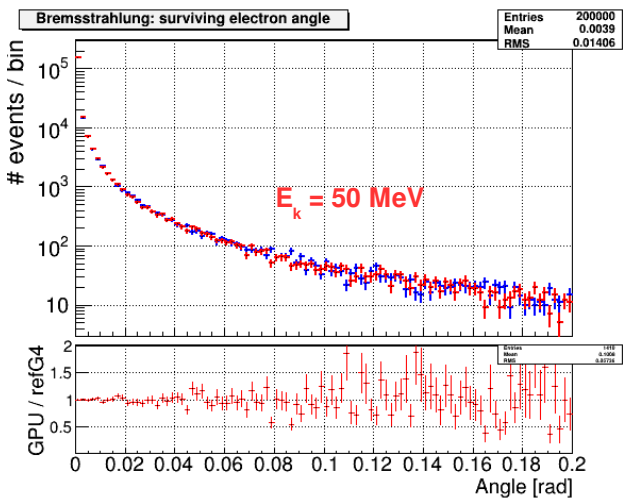


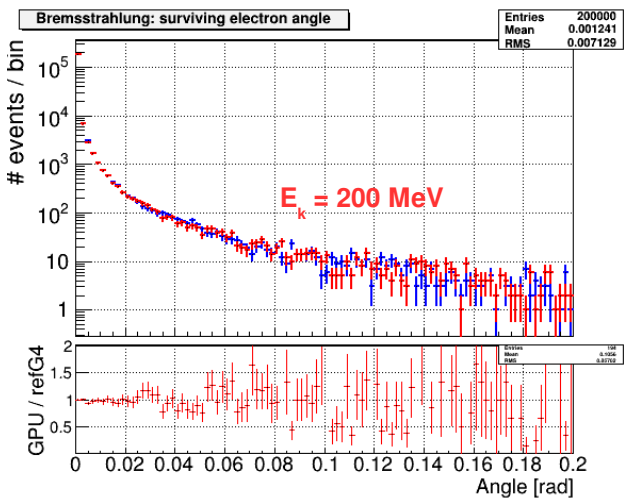
Bremsstrahlung: surviving electron's angular distribution

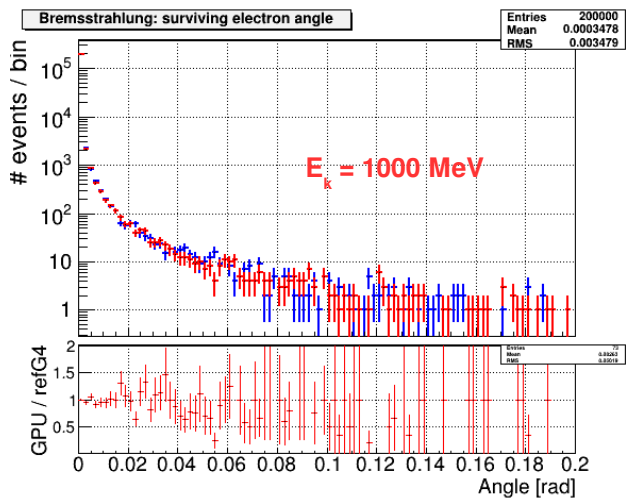




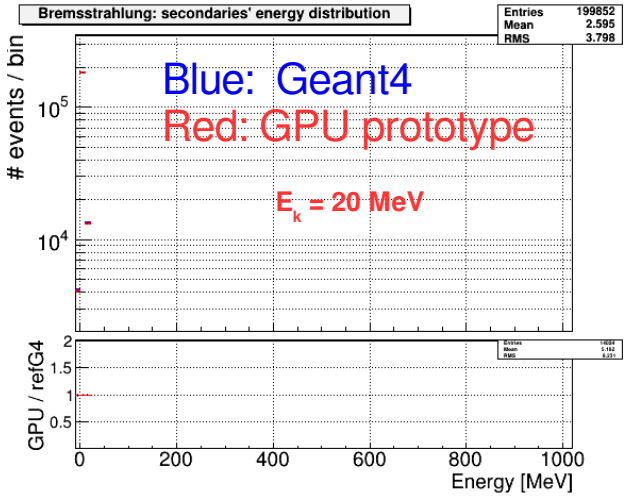


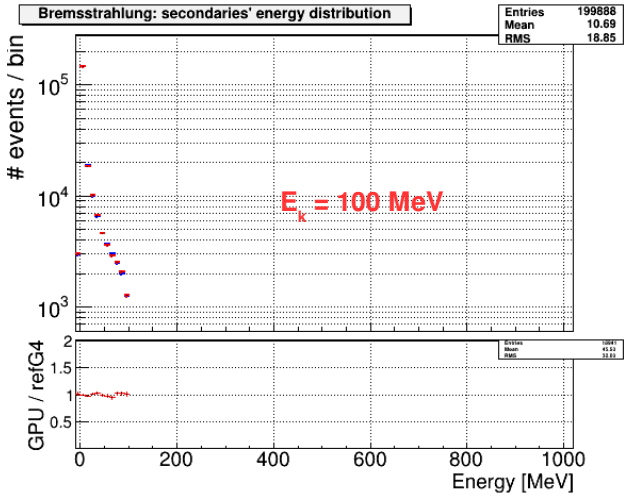


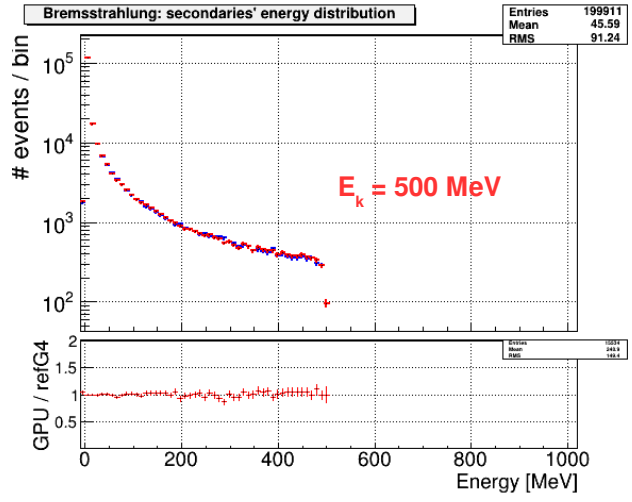


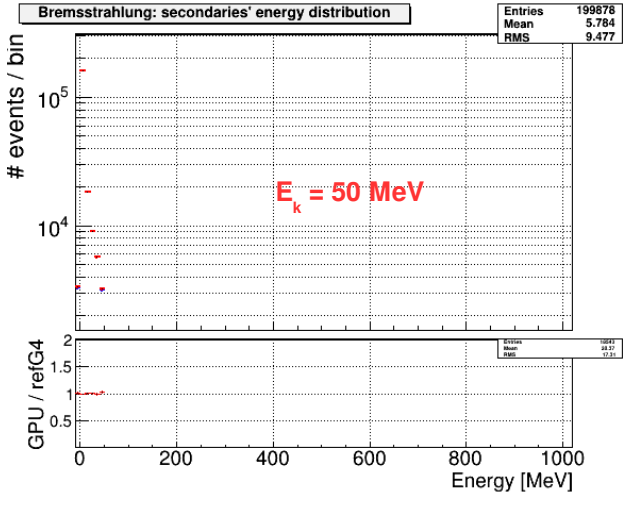


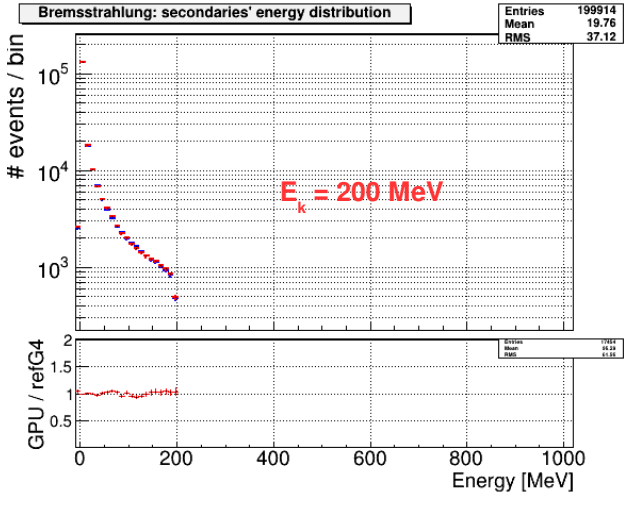
Bremsstrahlung: secondaries' energy distribution

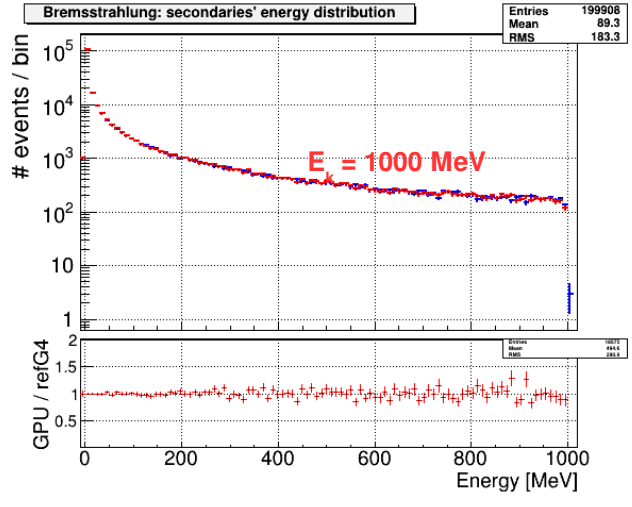




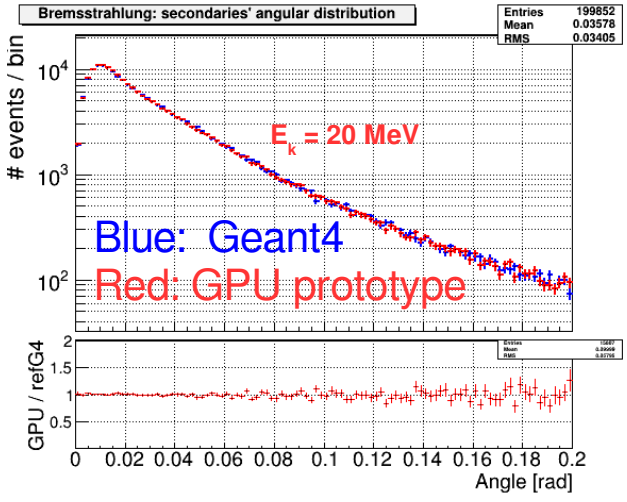


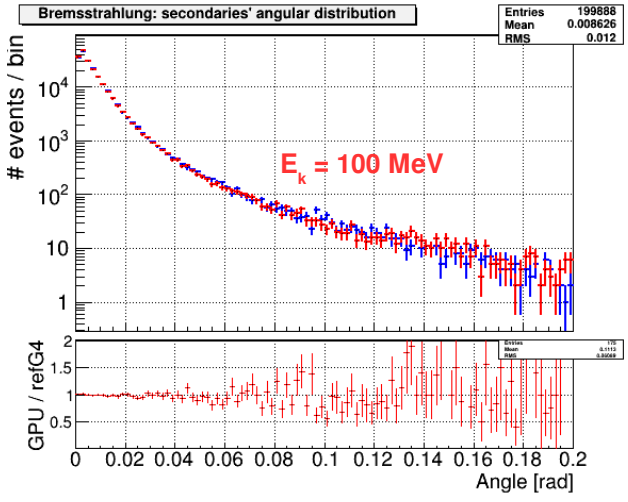


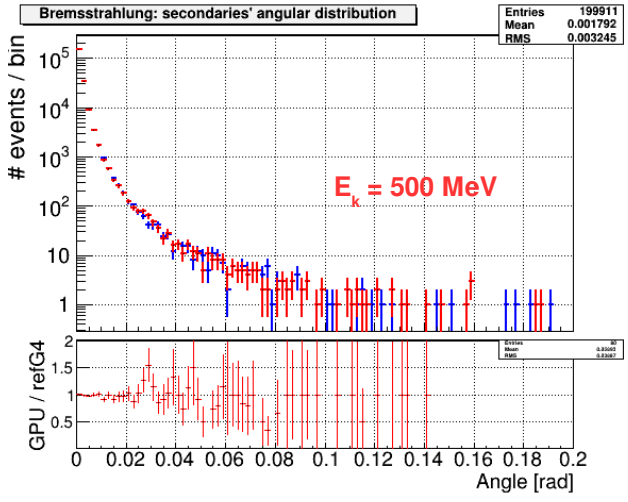


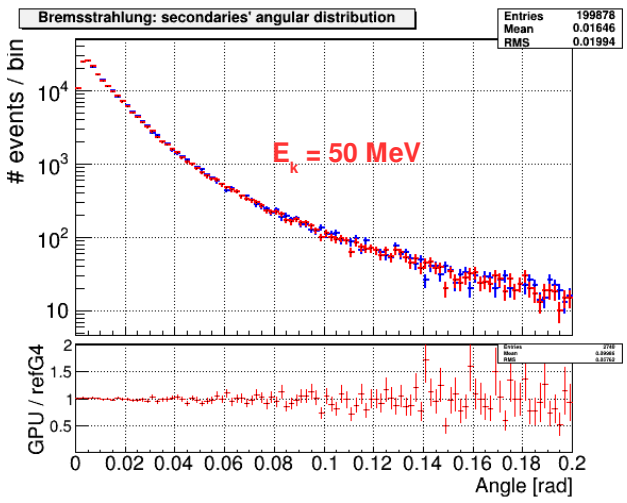


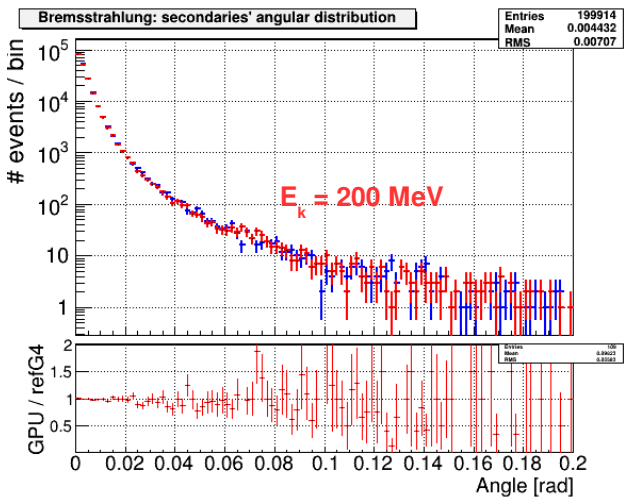
Bremsstrahlung: secondaries' angular distribution

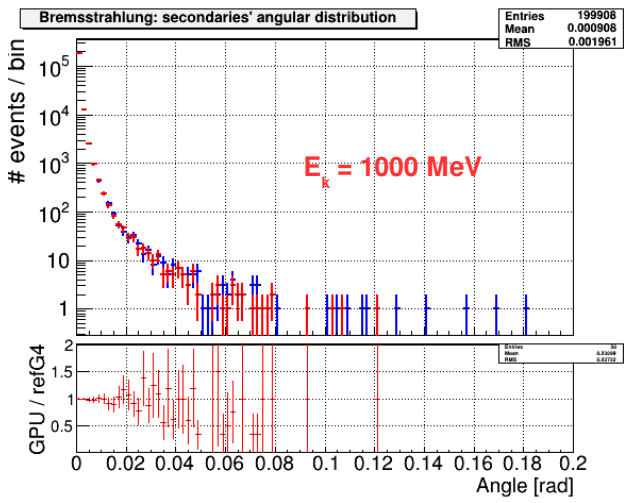




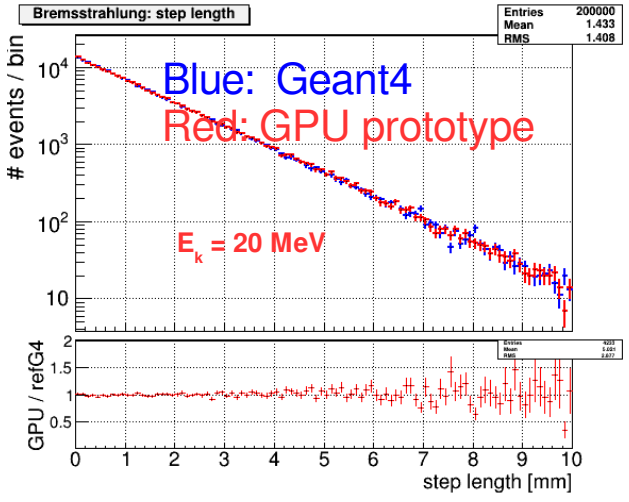


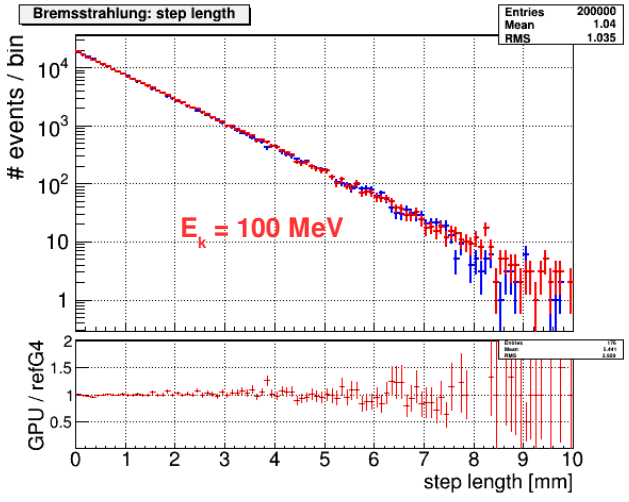


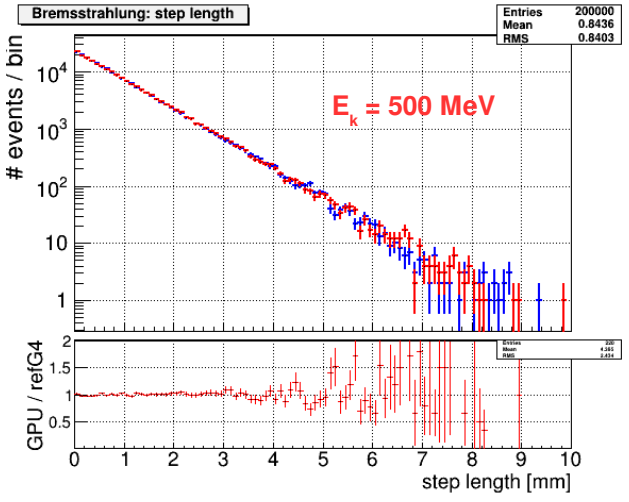


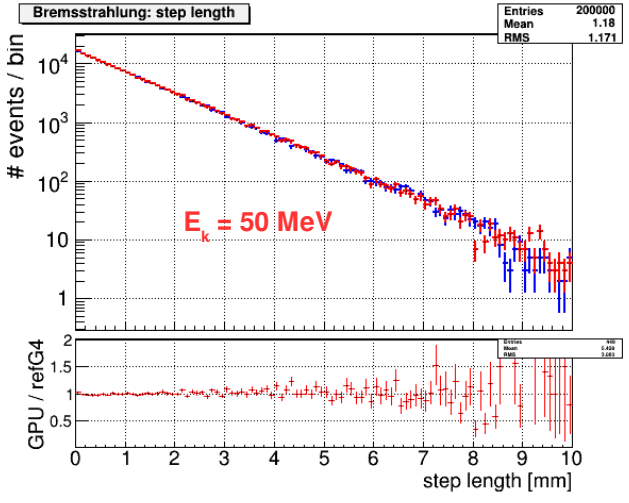


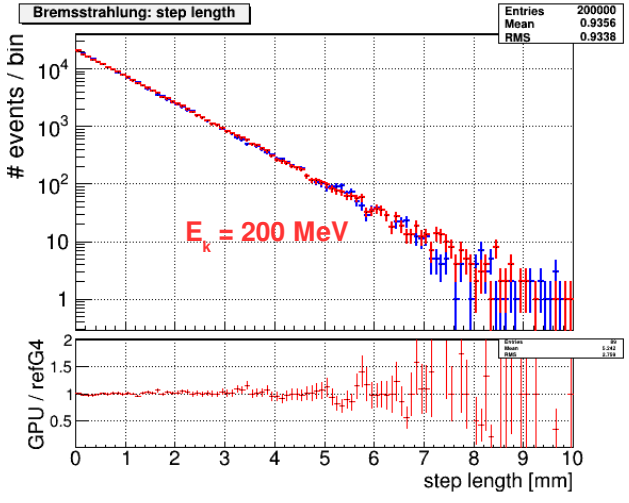
Bremsstrahlung: step length distributions

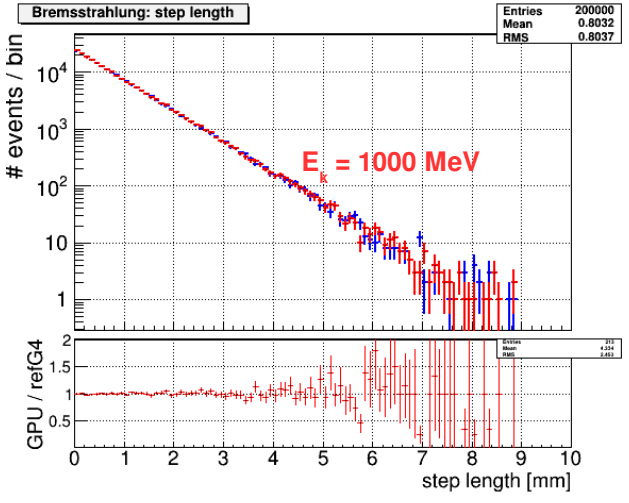




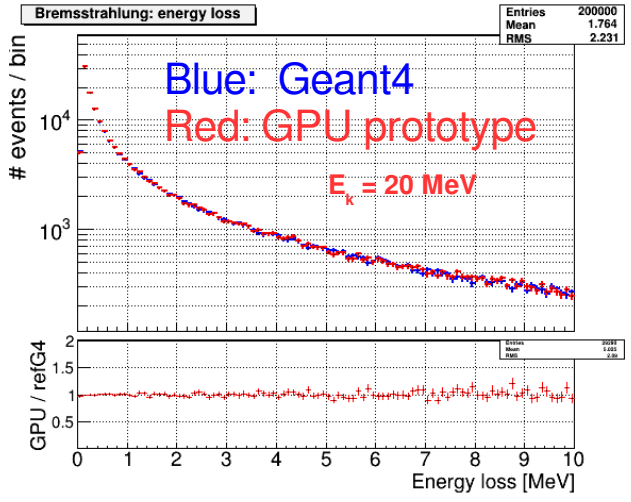


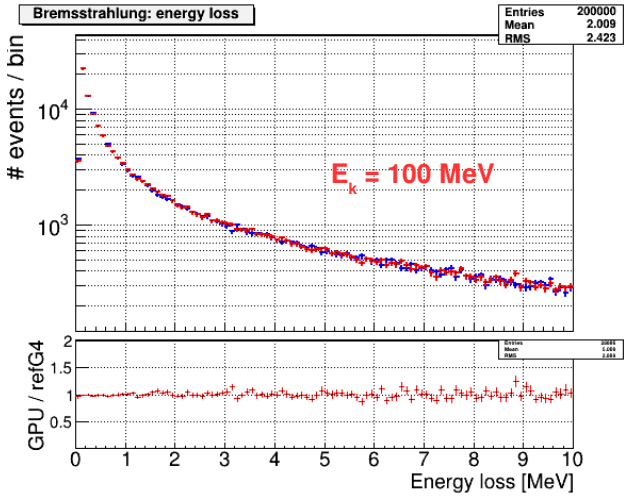


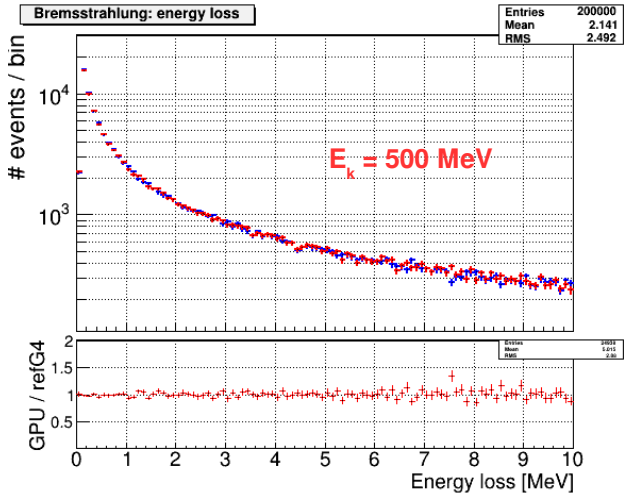


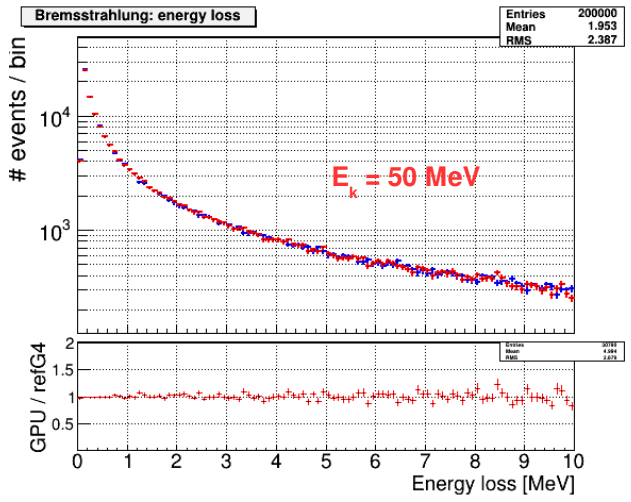


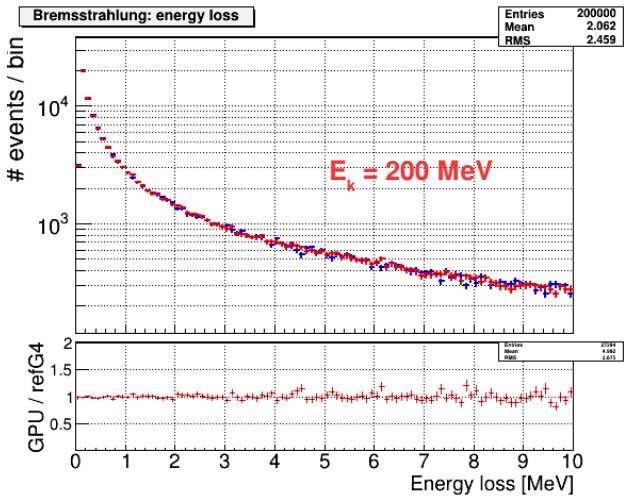
Bremsstrahlung: energy loss distributions

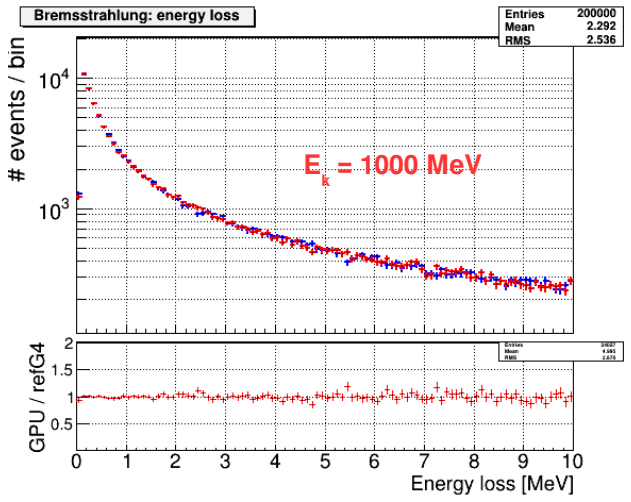








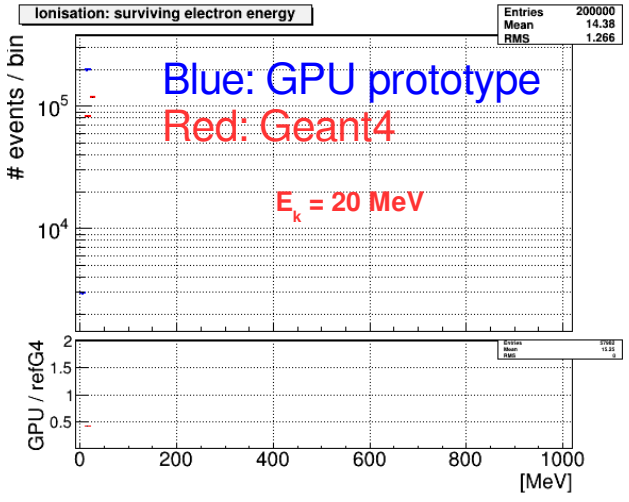


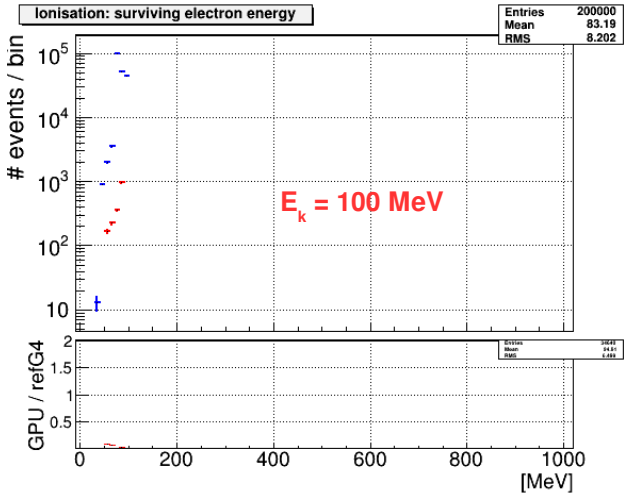


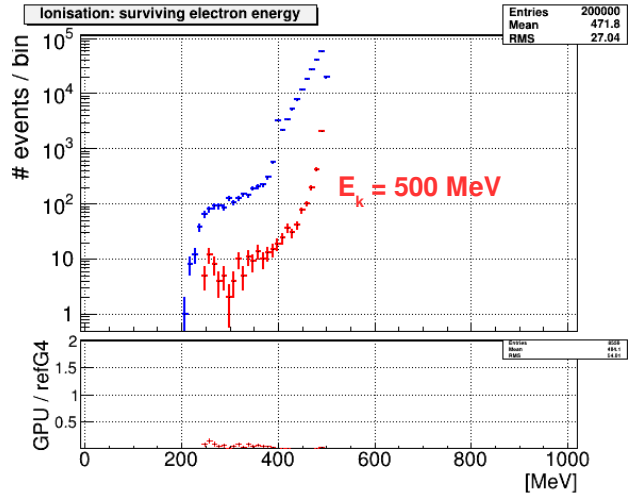
Ionisation Validation

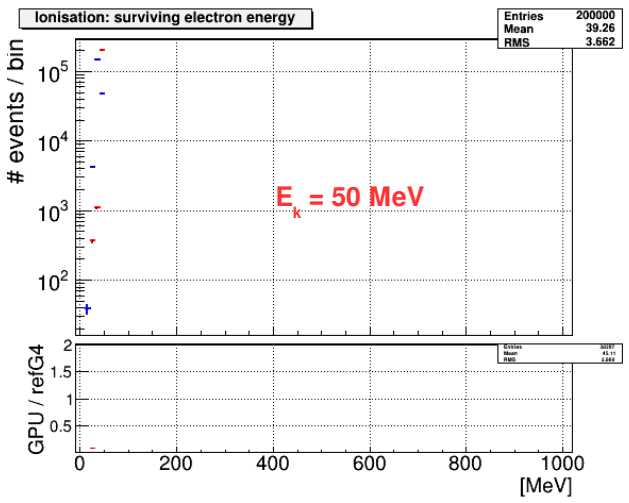
making sure that our GPU vs. Geant4 performance comparisons are fair

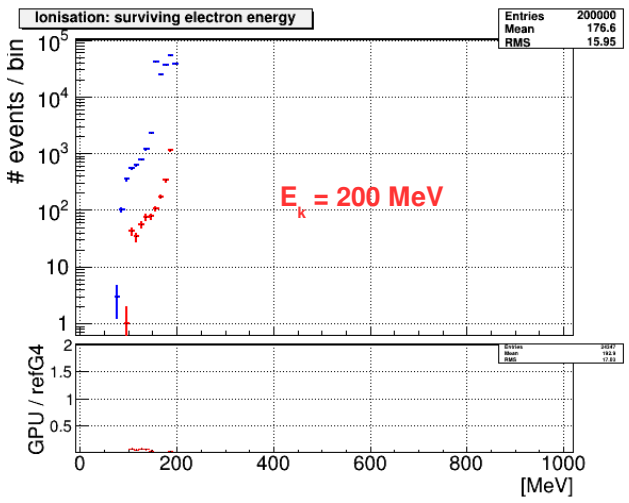
Ionisation: surviving electron's energy distribution

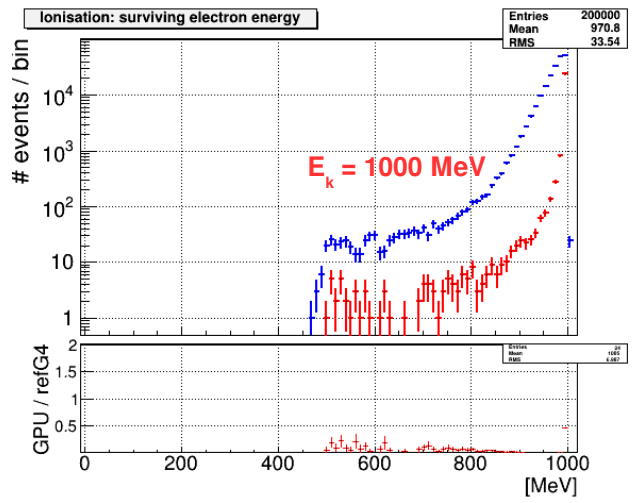




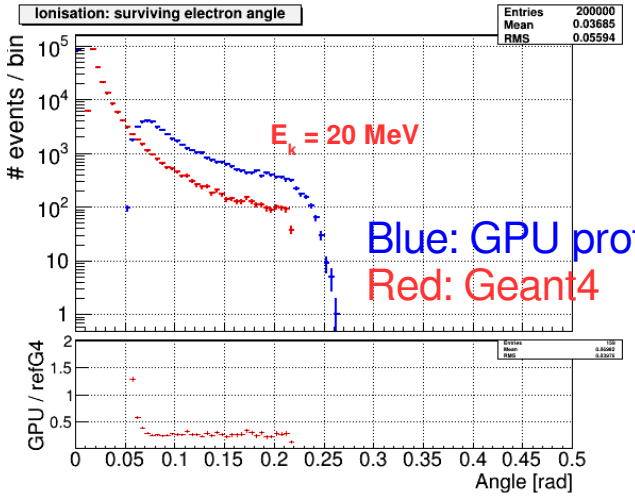






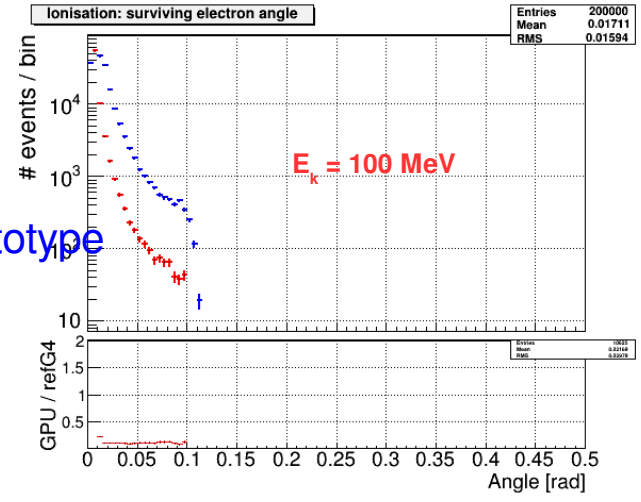


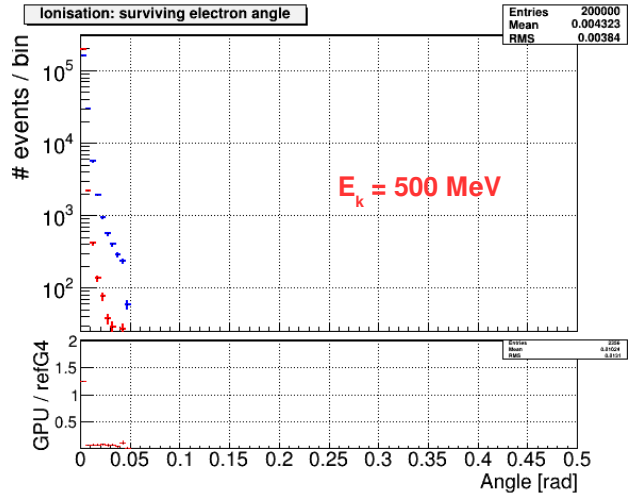
Ionisation: surviving electrons' angular distribution

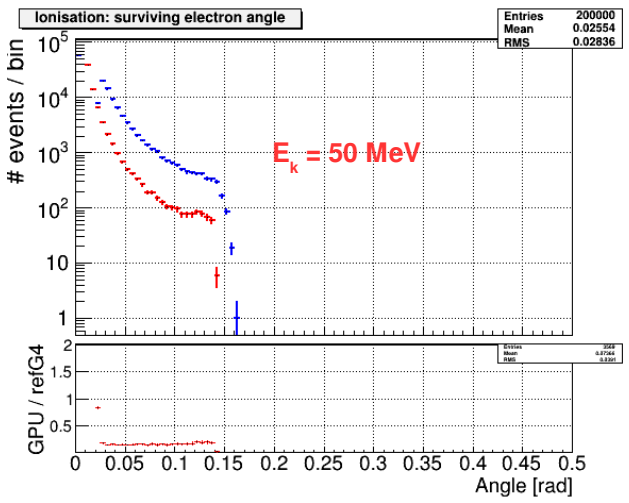


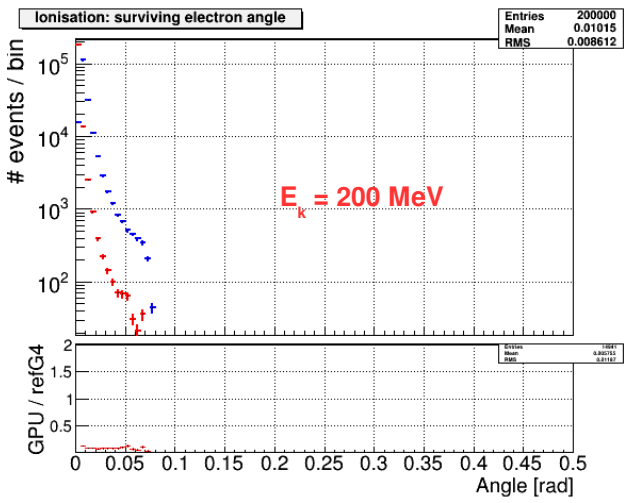
Blue: GPU prototype

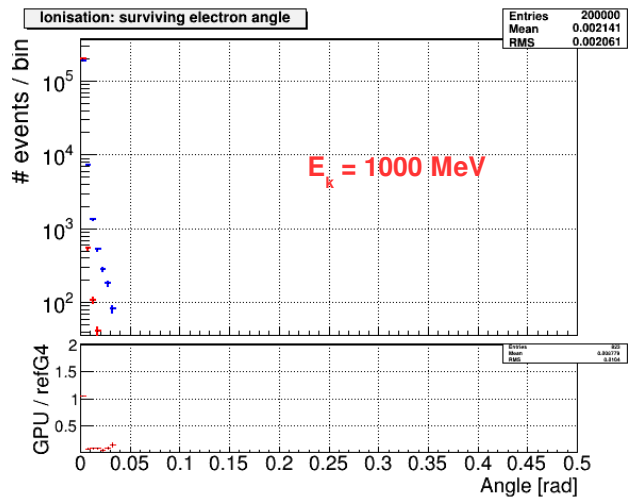
Red: Geant4



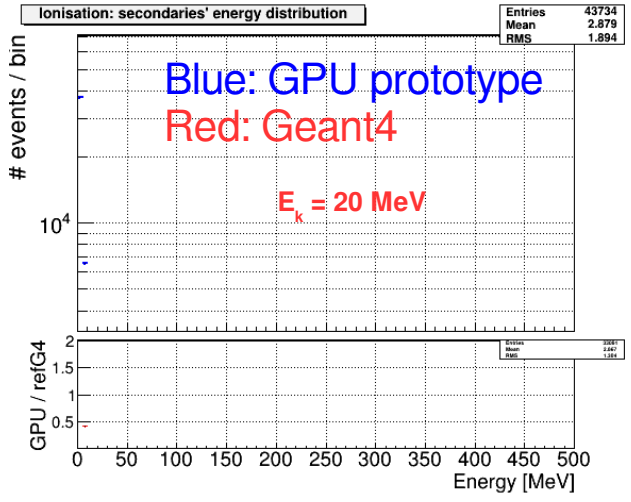


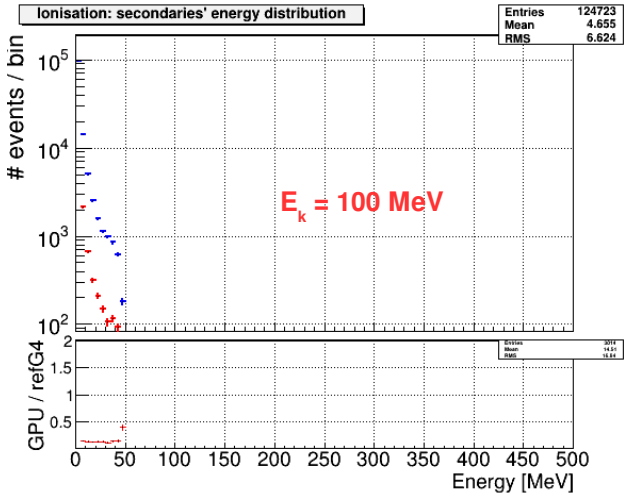


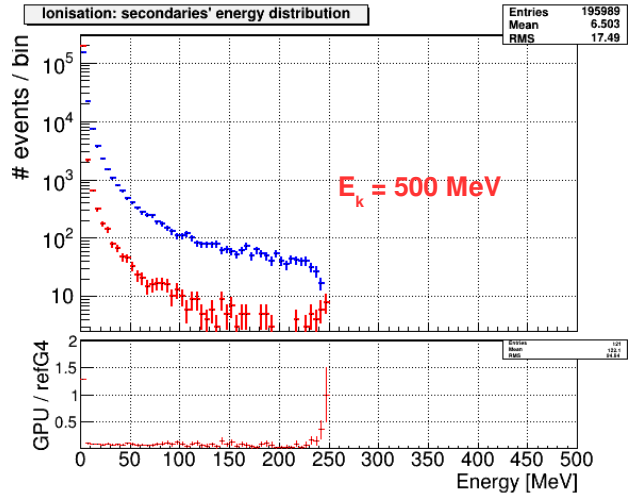


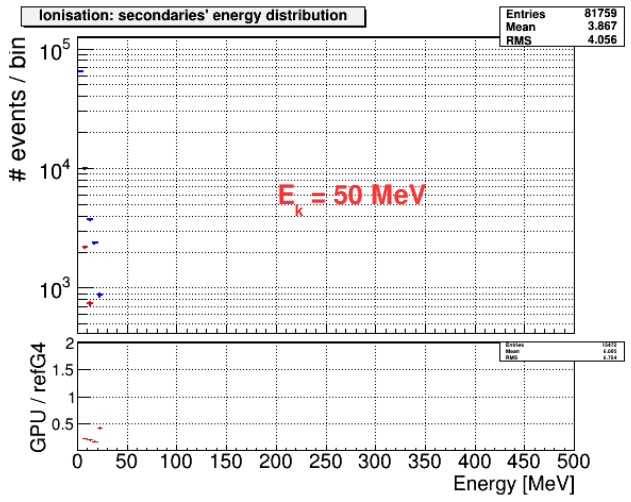


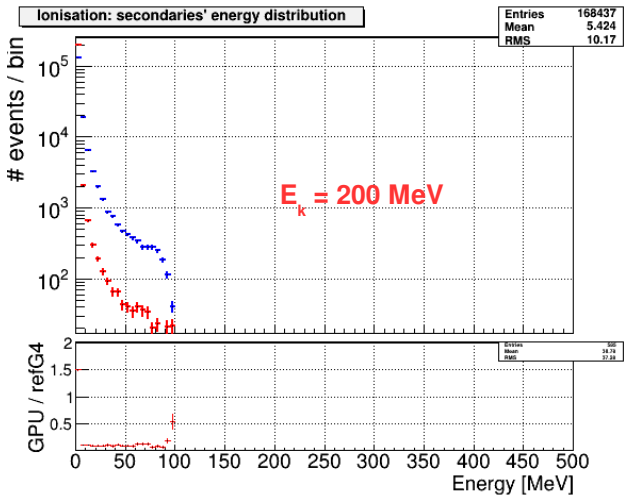
Ionisation: secondaries' energy distribution

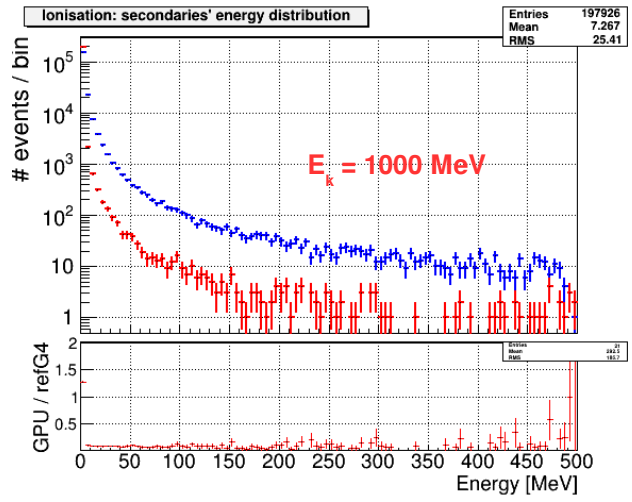




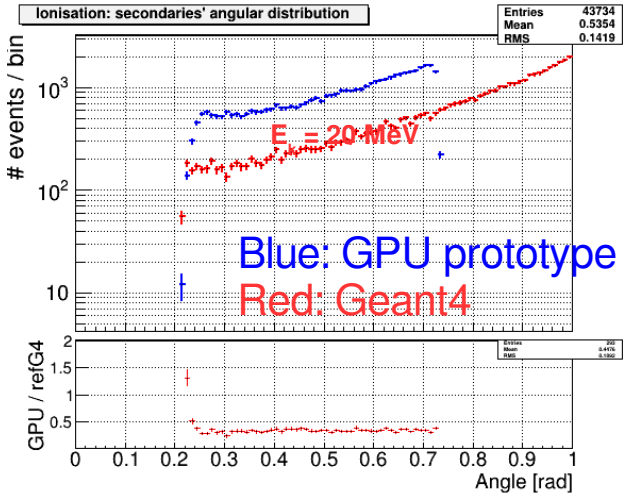


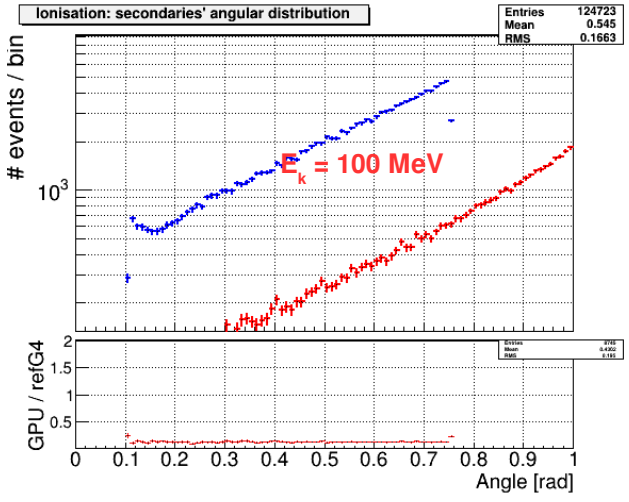


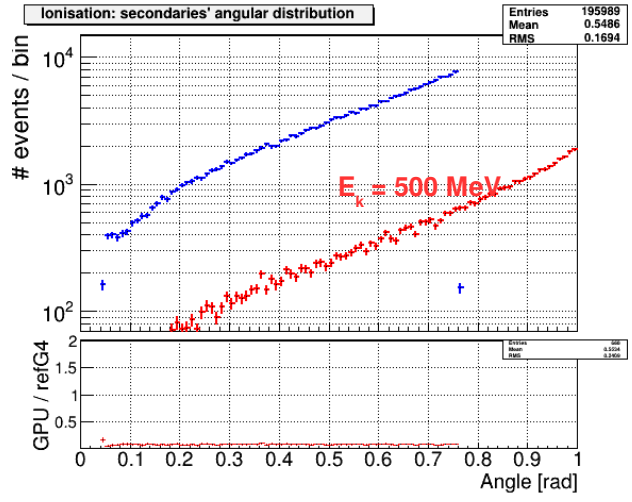


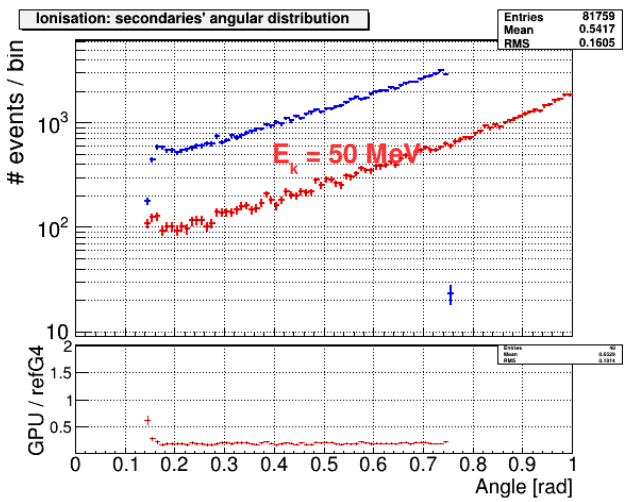


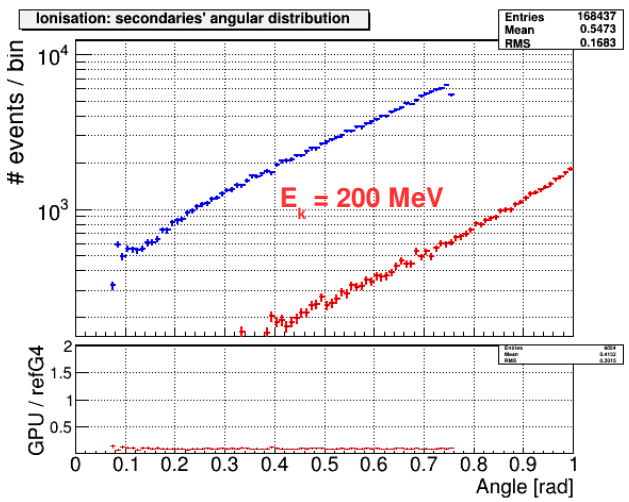
Ionisation: secondaries' angular distribution

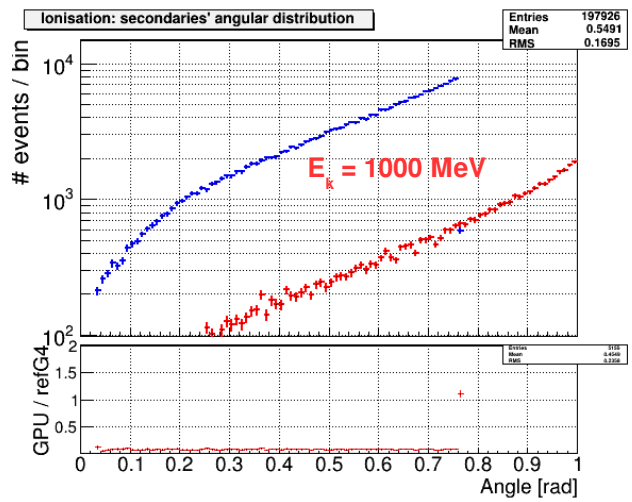




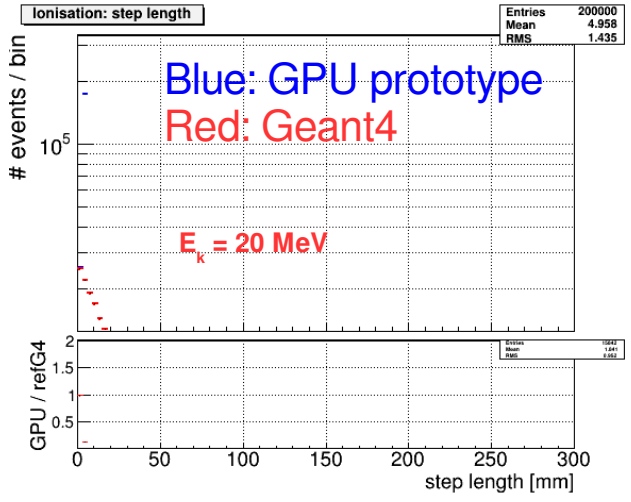


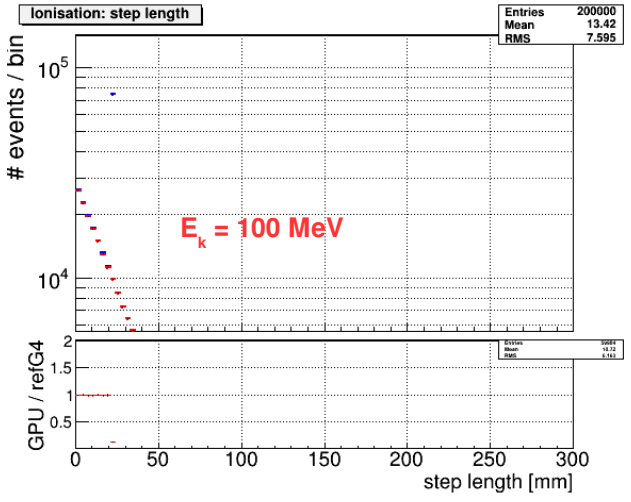


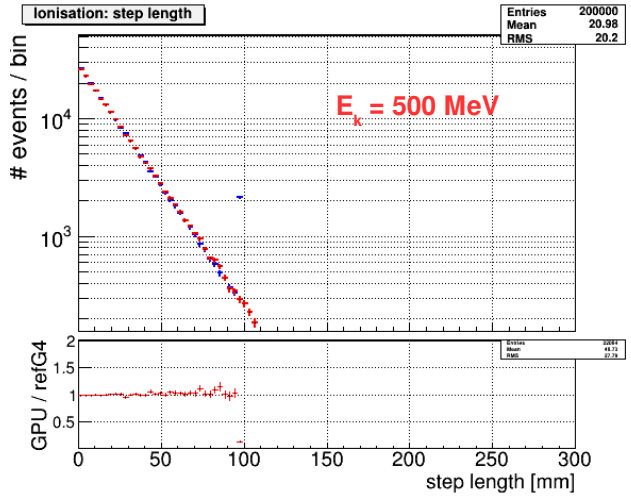


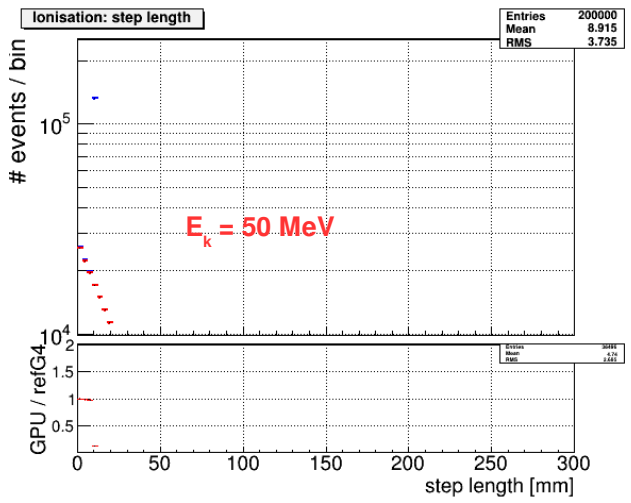


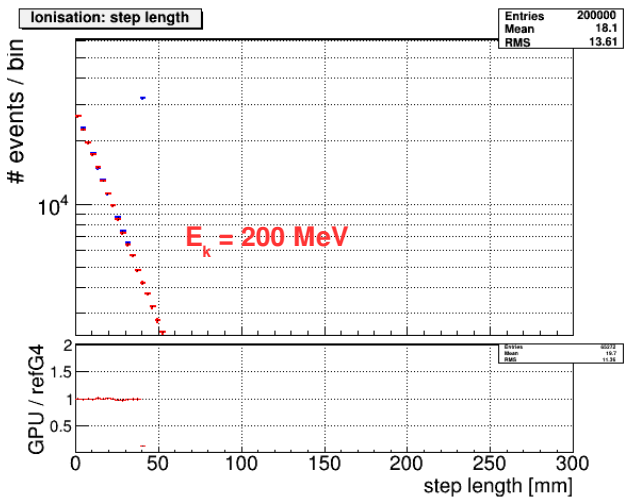
Ionisation: step length distributions

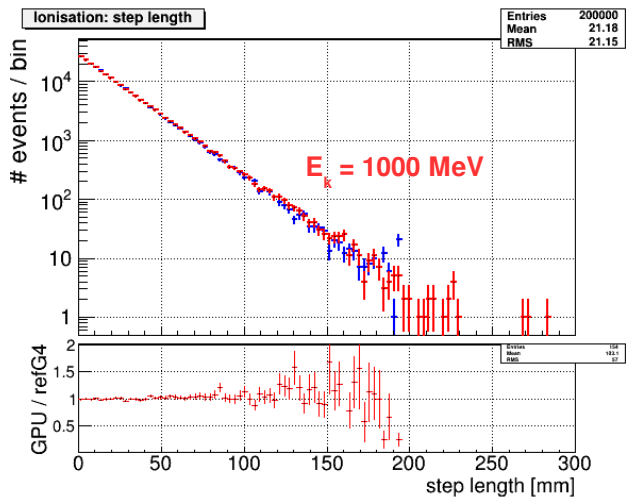




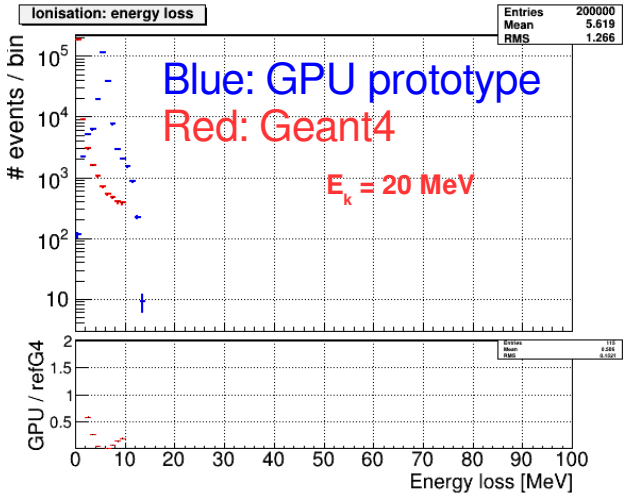


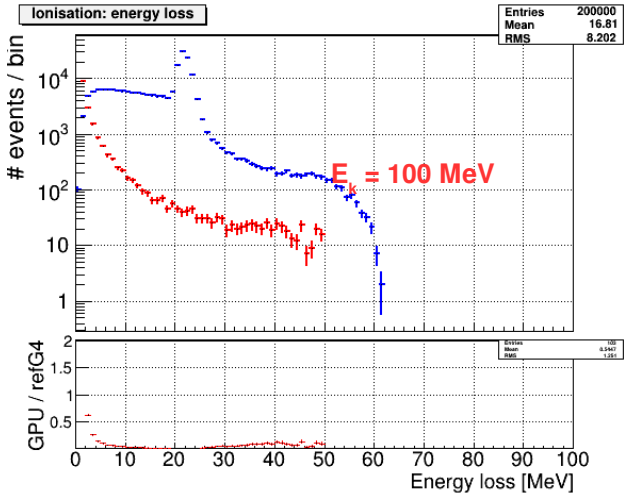


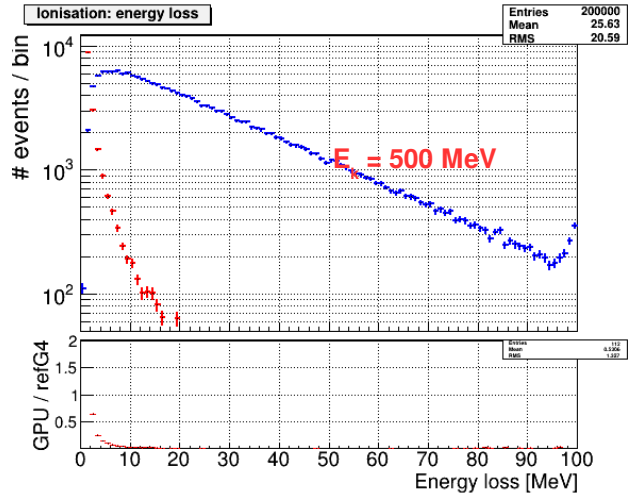


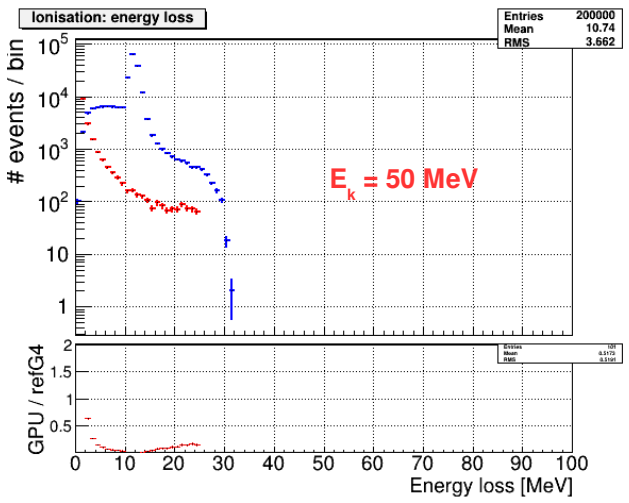


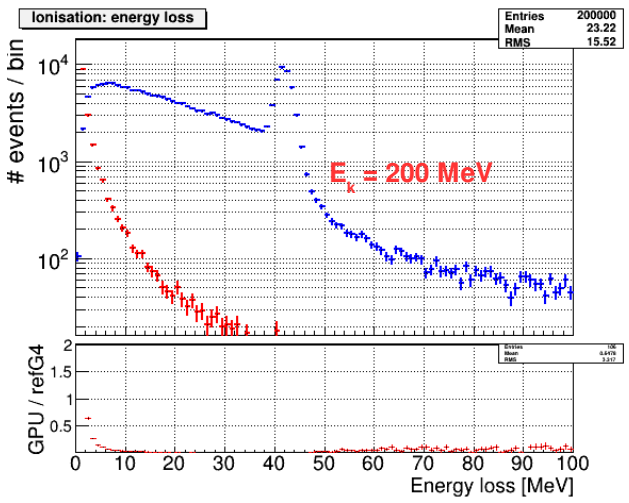
Ionisation: energy loss distributions – first look!

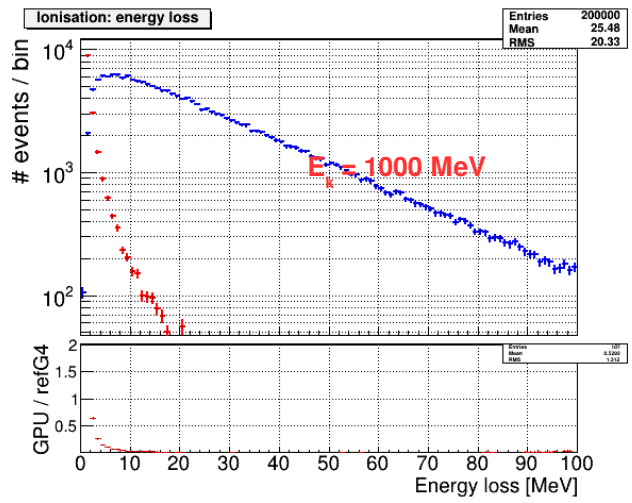




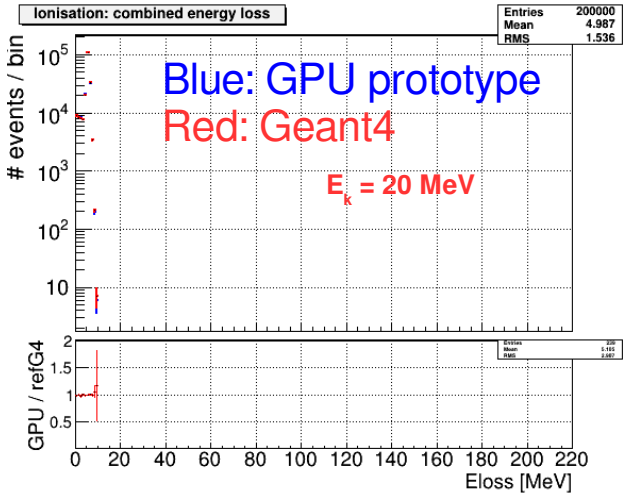


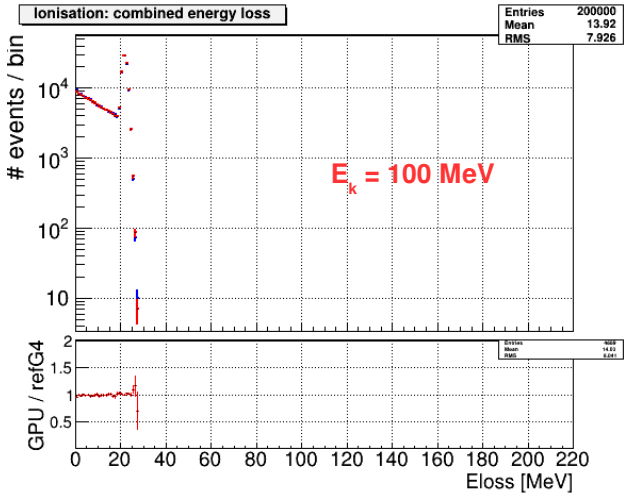


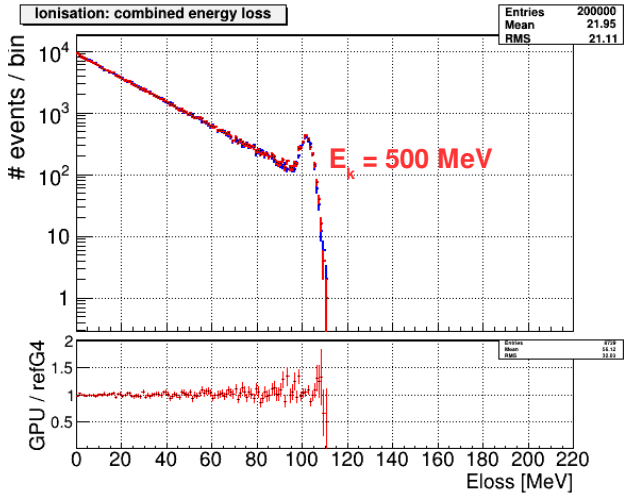


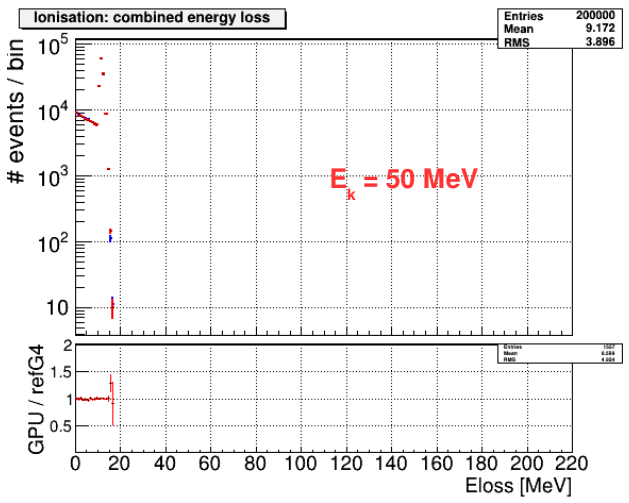


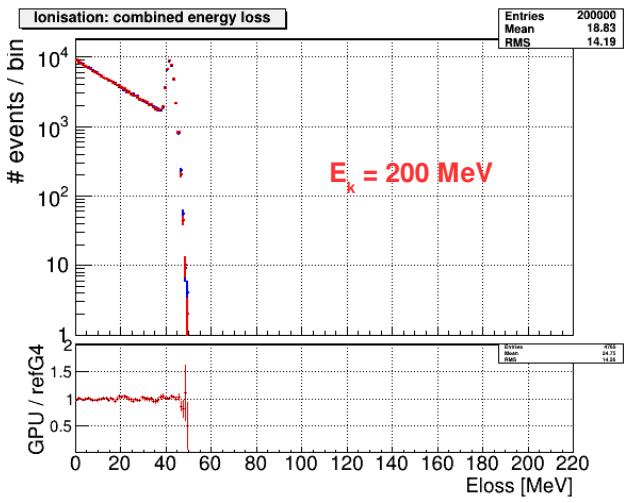
Ionisation: energy loss distributions – current status

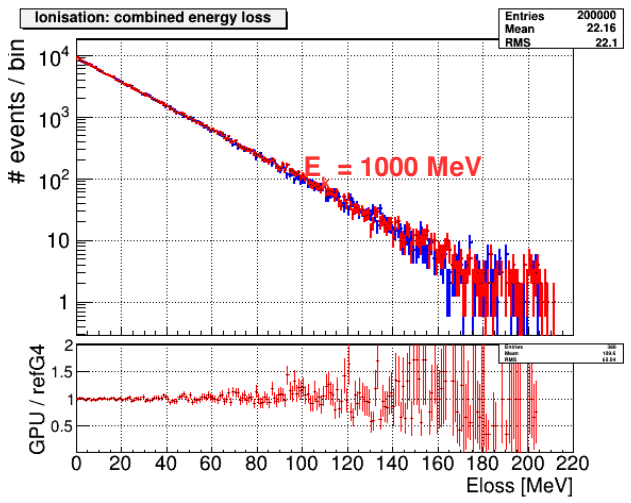






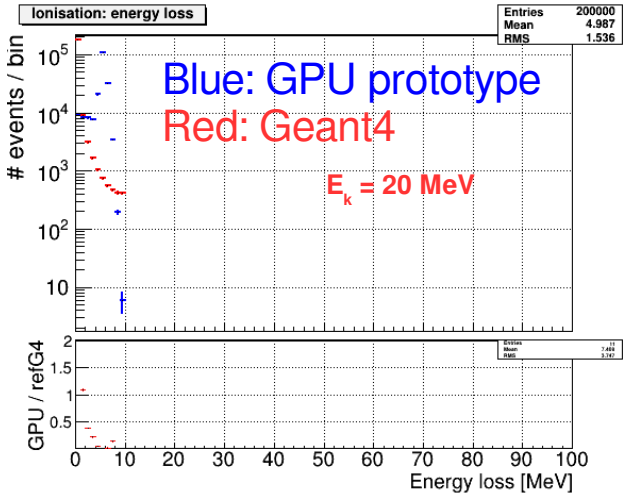


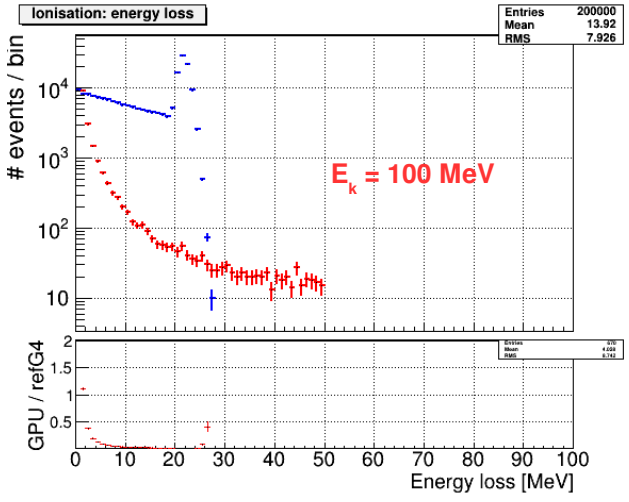


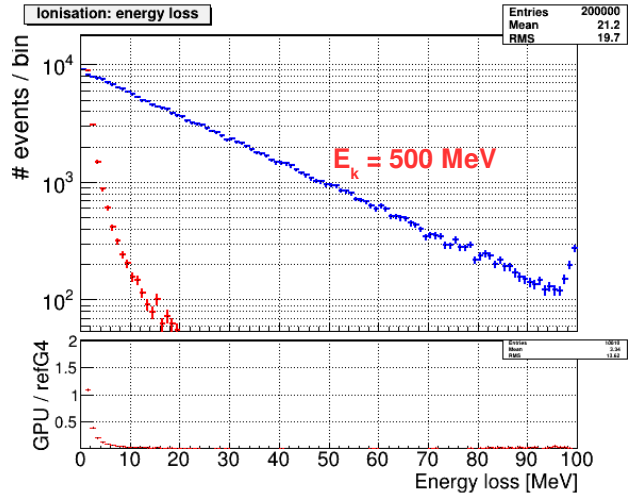


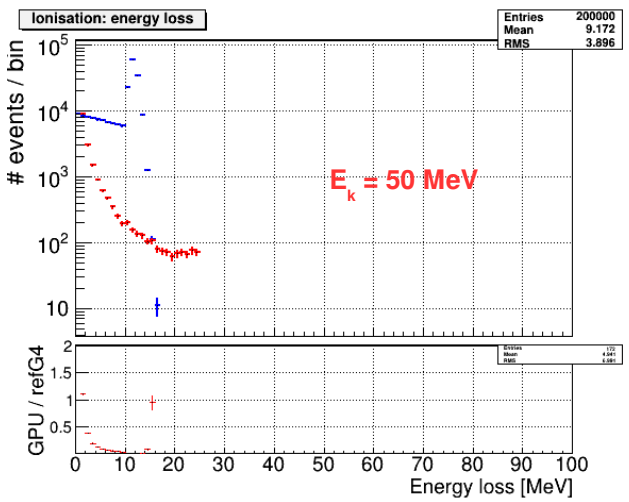
Inside AlongStepDoIt()

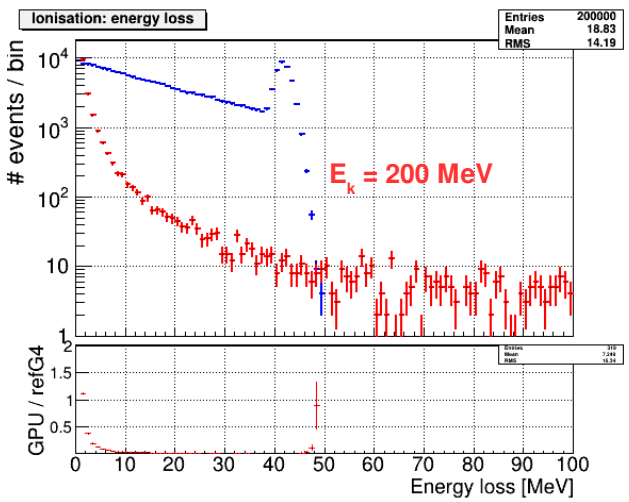
Ionisation: energy loss distributions – current status

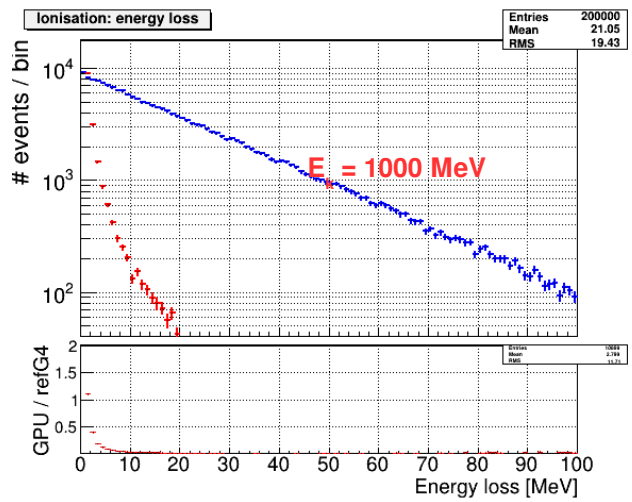












From SteppingAction(): inputKinE – outputKinE

Validation of physics processes in GPU prototype

- Next steps
  - Finish investigation on electron ionisation
  - Validate remaining processes:
    - Bremsstrahlung (done)
    - Ionisation (under way)
    - Multiple scattering
    - Compton scattering
    - Photo-electric effect
    - e^+e^- pair production

- **However** this validation work will be done with lower priority, without interfering with work on geometry vectorization developments.