An Event Timer for the merittuple

Leon R.

Core EVO Meeting

11 December 2007

Why?

- We’ve often wished that we could easily find the events that take a long time.
  - Can they be screened out? The most time-consuming events are probably useless, but we don’t know for sure unless we can look.

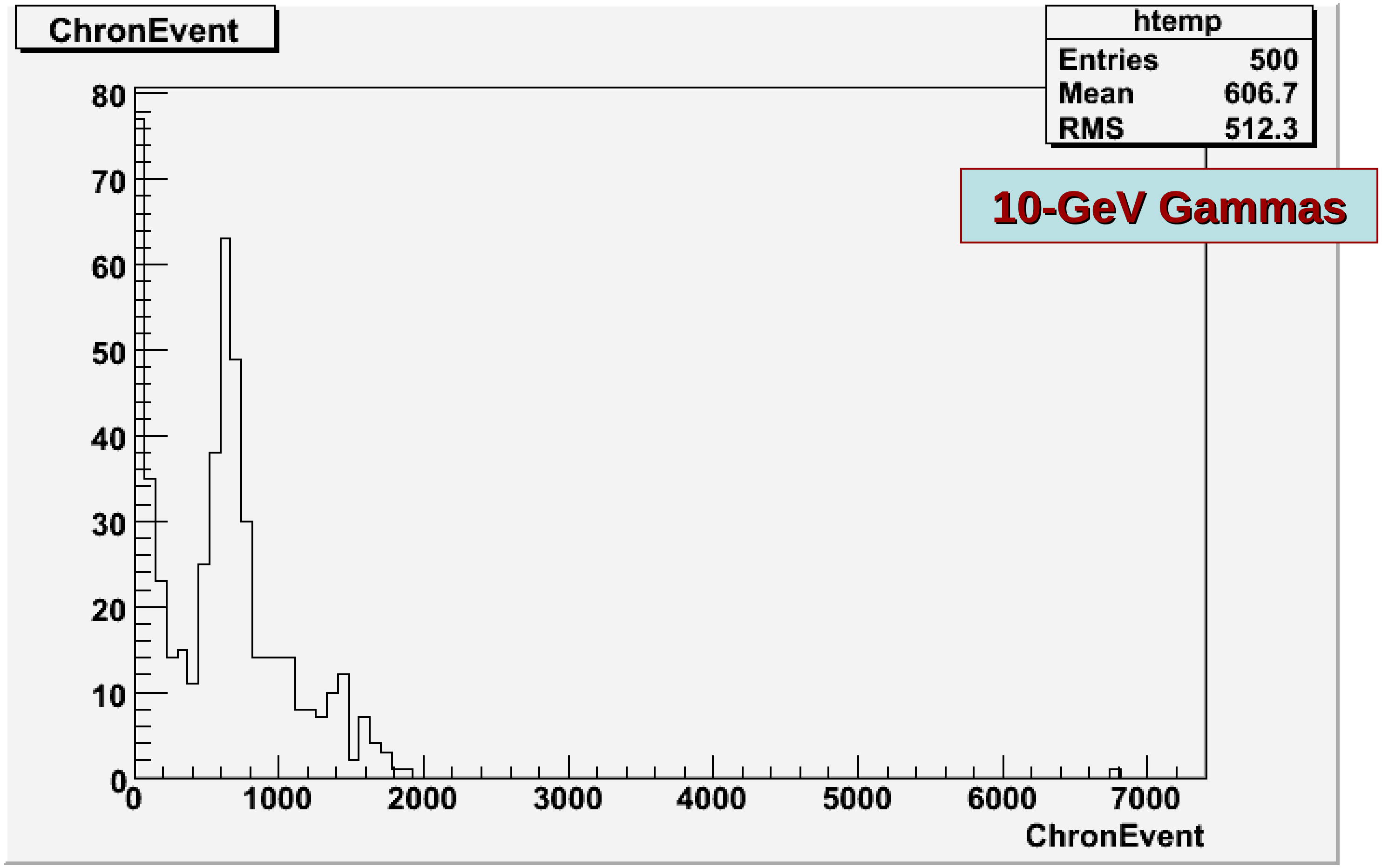
- It would be nice to be able to measure the cpu time for events of a certain type, without having to generate separate datasets.

- There are probably some surprises!

Gaudi Auditor Class

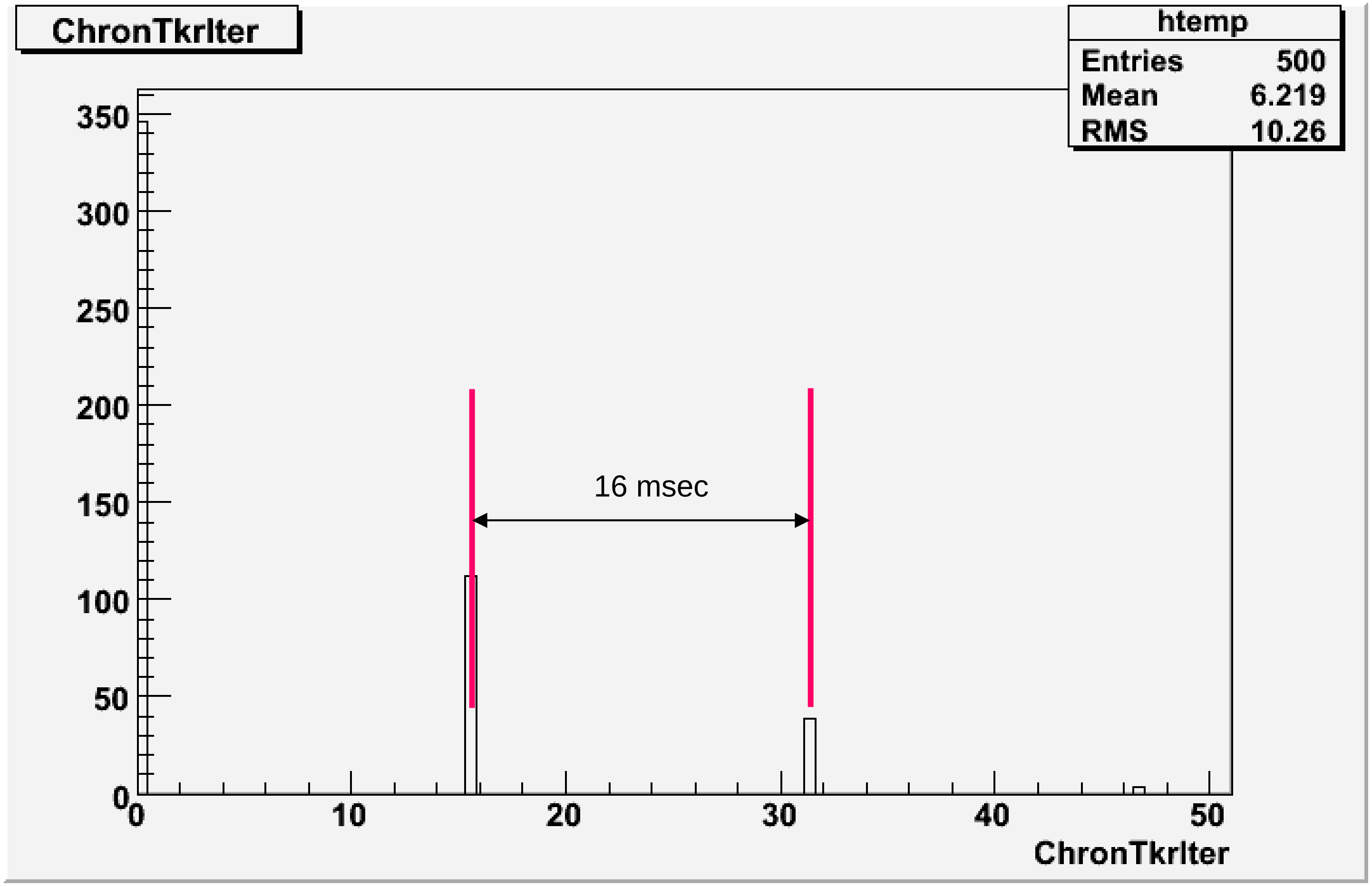
- Many thanks to David Chamont. He wrote a demonstration routine using the Auditor class after I asked him to look into this possibility. (This was a long time ago!)
  - I’ve cloned David’s routine in the context of the merittuple.
- Similar to an IncidentListener
  - The Auditor is called before and after the execution of each Algorithm or sequence.
  - The Auditor can start and stop timers.
- Since the merittuple is written out at the end of the event, the sequences/algorithms are all done by the time the merittuple row gets written.

ChronEvent, msec



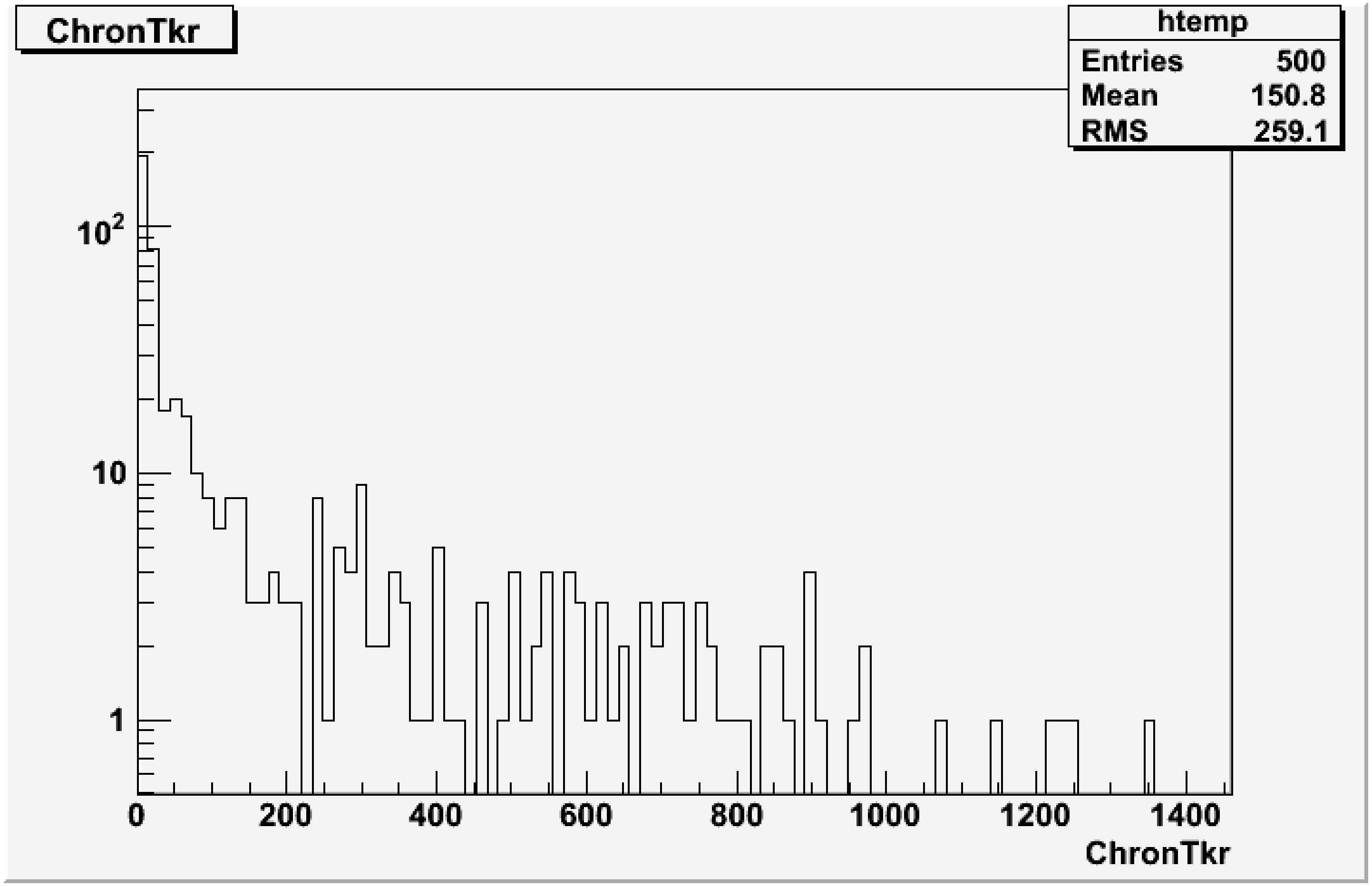
10-GeV Gammas

But… Coarse Granularity!



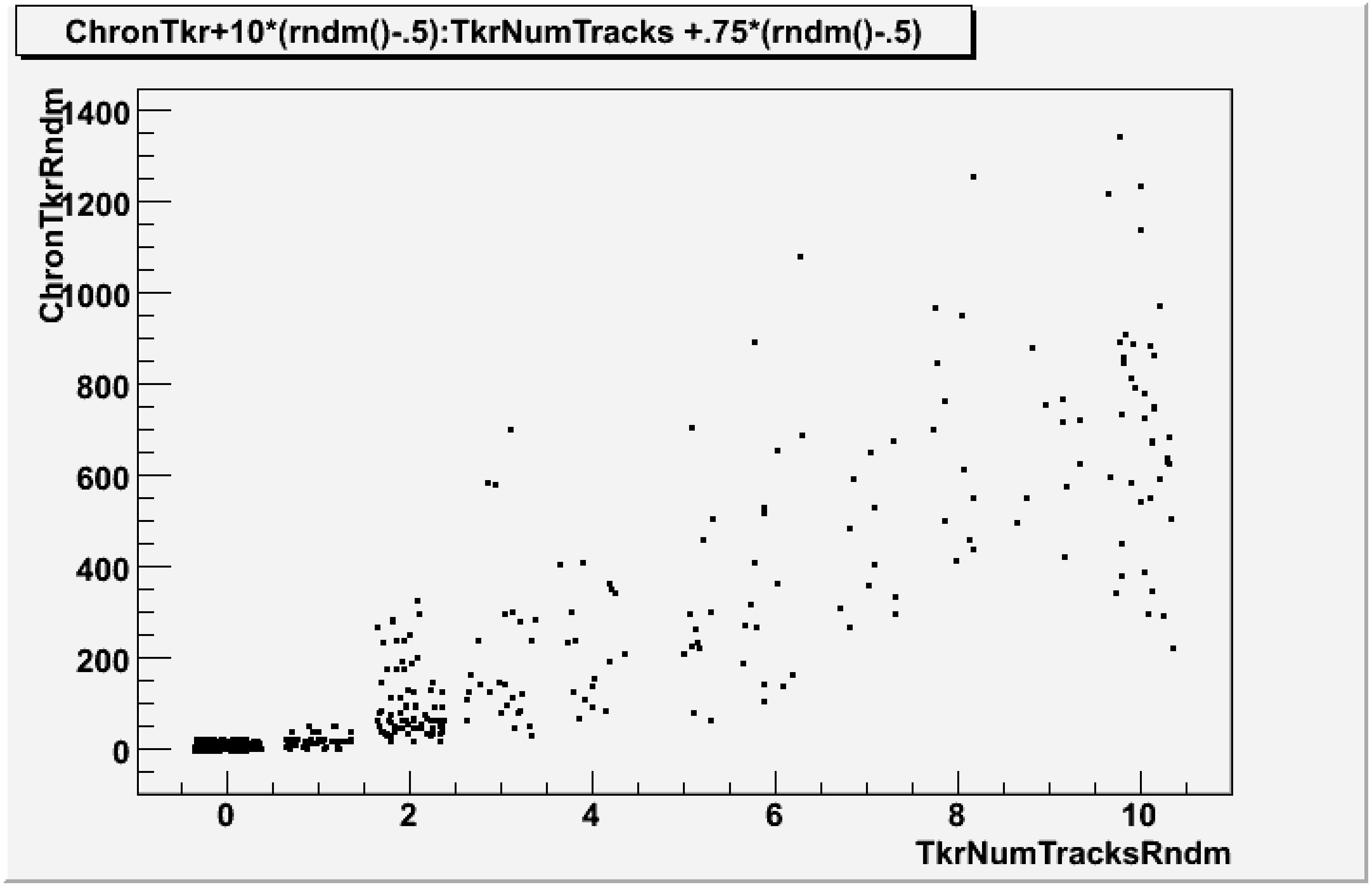
Iteration of TkrRecon, showing timing granularity.
(I haven’t checked this on linux.)

TkrRecon FirstPass (Tkr)

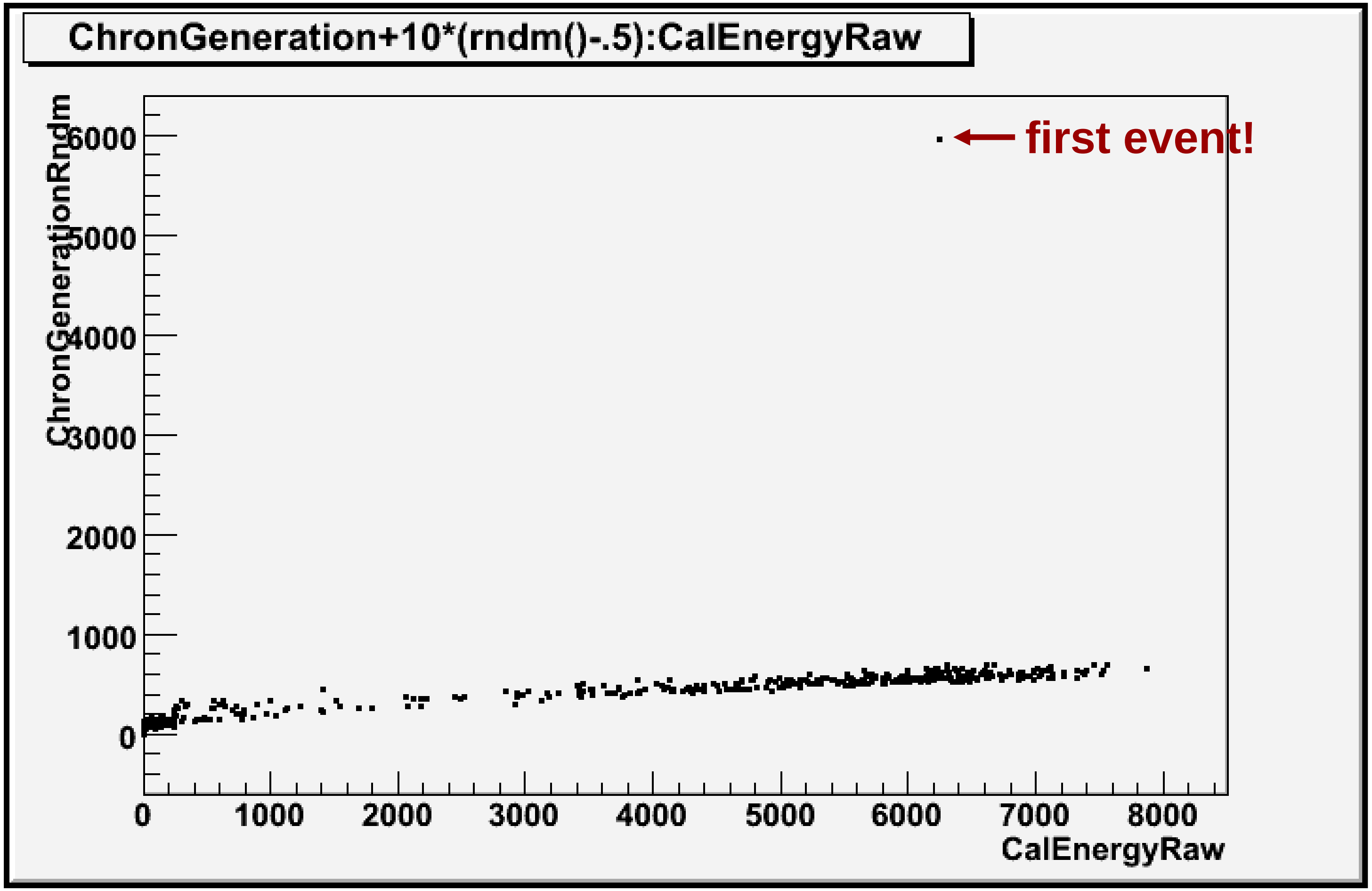


TkrRecon, PatRec Step

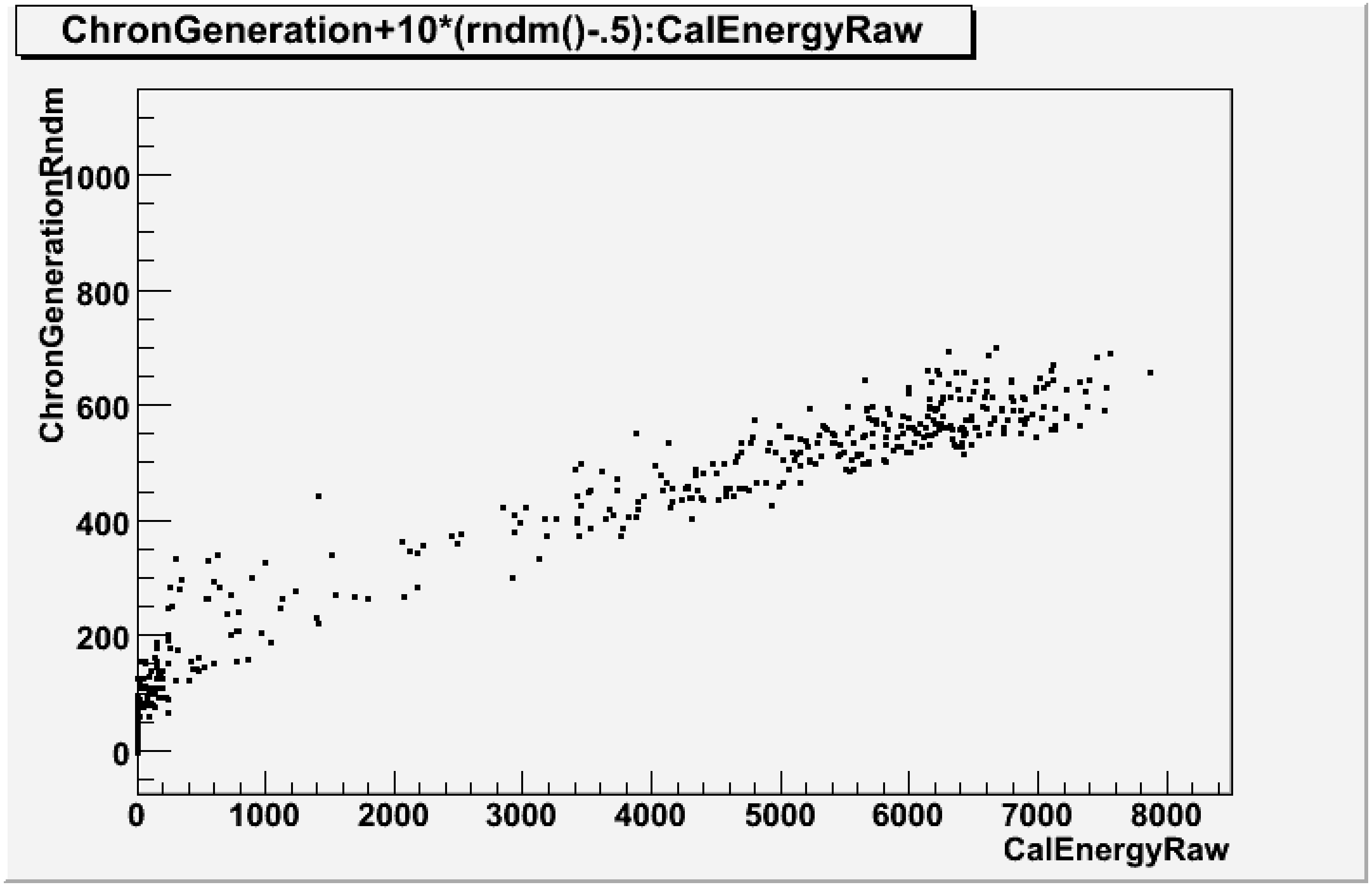
ChronTkr vs TkrNumTracks



ChronGen vs CalEnergyRaw



ChronGen vs CalEnergyRaw (zoom)



What Next?

- Timing information seems like a useful addition to the merittuple…
- but it would violate the rule that the merittuple be deriveable from the output root files.
- This might be a good time to implement a “diagnostics” class in Event for this kind of information.