

Speeding Up the Hadronic Code

Dennis Wright

Geant4 Collaboration Workshop

19 October 2009

Motivation

- Hadronic interactions, despite being a relatively small part of showers, still take up a lot of CPU time
- There are likely significant time savings if we look
 - Bertini improvements by FNAL, for example
- Not just speed but memory usage should be looked at
 - ATLAS reports 1.3 Gb of memory transactions in Bertini

Codes to Examine

- Precompound
- Binary
- Bertini
- INCL/ABLA
- QMD
- NeutronHP
- Some of the slowness may be due to detailed physics
 - not much we can do about that
 - but could make alternative faster model if needed

What to do?

- **Adopt faster algorithms**
 - faster version of `std::pow`, or other special functions
 - use look-up tables
 - `G4PhysicsVector`?
- **Use better programming practices**
 - don't use string comparisons
 - check loops for repeated code
 - avoid frequent newing and deleting
- **Profile all hadronic codes**
- **Other ideas?**