# Problems in Hadronic Interfaces

Dennis Wright
Geant4 Collaboration Meeting
20 September 2011

# Outline

- Migration to Integer Z and A
- Interfaces between cascade and string models
- Interfaces between precompound and cascade models
- Smooth model transition
- Automatic documentation

# Integer Z and A

- Two years ago, we adopted the convention that all hadronic interfaces should use integer values of Z and A to represent nuclei and cross sections
  - Hadronic cross sections have implemented in this way
  - Migration of G4Nucleus underway
  - Anything else left to be done?

# Intefaces Between Cascade and String Models

- FTF-Bertini
  - Uses Propagate() interface and new Bertini method reScatter()
  - Problems:
    - FTF may generate particle types that Bertini doesn't recognize
    - it used to be that once FTF passed its fragments to Bertini, residual nucleus is too high in energy to decay by precompund model – this is now fixed, but how does validation look?
- FTF-Binary
    - same problem with highly excited residual – same fix possible?
    - Binary only handles pions up to 1.5 GeV
- QGS-Binary
- QGS-Bertini

# Intefaces Between Precompound and Cascade Models

- Precompound-Bertini interface seems to work well below 60 MeV
- Poor agreement with data above 60 MeV
- Why? Should be OK up to ~170 MeV
  - fix of Bertini “do-nothing” output may have fixed this
  - re-check agreement with data

# How to Make Smooth Transition Between Hadronic Models

- Current method: linear merge in energy
  - throw a random number over energy range where models overlap
- Not the best approach
  - No physical basis, range of energy blending treated as a parameter
  - Can lead to discontinuities in other observables (angular distribution, energy desposit, etc.)
- What other approaches can we take?
  - Impact parameter
  - Formation time
  - Non-linear energy merge
  - Do precompound-to-cascade and cascade-to-string interfaces help?

# Automatic Documentation

- Automatic physics list documentation implemented
  - See Physics List in plenary session 8
  - currently requires user to set two environment variables specifying:
    - physics list name
    - location of directory where all html description files are kept
- Description() method generates html file for each component – each code author must implement
- G4HadronicProcessStore currently does most of the work
  - Method DumpHtml() writes physics list file
  - Method PrintHtml() collects descriptions of all processes, mosdels and cross sections for a given particle type
- User must run the physics list in an application to get html file