

Geant4

PhysicsList

Leo Piilonen, Virginia Tech

Why PhysicsList?

- Defines particles for geant as well as the specific processes for each particle (energy loss, multiple scattering, pair production, ... in addition to “transportation”)
- **GEANT4** [in `g4superb init()`]:
`m_run_manager->SetUserInitialization(new B4PhysicsList);`
- **GEANT4e** [in `g4ext init()` call to `InitGeant4e`]:
`theG4RunManagerKernel->SetPhysics(new G4ErrorPhysicsList);`
or
`theG4RunManagerKernel->SetPhysics(theUserPhysicsList);`

PhysicsList is a concrete implementation of class G4VUserPhysicsList, and must define:

- ConstructParticle()
- ConstructProcess()
- SetCuts()

These methods will be called by geant RunManager at the start of a run or whenever user says that PhysicsList has changed.

Lots of overhead to change PhysicsList (e.g., when switching between g4superb and g4ext in a basf job), so avoid this!

B4PhysicsList in [old] g4superb:

- ConstructParticle() defines B+ B- B0 Bs0 D+ D- D0 Ds+ Ds-
GenericIon He3 J/psi alpha anti_B0 anti_Bs0 anti_D0
anti_kaon0 anti_lambda anti_lambda_c+ anti_neutron
anti_nu_e anti_nu_mu anti_nu_tau anti_omega- anti_omega_c0
anti_proton anti_sigma+ anti_sigma- anti_sigma0 anti_sigma_c+
anti_sigma_c++ anti_sigma_c0 anti_xi- anti_xi0 anti_xi_c+
anti_xi_c0 chargedgeantino deuteron e+ e- eta eta_prime
gamma geantino kaon+ kaon- kaon0 kaon0L kaon0S lambda
lambda_c+ mu+ mu- neutron nu_e nu_mu nu_tau omega-
omega_c0 opticalphoton pi+ pi- pi0 proton sigma+ sigma- sigma0
sigma_c+ sigma_c++ sigma_c0 tau+ tau- triton xi- xi0 xi_c+ xi_c0
- ConstructProcess() does AddTransportation(),
ConstructDecayProcess(), ConstructEMProcess(),
ConstructHadronicProcess(), ConstructOpticalPhotonProcess()
for the relevant particles in the above list
- SetCuts() does SetCutsWithDefault() using default = 1.0*mm

B4ExtPhysicsList in [old] g4ext:

- ConstructParticle() defines e+ e- gamma mu+ mu- proton (and pi+ pi- kaon+ kaon- added by LEP). (gamma is never used, but geant4 fails if it isn't defined)
- ConstructProcess() does AddTransportation() and ConstructEM() for all particles in the above list. ConstructEM() defines only ionization energy loss and adds B4ExtNoHits() to avoid creating hits in sensitive detectors.
- SetCuts() does SetCutsWithDefault() using default = 1.0E9*cm

Lots of overhead to change between B4PhysicsList and B4ExtPhysicsList between each basf event, so avoid this!

Merge B4PhysicsList and B4ExtPhysicsList.

B4PhysicsList in [new] g4superb:

- ConstructParticle() defines all of the “regular” particles plus g4e_e+ g4e_e- g4e_gamma g4e_mu+ g4e_mu- g4e_proton g4e_pi+ g4e_pi- g4e_kaon+ g4e_kaon-; each of these “g4ext” particles has PIDcode = 1000000000 + stdPIDcode
- ConstructProcess() does AddTransportation(), ConstructDecayProcess(), ConstructEMProcess(), ConstructHadronicProcess(), ConstructOpticalPhotonProcess() for the relevant particles in the above list. For the “g4ext” particles, ConstructEMProcess() defines only ionization energy loss and adds B4ExtNoHits() to avoid creating hits in sensitive detectors.
- SetCuts() does SetCutsWithDefault() using default = 1.0*mm, then does SetParticleCuts() for “g4ext” particles using cut = 1.0E9*cm

Common B4PhysicsList is used by g4superb and by g4ext without interference.