Subject: Problems with FTF/DPM with Geant4
Posted by StefanoSpataro on Wed, 01 Apr 2015 15:30:43 GMT
View Forum Message <> Reply to Message

Dear DPM and FTF experts,
trying to use these event generators with new packages I have found several problems once using geant4, while with geant3 everything seems fine.

In particular, if I run dpm direct with geant4 (macro/qa/dpm4/sim_complete.C) I have the following error once TGeant4 is created (I believe):

Info in <TG4RootNavMgr::SetNavigator>: TG4RootNavigator created and registered to G4TransportationManager

-------- EEEE ------- G4Exception-START -------- EEEE -------
*** G4Exception : PART002
issued by : G4ParticleTable::CheckReadiness()
Illegal use of G4ParticleTable : Access to G4ParticleTable for finding a particle or equivalent operation occurs before G4VUserPhysicsList is instantiated and
assigned to G4RunManager. Such an access is prohibited by
Geant4 version 8.0. To fix this problem, please make sure that
your main() instantiates G4VUserPhysicsList and set it to
G4RunManager before instantiating other user classes such as
G4VUserPrimaryParticleGeneratorAction.
*** Fatal Exception *** core dump ***
-------- EEEE -------- G4Exception-END --------- EEEE -------

*** G4Exception: Aborting execution ***

If I try ftf with geant4 (macro/qa/ftf4/sim_complete.C):

Info in <TG4RootNavMgr::SetNavigator>: TG4RootNavigator created and registered to G4TransportationManager

-------- EEEE ------- G4Exception-START -------- EEEE -------
*** G4Exception : Run0002
issued by : G4RunManagerKernel::G4RunManagerKernel()
!!!
G4RunManagerKernel fatal exception
-- Following particles have already been registered
before G4RunManagerKernel is instantiated.
B+
B-
B0
Bc+

Toggle Spoiler

Bc-
Bs0

D+
D-
D0
Ds+
Ds-
GenericIon
He3
J/psi
N(1440)+
N(1440)0
N(1520)+
N(1520)0
N(1535)+
N(1535)0
N(1650)+
N(1650)0
N(1675)+
N(1675)0
N(1680)+
N(1680)0
N(1700)+
N(1700)0
N(1710)+
N(1710)0
N(1720)+
N(1720)0
N(1900)+
N(1900)0
N(1990)+
N(1990)0
N(2090)+
N(2090)0
N(2190)+
N(2190)0
N(2220)+
N(2220)0
N(2250)+
N(2250)0
Upsilon
a0(1450)+
a0(1450)-
a0(1450)0
a0(980)+
a0(980)-
a0(980)0
a1(1260)+
a1(1260)-
a1(1260)0
a2(1320)+
a2(1320)-
a2(1320)0
alpha

anti_B0
anti_Bs0
anti_D0
anti_He3
anti_N(1440)+
anti_N(1440)0
anti_N(1520)+
anti_N(1520)0
anti_N(1535)+
anti_N(1535)0
anti_N(1650)+
anti_N(1650)0
anti_N(1675)+
anti_N(1675)0
anti_N(1680)+
anti_N(1680)0
anti_N(1700)+
anti_N(1700)0
anti_N(1710)+
anti_N(1710)0
anti_N(1720)+
anti_N(1720)0
anti_N(1900)+
anti_N(1900)0
anti_N(1990)+
anti_N(1990)0
anti_N(2090)+
anti_N(2090)0
anti_N(2190)+
anti_N(2190)0
anti_N(2220)+
anti_N(2220)0
anti_N(2250)+
anti_N(2250)0
anti_alpha
anti_b_quark
anti_c_quark
anti_d_quark
anti_dd1_diquark
anti_delta(1600)+
anti_delta(1600)++
anti_delta(1600)-
anti_delta(1600)0
anti_delta(1620)+
anti_delta(1620)++
anti_delta(1620)-
anti_delta(1620)0
anti_delta(1700)+
anti_delta(1700)++
anti_delta(1700)-
anti_delta(1700)0
anti_delta(1900)+

anti_delta(1900)++
anti_delta(1900)-
anti_delta(1900)0
anti_delta(1905)+
anti_delta(1905)++
anti_delta(1905)-
anti_delta(1905)0
anti_delta(1910)+
anti_delta(1910)++
anti_delta(1910)-
anti_delta(1910)0
anti_delta(1920)+
anti_delta(1920)++
anti_delta(1920)-
anti_delta(1920)0
anti_delta(1930)+
anti_delta(1930)++
anti_delta(1930)-
anti_delta(1930)0
anti_delta(1950)+
anti_delta(1950)++
anti_delta(1950)-
anti_delta(1950)0
anti_delta+
anti_delta++
anti_delta-
anti_delta0
anti_deuteron
anti_k(1460)0
anti_k0_star(1430)0
anti_k1(1270)0
anti_k1(1400)0
anti_k2(1770)0
anti_k2_star(1430)0
anti_k2_star(1980)0
anti_k3_star(1780)0
anti_k_star(1410)0
anti_k_star(1680)0
anti_k_star0
anti_kaon0
anti_lambda
anti_lambda(1405)
anti_lambda(1520)
anti_lambda(1600)
anti_lambda(1670)
anti_lambda(1690)
anti_lambda(1800)
anti_lambda(1810)
anti_lambda(1820)
anti_lambda(1830)
anti_lambda(1890)
anti_lambda(2100)

anti_lambda(2110)
anti_lambda_b
anti_lambda_c+
anti_neutron
anti_nu_e
anti_nu_mu
anti_nu_tau
anti_omega-
anti_omega_b-
anti_omega_c0
anti_proton
anti_s_quark
anti_sd0_diquark
anti_sd1_diquark
anti_sigma(1385)+
anti_sigma(1385)-
anti_sigma(1385)0
anti_sigma(1660)+
anti_sigma(1660)-
anti_sigma(1660)0
anti_sigma(1670)+
anti_sigma(1670)-
anti_sigma(1670)0
anti_sigma(1750)+
anti_sigma(1750)-
anti_sigma(1750)0
anti_sigma(1775)+
anti_sigma(1775)-
anti_sigma(1775)0
anti_sigma(1915)+
anti_sigma(1915)-
anti_sigma(1915)0
anti_sigma(1940)+
anti_sigma(1940)-
anti_sigma(1940)0
anti_sigma(2030)+
anti_sigma(2030)-
anti_sigma(2030)0
anti_sigma+
anti_sigma-
anti_sigma0
anti_sigma_b+
anti_sigma_b-
anti_sigma_b0
anti_sigma_c+
anti_sigma_c++
anti_sigma_c0
anti_ss1_diquark
anti_su0_diquark
anti_su1_diquark
anti_t_quark
anti_triton

anti_u_quark
anti_ud0_diquark
anti_ud1_diquark
anti_uu1_diquark
anti_xi(1530)-
anti_xi(1530)0
anti_xi(1690)-
anti_xi(1690)0
anti_xi(1820)-
anti_xi(1820)0
anti_xi(1950)-
anti_xi(1950)0
anti_xi(2030)-
anti_xi(2030)0
anti_xi-
anti_xi0
anti_xi_b-
anti_xi_b0
anti_xi_c+
anti_xi_c0
b1(1235)+
b1(1235)-
b1(1235)0
b_quark
c_quark
chargedgeantino
d_quark
dd1_diquark
delta(1600)+
delta(1600)++
delta(1600)-
delta(1600)0
delta(1620)+
delta(1620)++
delta(1620)-
delta(1620)0
delta(1700)+
delta(1700)++
delta(1700)-
delta(1700)0
delta(1900)+
delta(1900)++
delta(1900)-
delta(1900)0
delta(1905)+
delta(1905)++
delta(1905)-
delta(1905)0
delta(1910)+
delta(1910)++
delta(1910)-
delta(1910)0

delta(1920)+
delta(1920)++
delta(1920)-
delta(1920)0
delta(1930)+
delta(1930)++
delta(1930)-
delta(1930)0
delta(1950)+
delta(1950)++
delta(1950)-
delta(1950)0
delta+
delta++
delta-
delta0
deuteron
e+
e-
eta
eta(1295)
eta(1405)
eta(1475)
eta2(1645)
eta2(1870)
eta_prime
etac
f0(1370)
f0(1500)
f0(1710)
f0(600)
f0(980)
f1(1285)
f1(1420)
f2(1270)
f2(1810)
f2(2010)
f2_prime(1525)
gamma
geantino
gluon
h1(1170)
h1(1380)
k(1460)+
k(1460)-
k(1460)0
k0_star(1430)+
k0_star(1430)-
k0_star(1430)0
k1(1270)+
k1(1270)-
k1(1270)0

k1(1400)+
k1(1400)-
k1(1400)0
k2(1770)+
k2(1770)-
k2(1770)0
k2_star(1430)+
k2_star(1430)-
k2_star(1430)0
k2_star(1980)+
k2_star(1980)-
k2_star(1980)0
k3_star(1780)+
k3_star(1780)-
k3_star(1780)0
k_star(1410)+
k_star(1410)-
k_star(1410)0
k_star(1680)+
k_star(1680)-
k_star(1680)0
k_star+
k_star-
k_star0
kaon+
kaon-
kaon0
kaon0L
kaon0S
lambda
lambda(1405)
lambda(1520)
lambda(1600)
lambda(1670)
lambda(1690)
lambda(1800)
lambda(1810)
lambda(1820)
lambda(1830)
lambda(1890)
lambda(2100)
lambda(2110)
lambda_b
lambda_c+
mu+
mu-
neutron
nu_e
nu_mu
nu_tau
omega
omega(1420)

omega(1650)
omega-
omega3(1670)
omega_b-
omega_c0
opticalphoton
phi
phi(1680)
phi3(1850)
pi(1300)+
pi(1300)-
pi(1300)0
pi+
pi-
pi0
pi2(1670)+
pi2(1670)-
pi2(1670)0
proton
rho(1450)+
rho(1450)-
rho(1450)0
rho(1700)+
rho(1700)-
rho(1700)0
rho+
rho-
rho0
rho3(1690)+
rho3(1690)-
rho3(1690)0
s_quark
sd0_diquark
sd1_diquark
sigma(1385)+
sigma(1385)-
sigma(1385)0
sigma(1660)+
sigma(1660)-
sigma(1660)0
sigma(1670)+
sigma(1670)-
sigma(1670)0
sigma(1750)+
sigma(1750)-
sigma(1750)0
sigma(1775)+
sigma(1775)-
sigma(1775)0
sigma(1915)+
sigma(1915)-
sigma(1915)0

sigma(1940)+
sigma(1940)-
sigma(1940)0
sigma(2030)+
sigma(2030)-
sigma(2030)0
sigma+
sigma-
sigma0
sigma_b+
sigma_b-
sigma_b0
sigma_c+
sigma_c++
sigma_c0
ss1_diquark
su0_diquark
su1_diquark
t_quark
tau+
tau-
triton
u_quark
ud0_diquark
ud1_diquark
uu1_diquark
xi(1530)-
xi(1530)0
xi(1690)-
xi(1690)0
xi(1820)-
xi(1820)0
xi(1950)-
xi(1950)0
xi(2030)-
xi(2030)0
xi-
xi0
xi_b-
xi_b0
xi_c+
xi_c0

!!

*** Fatal Exception *** core dump ***
-------- EEEE -------- G4Exception-END --------- EEEE -------

*** G4Exception: Aborting execution ***

The latter case comes from the fact that FTF is using Geant4 particle table without the G4RunManagerKernel initialization. The first i case I do not understand since it seems to me the DPM code is not using geant4.

I would like to ask the experts to check and fix, in practice now we cannot produce background events with geant4.

Subject: Re: Problems with FTF/DPM with Geant4
Posted by StefanoSpataro on Tue, 21 Apr 2015 10:15:29 GMT
View Forum Message <> Reply to Message

Dear all,
I have isolated the problem, and it seems to be connected to FTF: FTF instantiate a g4 application, and geant4 finds an already existing g4 session and it crashes. Since FtfDirect and DpmDirect were in the same folder, they were interfering and also Dpm was crashing.
I have moved PndFtfDirect outside PGen inside FtfEvtGen package and now DPM can be used also with G4.
The problem of using FTF with Geant4 does persist, and I have no clue on how to fix it.

Subject: Re: Problems with FTF/DPM with Geant4
Posted by Lu Cao on Sun, 26 Mar 2017 15:19:19 GMT
View Forum Message <> Reply to Message

Dear all,

I'm trying to use FTF+Geant4 with the latest release Feb17p1, and unfortunately I found the exact same problem of FTF as posted here two years ago.

Is there any updates/clues from the experts???

Best regards,
Lu

Subject: Re: Problems with FTF/DPM with Geant4
Posted by Ralf Kliemt on Sun, 26 Mar 2017 16:14:36 GMT
View Forum Message <> Reply to Message

Hi Lu,

PndFtfDirect does not work with Geant4 in our simulations. Please use the FTFGen executable located in your build/bin folder to create en event file and read that in your simulation macro (PndFtfGenertor).

Cheers!

Ralf

Subject: Re: Problems with FTF/DPM with Geant4
Posted by Lu Cao on Mon, 27 Mar 2017 10:14:39 GMT
View Forum Message <> Reply to Message

Thanks, Ralf! Your suggestion works.

In case someone else may have the same problem as me in the future, I provide the usable sim_complete.C in the attachment.

Regards,
Lu

File Attachments
1) sim_complete.C, downloaded 456 times

Subject: Re: Problems with FTF/DPM with Geant4
Posted by Lu Cao on Mon, 27 Mar 2017 15:53:07 GMT
View Forum Message <> Reply to Message

Hi all,

there's a problematic issue on the "indirect" FTF+G4 event generator.

Our event filter cannot dynamically interact with it, therefore one has to prepare a large enough(?) event file beforehand to ensure the filter can select out something. This obviously reduces the convenience and functionality of event filter. If no sufficient events can be fed to the filter, the filter will just produce some fake events (no particles in event) according to the number of tries set by primGen->SetFilterMaxTries(). At the end, Geant4 is aborted due to "No primary particles found on the stack."

Is any smart ways to use FTF+G4+EventFilter ?

Best regards,
Lu

Subject: Re: Problems with FTF/DPM with Geant4
Posted by Tobias Stockmanns on Tue, 28 Mar 2017 06:36:09 GMT
View Forum Message <> Reply to Message

Dear Lu,

I do not see a simple solution for this problem because you do not know beforehand how many events you need when you use the filter. In addition there is no possibility to access the standalone FTF generator by the event filter.

Cheers,

Tobias

Subject: Re: Problems with FTF/DPM with Geant4
Posted by Lu Cao on Tue, 28 Mar 2017 08:29:29 GMT
View Forum Message <> Reply to Message

Dear Tobias,

Yes, I couldn't know how many events I need beforehand, but this is the exact reason why the filter is designed to call generator to produce events until the number of events set by user has been found, as far as I learn from https://
panda-wiki.gsi.de/foswiki/bin/view/Computing/PandaRootEventFilterTutoria l . The filter can call the other generators, e.g. DpmDirect, EvtGenDirect, because they are "direct". The main problem is how to make FTF+G4 direct, or find out an alternative method.

Best regards,
Lu

Subject: Re: Problems with FTF/DPM with Geant4
Posted by StefanoSpataro on Tue, 28 Mar 2017 08:45:13 GMT
View Forum Message <> Reply to Message

The philosophy should be the same as done for the background of the tracking TDRs:
Estimate how many events you need to have a decent statistics, run a very big data production only at the generator level, which in any case produces a small amount of data, and run from them your required number of events.
I.e., if your filter skims with a factor 1%, and you need 100k skimmed events, run 2M events to have a safety factor or 2.
Until somebody really spend some time to fix this FTF+G4 problem, this is the only chance. In the past I spent some time many times, but I was never able to find the clue. The solution would be to rewrite FTF generator so that it does not use g4 classes as it is doing now.
