Subject: segmentation violation in FairPrimaryGenerator
Posted by Anastasia Karavdina on Fri, 28 Sep 2012 22:11:59 GMT
View Forum Message <> Reply to Message

Dear all,
I updated pandaroot today and have segmentation violation during MC simulation.
Output looks like this:

[INFO] Initialize Tasks-------------------------
[INFO] Simulation RunID: 1723011043
[INFO] Monte carlo Engine Initialisation with : TGeant3TGeo
[INFO] FairPrimaryGenerator: (Event 1) 2 primary tracks from vertex (0.000000, 0.000000, 0.000000) Event Time = 0.000000 (ns)

*** Break *** segmentation violation
Generating stack trace...
0x00002b7e30f94a82 in TGeant3::ProcessRun(int) + 0x62 from /cluster/gsi/fairsoft/gcc/fairsoft_sep12/lib/libgeant321.so
0x00002b7e2765aedc in FairMCApplication::RunMC(int) at /home/karavdin/pandaRoot12/base/FairMCApplication.cxx:237 from /home/karavdin/pandaRoot12/build/lib/libBase.so
0x00002b7e27691dc2 in FairRunSim::Run(int, int) at /home/karavdin/pandaRoot12/base/FairRunSim.cxx:352 from /home/karavdin/pandaRoot12/build/lib/libBase.so
0x00002b7e2770299f in <unknown> from /home/karavdin/pandaRoot12/build/lib/libBase.so
0x00002b7e1b2e3069 in Cint::G__ExceptionWrapper(int (*)(G__value*, char const*, G__param*, int), G__value*, char*, G__param*, int) + 0x39 from /cluster/gsi/fairsoft/gcc/fairsoft_sep12/lib/root/libCint.so.5.34
0x00002b7e1b386211 in G__execute_call + 0x61 from /cluster/gsi/fairsoft/gcc/fairsoft_sep12/lib/root/libCint.so.5.34
0x00002b7e1b387072 in G__call_cppfunc + 0x292 from /cluster/gsi/fairsoft/gcc/fairsoft_sep12/lib/root/libCint.so.5.34
0x00002b7e1b3642d8 in G__interpret_func + 0x1d98 from /cluster/gsi/fairsoft/gcc/fairsoft_sep12/lib/root/libCint.so.5.34
0x00002b7e1b35201c in G__getfunction + 0x18cc from /cluster/gsi/fairsoft/gcc/fairsoft_sep12/lib/root/libCint.so.5.34
0x00002b7e1b44344a in G__getstructmem(int, G__FastAllocString&, char*, int, char*, int*, G__var_array*, int) + 0x6ea from /cluster/gsi/fairsoft/gcc/fairsoft_sep12/lib/root/libCint.so.5.34
0x00002b7e1b43bc1e in G__getvariable + 0x429e from /cluster/gsi/fairsoft/gcc/fairsoft_sep12/lib/root/libCint.so.5.34
0x00002b7e1b32f38d in G__getitem + 0xed from /cluster/gsi/fairsoft/gcc/fairsoft_sep12/lib/root/libCint.so.5.34
0x00002b7e1b334698 in G__getexpr + 0x4058 from /cluster/gsi/fairsoft/gcc/fairsoft_sep12/lib/root/libCint.so.5.34
0x00002b7e1b3b405f in G__exec_statement + 0x638f from /cluster/gsi/fairsoft/gcc/fairsoft_sep12/lib/root/libCint.so.5.34
0x00002b7e1b3656b5 in G__interpret_func + 0x3175 from /cluster/gsi/fairsoft/gcc/fairsoft_sep12/lib/root/libCint.so.5.34
0x00002b7e1b35207b in G__getfunction + 0x192b from /cluster/gsi/fairsoft/gcc/fairsoft_sep12/lib/root/libCint.so.5.34
0x00002b7e1b32fb0e in G__getitem + 0x86e from /cluster/gsi/fairsoft/gcc/fairsoft_sep12/lib/root/libCint.so.5.34

0x00002b7e1b334698 in G__getexpr + 0x4058 from
/cluster/gsi/fairsoft/gcc/fairsoft_sep12/lib/root/libCint.so.5.34
0x00002b7e1b33db98 in G__calc_internal + 0x3f8 from
/cluster/gsi/fairsoft/gcc/fairsoft_sep12/lib/root/libCint.so.5.34
0x00002b7e1b3c3556 in G__process_cmd + 0x4ec6 from
/cluster/gsi/fairsoft/gcc/fairsoft_sep12/lib/root/libCint.so.5.34
0x00002b7e199bedb6 in TCint::ProcessLine(char const*, TInterpreter::EErrorCode*) + 0x536
from /cluster/gsi/fairsoft/gcc/fairsoft_sep12/lib/root/libCore.so.5.34
0x00002b7e199bc733 in TCint::ProcessLineSynch(char const*, TInterpreter::EErrorCode*) +
0x103 from /cluster/gsi/fairsoft/gcc/fairsoft_sep12/lib/root/libCore.so.5.34
0x00002b7e1992b1a8 in TApplication::ExecuteFile(char const*, int*, bool) + 0x848 from
/cluster/gsi/fairsoft/gcc/fairsoft_sep12/lib/root/libCore.so.5.34
0x00002b7e19929263 in TApplication::ProcessLine(char const*, bool, int*) + 0x7e3 from
/cluster/gsi/fairsoft/gcc/fairsoft_sep12/lib/root/libCore.so.5.34
0x00002b7e1a171468 in TRint::Run(bool) + 0x4d8 from
/cluster/gsi/fairsoft/gcc/fairsoft_sep12/lib/root/libRint.so.5.34
0x000000000040104c in main + 0x4c from /cluster/gsi/fairsoft/gcc/fairsoft_sep12/bin/root.exe
0x00002b7e1aaf3c5d in __libc_start_main + 0xfd from /lib64/libc.so.6
0x0000000000400f19 in <unknown> from /cluster/gsi/fairsoft/gcc/fairsoft_sep12/bin/root.exe
Function runLumi0DPMbkg() busy flag cleared

MZSTOR. ZEBRA table base TAB(0) in /MZCC/ at adr 206707327 C521A7F HEX

MZSTOR. Initialize Store 0 in /GCBANK/
with Store/Table at absolute adrs 206816669 206707327
HEX C53C59D C521A7F
HEX 1A72A 0
relative adrs 108330 0
with 1 Str. in 2 Links in 5300 Low words in 4999970 words.
This store has a fence of 16 words.

MZLOGL. Set Log Level 0 for store 0
1***** GEANT Version 3.21/11 Released on 100298
0***** Correction Cradle Version 0.1100

MZDIV. Initialize Division Constant in Store 0
NW/NWMAX= 20004000000, MODE/KIND= 1 2
Division 20 initialized.

MZLINK. Initialize Link Area /GCLINK/ for Store 0 NL/NS= 20 20

MZLINK. Initialize Link Area /GCSLNK/ for Store 0 NL/NS= 100 100

Calculating cross section tables, see gphysi.dat for more information

Cross section calculation concluded successfully
**** GTRIGI: IEVENT= 1 IDEVT= 1 Random Seeds = 4357 0

Does somebody have any glue what is the problem?

Subject: Re: segmentation violation in FairPrimaryGenerator
Posted by StefanoSpataro on Sat, 29 Sep 2012 08:56:53 GMT
View Forum Message <> Reply to Message

If you run again the same macro you have always the same error?

Subject: Re: segmentation violation in FairPrimaryGenerator
Posted by Anastasia Karavdina on Mon, 01 Oct 2012 07:04:55 GMT
View Forum Message <> Reply to Message

I have this crash 9 times from 10 tries.

Subject: Re: segmentation violation in FairPrimaryGenerator
Posted by StefanoSpataro on Mon, 01 Oct 2012 07:29:52 GMT
View Forum Message <> Reply to Message

Which macro, trunk version and OS? I am not able to reproduce the crash with the standard macro/pid/run_sim_sttcombi_pgun.C

Subject: Re: segmentation violation in FairPrimaryGenerator
Posted by Anastasia Karavdina on Mon, 01 Oct 2012 07:43:59 GMT
View Forum Message <> Reply to Message

Trunk:17198
It's macro/lmd/runLumi0DPMbkg.C
It's needed DPM input as a root file. Also one need to run macro/lmd/FirstRun to produce lumi geometry.
I'm running it on HIMster, so OS: Scientific Linux release 6.0 (Carbon) x86_64

PS: seems I don't have this problem with Box Generator

Subject: Re: segmentation violation in FairPrimaryGenerator
Posted by StefanoSpataro on Mon, 01 Oct 2012 12:34:17 GMT
View Forum Message <> Reply to Message

Hi,
I have no macro/lmd/runLumi0DPMbkg.C macro in my trunk, I have only the FirstRun.

Subject: Re: segmentation violation in FairPrimaryGenerator
Posted by Anastasia Karavdina on Mon, 01 Oct 2012 13:08:44 GMT
View Forum Message <> Reply to Message

Can you, please, update your macro/lmd?
I upload this macro today in the morning, it should be in the trunk now.

Subject: Re: segmentation violation in FairPrimaryGenerator
Posted by StefanoSpataro on Mon, 01 Oct 2012 14:51:20 GMT
View Forum Message <> Reply to Message

Dear Anastasia,
I find your macro quite obsolete, with old geometries, with the non-automatic construction of the field, and you use the old DPM generator and not the direct one. I am not able to run such a macro.

I commit in the same folder a runLumi0DPMbkgSte.C macro with the new things, in order to avoid old inheritances.

When I run such a macro, I had an error on the "EndOfEvent".
I corrected it by modifying line 213 of my local PndSdsDetector.cxx with:

if (fPndSdsCollection) fPndSdsCollection->Delete();

Could you please try such macro and see what happens? If you have such a crash, just to my modification, it is not yet in svn.

By the way, I compared PndLmdDetector with PndMvdDetector and I can see you fill the constructor and the destructor, while in Mvd they are left empty.
I believe they should be left empty, considering that the job is done inside PndSdsDetector, then you don't have to define your SdsCollection neither destroy it, SdsDetector is doing already such a job. And you don't have to delete fGeoH.

Please try and let me know.

Subject: Re: segmentation violation in FairPrimaryGenerator
Posted by Anastasia Karavdina on Mon, 01 Oct 2012 16:32:49 GMT
View Forum Message <> Reply to Message

Dear Stefano,
Thank you for looking in lmd code. Yeah, it's quite old and is needed to be cleaned\updated.
Concerning DPM we are using stand alone version, because it's more flexible for our studies.
Any way, I tried your macro. Good news: it doesn't crash.
Bad news: It stopped after

[INFO   ] Monte carlo Engine Initialisation with : TGeant3TGeo
And doesn't go to end of run.

On my local machine with older trunk version I can see what execution blocked before comes out following line:
[INFO   ] *** PndSensorNamePar written to ROOT file   version: 1

I agree with you, most probably problem isn't in svn, but in my local version or maybe because of their interference, but I really have no glue where to look.
I'll go through all changes which I did recently and write if I find something.

Subject: Re: segmentation violation in FairPrimaryGenerator
Posted by StefanoSpataro on Mon, 01 Oct 2012 16:39:11 GMT
View Forum Message <> Reply to Message

Can you please write us the gdb log?

From your shell, type:

gdb root.exe

then

r

then

.x run0LmdDPMBkgSte.C

after your crash, type:

bt

and paste what stays after.
maybe this could help.

P.S. type "q" to quit

Subject: Re: segmentation violation in FairPrimaryGenerator
Posted by Anastasia Karavdina on Mon, 01 Oct 2012 18:16:04 GMT
View Forum Message <> Reply to Message

Ok. in meanwhile I recompile pandaroot and tried your suggestion about magnetic field usages and so on together with stand alone DPM version. It works, without any problem!
But if I use PndDpmDirect instead (basicly macro runLumi0DPMbkgSte) I have following log from GDB:
Program received signal SIGINT, Interrupt.
0x00007ffff04a1552 in dpm_event (nhad=0) at
/home/karavdin/pandaRoot12/pgenerators/DpmEvtGen/dpm_gen.f:334
334 SumPz=SumPz/float(Nhad)
Missing separate debuginfos, use: debuginfo-install expat-2.0.1-9.1.el6.x86_64
fontconfig-2.8.0-3.el6.x86_64 freetype-2.3.11-6.el6_0.2.x86_64 keyutils-libs-1.4-4.el6.x86_64
krb5-libs-1.8.2-3.el6_0.7.x86_64 libX11-1.3-2.el6.x86_64 libXau-1.0.5-1.el6.x86_64
libXcursor-1.1.10-2.el6.x86_64 libXext-1.1-3.el6.x86_64 libXfixes-4.0.4-1.el6.x86_64

libXft-2.1.13-4.1.el6.x86_64 libXpm-3.5.8-2.el6.x86_64 libXrender-0.9.5-1.el6.x86_64 libcom_err-1.41.12-7.el6.x86_64 libselinux-2.0.94-2.el6.x86_64 libxcb-1.5-1.el6.x86_64 openssl-1.0.0-4.el6_0.2.x86_64 pcre-7.8-3.1.el6.x86_64 zlib-1.2.3-25.el6.x86_64
(gdb) bt
#0 0x00007ffff04a1552 in dpm_event (nhad=0) at /home/karavdin/pandaRoot12/pgenerators/DpmEvtGen/dpm_gen.f:334
#1 0x00007ffff049ff51 in dpm_gen (pluto=0, seed=50891) at /home/karavdin/pandaRoot12/pgenerators/DpmEvtGen/dpm_gen.f:13
#2 0x00007fffec7deb57 in PndDpmDirect::ReadEvent (this=0x1731f50
../../gdb/stack.c:289: internal-error: print_frame_args: Assertion `nsym != NULL' failed.

Well, here I should admit that I use DPM code with corrections done by Stefan Pflueger which is not in trunk yet. But I'm still surprise to see how different DPM works as a stand alone compiled version.
I guess it's difficult to dig up the problem until DPM code will be in SVN.
Once again, thank you for your help, Stefano.

Subject: Re: segmentation violation in FairPrimaryGenerator
Posted by StefanoSpataro on Tue, 02 Oct 2012 07:41:34 GMT
View Forum Message <> Reply to Message

Hi,
meanwhile you could try using the standard DPM, just to understand if the problem comes from the new DPM itself or from somewhere else.

Subject: Re: segmentation violation in FairPrimaryGenerator
Posted by Anastasia Karavdina on Thu, 04 Oct 2012 10:13:53 GMT
View Forum Message <> Reply to Message

With standard DPM runLumi0DPMbkgSte.C runs without problem (crashes, stops etc)

Subject: Re: segmentation violation in FairPrimaryGenerator
Posted by StefanoSpataro on Thu, 04 Oct 2012 10:20:57 GMT
View Forum Message <> Reply to Message

Then probably there is something in the DPM, at least it is not compatible with the Direct version. Maybe it is producing some fake tracks which create also problems in the geant?

Subject: Re: segmentation violation in FairPrimaryGenerator
Posted by Anastasia Karavdina on Thu, 04 Oct 2012 10:36:28 GMT
View Forum Message <> Reply to Message

Well, according gdb output which I posted before DPMDirect stops during numerical calculation of integral for cross-section(s) calculation. I've already asked Stefan have a look at this. (He indeed made some changes in this numerical calculation)

We'll write here as soon as this problem would be fixed.

But you're maybe right about problem which I had during my 1st message in this tread.
Surprisingly it was resolved with several pandaroot recompilation

---

Subject: Re: segmentation violation in FairPrimaryGenerator
Posted by StefanoSpataro on Thu, 04 Oct 2012 10:57:42 GMT
View Forum Message <> Reply to Message

Let us know.
