

---

Subject: Stable version didn't work with Geant4.8.3.p01  
Posted by [Anonymous Poster](#) on Fri, 24 Aug 2007 14:09:34 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hello,

we, together with Tobias, have managed to replace Geant4.8.2 (which contains the bug) to Geant4.8.3.p01 in Jülich. Just some remarks how to do that:

Geant4.8.3.p01 was downloaded from official Geant4 site.

one has to compile Geant4.8.3.p01 by hand. We also tried use configure.sh script, but in this case G3toG4 package was not compiled for whatever reason.

after the Geant4.8.3.p01 compilation, geant4\_vmc package has to be recompiled. configure.sh can be used for that, one just has to change the version of Geant4 from 8.2 to 8.3.p01 inside the script.

after that recompile your build of the PandaRoot.

!! I noticed, that if I using the so-called stable version of PandaRoot (v01-00-00), the Geant4 craches with segmentation fault , while for the HEAD version of the PandaRoot it works without problem. !!

Best regards,

Andrey Sokolov

---

Subject: Re: Stable version didn't work with Geant4.8.3.p01  
Posted by [Stefano Spataro](#) on Fri, 24 Aug 2007 14:21:28 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hello,  
could you please send the error you get with the 1\_0\_0 release?  
Thanks

In each case, soon or later we will move to geant4 9.0 + root 5\_16, just to inform you  
Bye

---

Subject: Re: Stable version didn't work with Geant4.8.3.p01  
Posted by [Jens Sören Lange](#) on Sun, 26 Aug 2007 12:55:28 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hi Andrei,

but v01\_00\_00 has the identical g4  
(i.e. external packages from 10. Jan 2007).

and g4+emc is tested O.K.

so it must be a detector.

what is crashing? the run\_sim1.C? If yes, can you say in which detector the segmentation fault occurs?

(or even better, maybe run the Dart.sh in the crashing version, so that we can see the crash in the dashboard?)

v01\_00\_00 is rev 1079.

some detector changes (stt1, stt2 vs. stt; tst vs. mvd)  
were made in later revisions.

Soeren

---

---

Subject: Re: Stable version didn't work with Geant4.8.3.p01  
Posted by [Anonymous Poster](#) on Mon, 27 Aug 2007 09:54:29 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hi Soeren and Stefano,

first off sorry for the delay with the answer. Especially for you I have reproduced this crash:

```
root [0] .x mvd/runMvdSim.C
```

```
PSaid instance created... access via gSaid->f()
```

```
- RTDB container factory CbmBaseContFact
- RTDB container factory CbmFieldContFact
- RTDB container factory CbmPassiveContFact
- RTDB container factory MvdContFact
-I- CbmRun::SetMaterials() Media file used:
/home/sokolov/fairroot/cbmsoft/pandaroot_stable/geometry/media_pnd.geo
- I - MvdDetector: fListOfSensitives contains:
    StripSensor
    SensorActiveArea
SetStoreTraj0
```

```
===== CbmRunSim: Initialising simulation run =====
Info in <TGeoManager::TGeoManager>: Geometry CBMGeom, CBM geometry created
-I- CbmGeoMedia Read media
Loading Geant4 granular libraries ...
dlopen error: /home/sokolov/fairroot/cbmsoft/transport/geant4/lib/Linux-g++/libG4hadro
nic_xsect.so: undefined symbol: _ZN16G4StableIsotopes12nucleonCountE
Load Error: Failed to load Dynamic link library
```

```

/home/sokolov/fairroot/cbmsoft/transport/geant4/lib/Linux-g++/libG4hadronic_xsect.so
(int)(-1)
*** Interpreter error recovered ***
dlopen error: /home/sokolov/fairroot/cbmsoft/cbuild_stable/lib/libCbmG4.so: undefined symbol:
G4cout
Load Error: Failed to load Dynamic link library
/home/sokolov/fairroot/cbmsoft/cbuild_stable/lib/libCbmG4.so
CbmMCApplication::InitMC
*** Interpreter error recovered ***
Error: Function Config() is not defined in current scope (tmpfile):1:
*** Interpreter error recovered ***

*** Break *** segmentation violation
Using host libthread_db library "/lib/tls/libthread_db.so.1".
Attaching to program: /proc/818/exe, process 818
`system-supplied DSO at 0xffffe000' has disappeared; keeping its symbols.
[Thread debugging using libthread_db enabled]
[New Thread 1099541568 (LWP 818)]
0xffffe410 in __kernel_vsyscall ()
#1 0x417d30a3 in __waitpid_nocancel () from /lib/tls/libc.so.6
#2 0x4177bcb9 in do_system () from /lib/tls/libc.so.6
#3 0x417407ad in system () from /lib/tls/libpthread.so.0
#4 0x402a533f in TUnixSystem::Exec () from
/home/sokolov/fairroot/cbmsoft/tools/root/lib/libCore.so.5.14
#5 0x402ace80 in TUnixSystem::StackTrace () from
/home/sokolov/fairroot/cbmsoft/tools/root/lib/libCore.so.5.14
#6 0x402ac293 in TUnixSystem::DispatchSignals () from
/home/sokolov/fairroot/cbmsoft/tools/root/lib/libCore.so.5.14
#7 0x402ac419 in SigHandler () from
/home/sokolov/fairroot/cbmsoft/tools/root/lib/libCore.so.5.14
#8 0x402a61c4 in sighandler () from
/home/sokolov/fairroot/cbmsoft/tools/root/lib/libCore.so.5.14
#9 <signal handler called>
#10 0x44654f5a in CbmMCApplication::InitMC (this=0x8ca6610,
      setup=0x8d06968 "/home/sokolov/fairroot/cbmsoft/pandaroot_stable/gconfig/g4Config.C")
      at /home/sokolov/fairroot/cbmsoft/pandaroot_stable/base/CbmMCApplication.cxx:210
#11 0x44667f33 in CbmRunSim::Init (this=0x8ba8b88)
      at /home/sokolov/fairroot/cbmsoft/pandaroot_stable/base/CbmRunSim.cxx:108
#12 0x4468cd33 in G__CbmDict_777_0_6 (result7=0xbfc18afc, funcname=0x8ba30e8 "\001",
      libp=0xbfc15058, hash=0)
      at /home/sokolov/fairroot/cbmsoft/cbuild_stable/base/CbmDict.cxx:9484
#13 0x4084c597 in Cint::G__ExceptionWrapper () from
/home/sokolov/fairroot/cbmsoft/tools/root/lib/libCint.so.5.14
#14 0x408f0d08 in G__call_cppfunc () from
/home/sokolov/fairroot/cbmsoft/tools/root/lib/libCint.so.5.14
#15 0x408db995 in G__interpret_func () from
/home/sokolov/fairroot/cbmsoft/tools/root/lib/libCint.so.5.14
#16 0x408d13b3 in G__getfunction () from
/home/sokolov/fairroot/cbmsoft/tools/root/lib/libCint.so.5.14
#17 0x4095c5fd in G__getstructmem () from
/home/sokolov/fairroot/cbmsoft/tools/root/lib/libCint.so.5.14
#18 0x40954628 in G__getvariable () from

```

/home/sokolov/fairroot/cbmsoft/tools/root/lib/libCint.so.5.14  
#19 0x408b2a66 in G\_\_getitem () from  
/home/sokolov/fairroot/cbmsoft/tools/root/lib/libCint.so.5.14  
#20 0x408bd5a3 in G\_\_getexpr () from  
/home/sokolov/fairroot/cbmsoft/tools/root/lib/libCint.so.5.14  
#21 0x4090acd7 in G\_\_exec\_function () from  
/home/sokolov/fairroot/cbmsoft/tools/root/lib/libCint.so.5.14  
#22 0x4091049d in G\_\_exec\_statement () from  
/home/sokolov/fairroot/cbmsoft/tools/root/lib/libCint.so.5.14  
#23 0x408a0546 in G\_\_exec\_tempfile\_core () from  
/home/sokolov/fairroot/cbmsoft/tools/root/lib/libCint.so.5.14  
#24 0x408a075d in G\_\_exec\_tempfile () from  
/home/sokolov/fairroot/cbmsoft/tools/root/lib/libCint.so.5.14  
#25 0x4091b939 in G\_\_process\_cmd () from  
/home/sokolov/fairroot/cbmsoft/tools/root/lib/libCint.so.5.14  
#26 0x402139cb in TCint::ProcessLine () from  
/home/sokolov/fairroot/cbmsoft/tools/root/lib/libCore.so.5.14  
#27 0x40210d97 in TCint::ProcessLineSynch () from  
/home/sokolov/fairroot/cbmsoft/tools/root/lib/libCore.so.5.14  
#28 0x4014ba0b in TApplication::ProcessFile () from  
/home/sokolov/fairroot/cbmsoft/tools/root/lib/libCore.so.5.14  
#29 0x40148e87 in TApplication::ProcessLine () from  
/home/sokolov/fairroot/cbmsoft/tools/root/lib/libCore.so.5.14  
#30 0x415dd0da in TRint::HandleTermInput () from  
/home/sokolov/fairroot/cbmsoft/tools/root/lib/libRint.so.5.14  
#31 0x415dc806 in TTermInputHandler::Notify () from  
/home/sokolov/fairroot/cbmsoft/tools/root/lib/libRint.so.5.14  
#32 0x415de648 in TTermInputHandler::ReadNotify () from  
/home/sokolov/fairroot/cbmsoft/tools/root/lib/libRint.so.5.14  
#33 0x402ac53a in TUnixSystem::CheckDescriptors () from  
/home/sokolov/fairroot/cbmsoft/tools/root/lib/libCore.so.5.14  
#34 0x402ac7a2 in TUnixSystem::DispatchOneEvent () from  
/home/sokolov/fairroot/cbmsoft/tools/root/lib/libCore.so.5.14  
#35 0x401c9786 in TSystem::InnerLoop () from  
/home/sokolov/fairroot/cbmsoft/tools/root/lib/libCore.so.5.14  
#36 0x401c972c in TSystem::Run () from  
/home/sokolov/fairroot/cbmsoft/tools/root/lib/libCore.so.5.14  
#37 0x401482c3 in TApplication::Run () from  
/home/sokolov/fairroot/cbmsoft/tools/root/lib/libCore.so.5.14  
#38 0x415dd560 in TRint::Run () from  
/home/sokolov/fairroot/cbmsoft/tools/root/lib/libRint.so.5.14  
#39 0x08048dc0 in main ()

Some remarks: if I compile the stable release with Geant4.8.2 or the HEAD version with Geant4.8.3 it works without any problems. Even warnings are gone.

Best regards,  
Andrey

Subject: Re: Stable version didn't work with Geant4.8.3.p01  
Posted by [StefanoSpataro](#) on Mon, 27 Aug 2007 11:08:48 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hi,  
trying to find some clue...

I can see that you are using the CbmG4.so library, why? You should comment out this include, if you have it. It is only when one want to use particular physics lists. This library is not used at the moment, and gives problems.

And second, did you change the g4Config.C file? It seems you activated the hadronic lists. It could be that there are some differences in G4 so that the virtualMC is not able to handle particular cases.

In each case it seems that all the errors you ahve are not dependent from the pandaroot release (apart the g4Config.C), so I am wondering why it does not prodice the same crashes in the main svn version...

---

Subject: Re: Stable version didn't work with Geant4.8.3.p01  
Posted by [Anonymous Poster](#) on Thu, 30 Aug 2007 14:19:46 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Stefano Spataro wrote on Mon, 27 August 2007 13:08Hi,  
trying to find some clue...

I can see that you are using the CbmG4.so library, why? You should comment out this include, if you have it. It is only when one want to use particular physics lists. This library is not used at the moment, and gives problems.

I use it because nobody told me that I shouldn't do this. Ok, I will comment it out.

Quote:

And second, did you change the g4Config.C file? It seems you activated the hadronic lists. It could be that there are some differences in G4 so that the virtualMC is not able to handle particular cases.

No, I didn't touch g4Config.C.

Quote:

In each case it seems that all the errors you ahve are not dependent from the pandaroot release (apart the g4Config.C), so I am wondering why it does not prodice the same crashes in the main svn version...

Yes it looks like that, but how you can explain the fact that exactly the same external packages are working with the head version without error?

Best regards,

Andrey

---