
Subject: problem in retriving data from runDemo output

Posted by [Anonymous Poster](#) on Thu, 20 Mar 2008 09:35:56 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hi All,

I'm facing problem while retriving the data from recotasks/demo/runDemo.C output. I tried with the latest version of pandaroot from svn. I could able to run the recotasks/demo/runMC.C and runDemo.C. But it is crashing while retriving the output from "demo.mcreco.root" file. Please let me know if I'm making any mistakes somewhere. From the errors it looks like there is some problem in trackrep/GeaneTrackRep.cxx file. It is giving the errors like:

```
root [0] .x readCov.C
```

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

```
- RTDB container factory CbmBaseContFact
- RTDB container factory PndFieldContFact
- RTDB container factory PndPassiveContFact
- RTDB container factory PndMvdContFact
- RTDB container factory PndEmcContFact
- RTDB container factory PndDrcContFact
- RTDB container factory PndTpcContFact
```

```
*** # of events are*** 100
```

```
GeaneTrackRep::Standard Ctor
```

```
*** Break *** segmentation violation
```

```
Using host libthread_db library "/lib/libthread_db.so.1".
```

```
Attaching to program: /proc/27236/exe, process 27236
```

```
Failed to read a valid object file image from memory.
```

```
done.
```

```
done.
```

```
[Thread debugging using libthread_db enabled]
```

```
[New Thread 16384 (LWP 27236)]
```

```
done.
```

```
done.
```

```
done.
```

```
done.
```

```
done.
```

```
done.
```

```
done.
```

```
done.
```

```
done.
```

```
-----
```

The macro which I use to retrived the data lokks like:

```
{
gROOT->Macro("loadRecoLibs.C");
TFile* f=TFile::Open("demo.mcreco.root");
TTree* t=(TTree*)f->Get("cbmsim");
TClonesArray *fTr = new TClonesArray("Track");
t->SetBranchAddress("Track",&fTr);
Track *tr1;
cout<<"*** # of events are*** "<<t->GetEntriesFast()<<endl;
for(Int_t i=0;i<t->GetEntriesFast();i++){
```

```

t->GetEntry(i);
cout<<"number of Tracks *** "<<fTr->GetEntriesFast()<<endl;
for(Int_t j=0;j<fTr->GetEntriesFast();j++){
    tr1= (Track *)fTr->At(j);
    //Getting info from AbsTrackRep
    Int_t Charge =tr1->getCardinalRep()->getCharge();
    Float_t x = tr1->getCardinalRep()->getPos().X();
    Float_t y = tr1->getCardinalRep()->getPos().Y();
    Float_t z = tr1->getCardinalRep()->getPos().Z();
}
}
}
}

```

Subject: Re: problem in retriving data from runDemo output
 Posted by [Sebastian Neubert](#) on Thu, 20 Mar 2008 10:17:04 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Dipak!

One problem that I can spot immediately is the following:

GeaneTrackRep obviously requires GEANE. So you cannot use it without setting up geant3. And in addition you always need the complete Framework environment (the runmanager in particular) This takes care of the magnetic field and so on.

This dependency is a bit awkward but it is needed.

For the moment I would suggest, that you put all the things you want to do into another task which you plug to your reco script. Like I have done with TrackStatTask.

Cheers! Sebastian.

PS: I see a bigger context here: persistence of track objects. Let's keep that in mind!

Subject: Re: problem in retriving data from runDemo output
 Posted by [Anonymous Poster](#) on Thu, 20 Mar 2008 13:08:35 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Sabastian,

I made a separate task which should read the demo output. Also I have loaded all the library functions of the pandaroot. When I plugged this task to my macro it crashes exactly at the same point where it was crashing without the task. It gives the errors like:

```
root [0] .x readTrackVertexFit.C
```

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

- RTDB container factory CbmBaseContFact

- RTDB container factory PndFieldContFact
- RTDB container factory PndPassiveContFact
- RTDB container factory PndMvdContFact
- RTDB container factory PndEmcContFact
- RTDB container factory PndDrcContFact
- RTDB container factory PndTpcContFact

-l- CbmRunAna: Opening Input file: ../recotasks/demo/demo.mcreco.root
 GeaneTrackRep::Standard Ctor

*** Break *** segmentation violation
 Using host libthread_db library "/lib/libthread_db.so.1".
 Attaching to program: /proc/30542/exe, process 30542
 Failed to read a valid object file image from memory.
 done.
 done.
 [Thread debugging using libthread_db enabled]
 [New Thread 16384 (LWP 30542)]
 done.
 done.
 done.
 done.
 done.
 done.
 done.
 done.
 done.
 0x40c90788 in waitpid () from /lib/libc.so.6
 #1 0x40d188c0 in __DTOR_END__ () from /lib/libc.so.6
 #2 0x40c29442 in do_system () from /lib/libc.so.6
 #3 0x40ba3c5f in system () from /lib/libpthread.so.0
 #4 0x401fa79f in TUnixSystem::Exec ()
 from /misc/cbmsoft/Debian3.1/mar08/fairsoft/tools/root/lib/libCore.so.5.18
 #5 0x401fac63 in TUnixSystem::StackTrace ()
 from /misc/cbmsoft/Debian3.1/mar08/fairsoft/tools/root/lib/libCore.so.5.18
 #6 0x401f85cc in TUnixSystem::DispatchSignals ()
 from /misc/cbmsoft/Debian3.1/mar08/fairsoft/tools/root/lib/libCore.so.5.18
 #7 0x401f63a8 in SigHandler ()
 from /misc/cbmsoft/Debian3.1/mar08/fairsoft/tools/root/lib/libCore.so.5.18
 #8 0x401fdefe in sighandler ()
 from /misc/cbmsoft/Debian3.1/mar08/fairsoft/tools/root/lib/libCore.so.5.18
 #9 0x40ba2825 in __pthread_sighandler () from /lib/libpthread.so.0
 #10 <signal handler called>
 #11 0x44ec0ad5 in CbmGeanePro (this=0x891a688)
 at /d/panda02/dipak/pndroot0308_1/pandaroot/geane/CbmGeanePro.cxx:34
 #12 0x44ecc945 in new_CbmGeanePro (p=0x0)
 at /d/panda02/dipak/pndroot0308_1/build/geane/GeaneDict.cxx:286
 #13 0x401db249 in TClass::New ()
 from /misc/cbmsoft/Debian3.1/mar08/fairsoft/tools/root/lib/libCore.so.5.18
 #14 0x40dd4165 in TBufferFile::ReadObjectAny ()
 from /misc/cbmsoft/Debian3.1/mar08/fairsoft/tools/root/lib/libRIO.so
 #15 0x40dd26be in TBufferFile::ReadFastArray ()
 from /misc/cbmsoft/Debian3.1/mar08/fairsoft/tools/root/lib/libRIO.so

#16 0x40e15fe1 in TStreamerInfo::ReadBuffer<char**> ()
from /misc/cbmsoft/Debian3.1/mar08/fairsoft/tools/root/lib/libRIO.so
#17 0x40dd66be in TBufferFile::ReadClassBuffer ()
from /misc/cbmsoft/Debian3.1/mar08/fairsoft/tools/root/lib/libRIO.so
Root >

But before collaboration meeting I was able to access all the required variables like x,y,z,px,py,pz and the respective cov. matrices and I'm using the same macro but it is crashing in GeaneTrackRep!!! If you have made any example macro to readback the demo output please let me know.

Best Regards,
Dipak

Subject: Re: problem in retriving data from runDemo output
Posted by [Sebastian Neubert](#) on Thu, 20 Mar 2008 13:14:11 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi!

Does the macro work if you select the LSLTrackRep?
You have to comment out the command UseGeane() in the macro.

Sebastian.

Subject: Re: problem in retriving data from runDemo output
Posted by [Anonymous Poster](#) on Thu, 20 Mar 2008 13:22:56 GMT
[View Forum Message](#) <> [Reply to Message](#)

Thanks! Finally after commenting out the "UseGeane()" it worked. I hope commenting out this doesn't affect the final values of global parameters (x,y,z,px,py,pz).

best regards,
Dipak

Subject: Re: problem in retriving data from runDemo output
Posted by [Sebastian Neubert](#) on Thu, 20 Mar 2008 13:26:50 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi!

Ok, but please let us not forget this issue - it means, that there is still something wrong in GeaneTrackRep. I will ,make a Wiki page where we can collect bug reports on Genfit.

Sebastian.

Subject: Re: problem in retriving data from runDemo output
Posted by [Mohammad Al-Turany](#) on Tue, 25 Mar 2008 22:59:48 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hallo Sebastian,

Quote:GeaneTrackRep obviously requires GEANE. So you cannot use it without setting up geant3. And in addition you always need the complete Framework environment (the runmanager in particular) This takes care of the magnetic field and so on.

This dependency is a bit awkward but it is needed.

For the moment I would suggest, that you put all the things you want to do into another task which you plug to your reco script. Like I have done with TrackStatTask.

Cheers! Sebastian.

PS: I see a bigger context here: persistence of track objects. Let's keep that in mind!

That we need to link against geane, geant3 or even geant4 is not a problem at all, this we do in simulation for example but we can still read the root files without any of these packages, the point is what is going to the streamer? if you have pointers as member variables of your data object (Track, TrackRep or what ever) you should tell rootcint not to stream these pointers by putting "///" in the header file directly after declaration of these pointers, other wise root will stream them to the file and you will need the libraries to read the root file!

regards

Mohammad

Subject: Re: problem in retriving data from runDemo output
Posted by [Sebastian Neubert](#) on Wed, 26 Mar 2008 09:24:40 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi Mohammad!

Quote:if you have pointers as member variables of your data object (Track, TrackRep or what ever) you should tell rootcint not to stream these pointers by putting "///" in the header file directly after declaration of these pointers, other wise root will stream them to the file and you will need the libraries to read the root file!

That would of course be a solution, but then the Track object will be of not much use. I can not think about many applications where you need track parameters without having the ability to propagate them.

Kind Regards, Sebastian.

Subject: Re: problem in retrieving data from runDemo output
Posted by [Mohammad Al-Turany](#) on Wed, 26 Mar 2008 09:35:38 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hallo Sebastian,

Quote: I can not think about many applications where you need track parameters without having the ability to propagate them.

This is true, but usually you have one or more propagator which you use to propagate your track parameters! and these should not be inside the track representation! otherwise with this logic the track should have also the field values every where and the geometry!
As I said normally you have a propagator(s) which knows about the geometry and field and can propagate track parameters to any where inside the geometry, given back the new track parameters and covariant matrix at the new point, plan,...etc. At least that how it works in CBM, Alice and Hades. The other way (That the track know how to propagate him self) will only create problems!

regards

Mohammad

Subject: Re: problem in retrieving data from runDemo output
Posted by [Sebastian Neubert](#) on Wed, 26 Mar 2008 10:03:35 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hm,

I see what you mean. It is clear that there could be several propagators.

My thinking was, that each set of parameters belongs to a specific propagator, so this is why I have not separated them in the first place.

Do I understand you correctly, that you wanted to achieve this separation with the TrackBase package?

Cheers! Sebastian.

Subject: Re: problem in retrieving data from runDemo output
Posted by [Mohammad Al-Turany](#) on Wed, 26 Mar 2008 10:09:25 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi,

Quote:Do I understand you correctly, that you wanted to achieve this separation with the TrackBase package?

Yes.

Subject: Re: problem in retriving data from runDemo output
Posted by [Sebastian Neubert](#) on Wed, 26 Mar 2008 10:25:12 GMT
[View Forum Message](#) <> [Reply to Message](#)

Ok,

I tried to do the same in principle with the "TrackFitStat" object that I mentioned.

One "dumps" the fitted parameters into a simple class and then afterwards you can use this to initialize a new propagator.

However this requires:

- a) There is a transformation of coordinates (can of course be done)
- b) The user has to take care that the new propagator respects the conditions under which the old parameters were fitted (magnetic field, material ...).

In the case of TrackBase however there is a deep dependency on the base package, because the Run object singleton is used to access the magnetic field. You cannot simply use it outside the framwork. Can you?

Sebastian.

Subject: Re: problem in retriving data from runDemo output
Posted by [Mohammad Al-Turany](#) on Wed, 26 Mar 2008 11:34:37 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi,

Quote:b) The user has to take care that the new propagator respects the conditions under which the old parameters were fitted (magnetic field, material ...).

Normally a propagator has access to the geometry, material and field, then it needs only the initial track parameters to deliver new ones and the errors. So is there any other conditions? On the other hand the Fitter usually uses the propagator to estimate the new parameters but never act on the propagator it self! my understanding is that these two packages can be completely independent of each other, the only thing is to grantee that they use the same geometry and field and that is exactly what the CbmRun is doing!

Quote:In the case of TrackBase however there is a deep dependency on the base package,

because the Run object singleton is used to access the magnetic field. You cannot simply use it outside the framework. Can you?

If I have a manager which deliver me the field and geometry I will use it!
This dependancy is ok for me, I will never try to remove it! for following reasons:

1. I do not see any reason for this! except that you think it is better to be independent of any framework(Panda could change it again) but this means you should use the geane fortran directly and even though you are still dependent on G3 and CERN, doing this you have to write your own VMC main application loop to run G3 and Geane. Or you implement also your own propagator!

2. The CbmRun does not only deliver the field but take care that you have the proper one used in simulation and the same for the geometry!

regards

Mohammad

Subject: Re: problem in retriving data from runDemo output
Posted by [Sebastian Neubert](#) on Wed, 26 Mar 2008 12:42:35 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi!

CbmRun is making sure that geane is getting the right field and geometry. Ok.

However Dipak tried to access the track data in a simple root macro WITHOUT the framework. This is how we started the discussion.

Obviously Track parameters without it's context do not make sense. So you need the framework. If everybody is happy with this... Dipak, Soeren, what do you say?

But I think with a more explicit interface between the track parameters / propagators and it's environment this restriction could be partly lifted. After all the track only needs a BField and (maybe) a geometry - not the whole Run Information.
And this would maybe be usefull for tests like Dipak's.

Regards, Sebastian.

Subject: Re: problem in retriving data from runDemo output
Posted by [Mohammad Al-Turany](#) on Wed, 26 Mar 2008 13:01:47 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi,

It seems that you miss understand me , or I did not make my point clearly! does not matter.

Quote:However Dipak tried to access the track data in a simple root macro WITHOUT the framework. This is how we started the discussion

This is clear for me, and this what I want to have also, reading data files should be and is possible without the framework except for your tracks!

Quote:But I think with a more explicit interface between the track parameters / propagators and it's environment this restriction could be partly lifted. After all the track only needs a BField and (maybe) a geometry - not the whole Run Information.
And this would maybe be usefull for tests like Dipak's.

Track parameters for me are basic data types who know nothing about framework geometry or what ever, they only know in which frame they are and that they are calculated at a certain point in space or in a certain plan (reference frame). From this information the Run can tell what is the field at this point and in which material it is, if this is needed. In case that somebody just want to see the distributions of a certain parameter or the errors then plain root should be enough.

Regards,

Mohammad

Subject: Re: problem in retriving data from runDemo output
Posted by [Sebastian Neubert](#) on Wed, 26 Mar 2008 13:14:20 GMT
[View Forum Message](#) <> [Reply to Message](#)

But in the Ctor of CbmTrackParP the CbmRunAna singleton is called. So what happens if I have not set this up correctly?

Indeed with the GeaneTrackRep it is not possible to do it in that way. You should make a task to read back in that information.

Sebastian.

Subject: Re: problem in retriving data from runDemo output
Posted by [Mohammad Al-Turany](#) on Wed, 26 Mar 2008 13:33:02 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi,

Quote:But in the Ctor of CbmTrackParP the CbmRunAna singleton is called. So what happens if I have not set this up correctly?

Only member variables go to the file unless you add "//!" in the header after declaring them. Now if I save this CbmTrackParP to a file and read it again I do not need the CbmRunAna at all to plot the parameters or even to use them! So as long as we speak about variables from files there is no problem of having what ever in the implementation file! the streamer always call the default constructor of the class which in case of this class contain nothing!

Mohammad

Subject: Re: problem in retriving data from runDemo output
Posted by [Jens Sören Lange](#) on Sun, 30 Mar 2008 16:35:52 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi all,

Quote:

However Dipak tried to access the track data in a simple root macro WITHOUT the framework. This is how we started the discussion.

Obviously Track parameters without it's context do not make sense. So you need the framework.

This is exactly the core question and I think I know a compromise now (maybe). Can we talk about this in detail on the next meeting on WED Apr 2 (will everybody be there?). It is quite important.

cheers, Soeren
