

---

Subject: [FIXED] FairEvtFilter problem with c++11  
Posted by [StefanoSpataro](#) on Wed, 12 Nov 2014 17:16:38 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Trying to install pandaroot with the new external packages and git, in an Ubuntu 14.10 (gcc 4.9.1) I have seen that the code has problems with the C++11 options.  
In particular:

```
In file included from
/home/lavezzi/git/pandaroot/pgenerators/eventFilter/FairFilteredPrimaryGenerator.h:27:0,
      from
/home/lavezzi/git/pandaroot/pgenerators/eventFilter/FairFilteredPrimaryGenerator.cxx:1:
/home/lavezzi/git/pandaroot/pgenerators/eventFilter/FairEvtFilter.h:129:46: error: 'constexpr'
needed for in-class initialization of static data member 'const Double_t
FairEvtFilter::kNoChargeSpecified' of non-integral type [-fpermissive]
  static const Double_t kNoChargeSpecified = -999.9;
                        ^
```

```
pgenerators/CMakeFiles/PGen.dir/build.make:497: recipe for target
'pgenerators/CMakeFiles/PGen.dir/eventFilter/FairFilteredPrimaryGenerator.cxx.o' failed
```

In c++11 static const of a non-integer number is not allowed, one should use constexpr. But if I use constexpr:

```
[ 9%] Generating G__PGenDict.cxx
Error: Invalid type 'constexpr' in declaration of 'Double_t kNoChargeSpecified'
/home/lavezzi/git/pandaroot/pgenerators/eventFilter/FairEvtFilter.h:129:
Warning: Error occurred during reading source files
Warning: Error occurred during dictionary source generation
!!!Removing /home/lavezzi/git/pandaroot/cbuild/pgenerators/G__PGenDict.cxx
/home/lavezzi/git/pandaroot/cbuild/pgenerators/G__PGenDict.h !!!
Error: /home/lavezzi/git/fairsoft_jul14p3/externals/bin/rootcint: error loading headers...
pgenerators/CMakeFiles/PGen.dir/build.make:77: recipe for target
'pgenerators/G__PGenDict.cxx' failed
make[2]: *** [pgenerators/G__PGenDict.cxx] Error 1
```

Martin, can you fix this problem? This is present in several classes, including also FTS. From some comments I read in the forum, what you are doing is a bit nasty and should be redesigned in a different and safer way.

Thanks in advance.

---

Subject: Re: FairEvtFilter problem with c++11  
Posted by [MartinJGaluska](#) on Wed, 12 Nov 2014 18:06:55 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hi Stefano,

how do I install the new external packages in order to replicate the problem?

Do I just need to follow the instructions here?

<https://github.com/FairRootGroup/FairSoft>

Quote:From some comments i read in the forum, what you are doing is a bit nasty and should be disegned in a different and safer way.

From what you wrote it seems that you have already encountered a solution to the problem. Could you point me to this safer way which you are talking about?

Kind regards,  
Martin

---

Subject: Re: FairEvtFilter problem with c++11  
Posted by [StefanoSpataro](#) on Wed, 12 Nov 2014 18:20:49 GMT  
[View Forum Message](#) <> [Reply to Message](#)

This is what I found:

<http://stackoverflow.com/questions/9141950/initializing-const-member-wit-hin-class-declaration-in-c>

In Ubuntu 14.10 there is gcc 4.9.1, most probably it is not just a matter to install new externals. I will try to setup a VM so that I can send it to you for tests.

---

Subject: Re: FairEvtFilter problem with c++11  
Posted by [MartinJGaluska](#) on Wed, 12 Nov 2014 18:33:41 GMT  
[View Forum Message](#) <> [Reply to Message](#)

Thank you. Tomorrow I will be gone, but I will start to implement the changes on Friday.

---

Subject: Re: FairEvtFilter problem with c++11  
Posted by [StefanoSpataro](#) on Fri, 14 Nov 2014 10:19:48 GMT  
[View Forum Message](#) <> [Reply to Message](#)

I fixed in the code and committed in svn.  
Simply you cannot assign the value of a static const in the header, but you need to move it to the .cxx (thanks Mohammad and Florain for the help).  
A good question is why you are defining pi, instead of using TMath::Pi().  
You can find below the changes I did.

```
pandauser@panda-VirtualBox:~/fairsoft_jul14p3/pandaroot$ svn diff fts/FtsTracking/  
Index: fts/FtsTracking/PndFtsHoughTrackFinder.cxx
```

```
=====  
--- fts/FtsTracking/PndFtsHoughTrackFinder.cxx (revision 26438)  
+++ fts/FtsTracking/PndFtsHoughTrackFinder.cxx (working copy)
```

@ @ -4,8 +4,10 @ @

```
ClassImp(PndFtsHoughTrackFinder);
```

```
+const Double_t PndFtsHoughTrackFinder::meinpi = 3.14159265359;  
+const Double_t PndFtsHoughTrackFinder::fZLineParabola = 368.;  
+const Double_t PndFtsHoughTrackFinder::fZParabolaLine = 605.;
```

```
-  
PndFtsHoughTrackFinder::PndFtsHoughTrackFinder(PndFtsHoughTrackerTask *trackerTask)  
:  
    fTrackerTask(trackerTask),
```

Index: fts/FtsTracking/PndFtsHoughTrackFinder.h

```
=====  
--- fts/FtsTracking/PndFtsHoughTrackFinder.h (revision 26438)  
+++ fts/FtsTracking/PndFtsHoughTrackFinder.h (working copy)  
@ @ -164,10 +164,10 @ @  
    // std::vector<PndTrackCand> fTrackCand; // resulting track candidates, also used for  
returning PndTracks
```

```
- static const Double_t meinpi = 3.14159265359;  
+ static const Double_t meinpi;  
    ///< sets where the apex of the parabola is supposed to be  
- static const Double_t fZLineParabola = 368.; // the value should coincide with the start of the  
dipole field // 368. was ok  
- static const Double_t fZParabolaLine = 605.; // the value should coincide with the end of the  
dipole field // TODO determine this value  
+ static const Double_t fZLineParabola; // the value should coincide with the start of the dipole  
field // 368. was ok  
+ static const Double_t fZParabolaLine; // the value should coincide with the end of the dipole  
field // TODO determine this value
```

pandauser@panda-VirtualBox:~/fairsoft\_jul14p3/pandaroot\$ svn diff pgenerators/eventFilter/  
Index: pgenerators/eventFilter/FairEvtFilter.cxx

```
=====  
--- pgenerators/eventFilter/FairEvtFilter.cxx (revision 26438)  
+++ pgenerators/eventFilter/FairEvtFilter.cxx (working copy)  
@ @ -5,8 +5,8 @ @
```

```
#include "FairEvtFilter.h"
```

```
+const Double_t FairEvtFilter::kNoChargeSpecified = -999.9;
```

```
-
```

```
// ----- Default constructor -----  
FairEvtFilter::FairEvtFilter()  
: TNamed(), fEventNr(0), fVerbose(0), fTestMode(0)  
Index: pgenerators/eventFilter/FairEvtFilter.h
```

```
=====
--- pgenerators/eventFilter/FairEvtFilter.h (revision 26438)
+++ pgenerators/eventFilter/FairEvtFilter.h (working copy)
@@ -126,7 +126,7 @@
  TDatabasePDG* fdbPdg;
  // constant holding a double number which is not a valid charge
  // this serves to indicate that the value has not been specified by the user
- static const Double_t kNoChargeSpecified = -999.9;
+ static const Double_t kNoChargeSpecified;

  TClonesArray* fParticleList; // list of particles in the event which was generated
  Int_t fVerbose; // level of commenting output for your filter, between 0 and 12
```

---

---

Subject: Re: FairEvtFilter problem with c++11  
Posted by [MartinJGaluska](#) on Fri, 14 Nov 2014 11:17:56 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Thank you for fixing the problem.

I assume that either a compiler flag or the standard behavior of gcc was changed as I found this information:

Quote:

Initializing static member variables other than const int types is not standard C++ prior C++11. The gcc compiler will not warn you about this (and produce useful code nonetheless) unless you specify the -pedantic option.

Good to know that the constants can be initialised in the cxx file. I assumed that I would need to initialise them in the constructors' initialisation lists, for instance here:

```
FairEvtFilter::FairEvtFilter()
: TNamed(), fEventNr(0), fVerbose(0), fTestMode(0)
```

Kind regards,  
Martin

---

---

Subject: Re: FairEvtFilter problem with c++11  
Posted by [StefanoSpataro](#) on Fri, 14 Nov 2014 11:50:56 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Indeed with new base packages C++11 is used, and few modifications are needed.

---