Subject: Adapter from Genfit to Pnd objects checked in
Posted by Anonymous Poster on Thu, 04 Jun 2009 17:58:21 GMT

Hi,

I checked in the promised code (r5722). There are still two issues to be solved. The first is that I put the files PndAdapters.{cxx,h} which contain the two functions in trackrep. They should not stay there. But it was the only place where the dependencies allowed me to put them. Could a framework expert, please make a suggestion on where to put this.

The other issue is more complicated: In PndTrack I save for the first and last point a FairTrackPar*. I have to use the pointer to be able to also put FairTrackParP (polymorphism). But this means that I can not write a proper copy ctor, because calling FairTrackPar(const FairTrackPar&) is not sufficient, because this will not copy the data in FairTrackParP. There is no solution to this problem unless FairTrackPar becomes purely virtual, and we have a method

virtual FairTrackPar FairTrackPar::clone()=0;

which calls the copy ctor in the derived classes, like

FairTrackPar* FairTrackParP::clone(){return FairTrackParP(*this);}

I have the same problem with the Print method: If we use FairTrackPar* in the PndTrack, a call to FairTrackPar::Print() inside PndTrack::Print() will never reach the implementation in FairTrackParP::Print() like it has to. For this we absolutely need a purely FairTrackPar.

Here is what I propose:

- We make FairTrackPar purely virtual
- We make another derived class like the P/H classes to replace the simple behaviour, which is now in FairTrackPar.

I'm looking forward to your comments.

Oh, and BTW, this 5722 commit contained a lot of changes, I coded since the meeting. So I hopw I didnt skrew up. Please test if all your fitting code still works. I will be out for a long weekend (starting tomorrow noon).

CU, Christian

Subject: Re: Adapter from Genfit to Pnd objects checked in
Posted by Tobias Stockmanns on Fri, 05 Jun 2009 07:13:51 GMT

Hi Christian,

maybe this is a stupid question, but if you change PndTrackPar to pointers within PndTrack, then the data is no longer stored within PndTrack but only the position of the place where the

data is stored.

Does this work with a TClonesArray? Do you have really access to the data, the pointers are pointing to?

Cheers,

Tobias

Subject: Re: Adapter from Genfit to Pnd objects checked in
Posted by Anonymous Poster on Fri, 05 Jun 2009 07:40:51 GMT
View Forum Message <> Reply to Message

Hi,

in all of my experience the answer is: Yes. ROOT dereferences those pointers.

But is it clear why I need pointers and thus have the problems with the lacking abstract interface?

Cheers, Christian

Subject: Re: Adapter from Genfit to Pnd objects checked in
Posted by Anonymous Poster on Thu, 11 Jun 2009 07:57:49 GMT
View Forum Message <> Reply to Message

Hi everybody,

I was wondering why there are no answers to my message from 1 week ago about the problem with PndTrack and FairTrackParX. These are important issues, and they can not be left to me alone to deal with. I neither have the responsibility nor the experience to decide these things on my own.

The track parameters are a very central part of the framework. If you would like to keep them as they are now, please tell me how to implement them in PndTrack.

I still suggest to change the way they are implemented, but as I said: My opinion on this can not be the only one. And there is a lot of code which already depends on these classes, so any changes are a lot of work for a lot of people.

Cheers, Christian

Subject: Re: Adapter from Genfit to Pnd objects checked in
Posted by Ralf Kliemt on Thu, 11 Jun 2009 09:07:08 GMT
View Forum Message <> Reply to Message

Hello Christian,

I have a bad feeling about storing pointers, mainly because I never do that. In my opinion you shaould store an index to the object in a TClonesArray. You would need one TClonesArray for each derived TrackPar class. This is the way it is done e.g. in the Mvd Digitization.

However, you are encouraged to make all the necessary changes to get it running!
For example moving the functionality of this Print() function to the daughter classes seems very convenient. The framework classes are write protected in svn anyway, so you have to distribute these on the development branch for a review.

Kind regards,
Ralf.

Subject: Re: Adapter from Genfit to Pnd objects checked in
Posted by Anonymous Poster on Thu, 11 Jun 2009 09:27:13 GMT
View Forum Message <> Reply to Message

Hi Ralf,

I want create a PndTrack from a Genfit Track. Do you suggest that I book TClonesArrays in the framework in the process of doing that? It would be better, if the PndTrack directly contains the track parameters.

Ans what is the problem about storing pointers? ROOT understands that.

Cheers, Christian

Subject: Re: Adapter from Genfit to Pnd objects checked in
Posted by Ralf Kliemt on Thu, 11 Jun 2009 09:54:40 GMT
View Forum Message <> Reply to Message

Hello again,

Surely you will have to register at least the TClonesArray which stores the PndTrack objects (I tell it just to avoid confusion). What I meant is to store the track parameter objects as well in TCAs and reference them by storing an index in the PndTrack.
Anyway you're right that the PndTrack should carry these objects by itself.
Can root carry these pointers even into a root file and recover it later on? I thought that not. If yes, there is no big point in discussing further - do it as you like.

Greetings, Ralf.

Subject: Re: Adapter from Genfit to Pnd objects checked in
Posted by StefanoSpataro on Thu, 11 Jun 2009 11:46:46 GMT
View Forum Message <> Reply to Message

I think that for ROOT the problem is when the memory address of the pointer is not available

anymore (i.e. it is pointing to an object which is stored in a friend file which is not loaded). In such a case, when analysing the first event of the tree, even if you are not using the initial information which has this kind of problem, root does and could crash. Storing indexes is much safer, an integer is always an integer and even if you click on it you will never crash.
But I don't know if this was solved in recent root versions.

Subject: Re: Adapter from Genfit to Pnd objects checked in
Posted by Sebastian Neubert on Thu, 11 Jun 2009 17:52:01 GMT
View Forum Message <> Reply to Message

Hi Stefano!

What you describe is a problem with the ownership of the object that is pointed to. When this object is living in a different branch, possibly in a different file then using standard pointers does indeed not work. One could use TRef but I do not recommend it. I have not made good experience with that.

Anyhow the situation we are having here is different: The Parameter object clearly is owned by the PndTrack object. This case can be handled by ROOT without any problem. You can even define a TClonesArray INSIDE your object to store several objects (See the Event class example that comes with ROOT). I do this all the time. It works well.

The only thing that you should care about in such a situation is the split level of the branch. I have good experiences with NOT allowing root to split such a branch.

Cheers! Sebastian.

Subject: Re: Adapter from Genfit to Pnd objects checked in
Posted by Anonymous Poster on Thu, 11 Jun 2009 18:17:53 GMT
View Forum Message <> Reply to Message

Hi,

I agree with what Sebastian said. But let me try to bring your focus back to the open problems.

If I do not use pointers, I have to make a definite choice whether to use FairTrackPar, FairTrackParP, or FairTrackParH in PndTrack. This seems unfavorable. If I use points, there remains the problem that FairTrackParam (the base class) is not a abstract interface class, which brings problems, like the impossibility of coying PndTrack (please see the first message of this thread).

Mohammad, what is your opinion on this issue?

Cheers, Christian