Subject: EvtGen - time information lost
Posted by Dominik Steinschaden on Wed, 01 Mar 2017 09:04:50 GMT
View Forum Message <> Reply to Message

Hi all,

I'm currently working on some algorithms for Pandaroot, which make use of the TOF counters (BTof/ScTil and FTof).
To test them under certain conditions I want to use the EvtGen generator to produce specific hyperon events.

now I realized that for such events the timing information of the tracks is wrong. It seems that the decay described in my decay file is handled by the evtGen till the final state particles are reached. and then all resulting tracks are simulated with the same start time which is equivalent to the event start time. Therefore the time information in the detectors is not correct any more. this effect can be very large for example for hyperons with a strong forward boost and therefore a large displaced vertex.

I attached a file, showing the time stamp distribution in the FTof detector for pbar p -> lambda lambda bar for 15 GeV primary momentum. Just to remind, the FTof is located around 7.8 meters in forward direction. a particle moving with speed of light should take around 26 ns to reach this detector. As shown there are signals in the detector after a few ns. Therefore for example TOF based Pid algorithms breaks down completely.

is there a possible work around to get the evtGen to also pass the correct time information to Geant3/4. or maybe geant3/4 can handle the decay of the lambdas instead of the evtGen.

LG Dominik

File Attachments
1) ftofTimeDist_genLast.pdf, downloaded 630 times

Subject: Re: EvtGen - time information lost
Posted by Ralf Kliemt on Wed, 01 Mar 2017 09:09:44 GMT
View Forum Message <> Reply to Message

Hi Dominik,

Try setting the Lambdas as stable in the decay file and let Geant handle the time and place for the decay. This should be in principle the solution.

Cheers
  Ralf

Subject: Re: EvtGen - time information lost
Posted by Tobias Stockmanns on Wed, 01 Mar 2017 09:15:36 GMT
View Forum Message <> Reply to Message

Hi Ralf and Dominik,

Ralfs solution has the drawback that you cannot specify how the lambdas will further decay. You will get both the p pi- and n pi0 decay.

Cheers,
Tobias

Subject: Re: EvtGen - time information lost
Posted by Dominik Steinschaden on Wed, 01 Mar 2017 09:38:52 GMT
View Forum Message <> Reply to Message

ok, but at least for the tests now this should work as a work around.

maybe a stupid question, how can I set the lambdas as stable? because I already tried to do something like this as a work around, and was not able to do this.

Subject: Re: EvtGen - time information lost
Posted by StefanoSpataro on Wed, 01 Mar 2017 16:09:08 GMT
View Forum Message <> Reply to Message

Hi,
it seems that in the PndEvtGenDirect the time information is not correctly propagated.
You could try to edit the class (pgenerators/EvtGen/EvtGenDirect/PndEvtGenDirect.cxx), and modify line 247:

```
primGen->AddTrack(Id, Px, Py, Pz, fX, fY, fZ, evtstdhep.getFirstMother(i),(nFD==-1 && nLD==-1),fE);
```

into:

```
primGen->AddTrack(Id, Px, Py, Pz, fX, fY, fZ, evtstdhep.getFirstMother(i),(nFD==-1 && nLD==-1),fE,fT);
```

and line 249:

```
primGen->AddTrack(Id, Px, Py, Pz, fX, fY, fZ);// default -1, true
```

into:

```
primGen->AddTrack(Id, Px, Py, Pz, fX, fY, fZ, -1, true, fE, fT);// default -1, true
```

recompile and tell me if it does work.

Subject: Re: EvtGen - time information lost
Posted by Dominik Steinschaden on Mon, 13 Mar 2017 15:24:18 GMT
View Forum Message <> Reply to Message

Hi all,

as Stefano suggested, I modified PndEvtGenDirect.cxx.

it was not as straight forward, but finally it worked.
As expected the time information was not handled to the pndStack (primGen->AddTrack).
when I changed this I had the problem that the used Units didn't fit.
Meanwhile I figured out that EvtGen is providing the time information in [mm] (natural units c=1)
and pndStack (more precise TParticle) is expecting time information in [s]. However I thought PandaRoot is using [ns] as unit for time. Therefore I'm not sure if i really fixed it correctly.

So I also implemented the following line:

```
fT=vxyz.get(0)/(1000*TMath::C());  //mm - > s conversion
```

The File Attached shows now the time stamp distribution for the FTof. Now it looks like expected!

I uploaded the changed file to

pandaroot/development/dsteinschaden/pgenerators/EvtGen/EvtGenDirect

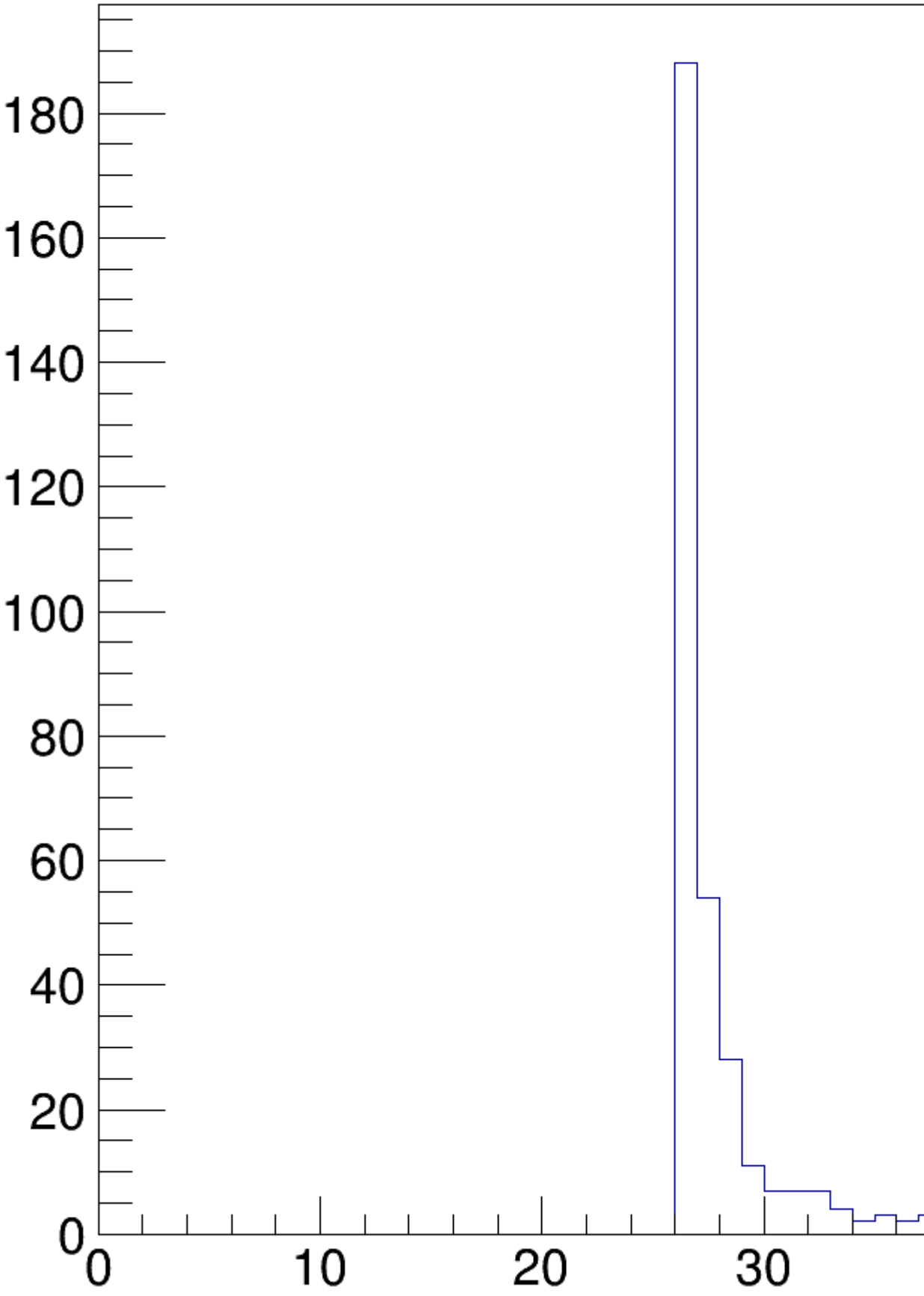
If someone wants to have a look. otherwise I'll ask someone to merge it with the trunk

beside I'm not sure if I'm correct with this mm to second conversion, and if seconds or nanoseconds should be used, I'm now worried about the timing information if other Generators are used . . .

regards Dominik

File Attachments

1) timestamps_EvtGen_fix1.png, downloaded 1049 times


180
160
140
120
100
80
60
40
20
0
0
10
20
30

Subject: Re: EvtGen - time information lost
Posted by StefanoSpataro on Mon, 13 Mar 2017 16:35:00 GMT
View Forum Message <> Reply to Message

Hi, you are right, Geant uses seconds but we Panda use nanoseconds. If you check the ProcessHits of your detector, you can see this conversion:

fTime = gMC->TrackTime() * 1.0e09;

Then, it is correct that EvtGen provides seconds to Geant. After, the Montecarlo will convert the time info into nanoseconds inside our MCPoints.

Subject: Re: EvtGen - time information lost
Posted by Dominik Steinschaden on Wed, 15 Mar 2017 13:35:25 GMT
View Forum Message <> Reply to Message

Meanwhile the changes are also implemented in the current trunk Version.

EvtGen should now produce realistic TimeStamps also for longliving intermediate particles

regards Dominik