

---

Subject: MCTrack

Posted by [Klaus Götzen](#) on Thu, 15 Nov 2007 11:47:06 GMT

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

---

Hi!

I somehow stumbling always about the same problems... my question is, where and when the MCTrack array is created and filled, so that I can access that from a task.

What I'd like to do is to access the MCTrack array of a simulated event, without running a transport code. My simple idea would be to init an arbitrary event generator, which produces the tracks and somehow stores these primaries.

A code snippet from my macro looks like:

.  
.  
.

```
CbmRunSim *fRun = new CbmRunSim();
```

```
fRun->SetOutputFile("fastsim.root");
```

```
CbmPrimaryGenerator* primGen = new CbmPrimaryGenerator();
```

```
fRun->SetGenerator(primGen);
```

```
CbmEvtGenGenerator* evtGen = new CbmEvtGenGenerator("output.evt");
```

```
primGen->AddGenerator(evtGen);
```

```
primGen->DoTracking(kFALSE);
```

```
CbmFastSim* fastSim = new CbmFastSim();
```

```
fRun->AddTask(fastSim);
```

.  
.  
.

but in my task CbmFastSim MCTrack is not accessible, I get the message

-E- CbmRootManager Branch: MCTrack not found in Tree

-W- CbmFastSim::Init: No MCStack array!

Most likely this is no big deal to solve. Somebody has a hint for me?

Cheers,  
Klaus

---

Subject: Re: MCTrack

Posted by [Mohammad Al-Turany](#) on Thu, 15 Nov 2007 12:20:06 GMT

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

---

Hi klaus,

The MCTrack is filled when finishing an event (in your case after filling the Stack). but your task is trying to access the MCTrack in the initialization phase of a simulation session (CbmRunSim!). So either you do this in two steps:

1. in your macro, remove the CbmFastSim and run it. It will produce an output file(fastsim.root)  
In fact, here you just copy the information from your event generator format to a format similar to the full simulation (There is not transport at all here)
2. create a new macro (Analysis macro, ie: CbmRunAna), set input file (output of step 1), add your task , and you will see the MCTrack and can use them

Or:

if you want to do it in one step, you have to use Stack, instead of MCTrack:

```
CbmStack *fStack= gMC->GetStack();  
and from CbmStack you have access to all info on the fly  
before they are filtered(if?) and written out as an array of  
MCTracks
```

regards  
Mohammad

---

Subject: Re: MCTrack  
Posted by [Klaus Götzen](#) on Thu, 15 Nov 2007 12:57:26 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hi Mohammad!

Thanks for your reply! The first method I just discussed with Dipak some minutes ago. I'd like to try the second option as well, but I don't know where this gMC comes from. I already tried to grep over all source code without finding out what to include to have access to it. There are thousand matches where gMC is accessed, but none where it is initialized ...

Regards,  
Klaus

---

Subject: Re: MCTrack  
Posted by [Stefano Spataro](#) on Thu, 15 Nov 2007 13:24:13 GMT

---

Hi,  
gMC is a global variable which is defined by root/vmc, not by PandaRoot but somewhere "higher" (such as gStyle gROOT and similar) in TVirtualMC.h, and it is of a TVirtualMC type (of course)

I don't know if these informations can be of help.

---

---

Subject: Re: MCTrack

Posted by [Mohammad Al-Turany](#) on Thu, 15 Nov 2007 14:10:02 GMT

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

---

Hi,

gMC is the global pointer to TGean3/TGeant4. it is created when ever you start a simulation session. As Stefano wrote it is of type TVirtualMC which is the base class for Tgeant3 and TGeant4.

Is it not initialized or what is going wrong?

Mohammad

---

---

Subject: Re: MCTrack

Posted by [Klaus Götzen](#) on Thu, 15 Nov 2007 14:17:40 GMT

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

---

Hi!

No, I just had to include "TVirtualMC.h", that was the problem. In principle it's working fine now!

But maybe I could ask another (more general) question. I'm trying to port some of the classes from the FastSim in the other framework to that directory (for instance FsmTrack, FsmResponse etc.) So I simply copied over FsmTrack.cc and FsmTrack.hh.

The compiling and linking works, but in runtime, I get at the first point I try to use the FsmTrack the error

```
/misc/cbmsoft/Debian3.1/new/tools/root/bin/root.exe: relocation error:
```

```
/u/kgoetzen/work/fairroot3/build/lib/libfastsim.so: undefined symbol: _ZN8FsmTrackC1Ev
```

Probably I have to add the class somewhere in a config/linkdef/Makefile?

Regards,  
Klaus

PS: Yees, I know.... simple C++/C basics

---

Subject: Re: MCTrack  
Posted by [Mohammad Al-Turany](#) on Thu, 15 Nov 2007 14:25:25 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hi,

In this new classes do you have ClassDef and ClassImp macros?

you have also to add these classes to the XXXLinkDef.h file in your directory.

Mohammad

---

---

Subject: Re: MCTrack  
Posted by [Stefano Spataro](#) on Thu, 15 Nov 2007 14:27:12 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

You forgot to add FsmTrack.cc in CMakeLists.txt, maybe this is the problem. And I cannot see your fastsim in the main Makefile.am.

In each case I remember you that all the classes should be named PndXXX  
Bye

Ste

---

---

Subject: Re: MCTrack  
Posted by [Klaus Götzen](#) on Thu, 15 Nov 2007 14:47:18 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

Hi Mohammad and Stefano!

Thanks a lot, now everything basically works as I wanted!

Best regards,  
Klaus

---