Subject: FairMCEventHeader
Posted by Volker Friese on Fri, 16 Mar 2012 16:06:04 GMT
View Forum Message <> Reply to Message

Hi,

from CBM, we would need to store the event plane angle (from the generator) as MC event variable. The proper place seems to be FairMCEventHeader. Is it possible to add a corresponding member to this class, with proper accessor and modifier?

More strategically thinking, it may happen again in the future that a particular application wants to store additional info in the MC event header (e.g., settings of the event generator). What would you propose as solution for this? I understand that FairMCEventHeader is used by the framework, e.g. for the run number and maybe other things. Of course, we can implement an additional class and branch in the tree, but then would have to change all generator classes to fill that one.

What do you think is the best solution? Some inheritance scheme?

Subject: Re: FairMCEventHeader
Posted by Florian Uhlig on Sun, 18 Mar 2012 19:01:41 GMT
View Forum Message <> Reply to Message

Hi Volker,

Quote:
from CBM, we would need to store the event plane angle (from the generator) as MC event variable. The proper place seems to be FairMCEventHeader. Is it possible to add a corresponding member to this class, with proper accessor and modifier?

in principle it would be possible to add the new data members, but I think it is not a good idea, as you already pointed out in your last posting about the FairMCPoint [message #13149].

Quote:
More strategically thinking, it may happen again in the future that a particular application wants to store additional info in the MC event header (e.g., settings of the event generator). What would you propose as solution for this? I understand that FairMCEventHeader is used by the framework, e.g. for the run number and maybe other things. Of course, we can implement an additional class and branch in the tree, but then would have to change all generator classes to fill that one.
What do you think is the best solution? Some inheritance scheme?

Such as scheme is already implemented since more than a year. For CBM there is a CbmMCEvent and CbmMCEventHeader which can be used instead of the FairMCEvent or FairMCEventHeader. Both classes inherit from the base classes and add new data members. The CbmMCEventHeader for example add an additional data member for the reaction plane angle. To use the CbmMCEventHeader one has to add the following lines in the simulation

macro

```
    // Use the experiment specific MC Event header instead of the default one
    // This one stores additional information about the reaction plane
    CbmMCEventHeader* mcHeader = new CbmMCEventHeader();
    fRun->SetMCEventHeader(mcHeader);
```

To have an example I changed the run_sim.C macro in CbmRoot. This macro now calculates an event by event reaction plane and stores the information in the CbmMCEventHeader for later usage.

Hope this info answers your question.

Ciao

Florian

Subject: Nice
Posted by Volker Friese on Thu, 22 Mar 2012 09:39:47 GMT
View Forum Message <> Reply to Message

Quote:Such as scheme is already implemented since more than a year. For CBM there is a CbmMCEvent and CbmMCEventHeader which can be used instead of the FairMCEvent or FairMCEventHeader. Both classes inherit from the base classes and add new data members. The CbmMCEventHeader for example add an additional data member for the reaction plane angle. To use the CbmMCEventHeader one has to add the following lines in the simulation macro

Nice feature. I know that we discussed it some times ago, but was not aware that is has been implemented. Must be the age.

By the way: What happens if you do not specify the usage of CbmMCEventHeader in the macro? I suppose the CbmUrqmdGenerator crashes when doing the dynamic cast.