Subject: Access to EventHeader. after Digitization
Posted by Dominik Steinschaden on Fri, 27 May 2016 13:16:29 GMT
View Forum Message <> Reply to Message

Hi all,

At the moment I try to analyse some data created using the time based simulation/digitization. Therefore I need access to the EventHeader. which stores the MC Event time for every event.

So far I have access to all the data stored in the output file of the digitization stage, beside the Event time information.

Below a short version of the Macro I'm using is attached. May I have to do something different to access these information, so if somebody has an idea pleas tell me

In this macro the EventTime as well as the individual Timestamps of all Tracks hitting the SciTil should be displayed.
But only the SciTil Timestamps are displayed correctly. For the EventTime only 0 is displayed.
If I look into the digi.root files using the TBrowser, the Eventheader. is filled with the correct, non zero EventTimes

regards Dominik

File Attachments

1) test.C, downloaded 520 times

Subject: Re: Access to EventHeader. after Digitization
Posted by StefanoSpataro on Fri, 27 May 2016 15:00:40 GMT
View Forum Message <> Reply to Message

Hi,
I have the feeling that in such a way you retrieve the event header only one time and before the event loop, and not event by event. I have never used in such a way the code, in general I have always set the branch addess from the tee.
One example can be found in macro/gem/TB_checkHitReconstruction.C. I don0t know of moving the GetObject inside the event loop will help (but you could try).

Subject: Re: Access to EventHeader. after Digitization
Posted by Dominik Steinschaden on Tue, 31 May 2016 13:48:29 GMT
View Forum Message <> Reply to Message

as expected, moving it into the loop didn't help.
But I was able to access the information by directly opening the tree and setting the branch address.

Thanks for the help.

ps: its a little bit counter intuitive that the way one can access the information stored by the

detectors is not working for the EventHeader. may we can update this sometimes?

Dominik

Subject: Re: Access to EventHeader. after Digitization
Posted by Michael Papenbrock on Thu, 17 Nov 2016 12:18:05 GMT
View Forum Message <> Reply to Message

Hi!

I just ran into a similar problem, but within a FairTask, which was also executed after digitization. I used the following code in order to access the event time:

```
FairRootManager *ioManager = FairRootManager::Instance();
  if (FairRunAna::Instance()->IsTimeStamp()) {
    Double_t eventTime = ioManager->GetEventTime();
  }
```

Whether I call it in the Init() or Exec() method yields zero. However, the event times in the digi root-file look fine. Has this been resolved or is there a common workaround?

Subject: Re: Access to EventHeader. after Digitization
Posted by Radoslaw Karabowicz on Thu, 17 Nov 2016 14:11:19 GMT
View Forum Message <> Reply to Message

Did you try to run it without the

if (FairRunAna::Instance()->IsTimeStamp()) {

check?

Subject: Re: Access to EventHeader. after Digitization
Posted by Michael Papenbrock on Thu, 17 Nov 2016 14:18:49 GMT
View Forum Message <> Reply to Message

Yes, same thing, unfortunately.

Subject: Re: Access to EventHeader. after Digitization
Posted by Radoslaw Karabowicz on Sat, 19 Nov 2016 11:15:53 GMT
View Forum Message <> Reply to Message

Dear Michael,

Indeed it seems that I have introduced a bug while creating FairFileSource.

It causes it not to read the fEventTime from the input tree, when
the user does not run in TimeBased mode.

I have traced down the error to coming from:

FairFileSource.cxx
FairFileSource::ReadEvent(UInt_t i){
...
SetEventTime();
...
}

This line should be changed to fEventTime = GetEventTime();
In the dev/ it is line number 331.

Is it possible for you to change the line, build FairRoot again, and test if digitization
works as before, and in the reconstruction you do retrieve the event time properly?

cheers
radek

Subject: Re: Access to EventHeader. after Digitization
Posted by Michael Papenbrock on Mon, 21 Nov 2016 09:05:24 GMT
View Forum Message <> Reply to Message

Dear Radek,

I implemented the change and it seems to be working for the first event, i.e. the event time for the first event is carried over and can be accessed. However, for subsequent events I get the output below. The returned value of GetEventTime() is always the one of the first event.

Error in <TClonesArray::At>: index 0 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 1 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 2 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 3 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 4 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 5 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 6 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 7 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 8 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 9 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 10 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 11 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 12 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 13 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 14 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 15 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 16 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 17 out of bounds (size: 0, this: 0x35df510)

Error in <TClonesArray::At>: index 18 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 19 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 20 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 21 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 22 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 23 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 24 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 25 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 26 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 27 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 28 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 29 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 30 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 31 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 32 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 33 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 34 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 35 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 36 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 37 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 38 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 39 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 40 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 41 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 42 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 43 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 44 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 45 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 46 out of bounds (size: 0, this: 0x35df510)
Error in <TClonesArray::At>: index 47 out of bounds (size: 0, this: 0x35df510)

Cheers,
Michael

Subject: Re: Access to EventHeader. after Digitization
Posted by Radoslaw Karabowicz on Mon, 21 Nov 2016 17:08:41 GMT
View Forum Message <> Reply to Message

Hi Michael,

could you send me the task and macro that you are trying to run?
I would like to test it myself.

yours
radek

Subject: Re: Access to EventHeader. after Digitization

Posted by Michael Papenbrock on Tue, 22 Nov 2016 10:33:59 GMT
View Forum Message <> Reply to Message

Dear Radek,

I have attached the task together with the project files to run it. "sim_complete.C" and "digi_complete.C" produce the necessary data, "reco_complete.C" executes the task. Let me know if you need anything else!

Best regards,
Michael

File Attachments

1) TimingTests.tar.gz, downloaded 422 times

Subject: Re: Access to EventHeader. after Digitization
Posted by Radoslaw Karabowicz on Tue, 22 Nov 2016 15:38:07 GMT
View Forum Message <> Reply to Message

Hi Michael,

This time I want you to add three lines in two files:

```
void FairFileSource::FillEventHeader(FairEventHeader* feh)
{
  fEventTime = GetEventTime();        // <===== ADD THIS LINE
  feh->SetEventTime(fEventTime);


Int_t FairRootManager::ReadNonTimeBasedEventFromBranches(Int_t Entry)
{
    if ( fSource ){
        TObject *Obj;
        fListOfNonTimebasedBranchesIter->Reset();
        while ( (Obj=fListOfNonTimebasedBranchesIter->Next())) {
   //   LOG(INFO) << "GETTING EVENT " << Entry << " FOR OBJ >" << Obj->GetName() <<
"<" << FairLogger::endl;
            fSource->ReadBranchEvent(Obj->GetName(),Entry);
        }
    }else{
      return 0;
    }

    fSource->FillEventHeader(fEventHeader);    // / <===== ADD THIS LINE
    fCurrentTime = fEventHeader->GetEventTime();   // / <===== ADD THIS LINE

    return 1;
}
```

Then please rebuild FairRoot and let me know the results.
I still try to understand, why it seems to have been working before.

yours
radek

Subject: Re: Access to EventHeader. after Digitization
Posted by Michael Papenbrock on Wed, 23 Nov 2016 11:30:37 GMT
View Forum Message <> Reply to Message

Dear Radek,

this produced some interesting results. First, let me show you the number I got from a simulation of just 10 events:

ioManager->GetEventTime(): 8.89033   myEventHeader->GetEventTime(): 8.89033
ioManager->GetEventTime(): 239.621   myEventHeader->GetEventTime(): 8.89033
ioManager->GetEventTime(): 290.828   myEventHeader->GetEventTime(): 239.621
ioManager->GetEventTime(): 300.156   myEventHeader->GetEventTime(): 290.828
ioManager->GetEventTime(): 312.479   myEventHeader->GetEventTime(): 300.156
ioManager->GetEventTime(): 382.595   myEventHeader->GetEventTime(): 312.479
ioManager->GetEventTime(): 386.654   myEventHeader->GetEventTime(): 382.595
ioManager->GetEventTime(): 428.155   myEventHeader->GetEventTime(): 386.654
ioManager->GetEventTime(): 439.536   myEventHeader->GetEventTime(): 428.155
ioManager->GetEventTime(): 445.36   myEventHeader->GetEventTime(): 439.536
ioManager->GetEventTime(): 445.36   myEventHeader->GetEventTime(): 445.36
ioManager->GetEventTime(): 445.36   myEventHeader->GetEventTime(): 445.36
ioManager->GetEventTime(): 445.36   myEventHeader->GetEventTime(): 445.36
ioManager->GetEventTime(): 445.36   myEventHeader->GetEventTime(): 445.36
ioManager->GetEventTime(): 445.36   myEventHeader->GetEventTime(): 445.36
ioManager->GetEventTime(): 445.36   myEventHeader->GetEventTime(): 445.36
ioManager->GetEventTime(): 445.36   myEventHeader->GetEventTime(): 445.36
ioManager->GetEventTime(): 445.36   myEventHeader->GetEventTime(): 445.36
ioManager->GetEventTime(): 445.36   myEventHeader->GetEventTime(): 445.36
ioManager->GetEventTime(): 445.36   myEventHeader->GetEventTime(): 445.36

After simulation, I have 10 events in the root file as defined in my macro. After digitization I have 11, which is also expected, but after reconstruction there are 20. However, looking at the output above the last 10 seem to be more or less the same. Also, it is noteworthy that the FairEventHeader lags one event behind the FairRootManager. Overall, it looks like a step in the right direction!

Cheers,
Michael

Subject: Re: Access to EventHeader. after Digitization
Posted by Tobias Stockmanns on Wed, 23 Nov 2016 12:03:15 GMT

View Forum Message <> Reply to Message

Dear Michael,

the difference between the number of events in simulation, digitization and reconstruction is not surprising. The simulation file is identical in TimeBased and EventBased simulation, so you have here the number of events you have simulated. In the digitization stage the ordering of the digi stream is randomized depending on your detector and afterwards sorted by the TimeStamp. The digis of one event are not longer in the same entry as in the simulation and they can be completely mixed up. The last entry in the digitization file contains all data of the sorter which has not finished sorting before the last event happend.

The number of events in the reco file can be anything from 1 to (almost) infinity. It only depends on the way you take the data from the digi file via the GetData method. You can have less events if you have events merged together or more, if one event is split into many "pseudo-events".

Cheers,

Tobias

Subject: Re: Access to EventHeader. after Digitization
Posted by Radoslaw Karabowicz on Thu, 24 Nov 2016 08:06:55 GMT
View Forum Message <> Reply to Message

Hi Michael,

```
  ioManager->FillEventHeader(myEventHeader);
```

I added the above line, and then I do get proper data from myEventHeader.

yours
radek

Subject: Re: Access to EventHeader. after Digitization
Posted by Michael Papenbrock on Thu, 24 Nov 2016 09:33:40 GMT
View Forum Message <> Reply to Message

Dear Radek,

great! Thank you for the help!

Also thank you to Tobias for the explanations!

Cheers,
Michael
