Subject: Access to MC information after the digitization stage
Posted by Dominik Steinschaden on Tue, 04 Aug 2015 12:01:18 GMT
View Forum Message <> Reply to Message

Hallo,

recently I updated the stored Fairlinks in the implementation of the SciTil detector.

Now I'm trying to make use out of them in a timebased simulation . Therefore I created root files for the simulation and the digitization stage.
As usual I'm able to get access to the stored data in the root files using a macro which is basically based on the example shown on the wiki page:
https://panda-wiki.gsi.de/foswiki/bin/view/Computing/PandaRootRawHits

has someone maybe a simple example macro, that shows me how to load both root files (sim.root and digi.root) in a macro and get for example for every hit in the digi.root file the corresponding MCTrack infos using these Fairlinks.

or is there an other standard way of accessing these data. (like a mandatory use of the FairrunAna() class and writing and compiling a specific Task for analysing. but this seems rather complicate if I just want to compare the numbers of primaries in a detector after the simulation stage and after the digitization for example)

regards dominik

Subject: Re: Access to MC information after the digitization stage
Posted by StefanoSpataro on Wed, 05 Aug 2015 15:18:31 GMT
View Forum Message <> Reply to Message

Dear Dominik,
the FairLink structure has been changed and I believe there are no recent tutorials about, I can suggest you to take a look into the Exec function of PndMCMatchNewLinks/PndMCIdealTrackFinderNewLinks.cxx to see how to move between different data levels with FairLinks.

Like line 98:

```
PndMCTrack *mc = (PndMCTrack *)
FairRootManager::Instance()->GetCloneOfLinkData(iter->first);
```

where iter->first is the FairLink of your supposed digi.

Or lines 115/116:

```
FairMultiLinkedData gemhitlink =
gemhit->GetLinksWithType(FairRootManager::Instance()->GetBranchId("GEMPoint"));
```

PndGemMCPoint *gempoint = (PndGemMCPoint *)
FairRootManager::Instance()->GetCloneOfLinkData(gemhitlink.GetLink(0));

Where from the GemHit it finds the link of the GemPoint, and retrieve the PndGemMCPoint object.

Hope it helps, Tobias is the expert (and he should be in vacation right now).

Subject: Re: Access to MC information after the digitization stage
Posted by Dominik Steinschaden on Tue, 11 Aug 2015 13:47:46 GMT
View Forum Message <> Reply to Message

Meanwhile I solved my problem accessing the data and linking the MC data.
I don't know if someone else may have similar troubles with the FairLinks so i post my solutions. Maybe someone can make use out of it.

Attached and on the end of the Post you find a basic root macro which gives access to the sorted data in the digitization output and also to the linked simulation and MC output.
(For a solution using the Pandaroot classes look at the following post)

This macro does not make use out of special Panda or FairRoot classes, but mostly uses basic root commands. Therefore one has more direct access to the data but on the other side loose some comfort.

to use the Fairlinks it is important that already in the simulation stage and all ongoing data processing stages "fRun->SetUseFairLinks(kTRUE);" is set!

```
void link_basic_1(){

  TString InputSimFile ="data_sim/sim_s_gc_G4_6.root";
  TString InputDigFile ="data_digi/digi_s_gc_G4_6_1000.root";

  TFile *simFile = TFile::Open(InputSimFile);
  TTree *simTree = (TTree*) simFile->Get("cbmsim");

  TFile *digFile = TFile::Open(InputDigFile);
  TTree *digTree = (TTree*) digFile->Get("cbmsim");

  FairMCEventHeader* simEvHeader = NULL;
  FairEventHeader* digEvHeader = NULL;

  TClonesArray* mcTrackArray = new TClonesArray("PndMCTrack");
  TClonesArray* scitPointArray = new TClonesArray("PndSciTPoint");
  TClonesArray* scitHitArray = new TClonesArray("PndSciTHit");

  TBranch* mcTrackBranch = (TBranch*) simTree->GetBranch("MCTrack");
  TBranch* scitPointBranch = (TBranch*) simTree->GetBranch("SciTPoint");
  TBranch* scitHitBranch = (TBranch*) digTree->GetBranch("SciTSortedHit");
```

```
  simTree->SetBranchAddress("MCTrack",&mcTrackArray);
  simTree->SetBranchAddress("SciTPoint",&scitPointArray);
  simTree->SetBranchAddress("MCEventHeader.",&simEvHeader);

  digTree->SetBranchAddress("SciTSortedHit",&scitHitArray);
  digTree->SetBranchAddress("EventHeader.",&digEvHeader);


  PndMCTrack* mcTrack = NULL;
  PndSciTPoint* scitPoint = NULL;
  PndSciTHit* scitHit = NULL;


  for (Int_t i_Event=0; i_Event<scitHitBranch->GetEntries();i_Event++){ // ----- loop oveer
digitized events ---

   scitHitBranch->GetEntry(i_Event);

    for (Int_t i_hitArray=0; i_hitArray < scitHitArray->GetEntries() ;i_hitArray++){
      scitHit = (PndSciTHit*) scitHitArray->At(i_hitArray);

 // ------------- do some stuff with the SciTHit data here -------------

      // be carefull there are more links stored.
      // GetLink(0) seems to point to the MC Track
      // GetLink(1) seems to point to the SciTPoint (the simulation output)
      // I guess this is different for the reco or pid Output files and maybe also for other detectors

      scitPointBranch->GetEntry(scitHit->GetLink(1).GetEntry());
      scitPoint = (PndSciTPoint*) scitPointArray->At(scitHit->GetLink(1).GetIndex());

      // ------------- do some stuff with the SciTPoint data here -------------


      mcTrackBranch->GetEntry(scitPoint->GetLink(0).GetEntry());
      mcTrack = (PndMCTrack*) mcTrackArray->At(scitPoint->GetLink(0).GetIndex());

      //-------------- do some stuff with the corresponding mcTrack data here ----------------

    }
  } // ----- loop oveer digitized events --- END ---

  simFile->Close();
  digFile->Close();

}
```

File Attachments

1) link_basic_1.C, downloaded 427 times

Subject: Re: Access to MC information after the digitization stage
Posted by Tobias Stockmanns on Tue, 11 Aug 2015 14:00:31 GMT
View Forum Message <> Reply to Message

Dear all,

please be careful with using GetLink(...). It is not defined that GetLink(0) always points to a MCTrack and GetLink(1) to an SciTPoint.

Better use GetLinksWithType(branchName). This returns a FairMultiLinkedData object with links all from the same type. You have to loop over the FairMultiLinkedData because you could have more than one link of the same type.

Cheers,

Tobias

Subject: Re: Access to MC information after the digitization stage
Posted by Dominik Steinschaden on Tue, 11 Aug 2015 14:07:00 GMT
View Forum Message <> Reply to Message

Attached you find an macro using the PandaRoot classes to manage the data and using the FairLinks to go through the different data stages.

```
using std::cout;
using std::endl;

void link_basic_2(){

  // Set your inpute files here
  TString simFile ="data_sim/sim_s_gc_G4_6.root";
  TString parFile = "data_sim/sim_s_gc_G4_6_params.root";
  TString digiFile ="data_digi/digi_s_gc_G4_6_1000.root";
  TString outFile = "output.root";

  int Ev_start = 24;
  int Ev_end = 61;     // take 0 for all events

 TStopwatch timer;
  timer.Start();

  // -----   Initialize Run manager   ------------------------------------------
 FairRunAna *fRun= new FairRunAna();
 fRun->SetInputFile(digiFile);
 fRun->AddFriend(simFile);
 fRun->SetOutputFile(outFile);
 fRun->SetUseFairLinks(kTRUE);
```

```
  // -----  Parameter database   ----------------------------------------
  TString allDigiFile = gSystem->Getenv("VMCWORKDIR");
  allDigiFile += "/macro/params/all.par";

  FairRuntimeDb* rtdb = fRun->GetRuntimeDb();
  FairParRootFileIo* parInput1 = new FairParRootFileIo();
  parInput1->open(parFile.Data());

  FairParAsciiFileIo* parIo1 = new FairParAsciiFileIo();
  parIo1->open(allDigiFile.Data(),"in");
  rtdb->setFirstInput(parInput1);
  rtdb->setSecondInput(parIo1);

  fRun->Init();

  // ----------------- Initialize Used Variables and Branches ------------------------------
  FairRootManager* ioman = FairRootManager::Instance();

  TClonesArray* mcTrackArray = (TClonesArray*)ioman->GetObject("MCTrack");  // if not
"initialized" here it may produces error at the first access
  TClonesArray* scitPointArray = (TClonesArray*)ioman->GetObject("SciTPoint");
  TClonesArray* scitHitArray = (TClonesArray*)ioman->GetObject("SciTHit");
  PndMCTrack* mcTrack = NULL;
  PndSciTPoint* scitPoint = NULL;
  PndSciTHit* scitHit = NULL;
  FairMultiLinkedData mcLink, pointLink;

  if (Ev_end == 0)  Ev_end = Int_t((ioman->GetInChain())->GetEntries())-1;
  for (int i_Event=Ev_start; i_Event <= Ev_end; i_Event++) { // ------- Loop over digitized events
---------------

    ioman->ReadEvent(i_Event);

    for (Int_t i_Array=0; i_Array < scitHitArray->GetEntries() ;i_Array++){ //------- loop over array
entries
      scitHit = (PndSciTHit*) scitHitArray->At(i_Array);

 // ------------- do some stuff with the SciTHit data here -------------

      pointLink = scitHit->GetLinksWithType(ioman->GetBranchId("SciTPoint")); // --- get all links
to the wanted Branch.
      //mcLink = scitHit->GetLinksWithType(ioman->GetBranchId("MCTrack"));      //You can also
get direct Access to "MCTrack"
      scitPoint = (PndSciTPoint* ) ioman->GetCloneOfLinkData(pointLink.GetLink(0)); // -- load
the Data of the chosen Link

      // ------------- do some stuff with the SciTPoint data here -------------

      mcLink = scitPoint->GetLinksWithType(ioman->GetBranchId("MCTrack"));   // you can also
use the Links of the linked file.
      mcTrack = (PndMCTrack *) ioman->GetCloneOfLinkData(mcLink.GetLink(0));
```

```
    // ------------ do stuff with the McTrack data ---------------------

    }
  }   //------- end of event loop


  timer.Stop();
  Double_t rtime = timer.RealTime();
  Double_t ctime = timer.CpuTime();
  cout << endl << endl;
  cout << "Macro finished succesfully." << endl;
  cout << "Output file is "    << outFile << endl;
  cout << "Parameter file is " << parFile << endl;
  cout << "Real time " << rtime << " s, CPU time " << ctime << " s" << endl;
  cout << endl;
  // -----------

}
```

File Attachments

1) link_basic_2.C, downloaded 478 times