Subject: TClonesArray Trouble with PndEmcDigi
Posted by Marcel Tiemens on Fri, 02 Sep 2016 12:01:41 GMT

I have some task that creates PndEmcClusters, and in a next task, I want to access those clusters' member digis. I do that using the FairLink information:

PndEmcCluster* cluster = (PndEmcCluster*) fClusterArray->At(iClus);

// add its corresponding digis to the digi array
FairMultiLinkedData digiLinks =
cluster->GetLinksWithType(FairRootManager::Instance()->GetBranchId("EmcDigiSorted"));

for (Int_t iDigi=0; iDigi<digiLinks.GetNLinks(); iDigi++){
    PndEmcDigi* myDigi =
(PndEmcDigi*)FairRootManager::Instance()->GetCloneOfLinkData(digiLinks.GetLink(iDigi));
    if(myDigi) {
        PndEmcDigi* newDigi = new((*fDigiArray)[fDigiArray->GetEntriesFast()])
PndEmcDigi(*myDigi);
 delete(myDigi);
    } else {
        std::cout << "-E in PndEmcMergePreclusters::Exec() FairLink " << digiLinks.GetLink(iDigi)
<< " to EmcDigi delivers null" << std::endl;
    }
}

In this example, I want to load the digis into a TClonesArray called fDigiArray. So far, so good. I know that the digi objects are retrieved, because calling newDigi->Print() in this loop shows me the digi information. However, for some reason, instead of appending new digis to the TClonesArray, it puts the first one in the first entry, and then overwrites this entry for each new digi that it finds. I can modify the writing to the TClonesArray by replacing PndEmcDigi* newDigi = new((*fDigiArray)[fDigiArray->GetEntriesFast()]) PndEmcDigi(*myDigi); by PndEmcDigi* newDigi = new((*fDigiArray)[ind++]) PndEmcDigi(*myDigi); for some index ind, and this does increase the size of the TClonesArray. However, if I try to loop over the entries in the TClonesArray using for example

for (int tc=0; tc<fDigiArray->GetEntriesFast(); tc++) {
    cout << "\tEntry " << tc << " ";
    PndEmcDigi* theDigi = (PndEmcDigi*)fDigiArray->At(tc);
    theDigi->Print();
}cout << "\n " << endl;
 I get a segmentation violation. I somehow can't access the content of the array. So, something very strange is going on, but I can't figure out what. The strange thing is, if I try the same procedure with TLine as a dummy class instead of PndEmcDigi, everything works fine from the beginning. Does anyone have an idea what could be going on?