Subject: Crash in TPC digitization
Posted by StefanoSpataro on Wed, 13 Oct 2010 15:55:24 GMT
View Forum Message <> Reply to Message

After some tries, I surrender, I hope that somebody else can take a look.

I have produced 10k events with dpm, using macro/pid/run_sim_tpccombi_dpm.C ... everything fine.
I run macro/pid/run_digi_tpccombi.C, I have a crash.

```
Error: Symbol #include is not defined in current scope  run_digi_tpccombi.C:148:
Error: Symbol exception is not defined in current scope  run_digi_tpccombi.C:148:
Syntax Error: #include <exception> run_digi_tpccombi.C:148:
Error: Symbol G__exception is not defined in current scope  run_digi_tpccombi.C:148:
Error: type G__exception not defined
FILE:/d/panda02/spataro/pandaroot/macro/pid/./run_digi_tpccombi.C LINE:148
*** Interpreter error recovered ***
```

In order to isolate it, I have commented out some stuff.
If I run, as tasks, only:

```
PndTpcClusterizerTask* tpcClusterizer = new PndTpcClusterizerTask();
if(mcMode=="TGeant3") tpcClusterizer->SetMereChargeConversion();
tpcClusterizer->SetPersistence();
fRun->AddTask(tpcClusterizer);

PndTpcDriftTask* tpcDrifter = new PndTpcDriftTask();
tpcDrifter->SetPersistence();
tpcDrifter->SetDistort(false);
fRun->AddTask(tpcDrifter);

PndTpcGemTask* tpcGem = new PndTpcGemTask();
tpcGem->SetPersistence();
fRun->AddTask(tpcGem);

PndTpcPadResponseTask* tpcPadResponse = new PndTpcPadResponseTask();
tpcPadResponse->SetPersistence();
fRun->AddTask(tpcPadResponse);
```

I can run 10k events (please note that I have turned on the persistency).
If I add the PndTpcElectronicTask, or if I run a macro with only the follwing task:

```
PndTpcElectronicsTask* tpcElec = new PndTpcElectronicsTask();
tpcElec->SetPersistence();
fRun->AddTask(tpcElec);
```

I have the crash again.
I have filled the code with cout, and I have found that the crash line is in FairRootManager::ForceFill() :

```
fOutTree->Fill();
```

Then it should be a problem of the data written into the file, and not in the tasks themselves.

After a discussion with Mohammad, it seems that this problem could rise when the data objects have no default constructor, or if there are some unitialized variable.
I tried to update PndTpcPrimaryCluster, PndTpcDriftedElectrons, PndTpcAvalance, PndTpcSignal, PndTpcSample, PndTpcDigi, but without any success. I have seen, however, that there are some unitialized variables, numbers but also std::vector and pointers.

I hope that some TPC expert could take a look, at least to reproduce the crash and then to investigate.

I give up.