Subject: Re: TPC MVD and GEM correlators
Posted by StefanoSpataro on Thu, 16 Jun 2011 13:56:28 GMT
View Forum Message <> Reply to Message

Hi,

Felix Boehmer wrote on Thu, 16 June 2011 14:52Hi Stefano,

do you have the latest version of the KalmanTask? It looks like you don't and there is one of the RecoHitProducers not properly defined.

It is the 12357, but I have seen you have done a modification at 14:30 which I had not. I will run again.

Quote:
To your questions:

Quote:
Ca we move the tpc clusterization to the digi macro, or remove it in the digi? At present we are running the same code, even if with different parameters, twice
I removed the clustering from the reco macro.

This means that the cluster code in the run_digi_tpc_*.C is fine, isn't it? I have seen it was a bit different from the one in the reco macro.

In the digi:

```
 PndTpcClusterFinderTask* tpcCF = new PndTpcClusterFinderTask();
  tpcCF->SetDigiPersistence(); // keep Digis refs in clusters
  tpcCF->SetPersistence();     // keep Clusters
  tpcCF->timeslice(10);        //in samples
  tpcCF->SetErrorPars(600,300);
  tpcCF->SetSimpleClustering(); // use PndTpcClusterFinderSimple
  fRun->AddTask(tpcCF);
```

In the reco:

```
 PndTpcClusterFinderTask* tpcCF = new PndTpcClusterFinderTask();
  //tpcCF->SetDigiPersistence(); // keep reference to digis in clusters
  tpcCF->SetPersistence(); // keep Clusters
  tpcCF->timeslice(4); //in samples
  tpcCF->SetThreshold(1);
  tpcCF->SetSingleDigiClusterAmpCut(0.);
  tpcCF->SetClusterAmpCut(0.); // cut on mean digi amplitude
  tpcCF->SetErrorPars(600.,400.);
  tpcCF->SetSimpleClustering(); // use PndTpcClusterFinderSimple
  fRun->AddTask(tpcCF);
```

Which of them?

Quote:
What you describe was my first approach. However, using the MVD hits after TPC+MVD correlation in a fit gives better results for the (possibly very long) extrapolation into the GEMs. I would keep it that way. I agree that it is slow, but it is the best I have right now. The crashes are completely isoloated to GEANE extrapolations into the GEMs as far as I can tell right now... I am working on it, as I said.

So maybe we can take out the first kalman. I am not scared by code crashes or geane crashed, these we can fix, it is matter of track cleaning, taking out tracks which could give errors in arithmetical calculations. I am more scared by crashes connected to memory usage, geane seems to eat a bit of memory and running it three times in the same macro... let's cross the fingers!

Quote:
Quote:Which particle hypothesis are you using for the kalman?
Is there a way to reduce all those messages? They make the log output file large
The particle hypothesis is obtained from MC information isndie the PndTpcRiemannTrackingTask. I can build in a setter if you want to have manual control during the reco.

Just to know, in stt case we are using the fixed particle hypothesis (muon as default). Probably the mc method should be also implemented in the RecoKalmanTask version.
But a flag, also for the verbose of the kalman, would be great (I suppose with lower priority).

Quote:
I can take a look at this after I have fixed all the other, more urgent problem. I would be very grateful if you could have a lokk at this. Probably the exchange would be a very technical task, but rather easy.

Ok, I have also some objections about the enum fDetectorId, because ti breaks a bit also the pid and analysis part, but first let's see what is happening with the "normal" version.