

dunetpc - Feature #18004

Ignore outer planes in protoDUNE data prep

10/23/2017 08:37 AM - David Adams

Status:	Closed	Start date:	10/23/2017
Priority:	Normal	Due date:	
Assignee:	David Adams	% Done:	0%
Category:		Estimated time:	0.00 hour
Target version:			
Description			
Each APA in the protoDUNE detector reads out two TPC volumes but the fields are set so no charge is collected from the outer TPC. We would like to have the option skip the outer collection plane to save a little time in the data prep stage of processing.			

History

#1 - 10/23/2017 09:00 AM - David Adams

As a first step, I have enhanced the test code for the DUNE geometries to display the ROP-channel and ROP-TPC mappings. This code is in `dunetpc/dune/Geometry/test/test_GeometryDune.cxx` Here is the ROP-TPC mapping for protoDUNE:

```
Cryostat 0 has 6 APAs
  APA 0 has 4 ROPs
    ROP 0 TPCs: 0, 1
    ROP 1 TPCs: 0, 1
    ROP 2 TPCs: 0
    ROP 3 TPCs: 1
  APA 1 has 4 ROPs
    ROP 0 TPCs: 2, 3
    ROP 1 TPCs: 2, 3
    ROP 2 TPCs: 2
    ROP 3 TPCs: 3
  APA 2 has 4 ROPs
    ROP 0 TPCs: 4, 5
    ROP 1 TPCs: 4, 5
    ROP 2 TPCs: 4
    ROP 3 TPCs: 5
  APA 3 has 4 ROPs
    ROP 0 TPCs: 6, 7
    ROP 1 TPCs: 6, 7
    ROP 2 TPCs: 6
    ROP 3 TPCs: 7
  APA 4 has 4 ROPs
    ROP 0 TPCs: 8, 9
    ROP 1 TPCs: 8, 9
    ROP 2 TPCs: 8
    ROP 3 TPCs: 9
  APA 5 has 4 ROPs
    ROP 0 TPCs: 10, 11
    ROP 1 TPCs: 10, 11
    ROP 2 TPCs: 10
    ROP 3 TPCs: 11
```

Robert tells me the outer TPCs are 0, 3, 4, 7, 8, 11, i.e. ROP 2 for APAs 0, 2, 4 and ROP 3 for APAs 1, 3, 5.

This is consistent with the spacepoint tests in `test_GemetryProtoDune` which report TPCs 1, 2, 5, 6, 9, 10.

#2 - 10/24/2017 08:39 AM - David Adams

Data prep is done by groups of channels with the grouping specified by the service interface `ChannelGroupService`. ProtoDUNE reco is specified in `protoDUNE_reco.fcl` which uses the service implementation `GeoApaChannelGroupService` which makes each APA a group.

I am modifying `GeoApaChannelGroupService` so its configuration specifies the ROPs to keep for each APA.

#3 - 10/24/2017 09:49 AM - David Adams

I have just pushed commit `0a07737` that adds the above capability to `GeoApaChannelGroupService` but changes the default configuration so that all planes are used in reconstruction. I.e. there should be no change in the reconstructed event. A few lines have been added to the log file.

I include Vito on this ticket so he can let us know here if any change is seen in the CI testing.

#4 - 10/24/2017 10:23 AM - David Adams

I have modified the fcl configuration to rename @local::changroup_apa to @local::changroup_apa_keepall and added the configuration changroup_apa_pd_drop_outer which has the above behavior (skips the outer collection planes). The latter is now the default for protoDUNE reconstruction. Reconstruction for FD and 35t still keep all planes.

I tested one event in my area and, as expected, data for one collection plane is empty or missing in my event displays and it is the plane that does not have the muon track.

Here is the relevant diff:

```
sw9:dunetpc> git diff ./dune/Utilities/services_dune.fcl
diff --git a/dune/Utilities/services_dune.fcl b/dune/Utilities/services_dune.fcl
index 80abd71..cb2f1a4 100644
--- a/dune/Utilities/services_dune.fcl
+++ b/dune/Utilities/services_dune.fcl
@@ -45,7 +45,7 @@ dunefd_services: {
  NuRandomService:          @local::dune_seedservice
  SignalShapingServiceDUNE: @local::dunefd_signalshapingervice
  ChannelStatusService:     @local::dunefd_channel_status
- ChannelGroupService:     @local::changroup_apa
+ ChannelGroupService:     @local::changroup_apa_keepall
}

dunefd_simulation_services: {
@@ -271,7 +271,7 @@ protodune_services: {
  NuRandomService:          @local::dune_seedservice
  SignalShapingServiceDUNE: @local::dunefd_signalshapingervice
  ChannelStatusService:     @local::dunefd_channel_status
- ChannelGroupService:     @local::changroup_apa
+ ChannelGroupService:     @local::changroup_apa_pd_drop_outer
}

protodune_simulation_services: {
@@ -336,7 +336,7 @@ dune35t_basic_services: {
  ChannelStatusService:     @local::dune35t_channel_status_may2016
  ChannelMapService:        @local::channelmap35t
  ChannelMappingService:    @local::dune35t_channelmapping
- ChannelGroupService:     @local::changroup_apa
+ ChannelGroupService:     @local::changroup_apa_keepall
}

dune35t_basic_services.DetPedestalService: @local::dune_dbped
```

Change protoDUNE to match the others to recover the old behavior.

I have pushed commit ad69319 with this change.

Vito, please let us know if you see changes consistent or inconsistent with the above. Events from protoDUNE reco should have fewer channels but similar numbers of hits and tracks.

#5 - 10/24/2017 10:40 AM - Vito Di Benedetto

The CI build for the commit 0a07737 into dunetpc didn't spot any difference in the output of protoDUNE.

The CI tests for protoDUNE and protoDUNE dual phase are successful.

The results of the CI build are available at:

http://dbweb5.fnal.gov:8080/LarCI/app/ns:dune/view_builds/index?offset=0&builds=dune_ci_slf%2F79+&builds=1

and in particular the CI test are available at:

http://dbweb5.fnal.gov:8080/LarCI/app/ns:dune/build_detail/phase_details?build_id=dune_ci_slf/79&platform=Linux%202.6.32-696.1.1.el6.x86_64&phase=ci_tests&buildtype=slf6%20e14:prof

The input/reference files used for protoDUNE single phase CI test are located at:

```
/pnfs/dune/persistent/users/vito/ci_tests_inputfiles/protoDUNE/detsim/protoDune_pion_2GeV_mono_detsim_Reference.root
/pnfs/dune/persistent/users/vito/ci_tests_inputfiles/protoDUNE/g4/protoDune_pion_2GeV_mono_g4_Reference.root
/pnfs/dune/persistent/users/vito/ci_tests_inputfiles/protoDUNE/gen/protoDune_pion_2GeV_mono_gen_Reference.root
/pnfs/dune/persistent/users/vito/ci_tests_inputfiles/protoDUNE/mergeana/protoDune_pion_2GeV_mono_mergeana_Reference.root
/pnfs/dune/persistent/users/vito/ci_tests_inputfiles/protoDUNE/reco/protoDune_pion_2GeV_mono_reco_Reference.root
```

While the input/reference files used for protoDUNE dual phase CI test are located at:

```
/pnfs/dune/persistent/users/vito/ci_tests_inputfiles/protoDUNE dp/detsim/protoDuneDP_muon_2GeV_mono_detsim_Reference.root
/pnfs/dune/persistent/users/vito/ci_tests_inputfiles/protoDUNE dp/g4/protoDuneDP_muon_2GeV_mono_g4_Reference.root
/pnfs/dune/persistent/users/vito/ci_tests_inputfiles/protoDUNE dp/gen/protoDuneDP_muon_2GeV_mono_gen_Reference.root
/pnfs/dune/persistent/users/vito/ci_tests_inputfiles/protoDUNE dp/reco/protoDuneDP_muon_2GeV_mono_reco_Reference.root
```

Could be that the event used by the CI is not affected by your code change?

The CI build testing your commit ad69319 into dunetpc is running now. I'll give an update on its results when ready.

#6 - 10/24/2017 10:45 AM - David Adams

Vito:

Thanks for following this.

There should be no change for 0a07737 other than a few lines added to the log file. Sounds consistent with you report.

For ad69319, we should see the changes described above for the single-phase protoDUNE processing. NO other detectors should be affected.

da

#7 - 10/24/2017 11:42 AM - Vito Di Benedetto

David:

the CI build testing dunetpc commit ad69319b completed without reporting any difference in the output of CI tests.

The CI web application link is:

http://dbweb5.fnal.gov:8080/LarCI/app/ns:dune/view_builds/index?offset=0&builds=dune_ci_slf%2F80+&builds=1

The CI tests are available at:

http://dbweb5.fnal.gov:8080/LarCI/app/ns:dune/build_detail/phase_details?build_id=dune_ci_slf/80&platform=Linux%202.6.32-696.1.1.el6.x86_64&phase=ci_tests&buildtype=slf6%20e14.prof

- Vito

#8 - 10/24/2017 01:11 PM - David Adams

- Status changed from Assigned to Closed

Thanks again Vito. I assume your test is not sensitive to this change and that all is OK here. I close this report.

Robert, Vito, please reopen if you see any problems or want to change the default behavior.