

dunetpc - Bug #13550

New prep data Wire output has different size than old caldata

08/15/2016 08:52 AM - David Adams

Status:	Closed	Start date:	08/15/2016
Priority:	Normal	Due date:	
Assignee:	David Adams	% Done:	0%
Category:		Estimated time:	0.00 hour
Target version:			
Description			
<p>Here is the report from Vito:</p> <p>Hi David,</p> <p>thank you to look at this.</p> <p>The reference files for dunetpc have been generated using the tag v06_00_01 using the simulation workflow. The FHiCL files used include the following:</p> <p>AntiMuonCutEvents_LSU_dune35t.fcl standard_g4_dune35t.fcl standard_detsim_dune35t.fcl standard_reco_dune35t.fcl standard_ana_dune35t.fcl</p> <p>and the only addition to these FHiCL is the configuration of a fixed seed for the random engine to ensure the reproducibility of the processing.</p> <p>You can access the logs of the CI test for the DUNE reco stage at http://lar-ci-history.fnal.gov:8080/LarCI/app/build_detail/test_details?build_id=lar_ci_beta/3104&platform=Linux%202.6.32-573.26.1.el6.x86_64&phase=ci_tests&test=ci_reco_regression_test_dunetpc</p> <p>there are a "stdout" and "stderr" links</p> <p>At the end of the "stdout" log you will see the info of the dumpaed data products in the output and reference file of the reco stage. There are only the 2 mentioned data product that has different size out of 266 Reco data products.</p> <p>The reference files used by the DUNE CI tests are located at:</p> <p>/pnfs/dune/persistent/users/vito/ci_tests_inputfiles and are in the form AntiMuonCutEvents_LSU_v2_dune35t_Reference_<processing_stage>_default.root</p> <p>The output of the CI tests are created on Jenkins slaves used to run the CI build and they are not copied back, I collect only the outcome of the tests and some stats that are showed on the LAr CI web application: http://lar-ci-history.fnal.gov:8080/LarCI/app/view_builds/index</p> <p>Here clicking on the bullet in the "checkout" box for the desired CI build number you can get the list of the code revision for each software module used for that build.</p> <p>You can also run the CI tests on your machine. There are general information on the Lar CI wiki page (https://cdcvns.fnal.gov/redmine/projects/lar-ci/wiki/Test_Runner_Introduction) However for your specific use case you can follow the information below:</p> <p>To run the CI test on your machine you need to:</p> <ul style="list-style-type: none">o) checkout, build and setup the code you want to test;o) checkout lar-ci module:o) git clone http://cdcvns.fnal.gov/projects/lar_ci			

o) export PATH=`pwd`/lar_ci/bin:\$PATH
o) the previous is to add the lar_ci/bin directory to PATH

at this point you are almost ready to run the CI tests

o) You need a valid proxy with "dune" VO
voms-proxy-init -noregen -rfc -voms dune:/dune/Role=Analysis
o) choose a directory where to run the CI tests
o) run the command

```
test_runner -v --statistics quick_test_dunetpc
```

this will run in parallel the 5 dunetpc CI test in the quick_test_dunetpc test suite using the simulation workflow

In the chosen directory it will be created a directory for each of the 5 CI tests in the suite where you will have logs and output files.

Hope this can help, if you need more details, please, let me know.

Thanks,
Vito

On 12/08/16 08:04, David Adams wrote:

There might be some change for standard_reco_dune35tdata.fcl but the results for simulation should be identical. Reading more carefully, I see your results are for simulation and so no change is expected.

I did some renaming of the fcl files and this is the first tag with standard_reco_dune35tsim.fcl. What do you use as reference in the results you report here?

The mapping between old and new fcl file names is here:

<https://cdcv.s.fnal.gov/redmine/issues/12701#note-27>

Are you able to compare results for different names in your testing?

da

On 08/12/2016 08:25 AM, David Adams wrote:

Vito:

Could be my fault. I expected some change but thought the size would be about the same. I would like to have a look.

Can you tell me the following?:

1. How to access the output event data for both jobs.
2. How to run the the jobs on my machine.

How do the subsequent data products look?

Thanks.

da

On 08/11/2016 01:07 PM, Vito Di Benedetto wrote:

Hi Tom, Tingjun, David,

the CI build of the develop branch of LArSoft + exp codes show a data product size mismatch in reco stage of the dunetpc (FHiCL file standard_reco_dune35tsim.fcl).

The details of the data products affected is reported below:

< Reco | caldata | | art::Assns<raw::RawDigit,recob::Wire,void> | 306

Reco | caldata | | art::Assns<raw::RawDigit,recob::Wire,void> | 399

< Reco | caldata | | std::vector<recob::Wire> | 306

Reco | caldata | | std::vector<recob::Wire> | 399

Is this expected?

Could be this related to a commit from David Adams?

Commit links:

<https://cdcv.sfnal.gov/redmine/projects/dunetpc/repository/revisions/1de9b5e97b04f47e2b7fe9b7d7447ee9c2af0e2a>

<https://cdcv.sfnal.gov/redmine/projects/dunetpc/repository/revisions/8dc1183e4fb6ce25b498a4fe4b298dd6b8b7ba70>

Can I generate a new set of reference files for dunetpc, or you need more details to make sure why this is happening?

Thank you,
Vito

Related issues:

Blocks dunetpc - Feature #12701: New module and services for raw data prepara...

Closed

05/18/2016

History

#1 - 08/15/2016 08:55 AM - David Adams

- *Blocks Feature #12701: New module and services for raw data preparation added*

#2 - 08/15/2016 11:16 AM - David Adams

- *Status changed from Assigned to Closed*

I checked earlier (and reported at the last 35t reco meeting) that the Wire values for each channel and tick are the same to one part in 1.e6 for the old and new reco. Vito's report does not contradict this but indicates the new reco is writing more channels than the old. Looking into the code, I see that the new code writes an empty vector when a channel has no ROIs where the old presumably skipped the channel.

I have modified the new code so it no longer records Wires for channels with no ROIs and I now see in my test that old and new have the same number of channels in the wire collections. The change is in dunetpc 92e70e7629591dde72dc847eb981fcc32189a67f.

I close this report.

#3 - 08/15/2016 11:49 AM - David Adams

I have confirmed that the Wires in an event produced with the new code are now equivalent to the old, i.e. the signals in each channel/tick agree to one part in 1.e6.