

dunetpc - Task #25409

Changes in dataprep data model

01/18/2021 01:09 PM - David Adams

Status:	Work in progress	Start date:	01/18/2021
Priority:	Normal	Due date:	
Assignee:	David Adams	% Done:	0%
Category:		Estimated time:	0.00 hour
Target version:			
Description			
I am making some modifications to the dataprep data model (AdcChannelData, etc.) in advance of adding explicit support for wire planes (2D channel-tick) and 2D ROIs.			

History

#1 - 01/18/2021 01:18 PM - David Adams

The first change is to move some header files in DuneInterface. Most headers were in subdirectories but a few, notably AdcChannelData.h, were in the top level directory. These were moved last week. Code in dunetpc and protoduneana was updated for the new locations and changes committed Jan 14. I have not heard any problems with CI testing. Please let me know if anything has changed. The changes here should not have any effect.

#2 - 01/18/2021 01:29 PM - David Adams

AdcChannelData has many fields including seven that hold event-level data (run, event, trigger, ...) that is replicated for each channel. It has been modified so, instead of holding the data directly, each channel now holds a shared pointer to a DuneEventData object that holds the fields. Methods are provided to read the fields and to set the full event data.

This required modification of many tools and other actors that were accessing the event data directly. Those changes have been made locally and the tests run successfully. No changes were required in protoduneana.

I am committing the changes now. Please let me know if any problems turn up in the CI testing. Again, there should be no effect there.

#3 - 01/19/2021 08:09 AM - David Adams

Etienne reports CI testing is fine in [#25410](#). Release v09_13_00 will include the changes above.

#4 - 01/19/2021 12:10 PM - David Adams

More preparatory changes. Real2dData is moved to DuneInterface. DuneChannelInfo holding channel, FEMB ID and status is added. Root dictionary entries are added for both.

#5 - 01/19/2021 04:35 PM - Tingjun Yang

Hi David,

There is a failure in CI test:

<https://dbweb8.fnal.gov:8443/LarCI/app/ns:dune/storage/docs/2021/01/19/buildDUNE%23enl4Vqs.log>

```
388: input_line_9:9:10: fatal error: 'dune/DuneInterface/Data/DuneChannelInfo.h' file not found
```

```
389: #include "dune/DuneInterface/Data/DuneChannelInfo.h"
```

```
390:      ^_~::~::~::~::~::~::~::~::~::~::~::~::~::~::~::~::~::~
```

```
391: Error: rootcling: compilation failure
```

```
(/scratch/workspace/dune_ci/label_exp/SLF7/label_exp2/swarm/DUNE/build_slf7.x86_64/dunetpc/lib/libdune_DuneInterface_Data_dict660abb831a_dictUmbrella.h)
```

Tingjun

#6 - 01/20/2021 08:59 AM - David Adams

Oops. Thanks for the report. I added the missing file and pushed. Pls let me know if or if not the problem is resolved.

#7 - 01/21/2021 09:28 AM - David Adams

AdcChannelData has been modified to hold channel data through DuneChannelInfo. Code in dunetpc has been updated accordingly. I am about to commit these changes.

#8 - 01/21/2021 03:10 PM - David Adams

I have added Tpc2dRoi which holds the channel-tick data for an ROI in a TPC plane. That ROI can be the full plane.

The dataprep tool interface will be extended to support this new data class in addition to AdcChannelData and AdcChannelDataMap.

Design is still underway, but most likely I will also add another data class TpcData or TpcPlaneData which can hold zero or one (or more) objects of the types AdcChannelMap and Tpc2dRoi. This new container will be added to the dataprep tool interface and that method will be the one called in the dataprep tool loop.

#9 - 01/25/2021 02:11 PM - David Adams

I have added TpcData which holds both a map of AdcChannelData and a vector of Tpc2dRoi. It also holds other named objects of the same type so data can be distributed over a tree structure. This has just been committed.

The next step is to modify the dataprep tool interface to accept TpcData objects with a base implementation that passes the ADC channel map to that method.

#10 - 01/26/2021 08:07 AM - David Adams

I have added interface TpcDataTool which inherits the interface AdchannelTool. The new interface adds methods to update and view TpcData objects with default implementations that update and view the ADC channel map held in that data object.

Next I will use the new class as base for the dataprep tools and then modify dataprep to loop over updateTpcData instead of updateMap.

#11 - 01/27/2021 11:51 AM - David Adams

I copied the dataprep service ToolBasedRawDigitPrepService to a new service TpcToolBasedRawDigitPrepService which loops over TpcDataTool tools instead of AdcChannelTool tools.

I also restored the assignment of time stamps to AdcChannelData objects which was somehow lost in earlier changes here.

The CI tests should continue to run as they always have.

#12 - 01/28/2021 01:15 PM - David Adams

The names of the tool name lists in ToolBasedRawDigitPrepService and TpcToolBasedRawDigitPrepService have been changed from AdcChannelToolNames and TpcDataToolNames to both be ToolNames. This was done because the service is typically configured as its base class. The fcl files in dunetpc have been updated accordingly.

There are two files in protoduneana that are affected: pdwire.fcl and runProtoDUNEDPAnalCosmicTree.fcl. I added the new name field for those but also left the old name with a comment that it is needed for dunetpc v09_15_00 and earlier. Those can presumably be deleted in a week or two.

All data prep tools now inherit from TpcDataTool instead of AdcChannelTool. Because the latter inherits from the former, they should continue to work in the old service and this appears to be the case.

These changes have just been committed to dunetpc. Note this is after the v09_15_00 release.

There is one more change for this task: switch from ToolBasedRawDigitPrepService to a new service TpcToolBasedRawDigitPrepService if the dunetpc fcl. I will do that tomorrow if no problems are reported with the changes thus far.

#13 - 01/29/2021 10:11 AM - David Adams

Etienne reports a change in the CI testing:

```
I see a quite surprising change after your pushes in number of hitpdune in ci_datareco_regression_test_protoDUNEsp :
```

```
DecoderandReco | hitpdune | | std::vector<recob::Hit> | 46479  
DecoderandReco | hitpdune | | std::vector<recob::Hit> | 172023
```

I have added him to this ticket. Previous omission was inadvertent.

I am running some checks on my own. I see the reco fcl has changed since I last ran such tests. I suppose I should now use protoDUNE_refactored_reco.fcl for protoDUNE data. Correct?

I ran processing through dataprep with v09_15_00 and with the latest changes and the displays look the same. I attach a couple plots.

#14 - 01/29/2021 10:40 AM - David Adams

More from Etienne:

```
So the push from which the changes appeared is this one (023347d6)  
and the CI tests is this one: href=https://dbweb8.fnal.gov:8443/LarCI/app/ns:dune/build_detail/test_details?build_id=dune_ci/10809&platform=Linux%203.10.0-1160.11.1.el7.x86_64&buildtype=slf7%20e19:prof&phase=ci_tests&test=ci_datareco_regression_test_protoDUNEsp
```

The change he references is the one described in #13.

I ran my test with v09_14_00, v09_15_00 and with the latest change. The event displays all look the same but do show a lot of coherent noise. I may have the wrong top-level fcl.

#15 - 01/29/2021 10:50 AM - Etienne Chardonnet

Hi, the CI tests for push 27bbf1aa are ok (the 09_15_00 release) and the error shows up for CI test related to next push (023347d6)

#16 - 01/29/2021 11:56 AM - David Adams

Thank you Etienne. Good to know all is OK in v09_15_00.

Poking into the CI files, I conclude the top-level reco I should use for protoDUNE data reco is protoDUNE_SP_keepup_decoder_reco.fcl. The one I listed earlier is presumably for MC. With this new top-level fcl, the results for v09_15_00 look much better with coherent noise now suppressed. And I do see problems in the current dunetpc tag. I am investigating.

#17 - 01/29/2021 12:51 PM - David Adams

I found and fixed the problem. AdcChannelToolNames was not updated to ToolNames in a couple files.

Now my test results match those for v09_15_00.

Etienne, thank you very much for reporting the CI problem.

#18 - 02/02/2021 02:11 PM - David Adams

I have made a change in the data model. TpcData now holds a vector of shared pointers to AdcChannelData maps so that the ADC data associated with a TpcData object (and hence with its 2D ROIs) can be shared with other objects. E.g. we can have separate TpcData for each wire plane and then have a TpcData object which shares their AdcChannelData maps.

The default processing of TpcData in TpcDataTool has been modified to process its AdcChannelData maps in turn.

Again, there should be no change in CI tests.

#19 - 02/03/2021 11:33 AM - David Adams

I have added a global tick offset to DuneEventInfo and modified the dataprep modules to fill this field with triggerClock/25 so tools do not have make any assumptions on how to convert trigger clocks to tick offset.

I also added AdcChannelData::tickOffset() which adds the tick offset (tick0) to the above global offset, i.e returns the global tick index for the first sample in AdcChannelData.

The change is committed. Again, there should no change in CI.

#20 - 02/05/2021 08:33 AM - Tingjun Yang

Hi David,

When compiling protoduneana against dunetpc v09_15_00, I got the following error:

```
/data/tjyang/dune/larsoft_em/srcs/protoduneana/protoduneana/DataDump/RawWaveformDump_module.cc:33:10: fatal error:
dune/DuneInterface/Service/SimChannelExtractService.h: No such file or directory
#include "dune/DuneInterface/Service/SimChannelExtractService.h"
^~~~~~
compilation terminated.
```

Is there a solution to this?

Thanks,
Tingjun

#21 - 02/05/2021 08:40 AM - David Adams

Tingjun:

I don't think I have made any changes in DuneInterface/Service and it appears the header file is present. I have included protoduneana in the above builds and do not see any problems.

Do you have up-to-date pulls for dataprep and protoduneana?

Never mind that I see you are using v09_15_00. I still can't remember any changes in service headers. Can you move to the head (or more recent) dunetpc? Or you can try commenting out RawWaveformDump in cmake. If you need to work in v09_15_00, let me know and I will try to reproduce your problem.

da

#22 - 02/05/2021 09:11 AM - Tingjun Yang

David Adams wrote:

Tingjun:

I don't think I have made any changes in DuneInterface/Service and it appears the header file is present. I have included protoduneana in the above builds and do not see any problems.

Do you have up-to-date pulls for dataprep and protoduneana?

Never mind that I see you are using v09_15_00. I still can't remember any changes in service headers. Can you move to the head (or more recent) dunetpc? Or you can try commenting out RawWaveformDump in cmake. If you need to work in v09_15_00, let me know and I will try to reproduce your problem.

da

Hi David,

I am quite confused:

<https://cdcvns.fnal.gov/redmine/projects/dunetpc/repository/revisions/develop/show/dune/DuneInterface>

It seems the directory Service is not included in the CMakeLists.txt. Also there is no CMakeLists.txt in the Service directory so I am not surprised the header files are not installed.

I can see it's not a problem if you build dunetpc and protoduneana together, but that defeats the purpose of creating protoduneana.

Thanks,
Tingjun

#23 - 02/05/2021 10:17 AM - David Adams

I don't think the problem Tingjun reports is relevant to this task and I have opened a new report: [#25478](#).

#24 - 02/10/2021 09:09 AM - David Adams

Above problems are resolved and dunetpc v09_16_00 has been released.

I am adding DFTs for the 2d ROIs to TpcData. They are type

```
FftwDouble2dDftData = FftwReal2dDftData<double>
```

The header for this class is moved from DuneCommon to DuneInterface/Data.

#25 - 02/10/2021 11:00 AM - David Adams

I have moved the following headers from DuneCommon:

```
FftwReal2dDftData.h    -> ../DuneInterface/Data/FftwReal2dDftData.h  
Real2dDftData.h       -> ../DuneInterface/Data/Real2dDftData.h  
RealDftNormalization.h -> ../DuneInterface/Data/RealDftNormalization.h
```

so DuneInterface has no dependence on DuneCommon.

#26 - 02/10/2021 01:35 PM - David Adams

<Text removed>

#27 - 02/22/2021 08:25 AM - David Adams

I have added dataprep TPC tool Roi2dToAdc that copies the TpcData samples from the 2D ROIs to ADC channels and sets the signal flags. The samples, signal and DFTs for all channels are cleared before making the copies.