

## dunetpc - Task #20303

### Switch protodune to tool-based dataprep

07/10/2018 02:31 PM - David Adams

<b>Status:</b>	Closed	<b>Start date:</b>	07/10/2018
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>		<b>% Done:</b>	0%
<b>Category:</b>		<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>			
<b>Description</b>			
As shown in my talk at the last collaboration meeting, a simple configuration of the tool-based dataprep gives the same results as the default service-based configuration for protoDUNE simulation.			
We should make the tool-based dataprep the default.			

#### History

##### #1 - 07/10/2018 02:44 PM - David Adams

- Status changed from New to Feedback

I modified dune/DataPrep/Tool/fcl/dataprep\_dune.fcl to rename the block adcprep\_sim to adcprep\_with\_services\_sim. This is the default block for configuring service-based dataprep. I updated dune/Utilities/services\_dune.fcl to use the new name and confirmed protodune dataprep still runs.

I then added the blocks adcprep\_with\_tools\_sim and adcprep\_with\_tools\_data to the former to configure tool-based dataprep and updated protodune\_reco\_services.RawDigitPrepService in the latter to point to the first block. I confirmed that protoDUNE simulation reco now uses the tool-based data and, for one event by eye, gives the same results as the old reco.

The change is committed to dunetpc.

Tingjun, could you check that this has not destroyed downstream protodune reco?

##### #2 - 07/10/2018 08:51 PM - Tingjun Yang

Hi David,

According to the CI test:

[http://dbweb5.fnal.gov:8080/LarCI/app/ns:dune/storage/docs/2018/07/10/stdout\\_bVQNQWq.log](http://dbweb5.fnal.gov:8080/LarCI/app/ns:dune/storage/docs/2018/07/10/stdout_bVQNQWq.log)

there do seem to be some changes in data product sizes:

```
764: Check for differences in the size of data products
765: difference(s)
766:
767: 38c38
768: < Reco | caldata | | art::Assns<raw::RawDigit,recob::Wire,void> | 715
769: ---
770: > Reco | caldata | | art::Assns<raw::RawDigit,recob::Wire,void> | 896
771: 57c57
772: < Reco | pmtrack | | art::Assns<recob::Track,recob::Hit,recob::TrackHitMeta> | 1391
773: ---
774: > Reco | pmtrack | | art::Assns<recob::Track,recob::Hit,recob::TrackHitMeta> | 1409
775: 59c59
776: < Reco | pmtrack | | art::Assns<recob::Track,recob::Hit,void> | 1391
777: ---
778: > Reco | pmtrack | | art::Assns<recob::Track,recob::Hit,void> | 1409
779: 79,80c79,80
780: < Reco | pmtrack | | art::Assns<recob::Track,recob::SpacePoint,void> | 1391
781: < Reco | caldata | | std::vector<recob::Wire> | 715
782: ---
783: > Reco | pmtrack | | art::Assns<recob::Track,recob::SpacePoint,void> | 1408
784: > Reco | caldata | | std::vector<recob::Wire> | 896
785: 88c88
786: < Reco | pmtrack | | art::Assns<recob::SpacePoint,recob::Hit,void> | 1391
787: ---
788: > Reco | pmtrack | | art::Assns<recob::SpacePoint,recob::Hit,void> | 1409
789: 98c98
790: < Reco | pmtrack | | std::vector<recob::SpacePoint> | 1391
```

791: ---

792: > Reco | pmtrack | | std::vector<recob::SpacePoint> | 1408

Are they expected?

Tingjun

**#3 - 07/11/2018 07:39 AM - David Adams**

Tingjun:

We do expect some difference as the new reco uses a different algorithm to find ROIs. This is discussed in the collaboration meeting talk. The new reco uses a simple threshold algorithm which (as configured) has a higher signal efficiency but lets through more noise. Do you have any downstream metrics such as track-finding efficiency and purity?

Thanks.

da

**#4 - 07/17/2018 11:08 AM - Christoph Alt**

Hi,

Is there any update concerning the changes in the data product sizes?

Christoph

**#5 - 08/03/2018 04:17 PM - David Adams**

Tingjun has accepted the change for protodune.

**#6 - 08/03/2018 04:18 PM - David Adams**

- *Status changed from Feedback to Closed*