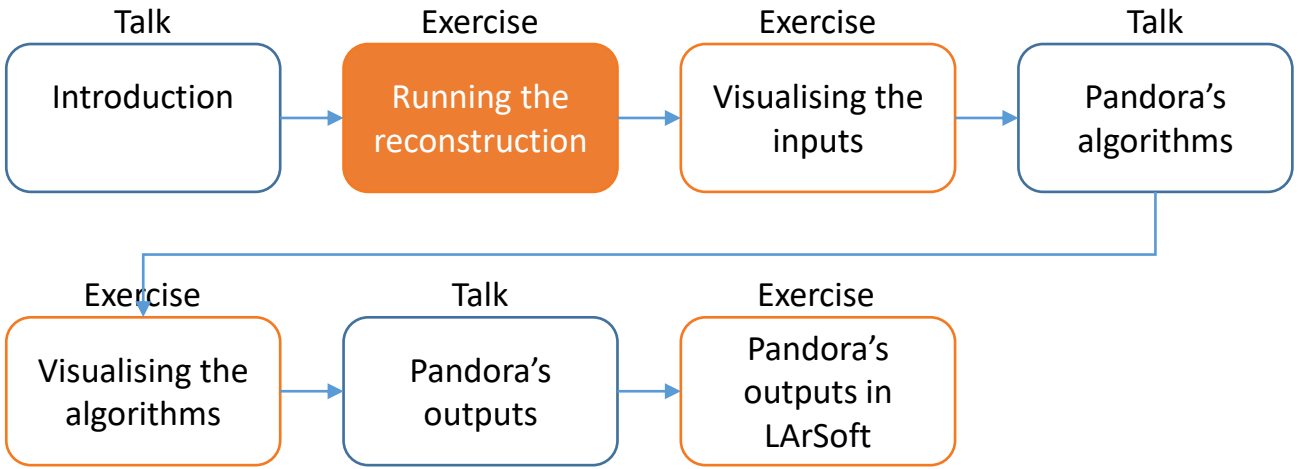
Running the reconstruction

Dom Brailsford for the Pandora team

08/09/2022 UK-Latin America LArSoft Workshop

WARWICK

Reconstruction session



Credit: These slides are based on previous LArSoft workshop slides by Lorena Escudero and Andrew Smith

Key references:
[Pandora ProtoDUNE paper]
[Pandora MicroBooNE paper]

Goals

WARWICK

- This session scheduled for 1 hour
- Main goal 1 - Find and get to grips with the DUNEFD horizontal drift reconstruction FHiCL files
  - Find the standard_reco_dune10kt configuration files
  - Look at the different reconstruction steps that we will run
  - Understand what each of them do
- Main goal 2 - Run the reconstruction
  - Run the reconstruction on the files we simulated yesterday
  - This includes running Pandora
  - Dump out the new output products to confirm we produced what we wanted

Before we get started…

WARWICK

- Later today we'll be running the event display
- Follow the steps from yesterday to get everything set up. To check, make sure MRB_TOP points to where you expect. Let us know if you have any issues here

Please replace machine as required!

```
$ ssh machine
$ echo $MRB_TOP # This should print the path to your development area
```

Main Goal 1

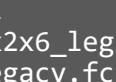
WARWICK

Understanding the DUNE reconstruction FHiCL files

DUNE reconstruction FHiCL file

- Open standard_reco_dune10kt_1x2x6.fcl, we'll use this to run the reconstruction

```
$ less $MRB_SOURCE/dunesw/fcl/dunefd/reco/standard_reco_dune10kt_1x2x6.fcl
$ less $MRB_SOURCE/dunesw/fcl/dunefd/reco/legacy/standard_reco_dune10kt_1x2x6_legacy.fcl
$ less $MRB_SOURCE/dunesw/fcl/dunefd/reco/legacy/standard_reco_dune10kt_legacy.fcl
```



- Find the trigger_paths: [...].
  - Q: which producers are we going to run?
  - A: It's the ones in the reco path:

```
reco: [
   rns,
   ophit, opflash,
   caldata,
   fasthit,
   gaushit,
   hitfd,
   linecluster,
   trajcluster,
   trkshowersplit
   ...
```

```
   ...
   pandora,
   pandoraTrack,
   pandoraShower,
   pandoracalo,
   pandorapid,
   ...
]
```

First time using less? Use the ↑ / ↓ arrow keys to navigate the file, and press q to quit

File looks empty? Make sure you setup your working area again, otherwise $MRB_SOURCE won't point to anywhere!

- In the last session, we were introduced to pandora, but there are many other steps in the reconstruction chain too!
- Next is a single-slide overview of these steps - not nearly enough to do them justice - but today we will mainly be focusing on pandora

DUNE reconstruction chain on one slide

WARWICK

rns

"Random number saver" saves the state of art's random number generator

caldata, gaushit, …

These modules process the wire signals (e.g. to remove noise), and then find peaks to make hits

linecluster, …

Runs a clustering algorithm on the hits, used later by PMA

pandora

Runs Pandora's pattern recognition itself.

Produces reconstructed particles from the input hits which are used by downstream modules and eventual physics analyses

Confusingly, some other modules contain the word "pandora" but this just refers to the fact that their inputs come from Pandora

pm*, …

PMA (Projection Matching Algorithm) is an alternative method to Pandora for reconstructing tracks

emshower, blurredcluster, …

Advanced electromagnetic shower reconstruction

pandoraTrack, pandoraShower

Simple track and shower fitting modules using particles created by Pandora particle as input

pandoracalo, pandorapid

Reconstructs calorimetric information (dE/dx) from Pandora tracks, and runs particle identification (PID) to distinguish between reconstructed particle types

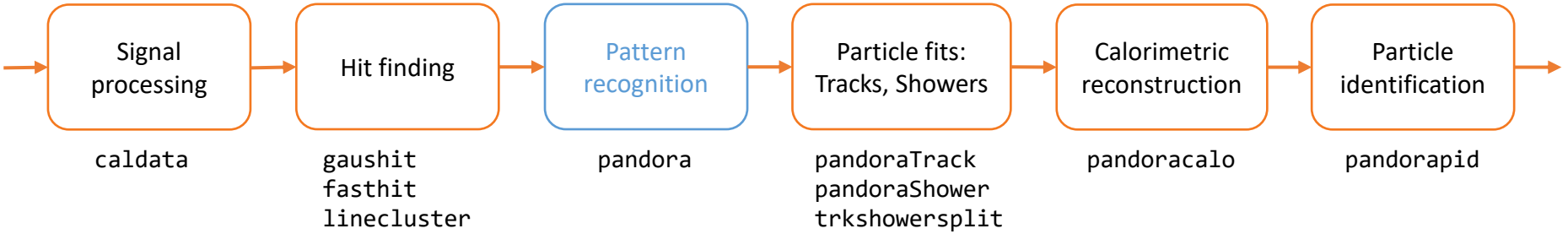
```
reco: [
   rns,
   ophit, opflash,
   caldata,
   fasthit,
   gaushit,
   hitfd,
   linecluster,
   trajcluster,
   trkshowersplit
   ...
```

```
 ...
   pandora,
   pandoraTrack,
   pandoraShower,
   pandoracalo,
   pandorapid,
   ...
]
```

A note on other experiments

WARWICK

- Remember here we are looking at the configuration for **DUNE**
- Each experiment has its own unique needs, so expect to see some differences in the reconstruction chain if you work on MicroBooNE, ProtoDUNE, SBND, etc.
- As far as Pandora is concerned, we can generalise the reconstruction chain to the following steps:



- Next, let's see how pandora is configured for DUNE

Pandora's configuration

- fhicl-dump follows all #includes to get the bottom-line configuration. We can pipe (|) its output to less and search for a producer to learn more. Search for pandora by typing:

```
$ fhicl-dump standard_reco_dune10kt_1x2x6.fcl | less -p "pandora:“
```

less's -p option allows us to jump straight to the part of the file we are interested in

```
pandora: {
         ConfigFile: "PandoraSettings_Master_SBND.xml"
         EnableLineGaps: true
         EnableMCParticles: false
         EnableProduction: true
         GeantModuleLabel: "largeant"
         HitFinderModuleLabel: "linecluster"
         PrintOverallRecoStatus: false
         ShouldPerformSliceId: false
         ShouldRunAllHitsCosmicReco: false
         ShouldRunCosmicHitRemoval: false
         ShouldRunCosmicRecoOption: false
         ShouldRunNeutrinoRecoOption: true
         ShouldRunSlicing: false
         ShouldRunStitching: false
         UseGlobalCoordinates: true
         UseHitWidths: true
         module_type: "StandardPandora"
      }
```

The settings file that contains the list of algorithms that Pandora will run

The producer module that created the hits that we are going to feed into Pandora

The steering parameters that tell Pandora which of it's high level reconstruction steps it should execute

The type of the LArSoft module to use

Main Goal 2

WARWICK

Running the reconstruction

Running the reconstruction

- We are now poised to run the reconstruction! Make a directory to work in, and run it:

```
$ mkdir -p $MRB_TOP/reco/work
$ cd $MRB_TOP/reco/work
$ lar -c standard_reco_dune10kt_1x2x6.fcl -n -1 -s /path/to/my/detsim/file.root -o reco_1mu1p.root
```

This step can take some time, so please be patient!

The -n -1 option means run over all events in the input file

Can also run on pre-made gen+g4+detsim files in
/home/share/september2022/simulation/detsim_1mu1p.root

Full event	30.33	36.0671	41.6728	36.4245	3.52474	10
source:RootInput(read)	0.000763427	0.00177477	0.00313986	0.00184294	0.00080113	10
reco:rns:RandomNumberSaver	4.2575e-05	8.41849e-05	0.000372039	5.36485e-05	9.62063e-05	10
reco:ophit:OpHitFinder	0.449261	0.622272	1.03332	0.501103	0.206869	10
...						
reco:linecluster:LineCluster	0.253137	0.404216	0.700568	0.370396	0.111067	10
...						
reco:pandora:StandardPandora	0.260475	0.457744	0.774284	0.390011	0.1511	10
reco:pandoraTrack:LArPandoraTrackCreation	0.0084431	0.0110021	0.0141811	0.0107761	0.0014914	10
reco:pandoraShower:LArPandoraModularShowerCreation	0.00101441	0.00149197	0.00336328	0.00132518	0.000632474	10
reco:pandoracalo:Calorimetry	0.00770421	0.013151	0.0182823	0.0132139	0.00243814	10
reco:pandorapid:Chi2ParticleID	0.000236185	0.000366934	0.0010507	0.000289568	0.000230409	10
...						
end_path:out1:RootOutput	5.751e-06	1.01145e-05	2.4017e-05	8.646e-06	5.07523e-06	10
end_path:out1:RootOutput(write)	0.925388	0.949373	0.969371	0.949746	0.0134437	10

We can check to see that everything we expected has been executed, and see how long each took

So… what's new?

WARWICK

- Run eventdump.fcl to see all of the new collections we just made

```
$ lar -c eventdump.fcl -s reco_1mu1p.root -n 1
```

PROCESS NAME	MODULE LABEL.....	PRODUCT INSTANCE NAME	DATA PRODUCT TYPE.....	
SingleGen...	TriggerResults.....	.....	art::TriggerResults.....	
SingleGen...	generator.....	.....	std::vector<simb::MCTruth>.....	1
SingleGen...	rns.....	.....	std::vector<art::RNGsnapshot>.....	1
G4.....	largeant.....	.....	std::vector<sim::OpDetBacktrackerRecord>.....	..209
G4.....	rns.....	.....	std::vector<art::RNGsnapshot>.....	2
G4.....	TriggerResults.....	.....	art::TriggerResults.....	....-
G4.....	largeant.....	.....	std::vector<sim::AuxDetSimChannel>.....	0
G4.....	largeant.....	.....	std::vector<simb::MCParticle>.....	...73
G4.....	largeant.....	.....	std::vector<sim::SimChannel>.....	.2038
G4.....	largeant.....	.....	std::vector<sim::SimPhotonsLite>.....	..209
DetSim.....	opdigi.....	.....	std::vector<raw::OpDetWaveform>.....	..375
DetSim.....	rns.....	.....	std::vector<art::RNGsnapshot>.....	2
DetSim.....	opdigi.....	.....	std::vector<raw::OpDetDivRec>.....	..904
DetSim.....	daq.....	.....	std::vector<raw::RawDigit>.....	.4785
DetSim.....	TriggerResults.....	.....	art::TriggerResults.....	....-
Reco.....	pandoraShower.....	.....	std::vector<recob::PCAxis>.....	
Reco.....	pandoraShower.....	.....	art::Assns<recob::Shower,recob::Track,void>.....	
Reco.....	pandora.....	.....	std::vector<recob::Vertex,void>.....	

These are the existing data products from previous steps

These are the new data products that we have just produced

WARWICK

Got spare time?

Try starting the next tutorial – running the event display

WARWICK

Additional information

Configuring Pandora steps

(For reference)

WARWICK

- Pandora's full reconstruction chain is designed to handle neutrino interactions in dense cosmic environments. As you will hear later, there are two main algorithm chains optimised for cosmic rays, and neutrinos respectively
- For SBND, an experiment that will have neutrinos and cosmics - we normally want to run all of the steps
- For cosmic events, we only need to run the cosmic algorithm chain. For neutrino events, we only need to run the neutrino algorithm chain. We can configure Pandora to run one, many or all of the steps in its full reconstruction chain by modifying the FHiCL steering parameters
- Make a new directory to work in for this session, and add a new FHiCL file with the following lines, then save and close the file:

```
$ mkdir -p $MRB_TOP/reco/config
$ cd $MRB_TOP/reco/config          # Put your new .fcl file here
$ vim my_reco.fcl
```

Please use your favourite text editor, here we use vim. If you accidentally opened vim and want to close it type Esc, :qa, Return ↵

```
#include "standard_reco_dune10kt_1x2x6.fcl"

physics.producers.pandora.ShouldRunAllHitsCosmicReco: false
physics.producers.pandora.ShouldRunStitching: false
physics.producers.pandora.ShouldRunCosmicHitRemoval: false
physics.producers.pandora.ShouldRunSlicing: false
physics.producers.pandora.ShouldRunCosmicRecoOption: false
physics.producers.pandora.ShouldRunNeutrinoRecoOption: true
physics.producers.pandora.ShouldPerformSliceId: false
```

Include the standard configuration

Example:
Only run the neutrino algorithm chain

Pointing to a new configuration (For reference)

WARWICK

- We want to make sure that LArSoft will know where to look for our new FHiCL file, to do this we add it to the FHICL_FILE_PATH environment variable. Start by printing it to the terminal:

```
$ echo $FHICL_FILE_PATH
```

- You will see many, many directories, all separated by a ‘:’. To add our reco/config folder to this list run the following command:

```
$ export FHICL_FILE_PATH=$MRB_TOP/reco/config:$FHICL_FILE_PATH
```

- Echo the FHICL_FILE_PATH again to check that everything worked (it should be the first in the list)
- Now run fhicl-dump again to make sure our new configuration file is set up as we want

```
$ fhicl-dump my_reco.fcl | less -p "pandora:"
```