LArSoft Simulation of the DUNE 10kt Dual-Phase TPC

Vyacheslav Galymov

IPN Lyon

LArSoft Coordination Meeting

02.02.2016

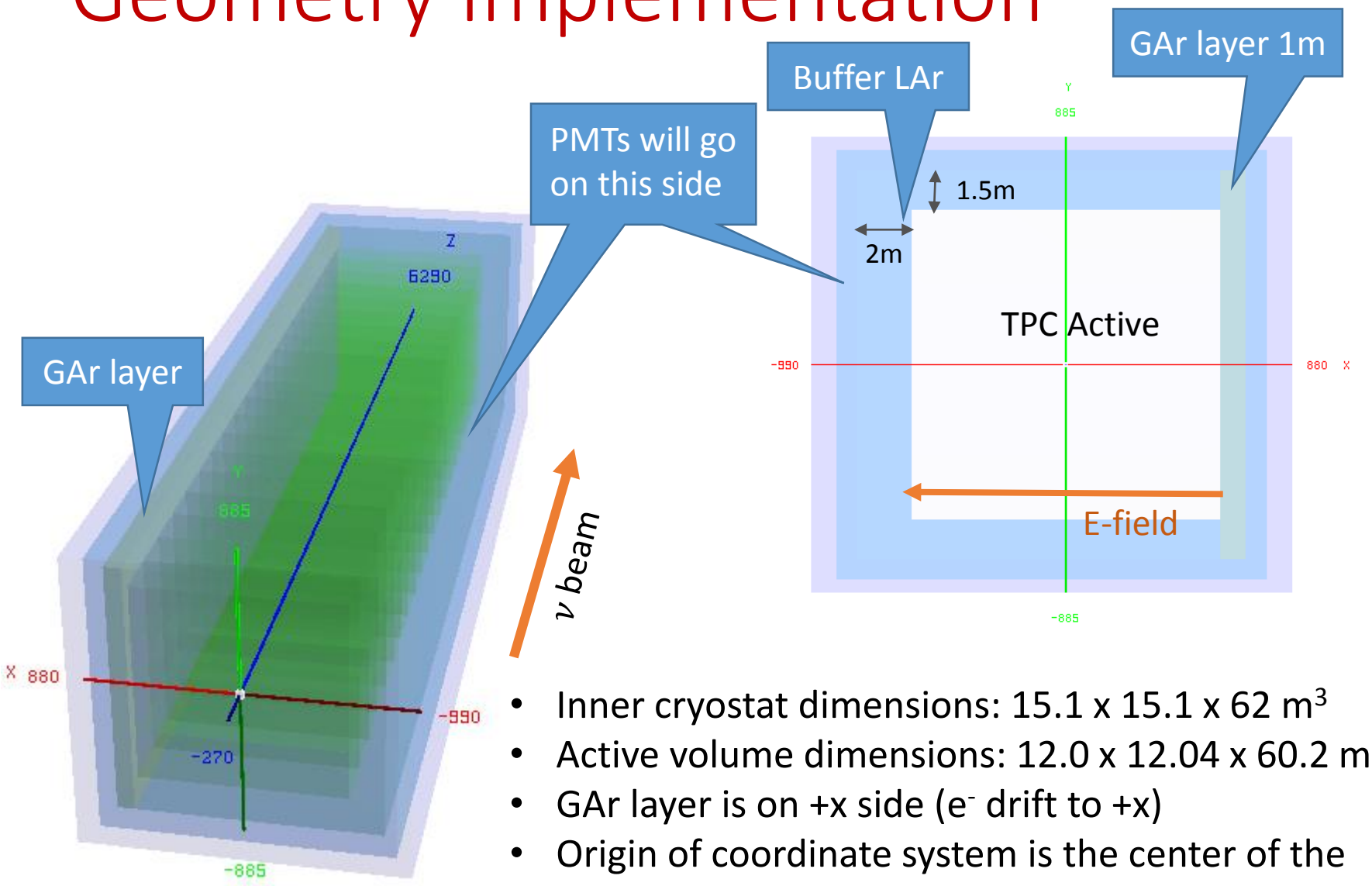
Introduction

Currently the framework can handle only electric field direction along X axis (in SP geometry this is horizontal axis)

To follow single-phase DUNE geometry axis convection this means that field in dual-phase is also along horizontal direction (not vertical!)

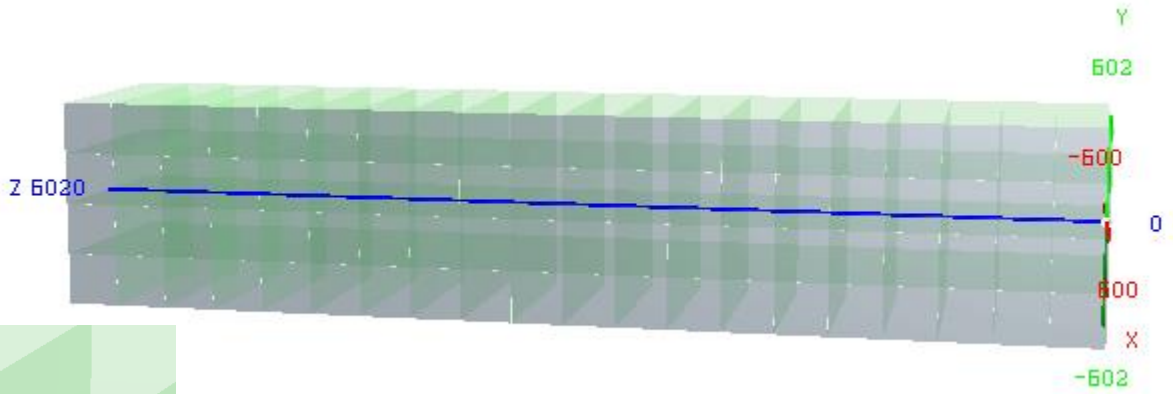
- This is the solution adopted to proceed with DP implementation, which is part of the DUNE FD TF deliverables

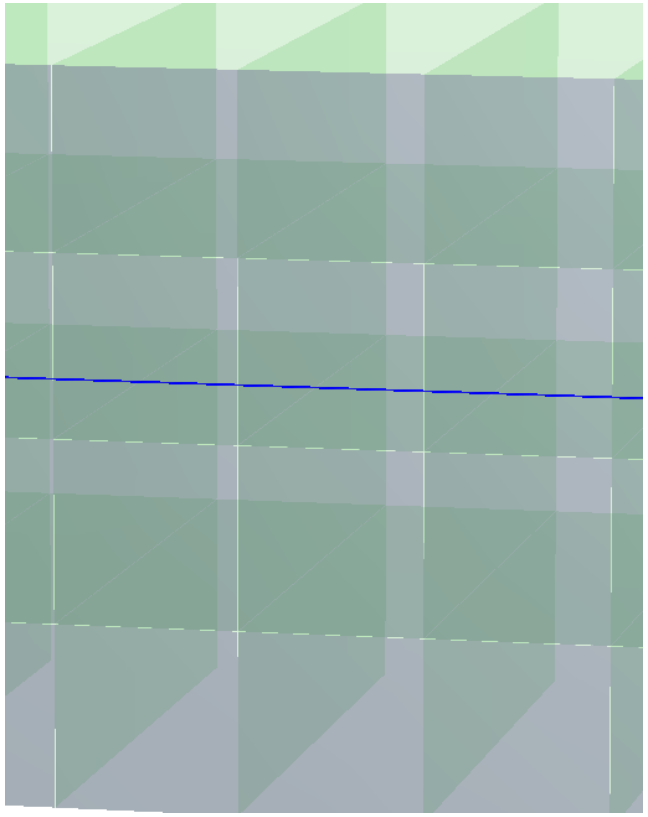
Geometry Implementation



- Inner cryostat dimensions: 15.1 x 15.1 x 62 m^3
- Active volume dimensions: 12.0 x 12.04 x 60.2 m^3
- GAr layer is on +x side (e^- drift to +x)
- Origin of coordinate system is the center of the TPC active volume at the upstream end

Charge readout modules (CRM)





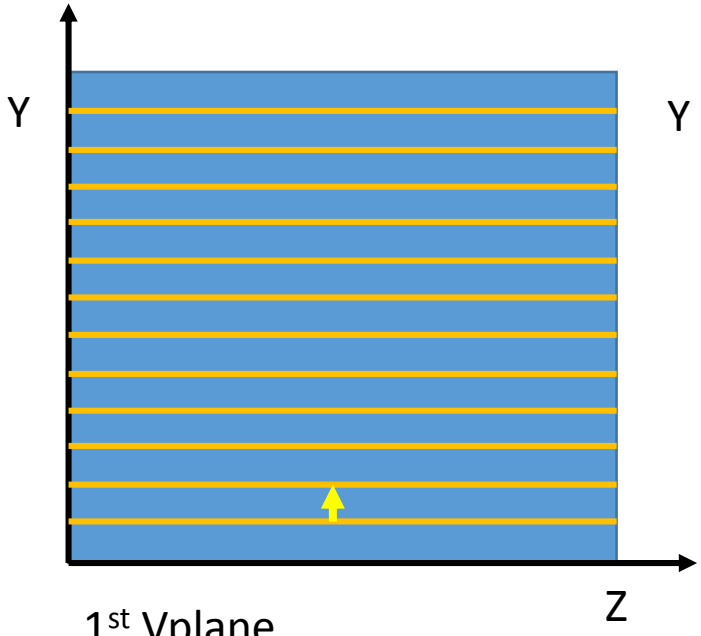
- Active volume consists of copy of identical rectangular prisms to be read out by 960 ch in “Y” and 960 ch in Z collection views
- The dimensions of each module are 3x3 m^2
- There is a dead space between each module of 1 cm associated with a border size of 0.5 cm for each 3x3 CRM
  - This could be modified easily via perl script which generates GDML geometry file

“Wire” planes: representation of collection anode

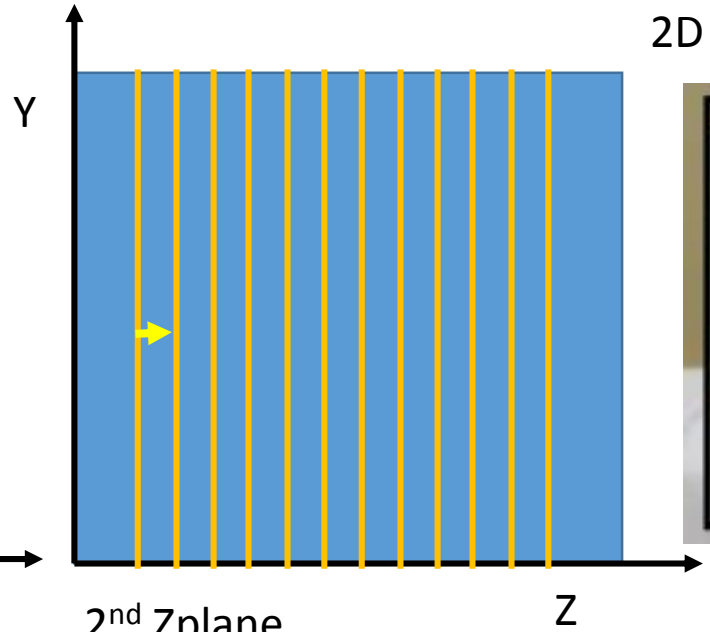
To accomplish charge readout in larsoft need to define wire plane volumes with “wires”

For DP we have two views ➔ two wire planes

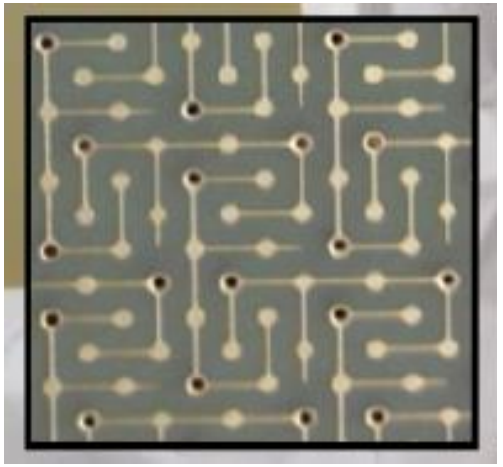
The actual readout is done with multilayered PCB with 2D pattern of strips



1st Vplane
(assigned geo::kU in larsoft)



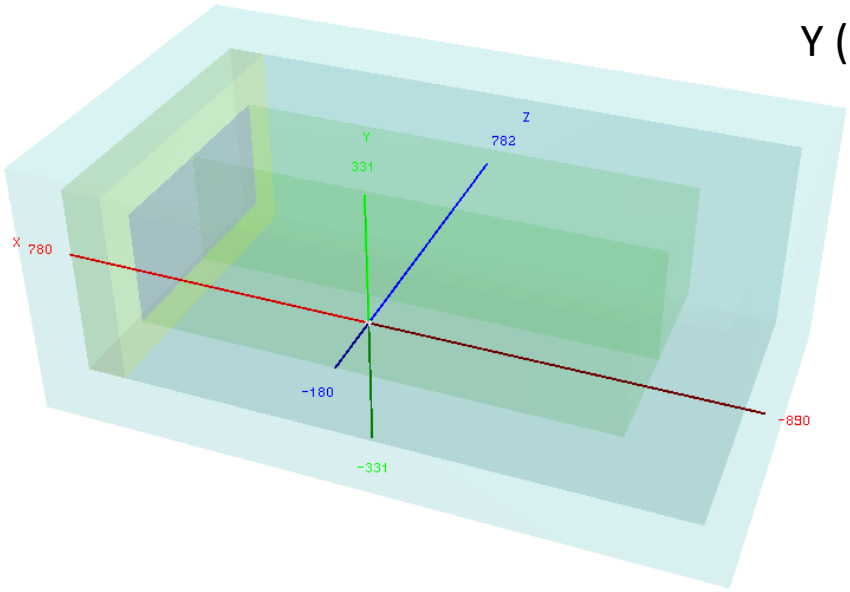
2nd Zplane
(assigned geo::kV in larsoft)



The arrow indicates direction of increasing channel count (sorting)

“Workspace” geometry

For quick testing useful to have small version of the geometry
→ Small number of channels
DP workspace geometry has been generated with just two CRM modules



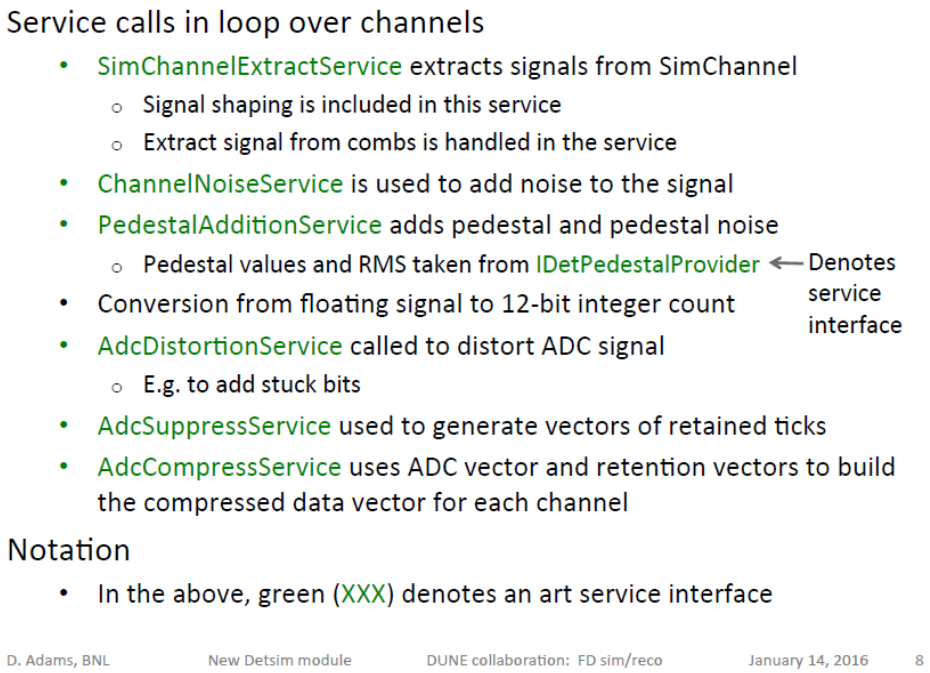
Max drift is 12 m, but there are only 2 CRMs in Y (now changed to Z) == only 3840 channels

Status of geometry

- Geometry implementation is completed (with horizontal drift)
  - Missing light readout: optical detectors
- The gdmal files & geometry scripts are available from dunetpc version v04_33_00

DetSim updates

- New modular structure was presented at the DUNE GM in Texas by David Adams



Service calls in loop over channels

- SimChannelExtractService extracts signals from SimChannel
  - Signal shaping is included in this service
  - Extract signal from combs is handled in the service
- ChannelNoiseService is used to add noise to the signal
- PedestalAdditionService adds pedestal and pedestal noise
  - Pedestal values and RMS taken from IDetPedestalProvider ← Denotes service interface
- Conversion from floating signal to 12-bit integer count
- AdcDistortionService called to distort ADC signal
  - E.g. to add stuck bits
- AdcSuppressService used to generate vectors of retained ticks
- AdcCompressService uses ADC vector and retention vectors to build the compressed data vector for each channel

Notation

- In the above, green (XXX) denotes an art service interface

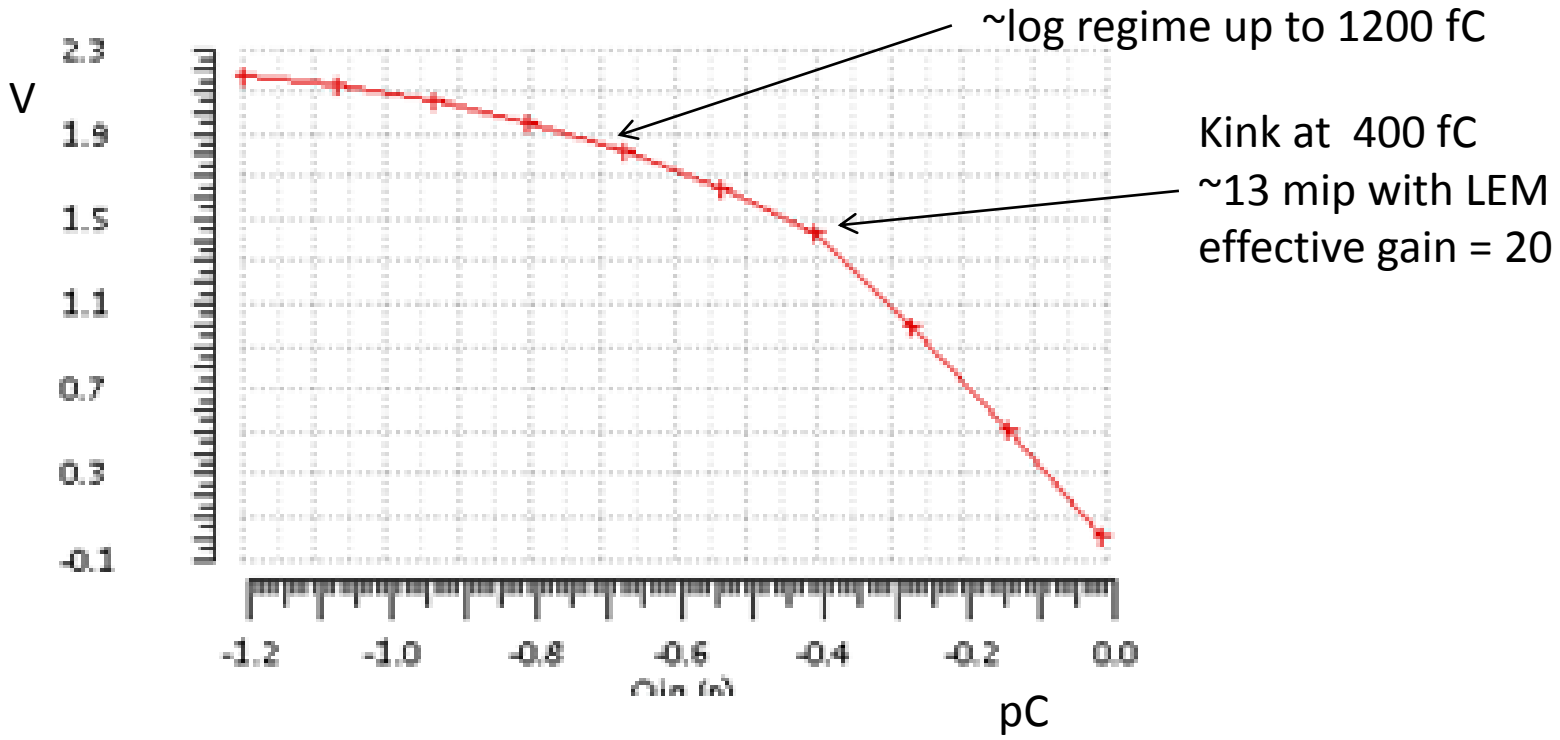
D. Adams, BNL New Detsim module DUNE collaboration: FD sim/reco January 14, 2016 8

- See dune/DetSim/Module/SimWireDUNE_module.cc
- Integrate DP DetSim with this updated structure

Dual-phase cold electronics

To increase dynamic range of the front-end electronics up-to 1200 fC the cryogenic amplifier have a double-slope feature:

- High gain up-to 400 fC (~13 mip with LEM gain of 10 per collection view)
- Smaller slope for high energy depositions



This is a complication for the convolution with FFT (and even more so for deconvolution) that has to be studied

Dual-phase response

Normalization:

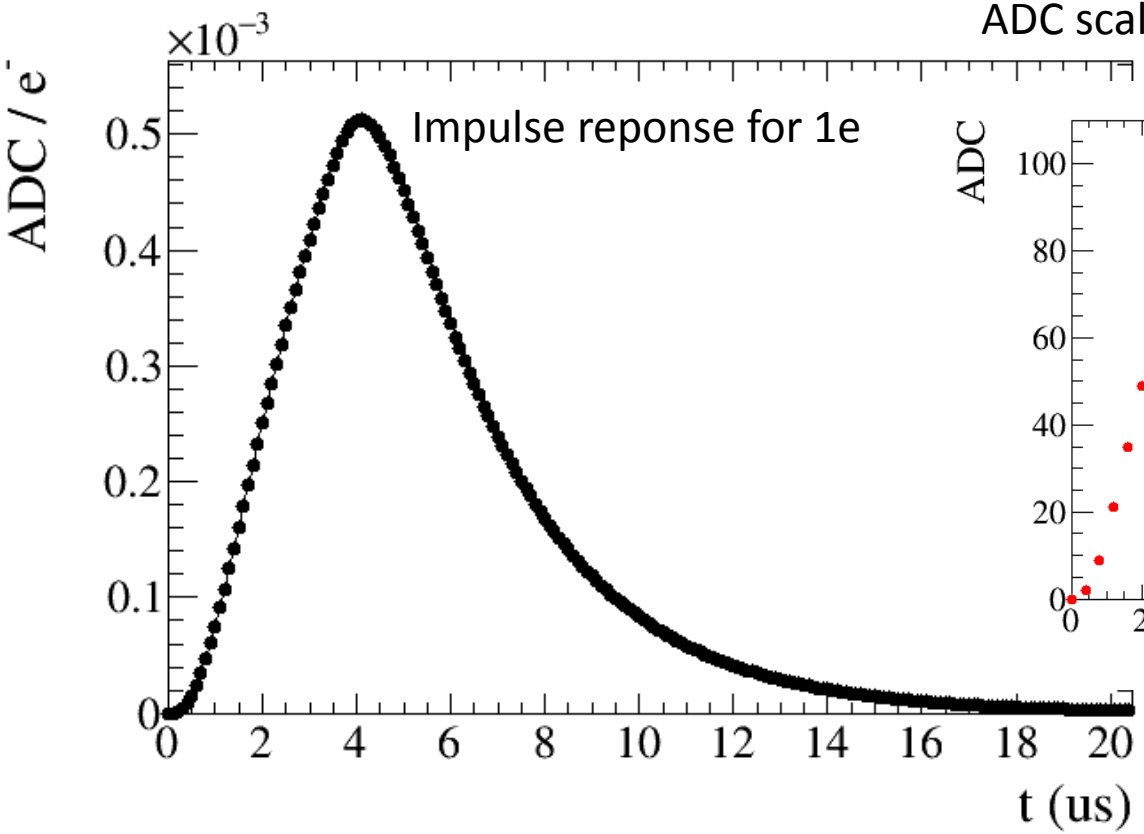
Set such that MIP is at 100 ADC → 100 ADC / 31.25 = 3.2 ADC / fC

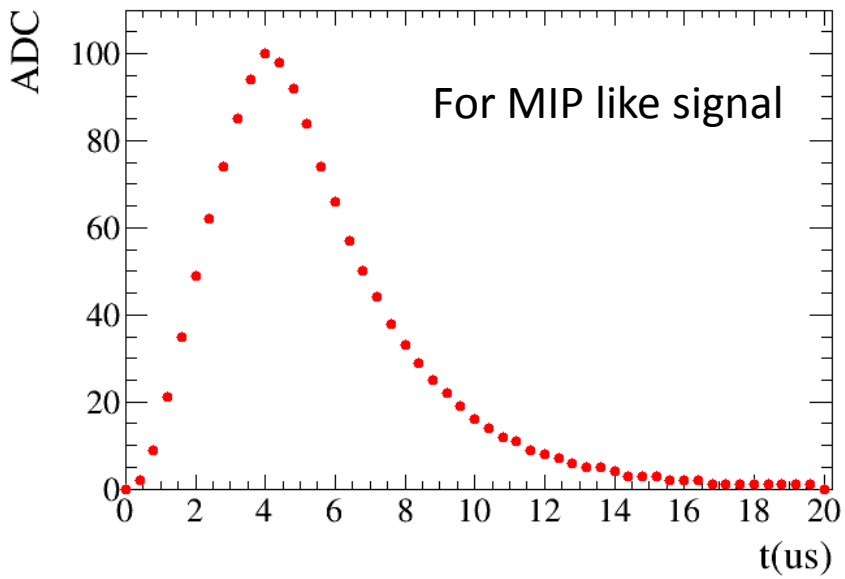
So parameters: *ASICmVperfC* = 3.2

ADCpermV = 1.0

for now I just lumped the factor into one place *ASICmVperfC* and no double-slope

But in reality FE gain would be fixed and ADC scale could be adjusted





DP signal service
dune/Utilities/SignalShapingDUNEDPhase

```
# DUNE dual-phase signal service
#
dunefddphase_signalshapingservice:
{
  ASICmVperfC: 3.2     # Amplifier gain in mV/fC
  ADCpermV:    1.0     # ADC conversion factor
  AmpENC:      1700.0 # Amplifier ENC for Cdet=450pF
```

The list to be expanded with my understanding of what is required for deconvolution

The parameters are set in signalservices_dune.fcl / **dunefddphase_signalshapingservice**

```
double max = 0;
for(size_t i = 0; i < ElecResp.size(); ++i)
  {
    //convert time to microseconds, to match response function definition
    time[i]     = (1.*i)*fRespSamplingPeriod*1e-3;
    ElecResp[i] = PreampETHZ(time[i]);

    if(ElecResp[i] > max) max = ElecResp[i];
  }// end loop over time buckets


LOG_DEBUG("SignalShapingDUNEDPhase") << " Done.";

//normalize to 1e charge before the convolution
for(auto& element : ElecResp)
  {
    element /= max;
    element *= fASICfCtomV * 1.60217657e-4; //mV
    element *= fADCmVtoADC;                 //ADC
  }
```

Calculation of the impulse response vector fRespSamplingPeriod could be the same or much finer than ADC sampling
→ Rebin later

Normalization of the response vector

DP channel extractor service

From DetSim/detsimmodules_dune.fcl

```
# signal extractor service for dune dual-phase detector
scxdp: {
  service_provider: DPhaseSimChannelExtractService
  DPGainPerView: 10.0
}
```

Our entry to point of access to SimChannel after GEANT simulation

The key item is the DP amplification factor

Could in principle specify LEM field value

→ convert to gain via some parametrized functional dependence inside the service

```
class DPhaseSimChannelExtractService : public SimChannelExtractService {

public:

  DPhaseSimChannelExtractService(fhicl::ParameterSet const& pset, art::ActivityRegistry&);

  int extract(const sim::SimChannel* psc, AdcSignalVector& sig) const;

private:

  // standard larsoft FFT service
  art::ServiceHandle<util::LArFFT> m_pfft;
  // dual-phase signal response service
  art::ServiceHandle<util::SignalShapingServiceDUNEDPhase> m_psss;

  unsigned int m_ntick;

  float fDPGainPerView; // gain in dual-phase

};
```

← Signal service for DP: simulation of FE electronics

DUNE DP services
(Utilities/services_dune.fcl)

```
dunefddphase_simulation_services:
{
 LArFFT:                          @local::dunefd_larfft
 LArG4Parameters:                 @local::dunefd_largeantparameters
 ExptGeoHelperInterface:          @local::dune_geometry_helper
 LArVoxelCalculator:              @local::dunefd_larvoxelcalculator
 MagneticField:                   @local::no_mag
 BackTracker:                     @local::dunefd_backtracker
 SeedService:                     @local::dune_seedservice
 DatabaseUtil:                    @local::dunefd_database
 OpDetResponseInterface:          @local::dunefd_opdetresponse
 IChannelStatusService:           @local::dunefd_channel_status

 Geometry:                        @local::dunedphase10kt_geo
 TimeService:                     @local::dunefddphase_timeservice
 DetectorProperties:              @local::dunefddphase_detproperties
 LArProperties:                   @local::dunefddphase_properties
 SignalShapingServiceDUNEDPhase:  @local::dunefddphase_signalshapingservice
 SimChannelExtractService:        @local::scxdp
}
```

Services common with SP

Not used & to be modified with DP light readout implementation

DUNE DP services (Utilities/services_dune.fcl)

```
dunefddphase_simulation_services:
{
 LArFFT:                          @local
 LArG4Parameters:                 @local
 ExptGeoHelperInterface:          @local
 LArVoxelCalculator:              @local
 MagneticField:                   @local
 BackTracker:                     @local
 SeedService:                     @local
 DatabaseUtil:                    @local
 OpDetResponseInterface:          @local
 IChannelStatusService:           @local

 Geometry:                        @local::dunedphase10kt_geo
 TimeService:                     @local::dunefddphase_timeservice
 DetectorProperties:              @local::dunefddphase_detproperties
 LArProperties:                   @local::dunefddphase_properties
 SignalShapingServiceDUNEDPhase:  @local::dunefddphase_signalshapingservice
 SimChannelExtractService:        @local::scxdp
}
```

Geometry/geometry_dune.fcl

```
dunedphase10kt_geo:
{
 Name:       "dunedphase10kt_v2"

 GDML:       "dunedphase10kt_v2.gdml"
 ROOT:       "dunedphase10kt_v2.gdml"

 SurfaceY: 147828

 DisableWiresInG4:    true
}
```

DUNE DP services (Utilities/services_dune.fcl)

Utilities/timeservice_dune.fcl

```
dunefddphase_timeservice: @local::standard_timeservice
# dunefddphase_timeservice.TrigModuleName:      ""
dunefddphase_timeservice.InheritClockConfig:   false
dunefddphase_timeservice.G4RefTime:             0.   # G4 time [us] where electronics clock counting start
dunefddphase_timeservice.TriggerOffsetTPC:      0.   # Time [us] for TPC readout start w.r.t. trigger time
dunefddphase_timeservice.FramePeriod:        8000.   # Frame period [us]
dunefddphase_timeservice.ClockSpeedTPC:         2.5 # TPC clock speed in MHz
dunefddphase_timeservice.ClockSpeedOptical:    65.   # Optical clock speed in MHz
dunefddphase_timeservice.ClockSpeedTrigger:    16.   # Trigger clock speed in MHz
dunefddphase_timeservice.DefaultTrigTime:       0.   # Default trigger time [us].
dunefddphase_timeservice.DefaultBeamTime:       0.   # Default beam time [us].
```

```
 Geometry:                       @local::dunedphase10kt_geo
 TimeService:                    @local::dunefddphase_timeservice
 DetectorProperties:             @local::dunefddphase_detproperties
 LArProperties:                  @local::dunefddphase_properties
 SignalShapingServiceDUNEDPhase: @local::dunefddphase_signalshapingservice
 SimChannelExtractService:       @local::scxdp
}
```

DUNE DP services (Utilities/services_dune.fcl)

```
dunefddphase_simulation_services:
{
 LArFFT:                          @local::dunefd_larfft
```

Utilities/detproperties_dune.fcl

```
dunefddphase_detproperties:                  @local::standard_detproperties
# dunefddphase_detproperties.SamplingRate:   400.          # in ns
dunefddphase_detproperties.ElectronsToADC:   5.1267e-04   #  Not sure where this is used
dunefddphase_detproperties.NumberTimeSamples: 20000       # drift length/drift velocity*sampling rate
dunefddphase_detproperties.ReadOutWindowSize: 20000       # drift length/drift velocity*sampling rate
dunefddphase_detproperties.TimeOffsetU:      0.
dunefddphase_detproperties.TimeOffsetV:      0.
dunefddphase_detproperties.TimeOffsetZ:      0.
```

```
 TimeService:                     @local::dunefddphase_timeservice
 DetectorProperties:              @local::dunefddphase_detproperties
 LArProperties:                   @local::dunefddphase_properties
 SignalShapingServiceDUNEDPhase:  @local::dunefddphase_signalshapingservice
 SimChannelExtractService:        @local::scxdp
}
```

Calculated assuming mip at 100 ADC with 31.25 fC/strip @ Gain = 10 per view : 1fC to ADC = 3.2

DUNE DP services (Utilities/services_dune.fcl)

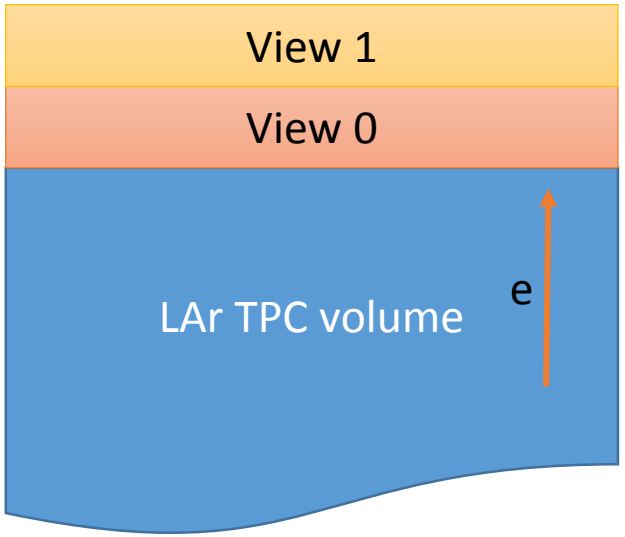
```
dunefddphase_simulation_services:
{
 LArFFT:                          @local::dunefd_
 LArG4Parameters:                 @local::dunefd_
 ExptGeoHelperInterface:          @local::dune_ge
 LArVoxelCalculator:              @local::dunefd_
 MagneticField:                   @local::no_mag
 BackTracker:                     @local::dunefd_backtrac
 LArProperties:                   @local::dunefddphase_properties
 SignalShapingServiceDUNEDPhase:  @local::dunefddphase_signalshapingservice
 SimChannelExtractService:        @local::scxdp
}
```

Utilities/larproperties_dune.fcl

```
dunefddphase_properties:                    @loca      :standard_properties
dunefddphase_properties.Temperature:        87
dunefddphase_properties.Electronlifetime:   3.0e3                # us
dunefddphase_properties.Efield:             [0.5,4.0,0.0]        #  kV/cm
```

The second value should be large enough (max is actually 4 kV/cm) such that: $d_U/v_{d,U} = t_U \ll t_{sample}$, the charge depositions appear at the same time bin in two view
The last value should be 0

Timing issue for DP

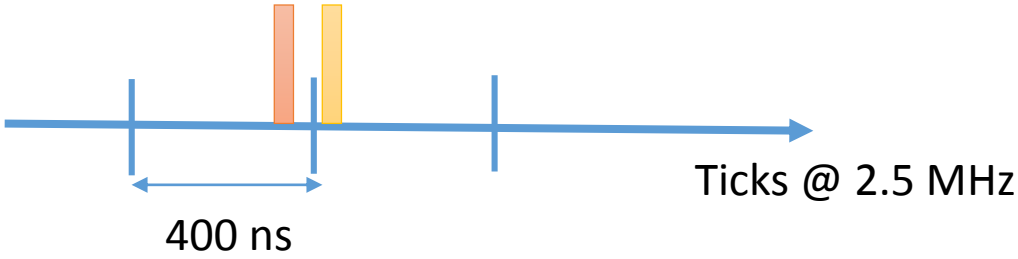


$\Delta = 0.3$ mm, with drift velocity v_d^0 @ $\left(x \frac{kV}{cm}\right)$

Time of arrival in view 1:

$$t_1 = t_0 + \frac{\Delta}{v_d^0} = t_0 + \frac{0.3}{3.5}$$ ➔ ~86 ns offset

Though this is large enough compared to 400ns tick that one can get charge distributions between two views appearing shifted by 1 time tick in some cases



DUNE DP services (Utilities/services_dune.fcl)

```
dunefddphase_simulation_services:
{
 LArFFT:                          @local::dunefd_larfft
 LArG4Parameters:
 ExptGeoHelperInterface:
 LArVoxelCalculator:
 MagneticField:
 BackTracker:
 SeedService:
 DatabaseUtil:
 OpDetResponseInterface:
 IChannelStatusService:

 Geometry:
 TimeService:
 DetectorProperties:
 LArProperties:                   @local::dunefddphase_properties
 SignalShapingServiceDUNEDPhase:  @local::dunefddphase_signalshapingservice
 SimChannelExtractService:        @local::scxdp
}
```

Utilities/signalservices_dune.fcl

```
#
# DUNE dual-phase signal service
#
dunefddphase_signalshapingservice:
{
  ASICmVperfC: 3.2    # Amplifier gain in mV/fC
  ADCpermV:    1.0    # ADC conversion factor
  AmpENC:      1700.0 # Amplifier ENC e for Cdet=450pF

  RespSamplingPeriod: 100. # in nano second
}
```

DUNE DP services (Utilities/services_dune.fcl)

```
dunefddphase_simulation_services:
{
 LArFFT:                          @local::dunefd_larfft
 LArG4Parameters:                 @local::dunefd_largeantparameters
 ExptGeoHelperInterface:          @local::dune_geometry_helper
 LArVoxelCalculator:              @local::dunefd_larvoxelcalculator
 MagneticField:                   @local::no_mag
 BackTracker:                     @local::dunefd_backtracker
 SeedService:                     @local::dune_seedservice
```

DetSim/detsimmodules_dune.fcl

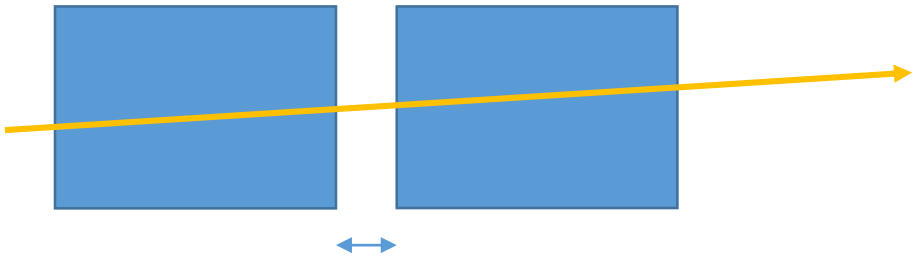
```
# signal extractor service for dune dual-phase detector
scxdp: {
  service_provider: DPhaseSimChannelExtractService
  DPGainPerView: 10.0
}
```

```
 SignalShapingServiceDUNEDPhase:  @local::dunefddphase_signalshapingservice
 SimChannelExtractService:        @local::scxdp
}
```

Some warnings during execution

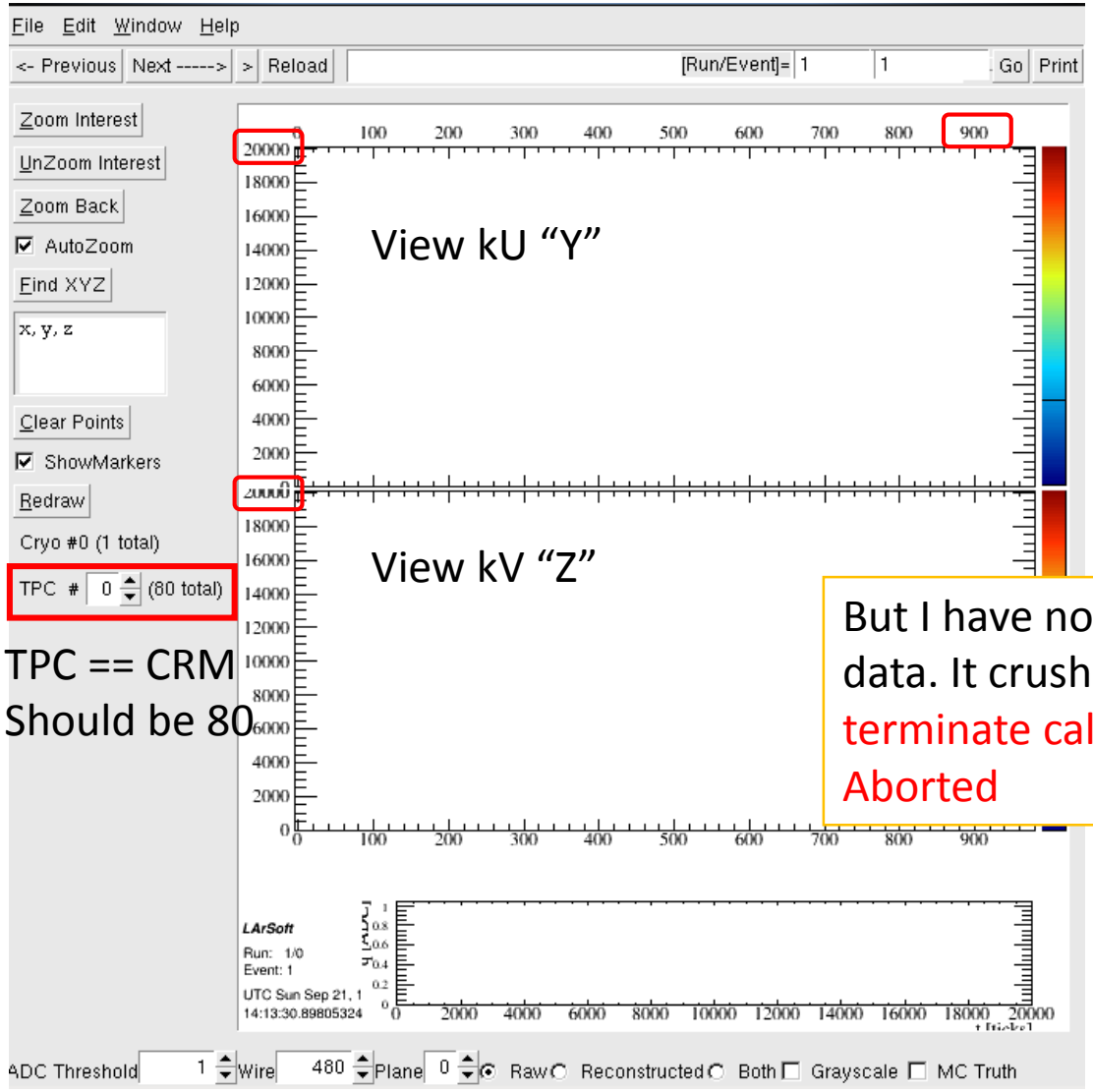
```
%MSG
%MSG-w LArVoxelReadout:  LArG4:largeant 22-Jan-2016 15:36:52 CET  run: 1 subRun: 0 event: 1
unable to drift electrons from point (-0.0141596,-150.135,300.11) with exception ---- Geometry BEGIN
  Can't Find Nearest Wire for position (600.112,-150.693,300.489) in plane C:0 T:1 P:1 approx wire number # 960 (capped from 959)
---- Geometry END

%MSG
%MSG-w LArVoxelReadout:  LArG4:largeant 22-Jan-2016 15:36:52 CET  run: 1 subRun: 0 event: 1
unable to drift electrons from point (-0.0141454,-150.135,300.17) with exception ---- Geometry BEGIN
  Can't Find Nearest Wire for position (600.112,-150.513,300.496) in plane C:0 T:1 P:1 approx wire number # 960 (capped from 959)
---- Geometry END

%MSG
%MSG-w LArVoxelReadout:  LArG4:largeant 22-Jan-2016 15:36:52 CET  run: 1 subRun: 0 event: 1
unable to drift electrons from point (-0.0141384,-150.135,300.2) with exception ---- Geometry BEGIN
  Can't Find Nearest Wire for position (600.112,-150.606,300.825) in plane C:0 T:1 P:1 approx wire number # 961 (capped from 959)
---- Geometry END
```

These appear to be generated when a particle crosses the “dead” regions b/w or around CRMs



Event display

Configured evdservices_dune.fcl to pick up correct geometry and other services



TPC == CRM

Should be 80

Should show 960 ch x 20,000 time samples

But I have not managed to get it to display raw data. It crushes without saying much:

terminate called without an active exception

Aborted

Basic check

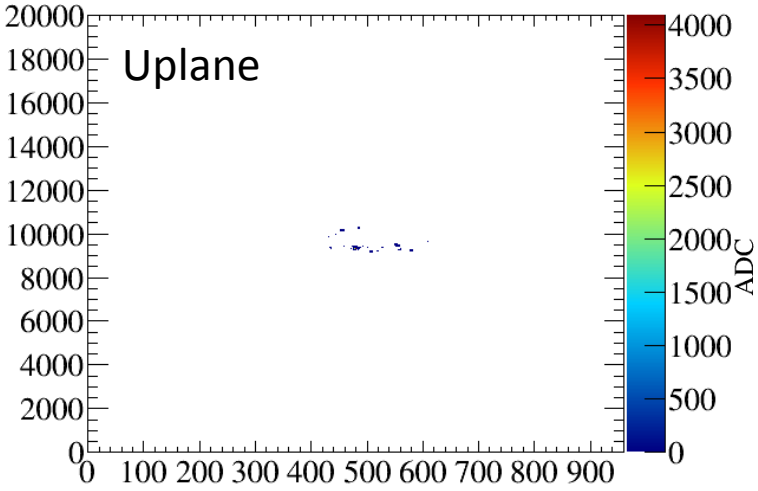
Created “analysis” module in dune/RawdataDisplay similar to the ones found for SP which dumps raw data into histograms

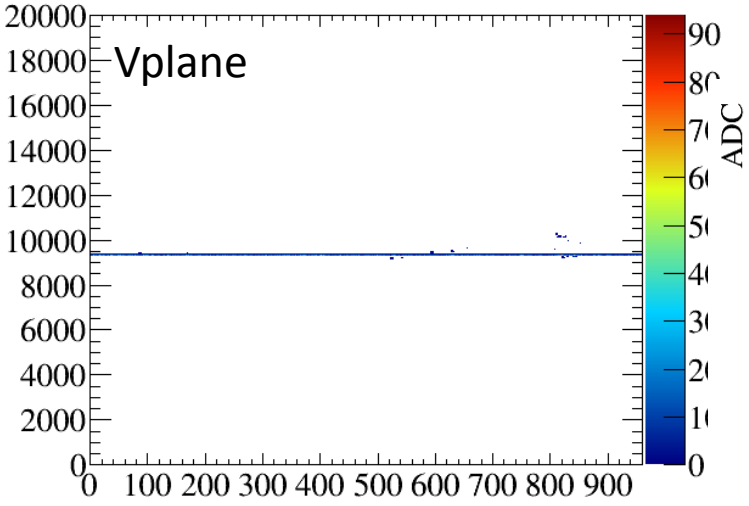
Example created with workspace geometry (= 2 CRMs):

```
root [3] .ls
TDirectoryFile*         rawdraw rawdraw (RawEVDDP) folder
 KEY: TH2I      fTimeChanU0;1   Time vs Channel(Plane U, TPC0)
 KEY: TH2I      fTimeChanThumbU0;1      Time vs Channel(Plane U, TPC0)
 KEY: TH1I      fADCMaxU0;1     Max ADC per channel (Plane U, TPC0)
 KEY: TH2I      fTimeChanV0;1   Time vs Channel(Plane V, TPC0)
 KEY: TH2I      fTimeChanThumbV0;1      Time vs Channel(Plane V, TPC0)
 KEY: TH1I      fADCMaxV0;1     Max ADC per channel (Plane V, TPC0)
 KEY: TH2I      fTimeChanU1;1   Time vs Channel(Plane U, TPC1)
 KEY: TH2I      fTimeChanThumbU1;1      Time vs Channel(Plane U, TPC1)
 KEY: TH1I      fADCMaxU1;1     Max ADC per channel (Plane U, TPC1)
 KEY: TH2I      fTimeChanV1;1   Time vs Channel(Plane V, TPC1)
 KEY: TH2I      fTimeChanThumbV1;1      Time vs Channel(Plane V, TPC1)
 KEY: TH1I      fADCMaxV1;1     Max ADC per channel (Plane V, TPC1)
```

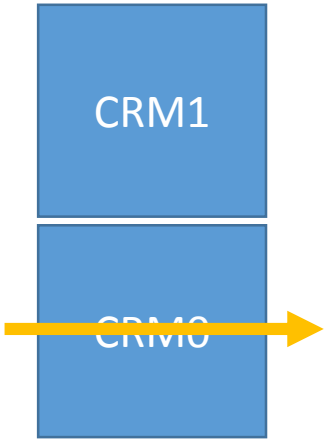
fTimeChanXX – standard time tick vs channel histo
fADCMaxXX – distribution of max ADC values from each channel

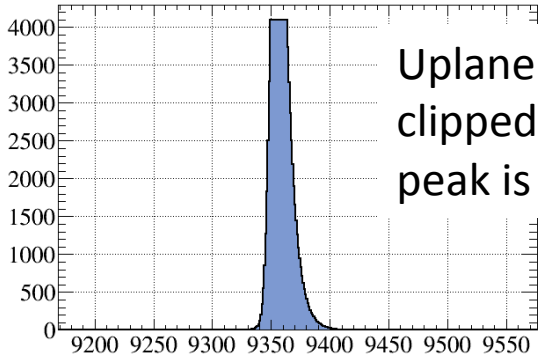
Time tick = 6000 / 1.6056 mm/us / 0.4 us 9342
From “Vplane” 9342 ➔ OK



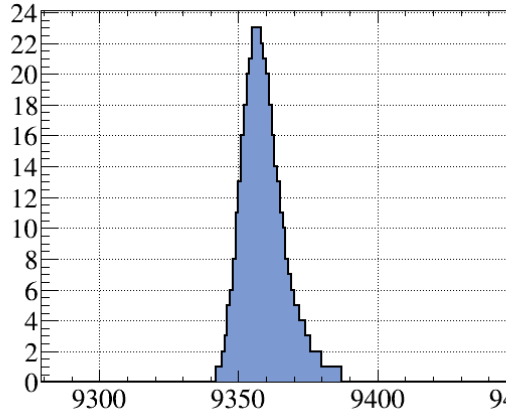


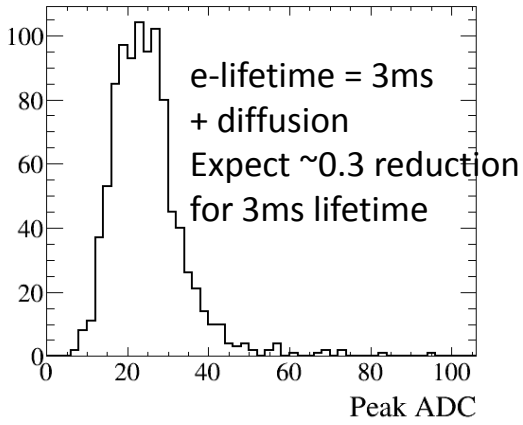
60 GeV mu through the
middle of 1 TPC at d = 6m



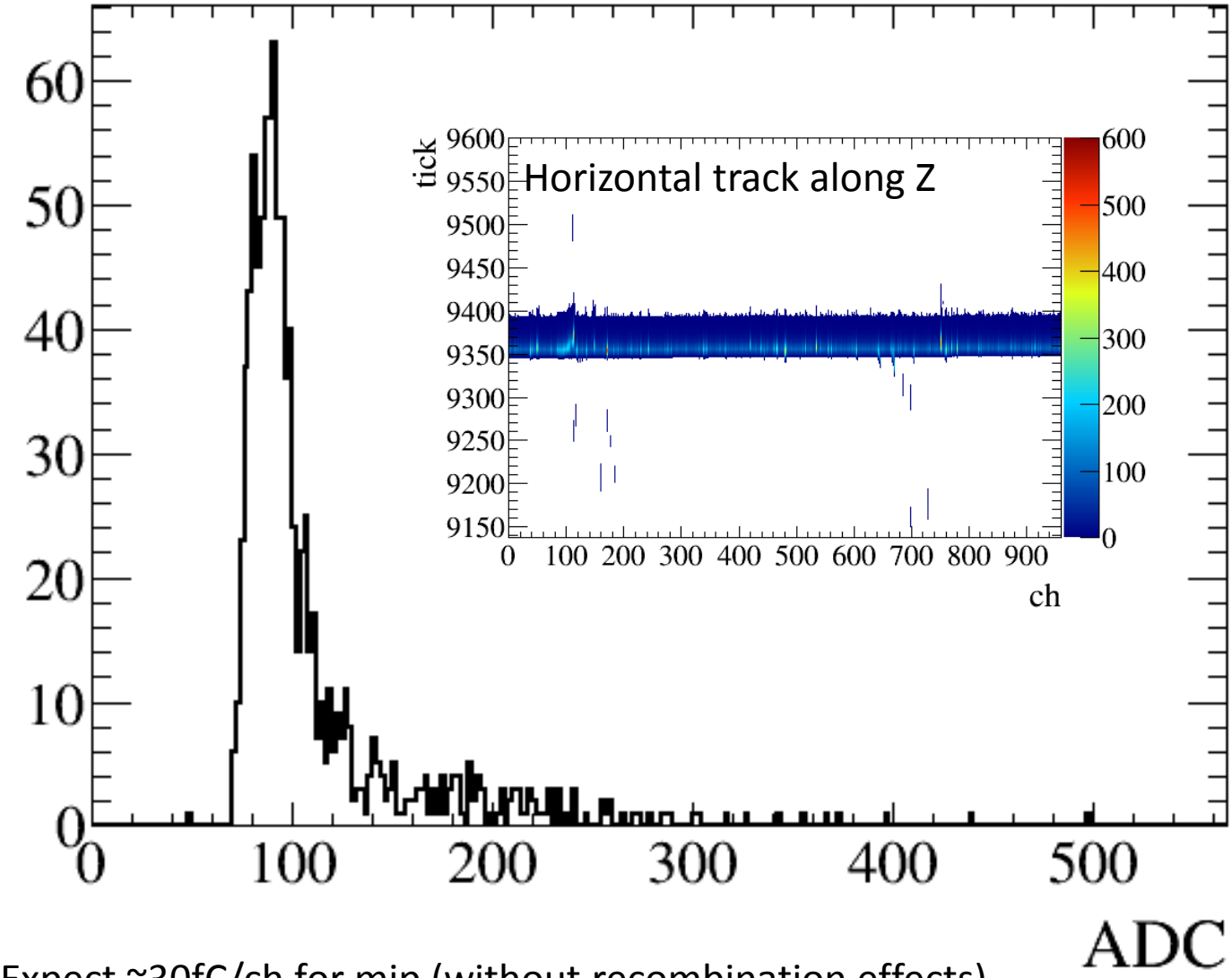


Uplane ch with max hit
clipped at 4095 (12bit ADC)
peak is ~480 ch ← ok



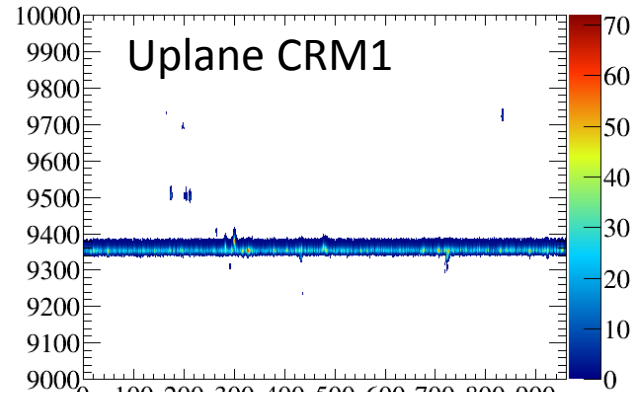


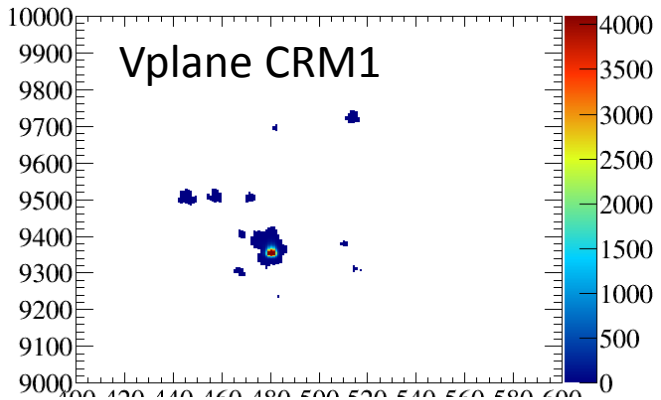
Pulse height distribution in ADC (infinite lifetime / no diffusion)

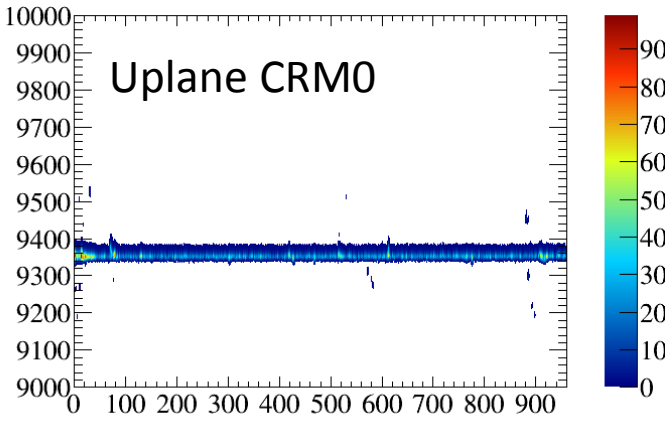


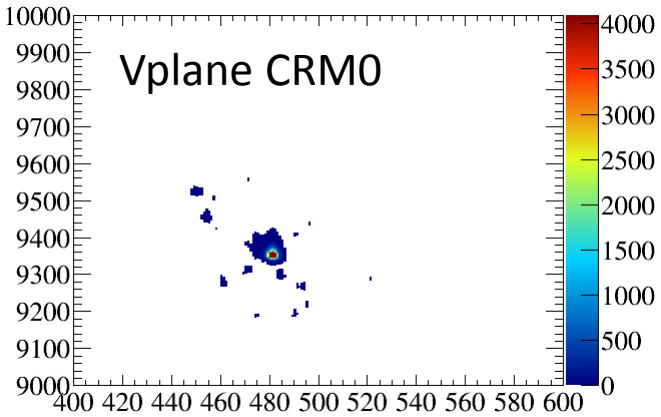
Expect ~30fC/ch for mip (without recombination effects)

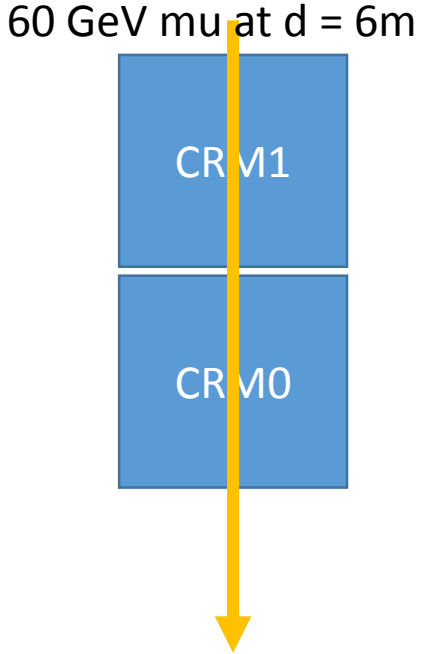
➔ ~100ADC/ch in our normalization


Uplane CRM1

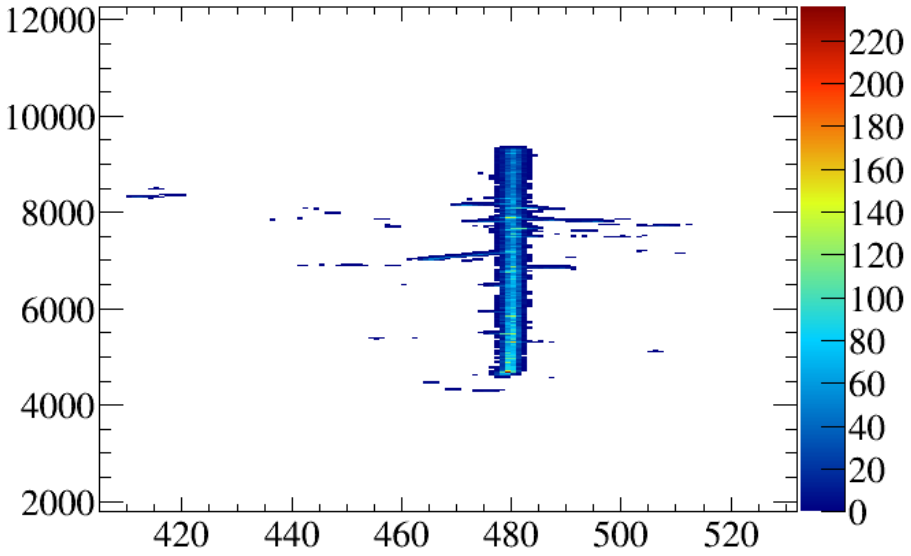

Vplane CRM1


Uplane CRM0

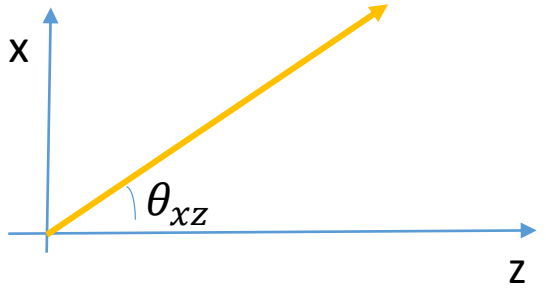

Vplane CRM0


60 GeV mu at d = 6m
CRM1
CRM0

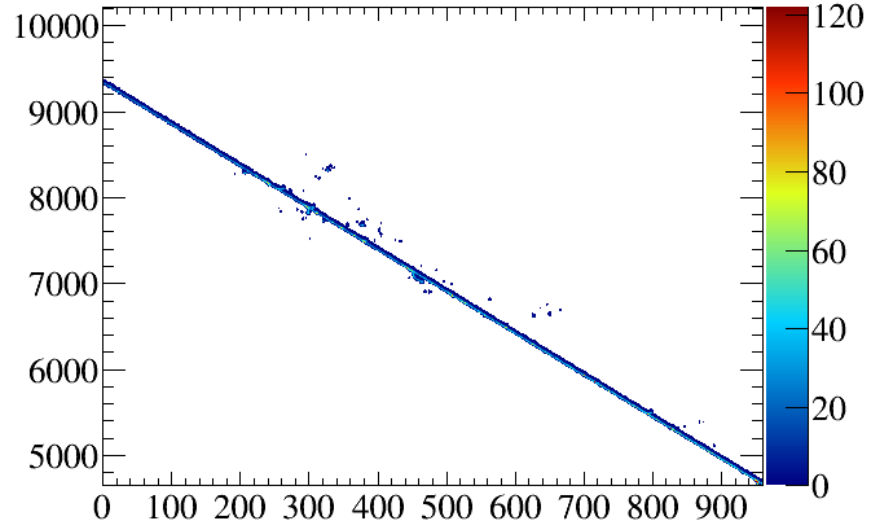
Plane U in CRM



60 GeV mu through the center of 1 TPC at d = 6m & theta0XZ = 45deg



Plane V in CRM



Drift velocity from slope:

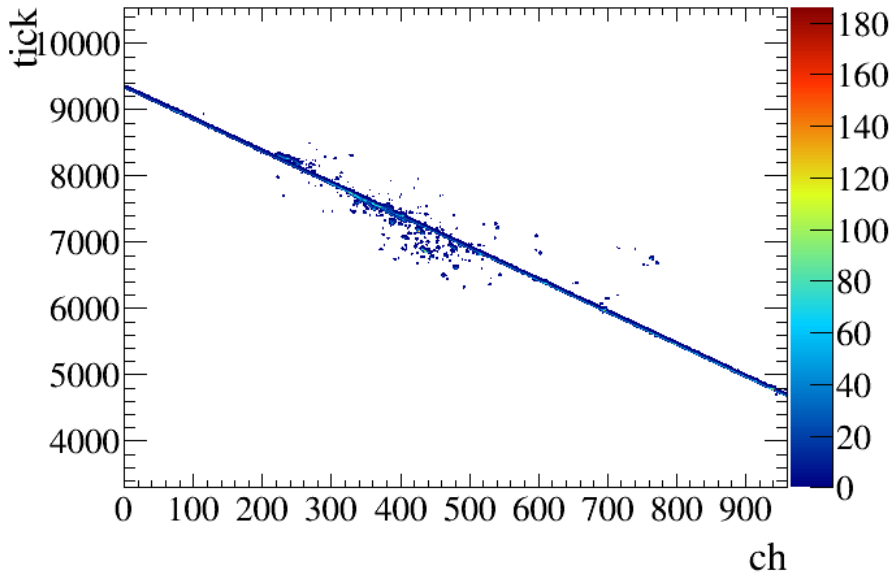
$$\frac{3000}{\Delta \times 0.4} = 1.61 \text{ mm/us}$$

Theoretical (from larsoft function) at 0.5 kV/cm and 87K = 1.6056 mm/us

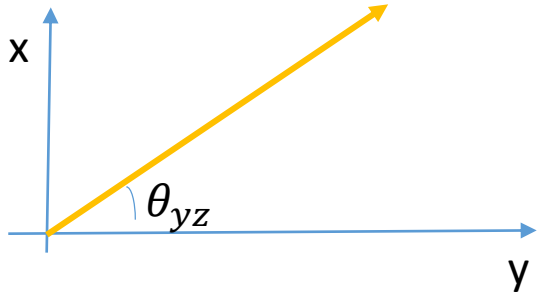
➔ Looks OK

Also the direction is consistent as in this case particle travels towards readout so the drift d gets shorter

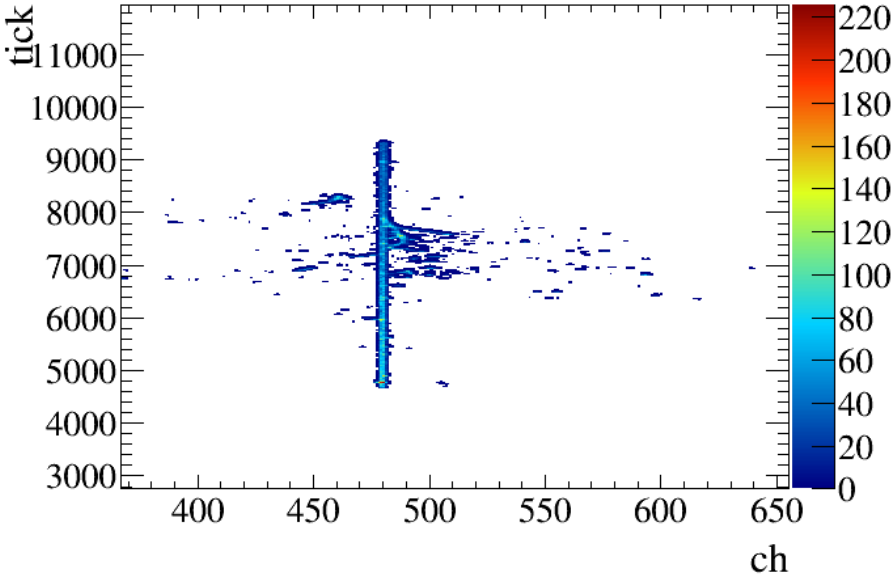
Plane U in CRM



60 GeV mu through the center of 1 TPC at d = 6m & theta0YZ = 45deg (theta0XZ = 90)



Plane V in CRM



Drift velocity from slope:

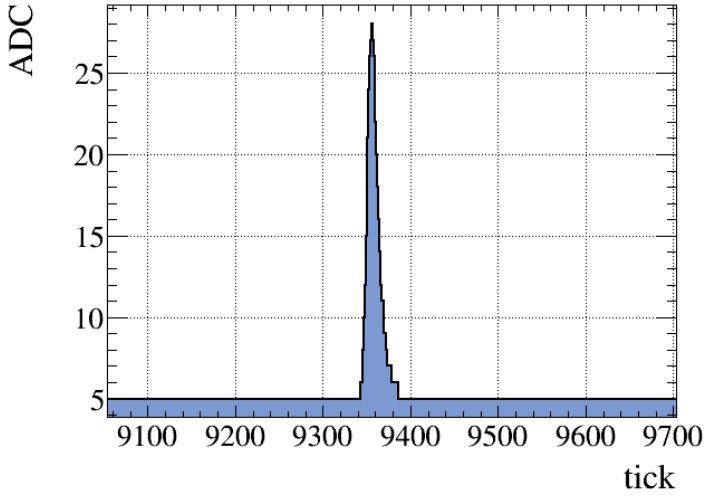
$$\frac{3000}{\Delta \times 0.4} = 1.61 \text{ mm/us}$$

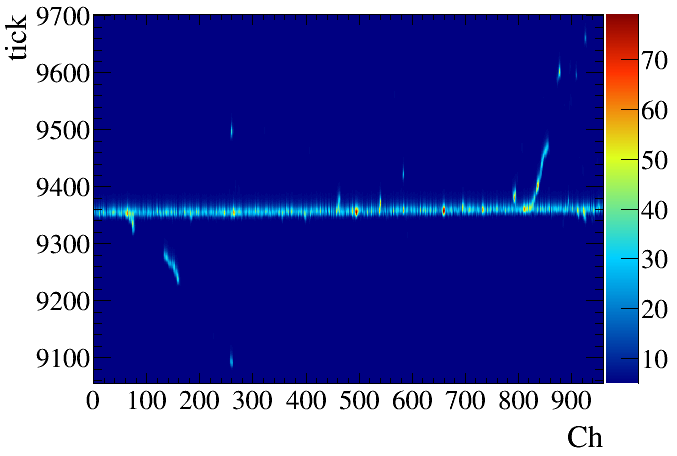
The direction looks ok: larger channel # correspond to earlier time bins

Adding pedestal

Use existing framework structure for DUNE SP

```
peddplegacy: {
  service_provider: SIOVDetPedestalService
  DetPedestalRetrievalAlg: {
    AlgName:              "DetPedestalRetrievalAlg"
    DatabaseRetrievalAlg: {
      AlgName: "DatabaseRetrievalAlg"
      DBFolderName:  ""
      DBUrl: ""
      DBTag: ""
    }
    UseDB: false
    UseFile: false
    DefaultCollMean:  5.0
    DefaultCollRms:   0.0
    DefaultIndMean:   5.0
    DefaultIndRms:    0.0
    DefaultMeanErr:   0.0
    DefaultRmsErr:    0.0
  }
}
```





For DP the two numbers should be identical here!

Noise implementation

Noise service fcl

The noise services add noise to signals

- What was previously noise option 1 is now

```
chnoiseold: {
  service_provider: ExponentialChannelNoiseService
  NoiseFactU: 0.05
  NoiseFactV: 0.05
  NoiseFactZ: 0.05
  NoiseWidthU: 2000
  NoiseWidthV: 2000
  NoiseWidthZ: 2000
  LowCutoffU: 7.5
  LowCutoffV: 7.5
  LowCutoffZ: 7.5
  NoiseArrayPoints: 1000
  OldNoiseIndex: true
}
```

Could be probably be used as is with appropriate parameters

- And noise option 2 is

```
chnoisewhite: {
  service_provider: WhiteChannelNoiseService
}
```

This is more specific to SP (i.e., relies explicitly on SingalShapingServiceDUNE35t)

Summary

- Simulation of charge readout appears to be working (largeant → daq → rawdigit)
- The code has been committed in dunetpc: should appear in v04_35 release
- Major part remaining is simulation of light readout