

Pandora event display

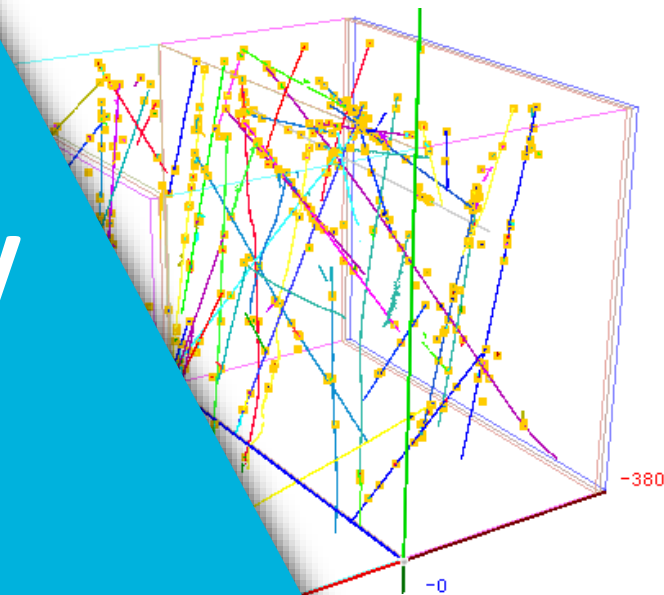
Part 2: Visualizing the algorithms

(Exercise)

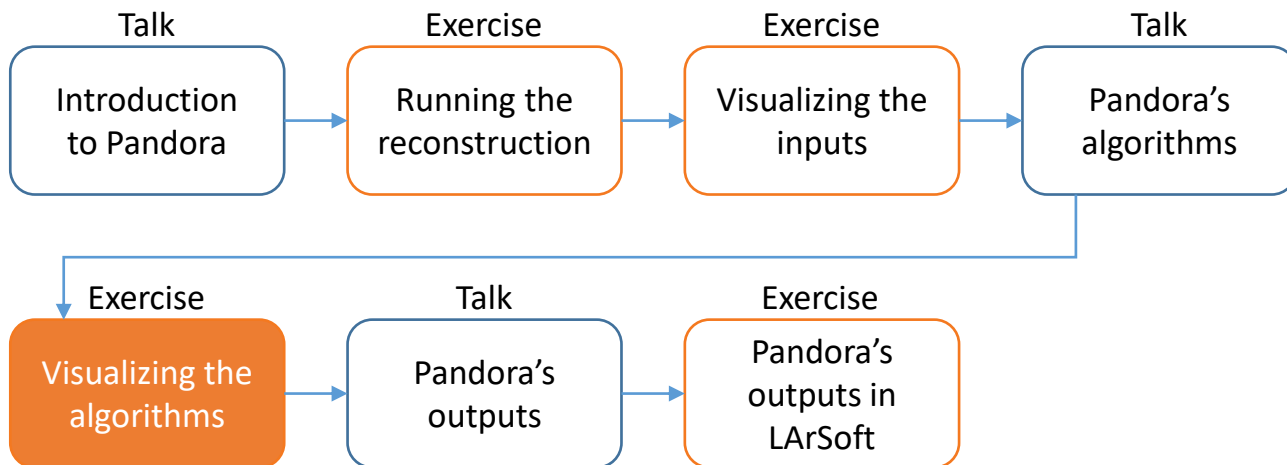
Isobel Mawby and Steve Dennis for the Pandora team

09/11/2022

7th UK LArTPC Software and Analysis Workshop



Reconstruction session



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

Key references: [Pandora ProtoDUNE paper](#)
[Pandora MicroBooNE paper](#)

Goals

- This session scheduled for 45 minutes
- Main goal - Visualize the status of the pattern-recognition after each main stage
 - Add the visual monitoring algorithm to the Pandora configuration XML file after running the:
 - 2D reconstruction
 - 3D vertex reconstruction
 - Track & Shower reconstruction & particle refinement
 - 3D hit reconstruction
 - Neutrino hierarchy reconstruction
- Please don't worry if you don't get through all of the steps
 - This session is just for you to get some intuition for what Pandora's algorithms do

Main goal

Visualize the algorithms

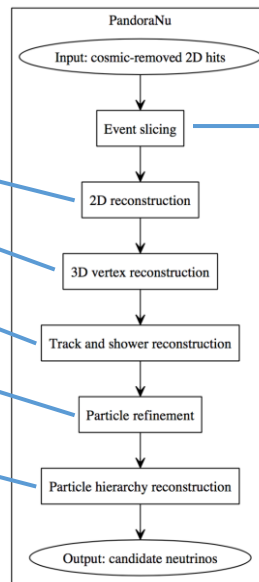
The neutrino algorithm chain

- Go to our config directory and make a copy of the Pandora `neutrino` XML settings file

```
$ cd $MRB_TOP/reco/config
$ cp $LARPANDORA_DIR/scripts/PandoraSettings_Neutrino_Standard.xml MyPandoraSettings_Neutrino_Standard.xml
$ vim MyPandoraSettings_Neutrino_Standard.xml
```

- Look through the file for the sections listed below:

```
<!-- TwoDReconstruction -->
<!-- VertexAlgorithms -->
<!-- ThreeDTrackAlgorithms -->
<!-- ThreeDShowerAlgorithms -->
<!-- Repeat ThreeDTrackAlgorithms -->
<!-- ThreeDRecoveryAlgorithms -->
<!-- TwoDMopUpAlgorithms -->
<!-- ThreeDHitAlgorithms -->
<!-- ThreeDMopUpAlgorithms -->
<!-- NeutrinoAlgorithms -->
<!-- Track and shower building -->
```



We're not running the event slicing because we don't have cosmics to deal with

Point to our neutrino settings file

- Modify `MyPandoraSettings_Master_Standard.xml` and point it to our new neutrino settings file
- Remove the visual monitoring algorithms that we've been using so far

```
+ MyPandoraSettings_Master_Standard.xml
<pandora>
  <!-- GLOBAL SETTINGS -->
  <IsMonitoringEnabled>true</IsMonitoringEnabled>
  <ShouldDisplayAlgorithmInfo>true</ShouldDisplayAlgorithmInfo>
  <SingleHitTypeClusteringMode>true</SingleHitTypeClusteringMode>

  <!-- ALGORITHM SETTINGS -->
  <algorithm type = "LArPreProcessing">
    <OutputCaloHitListNameU>CaloHitListU</OutputCaloHitListNameU>
    <OutputCaloHitListNameV>CaloHitListV</OutputCaloHitListNameV>
    <OutputCaloHitListNameW>CaloHitListW</OutputCaloHitListNameW>
    <FilteredCaloHitListName>CaloHitList2D</FilteredCaloHitListName>
    <CurrentCaloHitListReplacement>CaloHitList2D</CurrentCaloHitListReplacement>
  </algorithm>

  <algorithm type = "LArVisualMonitoring">
    <CaloHitListNames>CaloHitListU CaloHitListV CaloHitListW</CaloHitListNames>
    <ShowDetector>true</ShowDetector>
  </algorithm>

  <algorithm type = "LArMaster">
    <CRSettingsFile>PandoraSettings_Cosmic_Standard.xml</CRSettingsFile>
    <NuSettingsFile>MyPandoraSettings_Neutrino_Standard.xml</NuSettingsFile>
    <SlicingSettingsFile>PandoraSettings_Slicing_Standard.xml</SlicingSettingsFile>

  ... more settings ...

  <algorithm type = "LArVisualMonitoring">
    <ShowCurrentPfos>true</ShowCurrentPfos>
    <ShowDetector>true</ShowDetector>
  </algorithm>
</pandora>
```

Remove this algorithm block

Change this line to point to
`MyPandoraSettings_Neutrino_Standard.xml`

Remove this algorithm block

2D reconstruction

Add in some visualizations

- Add to `MyPandoraSettings_Neutrino_Standard.xml` at the end of the `TwoDReconstruction` section

```

../c/MyPandoraSettings_Neutrino_Standard.xml
<pandora>
  <!-- Output list management -->
  <!-- GLOBAL SETTINGS -->
  <IsMonitoringEnabled>true</IsMonitoringEnabled>
  <ShouldDisplayAlgorithmInfo>true</ShouldDisplayAlgorithmInfo>
  <SingleHitTypeClusteringMode>true</SingleHitTypeClusteringMode>

  ... more settings ...

  <algorithm type = "LARkinkSplitting"/>
  <algorithm type = "LARTrackConsolidation">
    <algorithm type = "LARSimpleClusterCreation" description = "ClusterRebuilding"/>
  </algorithm>

  <algorithm type = "LARVisualMonitoring">
    <CaloHitListNames>CaloHitListU</CaloHitListNames>
    <ClusterListNames>ClustersU</ClusterListNames>
    <ShowDetector>true</ShowDetector>
  </algorithm>

  <algorithm type = "LARVisualMonitoring">
    <CaloHitListNames>CaloHitListV</CaloHitListNames>
    <ClusterListNames>ClustersV</ClusterListNames>
    <ShowDetector>true</ShowDetector>
  </algorithm>

  <algorithm type = "LARVisualMonitoring">
    <CaloHitListNames>CaloHitListW</CaloHitListNames>
    <ClusterListNames>ClustersW</ClusterListNames>
    <ShowDetector>true</ShowDetector>
  </algorithm>

  <!-- VertexAlgorithms -->
  <algorithm type = "LARCandidateVertexCreation">
    <InputClusterListNames>ClustersU ClustersV ClustersW</InputClusterListNames>
    <OutputVertexListName>CandidateVertices3D</OutputVertexListName>
    <ReplaceCurrentVertexList>true</ReplaceCurrentVertexList>

```

Set this to `true` - this will print to the terminal all of the algorithms we are running

Modify the **Neutrino** file not the **Master** settings file

Add these visual monitoring blocks. When we run, this will make 3 event displays - each showing the `hits` and `clusters` in the U, V and W views respectively

Add the above lines just before the `VertexAlgorithms` section

Visualizing the initial 2D reconstruction

```
$ cd $MRB_TOP/reco/work
$ lar -c event_display_driver.fcl -s reco_events.root -n 1
```

Let's just look at 1 event for now!

Can also run on pre-made reco file in /home/share/november2022/reconstruction/

2D clustering
algorithms in
the U-view

2D clustering
algorithms in
the V-view

2D clustering
algorithms in
the U-view

First visualization

The screenshot shows the LArSoft event display interface. The left pane displays a list of algorithms running, including:

- registering to steppingActionsMap
- Loading the VUV time parametrization
- Loading the VIS time parametrization
- 07-Oct-2021 07:48:28 CDT Opened input file "reco_events.root"
- Running Algorithm: Alg0001, LarPreProcessing
- Running Algorithm: Alg0002, LarClusteringParent
- Running Algorithm: Alg0003, LarTrackClusterCreation
- Running Algorithm: Alg0004, LarLayersSplitting
- Running Algorithm: Alg0005, LarLongitudinalAssociation
- Running Algorithm: Alg0006, LarTransverseAssociation
- Running Algorithm: Alg0007, LarLongitudinalExtension
- Running Algorithm: Alg0008, LarTransverseExtension
- Running Algorithm: Alg0009, LarCrossgapsAssociation
- Running Algorithm: Alg0010, LarCrossgapsExtension
- Running Algorithm: Alg0011, LarOvershootSplitting
- Running Algorithm: Alg0012, LarBranchSplitting
- Running Algorithm: Alg0013, LarKinkSplitting
- Running Algorithm: Alg0014, LarTrackConsolidation
- Running Algorithm: Alg0016, LarClusteringParent
- Running Algorithm: Alg0017, LarTrackClusterCreation
- Running Algorithm: Alg0018, LarLayersSplitting
- Running Algorithm: Alg0019, LarLongitudinalAssociation
- Running Algorithm: Alg0020, LarTransverseAssociation
- Running Algorithm: Alg0021, LarLongitudinalExtension
- Running Algorithm: Alg0022, LarTransverseExtension
- Running Algorithm: Alg0023, LarCrossgapsAssociation
- Running Algorithm: Alg0024, LarCrossgapsExtension
- Running Algorithm: Alg0025, LarOvershootSplitting
- Running Algorithm: Alg0026, LarBranchSplitting
- Running Algorithm: Alg0027, LarKinkSplitting
- Running Algorithm: Alg0028, LarTrackConsolidation
- Running Algorithm: Alg0030, LarClusteringParent
- Running Algorithm: Alg0031, LarTrackClusterCreation
- Running Algorithm: Alg0032, LarLayersSplitting
- Running Algorithm: Alg0033, LarLongitudinalAssociation
- Running Algorithm: Alg0034, LarTransverseAssociation
- Running Algorithm: Alg0035, LarLongitudinalExtension
- Running Algorithm: Alg0036, LarTransverseExtension
- Running Algorithm: Alg0037, LarCrossgapsAssociation
- Running Algorithm: Alg0038, LarCrossgapsExtension
- Running Algorithm: Alg0039, LarOvershootSplitting
- Running Algorithm: Alg0040, LarBranchSplitting
- Running Algorithm: Alg0041, LarKinkSplitting
- Running Algorithm: Alg0042, LarTrackConsolidation
- Running Algorithm: Alg0044, LarVisualization

The right pane shows a 3D view of the event with a 2D projection of the clusters in the U-view. A blue box highlights the 2D clusters in the U-view, and a blue arrow points to them from the text "2D clusters in the U view".

Looking at the reconstructed particles – 3D View

Turn off the hits, we've included them so you can always refer back to the inputs if you like

Expand the list of clusters

Try turning on and off some of the clusters so you can see what they correspond to in the viewer

Clusters are ordered by the total energy deposited

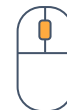
Viewer 1

Use Viewer 1 so we can check and uncheck boxes

After the initial 2D reconstruction you will probably find 2 main clusters (for the proton and muon) and many smaller clusters at kinks and bifurcations

Each colour corresponds to a different cluster

Command
Command (local):



Wheel up - zoom out
Wheel down - zoom in
Wheel press + drag - pan viewport



W - wireframe mode
R - return from wireframe mode

Looking at the other views

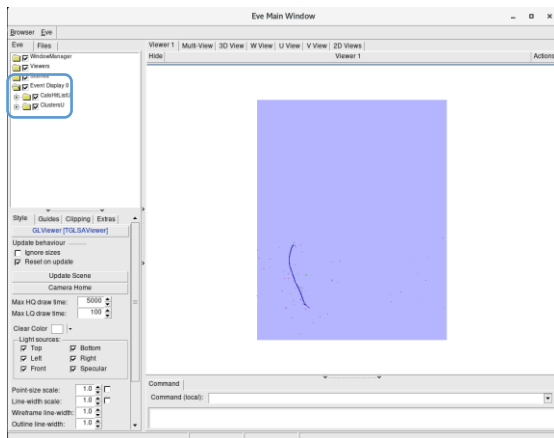
Click in the terminal window and press Return ↵ to visualize the other views

```

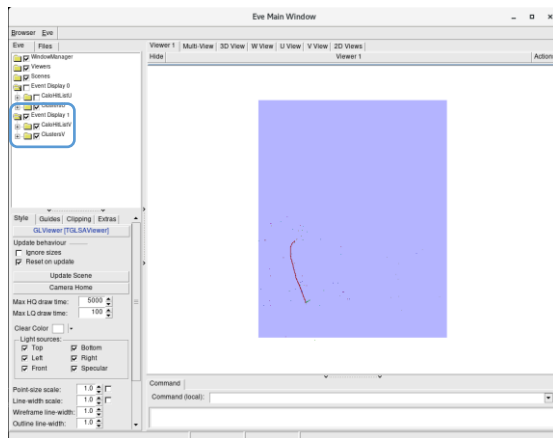
<algorithm type = "LArVisualMonitoring">
  <CaloHitListNames>CaloHitListU</CaloHitListNames>
  <ClusterListNames>ClustersU</ClusterListNames>
  <ShowDetector>true</ShowDetector>
</algorithm> Return ↵

<algorithm type = "LArVisualMonitoring">
  <CaloHitListNames>CaloHitListV</CaloHitListNames>
  <ClusterListNames>ClustersV</ClusterListNames>
  <ShowDetector>true</ShowDetector>
</algorithm> Return ↵

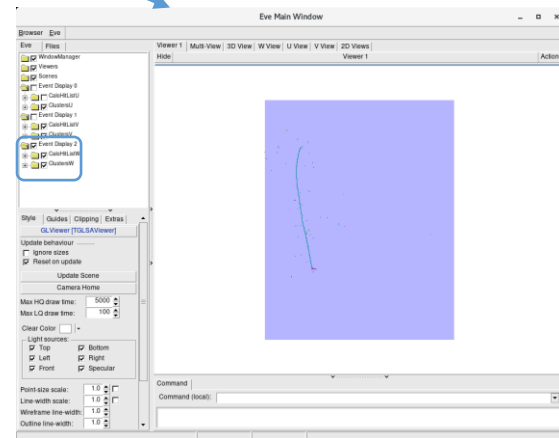
<algorithm type = "LArVisualMonitoring">
  <CaloHitListNames>CaloHitListW</CaloHitListNames>
  <ClusterListNames>ClustersW</ClusterListNames>
  <ShowDetector>true</ShowDetector>
</algorithm> Return ↵
  
```



U view clusters



V view clusters



W view clusters

3D vertex reconstruction

Add in some more visualizations

- Add to `MyPandoraSettings_Neutrino_Standard.xml` at the end of the `VertexAlgorithms` section

```

<!-- VertexAlgorithms -->
<algorithm type = "LArCandidateVertexCreation">
  <InputClusterListNames>ClustersU ClustersV ClustersW</InputClusterListNames>
  <OutputVertexListName>CandidateVertices3D</OutputVertexListName>
  <ReplaceCurrentVertexList>true</ReplaceCurrentVertexList>
  <EnableCrossingCandidates>>false</EnableCrossingCandidates>
</algorithm>
<algorithm type = "LArEnergyKickVertexSelection">
  <InputCaloHitListNames>CaloHitListU CaloHitListV CaloHitListW</InputCaloHitListNames>
  <InputClusterListNames>ClustersU ClustersV ClustersW</InputClusterListNames>
  <OutputVertexListName>NeutrinoVertices3D</OutputVertexListName>
  <ReplaceCurrentVertexList>true</ReplaceCurrentVertexList>
  <FeatureTools>
    <tool type = "LArEnergyKickFeature" />
    <tool type = "LArLocalAsymmetryFeature" />
  </FeatureTools>
</algorithm>
<algorithm type = "LArVertexSplitting">
  <InputClusterListNames>ClustersU ClustersV ClustersW</InputClusterListNames>
</algorithm>

<algorithm type = "LArVisualMonitoring">
  <ClusterListNames>ClustersV</ClusterListNames>
  <VertexListNames>CandidateVertices3D</VertexListNames>
  <ShowDetector>true</ShowDetector>
</algorithm>

<algorithm type = "LArVisualMonitoring">
  <ClusterListNames>ClustersW</ClusterListNames>
  <VertexListNames>NeutrinoVertices3D</VertexListNames>
  <ShowDetector>true</ShowDetector>
</algorithm>

<!-- ThreeDTrackAlgorithms -->
<algorithm type = "LArThreeDTransverseTracks">
  <InputClusterListNameU>ClustersU</InputClusterListNameU>
  <InputClusterListNameV>ClustersV</InputClusterListNameV>
  <InputClusterListNameW>ClustersW</InputClusterListNameW>

```

The `LArCandidateVertexCreation` algorithm creates a list of 3D candidate vertices at positions that project onto the ends of the existing 2D clusters

The `LArEnergyKickVertexSelection` algorithm selects the neutrino vertex from the candidates

Visualize the 3D candidate vertices along with the W-view clusters for comparison

Visualize the selected neutrino vertex along with the W-view clusters for comparison

Add the above lines just before the `ThreeDTrackAlgorithms` section

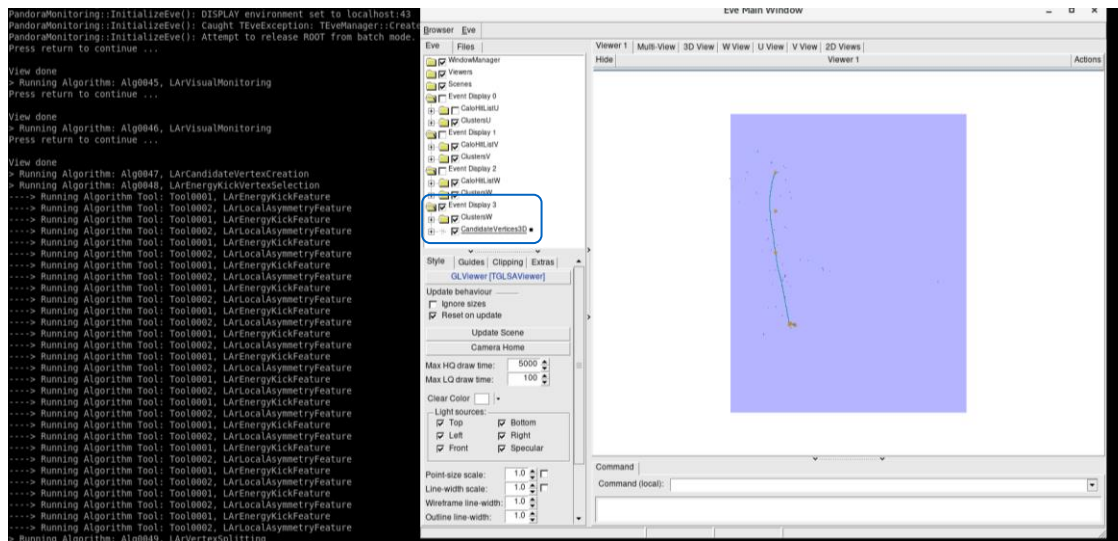
Run Pandora again!

- Run our FHiCl file again

```
$ cd $MRB_TOP/reco/work
```

```
$ lar -c event_display_driver.fcl -s reco_events.root -n 1
```

- After the event display has loaded press Return ↵ three times, to skip through our visualizations from part 1



Candidate 3D vertices vs W-view clusters

Expand the list of candidate vertices - there will be many!

Each vertex here is displayed as a yellow circle in the viewer

The screenshot shows the 'Eve Main Window' with a 'Viewer 1' window. The left sidebar shows a tree view of 'CandidateVertices3D' with a list of vertices. The main viewer area shows a 3D projection of these vertices as yellow circles, with a cyan line connecting two points. The text 'The yellow circles are the candidate vertices' points to the yellow circles in the viewer. The text 'In this viewer the 3D vertices are projected into the X-Z plane - this matches up with the W-view clusters' points to the cyan line in the viewer.

Viewer 1 | Multi-View | 3D View | W View | U View | V View | 2D Views | Viewer 1

Hide | Actions

The yellow circles are the candidate vertices

In this viewer the 3D vertices are projected into the X-Z plane - this matches up with the W-view clusters

Command | Command (local):



Wheel up - zoom out
Wheel down - zoom in
Wheel press + drag - pan viewport



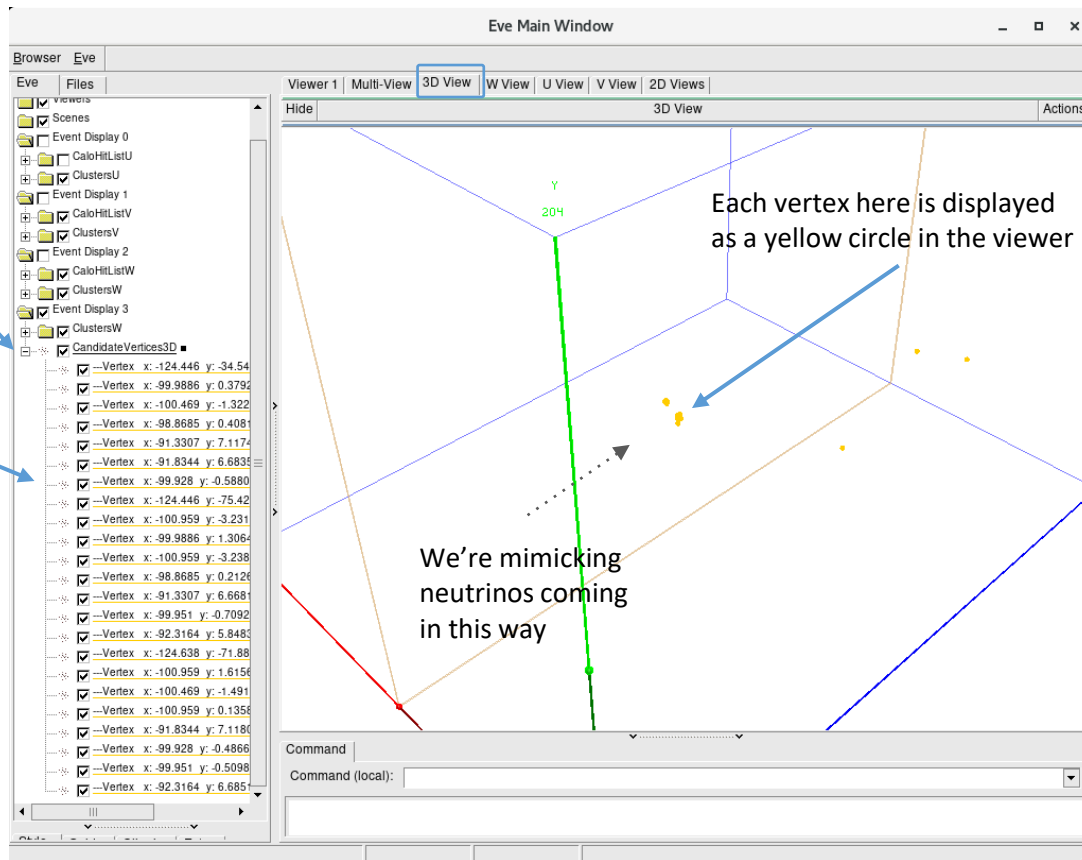
W - wireframe mode
R - return from wireframe mode

Candidate 3D vertices vs W-view clusters

Expand the list of candidate vertices - there will be many!

Each vertex here is displayed as a yellow circle in the viewer

When you are finished, press Return ↵ to move to the next display



Wheel up - zoom out
Wheel down - zoom in
Wheel press + drag - pan viewport



Left press + drag - rotate 3D view



W - wireframe mode
R - return from wireframe mode

Selected neutrino vertex vs W-view clusters

The screenshot shows the 'Eve Main Window' with a browser on the left and a viewer window in the center. The browser lists various objects, including 'NeutrinoVertices3D' and 'CandidateVertices3D'. A blue arrow points to the 'NeutrinoVertices3D' entry, which is expanded to show a single vertex: 'Vertex x:-100.469 y:-0.78358'. The viewer window displays a 3D scene with a blue wireframe structure and a red dot representing the selected neutrino vertex. A blue arrow points to this red dot with the text 'This is the selected neutrino vertex'. The viewer window has tabs for 'Multi-View', '3D View', 'W View', 'U View', 'V View', and '2D Views'. The 'W View' tab is selected. The command line at the bottom shows 'Command (local):'.

Expand the list of selected neutrino vertices - there should only be one

When you are finished, press Return ↵ to move to the next display



Wheel up - zoom out
Wheel down - zoom in
Wheel press + drag - pan viewport



W - wireframe mode
R - return from wireframe mode

3D track & shower reconstruction

Add in some more visualizations

- Add to `MyPandoraSettings_Neutrino_Standard.xml` at the end of the `TwoDMopUpAlgorithms` section

```

<!-- TwoDMopUpAlgorithms -->
<algorithm type = "LARBoundedClusterMopUp">
  <PfoListNames>ShowerParticles3D</PfoListNames>
  <DaughterListNames>ClustersU ClustersV ClustersW</DaughterListNames>
</algorithm>
<algorithm type = "LARConeClusterMopUp">
  <PfoListNames>ShowerParticles3D</PfoListNames>
  <DaughterListNames>ClustersU ClustersV ClustersW</DaughterListNames>
</algorithm>
<algorithm type = "LARNearbyClusterMopUp">
  <PfoListNames>ShowerParticles3D</PfoListNames>
  <DaughterListNames>ClustersU ClustersV ClustersW</DaughterListNames>
</algorithm>

<algorithm type = "LARVisualMonitoring">
  <PfoListNames>TrackParticles3D ShowerParticles3D</PfoListNames>
  <ShowDetector>true</ShowDetector>
</algorithm>

<!-- ThreeDHitAlgorithms -->
<algorithm type = "LARCUTPfoCharacterisation">
  <TrackPfoListName>TrackParticles3D</TrackPfoListName>
  <ShowerPfoListName>ShowerParticles3D</ShowerPfoListName>
  <PostBranchAddition>true</PostBranchAddition>
  <UseThreeDInformation>>false</UseThreeDInformation>
</algorithm>

```

Visualize the track-like and shower-like reconstructed particles

Add the above lines just before the `ThreeDHitAlgorithms` section

Run Pandora once again!

```

$ cd $MRB_TOP/reco/work
$ lar -c event_display_driver.fcl -s reco_events.root -n 1

```

- After the event display has loaded press Return ↵ five times, to skip through our visualizations from parts 1-2

Reconstructed track & shower-like particles

Expand all of the menus to see the clusters at this point and how they have been matched together into reconstructed particles (PFOs)

Here there are 3 track-like PFOs reconstructed

Hover over a cluster to see which view it belongs to - in this case it's the W view

In this event there are no shower-like particles to see

Viewer 1 | Multi-View | 3D View | W View | U View | V View | 2D Views | Actions

Each cluster is given a different colour

Remember, in Viewer 1 we display all views on top of each other

Recall how many tiny clusters we previously had! Now pandora has merged and split them to have zero or one cluster per view per PFO

2nd PFO is this tiny fragment

Each PFO has up to one cluster per view

Clusters are matched between views!

Command |
Command (local):



Wheel up - zoom out
Wheel down - zoom in
Wheel press + drag - pan viewport



W - wireframe mode
R - return from wireframe mode

When you are finished,
press Return ↵ to move
to the next display

3D hit reconstruction

Add in some more visualizations

- Add to `MyPandoraSettings_Neutrino_Standard.xml` at the end of the `ThreeDHitAlgorithms` section

```

<!-- ThreeDHitAlgorithms -->
<!-- HitCreationTools -->
<!-- HitCreationTools -->
<!-- HitCreationTools -->
<!-- HitCreationTools -->
</HitCreationTools>
</algorithm>
<algorithm type = "LARThreeDHitCreation">
  <InputPfoListName>ShowerParticles3D</InputPfoListName>
  <OutputCaloHitListName>ShowerCaloHits3D</OutputCaloHitListName>
  <OutputClusterListName>ShowerClusters3D</OutputClusterListName>
  <HitCreationTools>
    <!-- HitCreationTools -->
    <!-- HitCreationTools -->
    <!-- HitCreationTools -->
  </HitCreationTools>
</algorithm>
<algorithm type = "LARVisualMonitoring">
  <PfoListNames>TrackParticles3D ShowerParticles3D</PfoListNames>
  <ShowDetector>true</ShowDetector>
</algorithm>
<!-- ThreeDMopUpAlgorithms -->
<algorithm type = "LARSlidingConePfoMopUp">
  <InputPfoListNames>TrackParticles3D ShowerParticles3D</InputPfoListNames>
  <DaughterListNames>ClustersU ClustersV ClustersW TrackClusters3D ShowerClusters3D</Daugh
</algorithm>

```

Visualize the reconstructed particles again

Add the above lines just before the
`ThreeDMopUpAlgorithms` section

Run Pandora once again!

```

$ cd $MRB_TOP/reco/work
$ lar -c event_display_driver.fcl -s reco_events.root -n 1

```

- After the event display has loaded press Return ↵ six times, to skip through our visualizations from parts 1-3

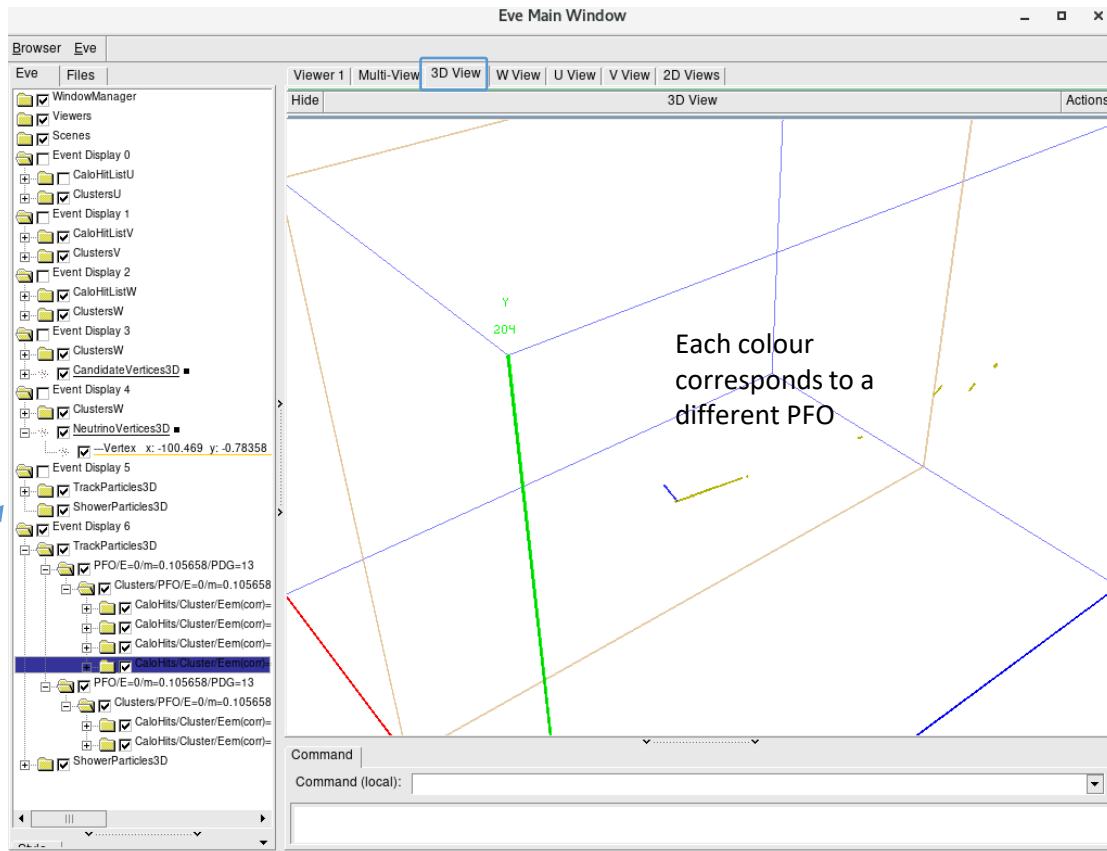
3D hits

When you are finished, press Return ↵ to close the event display

Expand the list of candidate vertices - there will be many!

Our PFOs now have a new cluster of 3D hits that we have just created

Since the last step, one of the particles has been classified as shower like - we use different 3D hit creation algorithms depending on the PFOs track-shower classification



Wheel up - zoom out
Wheel down - zoom in
Wheel press + drag - pan viewport



Left press + drag - rotate 3D view



W - wireframe mode
R - return from wireframe mode

Neutrino hierarchy reconstruction

Add in some more visualizations

- Add to `MyPandoraSettings_Neutrino_Standard.xml` at the end of the file

```

<OutputVertexListName>DaughterVertices3D</OutputVertexListName>
</algorithm>
<algorithm type = "LARNeutrinoProperties">
  <NeutrinoPfoListName>NeutrinoParticles3D</NeutrinoPfoListName>
</algorithm>

<!-- Track and shower building -->
<algorithm type = "LARTrackParticleBuilding">
  <PfoListName>TrackParticles3D</PfoListName>
  <VertexListName>DaughterVertices3D</VertexListName>
</algorithm>

<!-- Output list management -->
<algorithm type = "LARPostProcessing">
  <PfoListNames>NeutrinoParticles3D TrackParticles3D ShowerParticles3D</PfoListNames>
  <VertexListNames>NeutrinoVertices3D DaughterVertices3D CandidateVertices3D</VertexListNames>
  <ClusterListNames>ClustersU ClustersV ClustersW TrackClusters3D ShowerClusters3D</ClusterListNames>
  <CaloHitListNames>CaloHitListU CaloHitListV CaloHitListW CaloHitList2D</CaloHitListNames>
  <CurrentPfoListReplacement>NeutrinoParticles3D</CurrentPfoListReplacement>
</algorithm>

<algorithm type = "LARVisualMonitoring">
  <ShowCurrentPfos>true</ShowCurrentPfos>
  <ShowDetector>true</ShowDetector>
</algorithm>
</pandora>

```

Visualize the final reconstructed particles

Run Pandora once again!

```

$ cd $MRB_TOP/reco/work
$ lar -c event_display_driver.fcl -s reco_events.root -n 1

```

- After the event display has loaded press Return ↵ seven times, to skip through parts 1-4

The final outcome

Expand all of the menus again to see what we now have

The PFOs are now arranged in a hierarchy! The top-level PFO has PDG code = 14 $\Rightarrow \nu\mu$

The neutrino PFO has 3 daughter PFOs which each have clusters of 2D & 3D hits

The PFOs have been classified once more as track-like (assigned PDG 13) or shower-like (assigned PDG 11)

Every PFO has a vertex this is the reconstructed start position

The screenshot shows the 'Eve Main Window' interface. On the left is a 'Browser' pane with a tree view of objects. The tree view shows a hierarchy starting with 'Event Display 7' (PDG=14), which has three daughter PFOs. Each daughter PFO has clusters of 2D and 3D hits. The main area contains three views: '3D View', '2D U View', and '2D V View'. The '3D View' shows a 3D plot with a vertical green line and other colored lines. The '2D U View' and '2D V View' show 2D plots with axes labeled 'X' and 'Y' and values like '-202' and '202'. A 'Command' field is at the bottom.

When you are finished, press Return \leftarrow to close the event display

Secondary particles - a different event

Please note, this is now the final outcome of a different event

In this event, the muon undergoes a secondary interaction
In this event, the muon undergoes a secondary interaction

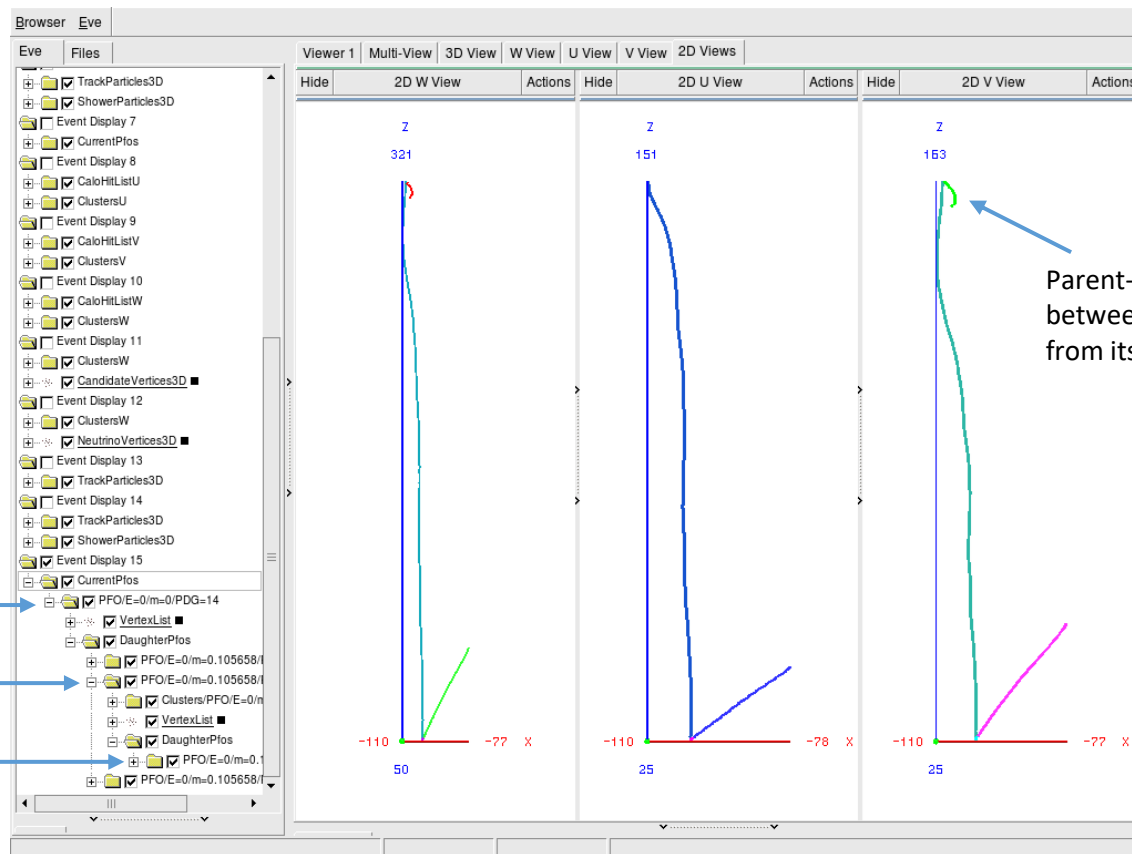
Neutrino PFO

Muon PFO

(Daughter of neutrino)

Electron PFO

(Daughter of muon)



Parent-daughter link is made between muon and the particle from its secondary interaction

Got spare time?

Run your FHiCL file again over multiple events
Do you understand what Pandora is doing in each of the steps?