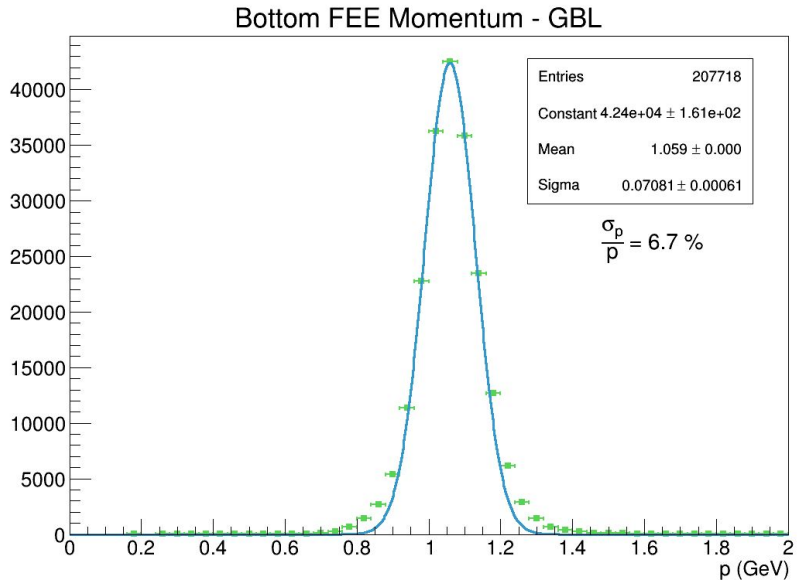
What's the Deal With All These Tracking Strategies?

Omar Moreno
Santa Cruz Institute for Particle Physics
University of California, Santa Cruz
omoreno1@ucsc.edu

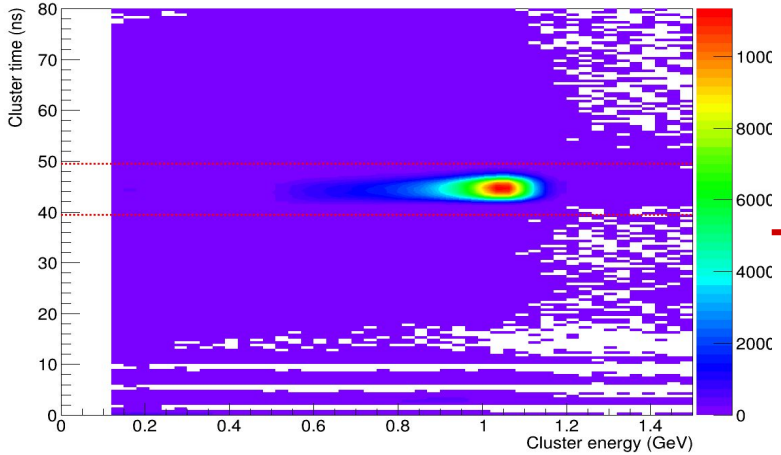
Heavy Photon Search Collaboration Meeting
October 26-28, 2015

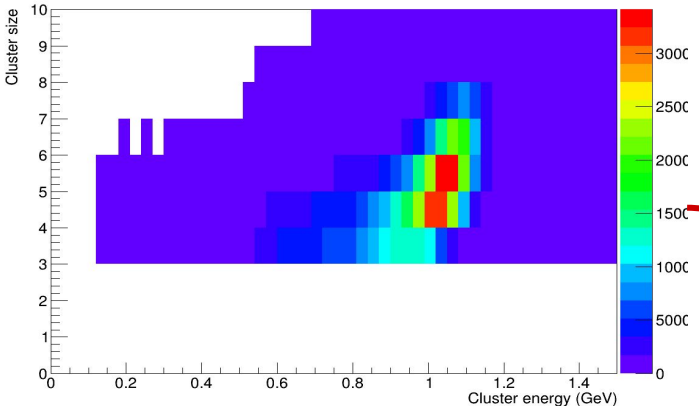
Some Preliminaries

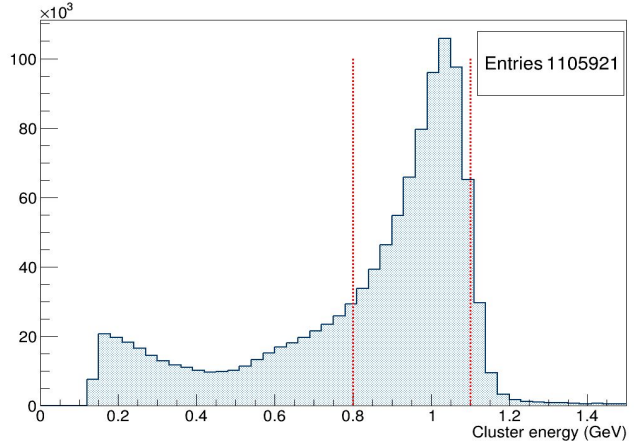
- All data plots shown in the slides that follow were generated using run 5772 using various detectors and different passes.
- Most of the plots were generated using the following full energy electron selection:
  - Require an Ecal cluster energy to be > .85 GeV and < 1.1 GeV
  - Require the cluster time to fall between 39.5 and 49.5 ns
  - Require the cluster size to be greater than 3
  - Require the cluster seed energy > .4 GeV
- Calculate efficiencies by taking the ratio of Ecal clusters that have a track matched to it to those that pass the FEE cuts.

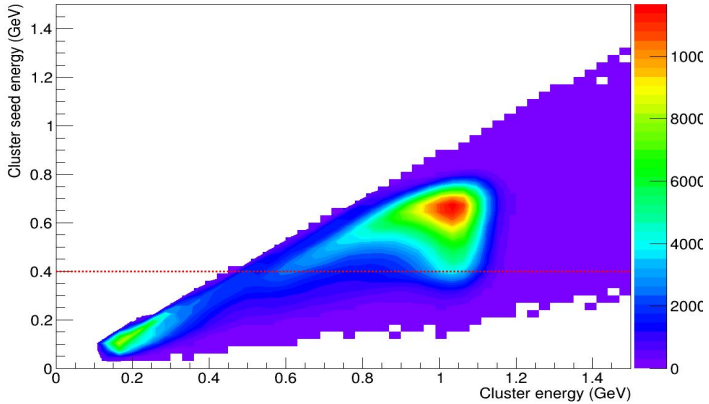


Full Energy Electron Selection - Graphically









Quick Overview of Track Finding & Fitting

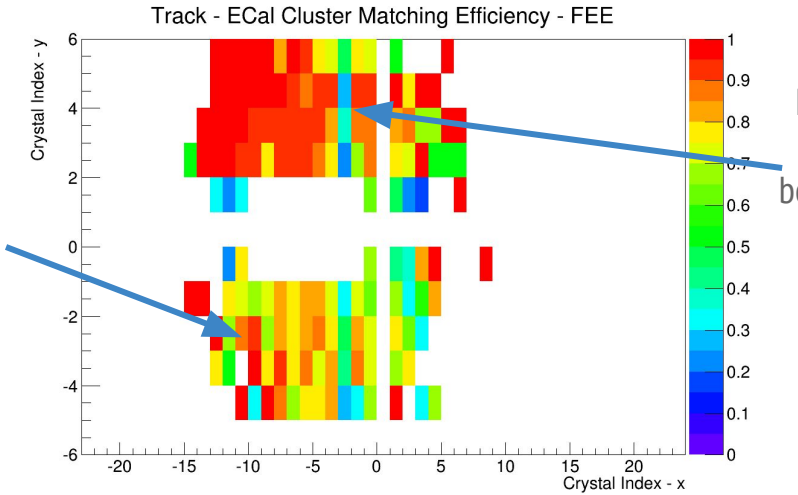
- Track finding and fitting is done in steps following a specified "tracking strategy"
  - The tracking strategy specifies which 3 layers are used to create seed tracks, which layer to use to confirm the track and which layers used to extend the track
  - The tracking strategy also specifies cuts on track fit chi^2, z0, d0, p_t as well as the minimum number of hits a track can have
- 3-hit seeds are created by looping over all 3D stereo hits (HelicalTrackHits) in the specified seed layers ⟶ The seed is then required to pass all cuts specified in the strategy
- The best hit from the confirm layer is then added to the seed track and the chi^2 is checked again
- Finally, the extended layers are added and the track is required to pass all cuts and have the minimum number of hits
- After all tracks have been found, make sure that none of them have more than a single shared hit

```
<MinPT>0.100</MinPT>
<MinHits>5</MinHits>
<MinConfirm>1</MinConfirm>
<MaxDCA>4.0</MaxDCA>
<MaxZ0>4.0</MaxZ0>
<MaxChisq>100.0</MaxChisq>   ← Was 25
```

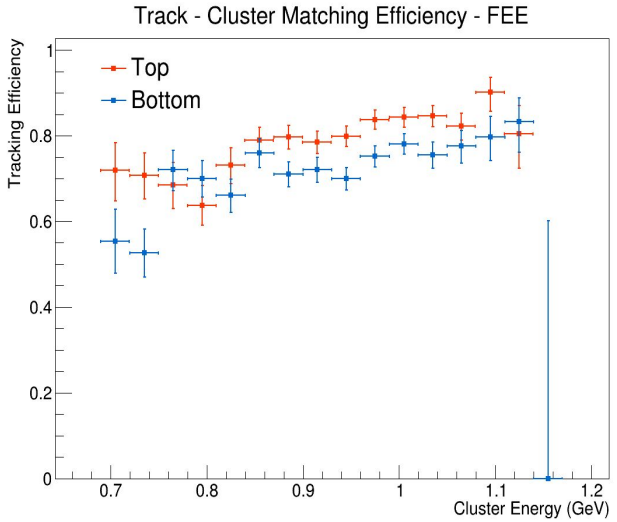
Just in Case You Forgot …

- Studying our ability to match tracks to clusters revealed an asymmetry in the track cluster matching efficiencies between top and bottom detector volumes
- At the time, only a single tracking strategy that used layers 123 to seed a track, extended it using layer 4 and confirmed it using layers 5 and 6 was being used
- Expected that the tracking strategy wouldn't find all of the tracks, but a large asymmetry between top and bottom? That was indicating that something else was wrong …

lower track-cluster matching efficiency



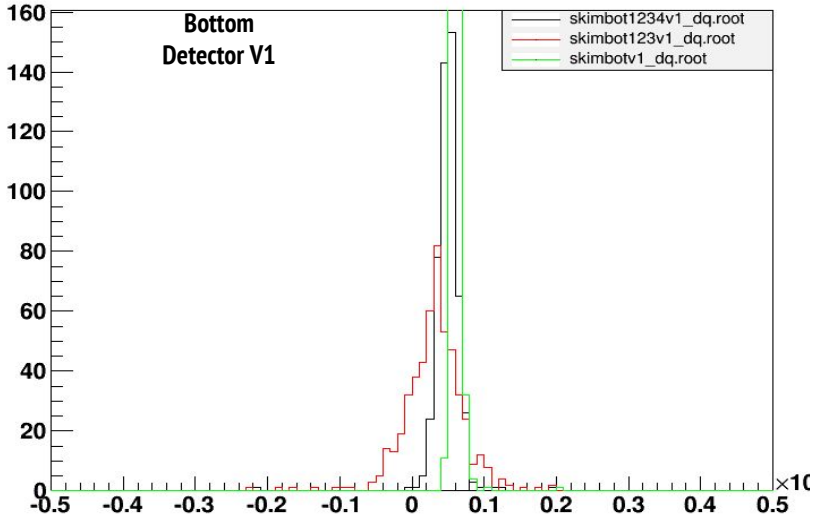
Low efficiency due to gap between layer 4-6 sensors

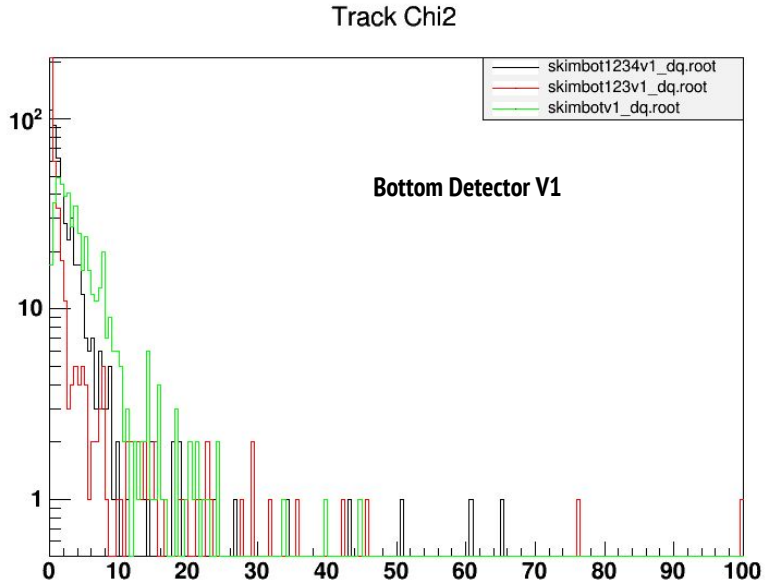


So What Was Actually Going On?

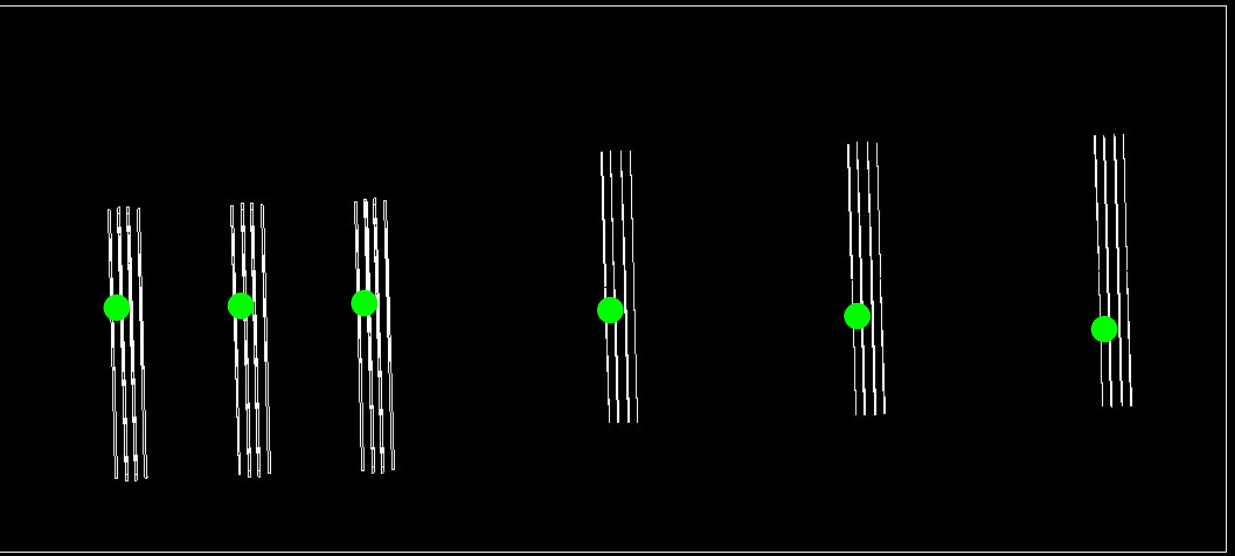
- More tracks on the bottom were failing the chi^2 cut after the confirm stage
  - When using the v1 detector, it seemed that after the seed stage, bottom tracks often had positron curvature; as hits were added to the track, the curvature converged to a reasonable value
- The initial seed fit of tracks that initially had positron curvature were being pulled when hits were added at the confirm stage ⟶ The result, a bad chi^2 and tracks not passing the strategy cut

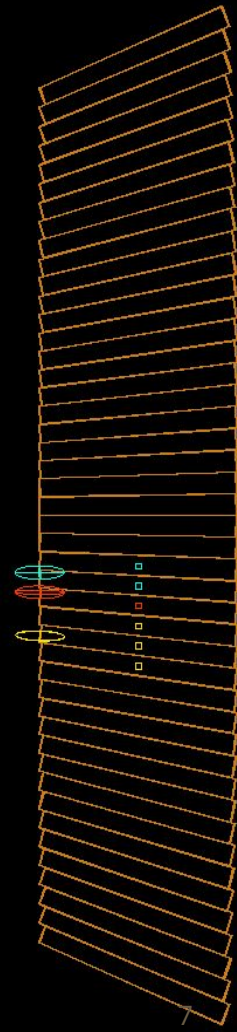
UEMURA





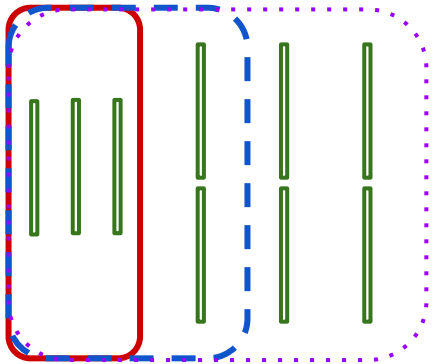
The leading theory is that the worse chi^2 on the bottom was being caused by misalignments ...



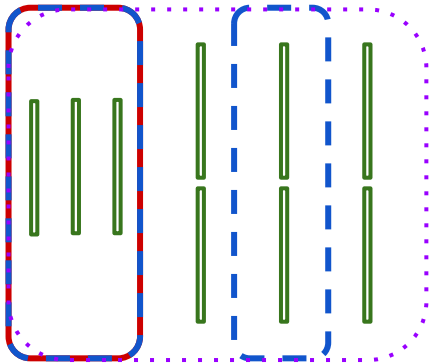


So What Strategies are we Using?

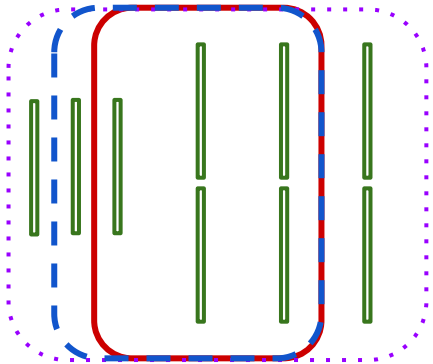
- Currently using four strategies:



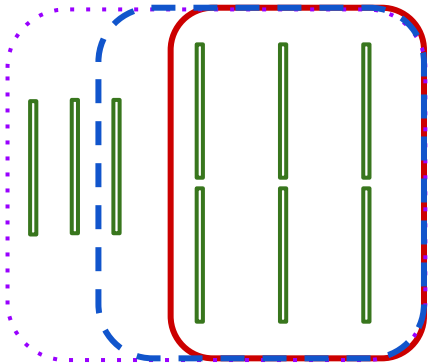
Seed Layers 1-3
Confirm Layer 4
Extend Layer 5/6



Seed Layers 1-3
Confirm Layer 5
Extend Layer 4/6



Seed Layers 3-5
Confirm Layer 2
Extend Layer 1/6



Seed Layers 4-6
Confirm Layer 3
Extend Layer 2/1

Can also have additional strategies with different seed layers e.g. s234, or use different confirm or extend layers but none of those strategies have been evaluated.

Performance

- Evaluated each of the strategies using a sample that pass the FEE cuts with additional cuts
- Require the FEE cluster to be within a 1x1 cm^2 fiducial region
  - -50 mm < cluster x < -40 mm && 50 mm < abs(cluster y) < 60 mm (row 3 of the Ecal)
- Require that half of the SVT has exactly one stereo hit per layer and no stereo hits on the other half

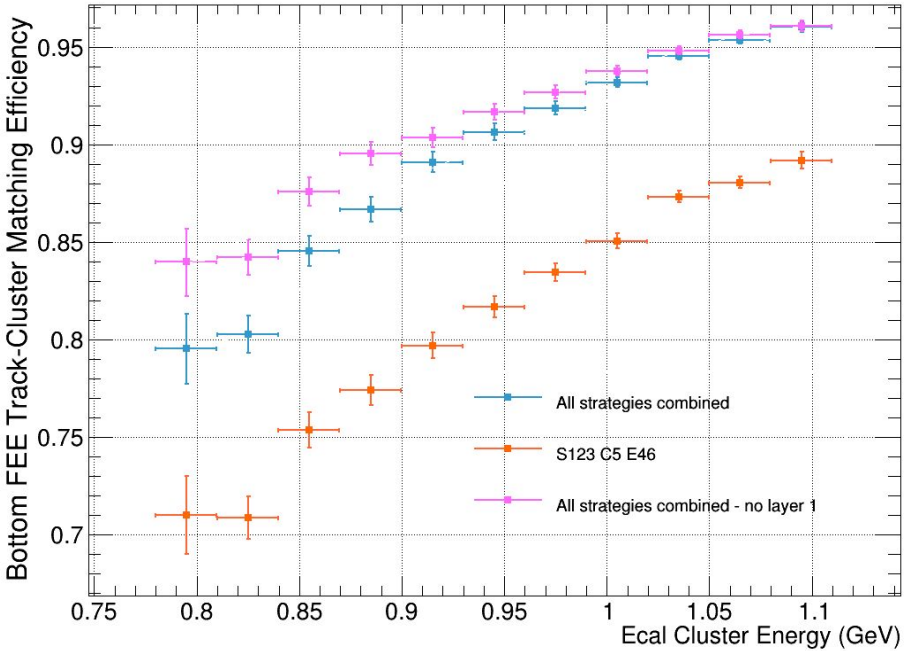
Using run 5772, the total number of top (bottom) events passing this selection is 522 (522) out of ~12 M

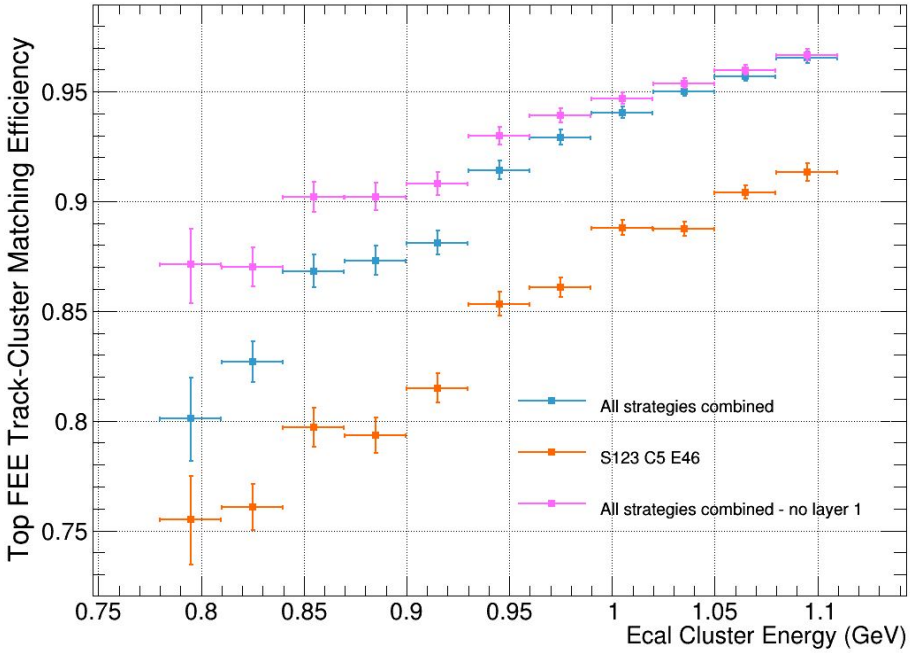
Tracking Strategy	Total top tracks found	Tracking Efficiency (%)	Total bottom tracks found	Tracking Efficiency (%)
s456 c3 e21	519	99.43	517	99.04
s345 c2 e16	518	99.23	519	99.43
s123 c5 e46	513	98.28	472	90.42
s123 c4 e56	512	98.08	473	90.61

It's clear that the tracking strategy we were initially using performed poorly. Loosening the Chi^2 cut helps, but using multiple strategies is the best approach.

Track-Cluster Matching Efficiency

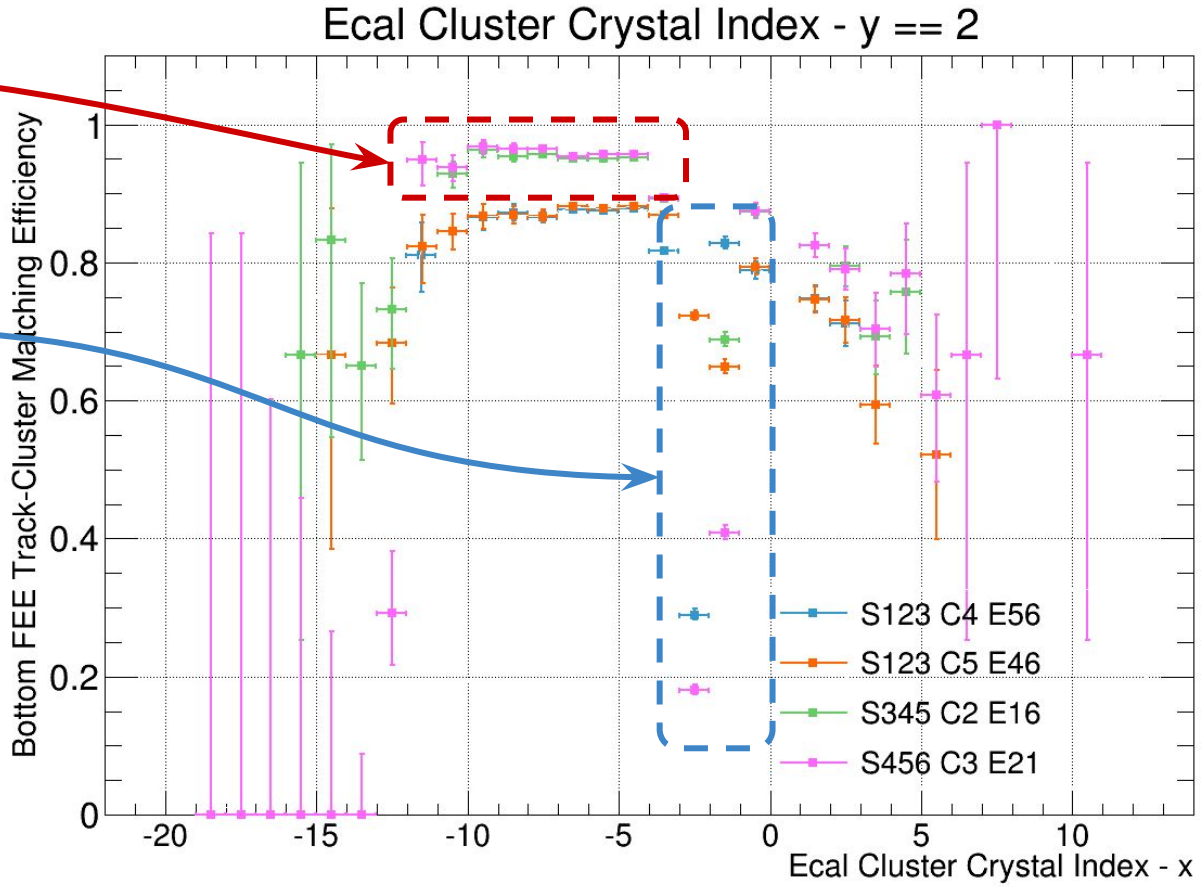
- Combine all tracks from all strategies into a single collection
  - Remove duplicate tracks i.e. tracks found by different strategies composed of the same hits
  - Remove partial tracks i.e. tracks found by different strategies whose hits are a subset of another track.
  - Check how many clusters that pass full energy electron cuts actually have tracks associated with them



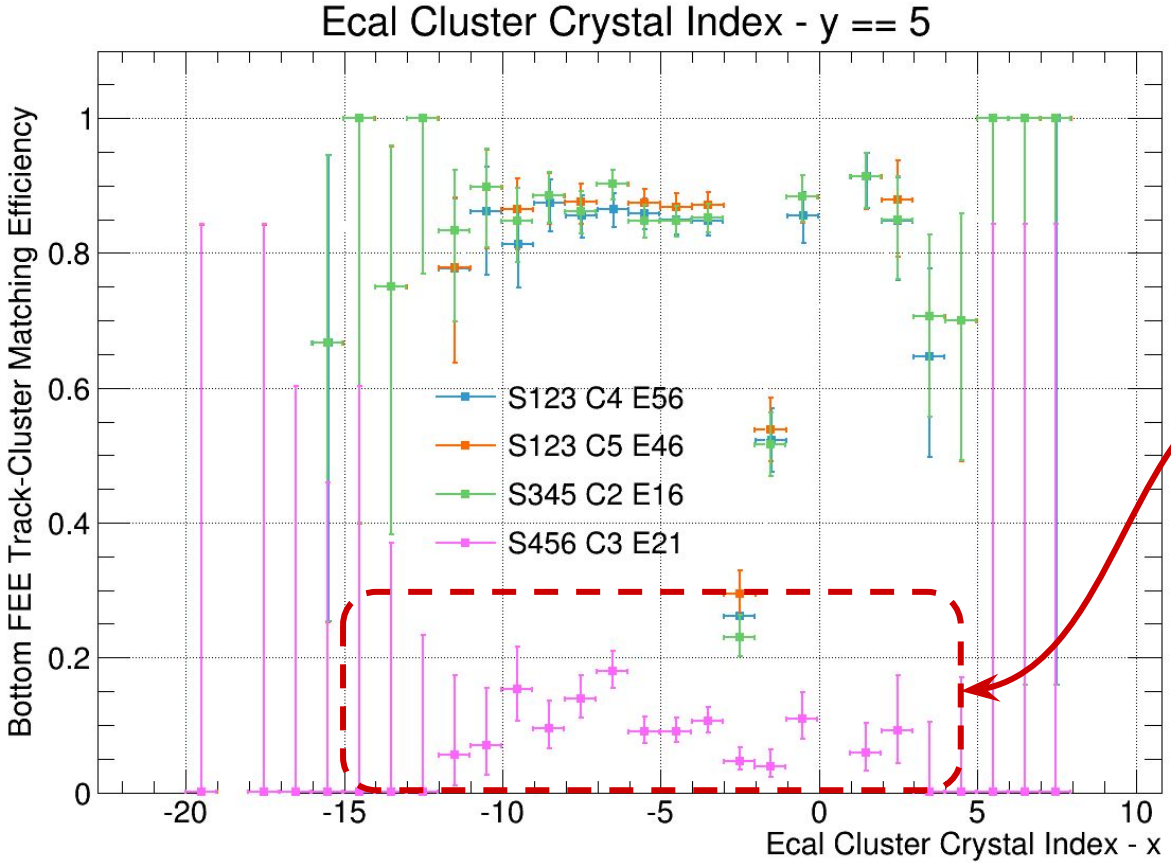


Track-Cluster Matching Efficiency

S345 and **S456** do better overall, but **S123 C5** and **S123 C4** have better gap coverage. This highlights the need for using multiple strategies.



Track-Cluster Matching Efficiency

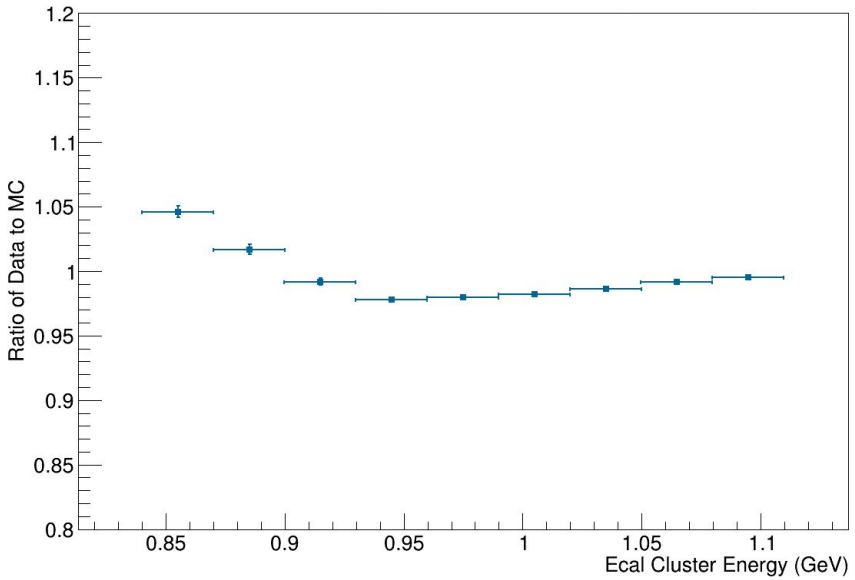


s456 performs terribly because requiring a hit in layer 6 limits your acceptance. Again, this highlights the need for multiple strategies.

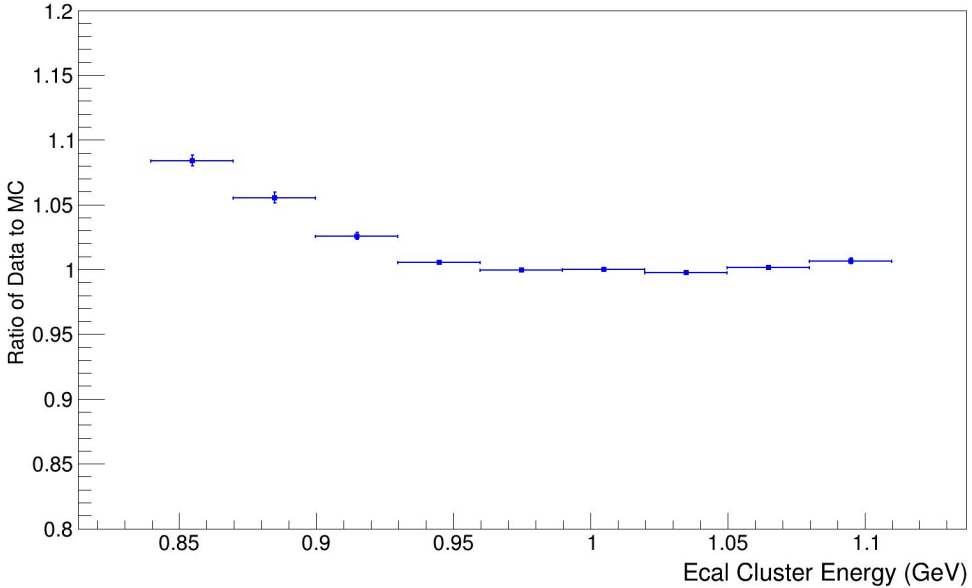
Track-Cluster Matching Efficiency - Data/MC

- Used pass 3, beam-tri, single1 Monte Carlo and calculated the track-cluster matching efficiency data/MC ratio
- Overall, data and MC agree to within 5% for higher Ecal cluster energies
- Anomalies are still present at lower energies but it's likely due to more fakes passing FEE cuts in data.

Bottom FEE Track-Cluster Efficiency Ratio



Top FEE Track-Cluster Efficiency Ratio



How do I know what Strategies Have Been Used?

- Each track objects (LCIO `Track` and `SvtTrack`) has a 32 bit int track type associated with it which encodes the tracking strategies that are used

Encoding position defined in the enum class `org.hps.recon.tracking.StrategyType`

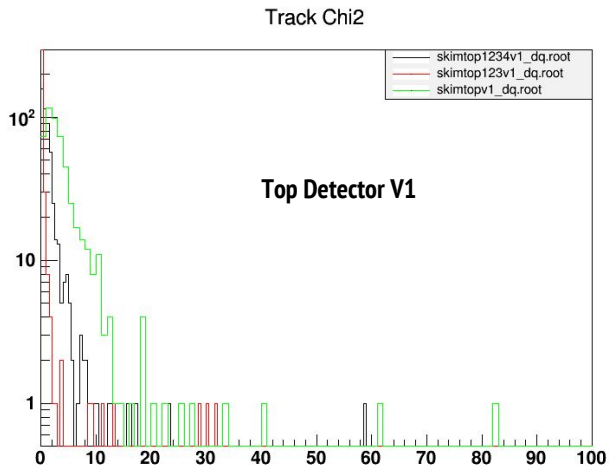
- A user can check if a defined strategy found a given track by using the utility class `org.hps.recon.tracking.TrackType` as follows

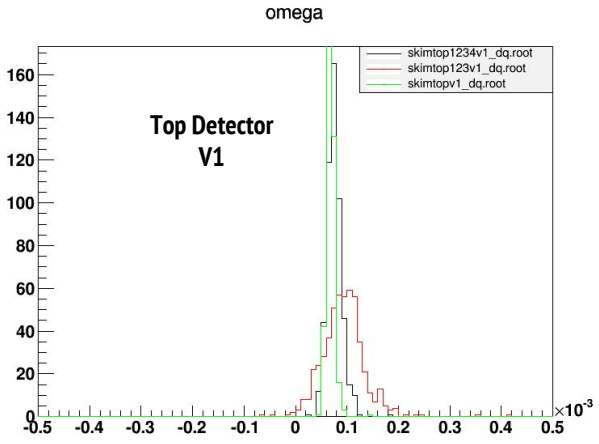
```
if (track.GetType()  == TrackType.getType(StrategyType.<strategy>))
```

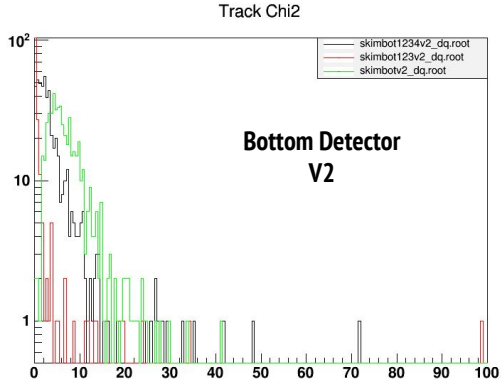
- At the DST level, a user can check if a tracking strategy found a given track by using the utility class TrackType as follows:

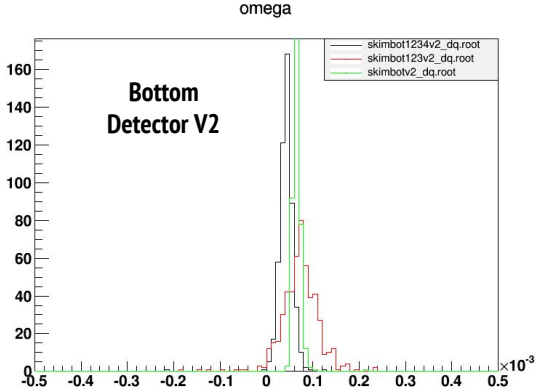
```
if (TrackType::foundByStrategy(track, StrategyType.<strategy>))
```

Backup Slides


Track Chi2
skimtop1234v1_dq.root
skimtop123v1_dq.root
skimtopv1_dq.root
Top Detector V1


omega
skimtop1234v1_dq.root
skimtop123v1_dq.root
skimtopv1_dq.root
Top Detector V1
×10-3


Track Chi2
skimbot1234v2_dq.root
skimbot123v2_dq.root
skimbotv2_dq.root
Bottom Detector V2


omega
skimbot1234v2_dq.root
skimbot123v2_dq.root
skimbotv2_dq.root
Bottom Detector V2
×10-3