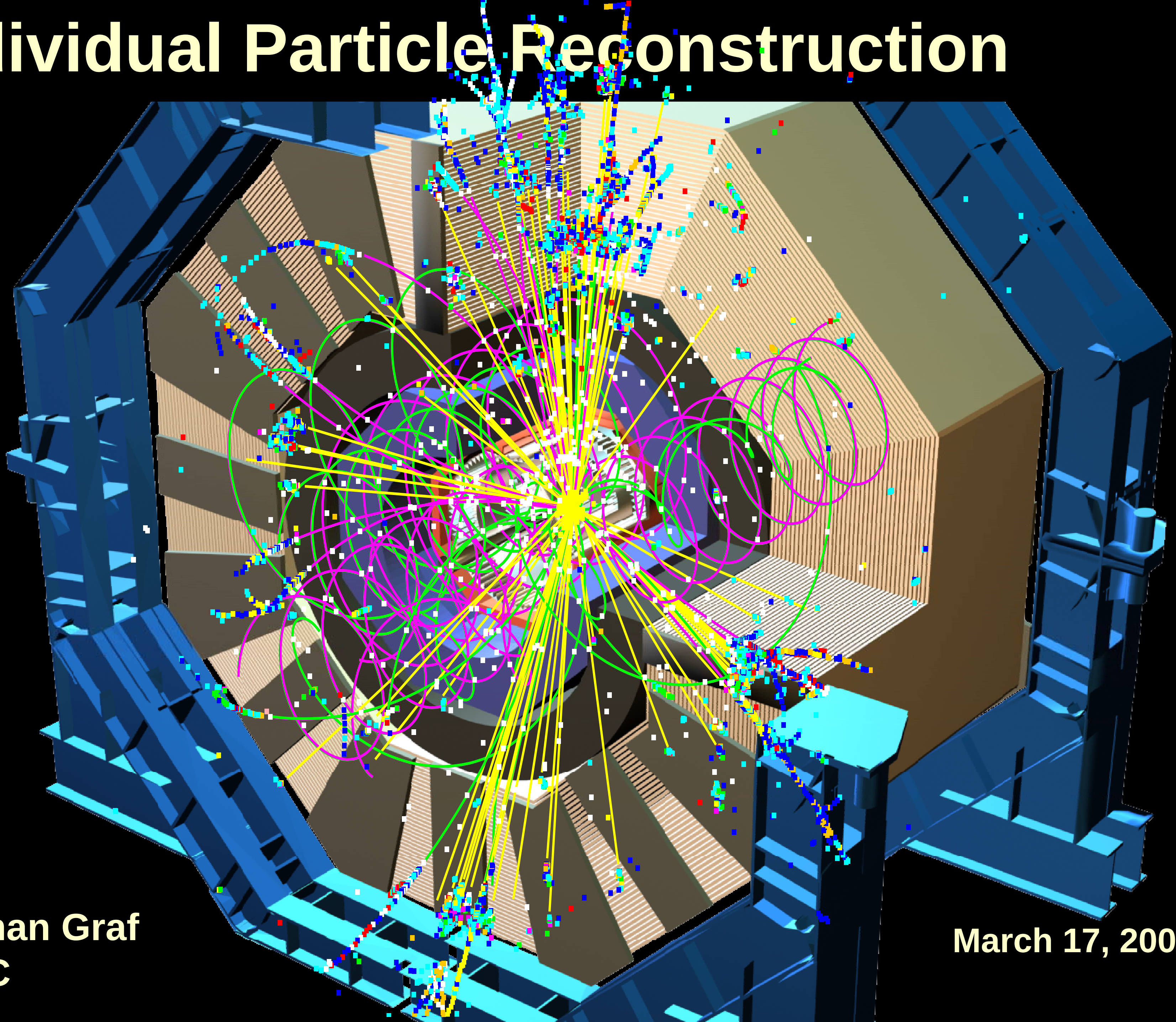
Individual Particle Reconstruction



Norman Graf
SLAC

March 17, 2005

Goal

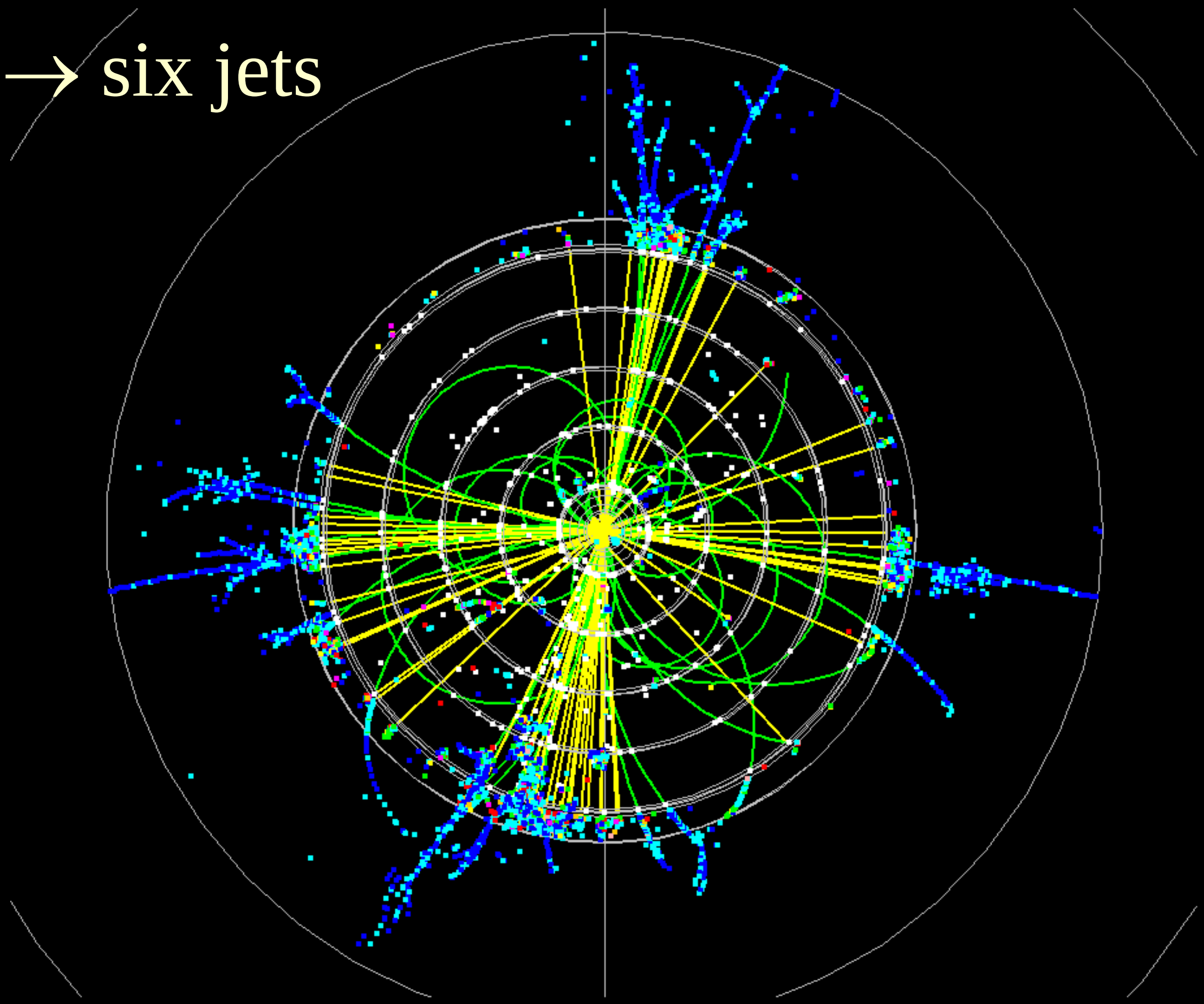
- The aim is to reconstruct individual particles in the detector with high efficiency and purity.
- Recognizing individual showers in the calorimeter is the key to achieving high di-jet mass resolution.
- High segmentation favored over compensation.
- Loss of intrinsic calorimeter energy resolution is more than offset by the gain in measuring charged particle momenta.

Calorimeter Segmentation

- Highly segmented calorimeters constructed of materials which induce compact shower size are necessary.
- Si-W default for electromagnetic calorimeter.
- Tungsten also being investigated for HCal
  - more compact design reduces cost of coil
- Need high segmentation to minimize the number of cells receiving energy deposits from more than one initial particle.

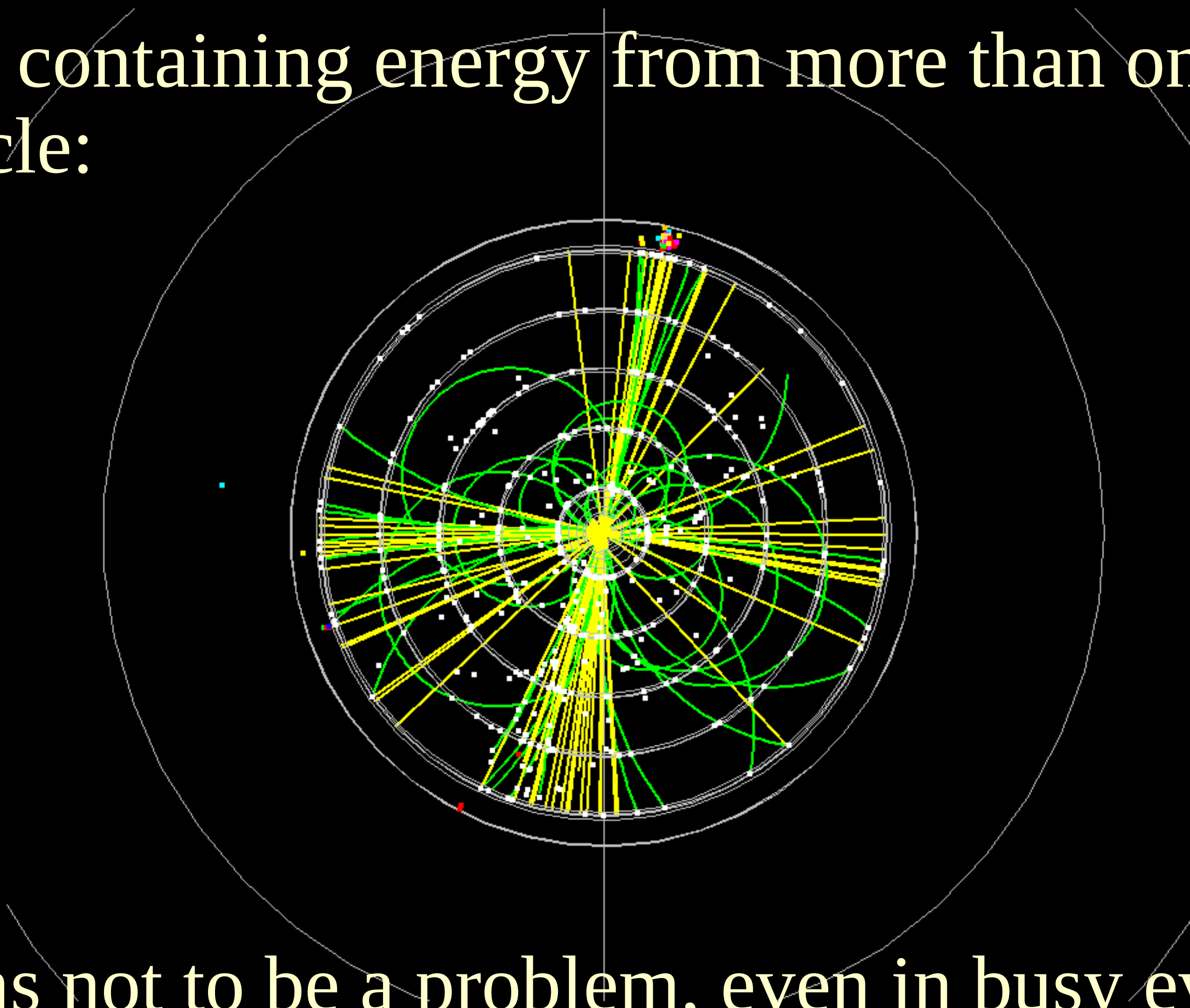
Occupancy Event Display

- ttbar → six jets



Occupancy Event Display

- Cells containing energy from more than one MC particle:



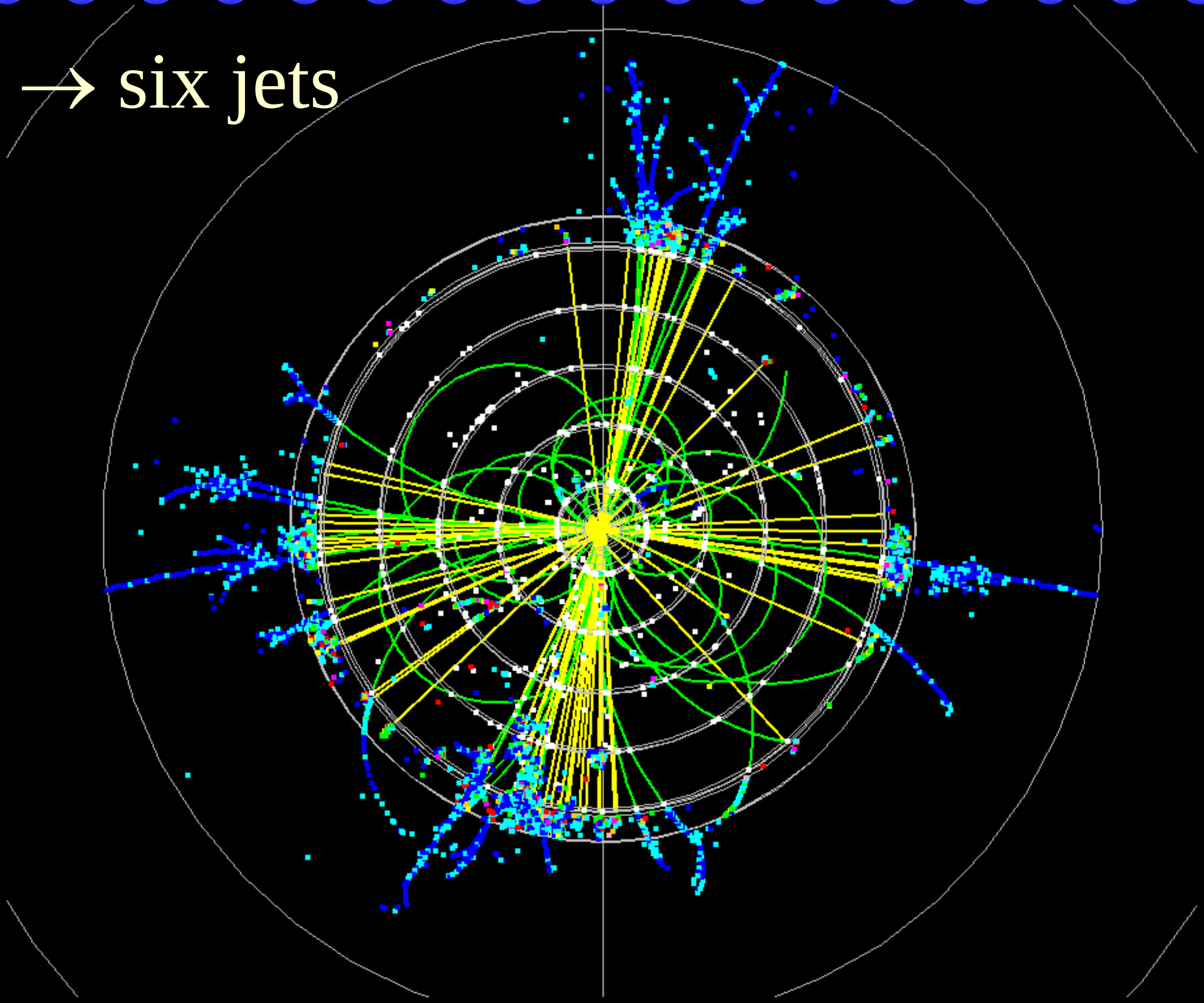
- Seems not to be a problem, even in busy events.

Clustering

- Two clustering algorithms available in current code release
  - “Nearest”-Neighbor, with user-defined domains available in longitudinal and two transverse dimensions.
    - (1,0,0) is simplest MIP-cluster finder.
  - Fixed-Cone algorithm (θ,φ)
    - fast, seed-based, but iterative centering
    - cluster splitting for overlapping cones.
- Cluster interface defined, so additional clustering algorithms are easily accommodated.

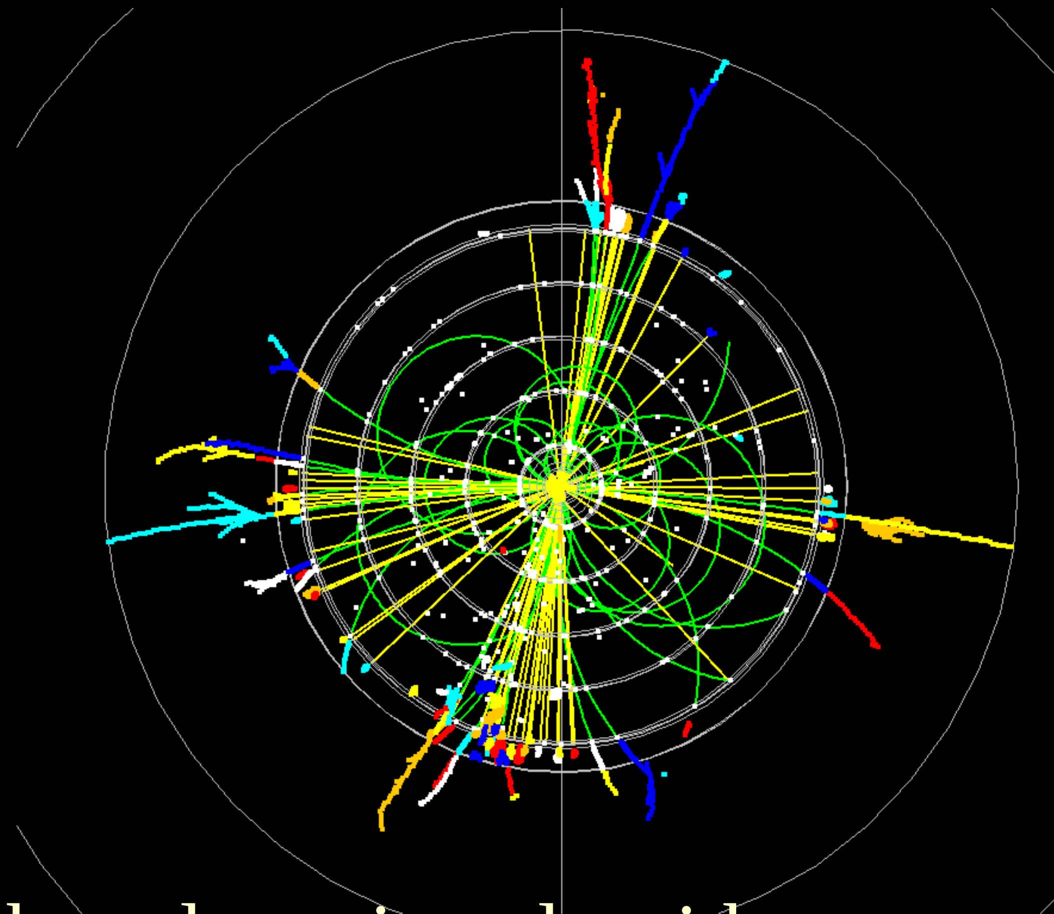
SimpleClusterBuilder

- ttbar → six jets



SimpleClusterBuilder

- Clusters color coded:



- (1,1,1) Nearest-Neighbor clustering algorithm performs quite well in the silicon-tungsten detector.

Track Finding and Fitting

- Nick Sinev has released standalone pattern recognition code for the 2D Barrel VXD hits.
  - High efficiency, even in presence of backgrounds.
  - Efficient at low momentum.
  - Propagates tracks into Central Tracker to pick up φ hits
- Conformal-mapping pattern recognition also available. Fast, but not yet tuned (97% vs 99+%).
- Work also ongoing to find MIP stubs in Cal and propagate inwards (Kansas State, Iowa).

Strategy I

- Begin by finding and fitting tracks.
  - (In following plots, used FastMC to smear tracks to decouple the two tasks, viz. I *assume* highly efficient track finding.)
- Cluster the calorimeter cells in in EM, HAD & MUON independently using SimpleClusterBuilder.
  - EM $\rightarrow$ photons & electrons +muon MIPs +others
  - HAD $\rightarrow$ hadrons + muon MIPS
  - MUON $\rightarrow$ muon MIPS (+punchthrough)

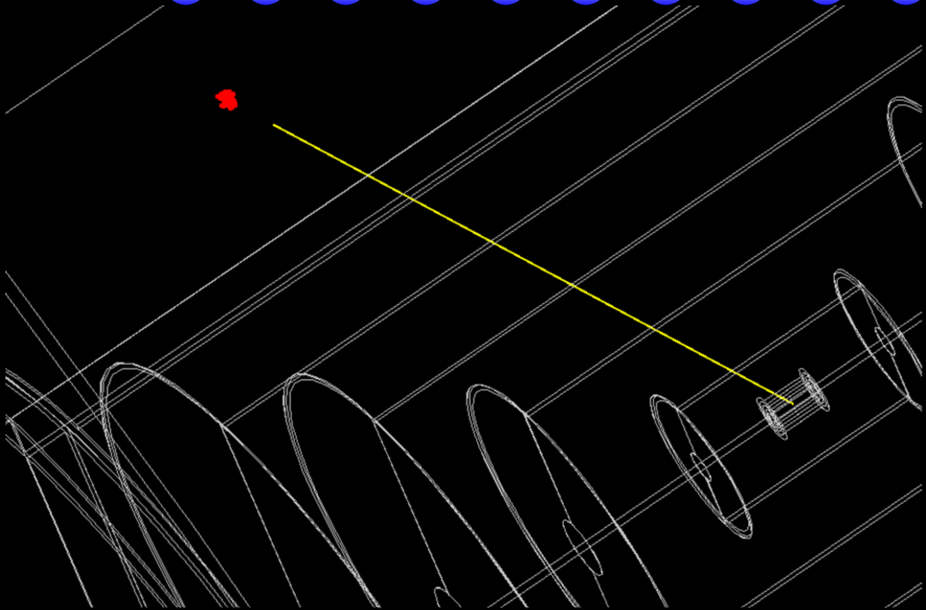
Strategy II

- Propagate tracks through the calorimeters and associate clusters to the track if trajectory intersects calorimeter cell in cluster.
  - Tracks associated to EM clusters and good match between cluster energy & track momentum become electron candidates.
  - Tracks associated with clusters in EM, HAD and MUON become muon candidates.
  - Remainder become pion candidates.
- Remove clusters from the event list.

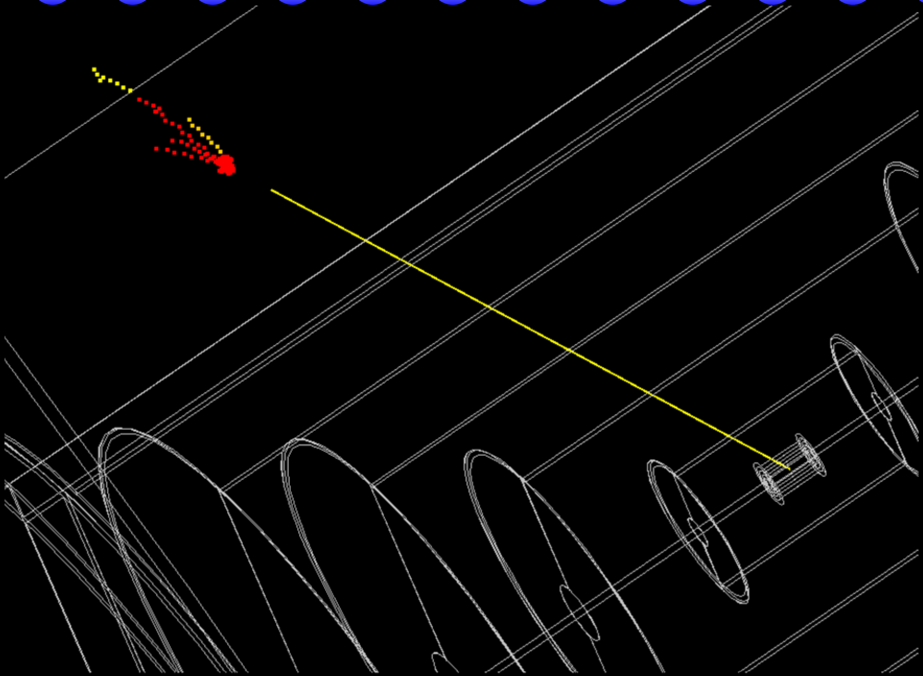
Neutral Clusters

- EM Clusters unassociated with a track are photon candidates.
  - Calculate chi-squared for longitudinal shower shape.
  - Calculate shower width.
  - Clusters passing cuts become photon candidates.
  - Remove photon candidate clusters.
- Unassociated EM neutral clusters failing photon cut + HAD clusters are clustered using fixed cone algorithm.
- These become neutron (K^0_L) candidates.

Single Neutron Event



Single neutron EM cluster



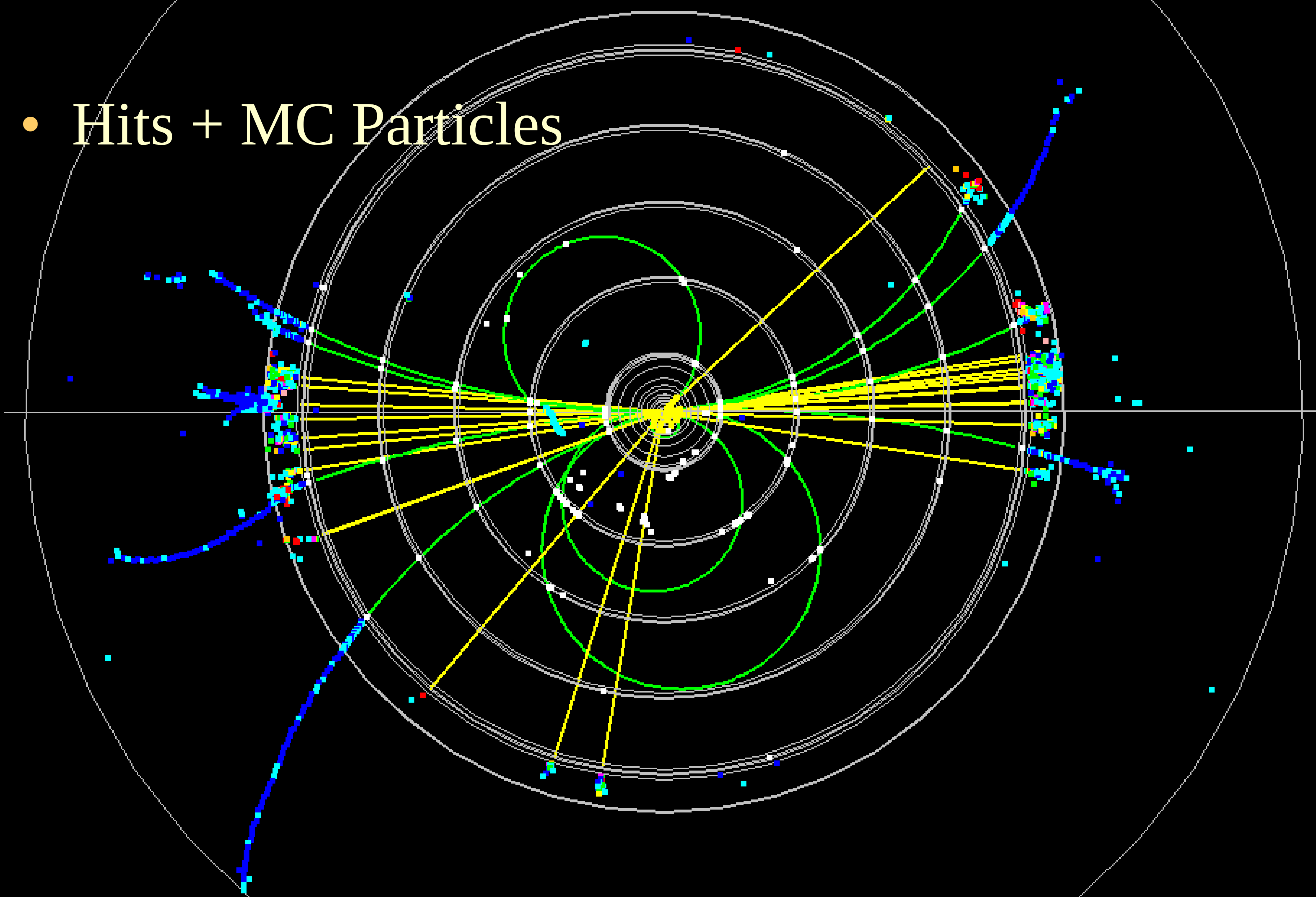
EM cluster + hadron clusters, combined using a fixed-cone clustering algorithm.

ReconstructedParticles

- These ReconstructedParticles (electron, photon, pion, muon, neutron) are added back to the event.
- Can easily sum up event energy in ZPole events.
  - Width of resulting distribution is direct measure of resolution, since events generated at 91GeV.
- Run jet-finder on RP four vectors, calculate dijet invariant mass.
- Make lots of plots matching RP-MC.

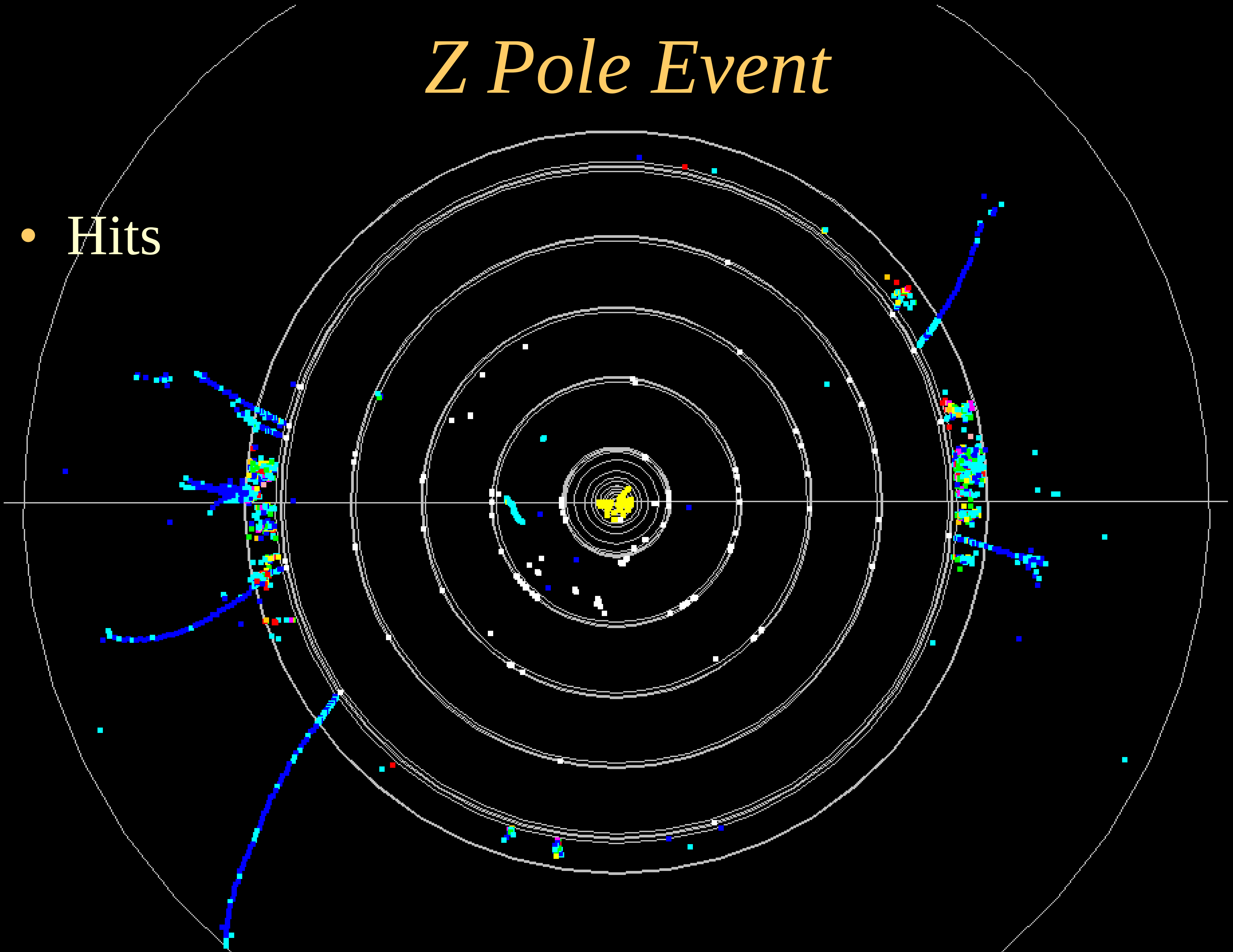
Z Pole Event

- Hits + MC Particles



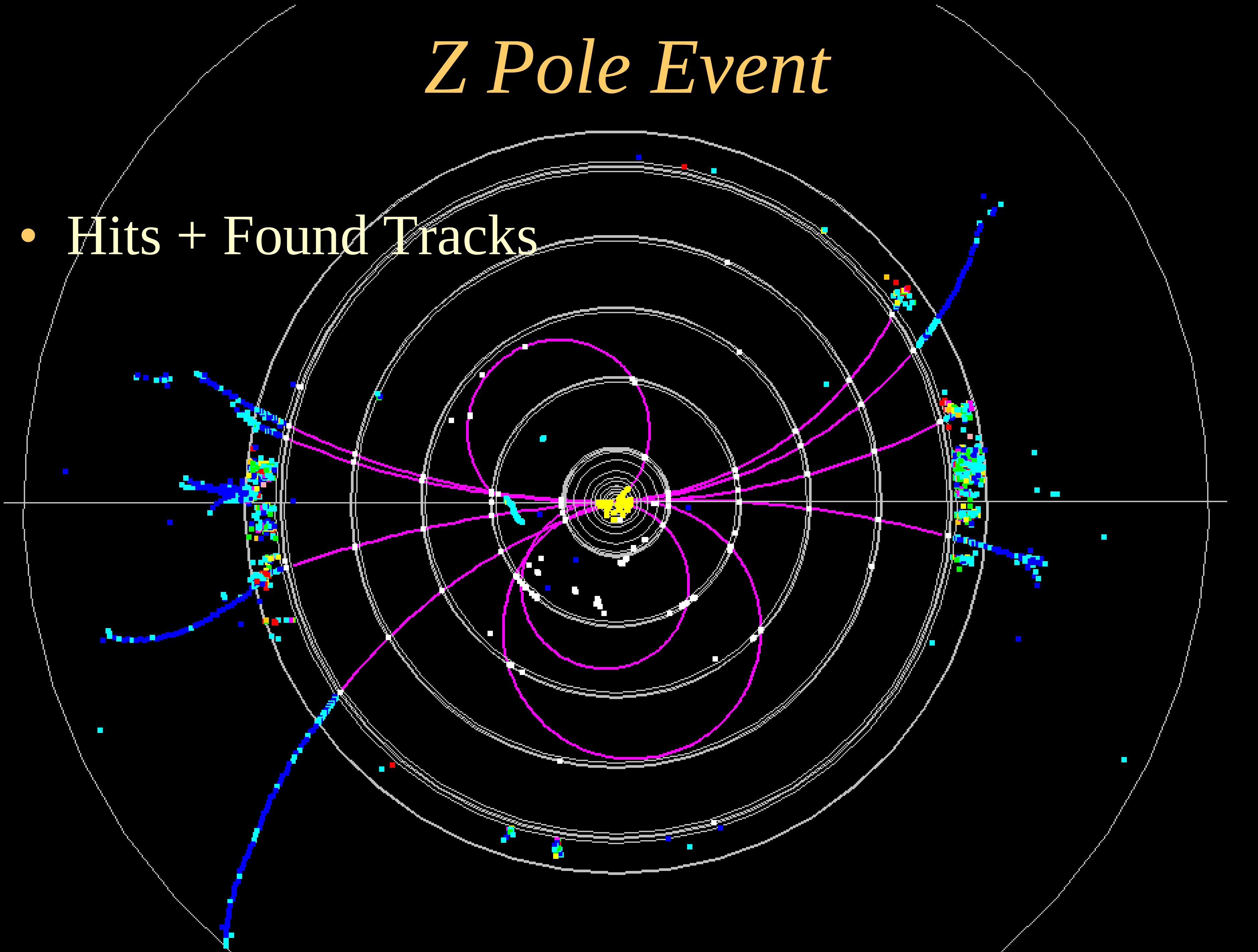
Z Pole Event

- Hits



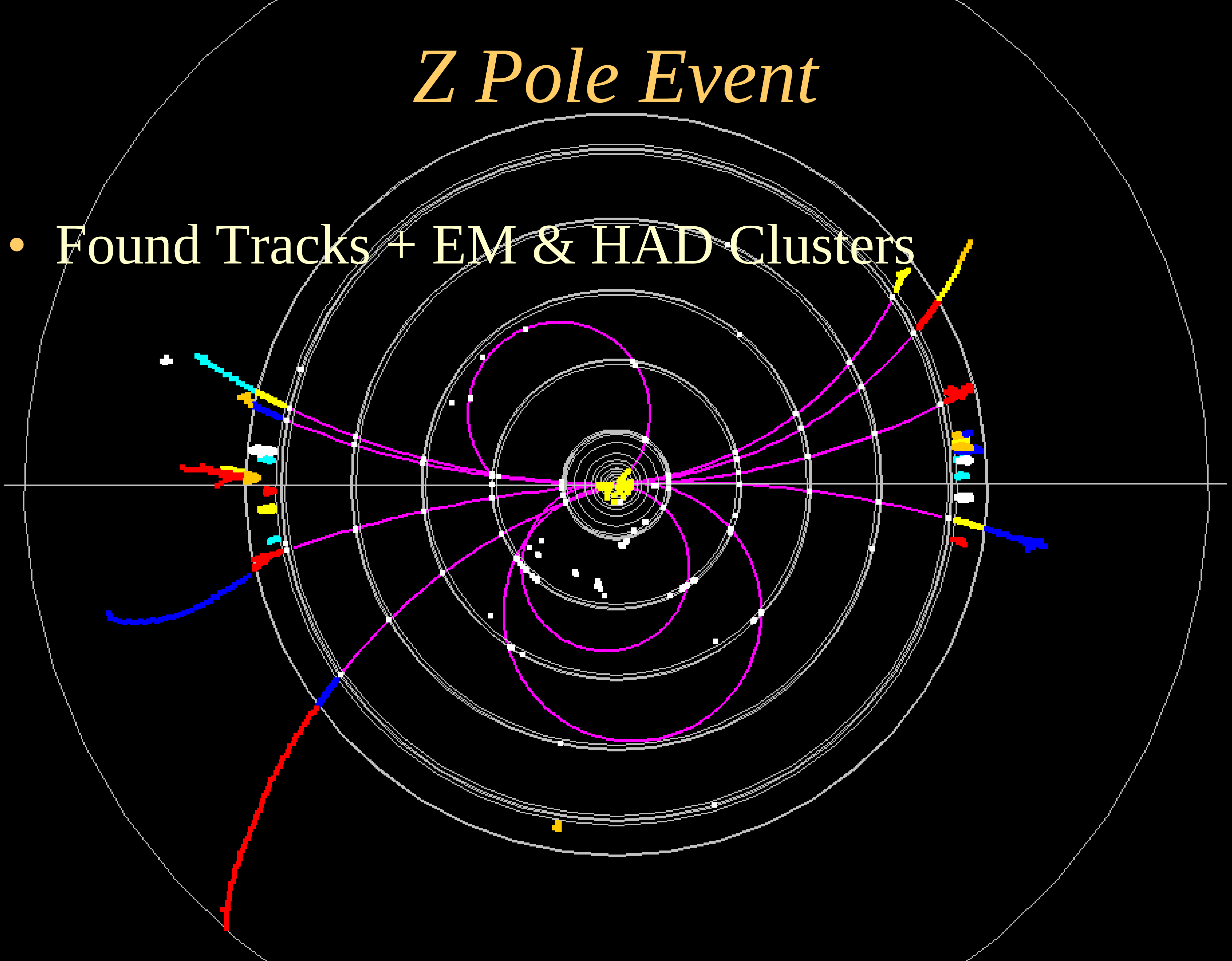
Z Pole Event

- Hits + Found Tracks



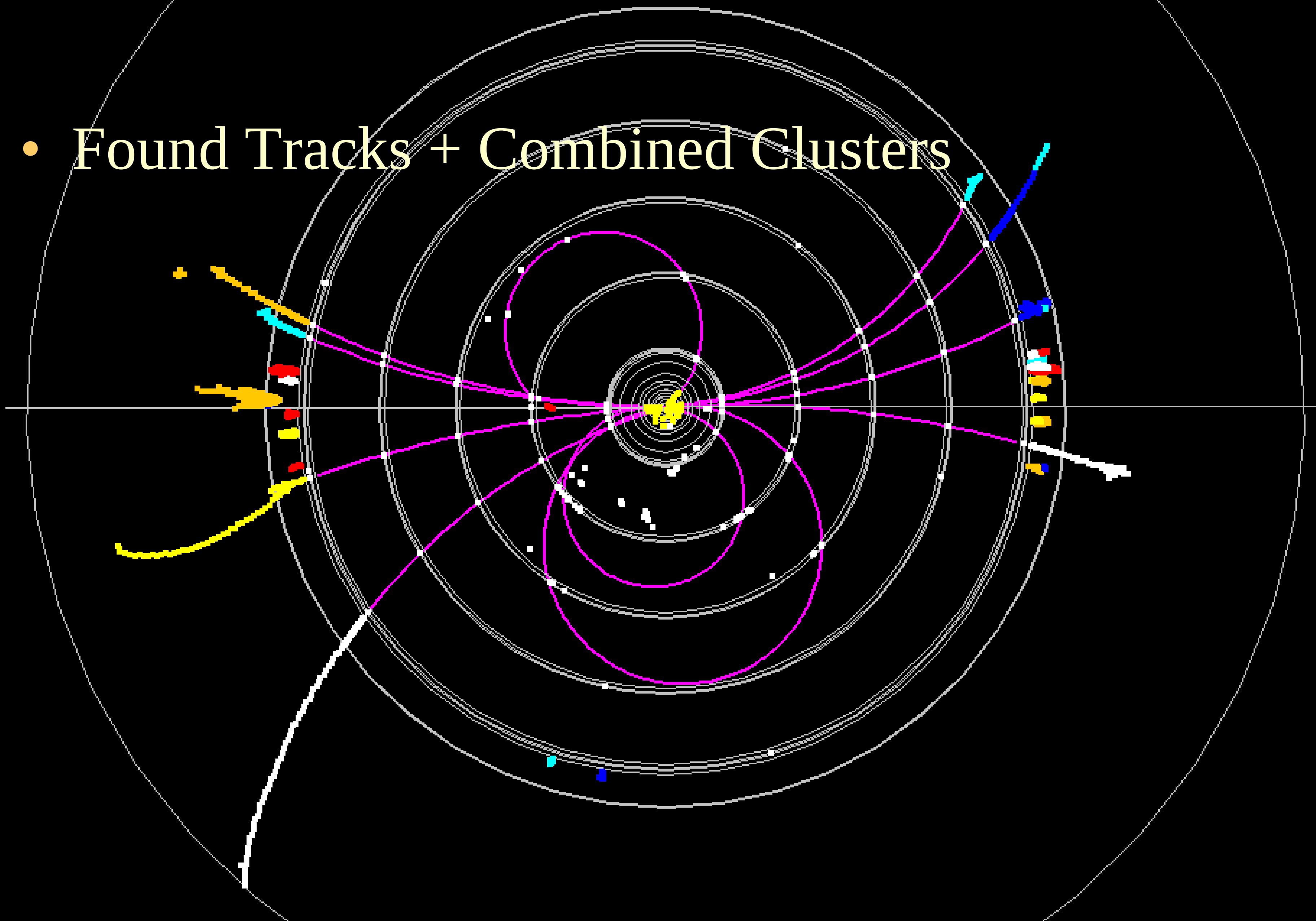
Z Pole Event

- Found Tracks + EM & HAD Clusters



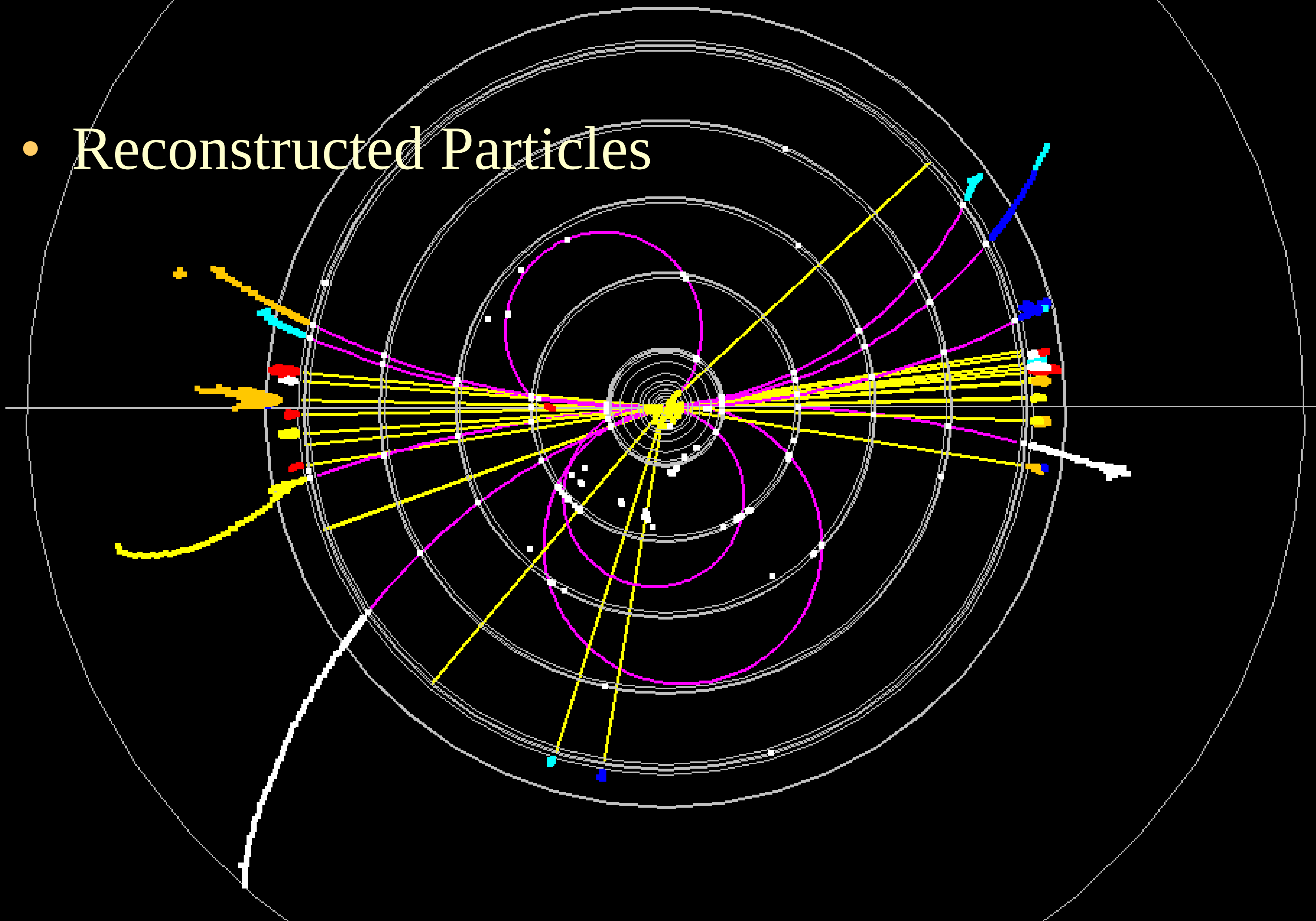
Z Pole Event

- Found Tracks + Combined Clusters

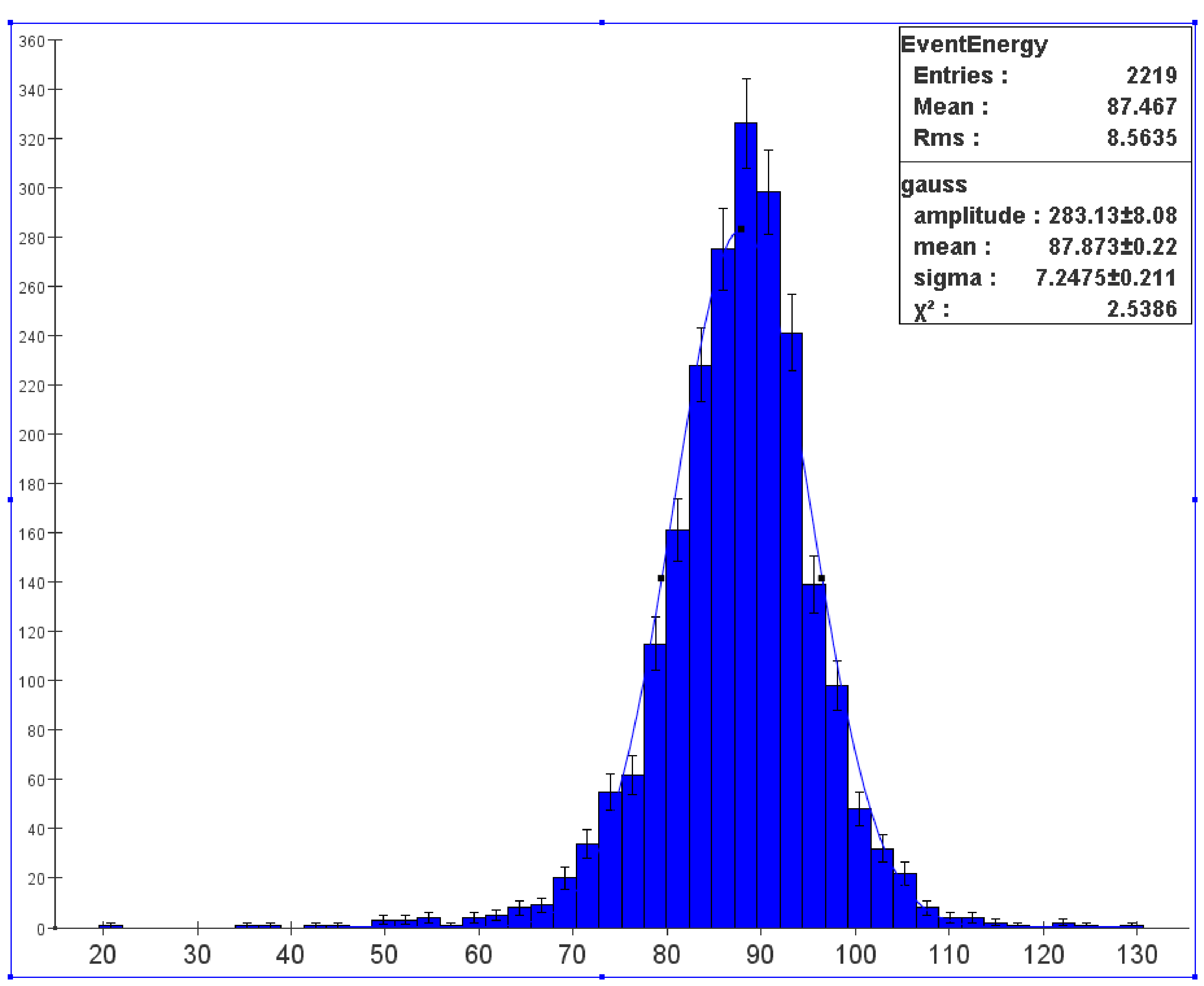


Z Pole Event

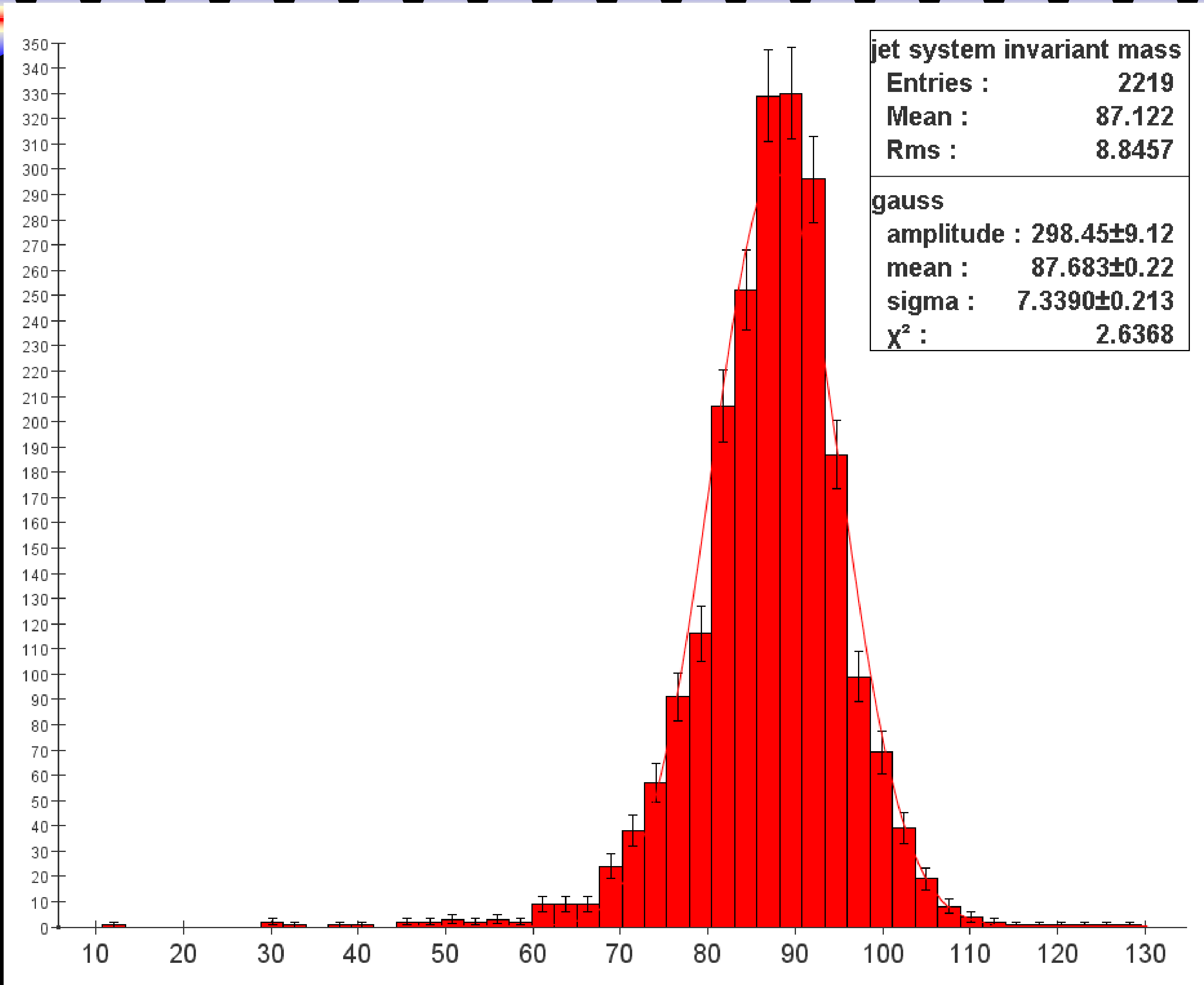
- Reconstructed Particles



Preliminary Results: Event Energy



Preliminary Results: Dijet Mass



Reconstruction Example

```
final public class ExampleReconstruction extends Driver
{
    add( new SmearDriver() );
    add( new VXDBasedReco() );
    add( new SimpleClusterBuilder(1,1,1) );
    add( new IndividualParticleReconstruction() );
    add( new EMClusterAnalyzer(task, eMin, chisqMax) );
    add( new NeutralHadronFinder(radius, seedNhitMin, nHitMin) );
    add( new ReconstructedParticleEventAnalyzer() );
}
```

fetch and return information from the event via the process(EventHeader event) method.

Status

- Results shown were done with hep.lcd analysis code.
- Had hoped to repeat this with org.lcsim. Didn't quite make it, but fairly close.
- Expect to finish this example soon and document as a tutorial.
- Although the distributions peak, and are centered roughly at the correct place, resolutions are somewhat poor.
- Time to tune and optimize.

Summary

- Simple example of individual particle reconstruction is available within hep.lcd framework, expect org.lcsim version soon.
- Few (if any) hardcoded values for either geometries, algorithms, or cuts. These are all determined from the event detector (geometry) or arguments to object constructors (algorithm and cut values).
- Many places along the analysis chain for improvement.