

Hits and Clusters

Rob Kutschke, Caroline Milstene,
Hans Wenzel

Fermilab

May 9, 2007

Overview

- Hits first, then clusters.
- A container of containers:
 - The event contains a
 - **HitStore**
 - For each sensor the HitStore contains
 - A **HitsOnSensor** object (plus HitsonSensor)
 - Each HitsOnSensor object contains
 - ArrayList<**Hit**> (plus HitsonSensor)
 - Example of HitsonSensor = link to geometry
 - Hit is an interface. Implementing classes are:
 - DataStripHit, MCStripHit
 - DataPixelHit, MCPixelHit
 - DataCCDHit, MCCCDHit
 - DataTPCHit, MCTPCHit
- Or maybe there is no Hit interface and the MC classes extend the Dataclasses?
- Or In any case we need the first 6 classes above and should consider the last pair.

Why Data and MC?

- Need to ensure that our algorithms only use what will really be available for use with real data.
 - Helps to prove to others that when we say we are not cheating, we mean it.
 - Of course we will be cheating for some time to come.
- I would like to see the core algorithms use only clusters, not hits. It can follow the trail back to the MCTruth info

DataStripHit

int	getStrip();	
int	getLADC();	
int	getTime();	
double	getADC();	
HitsonSensor	getParent();	Includes access to geometry. More later.

MCStripHit

int	getStrip();
int	getLADC();
int	getTime();
double	getADC();
HitsonSensor	getParent();
MCTruth	getMCTruth();

For now MCTruth is undefined (to avoid using RawTrackerHit).
This model Requires that a Hit come from exactly 1 MCTruth hit.
Implies that event merging must be done at the MCTruth level.
Could also consider returning a List of MCTruth?

DataPixelHit

int	getPixel();
int	getIADC();
int	getTime();
double	getADC();
HitsonSensor	getParent();

The getPixel() method returns an int that can be decoded by geometry system into a 2D local position.

This type could return a 3D position. But only this type of hit can do that.

MCPixelHit

int	getPixel();
int	getIADC();
int	getTime();
double	getADC();
HitsonSensor	getParent();
MCTruth	getMCTruth();

Same caveats as for MCStripHit and DataPixelHit.

DataCCDHit

Pair<int>	getPixel();
int	getIADC();
int	getTime();
double	getADC();
HitsonSensor	getParent();

I am thinking of a clocked CCD.

One cannot compute a 3D position until a bunch crossing is specified.

Can we do the same trick as we do for pixels and return an integer channel Id.

DataTPCHit

Pair<int>	get2DPoint();
int	getIADC();
int	getTime();
double	getADC();
HitsonSensor	getHitsonSensor();

One cannot compute a 3D position until a bunch crossing is specified.

Comment on TPC

- I am trying to see how this fits. It might not.
- Does the abstraction of a sensor fit with the readout on a TPC endplate? It is not a planar thing but it does have some sort of segmentation.

Hit Interface

int	getlADC();
int	getTime();
double	getADC();
HitsonSensor	getHitsonSensor();
MCTruth	getMCTruth();

HitsOnSensor

DetectorElement	getDetectorElement();
ArrayList<Hit>	getHits();
HitStore	getHitStore();

HitStore

HitsOnSensor	getHits(device,layer,phi,z);
HitsOnSensor	getHits(SensorID s);
HitsOnSensor	getHits(some other way);
bool	isMC();
...	Get other meta data ???

StripCluster

double	getCentroid();	Local coord.
double	getSigma();	Local coord.
ArrayList<Hit>	getHits();	
int	getTime();	
double	getADC();	
ClustersonSensor	getParent();	

No data/mc distinction since that is available via the hits.
Remember that a cluster may contain hits from different
MCParticles.

PixelCluster

Pair<double>	getCentroid();	Local coord
Pair<double>	getSigma();	Local coord.
ArrayList<Hit>	getHits();	
int	getTime();	
double	getADC();	
ClustersonSensor	getParent();	

Maybe getSigma() should return a 2x2 matrix instead of a 2 vector? Are we guaranteed that it is diagonal if in local coord?

ClustersOnSensor

DetectorElement	getDetectorElement();
int	getAlgorithmId();
int	getAlgorithmVersion();
ArrayList<Cluster>	getClusters();
ClusterStore	getClusterStore();

ClusterStore

ClustersOnSensor	getClusters(int device, int layer,int phi, int z);
ClustersOnSensor	getClusters(SensorID s);
ClustersOnSensor	getClusters(some other way);
bool	isMC();
...	Get other meta data ???

About ClusterStore

- Should ClusterStore have methods to return a group of clusters that are on more than 1 sensor?
- Or is that the job of a different class that takes a ClusterStore as an argument?