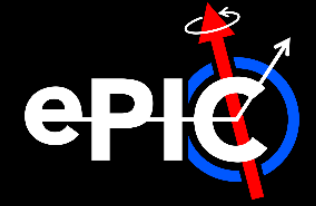


# Calo Data Model Discussion



- **Right:** summary of identified data model and reconstruction needs/wants from January CM
  - c.f. [this summary](#) of the CM discussion for more details
- **Since then:** there has been some discussion, but so far no additional *data model* needs have been identified
  - e.g. BEMC team identified need for including attenuated hits from right/left side of BEMC for threshold studies

## Identified Data Model Needs

- Improved truth-Cluster connections

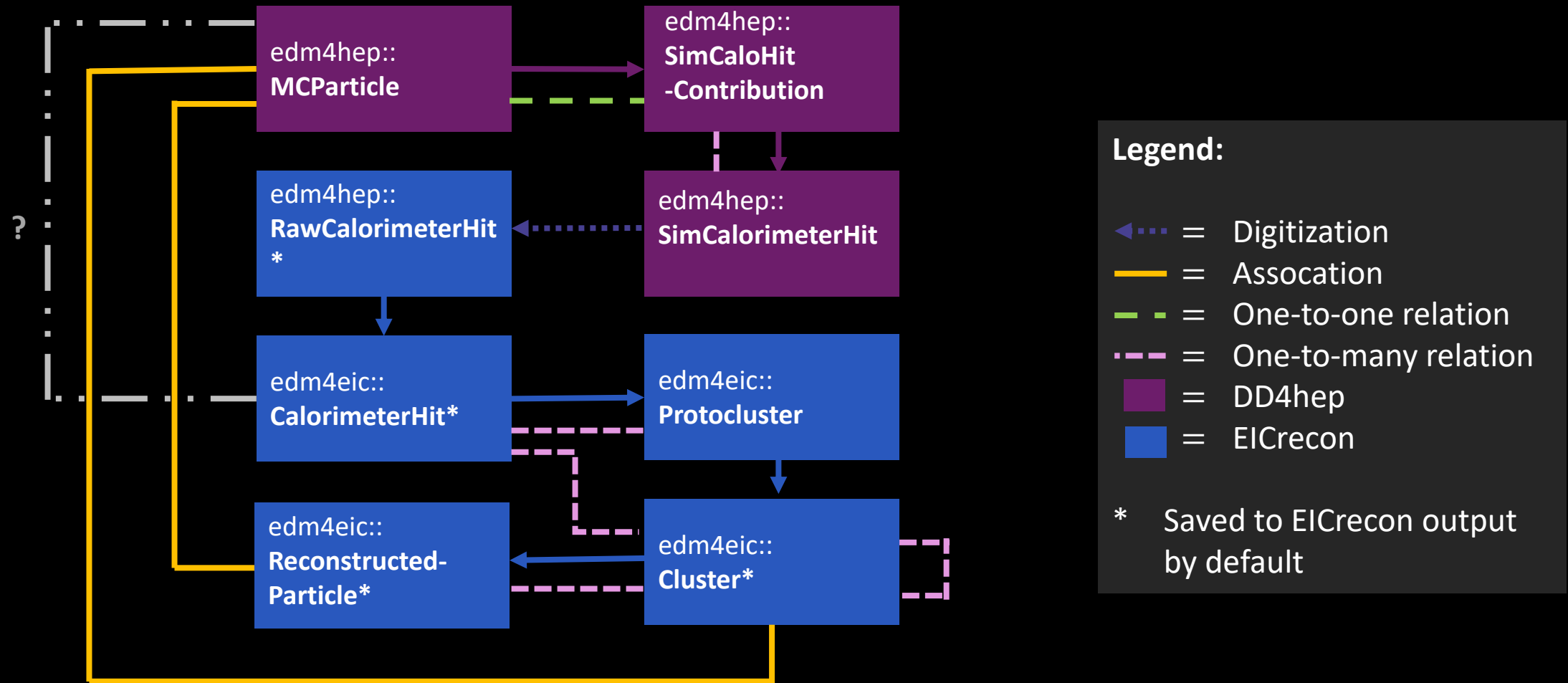
## Identified Reconstruction Needs/Wants

- Clustering implemented in all systems
- Cluster splitting/merging
- ML Integration
- Digitization noise, noise-masking and system-specific digitization model implementations
- Better neutral identification
- Easier access to janadot output

## Identified Simulation Needs/Wants

- Enhanced realism in BEMC implementation and implementation of end-of-sector box material
- Dedicated studies of HGCROC vs. waveform digitizer in BEMC
- Physics-driven performance studies for nHCal
- Update ZDC default to SiPM-on-tile
- Enhanced realism in pECal implementation

# Calo Data Model Discussion | Overview

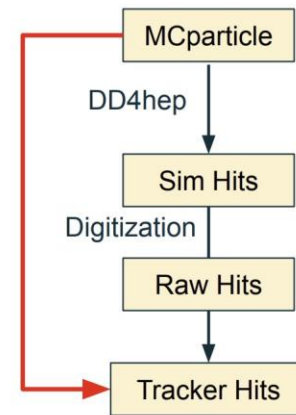


# Calo Data Model Discussion | Previous Discussion



- **Right:** Slide from Shujie & Joe for C/S meeting on February 7<sup>th</sup>
  - Calo situation very much parallels the TrackerHit → MCParticle discussion
  - As far as I can tell: discussion hasn't converged yet...

## Question: link tracker hits to MCparticles



### What we have:

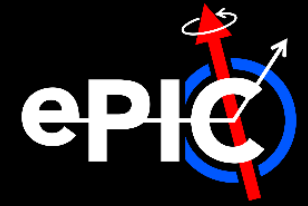
- Sim hits → MCparticle
- Sim hits ↔ Raw Hits

### Options:

1. TrackerHit → MCparticle:
  - direct reference is not feasible b/c it makes recon object depend on simulation
2. Provide an analysis script or ask the user to trace from TrackerHit → RawHit → SimHits → MCparticle offline:
  - It's too complicated and requires processing EICrecon output locally
  - The script will not be maintained within EICrecon
3. Add an EICrecon factory (plugin) to produce TrackerHit ↔ MCparticle association
  - Requires dedicated data structure which can be redundant
  - People may use the output blindly without knowing the process

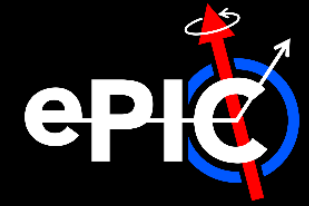
11

# Calo Data Model Discussion | Notes



- Note 1
- Note 2

# Backup | edm4hep::SimCaloHitContribution



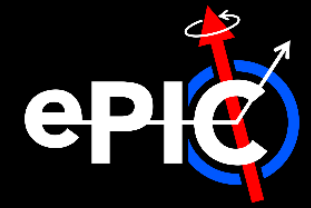
```
#----- CaloHitContribution
edm4hep::CaloHitContribution:
  Description: "Monte Carlo contribution to SimCalorimeterHit"
  Author: "F.Gaede, DESY"
  Members:
    - int32_t   PDG           //PDG code of the shower particle that caused this contribution.
    - float energy           //energy in [GeV] of the this contribution
    - float time             //time in [ns] of this contribution
    - edm4hep::Vector3f stepPosition //position of this energy deposition (step) [mm]
  OneToOneRelations:
    - edm4hep::MCParticle particle //primary MCParticle that caused the shower responsible for this contribution to the hit.
```

# Backup | edm4hep::SimCalorimeterHit



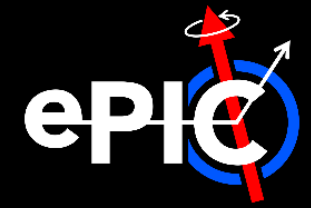
```
#----- SimCalorimeterHit
edm4hep::SimCalorimeterHit:
  Description: "Simulated calorimeter hit"
  Author: "F.Gaede, DESY"
  Members:
    - uint64_t cellID          //ID of the sensor that created this hit
    - float energy            //energy of the hit in [GeV].
    - edm4hep::Vector3f position //position of the hit in world coordinates in [mm].
  OneToManyRelations:
    - edm4hep::CaloHitContribution contributions //Monte Carlo step contribution - parallel to particle
```

# Backup | edm4hep::RawCalorimeterHit



```
#----- RawCalorimeterHit
edm4hep::RawCalorimeterHit:
  Description: "Raw calorimeter hit"
  Author: "F.Gaede, DESY"
  Members:
    - uint64_t cellID //detector specific (geometrical) cell id.
    - int32_t amplitude //amplitude of the hit in ADC counts.
    - int32_t timeStamp //time stamp for the hit.
```

# Backup | edm4eic::RawCalorimeterHit



```
edm4eic::RawCalorimeterHit:
```

```
Description: "Raw (digitized) calorimeter hit"
```

```
Author: "W. Armstrong, S. Joosten"
```

```
Members:
```

```
- uint64_t          cellID          // The detector specific (geometrical) cell id.
- uint64_t          amplitude       // The magnitude of the hit in ADC counts.
  ## @TODO: should we also add integral and time-over-threshold (ToT) here? Or should
  ##         those all be different raw sensor types? Amplitude is
  ##         really not what most calorimetry sensors will give us AFAIK...
- uint64_t          timeStamp       // Timing in TDC
```

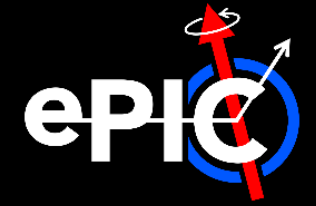


# Backup | edm4eic::CalorimeterHit



```
edm4eic::CalorimeterHit:
  Description: "Calorimeter hit"
  Author: "W. Armstrong, S. Joosten"
  Members:
    - uint64_t      cellID      // The detector specific (geometrical) cell id.
    - float         energy      // The energy for this hit in [GeV].
    - float         energyError // Error on energy [GeV].
    - float         time        // The time of the hit in [ns].
    - float         timeError   // Error on the time
    - edm4hep::Vector3f position // The global position of the hit in world coordinates [mm].
    - edm4hep::Vector3f dimension // The dimension information of the cell [mm].
    - int32_t       sector      // Sector that this hit occurred in
    - int32_t       layer       // Layer that the hit occurred in
    - edm4hep::Vector3f local   // The local coordinates of the hit in the detector segment [mm].
```

# Backup | edm4eic::ProtoCluster



```
edm4eic::ProtoCluster:
```

```
  Description: "Collection of hits identified by the clustering algorithm to belong together"
```

```
  Author: "S. Joosten"
```

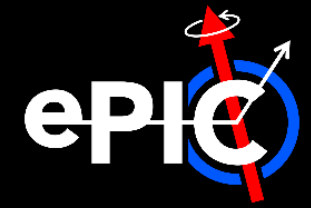
```
  OneToManyRelations:
```

```
    - edm4eic::CalorimeterHit hits           // Hits associated with this cluster
```

```
  VectorMembers:
```

```
    - float          weights                // Weight for each of the hits, mirrors hits array
```

# Backup | edm4eic::Protocluster



```
edm4eic::ProtoCluster:
```

```
  Description: "Collection of hits identified by the clustering algorithm to belong together"
```

```
  Author: "S. Joosten"
```

```
  OneToManyRelations:
```

```
    - edm4eic::CalorimeterHit hits           // Hits associated with this cluster
```

```
  VectorMembers:
```

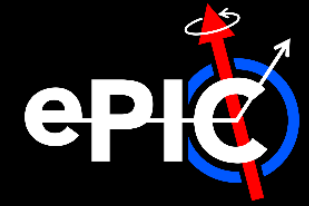
```
    - float          weights                // Weight for each of the hits, mirrors hits array
```

# Backup | edm4eic::Cluster

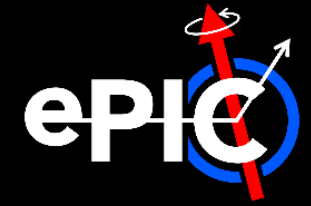


```
edm4eic::Cluster:
  Description: "EIC hit cluster, reworked to more closely resemble EDM4hep"
  Author: "W. Armstrong, S. Joosten, C.Peng"
  Members:
    # main variables
    - int32_t      type           // Flag-word that defines the type of the cluster
    - float       energy         // Reconstructed energy of the cluster [GeV].
    - float       energyError    // Error on the cluster energy [GeV]
    - float       time           // [ns]
    - float       timeError      // Error on the cluster time
    - uint32_t    nhits          // Number of hits in the cluster.
    - edm4hep::Vector3f position // Global position of the cluster [mm].
    - edm4eic::Cov3f positionError // Covariance matrix of the position (6 Parameters).
    - float       intrinsicTheta // Intrinsic cluster propagation direction polar angle [rad]
    - float       intrinsicPhi   // Intrinsic cluster propagation direction azimuthal angle [rad]
    - edm4eic::Cov2f intrinsicDirectionError // Error on the intrinsic cluster propagation direction
  VectorMembers:
    - float       shapeParameters // Should be set in metadata, for now it's a list of -- radius [mm], dispersion [mm], 2 entries for
    - float       hitContributions // Energy contributions of the hits. Runs parallel to ::hits()
    - float       subdetectorEnergies // Energies observed in each subdetector used for this cluster.
  OneToManyRelations:
    - edm4eic::Cluster      clusters // Clusters that have been combined to form this cluster
    - edm4eic::CalorimeterHit hits   // Hits that have been combined to form this cluster
    - edm4hep::ParticleID  particleIDs // Particle IDs sorted by likelihood
```

# Backup | edm4eic::ReconstructedParticle



```
edm4eic::ReconstructedParticle:
Description: "EIC Reconstructed Particle"
Author: "W. Armstrong, S. Joosten, F. Gaede"
Members:
- int32_t          type          // type of reconstructed particle. Check/set collection parameters ReconstructedParticleTypeNames and
- float           energy        // [GeV] energy of the reconstructed particle. Four momentum state is not kept consistent internally.
- edm4hep::Vector3f momentum    // [GeV] particle momentum. Four momentum state is not kept consistent internally.
- edm4hep::Vector3f referencePoint // [mm] reference, i.e. where the particle has been measured
- float          charge        // charge of the reconstructed particle.
- float          mass          // [GeV] mass of the reconstructed particle, set independently from four vector. Four momentum state
- float          goodnessOfPID // overall goodness of the PID on a scale of [0;1]
- edm4eic::Cov4f covMatrix     // covariance matrix of the reconstructed particle 4vector (10 parameters).
##@TODO: deviation from EDM4hep: store explicit PDG ID here. Needs to be discussed how we
##       move forward as this could easilly become unwieldy without this information here.
##       The only acceptable alternative would be to store reconstructed identified
##       particles in separate collections for the different particle types (which would
##       require some algorithmic changes but might work. Doing both might even make
##       sense. Needs some discussion, note that PID is more emphasized in NP than
##       HEP).
- int32_t          PDG          // PDG code for this particle
## @TODO: Do we need timing info? Or do we rely on the start vertex time?
OneToOneRelations:
- edm4eic::Vertex  startVertex  // Start vertex associated to this particle
- edm4hep::ParticleID particleIDUsed // particle ID used for the kinematics of this particle
OneToManyRelations:
- edm4eic::Cluster clusters     // Clusters used for this particle
- edm4eic::Track  tracks       // Tracks used for this particle
- edm4eic::ReconstructedParticle particles // Reconstructed particles that have been combined to this particle
- edm4hep::ParticleID particleIDs // All associated particle IDs for this particle (not sorted by likelihood)
ExtraCode:
declaration: "
    bool isCompound() const {return particles_size() > 0;}n
"
```



## edm4eic::MCRecoClusterParticleAssociation:

**Description:** "Association between a Cluster and a MCParticle"

**Author :** "S. Joosten"

### Members:

```
- uint32_t      simID      // Index of corresponding MCParticle (position in MCParticles array)
- uint32_t      recID      // Index of corresponding Cluster (position in Clusters array)
- float        weight     // weight of this association
```

### OneToOneRelations:

```
- edm4eic::Cluster  rec      // reference to the cluster
- edm4hep::MCParticle sim    // reference to the Monte-Carlo particle
```

- **Note:** seems to associates cluster to particle associated with highest energy cell