

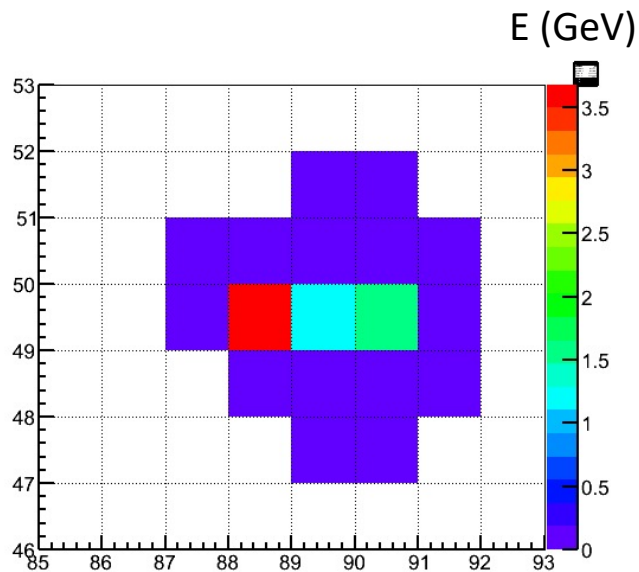
Clustering in Fun4All

A.Bazilevsky

April 28, 2021

Clustering

CEMC: 8 GeV π^0



One cluster

Two sub-clusters

1. **Cluster** as a group of contiguous towers (with common edge) with energy above threshold
2. Cluster is splitted on **sub-clusters**

Sub-Cluster are associated with local maxima (towers with energy larger than in any of 8 surrounding towers)

The energy in each tower of a cluster is shared between showers associated with each sub-cluster, using parameterized **shower profile**

Iterative procedure tuning shower energies and positions

Result not sensitive to the details of shower profile (default profile can be used for major of calorimeters)

Clustering in Fun4All



Once ECal/HCal detector is created in Fun4All, clustering can be called simply through **RawClusterBuilderTemplate** interface which will provide (sub)clustering, with output collected in **RawClusterContainer** :

- List of towers composing a subcluster

- Energy share of each tower in a subcluster

- Total subcluster energy

- Subcluster position

For energy and position fine corrections, as well as for shower shape evaluation one has to write a module inherited from **BEmcRec**

- Implements details of detector performance

Old interfaces (retired?):

- RawClusterBuilderGraph**

- RawClusterBuilderFwd**

- Don't do cluster splitting

Call from Fun4All

Minimal set of commands (for any EM/H-Cal):

```
RawClusterBuilderTemplate *ClusterBuilder = new RawClusterBuilderTemplate("RawClusterBuilderTemplate");  
ClusterBuilder->Detector("<CalorName>");  
ClusterBuilder->SetPlanarGeometry(); or ClusterBuilder->SetCylindricalGeometry();  
se->registerSubsystem(ClusterBuilder);
```

It outputs **RawCluster** with energy and position

To calculate shower profile parameter “prob”, and energy and position (e.g. due to shower depth) corrections, one needs to define them in the corresponding **BEmcRec<CalorName>** module inheriting basic functionality from **BEmcRec**

Existing BEmcRec<CalorName> modules:

BEmcRecCEMC – for barrel W/SciFi

BEmcRecEEMC – for PWO e-endcap

BEmcRecFEMC – for Pb/SciFi h-endcap

Backup

“Template” Clustering implementation in Fun4All

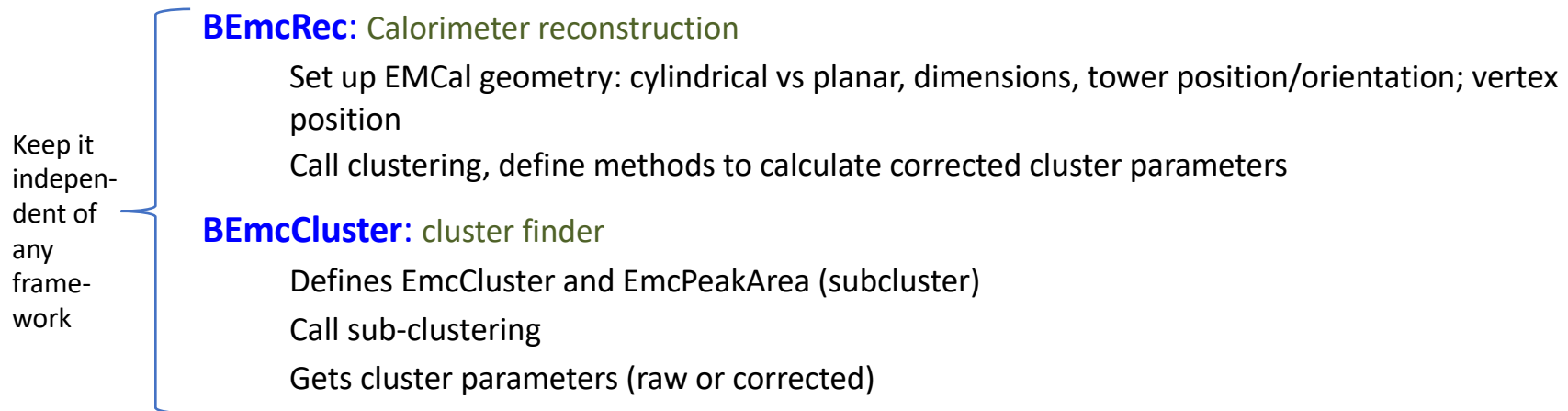


RawClusterBuilderTemplate: Interface between Fun4All framework and this clustering

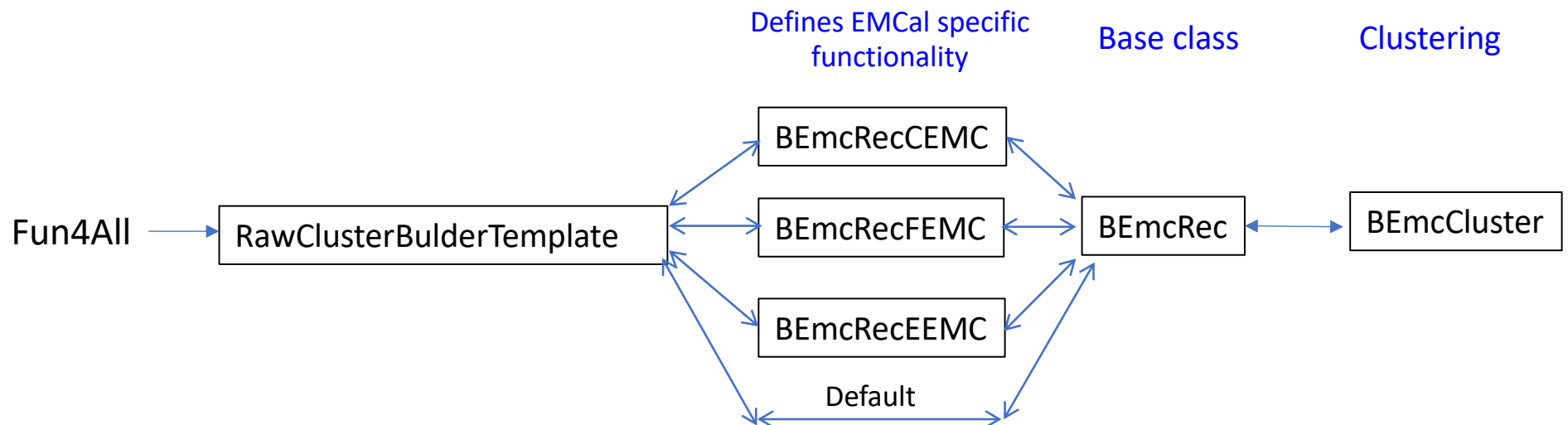
Gets tower geometry from **RawTowerGeomContainer** and feed it in BEmcRec

Gets towers fired in an event from **RawTowerContainer** and pass it to BEmcRec

Pass clustering results from BEmcCluster to **RawClusterContainer**



... for many EMCals



The bypass from RawClusterBulderTemplate → BEmcRec is used if the corresponding BEmcRec<CalorName> is not defined

In this way no shower profile evaluation ("prob") and Calorimeter specific corrections for energy/position performed