

Table of Contents

Spike Cleaning	1
Introduction.....	1
Spike Cleaning with Re-reconstruction.....	1
Tools to clean at the level of the objects.....	1
Swiss Cross.....	2
kOutOfTime.....	2
More information.....	2
Review status.....	2

Spike Cleaning

An area for Offline Guide information

Introduction

Below you can find some useful instruction that allow to apply spike cleaning at different level. All this instruction refer to CMSSW versions < 3_7_X.

Spike Cleaning with Re-reconstruction

There is the possibility to produce a cleaned collection of clusters and to re-reconstruct all the high-level objects (i.e. Electrons and Photons) on top of that. To perform the re-reconstruction with CMSSW_3_6_X some tags need to be checked out:

```
cvs co -r V00-01-20      RecoEcal/Configuration
cvs co -r V00-06-16      RecoEcal/EgammaClusterAlgos
cvs co -r V00-06-50-02   RecoEcal/EgammaClusterProducers
cvs co -r V00-11-02-01   RecoLocalCalo/EcalRecAlgos
cvs co -r V00-04-27      RecoEgamma/EgammaHLTProducers
cvs co --r V01--06-03    RecoEgamma/EgammaElectronProducers
cvs co --r V01--12-06    RecoEgamma/EgammaElectronAlgos
cvs co --r V00--03-19    RecoEgamma/EgammaIsolationAlgos
cvs co --r V01--00-06    RecoEgamma/PhotonIdentification
addpkg RecoEgamma/EgammaPhotonProducers
cvs co -d EGamma/EGammaSkims -r pm20100610a UserCode/EGamma/EGammaSkims
checkdeps -a
scram b
```

The full sequence that needs to be run is defined in

```
EGamma/EGammaSkims/python/cleanReRecoSequence_cff.py
```

A sequence like this one:

```
process.ecalCleanClustering = cms.Sequence(process.cleanedEcalClusters*process.cleanedEgammaSkimR
```

must be included in your path.

An example of a reReco with cleaning enabled is under EGamma/EGammaSkims/test

```
runCleaningReReco.py
```

It will produce objects whose name is the same of the original one (except the process name).

This solution has the advantage that superCluster production is not spoiled by the presence of isolated spikes close to the signal. The provided solution for the moment is cleaning only isolated spikes using the swiss cross algorithm.

To apply a cut for the timing in 3_6_X we suggest you to use the kOutOfTime recHit flag as explained later

Tools to clean at the level of the objects

Swiss Cross

If you want to use the "Swiss Cross" method a static function to compute this variable given the DetId of the hit is provided in RecoLocalCalo/EcalRecAlgos:

```
static float swissCross( const DetId id, const EcalRecHitCollection &, float recHitEtThreshold =
```

As input it takes also the RecHit collection and an optional RecHit Et threshold which by default is set to 0. The recommended cut is `swissCross > .95`.

kOutOfTime

If you want to clean the collection using the

```
EcalRecHit::Flags = kOutOfTime
```

flag you have to check directly the RecHit using the:

```
uint32_t recoFlag() const
```

method provided by the EcalRecHit interface.

More information

More information are available in the ECAL cleaning recipe twiki

Review status

Reviewer/Editor and Date (copy from screen)	Comments
---	----------

Responsible: ResponsibleIndividual

Last reviewed by: Most recent reviewer

This topic: CMSPublic > SWGuideEgammaSpikeCleaning

Topic revision: r9 - 2010-06-17 - FrancescoPandolfi



Copyright &© 2008-2021 by the contributing authors. All material on this collaboration platform is the property of the contributing authors.

or Ideas, requests, problems regarding TWiki? use Discourse or Send feedback