Rechit flagging/anomalous events

D.A. Petyt 19/07/12

- ECAL RecHit quality flags are used extensively in object reconstruction
  - They are used to exclude “bad” rechits from seeding or being included in jets and electrons/photons. Spike cleaning is the most commonly known example of this

- Analysis of anomalous or problematic events [*] allows the performance of the rechit flagging to be checked:
  - What types of anomalous events do we observe? Are new categories appearing as a function of time (lumi?)
  - How can such events be removed by the existing rechit flags or custom event filters?
  - What can be done to improve the flagging to reject such events in future CMSSW releases?

[*] POG/PAG posts to the ECAL performance HN

Rechit flagging reminder

- Flags are set during rechit formation.
- They encapsulate the “quality” of the reconstructed pulse:
  - whether the pulse is saturated
  - if the rechit has been recovered from TP info (or neighbours)
  - whether special reconstruction was performed (amplitude estimated from the leading edge of the pulse in case of saturation)
  - timing and topological information (for spike rejection)

Meaning of the rechit flags

CMSSW/DataFormats/EcalRecHit/interface/EcalRecHit.h

kGood	self-explanatory
kPoorReco	bad pulse shape (chi^2 fit)
kOutOfTime	rechit time inconsistent with zero
kFaultyHardware	not used
kNoisy	not used
kPoorCalib	anomalous value of laser correction
kSaturated	digi saturating full scale of ADC
kLeadingEdgeRecovered	saturated rechit with amplitude
kNeighboursRecovered	single channel recovery (not implemented)
kTowerRecovered	recovery from TP info
kDead	set to true if TP recovery fails
kKilled	not used
kTPSaturated	recovered from saturated TP (>127.5 GeV)
kL1SpikeFlag	spike-tagged at L1
kWeird	topological spike
kDiWeird	topological di-spike
kUnknown	self-explanatory!

Since CMSSW 4_2_X: rechit flags stored as bits in a 32 bit word.

A rechit can have more than one bit set...

Rechits and severities

- Severity level:
  - an OR of several related rechit flags
  - can be used to reject a specific category (or categories) of problematic rechit
  - defined in order of increasing severity.

Severity level

increasing severity

0	1	2	3	4	5
`kGood`	`kProblematic`	`kRecovered`	`kTime`	`kWeird`	`kBad`
no problems	saturated or poorly reconstructed pulse	alternative algorithms used to determine rechit energy	spike by timing or topology		dead or non-recovered channels
	rechit quality may be compromised?	*rechit energy resolution may be compromised*	*event quality will be compromised*		*no information from this channel*

Mapping between flags and severity level

Rechit Severity Level

Rechit Flag	kGood	kProblematic	kRecovered	kTime	kWeird	kBad
kGood						
kPoorReco						
kPoorCalib						
kNoisy						
kSaturated						
kLeadingEdgeRecovered						
kTowerRecovered					spikes	
kOutOfTime						
kWeird						
kDiWeird						
kFaultyHardware						
kDead						
kKilled						
kTPSaturated						
kL1SpikeFlag						
kNeighboursRecovered						
kUnknown						

http://cmssw.cvs.cern.ch/cgi-bin/cmssw.cgi/CMSSW/RecoLocalCalo/EcalRecAlgos/python/EcalSeverityLevelESProducer_cfi.py?revision=1.5&view=markup

Object cleaning

- Is performed using either Severity levels, rechit flags, or both.
- The next page summarises my understanding of which rechit flags are excluded in the formation of CaloTowers (CaloJets), PF clusters (PF Jet/MET) and ECAL superclusters (electron/photon)
- Where is all this defined?
  - **CaloTower:**
    CMSSW/RecoLocalCalo/CaloTowersCreator/python/calotowermaker_cfi.py
    (note explicit rejection of recovered rechits)
  - **PF Cluster:**
    CMSSW/RecoParticleFlow/PFClusterProducer/plugins/PFRecHitProducerECAL.cc
    (explicit rejection of rechit flags in the code)
  - **ECAL Supercluster:**
    CMSSW/RecoEcal/EgammaClusterProducers/python/hybridSuperClusters_cfi.py (EB)
    CMSSW/RecoEcal/EgammaClusterProducers/python/multi5x5BasicClusters_cfi.py (EE)

Object cleaning by rechit flag/severity

Rechit Flag	CaloTower	PFCluster	Supercluster
kGood			
kPoorReco			
kPoorCalib			
kNoisy			
kSaturated			
kLeadingEdgeRecovered			EE only
kTowerRecovered			
kOutOfTime			
kWeird			
kDiWeird			spikes
kFaultyHardware			
kDead			
kKilled			
kTPSaturated			
kL1SpikeFlag			
kNeighboursRecovered			
kUnknown			
	based on severity level only	based on rechit flags only	based on rechit flags and severity level

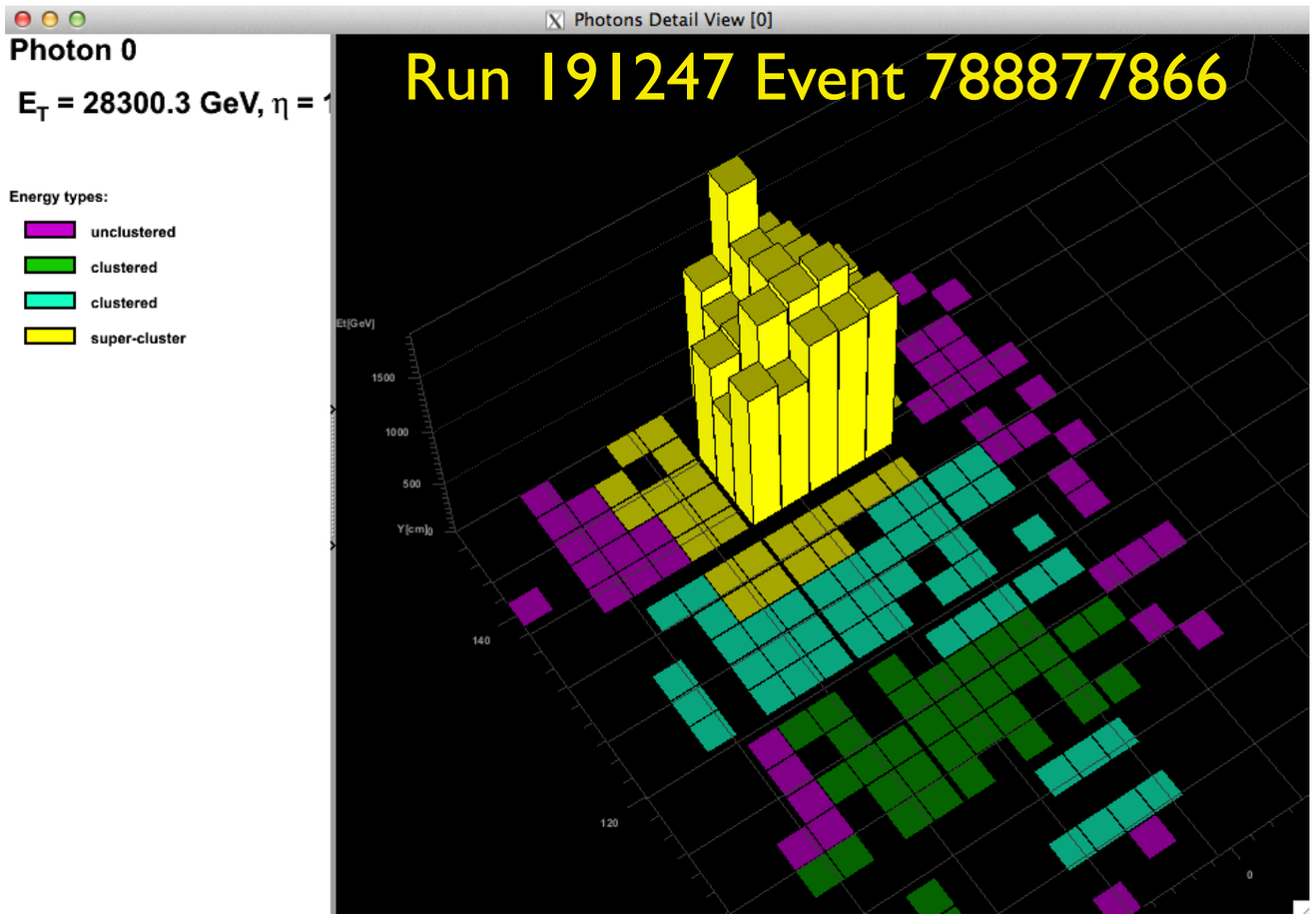
Note on consistent use of rechit flags between objects

- Obvious question to ask:
  - Why is the use of rechit flags different for Calo/PF/Egamma? Why not use the same flag-based rejection for all objects?
  - Needs of different objects may be not be the same, depending on performance requirements (i.e. ultimate resolution performance for electrons/photons). These should be defined by POG experts.
  - Nevertheless, I see several areas that need cleaning up:
    - **inconsistent use of kLeadingEdgeRecovered flag for Calo/PF**
    - **different flags used for EB and EE in Egamma reco**
  - In the longer term (6_1_X+) one could consider assigning quality flags to the objects themselves rather than rejecting specific classes of hits during object formation. It is then an analysis-level decision whether to use objects formed from recovered rechits or not.

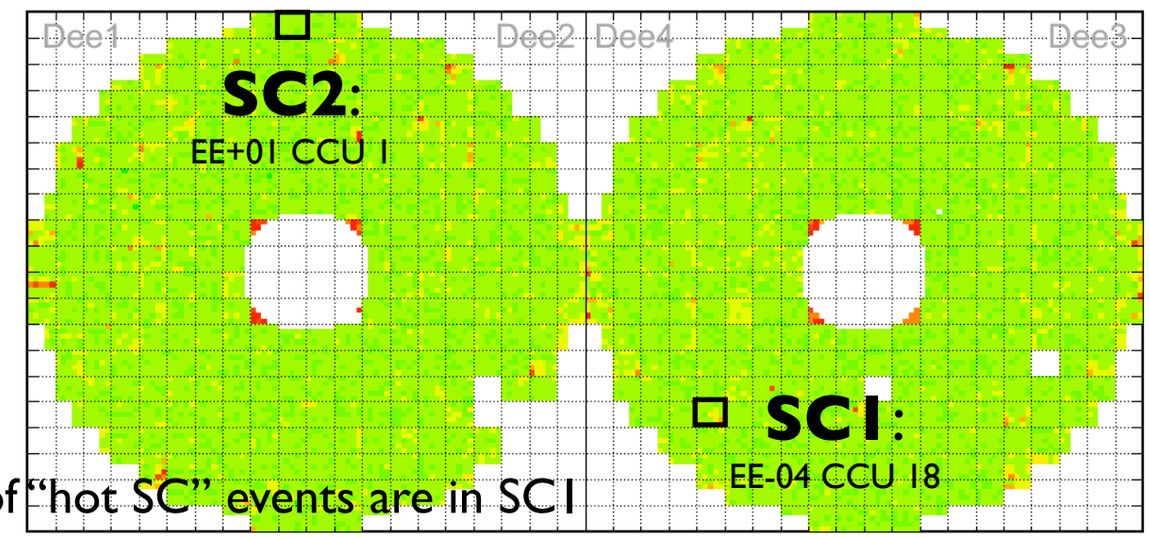
Recent examples of problematic events

- Rogues gallery of events posted to the ECAL performance HN
  - Their properties, and our best guess of their origin
  - How well the current flagging performed
  - How we get rid of the events with our current tools
  - What can be done to improve the flagging in the future?

"Hot SCs"



Location of hot SCs



>>90% of "hot SC" events are in SC1

- What:
  - ~20 TeV photons caused by anomalous pulses in 2 SCs in EE
  - often all 25 channels light up, rechits are not rejected by kWeird flag used to reject single VPT discharge channels
- When:
  - ~100 events observed in 2012A+B
  - first event brought to my attention was from run 190707.
  - rate much higher in 2012 than 2010-11...
- Why:
  - still to be investigated. Suspect possible HV-related origin.

Rechit characteristics

run: 191247 evt: 947682903

ix	iy	iz	E	t	flags[*]					
21	21	-1	2144.1	-27.56	0	0	0	0	0	1
21	22	-1	2198.51	-26.54	0	0	0	0	0	1
21	23	-1	2295.57	16.00	0	0	1	0	0	0
21	24	-1	2565.78	-25.01	0	0	0	0	0	1
21	25	-1	2638.22	-26.03	0	0	0	0	0	1
22	21	-1	3110.42	-26.84	0	0	0	0	0	1
22	23	-1	2761.95	8.47	0	1	0	0	0	0
22	24	-1	2260.03	29.41	0	0	1	0	0	0
22	25	-1	2772.45	-25.26	0	0	0	0	0	1
23	21	-1	3704.8	15.44	0	1	1	0	0	0
23	22	-1	3503.95	-25.90	0	0	0	0	0	1
23	23	-1	2431.14	-26.27	0	0	0	0	0	1
23	24	-1	2933.32	-26.87	0	0	0	0	0	1
23	25	-1	3126.78	-24.86	0	0	0	0	0	1
24	21	-1	3221.86	-26.58	0	0	0	0	0	1
24	23	-1	2143.41	-25.29	0	0	0	0	0	1
24	24	-1	2470.42	-25.69	0	0	0	0	0	1
24	25	-1	2632.66	-24.97	0	0	0	0	0	1
25	22	-1	1505.07	28.23	0	0	1	0	0	0
25	23	-1	1868.06	17.27	0	0	1	0	0	0
25	24	-1	1290.11	26.88	0	0	1	0	0	0
25	25	-1	2673.03	11.56	0	1	1	0	0	0

[*] kGood, kPoorReco, kOutOfTime, kPoorCalib, kSaturated, kLeadingEdgeRecovered

- A large number of high energy rechits with non-kGood rechit flags
- No single flag is common between all the rechits...
  - At the moment we cannot make a simple cut on one flag (kPoorReco, for example)
  - Can we define a specific combination of rechit flags (which includes optimising the definition of the flags) that can catch all of these problematic hits?

Rechit pulse shapes

Examination of the digis from two example rechits:

EE hit: Run 191062 Event 1914131117 ix=24 iy=23 iz=-1 E=3738.65

sample	01	02	03	04	05	06	07	08	09	10
ADC	**197**	**197**	**197**	**3941**	**3791**	**4031**	**4031**	**4031**	**4031**	**4031**
Gain	12	12	12	12	1	0	0	0	0	0

huge pulse, saturates full scale of ADC (4 TeV)

EE hit: Run 191201 Event 54277691 ix=23 iy=21 iz=-1 E=6486.09

sample	01	02	03	04	05	06	07	08	09	10
ADC	**390**	**530**	**2756**	2350	4031	4031	4031	4031	4031	4031
Gain	12	12	12	1	0	0	0	0	0	0

huge pulse, with significant activity in 3 pre-samples

Rechit filter

- Developed, with Kostas T. and members of the MET group, a filter to run on AOD and filter out these "Bad SC" events"
- Apply a conditional mask on rechits in these two SCs. **IF:**
  - more than **N** rechits in SC with non-kGood flag, and E> **Y** TeV
  - total sumE_T of rechits in this SC > **Z** TeV
- → **Reject event**

Parameters: N=2, Y=1 TeV Z=3 TeV

Code available in METFilter package:

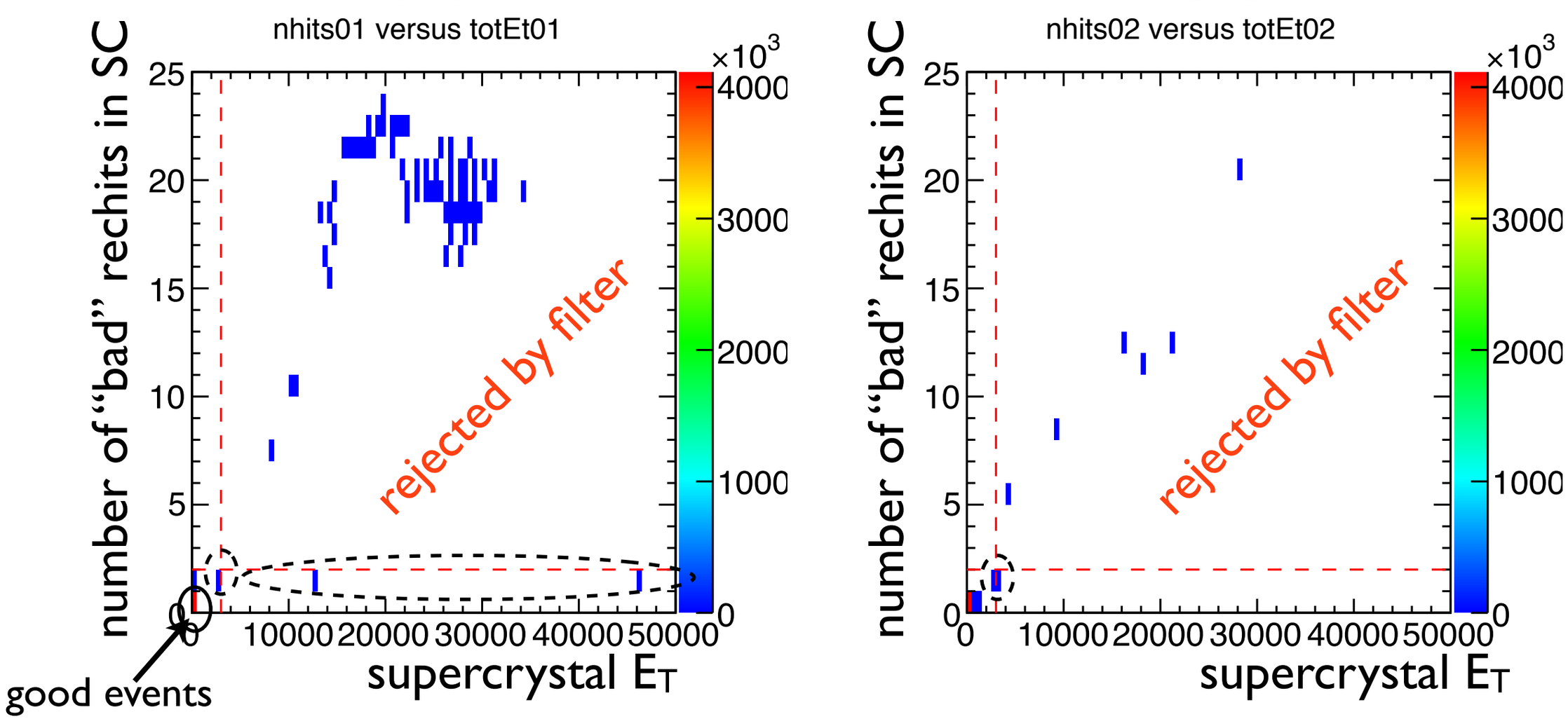
CMSSW/RecoMET/METFilters/plugins/EEBadScFilter.cc
CMSSW/RecoMET/METFilters/python/eeBadScFilter_cfi.py

recommended for use on 2012 data by MET group
https://twiki.cern.ch/twiki/bin/viewauth/CMS/MissingETOptionalFilters

Validation of filter

- Check performance of filter on Run2012A HT dataset

Hongxuan Liu

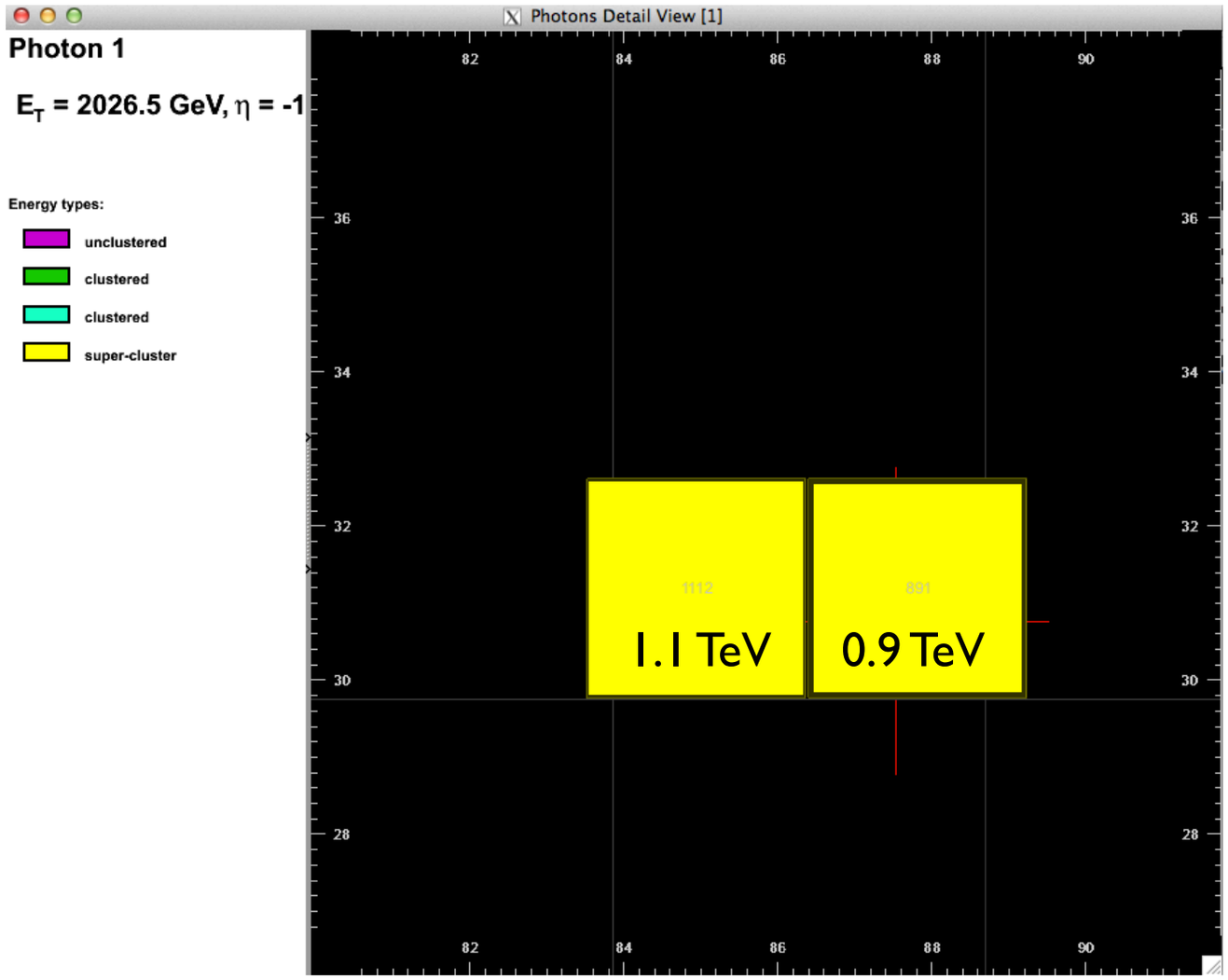


Problematic events effectively cleaned by this filter. No evidence of overcleaning of "good" events.

6 "problematic" events fail filter: 4 are single high E rechits cleaned by Swiss Cross. 2 could be cleaned by reoptimising cuts (lowering E_T rechit threshold)

“2 hot xtals”

Run 196249, Event 65046168



- What:
  - ~ TeV photons caused by anomalous pulses in 2 adjacent crystals in one SC in EE
- When:
  - ~5-10 events observed in 2012A +B
  - not seen prior to 2012
- Why:
  - possible data corruption from one problematic CCU: EE-08 CCU 21??

nothing in the neighbouring xtals.
Inconsistent with EM shower shape.

Rechit characteristics

Run 195304 Event 784804143
ix=80 iy=62 iz=-1 E=3952.22 Et=1117.04 time=-3.70036
swisscross=-0.0073508 ee_eta=-1.93614 ee_phi=0.370533

RECHIT flags:
EcalRecHit::kGood=0
EcalRecHit::kPoorReco=1
EcalRecHit::kOutOfTime=0
EcalRecHit::kFaultyHardware=0
EcalRecHit::kNoisy=0
EcalRecHit::kPoorCalib=0
EcalRecHit::kSaturated=0
EcalRecHit::kLeadingEdgeRecovered=0
EcalRecHit::kNeighboursRecovered=0
EcalRecHit::kTowerRecovered=0
EcalRecHit::kDead=0
EcalRecHit::kKilled=0
EcalRecHit::kTPSaturated=0
EcalRecHit::kL1SpikeFlag=0
EcalRecHit::kWeird=0
EcalRecHit::kDiWeird=0
EcalRecHit::kUnknown=0
SEVERITY=1

EE DIGI printout:

SAMPLE,ADC,GAIN:

01	02	03	04	05	06	07	08	09	**10**
195	197	196	196	197	195	195	194	196	**163**
12	12	12	12	12	12	12	12	12	**0**

pedestal gain 0

- TeV energies in a pair of rechits in the same CCU.
  - not always the same 2 hits, but they are always in neighbouring ix,iy
  - the first 9 samples are consistent with pedestal
  - **the 10th sample is always gain 0 for these events**
  - Is it data corruption (c.f. firmware issue affecting last 3 samples before April TS) -> to be investigated
  - Also to be checked: why are these hits being assigned rechit flag kPoorReco...

How to reject?

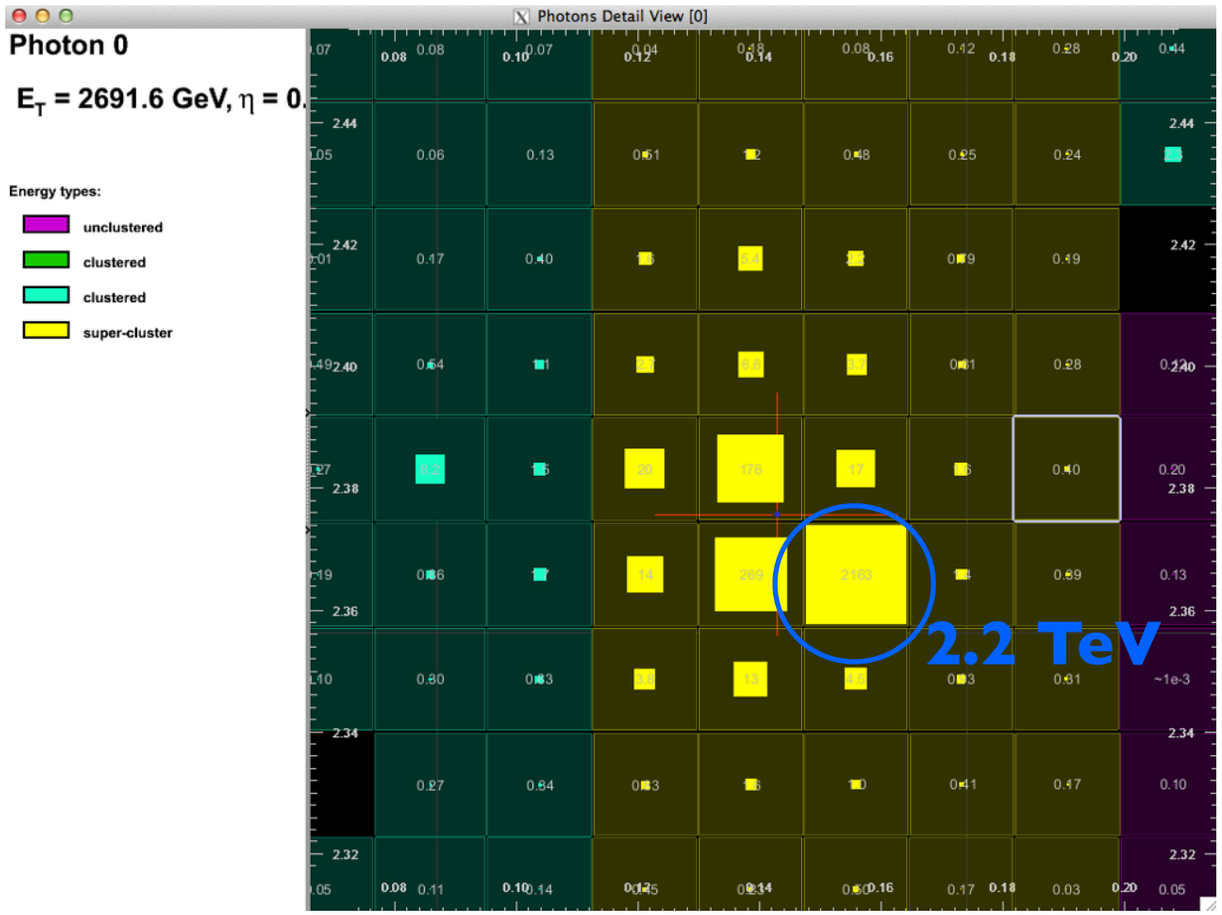
- While the origin of these rechits is being investigated, we may provide a short-term fix by adapting the Bad SC filter:
  - add a 3rd "bad SC" to the list: EE-08 CCU 21. Rechits in this SC will also be checked by the filter.
  - lower the requirements on the supercrystal sumE_T since fewer crystals are involved:

    Parameters (old): N=2, Y=1 TeV Z=3 TeV
    Parameters (new): N=2, Y=1 TeV Z=1 TeV

    - new config should still be safe (see slide 14). It is being investigated/ validated by the MET group.
- Such pulses should be flagged by kPoorReco or an equivalent rechit flag

kLeadingEdgeRecovered

Run 196364, Event 1039699820



caloMET ~110 GeV

pfMET ~2210 GeV

due to 1 hit with rechit flag kLeadingEdgeRecovered

- What:
  - Event with large discrepancy in CaloMET and PFMET
  - An EM shower with a seemingly anomalous 2 TeV embedded energy deposit
- When:
  - only one brought to our attention so far...
- Why:
  - most likely an embedded spike
  - E/p of reconstructed electron >>1

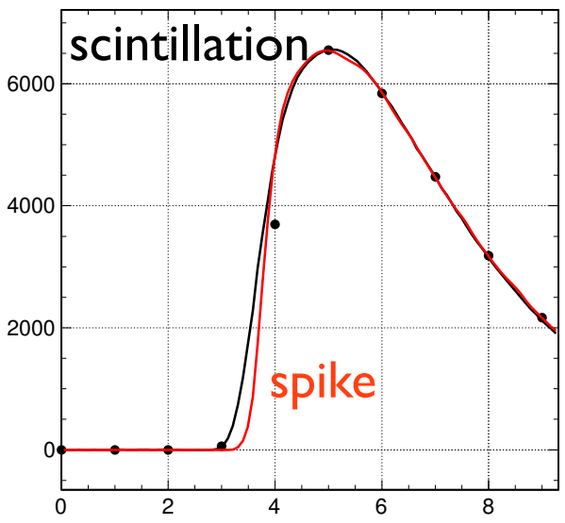
Is it a spike?

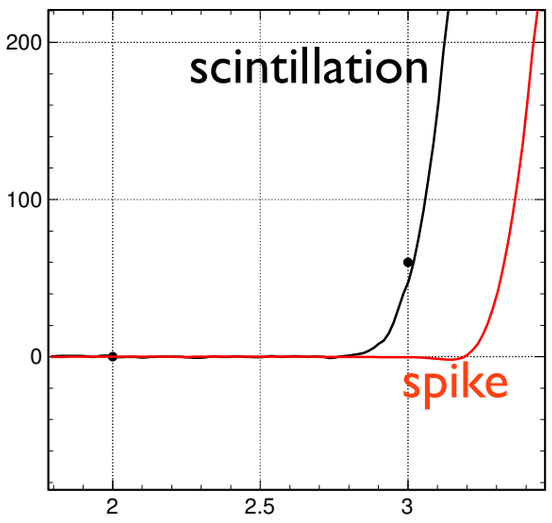
```
EB hit:  E=2187.41 E_t=2161.09 time=-26.4659 swisscross=0.865479 ieta=9 iphi=146 eta=0.155889
phi=2.36586 lasercalib=1.02996
RECHIT flags:
EcalRecHit::kGood=0
EcalRecHit::kPoorReco=0
EcalRecHit::kOutOfTime=0
EcalRecHit::kFaultyHardware=0
EcalRecHit::kNoisy=0
EcalRecHit::kPoorCalib=0
EcalRecHit::kSaturated=0
EcalRecHit::kLeadingEdgeRecovered=1
[...]
SEVERITY=2
```

Pulse shape

sample	01	02	03	**04**	05	06	07	08	09	10
ADC	202	201	201	**1680**	3986	4063	4063	4063	4063	4063
Gain	12	12	12	**12**	**1**	**0**	**0**	**0**	**0**	**0**

For illustration: Not the same rechit!





https://indico.cern.ch/getFile.py/access?contribId=3&sessionId=0&resId=0&materialId=slides&confId=107816

A. Ledovskoy, ECAL DPG, 4th Nov 2010

4th sample can be used to discriminate between pulses of spike and scintillation origin

(see previous talks by Sasha L.,Kostas etc...)

In this case, the situation is unclear:

Sample 4 is elevated - more scintillation-like

however, some scintillation component is expected - this is a rechit embedded in an EM shower

What do the likelihood estimators say for this hit?

Do we have enough non-saturated samples to say anything?

Actions

- Short timescale (~now)
  - Investigate with priority the events with gain 0 in sample 10. Hex dump created, to be examined by experts
  - Sign-off modification to EEBadSCFilter to remove these events
- Medium timescale (<6_1_0)
  - Optimise rechit flags based on signal pulse quality (e.g. kPoorReco). Assign the appropriate severity level to these flags, and use them to reject these hits in all physics objects. Replacement for ad-hoc filters
  - Turn on single channel recovery? (see talk this meeting)
  - Remove inconsistent treatment of rechit flags between Calo/PF/Egamma
- Longer timescale (6_1_0 and LS1)
  - Quality flags embedded in the objects themselves?
  - Full review/rationalisation of flagging code (where are flags being set, what are the interdependencies etc...)
  - Investigate the two bad SCs in EE. Power separately & reduce HV (LS1)