RAW2RecHit Unpacking Approach

Performance of Ecal local reconstruction in HLT going from RAW to RecHit

Jean-Roch Vlimant

Introduction

The default Ecal local reconstruction in HLT

BASE

- List of FEDs are created from L1 objects
  - If any FED# was already asked before: do not add it
- Ecal is digitized in the given FED#
- Uncalibrated/Calibrated RecHits are created
  - Two separate modules
  - Uncalibrated RH are **not used in HLT**
- Ecal RecHits created in this path are merged to any other (jets, muon, tau, egamma) made before
  - Trigger paths **are not independent**
  - Surperfluously large number of recHits.

The *RAW2RecHit* Ecal local reconstruction in HLT

https://twiki.cern.ch/twiki/bin/view/CMS/EcalRawToRecHit

R2R

- *RAW2RecHit* tools are created
- List of FEDs are created from L1 objects
  - or others (hltL2muons, ...)
- Calibrated RecHits are created for the given FED#
  - Any **FED# processed only once** (cached)
  - Digis as a possible side product
  - Trigger paths made **totally independent**

Outline

- Working release in 180p6
- Running HLT using

  https://twiki.cern.ch/twiki/bin/view/CMS/SWGuideGlobalHLT
- Using 180p3a, 180p4 RelVal samples (180p5/p6 are missing RAW/DIGIS ?)
  - Z->mumu
  - ttbar
  - bJets+cJets = b/c-jets

- Compare timing of path/module in the global HLT table
- Compare the number of reconstructed Ecal hits in each path

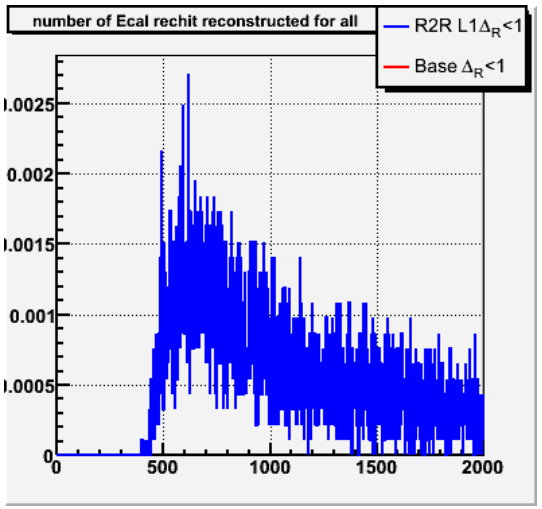
Code Status In CMSSW

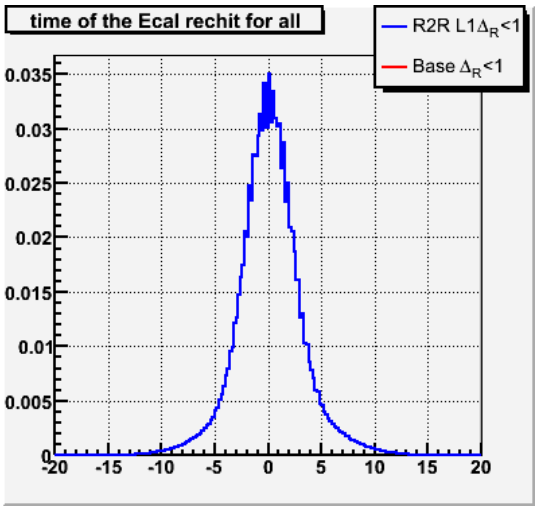
- Baseline requirements for *RAW2RecHit* are in 180p6
  - DataFormats/EcalRecHit V01-03-01
  - EventFilter/EcalRawToDigi V01-02-01
  - Works OK with these only
  - https://twiki.cern.ch/twiki/bin/view/CMS/EcalRawToRecHit

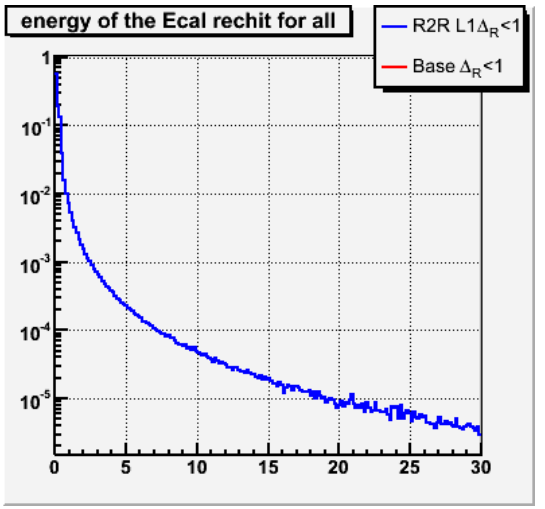
- Some major features have been queued to 18X
  - Due to "virtual" dependence on DataFormats/SiStripCommon waiting for a common package to host relevant C++ classes
  - DataFormats/EcalRecHit V01-04-00
  - EventFilter/EcalRawToDigi V01-03-00-02
  - Results shown in this presentation: using these tags, on top of 180p6

Local Reconstruction "Validation"

- Compare energy/timing of the full Ecal local reconstruction in ttbar events
- **All Y axis have arbitrary unit** (this slide and all others)







One sees no differences, because there are none

Paths That Matter

- First of the series
  - HLT1jet:jet/MET paths
  - HLT1Electron: eGamma paths
  - HLT1MuonIso: muon paths
  - HLT1Tau: tau paths

- The other paths re-use the Ecal module output
  - it's another level of study

- Compare timing from two different batch jobs
  - Put a dummy, computation expensive path in the schedule
  - Compare timing to the timing of this module/path

Z→mumu Events

- Looking only at the muon HLT path

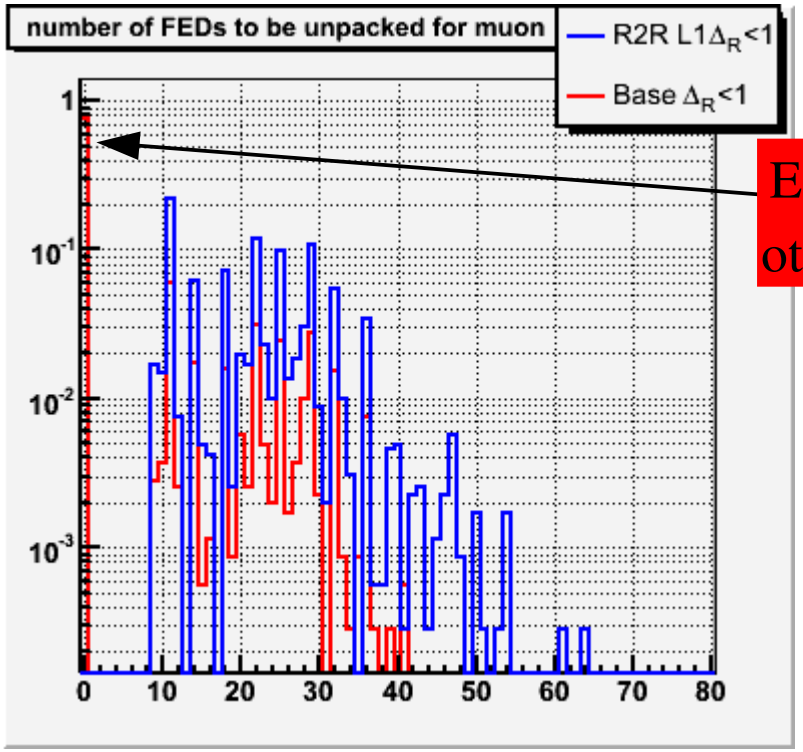
Z→mumu : HLT1MuonIso

- An isolated muon will trigger the Ecal isolation calculation
- 3475 events made it to the Ecal local reconstruction
- L2* : using hlt2MuonCandidate + propagator

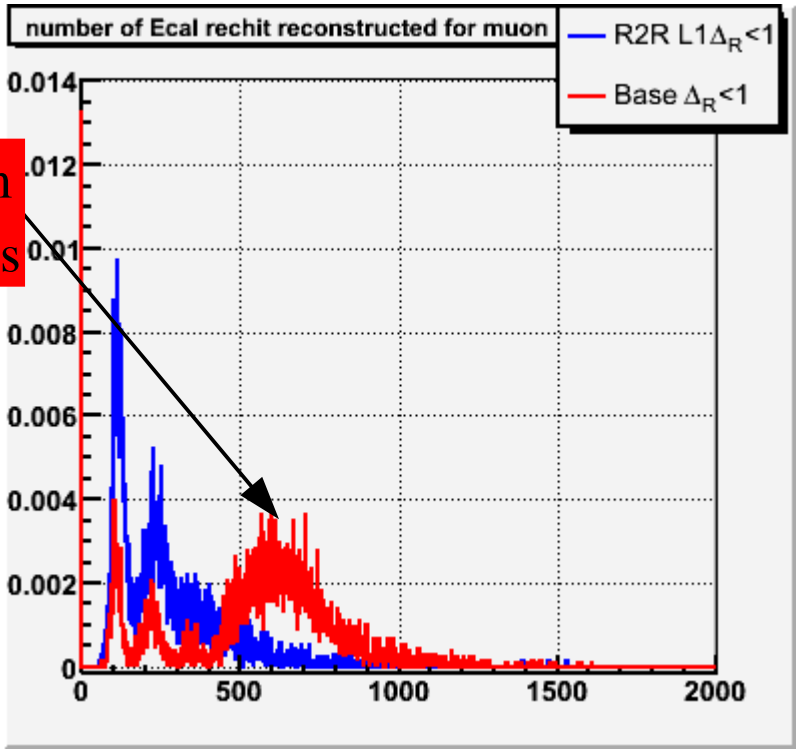
Mode	Ecal part (arbitrary unit)	Ratio to BASE	HL1MuonIso (arbitrary unit)	Ratio to BASE
BASE	0.7461	100%	3.3087	100%
R2R	0.1278	17%	2.8587	86%

- Most of time improvement comes from RecHitMerger (75% timing)
- Save ~80% timing of the burden on calorimeter in HLT1MuonIso
- Save ~15% in HLT1MuonIso total timing.

Z→mumu : Ecal Content for Muon



Ecal RecHit from other trigger paths



ttbar Events

- Looking only at
  - Jet HLT path
  - Electron HLT path
  - Muon HLT path

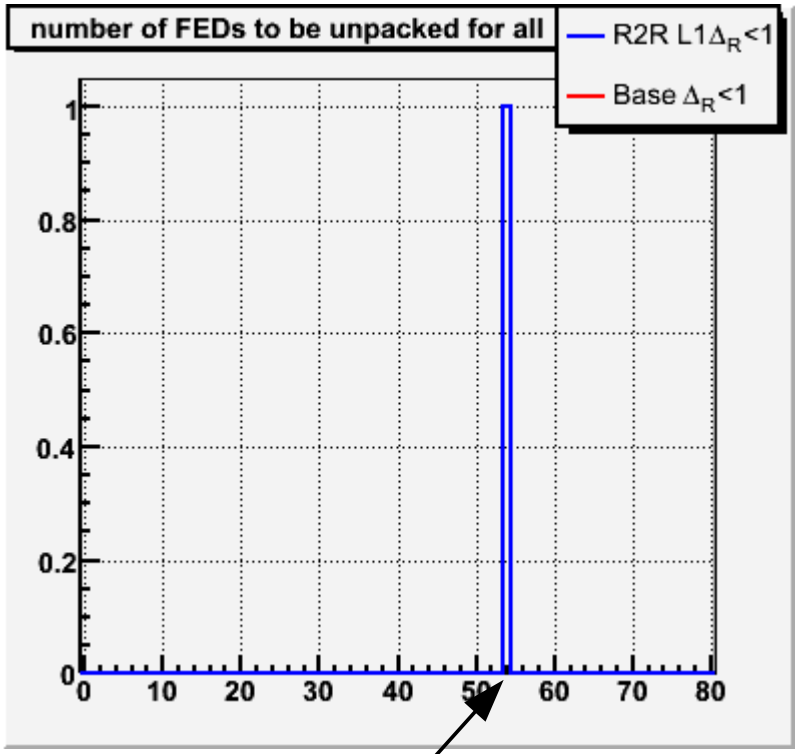
ttbar : HLT1jet

- 1175 events make it to the Ecal local reconstruction

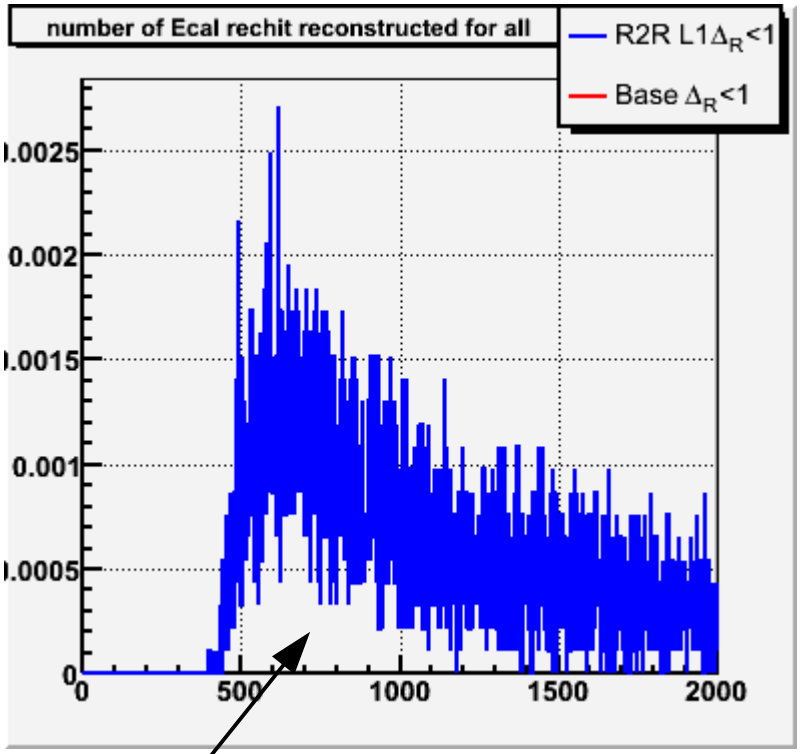
Mode	Ecal part (arbitrary unit)	Ratio to BASE	HL1jet (arbitrary unit)	Ratio to BASE
BASE	0.0768	100%	13.6667	100%
R2R	0.0090	12%	5.9441	43%

- Most of time improvement comes from RecHitMerger (85% timing)
- Save an **amazing ~60% on HLT1jet** timing

ttbar : Ecal Content for Jets



all FED are used in the jet path



no difference between R2R and BASE

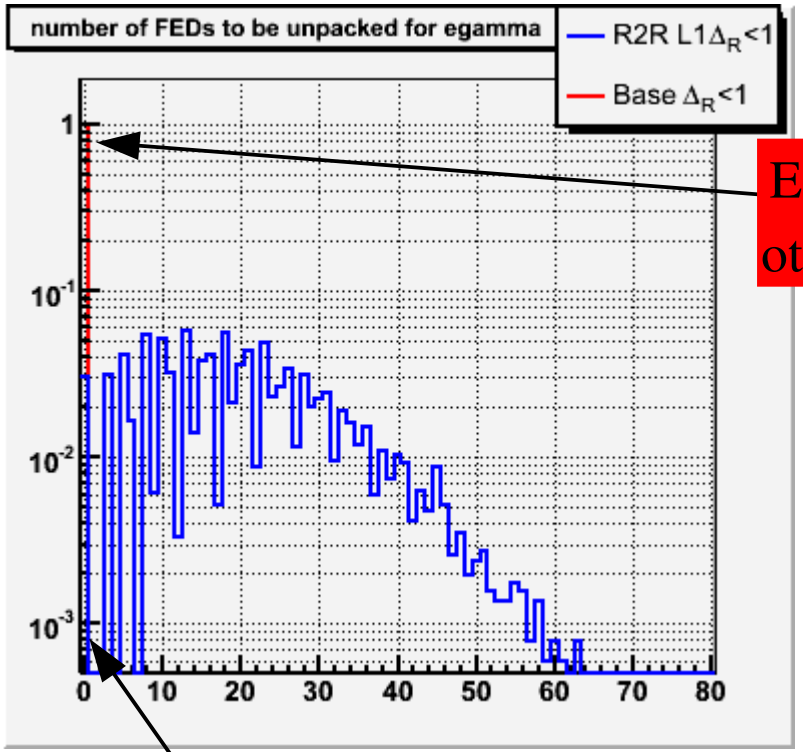
ttbar : HLT1Electron

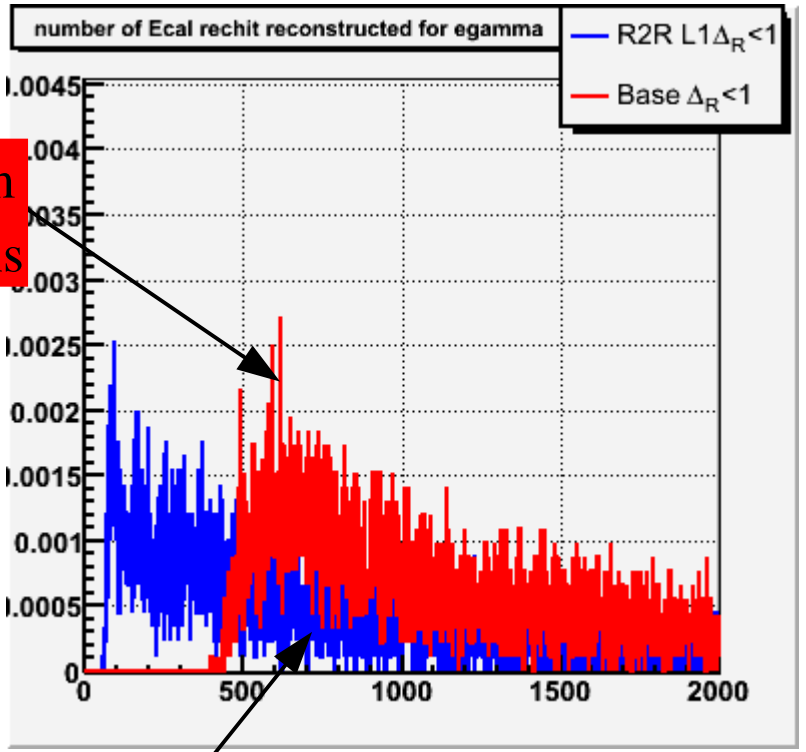
- 2299 events make it to the Ecal local reconstruction

Mode	Ecal part (arbitrary unit)	Ratio to BASE	HLT1Electron (arbitrary unit)	Ratio to BASE
BASE	0.0194	100%	1.3647	100%
R2R	0.0160	82%	1.2627	93%

- Most of time improvement comes from clustering modules
- Save ~20% timing of the burden on calorimeter in HLT1Electron
- Save ~10% in HLT1Electron total timing.

ttbar : Ecal Content for Electron





Ecal RecHit from other trigger paths

FED# for electron are fully covered by prior paths

but electron paths get too many RH

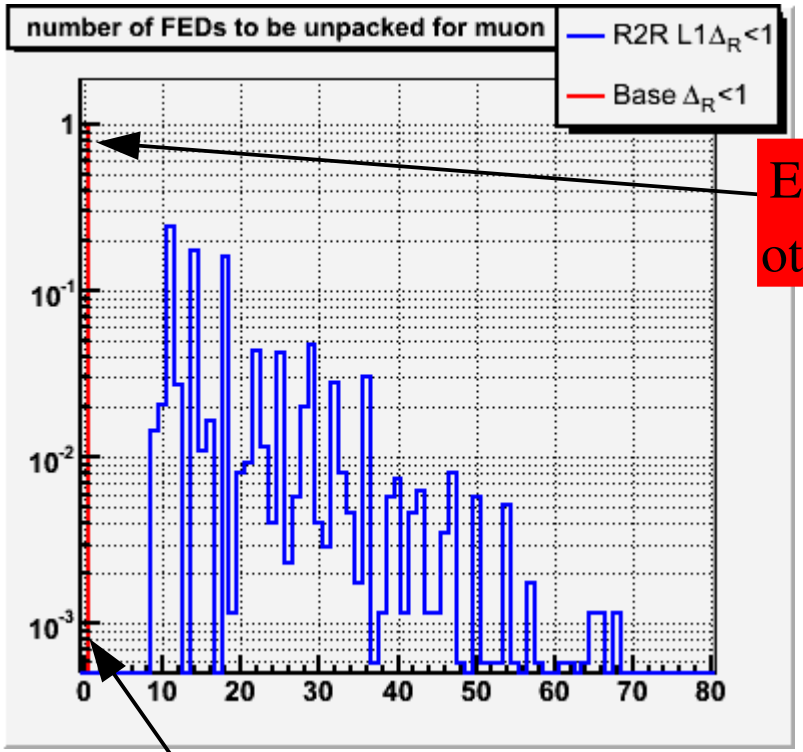
ttbar : HLT1MuonIso

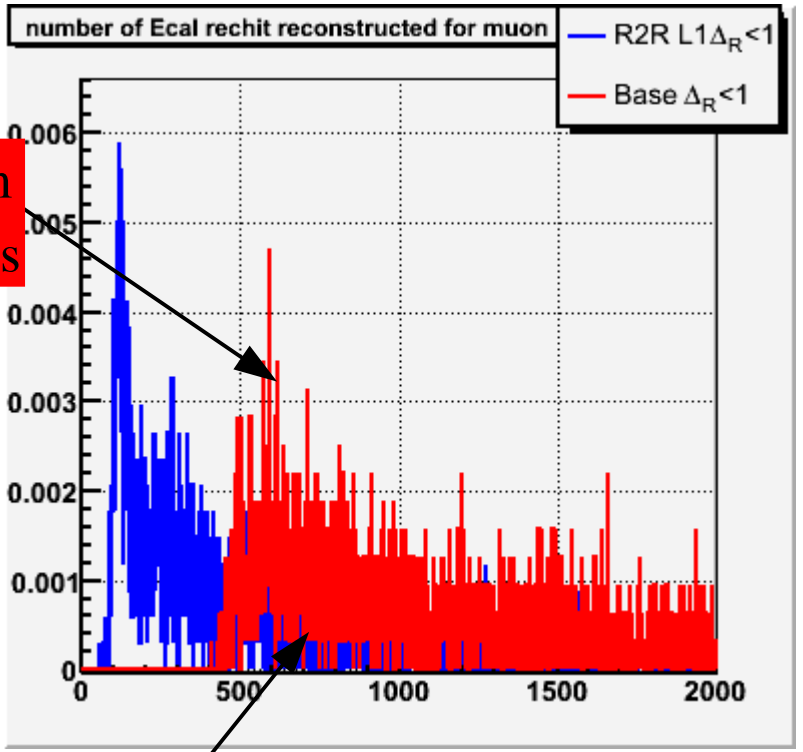
- 1427 events made it to the Ecal local reconstruction
- L2* : using hlt2MuonCandidate + propagator

Mode	Ecal part (arbitrary unit)	Ratio to BASE	HL1MuonIso (arbitrary unit)	Ratio to BASE
BASE	0.5615	100%	1.4285	100%
R2R	0.1194	21%	1.2613	88%

- Most of time improvement comes from RecHitMerger (75% timing)
- Save ~80% timing of the burden on calorimeter in HLT1MuonIso
- Save ~10% in HLT1MuonIso total timing

ttbar : Ecal Content for Muon





Ecal RecHit from other trigger paths

FED# for muon are fully covered by prior paths

but muon paths get too many RH

b/c-jets Events

- Looking only at
  - Jet HLT path
  - Electron HLT path
  - Muon HLT path

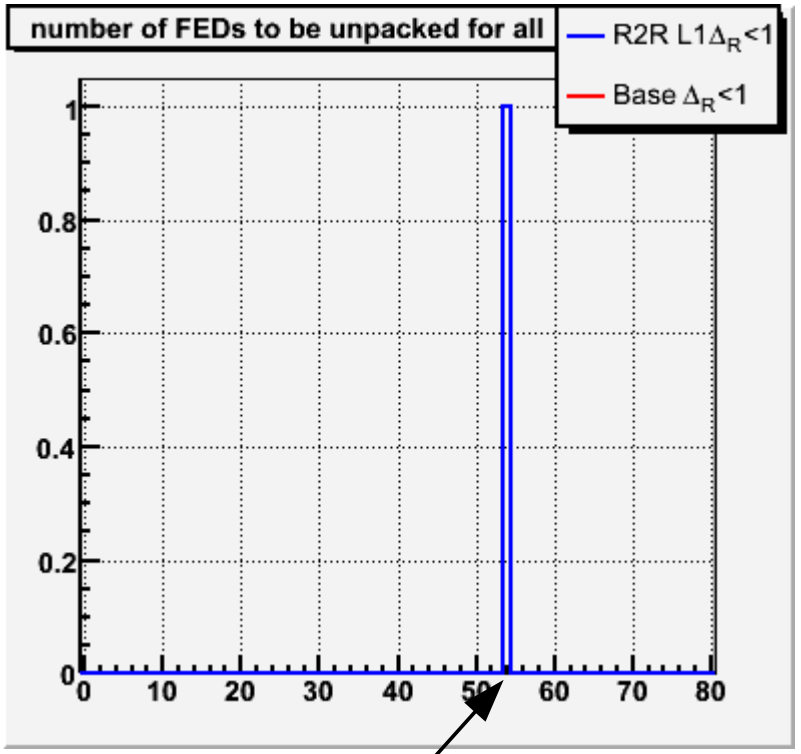
b/c-jets : HLT1jet

- ~1000 events make it to the Ecal local reconstruction

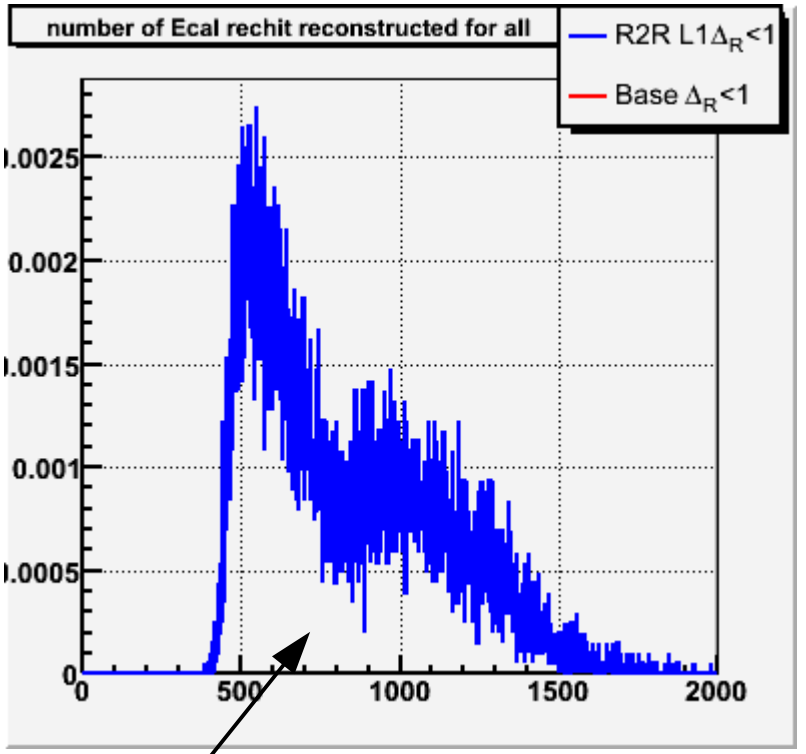
Mode	Ecal part (arbitrary unit)	Ratio to BASE	HL1jet (arbitrary unit)	Ratio to BASE
BASE	0.0766	100%	7.4576	100%
R2R	0.0245	32%	4.9119	66%

- Most of time improvement comes from RecHitMerger (55% timing)
- Save an ~35 % on HLT1jet timing

b/c-jets : Ecal Content for Jets



all FED are used in the jet path



no difference between R2R and BASE

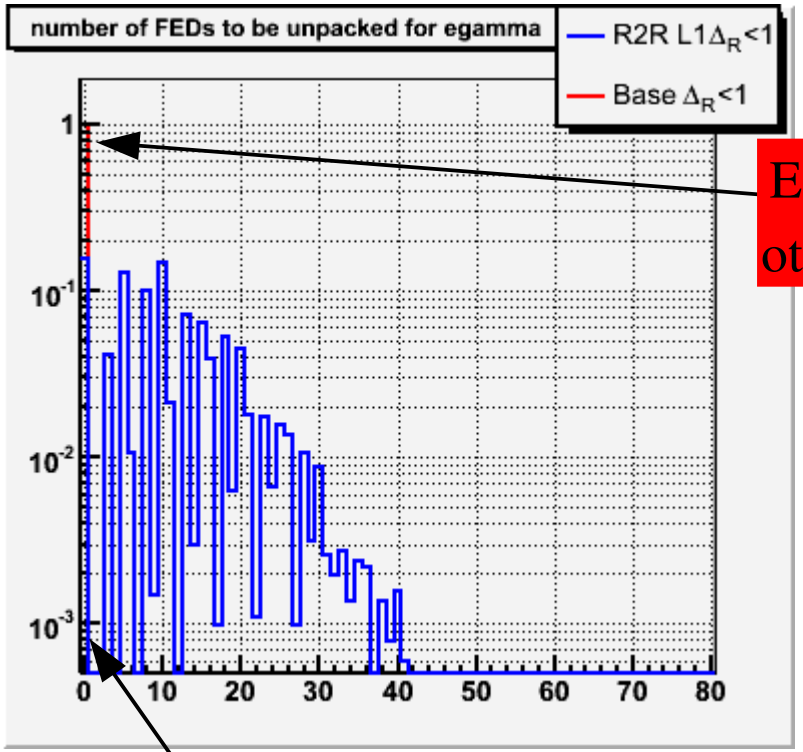
b/c-jets : HLT1Electron

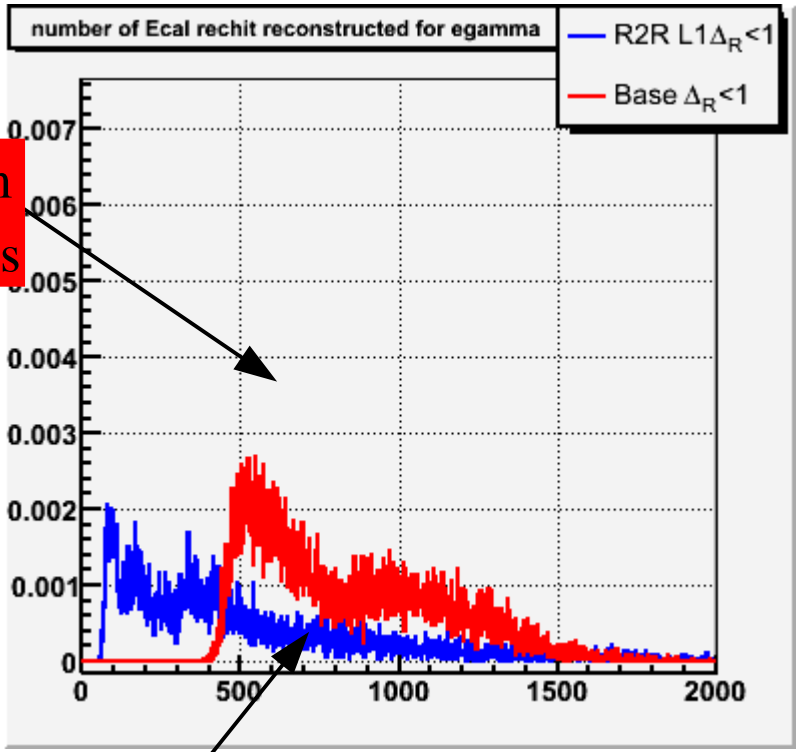
- 2299 events make it to the Ecal local reconstruction

Mode	Ecal part (arbitrary unit)	Ratio to BASE	HLT1Electron (arbitrary unit)	Ratio to BASE
BASE	0.0194	100%	1.3647	100%
R2R	0.0160	82%	1.2627	93%

- Most of time improvement comes from clustering modules
- Save ~20% timing of the burden on calorimeter in HLT1Electron
- Save ~10% in HLT1Electron total timing.

b/c-jets : Ecal Content for Electron





Ecal RecHit from other trigger paths

FED# for electron are fully covered by prior paths

but electron paths get too many RH

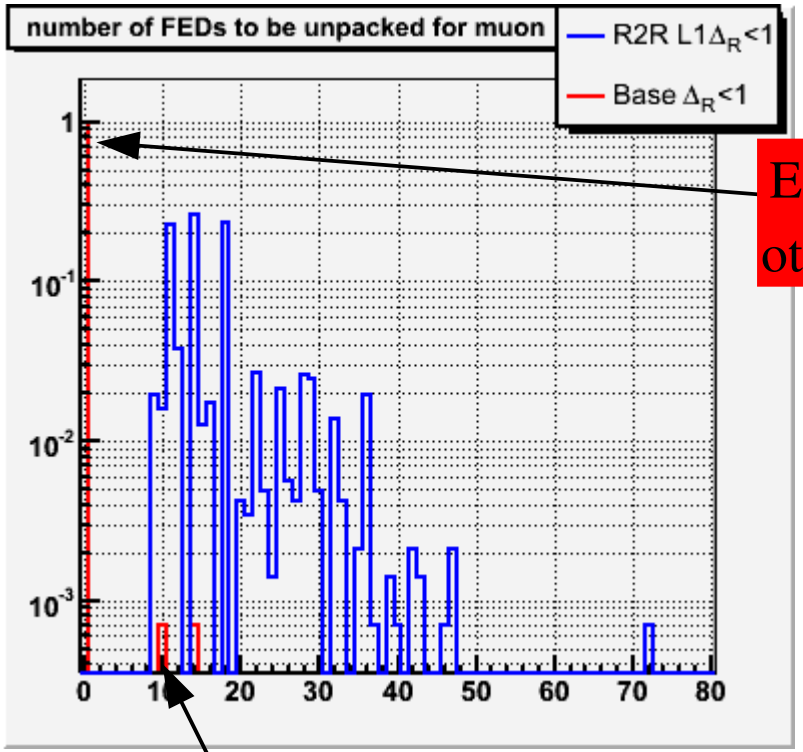
b/c-jets : HLT1MuonIso

- 1427 events made it to the Ecal local reconstruction
- L2* : using hlt2MuonCandidate + propagator

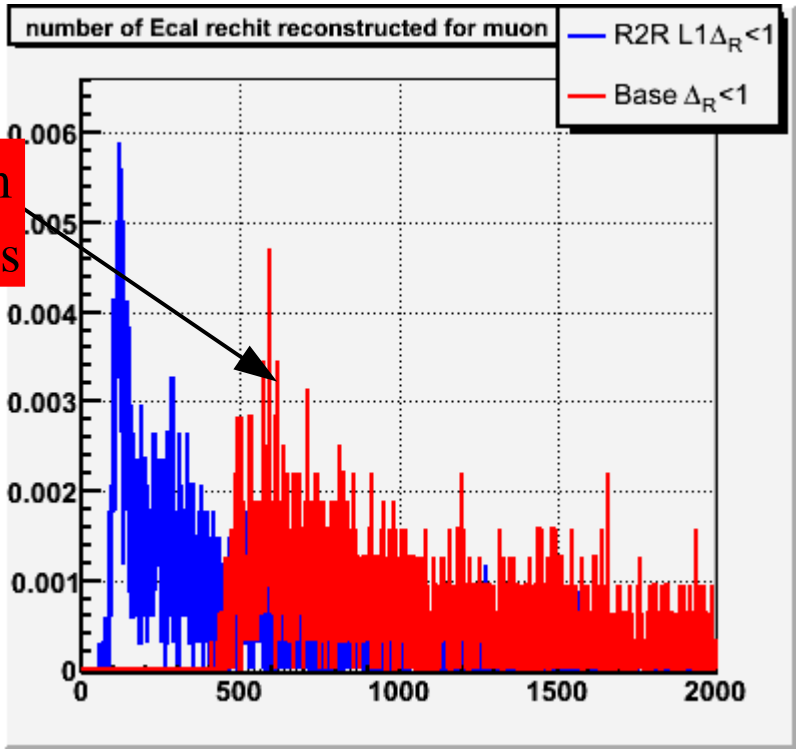
Mode	Ecal part (arbitrary unit)	Ratio to BASE	HL1MuonIso (arbitrary unit)	Ratio to BASE
BASE	0.0337	100%	0.1021	100%
R2R	0.0274	81%	0.0984	96%

- Save ~5% in HLT1MuonIso total timing

b/c-jets : Ecal Content for Muon



Ecal RecHit from other trigger paths



few events with additional FED# for muon paths

Conclusions

- Ecal *RAW2RecHit* is more flexible
  - ➔ not only seeding on L1: room for optimization
- Ecal *RAW2RecHit* makes trigger paths independent
  - ➔ more robust trigger menu
- Ecal *RAW2RecHit* is integrated in 18X
  - ➔ some additional features in the pipeline
- Ecal *RAW2RecHit* makes HLT faster
  - ➔ 10-60% timing improvement confirmed

https://twiki.cern.ch/twiki/bin/view/CMS/EcalRawToRecHit

Next

- Follow up on code integration
  - Do we want the latest-greatest in 18X ?
  - Support creation of a common DataFormats
- Room for optimization **within** *RAW2RecHit*
  - Skipping intermediate steps a la SiStrip
  - need to identify bottlenecks
  - ...
- Room for optimization outside of *RAW2RecHit*
  - Muon isolation from L2 tracks
  - ...

Backup slides

- More detail of muon timing and possible improvements
- HLT baseline Ecal local reconstruction
- *RAW2RecHit* step by step in details

Z→mumu : HLT1MuonIso

- An isolated muon will trigger the Ecal isolation calculation
- 3475 events made it to the Ecal local reconstruction
- L2* : using hlt2MuonCandidate + propagator

Mode	Seed	ΔR	Ecal part (arbitrary unit)	Ratio to BASE	HL1MuonIso (arbitrary unit)	Ratio to BASE	accept
BASE	L1	<1	0.7461	100%	3.3087	100%	3426
		<0.5	0.7528	101%	3.2967	100%	3426
		<0.25	0.6754	91%	3.2651	99%	3428
R2R	L1	<1	0.1278	17%	2.8587	86%	3423
		<0.5	0.1095	15%	2.8756	87%	3424
		<0.25	0.0980	13%	2.8196	85%	3427
R2R	L2*	<1	0.1174	16%	2.7827	84%	3426
		<0.5	0.1096	15%	2.8645	87%	3426
		<0.25	0.0789	11%	2.6118	79%	3427

- Most of time improvement comes from RecHitMerger (75% timing)
- Save ~80% timing of the burden on calorimeter in HLT1MuonIso
- Save ~15% in HLT1MuonIso total timing.

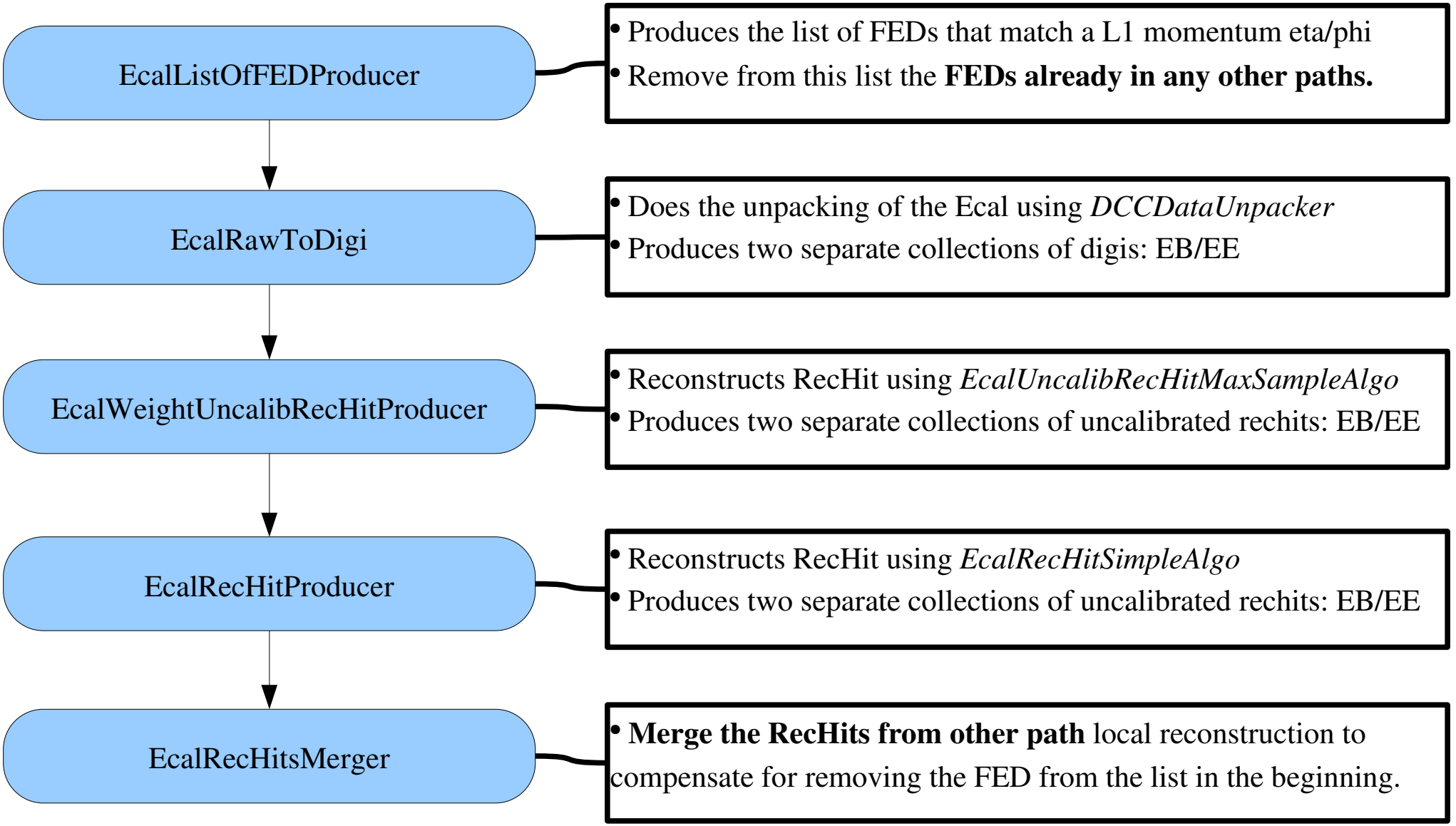
ttbar : HLT1MuonIso

- 1427 events made it to the Ecal local reconstruction
- L2* : using hlt2MuonCandidate + propagator

Mode	Seed	ΔR	Ecal part (arbitrary unit)	Ratio to BASE	HL1MuonIso (arbitrary unit)	Ratio to BASE	accept
BASE	L1	<1	0.5615	100%	1.4285	100%	845
		<0.5	0.5224	93%	1.3536	95%	314/543
		<0.25	0.5762	103%	1.4316	100%	852
R2R	L1	<1	0.1194	21%	1.2613	88%	835
		<0.5	0.1088	19%	1.3107	92%	838
		<0.25	0.1011	18%	3.9038	273%	850
R2R	L2*	<1	0.1218	22%	1.2948	91%	840
		<0.5	0.1046	19%	1.2807	90%	842
		<0.25	0.1036	18%	1.2945	91%	843

- Most of time improvement comes from RecHitMerger (75% timing)
- Save ~80% timing of the burden on calorimeter in HLT1MuonIso
- Save ~10% in HLT1MuonIso total timing

Baseline Ecal Local Reco (HLT)



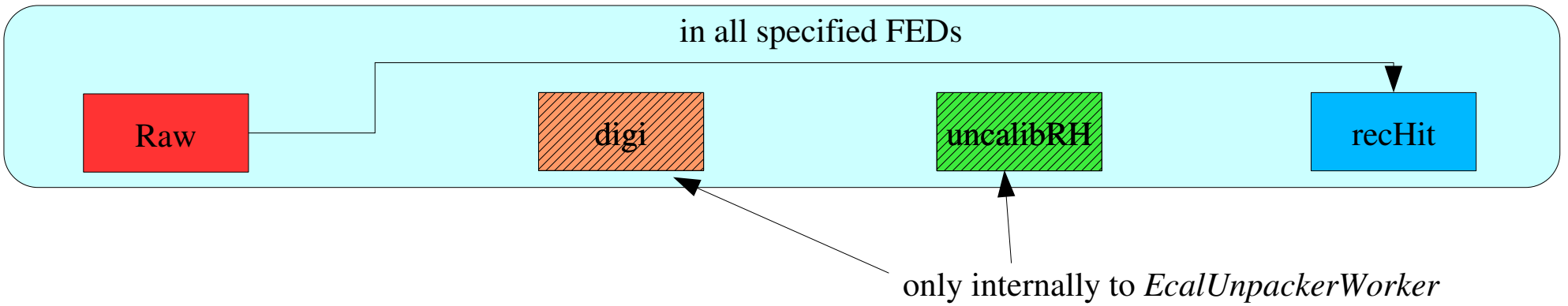
More details on merging in previous presentation from B. Dahmes (and E. Perez)
http://indico.cern.ch/conferenceDisplay.py?confId=22029

Ecal Local Reconstruction

Shortest description

- Baseline Reco
  - list of FEDs in baseline does not contain the FEDS number already processed

- RawToRecHit
  - list of FEDs is not modified for FED already processed in another path
  - list of FEDs is plain L1Object-eta/phi ⇔ FED# matching
  - Using the same classes as in the baseline:
    DCCDataUnpacker, *EcalUncalibRecHitMaxSampleAlgo*, *EcalRecHitSimpleAlgo*
    hold by new class *EcalUnpackerWorker*



Ecal RawToRecHit

Detailed description

STEP 1 (prepare)

prepare the tools, **no unpacking**

- *EcalRawToRecHitFacility*
  - Module similar to *SiStripRawToClusters* (called SiStripRawToClusterFacility in cfg)
  - Interfaces the RAW data with *EcalRawToRecHitLazyUnpacker*
    - Class adapted from *SiStripRawToClustersLazyUnpacker*
    - Requires a class *EcalUnpackerWorker,* holding the unpacker and recHit algorithms.
  - Put a *EcalRecHitLazyGetter* in the event
    - Actually *edm::SiStripLazyGetter<EcalRecHit>,* which has nothing to do with SiStrip

•EDProduct:
edm::SiStripLazyGetter<EcalRecHit>

Ecal RawToRecHit

Detailed description

STEP 2 (define)

define the FEDs to be unpacked (regions), **no unpacking**

- *EcalRawToRecHitRoI*
  - same old list of FEDs from L1EmParticle, L1JetParticle, L1MuonParticle as in *EcalListOfFEDProducer*
  - **do not remove any FED#** that has been done already in other paths
  - Works also on any class inheriting from reco::Candidate
  - Retrieve the *EcalRecHitLazyGetter* from the event
  - Defines the FED regions to a *EcalRecHitRefGetter*
    - Actually *edm::SiStripRefGetter<EcalRecHit>,* which has nothing to do with SiStrip
  - Put the *EcalRecHitRefGetter* in the event

EDProduct:
EcalListOfFEDS
EcalRecHitRefGetter

Ecal RawToRecHit

Detailed description

STEP 3 (unpack)

unpacking and local reconstruction

- *EcalRawToRecHitProducer*
  - Retrieve the *EcalRecHitRefGetter* in the event
  - Get the recHits from the RefGetter: unpacking is automatically done
  - put collection of EcalRecHit in the event
    - splitted EB/EE **or not**

If a FED# is requested in more than one path. It will be unpacked during the first module to be run. The nexts will get the previous result. Stored in the LezyGetter.

EDProduct:
EBRecHitCollection
EERecHitCollection