

# Table of Contents

<b>Tracking Studies with.....</b>	<b>1</b>
<b>Setup.....</b>	<b>2</b>
<b>Samples (with parentage).....</b>	<b>3</b>
Min Bias (MIT, ph_analysis_01).....	3
Pion Flat pt = 0 - 200 (MIT, Private).....	3
Ntuples & Histograms.....	3
Pythia pthat > 170 (MIT, Private) : Submitted.....	3
HLT_HIJet50U, 0 - 10%.....	3
Pion Flat pt = 0 - 200 (MIT, Private), pion pt < jet pt, for jets in all acceptance and all pt.....	3
Pion Flat pt = 0 - 200 (MIT, Private), pion pt not limited by jet pt, jets letal < 2.8 , pt > 30.....	4
HLT_HIJet50U.....	4
0 - 10%.....	4
<b>Progress Log.....</b>	<b>5</b>
pt distribution of MC input : Unresolved.....	5
: Track into jet or bad APV mixing : Unresolved.....	5
Sanity of results : Unresolved.....	5
High pT efficiency : October 6, 2011 : Unresolved.....	5
Memory Problem : October 6, 2011 : Unresolved.....	5
Status : October 6, 2011.....	6
HBHE crash : September 2011 : Solved by removing the ZDC part and use digi mixing.....	6
_ecalPreShowerDigis : September 2011 : Unresolved.....	6
: September 2011 : Resolved.....	6
: September 2011 : Resolved.....	7
Changing the main loop that produces the output rawDigis.....	7
Memory optimization of temporary vectors and maps : Resolved.....	7
ZDC change : Resolved.....	7
Production.....	7
Input RAW sample : September 2011 : Resolved.....	7
Preparing crab jobs : September 2011 : Testing.....	7
Debugging with all sub-detectors included.....	7
detector condition coustomization.....	8

# Tracking Studies with

# Setup

Currently working on 4\_4\_0.

```
cvns co -d edwenger/HiVertexAnalyzer UserCode/edwenger/HiVertexAnalyzer
cvns co -d edwenger/HiTrkEffAnalyzer UserCode/edwenger/HiTrkEffAnalyzer
cvns co -d MitHig/PixelTrackletAnalyzer UserCode/MitHig/PixelTrackletAnalyzer/
cvns co -r hi44X_04 -d SimGeneral/DataMixingModule UserCode/etkin/SimGeneral/DataMixingModule
cvns co HeavyIonsAnalysis/Configuration
cvns co -d CmsHi/Analysis2010 UserCode/CmsHi/Analysis2010
cvns co -d PbPbTrackingTools/HiTrackValidator UserCode/ASYoon/PbPbTrackingTools/HiTrackValidator
cvns co -d PbPbTrackingTools/VertexConstraintProducer UserCode/ASYoon/PbPbTrackingTools/VertexCons
```

**Problematic python configuration:**

/net/hisrv0001/home/etkin/hidisk0001/mix/cal01\_d1123/notWorkingCfg\_withHcal.py

# Samples (with parentage)

## Mother of All : RAW (MIT, Global)

- /HICorePhysics/StoreResults-HICorePhysics\_Skim\_MinimumBias\_RAW-a606dc809a29a92e17749e5652319
- /pnfs/cmsaf.mit.edu/t2bat/cms/store/results/heavy-ions/HICorePhysics/StoreResults-HICorePhysics\_Skim\_Mi

## Min Bias (MIT, ph\_analysis\_01)

- /HICorePhysics/yilmaz-MinBias\_PreMix\_440\_test26-2a3fd0a01656fd762e353d50bd946d2c/USER
- /pnfs/cmsaf.mit.edu/t2bat/cms/store/user/yilmaz/HICorePhysics/MinBias\_PreMix\_440\_test28/2a3fd0a01656fd
- Scripts: svmithi01:/d101/yetkin/prod/test29\_mbPreMix

## Pion Flat pt = 0 - 200 (MIT, Private)

- /pnfs/cmsaf.mit.edu/t2bat/cms/store/user/yetkin/mix/pions1\_HICorePhysics\_MinBiasSkim\_440\_test26\_DBS\_
- Scripts: /net/hisrv0001/home/yetkin/hidsk0001/mix/pion1\_DBS\_d0931

## Ntuples & Histograms

- /net/hisrv0001/home/yetkin/hidsk0001/mix/tracks1\_DBS\_d0931/test.root
- Scripts : /net/hisrv0001/home/yetkin/hidsk0001/mix/tracks1\_DBS\_d0931
- /net/hisrv0001/home/yetkin/hidsk0001/mix/tracks1\_DBS\_d0931/signal01/signal.root
- Scripts : /net/hisrv0001/home/yetkin/hidsk0001/mix/tracks1\_DBS\_d0931/signal01

## Pythia pthat > 170 (MIT, Private) : Submitted

- /pnfs/cmsaf.mit.edu/t2bat/cms/store/user/yetkin/mix/pythia170\_HICorePhysics\_MinBiasSkim\_440\_test28\_DE
- Scripts : /net/hisrv0001/home/yetkin/hidsk0001/mix/pythia170\_d0931

## Ntuples & Histograms

- /net/hisrv0001/home/yetkin/hidsk0001/mix/tracks\_pythia170\_d1009/test.root
- Scripts : /net/hisrv0001/home/yetkin/hidsk0001/mix/tracks\_pythia170\_d1009
- /net/hisrv0001/home/yetkin/hidsk0001/mix/tracks\_pythia170\_d1009/signal01/signal.root
- Scripts : /net/hisrv0001/home/yetkin/hidsk0001/mix/tracks\_pythia170\_d1009/signal01

## HLT\_HIJet50U, 0 - 10%

- /HICorePhysics/yilmaz-Jet50U\_0to10\_PreMix\_440\_test32-f17fd2ed8b87d0940b5b303282977643/USER
- Scripts: /d101/yetkin/prod/test32\_Jet50U\_0to10\_preMix

## Pion Flat pt = 0 - 200 (MIT, Private), pion pt < jet pt, for jets in all acceptance and all pt.

- /pnfs/cmsaf.mit.edu/t2bat/cms/store/user/yetkin/mix/probeJet50U\_0to10\_PreMix\_440\_test32v01
- Scripts: /net/hisrv0001/home/yetkin/hidsk0001/mix/probeJet\_d1008

## Ntuples & Histograms

- /net/hisrv0001/home/yetkin/hidsk0001/mix/tracks\_probeJet\_d1009/trees\_\*.root
- Scripts : /net/hisrv0001/home/yetkin/hidsk0001/mix/tracks\_probeJet\_d1009
- /net/hisrv0001/home/yetkin/hidsk0001/mix/tracks\_probeJet\_d1009/signal01/signal\_\*.root
- Scripts : /net/hisrv0001/home/yetkin/hidsk0001/mix/tracks\_probeJet\_d1009/signal01

**Pion Flat pt = 0 - 200 (MIT, Private), pion pt not limited by jet pt, jets  $|\eta| < 2.8$  , pt > 30**

- /pnfs/cmsaf.mit.edu/t2bat/cms/store/user/yetkin/mix/probeJet50U\_0to10\_PreMix\_440\_test32v02
- Scripts: /net/hisrv0001/home/yetkin/hidsk0001/mix/probeJet\_d1010

#### **Ntuples & Histograms**

- /net/hisrv0001/home/yetkin/hidsk0001/mix/tracks\_probeJet\_d1010/trees\_\*.root
- Scripts : /net/hisrv0001/home/yetkin/hidsk0001/mix/tracks\_probeJet\_d1010
- /net/hisrv0001/home/yetkin/hidsk0001/mix/tracks\_probeJet\_d1010/signal01/signal\_\*.root
- Scripts : /net/hisrv0001/home/yetkin/hidsk0001/mix/tracks\_probeJet\_d1010/signal01

#### **HLT\_HIJet50U**

Not ready

**0 - 10%**

Not ready

# Progress Log

## pt distribution of MC input : Unresolved

Currently we embed pions with flat pt. How good is this, when we want to make comparisons with data? We can start embedding pythia events, however, if the embedding is done in the direction of the jet, embedding another full jet into it may not be good. If we want a realistic pt distribution in particle gun events, should we use the PtYDistributorGun, or should we keep producing flat and reweight in analysis? If we will use the PtYDistributorGun, Andre has to generate a text input file such as:

<http://cmssw.cvs.cern.ch/cgi-bin/cmssw.cgi/CMSSW/GeneratorInterface/Pythia6Interface/data/jpsipbpb.txt?revision=1>

## : Track into jet or bad APV mixing : Unresolved

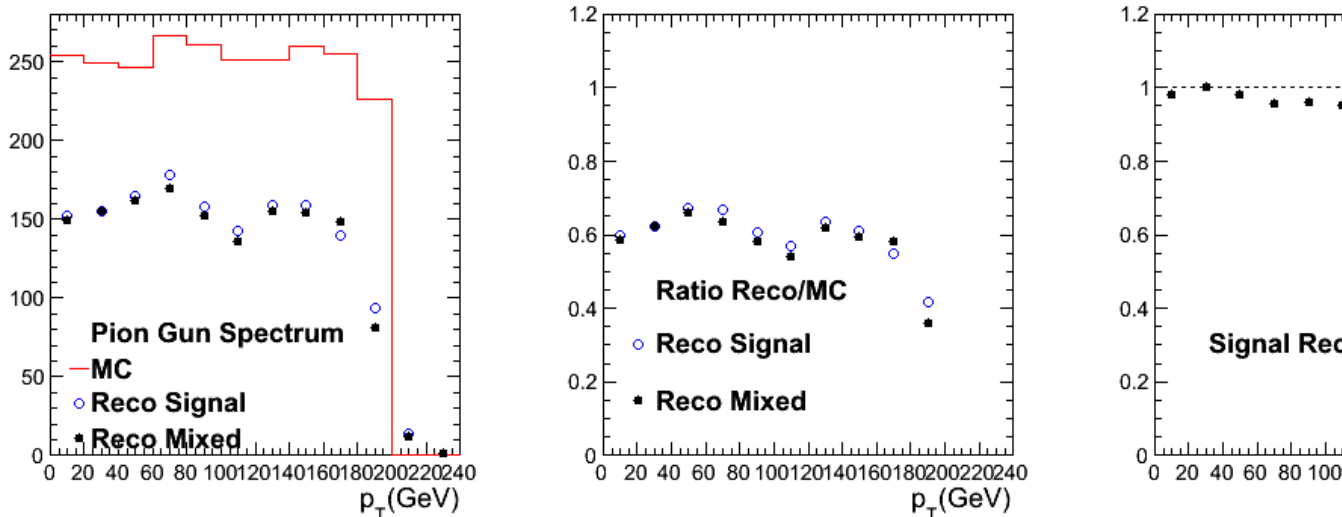
Yetkin will implement a generator that creates the pion in the direction of the jet or an interesting APV.

## Sanity of results : Unresolved

We need to compare the track parameter distributions (particularly those related to vertex compatibility) and check how realistic they are.

## High pT efficiency : October 6, 2011 : Unresolved

The tracking efficiency (by rough pt spectrum comparison) drops very rapidly at highest pt in the flat pt pion sample. The pt resolution is not sufficient to explain the effect:



- Need to plot efficiency properly.
- Need to investigate issues with vertexing.

## Memory Problem : October 6, 2011 : Unresolved

Currently running pion gun with more than 20 events per job causes a crash. Sometimes crash is explicitly reporting `std::bad_alloc`, sometimes not as clear. The `VectorInputSource` implementation in 440 was promising better performance. Is there a leak somewhere?

Note: It is crucial how many events per job are run, because one input file can be only consumed by single job.

## Status : October 6, 2011

Slides [↗](#)

## HBHE crash : September 2011 : Solved by removing the ZDC part and use digi mixing

- configuration :  
/net/hidsk0001/d00/scratch/jazzitup/eventMixing/CMSSW\_4\_4\_2\_patch5/test/test1/notWorkingCfg\_withHcal
- Error message

```
23-Nov-2011 10:40:01 EST Successfully opened file file:/mnt/hadoop/cms/store/user/yilmaz/HICoreP
shared pointer ok
pointer ok
done
isource 1
isource 2
isource 3
%MSG-e FatalSystemSignal: DataMixingModule:mixData 23-Nov-2011 10:40:02 EST Run: 152047 Event:
A fatal system signal has occurred: segmentation violation
%MSG
```

```
A fatal system signal has occurred: segmentation violation
The following is the call s
```

```
#0 0x0000003f5289a115 in waitpid () from /lib64/libc.so.6
#1 0x0000003f5283c481 in do_system () from /lib64/libc.so.6
#2 0x00002b899b01a2a2 in TUnixSystem::StackTrace() ()
   from /afs/cern.ch/cms/slc5_amd64_gcc434/cms/cmssw/CMSSW_4_4_0/external/slc5_amd64_gcc434/lib/1
#3 0x00002b899c594b33 in sig_dostack_then_abort ()
   from /afs/cern.ch/cms/slc5_amd64_gcc434/cms/cmssw/CMSSW_4_4_0/lib/slc5_amd64_gcc434/libFWCoreS
#4
```

## \_ecalPreShowerDigis : September 2011 : Unresolved

The standard event content of datamixing (DATAMIXEREventContent) has:

```
'keep ESDataFramesSorted_ecalPreshowerDigis_*_*',
```

but datamixer asks for:

```
ESDigiCollection_ecalPreShowerDigis
```

I added keep of the DigiCollection as well to my cfg.

## : September 2011 : Resolved

The parameter passed into the bind should be the number of events, not the bunch crossing:

```
boost::bind(&DataMixingModulle::pileWorker, boost::ref(*this),
            _1, bunchCrossing, _2, boost::cref(ES)),
```

Temporarily this works:

```
boost::bind(&DataMixingModule::pileWorker, boost::ref(*this),
            _1, 1, _2, boost::cref(ES)),
```

## : September 2011 : Resolved

### Changing the main loop that produces the output rawDigis

At the moment the code is generating as output the collection of the data and on top of that is simply added the signal of MC. This creates inconsistency in the reco using the MC GT. The code should be able to do the opposite and add on top of the MC modules, the data modules ads.

	Looping over	adding	to produce
Previously the code was	Data rawDigis	MC Digis	MC rawDigis
Now we want	MC Digis	Data rawDigis	MC rawDigis

### Memory optimization of temporary vectors and maps : Resolved

The changes that were made to provide the above introduced big memory problem. It produces huge vectors and maps. We have to optimize the code. The problem was due to forgetting to increment the iterator and filling up infinitely! In any case, we made the code more efficient by using DetSetVector properly instead of homemade map.

### ZDC change : Resolved

Head of DataMixingModule can only be used with 4\_4\_X due to ZDCRecHit dataformat change.

## Production

### Input RAW sample : September 2011 : Resolved

We use minimum bias skim of HICorePhysics. The reason I couldn't find the RAW before is I only looked for AllPhysics out of habit.

### Preparing crab jobs : September 2011 : Testing

In interactive jobs, we guarantee the equality of the primary and secondary source by adding the line:

```
process.mixData.input.fileNames = process.source.fileNames
```

into the cfg file. However, crab executes the input cfg file only in the job preparation, and the input file name actually changes later for the job in the working node. One has to add this line back again to the final cfg that crab creates.

Solution: When preparing crab jobs execute:

<http://cmssw.cvs.cern.ch/cgi-bin/cmssw.cgi/UserCode/yetkin/HIProdAlgos/scripts/setMix.sh?view=markup>

### Debugging with all sub-detectors included

- Working directory : /d101/kimy/cmssw/photonStudy/CMSSW\_5\_0\_0\_pre6/src
- Configuration is divided into two step ( as last year)



- ◆ GEN-SIM-Digi : /d101/kimy/cmssw/photonStudy/CMSSW\_4\_4\_2\_patch5/test/mix/cfg.py  
<= working fine
- ◆ MIX-DIGI-RECO : /d101/kimy/cmssw/photonStudy/CMSSW\_4\_4\_2\_patch5/test/reco/cfg.py  
<<= not working \* Problem not resolved in running MIX level.

```
ecmsRun: /build/zafar/442/slc5_amd64_gcc434/cms/cmssw/CMSSW_4_4_2/src/SimCalorimetry/HcalTrigPrim  
erPrimitiveDigi&, float): Assertion `shift >= 0' failed.
```

## detector condition customization

- New detector condition customization file

[http://cmssw.cvs.cern.ch/cgi-bin/cmssw.cgi/UserCode/EYKim/DataMixer/DataMixer\\_DataConditions\\_4\\_4\\_X\\_data20](http://cmssw.cvs.cern.ch/cgi-bin/cmssw.cgi/UserCode/EYKim/DataMixer/DataMixer_DataConditions_4_4_X_data20)

---

This topic: CMSPublic > HiDataMixer2011

Topic revision: r16 - 2011-12-02 - YongsunKim



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