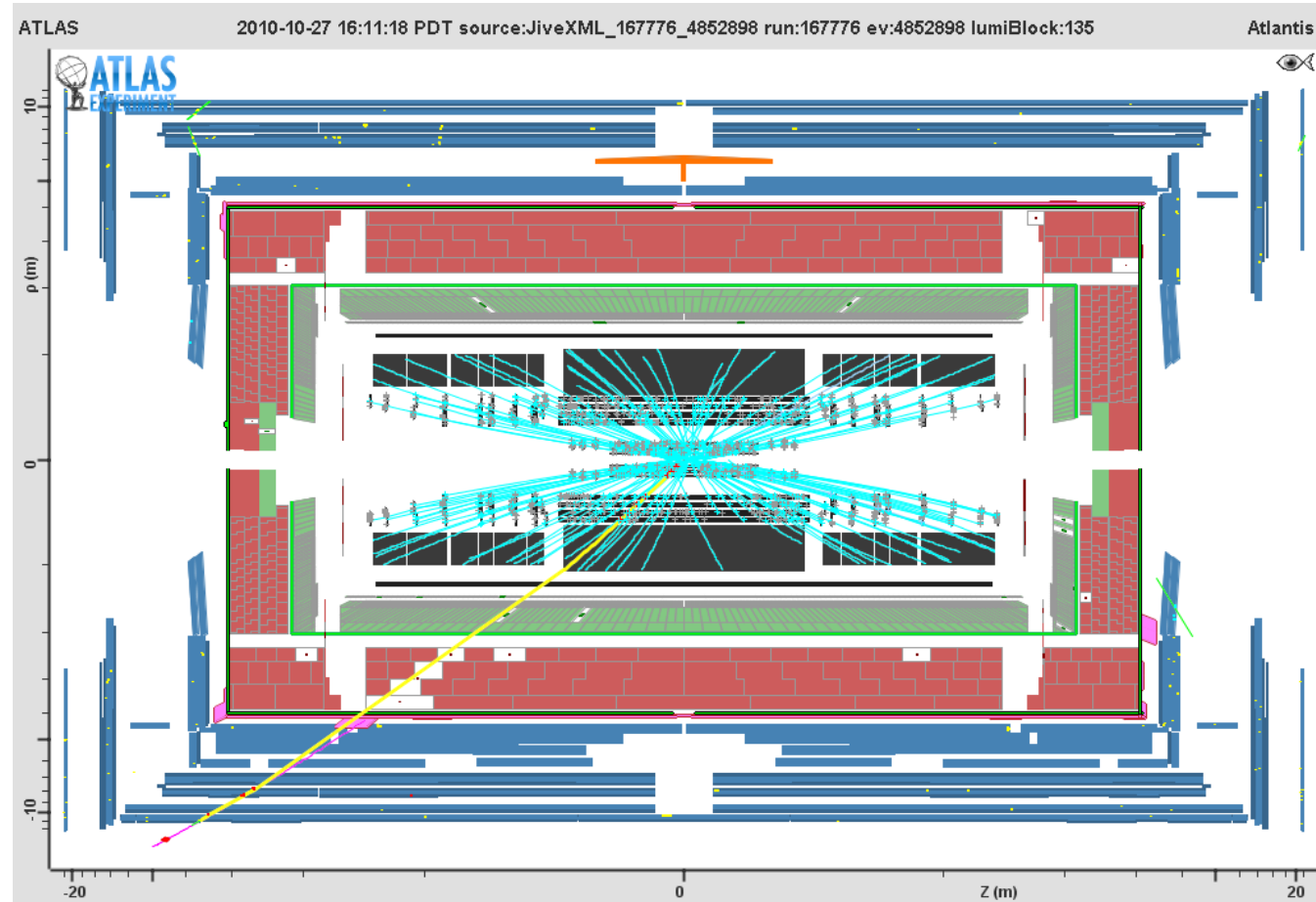


Overlay Status

Andy Haas
SLAC

November 30, 2010

*Simulation Meeting
ATLAS S&C Week*



100 GeV pT simulated muon + zerobias event from run 167776

Introduction

In addition to hard pp interaction:

- Pileup from other pp collisions in current and surrounding BCs
- Cosmics, Beam-gas, Beam-halo, Cavern bkgd., Detector noise, ...

Option 1:
“Pileup MC” (current default)

**Simulate all processes in MC
and mix together in proper
ratios with realistic timing**

Option 2:
“Overlay MC”

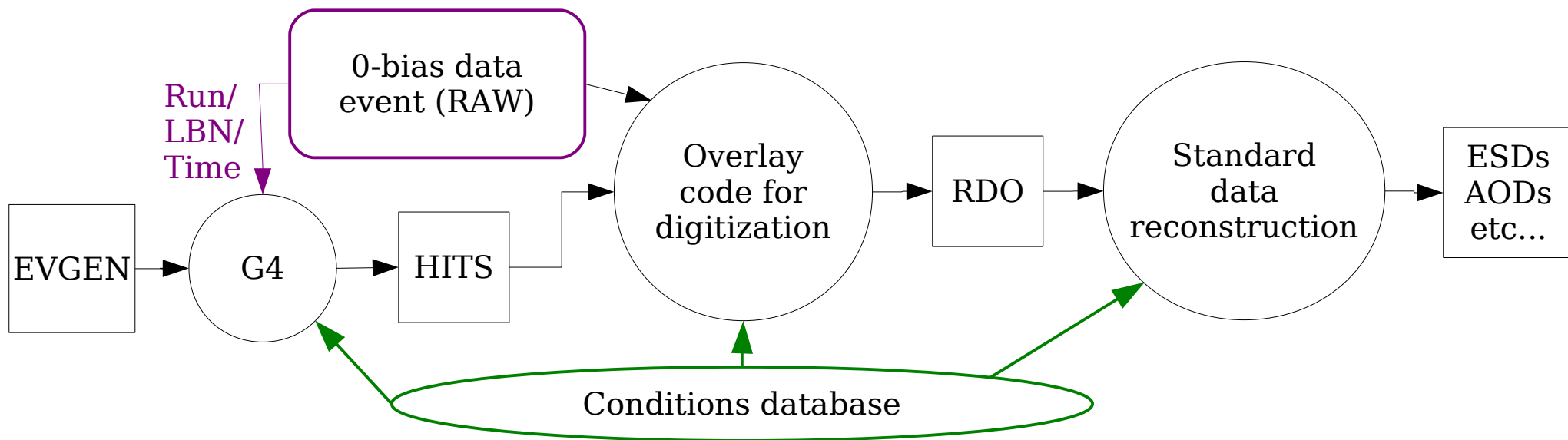
**Simulate hard pp interaction in MC
and overlay a “random” data
event to include all backgrounds**

Overlay method used successfully by BaBar, D0, ...

Important alternative, particularly for physics analyses where these other backgrounds are critical to model accurately

Overlay MC Steps

1. Define data period to simulate (e.g. good 2010 7 TeV runs)
2. Select "random" subset of zerobias background events
3. Simulate signal events (G4), use conditions for each event above
4. Overlay zerobias event during digitization
5. Reconstruct the combined event as data



Selecting 0-bias events

- For each MC sample, need to select 50k "random" events, passing L1_ZB, from specified run range
 - *#events from each lumi block should be proportional to its integrated luminosity*
- 1) Use RunQuery and LumiCalc to look up:
 - fraction of total integrated luminosity from each lumi block
 - number of zerobias events recorded in each lumi block
- 2) Calculate the number of events to select from each lumi block
- 3) ByteStreamMultipleOutputStreamCopyTool does the event selection and copying while running over RAW zerobias files
 - *For efficiency, create many (50) "zerobias sets" simultaneously*
 - *Use FileStager to process all RAW zerobias files of a run*
 - *Works on grid dataset names*
- Takes about 3 hours/job and one job for each run
- Storage: $50\text{k events} * 50 \text{ streams} * 3\text{MB/event} = \sim 7.5 \text{ TB}$

G4 with data conditions

Important to use same conditions for G4 signal event as existed for background event to be overlaid

- *beamspot* (SimuJobTransforms/VertexFromCondDB.py)
- *magnetic field* (SimFlags.MagneticField="COOL")
- *detector alignments*
 - Leads to G4 volume overlaps in various sub-detectors! (worst in Pixel, SCT) - can cause signal energy to be lost, material traversed to be underestimated, etc.
Most are ~microns, and code runs OK. But they are being fixed!

Currently set run/timestamp by hand at start of each MC job

- *What if beamspot or magnetic field changes mid-run?*
 - *Would like to set IOV for each event*
- *OK to have one MC job for each run? What if only 1 event in run?*

G4 with data conditions

- **Override run number:**

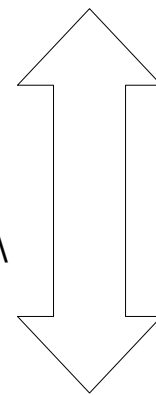
```
ServiceMgr.EventSelector.RunNumber = $run
ServiceMgr.EventSelector.FirstLB = 1
ServiceMgr.EventSelector.OverrideRunNumber = True
ServiceMgr.EventSelector.OverrideRunNumberFromInput=True
if runbase==167776:
    ServiceMgr.EventSelector.InitialTimeStamp = 1288231049
```

```
AtlasG4_trf.py runNumber=$run
```

- **Check for run number override *first* in SimAtlasKernel.py**
 - Needed for finding /Simulation/Parameters database

```
#ACH - check for manual runnumber first
if hasattr(SimFlags, 'RunNumber') and SimFlags.RunNumber.statusOn:
    minrunnum = SimFlags.RunNumber()
    maxrunnum = minrunnum + 1

elif hasattr(SimFlags, 'KinematicsMode') and \
    SimFlags.KinematicsMode.statusOn and \
    SimFlags.KinematicsMode.get_Value() == 'ReadGeneratedEvents' and \
    jobproperties.AthenaCommonFlags.PoolEvgenInput.statusOn:
    import PyUtils.AthFile as af
    f = af.fopen(jobproperties.AthenaCommonFlags.PoolEvgenInput()[0])
    if len(f.run_numbers) > 0:
        minrunnum = f.run_numbers[0]
        maxrunnum = minrunnum + 1
```



Overlay Code

- Validation of overlay code, using MC background, is ongoing
 - Compare input signal and background RDOs with overlay output
- Working on "MC+MC" note

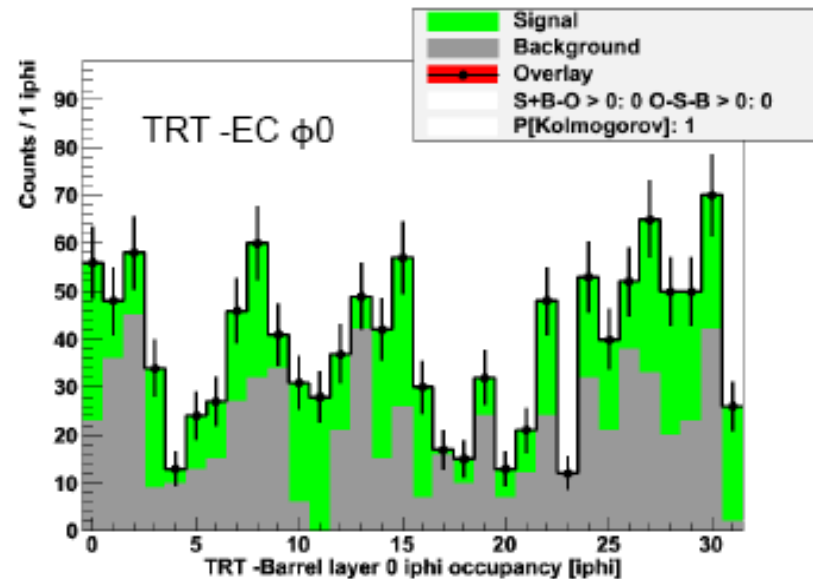
Pixel
SCT
TRT



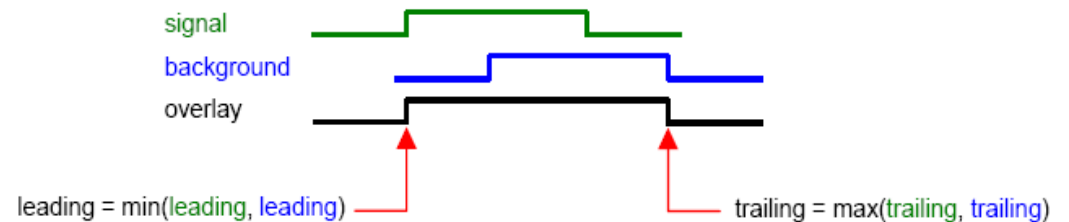
Lar
Tile
L1Cal



Muon ~ in progress
(challenge to read temporary
RDO in background...)

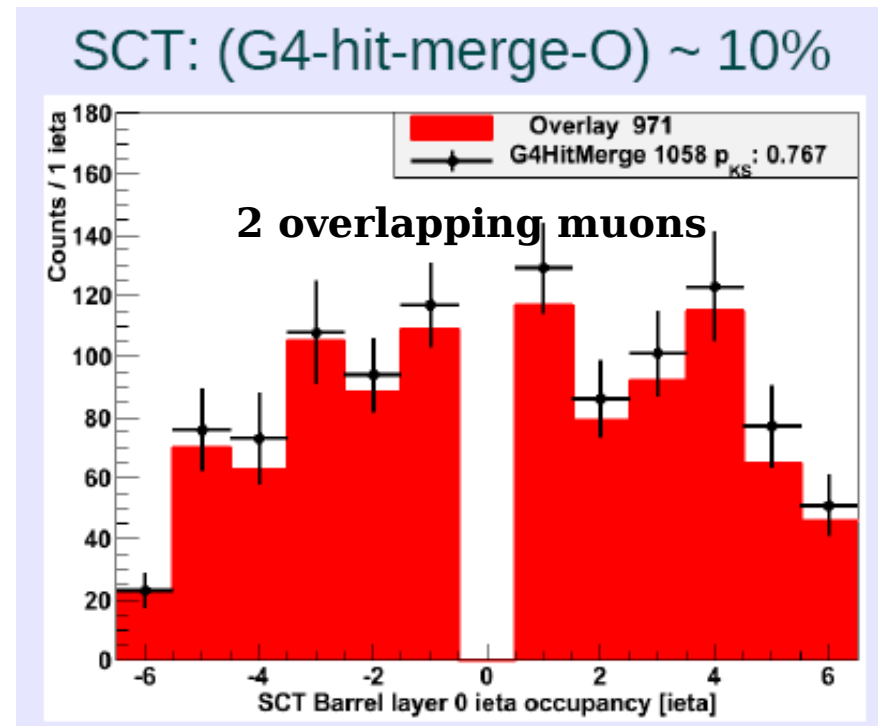
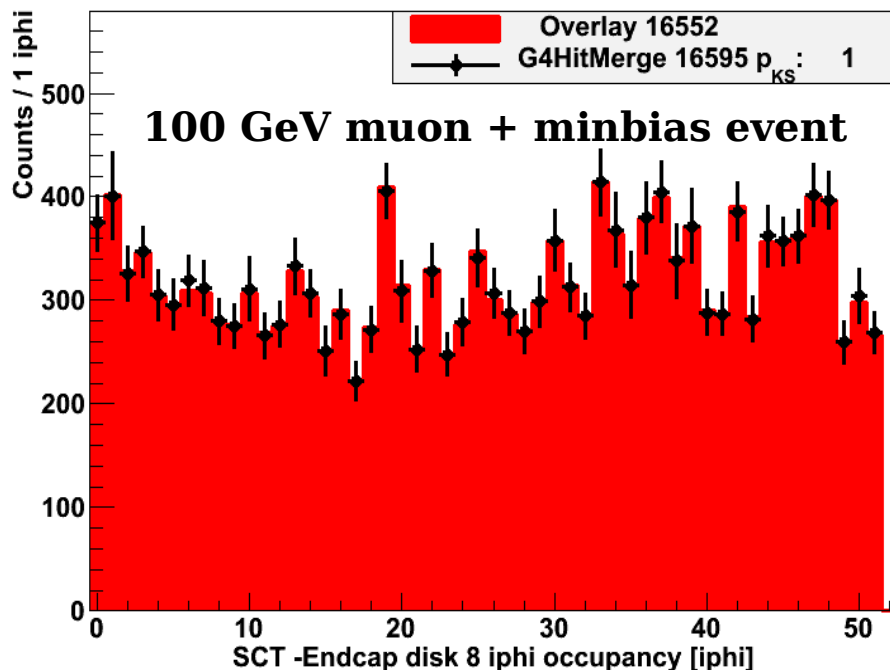


For TRT channels containing a signal and background RDO, the overlay RDO Time Over Threshold (TOT) is constructed as



G4HIT+G4HIT vs. Overlay

- Recently started comparison of *G4HIT+G4HIT vs. Overlay*
 - Wrote C++ code / Transform to add two G4HIT files
- **The two are in good agreement overall**
- Some interesting effects when two tracks overlap...

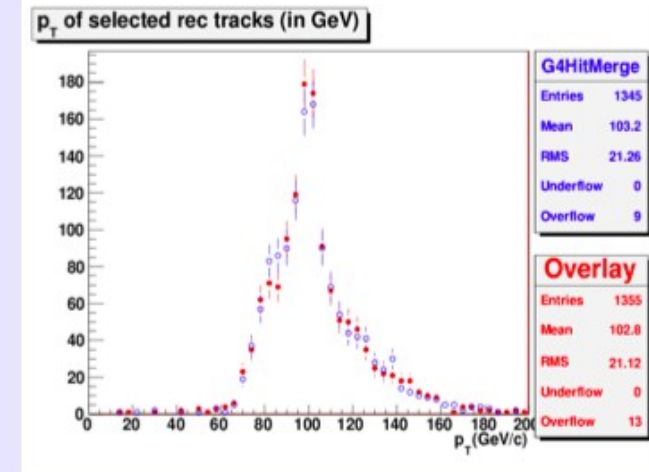
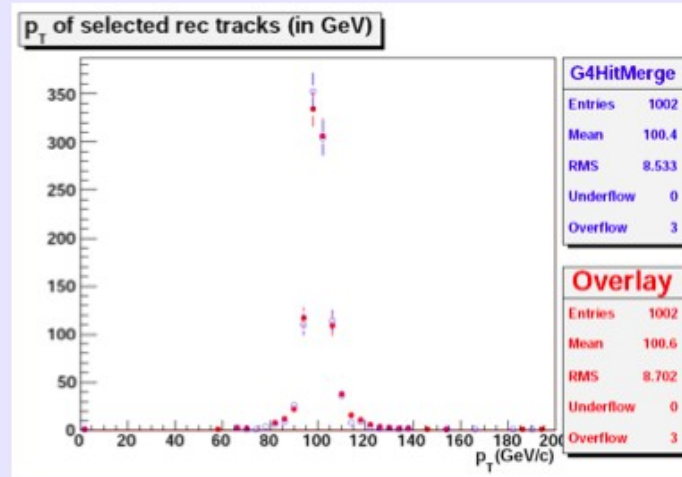
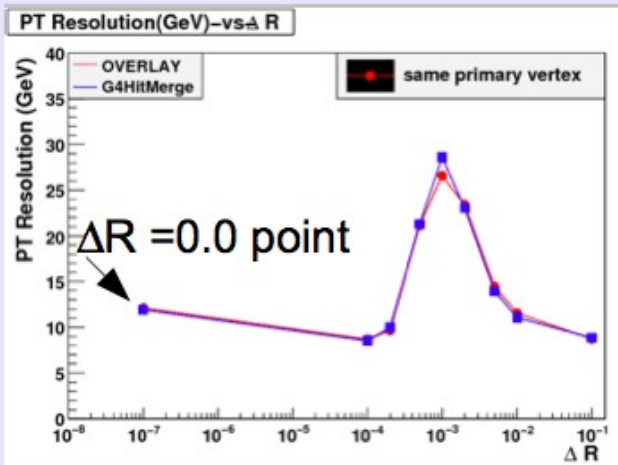


G4HIT+G4HIT vs. Overlay

- Two high-pt muons, separated by a small amount

$\Delta R=0.0001$

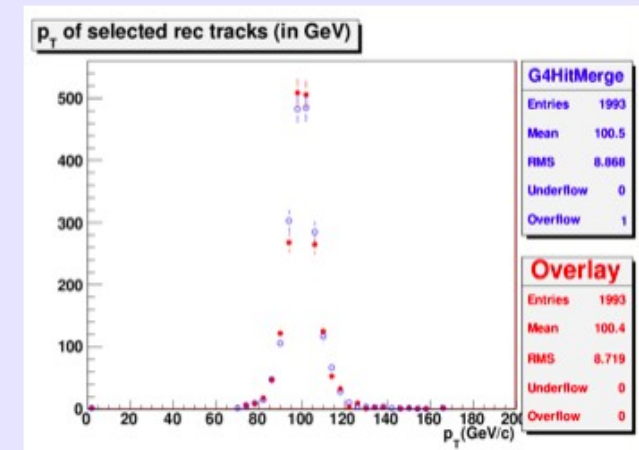
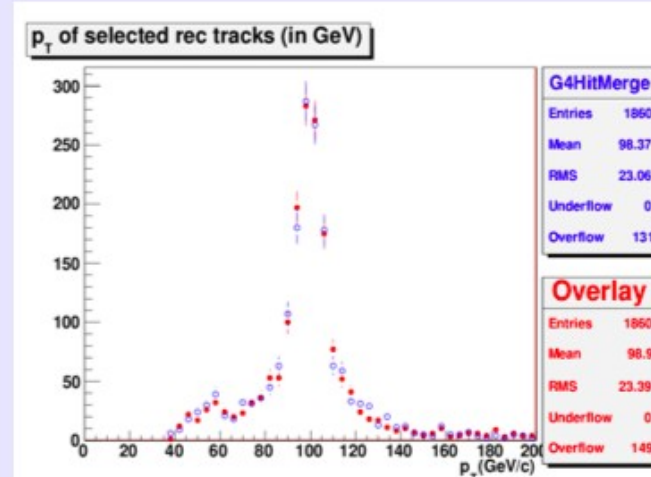
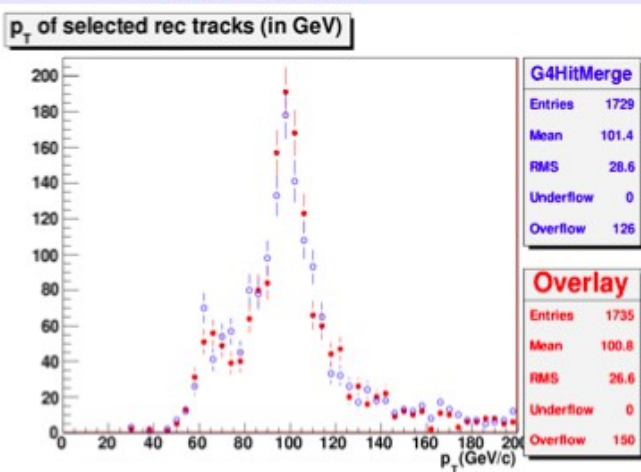
$\Delta R=0.0005$



$\Delta R=0.001$

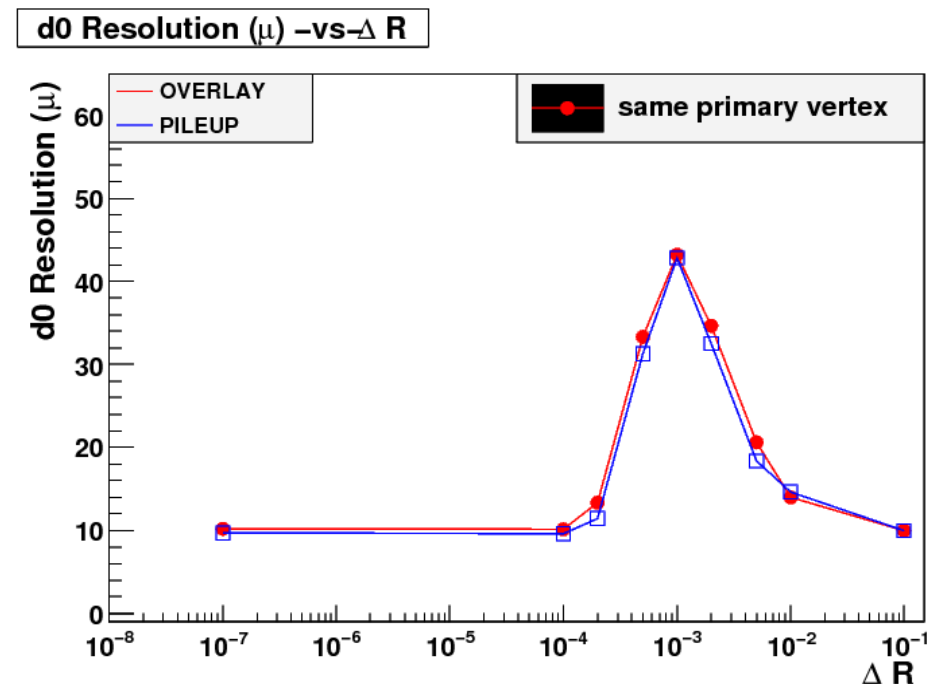
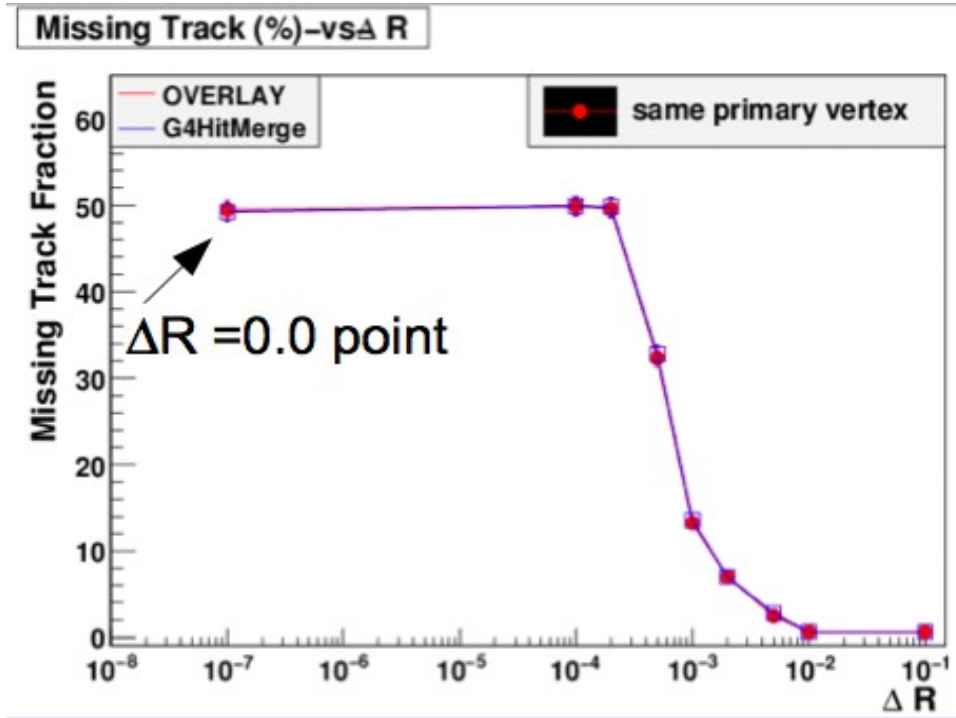
$\Delta R=0.002$

$\Delta R=0.01$



G4HIT+G4HIT vs. Overlay

- Two high-pt muons, separated by a small amount
- PT resolution and efficiency are well modeled for overlapping tracks in the silicon and TRT

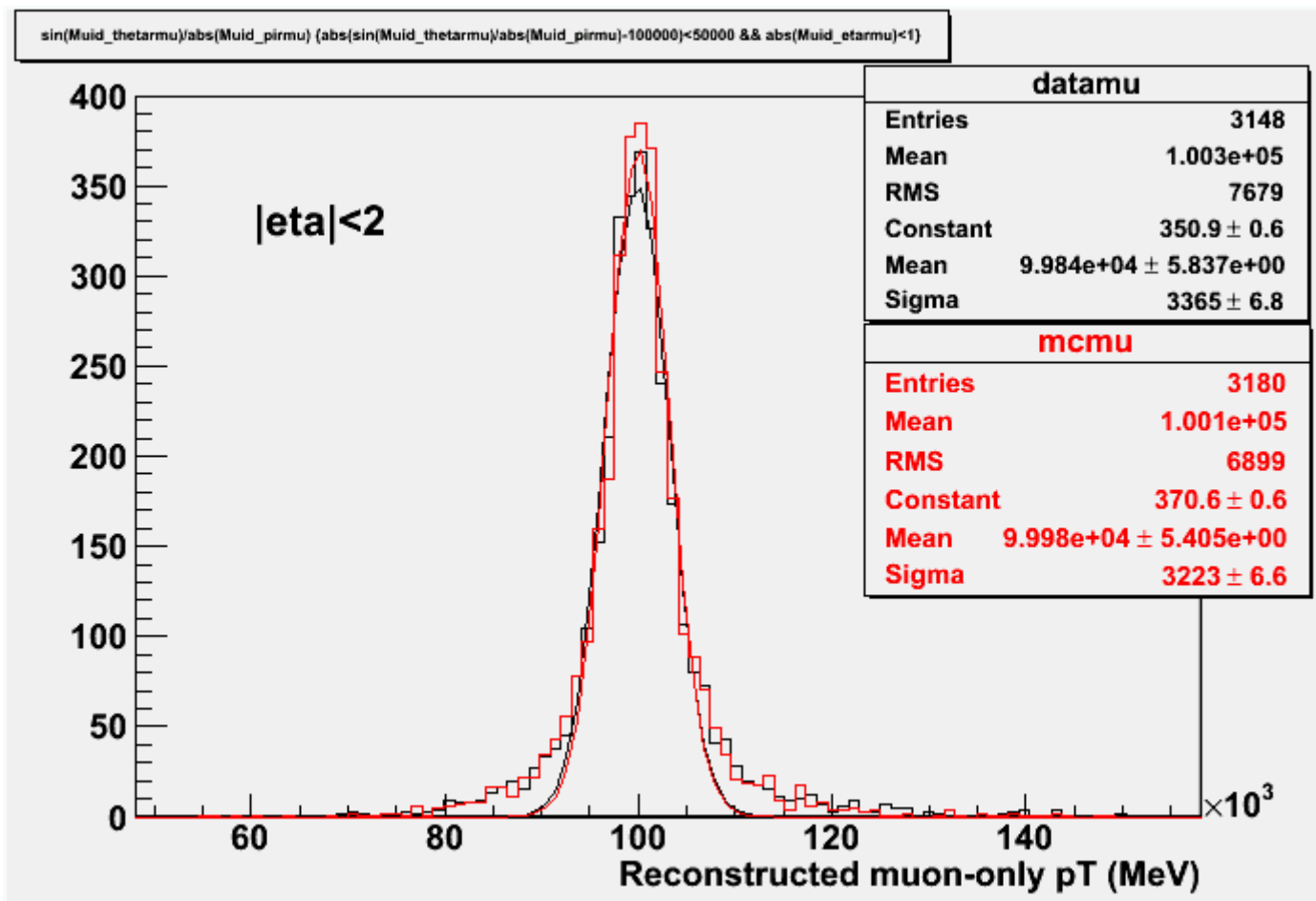


Overlay with Data Conditions

- Validation of MC+data overlay at ESD level
 - Check that signal MC objects with data conditions are reconstructed as well as they are with MC conditions
 - ID, LAr, TILE all look OK
 - Muons now mostly OK too (see following plots)
 - Check that background overlay events are reconstructed as well as in standard reconstruction
 - Work started, first looks show no major problems
 - Needs to be completed in detail

Muon-system only tracks

- 5000 single-muons, $pT=100$ GeV, on zerobias from run 167776
- **Standard MC vs. Data Overlay MC - release 16.0.2.3**
- **Resolution for $|\eta|<2$ now equivalent !!!**



Alignments were looking at *timestamp* in IOV, not run #

So before I was using old muon alignments at the G4 stage

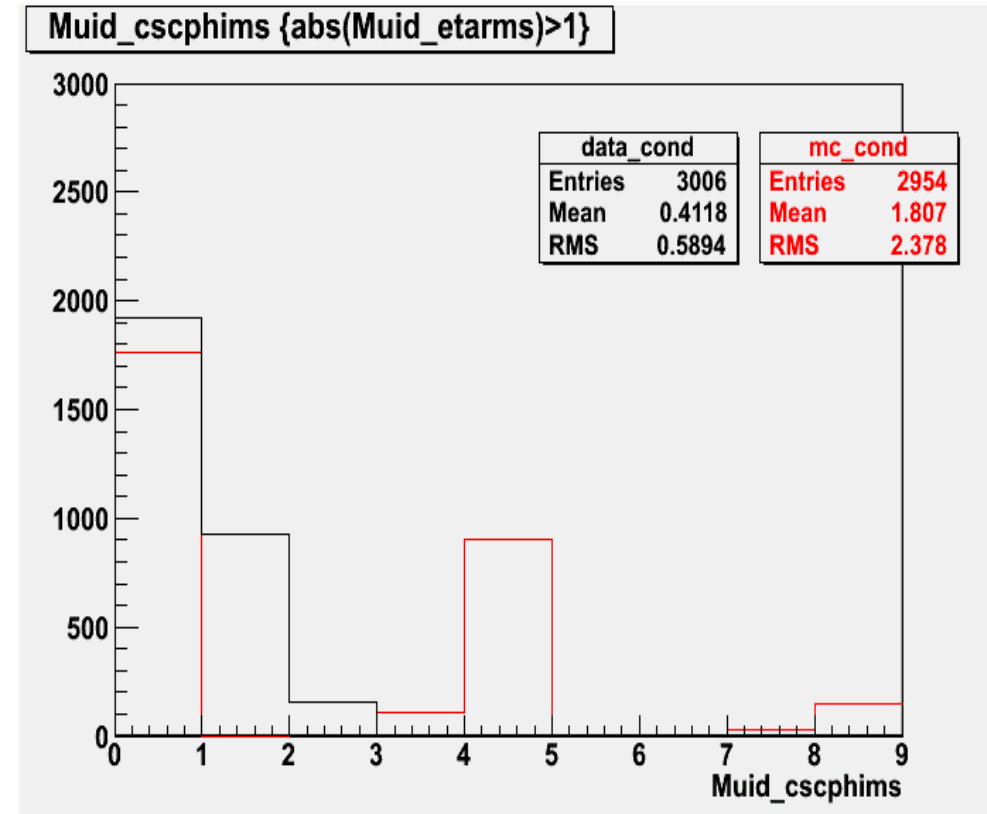
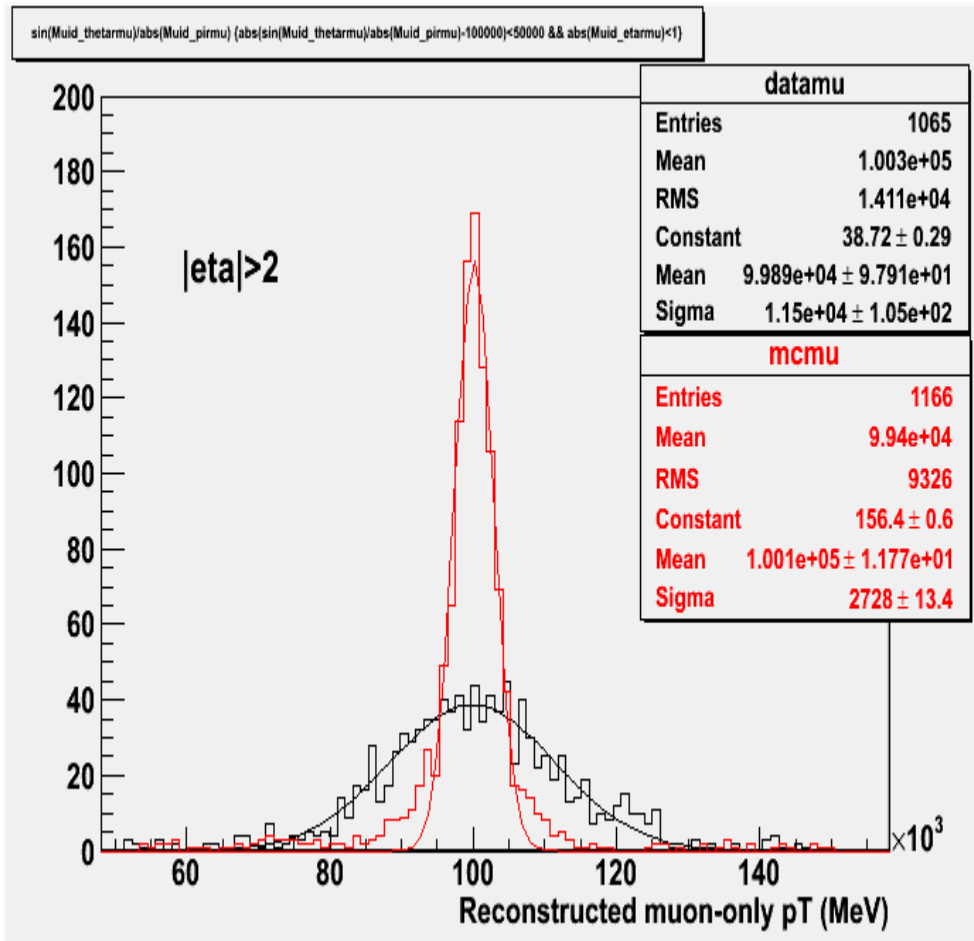
Muon-system only tracks

- Resolution poorer in the forward regions

MuonCalib::CscCoolStrSvc

WARNING Invalid chamber layer requested when retrieving noise.

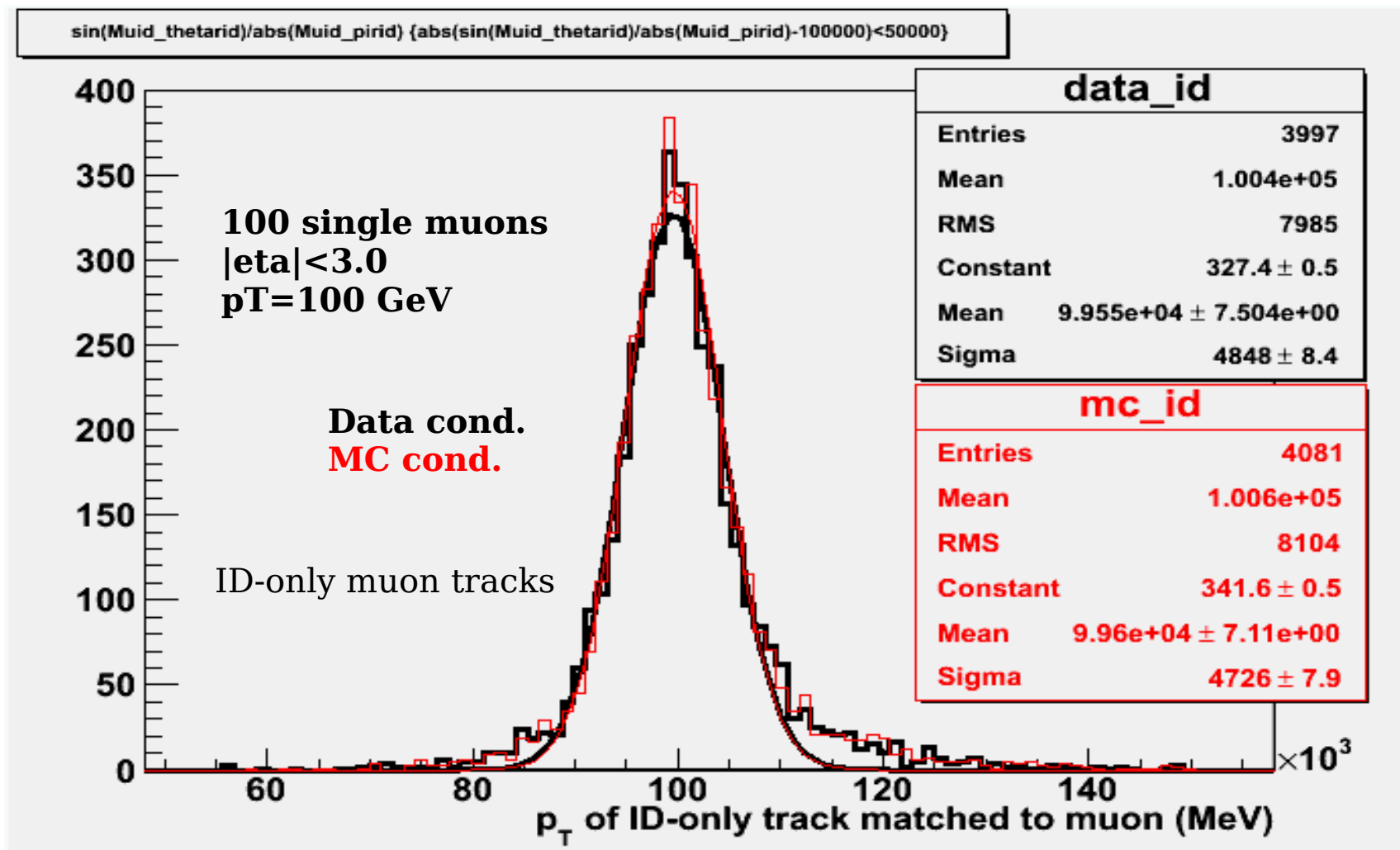
Tried corresponding channel in valid chamber layer and found good data.



Less CSC hits in data conditions ???

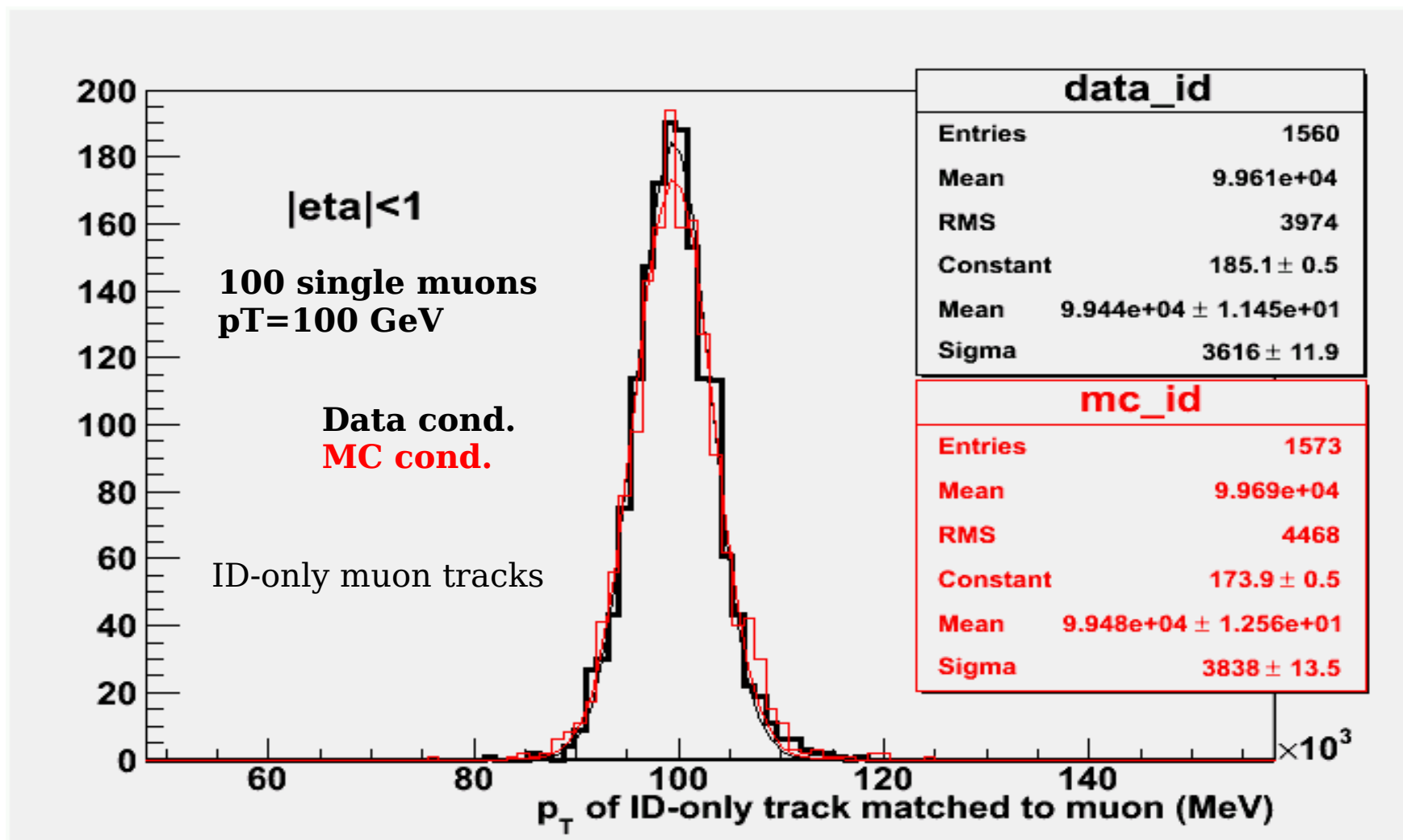
ID-only Tracks

- ID tracks still have equivalent resolution, all eta's



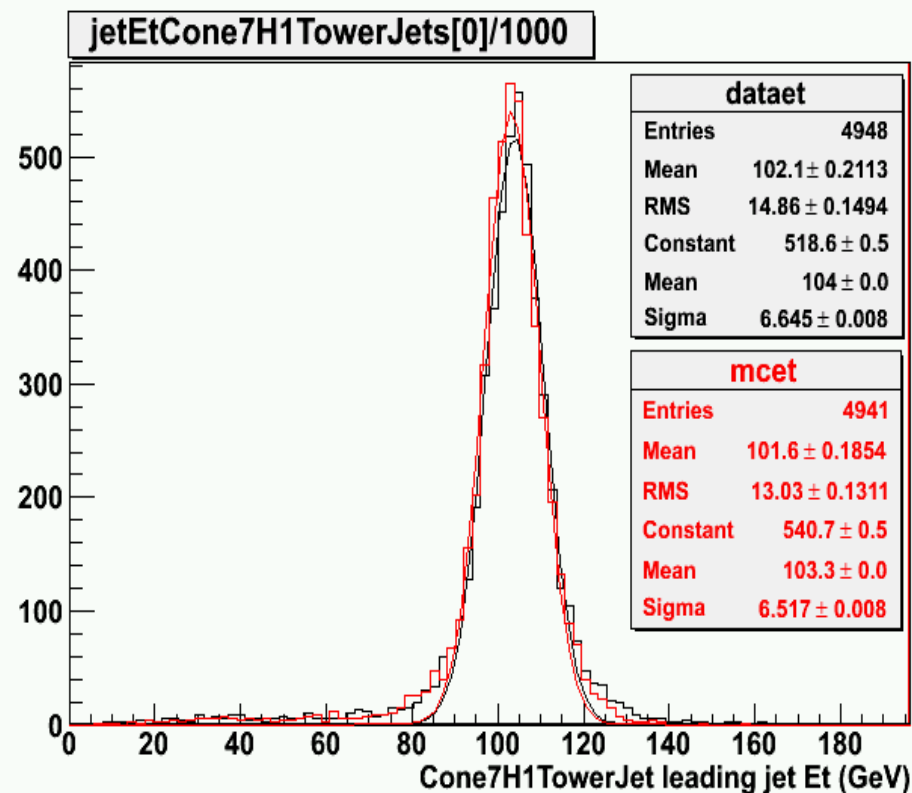
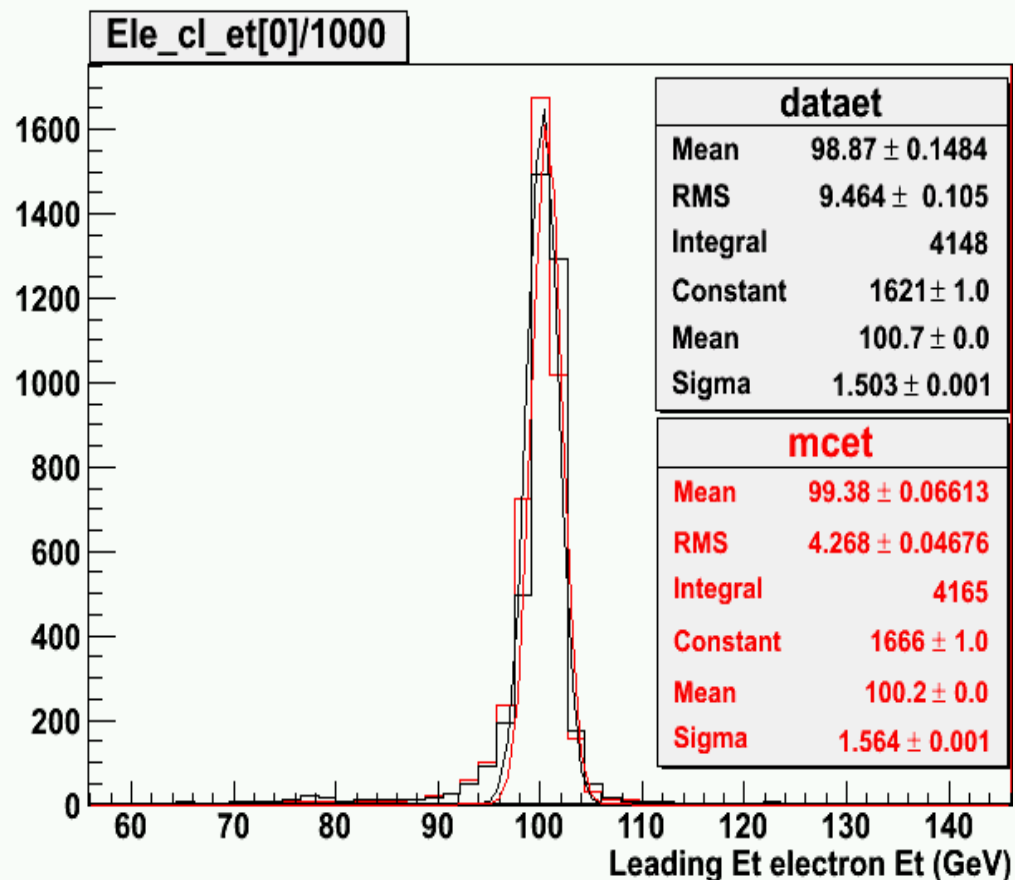
ID-only Tracks

- ID tracks still have equivalent resolution, all eta's



Calorimeter

- 100 GeV Et single electrons and single pions
 - Standard MC vs. Data Overlay MC
- Good resolution and response using data overlay!



Next Steps

Ready to try out overlay simulation in production environment

- Full scale zerobias event selection for all (good lbn of) 2010 runs
- Automated generation of G4 with proper runs/timestamps
- Overlay of zerobias on MC farms

Will force us to solve technical hurdles like access to data conditions, shipping of zerobias events, etc.

Plenty more validation / testing to do in standalone and physics / performance groups

- MC+MC validation, forward muon issues, *background resolutions*
- What samples should we make for groups to study?

Backup

Recording 0-bias events

- Use "0-bias" events for overlay:
random, but proportional to instantaneous luminosity
- *Trigger one turn after a "key" trigger fires (then prescale)*
 - Key trigger fires proportional to luminosity *in each BCID*
 - Capture bunch structure of background
 - Must have rate \ll BC rate, but $> \sim 1$ Hz
- **L1_ZB, bit 240**
 - Keyed on L1_EM10
 - *Unsuppressed detector readout*
 - Written to **physics_Zerobias** stream (along with flat rate Obias trigger, L1_RD1_FILLED)
- **0-bias rate aimed for ~ 1 Hz**
 - Prescale usually adjusted after lumi scan at start of fill

Selecting 0-bias events

- **AtIRunQuery** finds LBNs with stable beam and 3.5 TeV beams in run period of interest (SAP)

```
AtIRunQuery.py --run "154817" --olclumi "1ub+" --lhc "beamenergy 3400+" --lhc "stablebeams TRUE" --show run --show events --show time --show "lhc" --show olclumi --show olclbdata --projecttag "data08*,data09*,data10*" --partition "ATLAS" --verbose > runquery.log
```

- Read luminosity info from COOL, for each LBN from RunQuery

```
iLumiCalc.exe -t L1_ZB -V --online -x data/MyLBCollection.xml >lbn
```

- **ROOT** script then analyzes output, builds a big map of all the lumi in each RUN/LBN, shoots random numbers, and chooses a given number of events from each LBN, write to text file:

```
stream 1, run 154817, lbn 275, has 91 events and 1.620920/ub, 0.003324 intlumi of run, 19 wanted  
stream 1, run 154817, lbn 276, has 88 events and 1.605350/ub, 0.003292 intlumi of run, 12 wanted  
stream 1, run 154817, lbn 277, has 90 events and 1.601349/ub, 0.003284 intlumi of run, 23 wanted
```

- Athena job runs over Zerobias stream for each run, selects L1_ZB events, reads map file above, and picks given number of random events from each LBN to write out