

Updated

Update of Centrality determination (Run4 Au+Au 62 GeV)

4/11/2006

M. Konno

(Univ. of Tsukuba)

Introduction

- Peripheral issues in spectra measurements.
One of the reasons is the centrality determination.

(Ref: Analysis meeting on 1/18/06)

(Ref: G/H PWG meeting on 2/13/06)

- Centrality distribution is not flat completely when looking it in each z-vertex bin.

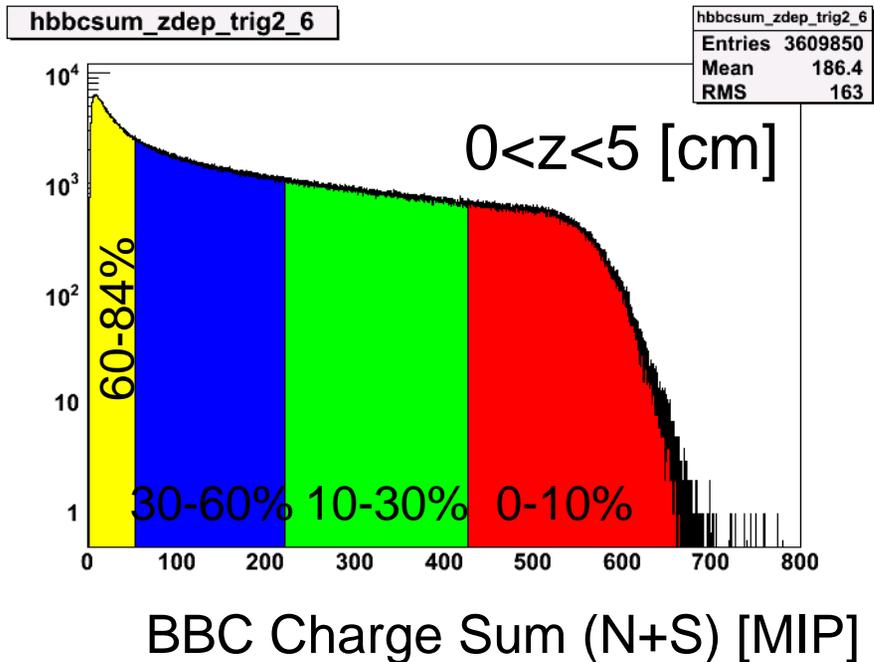
- npc1 correction to be removed.

(Ref: Jamie's slides on 1/18/06)

=> **Centrality function** (Run4 Au+Au 62 GeV) to be updated.

(offline/framework/recal/PercentileRecalReco.C)

How to define the centrality

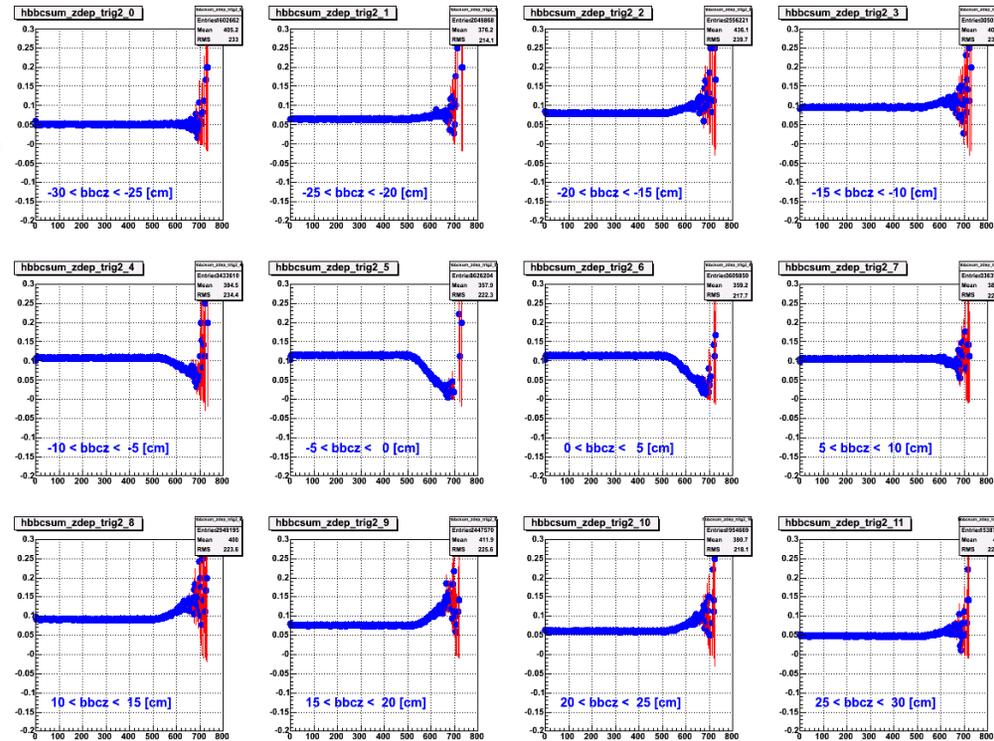
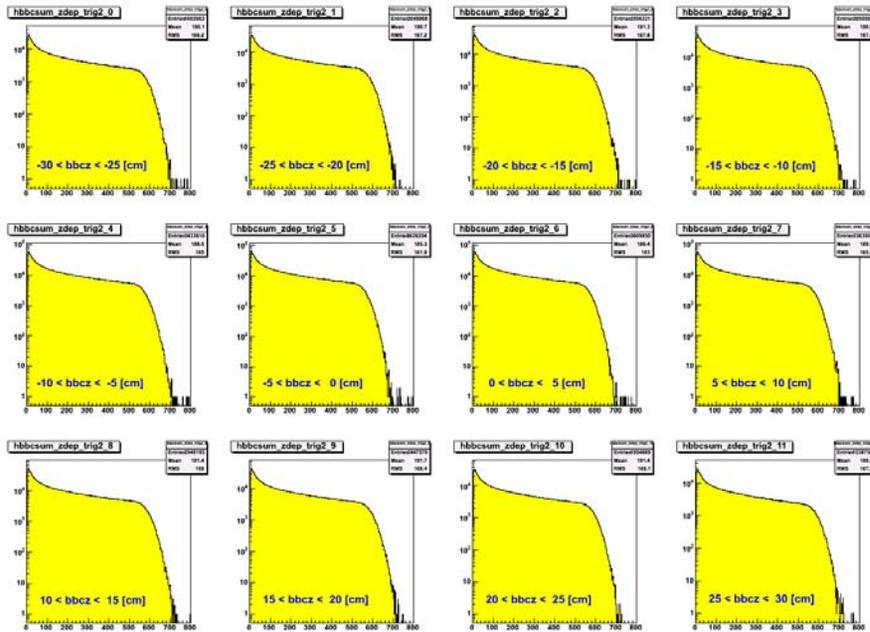


- Using BBC only
- Trigger condition: $\text{bbc}n > 1 \ \&\& \ \text{bbc}s > 1$ (to reject background events)
- Trigger efficiency: $83.7 \pm 2 \%$
- $|\text{BBC vertex}| < 40$ cm (offline cut) (w. the z-vertex step of 5 cm)

BBC Charge Sum (z-vertex sliced)

The shape of BBC charge sum distributions are different in z-vertex bin.
=> Divide the distributions in each z-vertex bin (5cm step).

Ratio to tot.

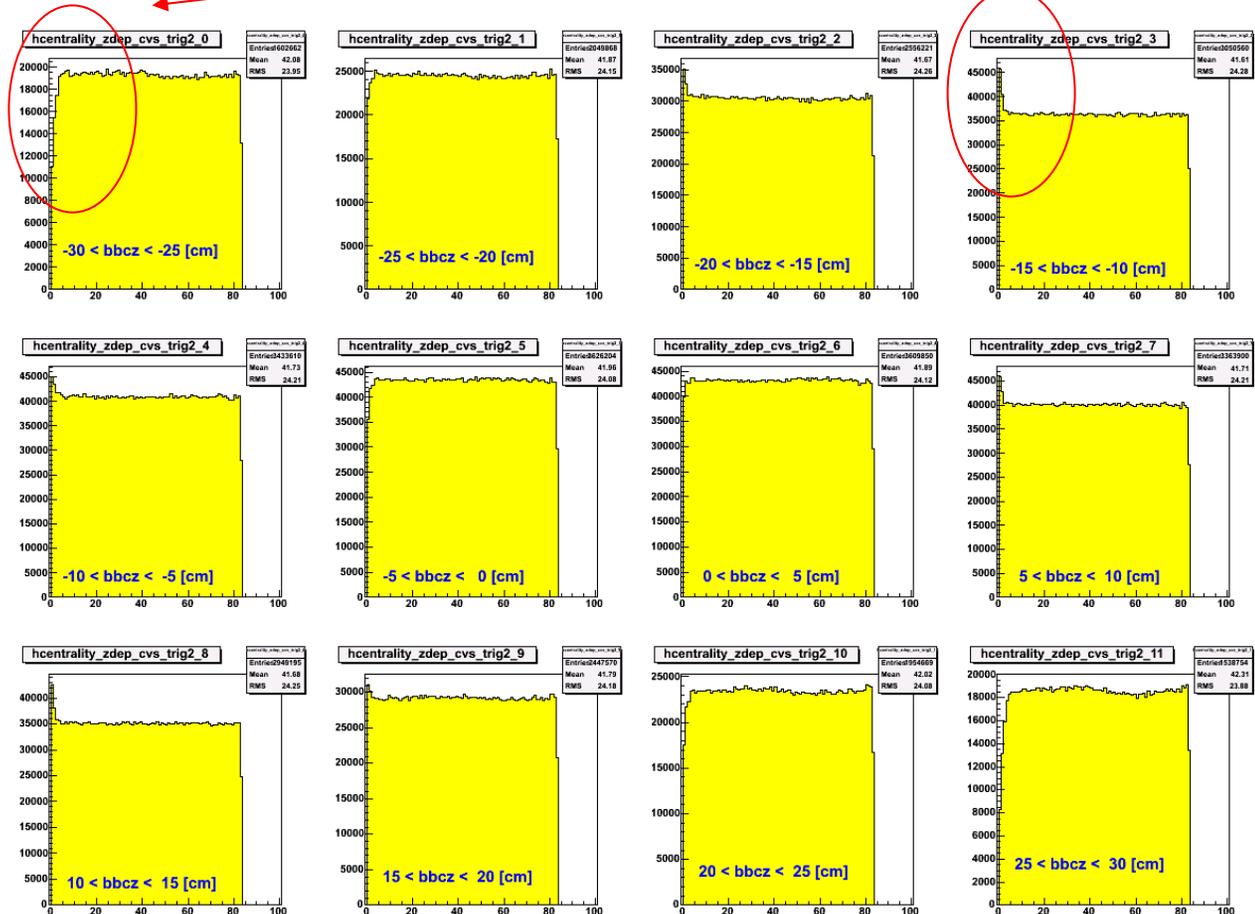


CVS Centrality

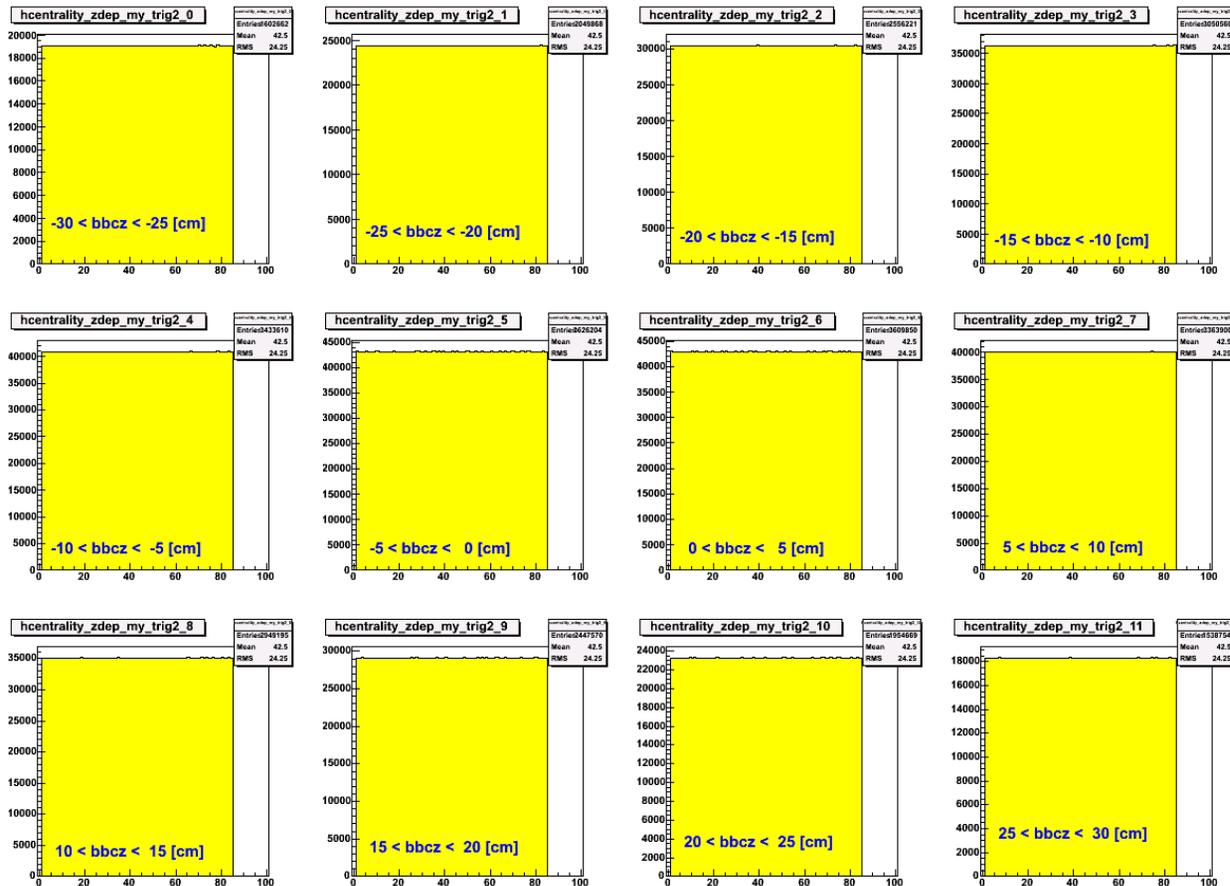
The current code applies a correction to the BBC charge sum by the function (PC1, z) below, for linearization of the BBC response. (AN290)

But, not complete flat

```
float K_30 = 0.00725082;  
float K = Kval(npcl); // *****  
float CN= K/K_30*exp(-K*z);  
float CS= K/K_30*exp(K*z);  
  
float cbbcqn = bbcqn*CN;  
float cbbcqs = bbcqs*CS;  
  
float A = cbbcqs+cqbbcqn; //
```

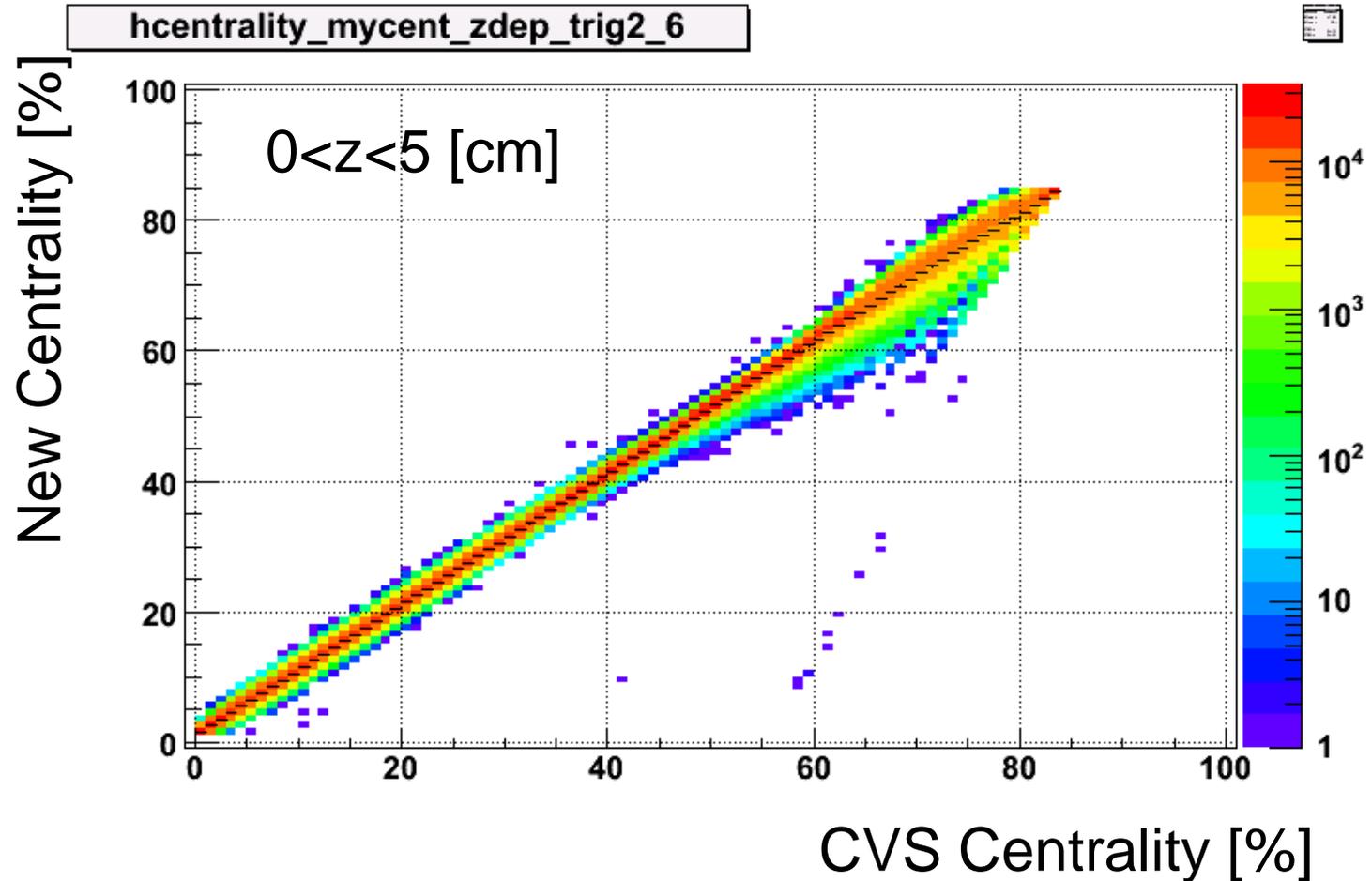


New Centrality



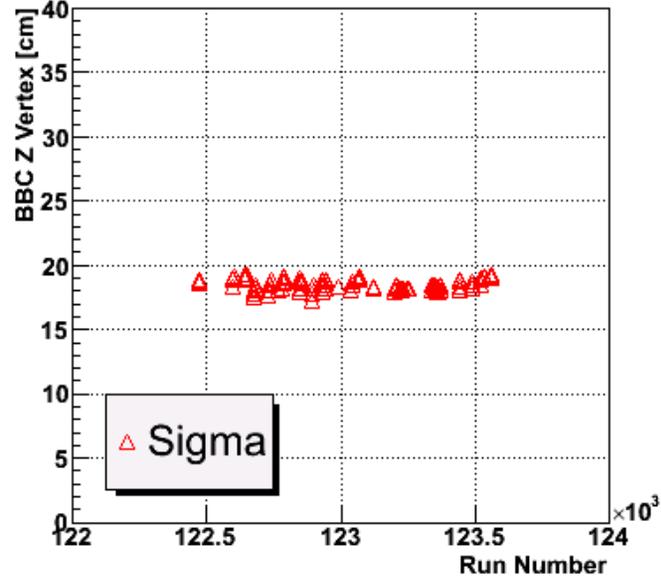
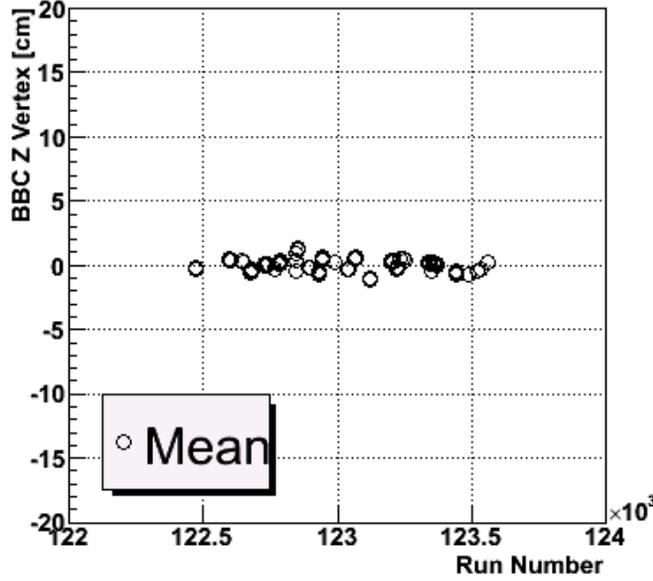
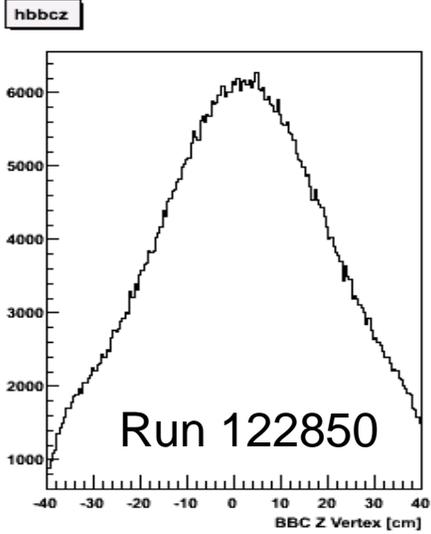
- Centrality function redefined **without the PC1 correction**.
- By definition, centrality distributions are flat.

2D Correlation



- Almost good agreement.
- Some difference in peripheral.

Run-by-run QA



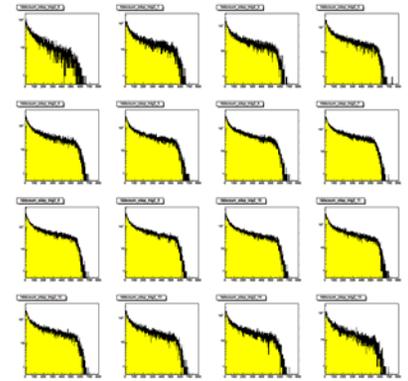
Example.

BBC z vertex distribution is stable.

Run-by-run QA

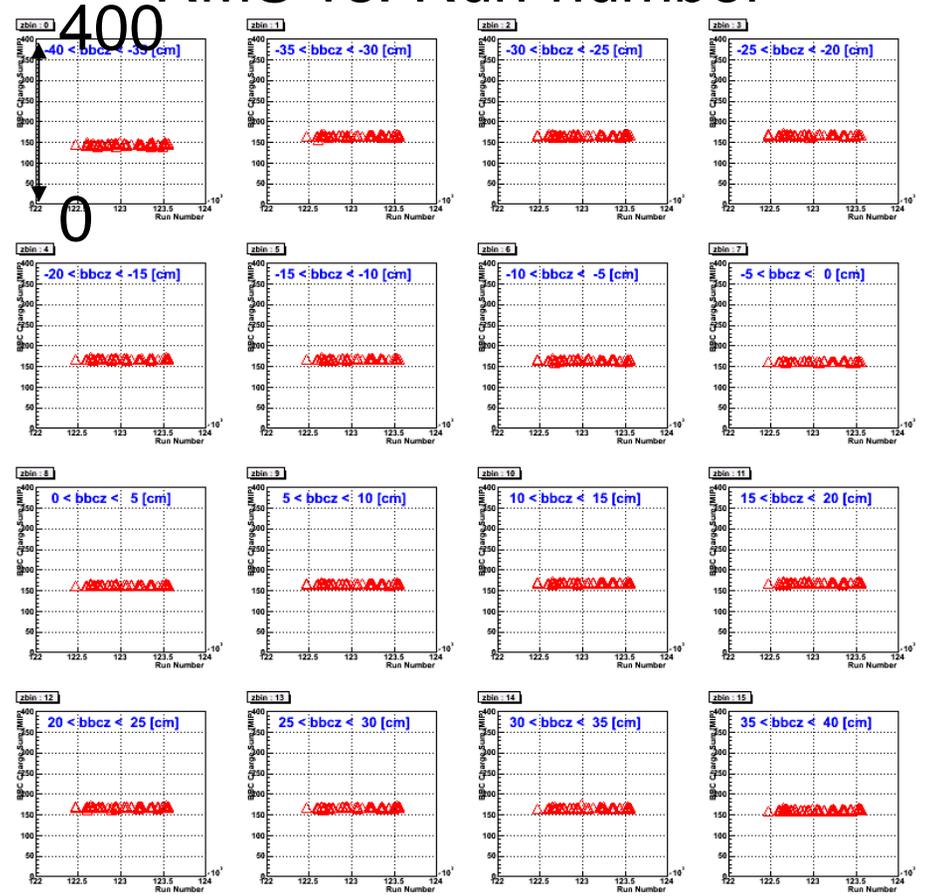
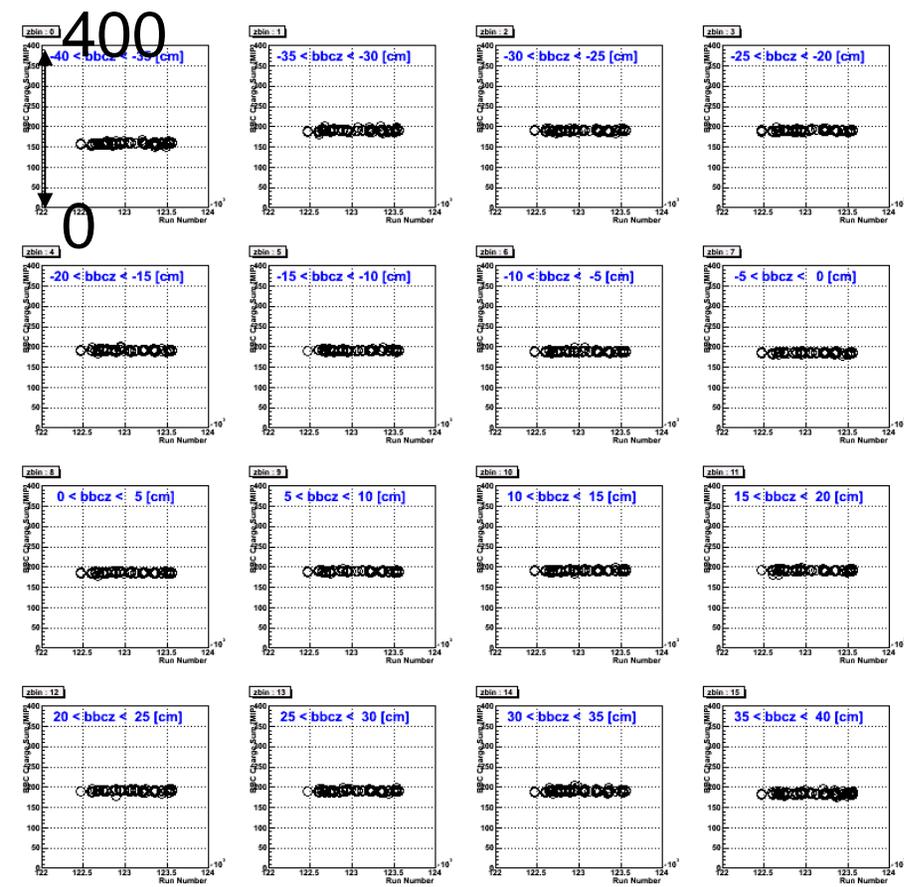
BBC charge sum (N+S) distribution is stable in terms of absolute value and shape.

Run 122850
(Example)



Mean vs. Run-number

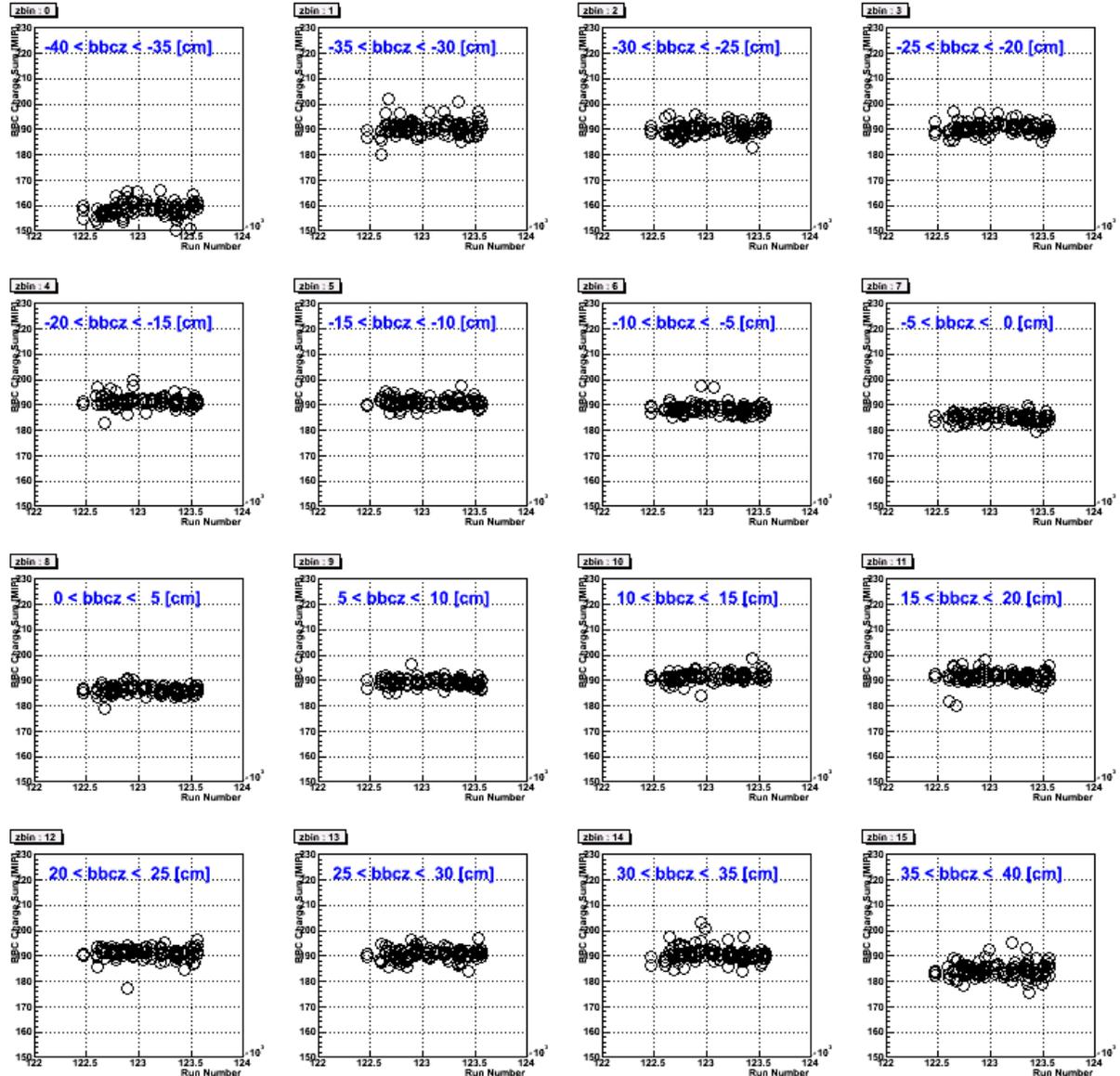
RMS vs. Run-number



Run-by-run QA

Mean vs. Run-number Closer look.

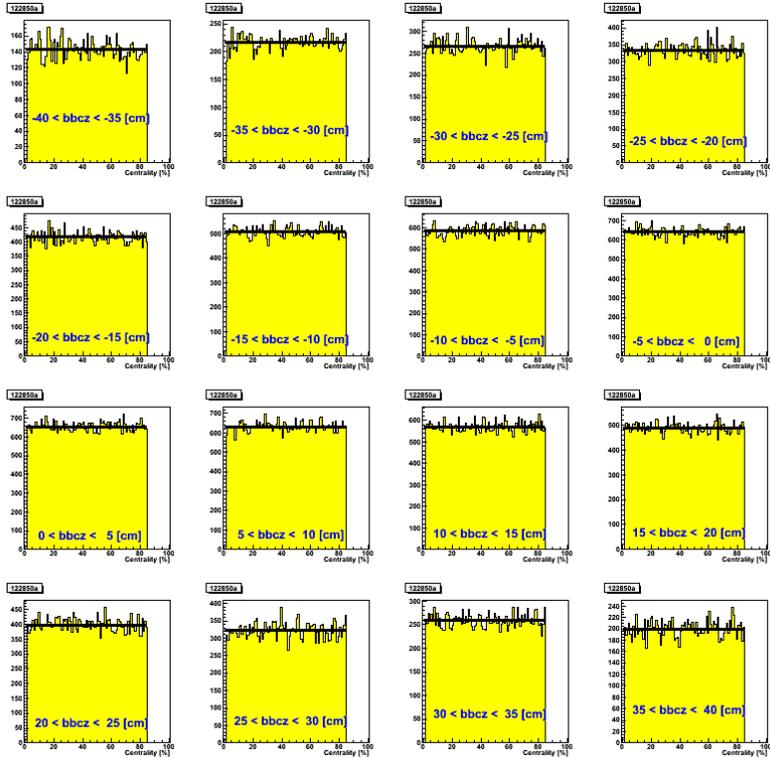
230
↑
150



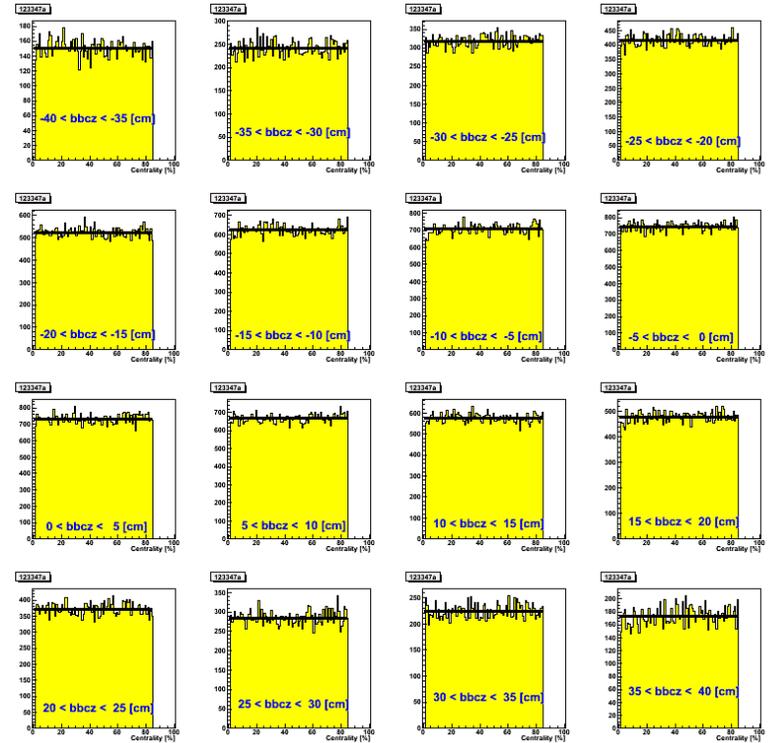
Slight increase : ~3 %
(as run-number)
=> Small correction.

Run-by-run QA

Run 122850



Run 123347



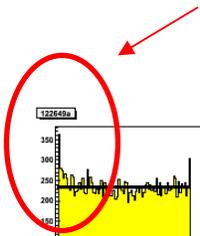
Flatness Good.

- See all the figures in:
http://www.phenix.bnl.gov/~konno/cent_check_run4auau62/

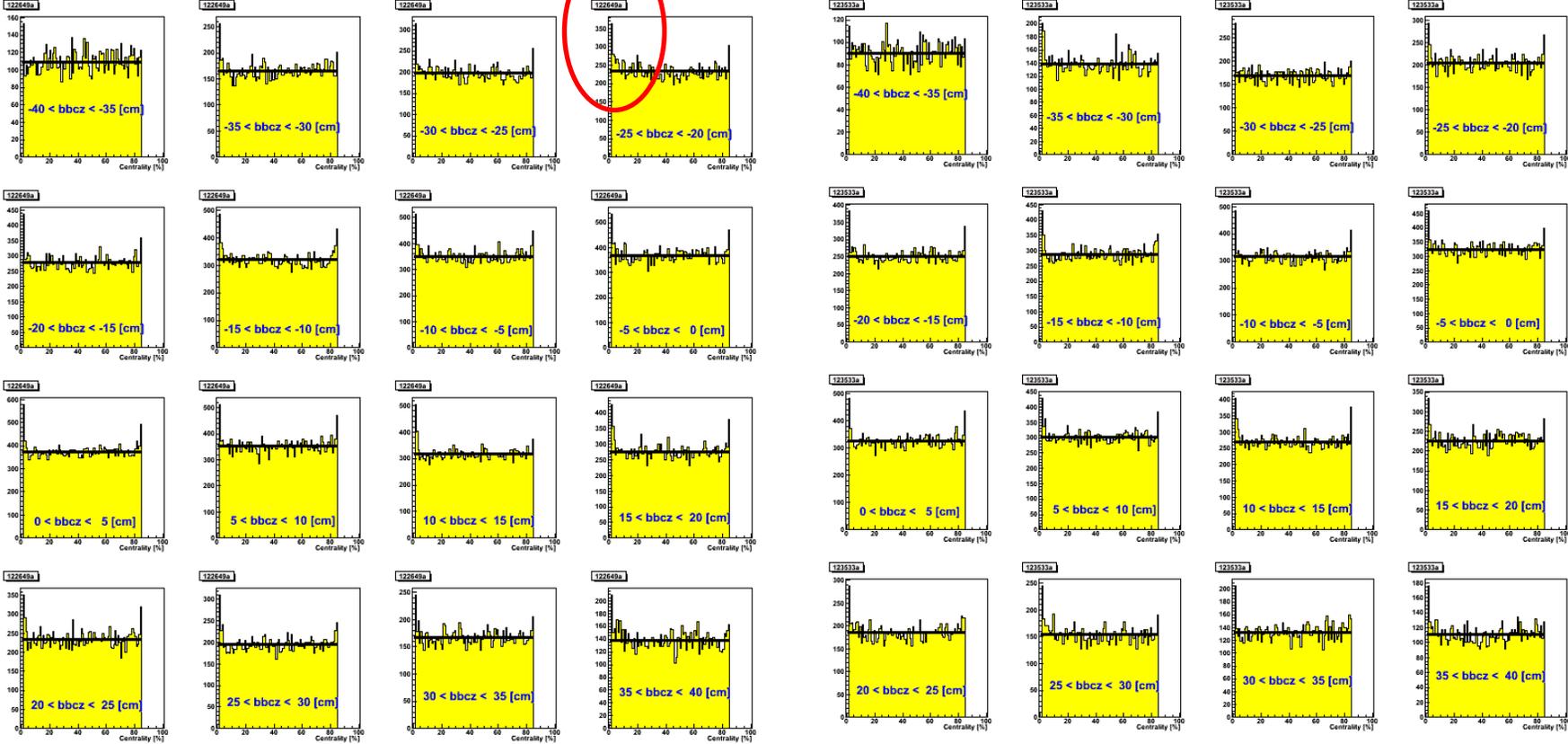
Run-by-run QA

For example, here.

Run 122649



Run 123533



Slightly Bad.
(because those runs are filed-off,
and excluded in making the function.)

Summary

- **Centrality function (Run4 Au+Au 62 GeV) updated.**
(Removed “npc1” from the centrality determination.
Only BBC charge sum is used. and re-define the centrality
in each z-vertex bin. This method is the same as in Cu+Cu.)
- **Recal. module** (`offline/framework/recal/PercentileRecalReco.C`)
should be updated accordingly.
- **Several QA checks**
(z-vertex, charge sum, centrality) done.