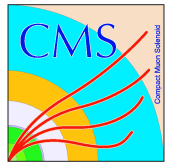
Effect of GlobalPositionRcd-removal Bug on Muon Alignment

Aysen Tatarinov
Vadim Khotilovich
Jim Pivarski
Alexei Safonov

Texas A&M University

8 June, 2010

Overview (not just this talk)



- We confirm the RPC-hit bias presented by Pablo
  - RPC hits were included in track refits because of a typo
  - correcting this fixes sawtooth and residuals vs. q/p_T biases
  - $p_T > 100$ GeV/c cut was used to limit the effect on alignment (to 1.5 mm, as it turns out); now this cut can be removed, making alignment with collisions possible
  - see http://indico.cern.ch/getFile.py/access?contribId=4&resId=0&materialId=slides&confId=88578
- Muon POG discovered a strange feature with the new alignment constants
  - low-p_T track seeds assigned a default value of 3.0 GeV/c, with a cut to remove them at 3.001 GeV/c
  - muon alignment spread this distribution above the cut
  - not an alignment problem, but discovered by muon alignment
- GlobalPositionRcd-removal bug: this talk

GlobalPositionRcd bug-fix

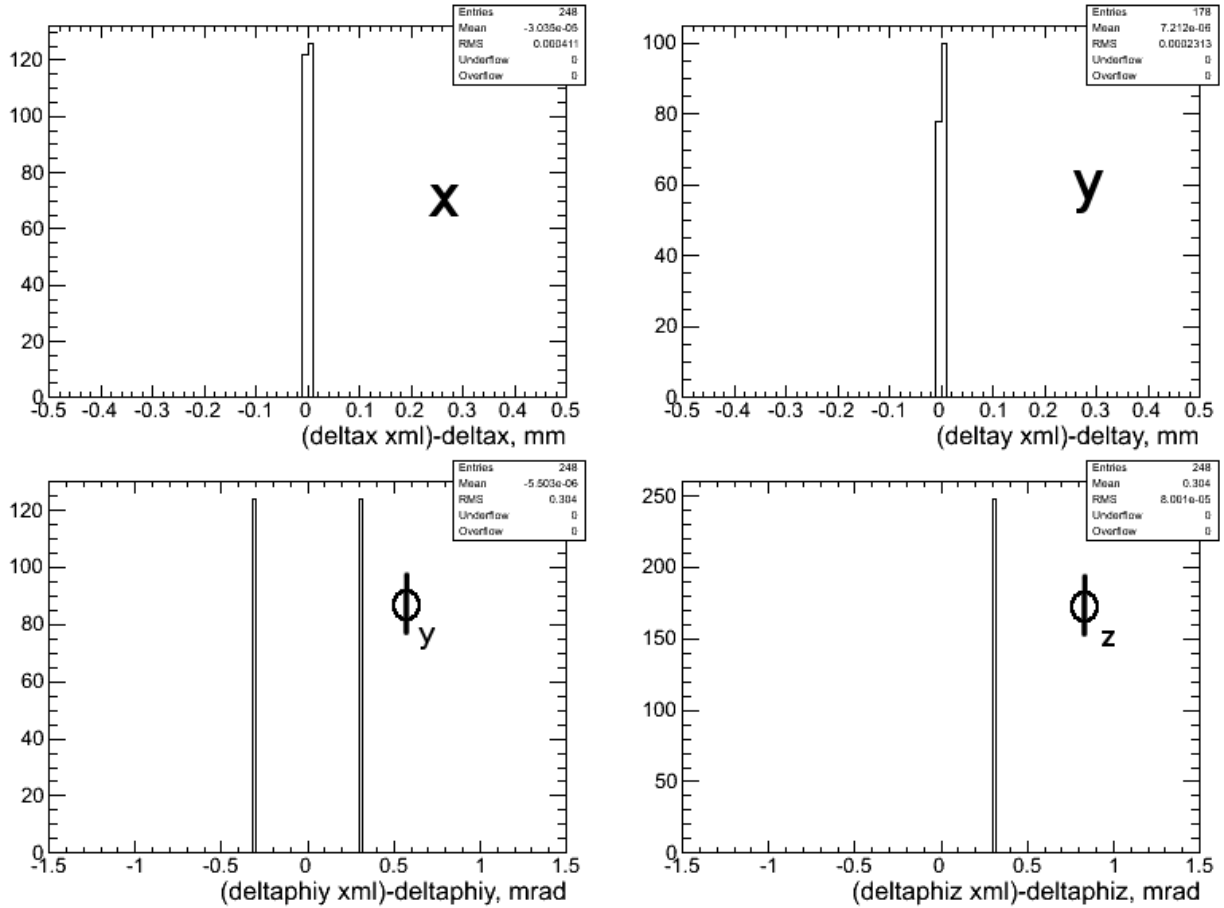
- Andreas discovered and corrected an error in the way GlobalPositionRcds are handled in alignment (not track-reco)
  - GlobalPositionRcd must be in geometry to calculate residuals
  - it must then be removed from the final results before outputing to TrackerAlignmentRcd/DTAlignmentRcd/CSCAlignmentRcd
  - rotations *(only)* were improperly removed

Diff of /CMSSW/Geometry /TrackingGeometryAligner/interface /GeometryAligner.h

#	revision 1.10, Fri May 14 13:43:20 2010 UTC Line 167	revision 1.11, Fri Jun 4 13:42:46 2010 UTC Line 167
167		
168	// Remove global position transformation from alignment	// Remove global position transformation from alignment
169	newPosition = inverseGlobalRotation * ((*iAlign).translation() - globalShift);	newPosition = inverseGlobalRotation * ((*iAlign).translation() - globalShift);
170	newRotation = globalRotation * (*iAlign).rotation();	newRotation = (*iAlign).rotation() * globalRotation;
171		
172	newAlignments->m_align.push_back(AlignTransform(newPosition,	newAlignments->m_align.push_back(AlignTransform(newPosition,
173	newRotation,	newRotation,

Effect on DTAlignmentRcd

- Plotting: (alignment fit output) − (change in DTAlignmentRcd) for each chamber; can only be non-zero if ∃ error in AlignmentProducer

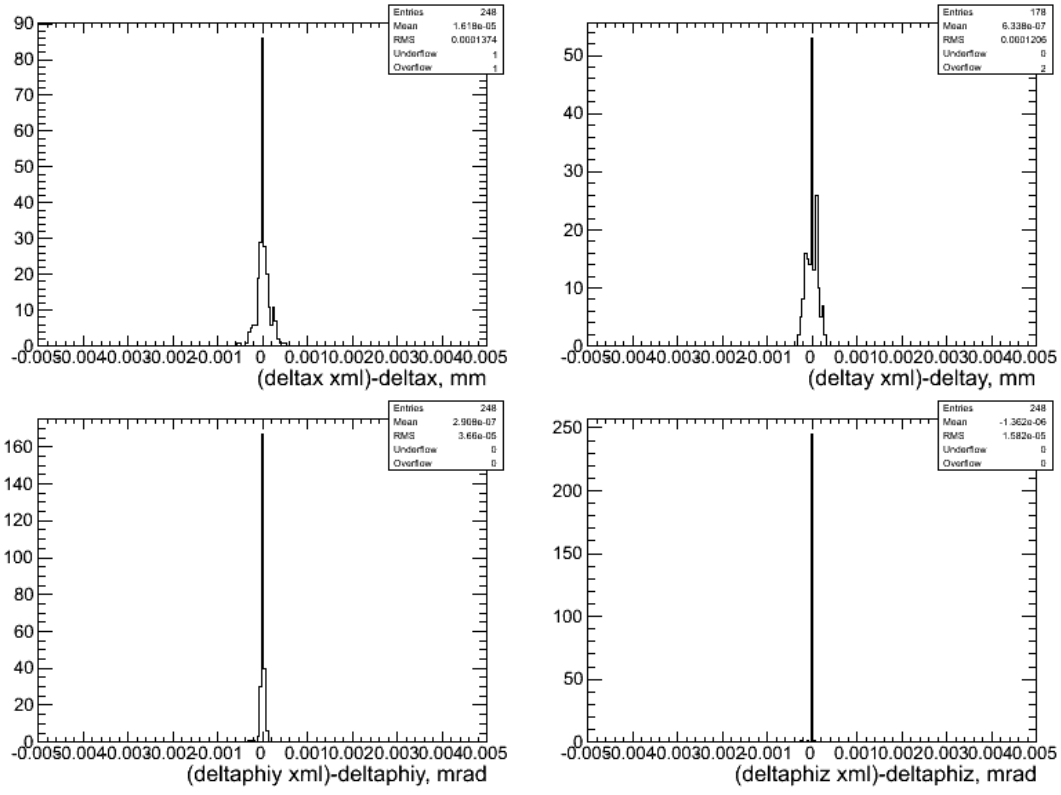


- After alignment results were sent to AlignmentParameters, ±0.3 mrad artificially added to chamber angles, nothing to chamber positions

  gamma = 0.3 mrad in muon GlobalPositionRcd entry: clear sign

Effect on DTAlignmentRcd

- After Andreas fixed it, the alignment fit results are almost exactly equal to what is found in final DTAlignmentRcd (note smaller scale)



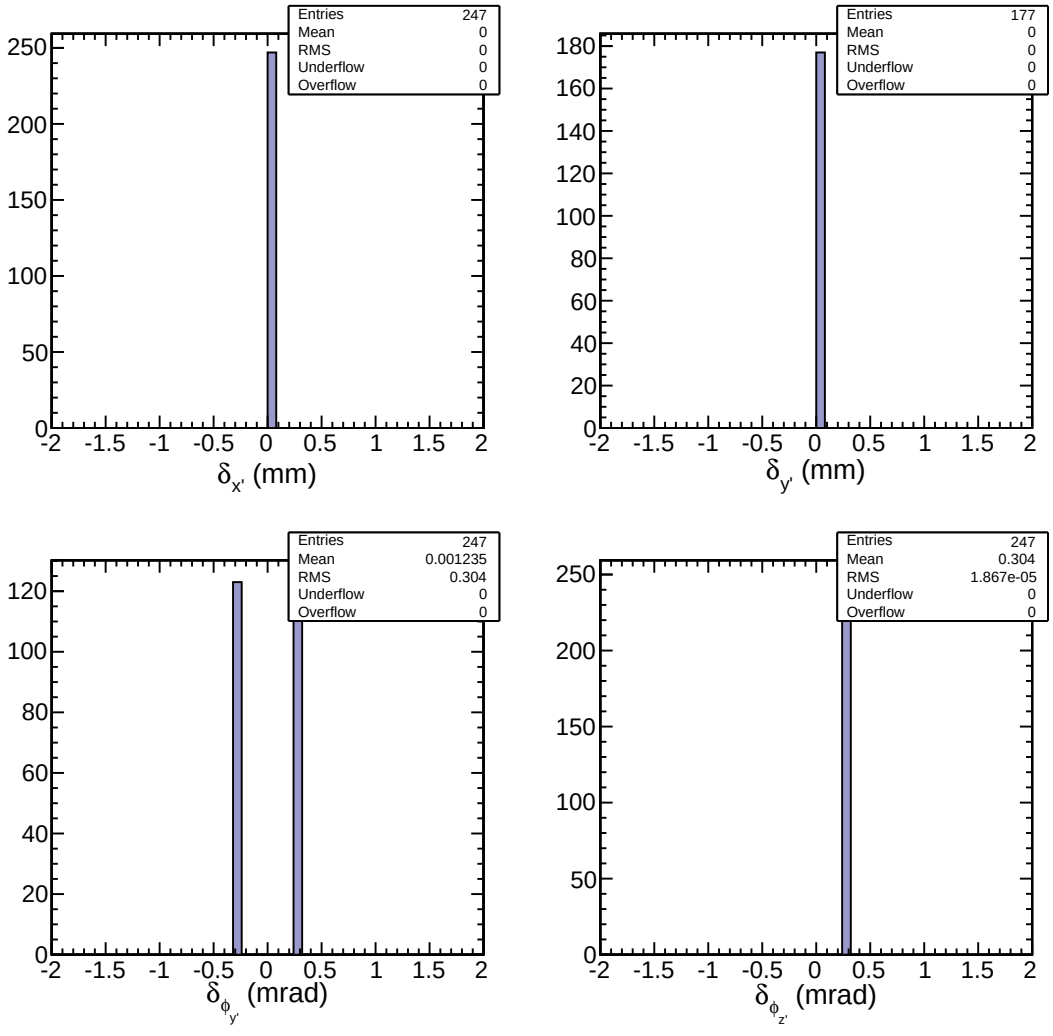
- This is *similar* to things that the Muon Alignment Quality Browser already checks; it is being added to the suite of automated tests
- Also investigating remaining differences: related to single-precision floats in DataFormats/GeometrySurface/interface/Surface.h?

What was affected and how?

- CRAFT-10 DT alignment
  - ± 0.3 mrad in ϕ_y, ϕ_z in first alignment pass (iteration)
  - but effects are nearly cumulative when the procedure is iterated (see next pages)
  - 5 iterations $\rightarrow$ at most 1.5 mrad errors
- CRAFT-10 CSC alignment was *not* affected
  - beam-halo step (internal) was performed with GlobalPositionRcd = (0,0,0,0,0,0)
  - disk alignment step uses GlobalPositionRcd, but not in AlignmentProducer
- Tracker alignments should be unaffected because tracker GlobalPositionRcd entry has no rotation

Effect on DT alignment

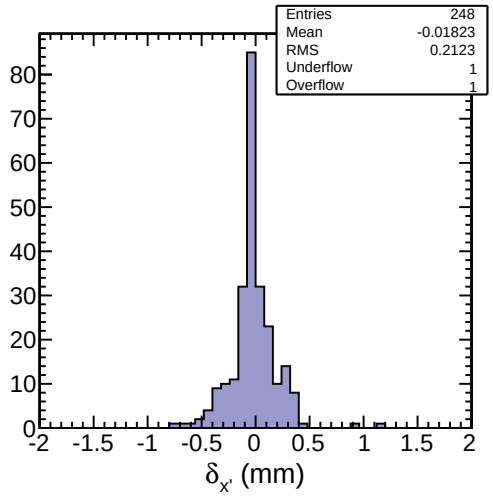
- Plotting (DTAlignmentRcd entries with bug) − (without bug)
- Iteration 1

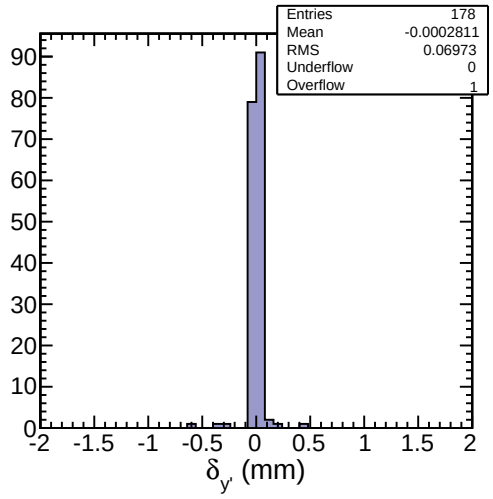


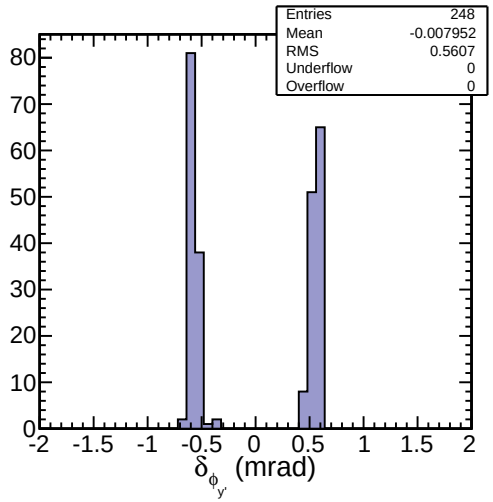
- First iteration differences are exactly ± 0.3 mrad
- Subsequent iterations mix ϕ_y/ϕ_z and x (spread by 250 microns)
- ϕ_y grows linearly, ϕ_z is roughly constant
- Peaks correspond to different wheels

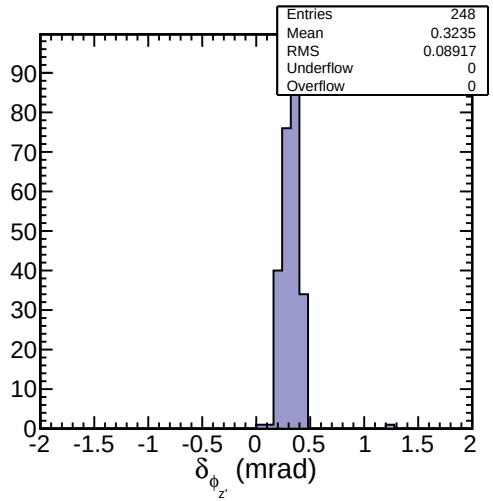
Effect on DT alignment

- Plotting (DTAlignmentRcd entries with bug) − (without bug)
- Iteration 2





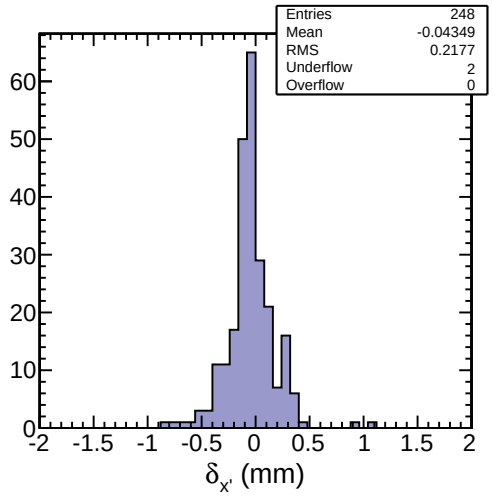


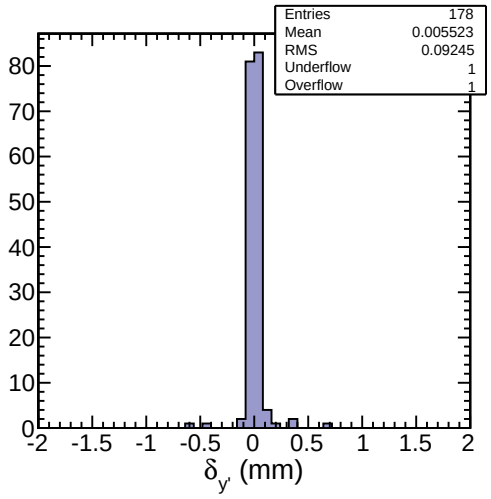


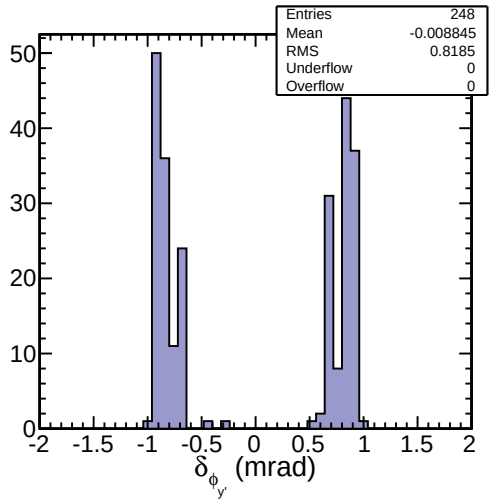
- First iteration differences are exactly ± 0.3 mrad
- Subsequent iterations mix ϕ_y/ϕ_z and x (spread by 250 microns)
- ϕ_y grows linearly, ϕ_z is roughly constant
- Peaks correspond to different wheels

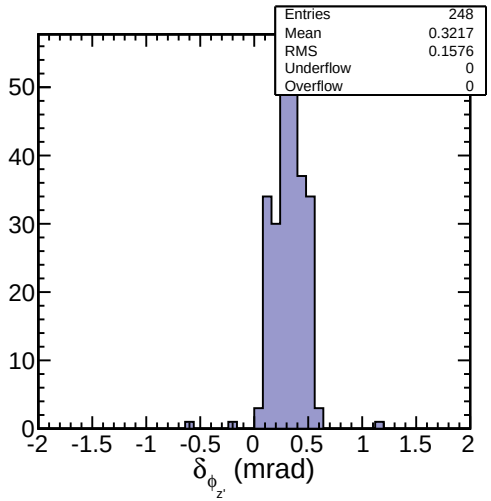
Effect on DT alignment

- Plotting (DTAlignmentRcd entries with bug) − (without bug)
- Iteration 3





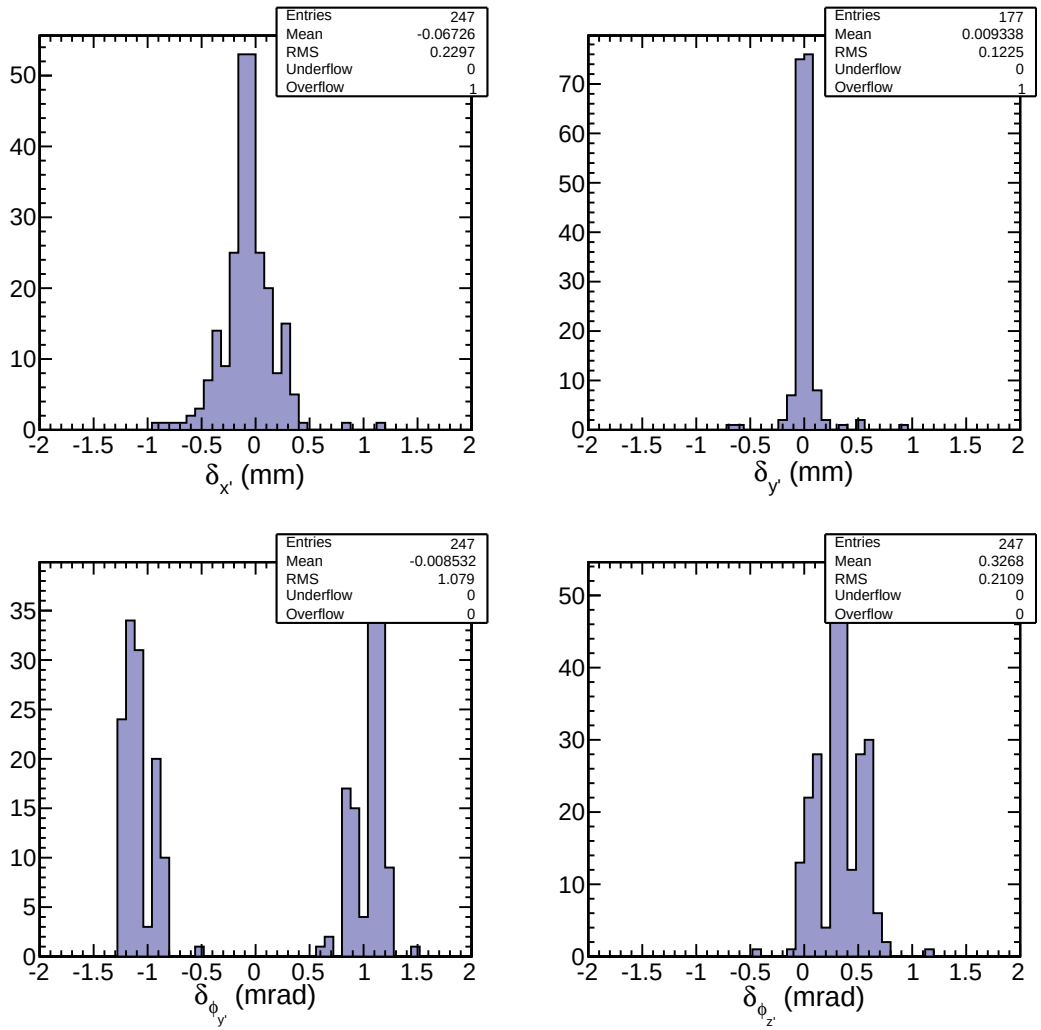




- First iteration differences are exactly ± 0.3 mrad
- Subsequent iterations mix ϕ_y/ϕ_z and x (spread by 250 microns)
- ϕ_y grows linearly, ϕ_z is roughly constant
- Peaks correspond to different wheels

Effect on DT alignment

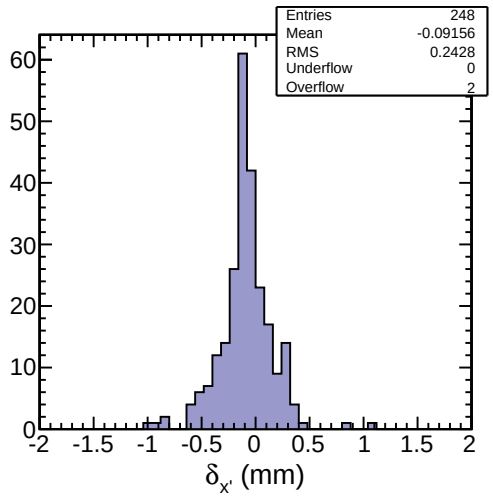
- Plotting (DTAlignmentRcd entries with bug) − (without bug)
- Iteration 4

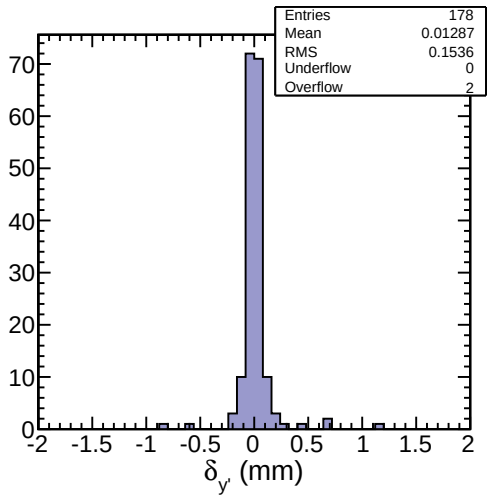


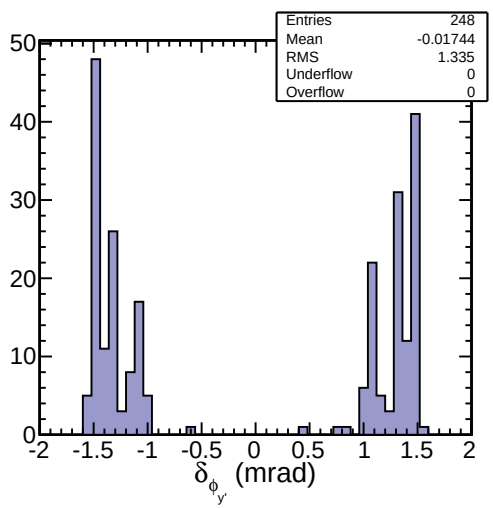
- First iteration differences are exactly ± 0.3 mrad
- Subsequent iterations mix ϕ_y/ϕ_z and x (spread by 250 microns)
- ϕ_y grows linearly, ϕ_z is roughly constant
- Peaks correspond to different wheels

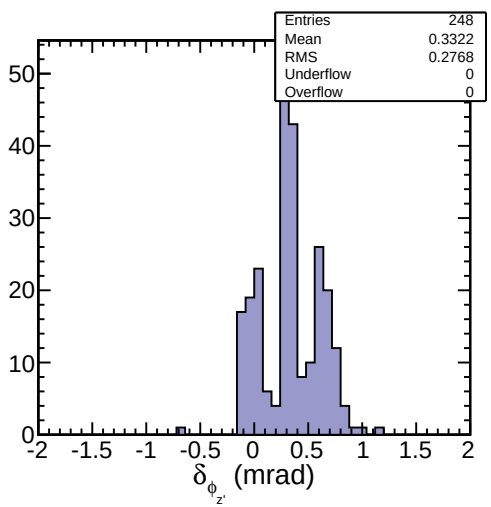
Effect on DT alignment

- Plotting (DTAlignmentRcd entries with bug) − (without bug)
- Iteration 5 (last)



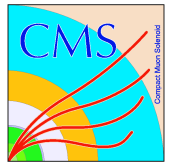






- First iteration differences are exactly ± 0.3 mrad
- Subsequent iterations mix ϕ_y/ϕ_z and x (spread by 250 microns)
- ϕ_y grows linearly, ϕ_z is roughly constant
- Peaks correspond to different wheels

Conclusions



- GlobalPositionRcd-removal is a necessary part of AlignmentProducer
- It was being performed incorrectly, with $x \sim 0.25$ mm, ϕ_y, $\phi_z \sim 1.5$ mrad effects on CRAFT-10 DT alignment
  - a different effect from RPC-hit bias
- Unrelated to track-based/hardware "twist" discrepancy
  - that was present in raw residuals in the first iteration
  - and is much larger (4 mm between wheels ± 2)
- Does not affect CRAFT-10 CSC alignment
- Similar to tests in Muon Alignment Quality Browser (μAQB)
  - μAQB previously only tested x for this kind of divergence
  - adding checks for all alignment parameters