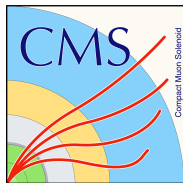


Status report: nAOD-like trigger bits, other stuff, ...

Afiq Anuar

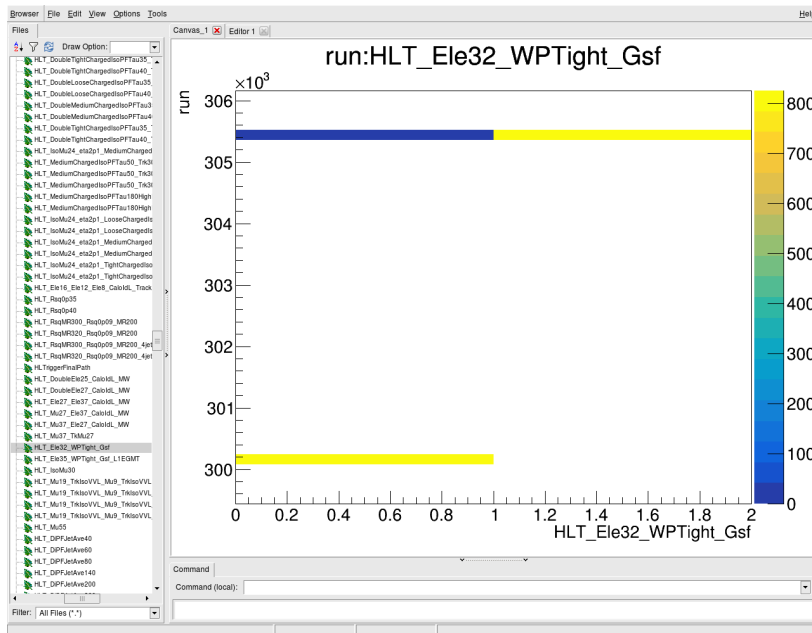
28 August 2019



Introduction

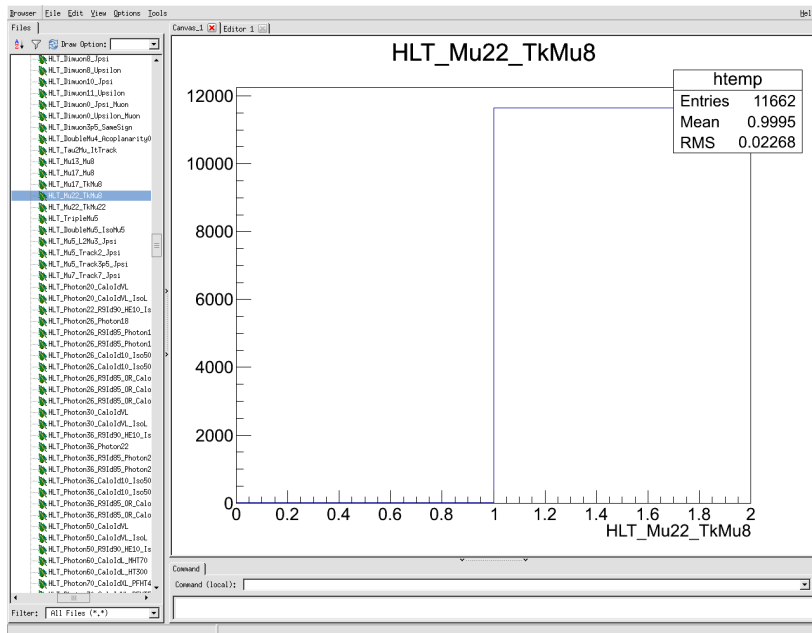
- As [discussed](#) I was in charge of the HLT bits
- Spoiler, they're there
 - Development started with the Dimu7 version of the code
 - Developed on 94X, 2017 data, SingleElectron, runs 300107 and 305518
 - Same source tested to run on 53X and 42X [thanks Nuha for 42X testing](#)
 - Some additions to the config to allow for reHLT corner cases
- Had to remove the NanoTrigger and dataset branches during development
 - Code is still there, but commented out
 - Doesn't coexist with HLT bits so far, care/work needed if they must be
- This talk is mainly about everything else encountered in between

94X HLT validation



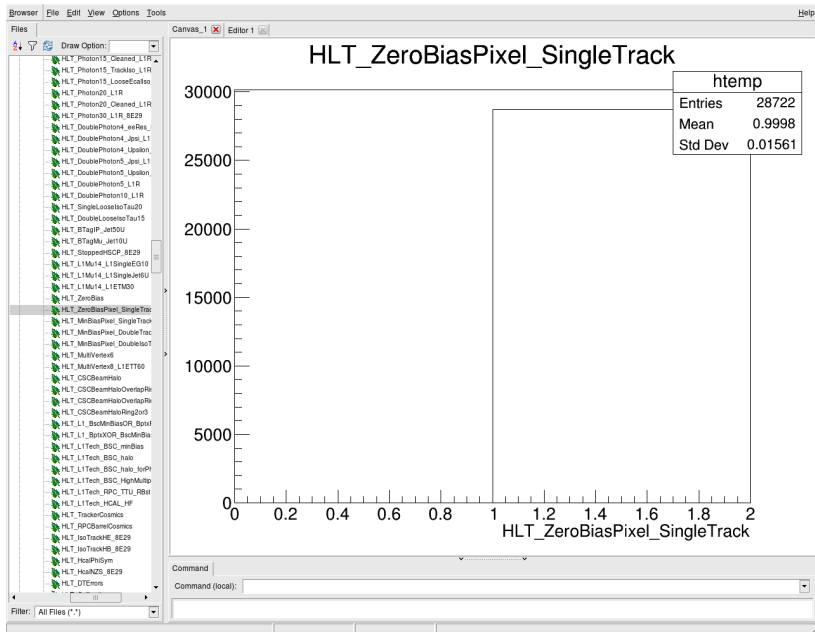
Saw what I expected to see if the code behaved correctly

53X HLT validation



Don't know 2012 menu but this looks right for a DoubleMuParked file

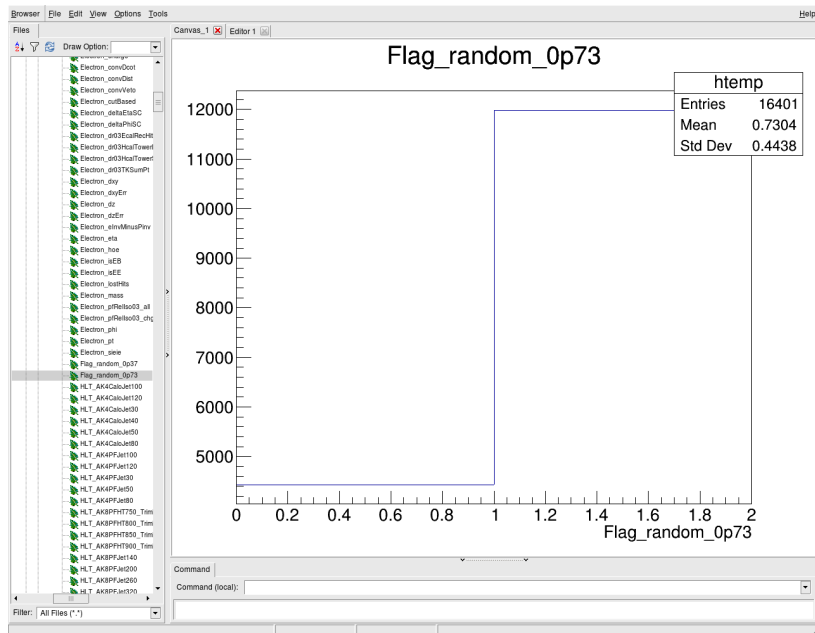
42X HLT validation



Same here but for 2010 ZeroBias

Flags

- Prepared also the flag technicalities
- Based on cms.Path syntax
- Didn't have good flags, so used RNG as example
- Need to discuss what flags to include in each year



And a couple other stuff

- A few niceties:
 - CMSSW branch now picks up the correct release number sans pre/patch shenanigans
 - nanoExtension flag is expanded to skip the analysis code when the branch isn't saved not a finished job, main goal was to remove meson part
 - Reduced hands-on ifdefs, CMSSW flags no longer need to be set by hand, while mAOD needs setting only when running Run 2 AOD
 - makeSL6.sh script picks up what cmsenv sets instead of manually counting dots
 - Pre- vs post-++ in all loops please use post-++ only if you know you need it
- Also some failed attempts:
 - Eliminating ifdefs loop openers works on all but 42X, C++03 is too ancient :(
 - Doable with templates, but that's much tidier with proper refactoring
NanoAnalyzer<pat::MuonCollection, pat::ElectronCollection, ...> anyone? ;)
 - An alternate approach to read-and-sort performance gain, eliminate the sorting loop
 - Didn't fail per se, but tests using std::chrono on muon showed a slowdown instead
 - Approach works better with sparse lists, not the case here so won't pursue further

Afterword

- Delivered the main deliverable of the task
 - Coexistence with previous dataset checker TBD
 - Validated in 94X, tested in 53X and 42X without any changes to source file
 - Documentation could be improved, TBD during integration with master version
- Code in its current form is available in
 - Main directory `/nfs/dust/cms/user/afiqaiize/cms/naod_epr_190717/CMSSW_9_4_14_patch1/src/NanoAOD/NanoAnalyzer/`
 - `src/NanoAnalyzer.cc`
 - `BuildFile.xml`
 - `test/nanoanalyzerData13TeV_cfg.py`
 - `interface/makesSL6.sh`