COMMON TOOLS UPDATES

Viatcheslav Shary

- JES
- Jet Controller
- Luminosity Reweighting
- New in MET
- B-fragmentation
- Other updates

See "Package" web page for recent packages version

JES

- New "stage 1" approved JES version available in caf_util::ApplyJES.
  - Detect automatically p17/p20
  - Depends on vertex Z and hadronic fractions
  - Luminosity dependent: zero-bias overlay luminosity for MC and lumi from TMBGlob for data.

    (a) If luminosity information is not available → estimate it from number of vertex: $L_{inst} = 0.7 \cdot N_{PV}^{0.63}$

    (b) If luminosity presents, but no vertex found, use 1 vertex with 0 Z position

    (c) If no luminosity available and no PV found, use 1 vertex with 0 Z position and follow (a) for luminosity estimation
- Several bug have been corrected: AddDataErrorsToMC, hasMU flag, JES muon correction propagation to MET.

Jet selection using JetController

(see also JetController Wiki page)

Why JetController ?

- **Jet selection for data:**
  1. Applying the certified JES with ApplyJES processor
  2. Recalculating the EM removal flag in TMBJet with EMJetMatching processor
  3. Selecting jets with JetSelector processor and remove jets matched with EM at step (2)
- **MC jets should be correct for data/MC pT resolution and reconstruction efficiency differences:**
  1. Applying the certified JES with ApplyJES processor
  2. Jet oversmearing and threshold removing with RunJetSSR processor (D0 Note 4914)
  3. "Plateau" jet ID efficiency correction with JetCorr and JetRemoval processors (D0 Note 5218).
  4. Recalculating the EM removal flag in TMBJet with EMJetMatching processor
  5. Selecting jets with JetSelector processor and remove jets matched with EM at step (4)

Why JetController ? (2)

- One need to configure all these processors consistently
  - Output branches matched to the input ones
  - JES should be treated coherently
  - Jet ID recommended configurations should be used
  - Systematics estimation require coherent modifications in config file
- Usually several jet branches used in the analysis
  - Several jet bins in analysis
  - Jets for the MET corrections
- Implement all these possibility in JetController and a limited number of configuration parameters available for users.

I/O configuration

- Input branches:
  - JetBranch: input jet branch name. Should be JCCB or JCCA.
  - electronBranch: If the name of the branch is specified, apply EM jet removal.
  - muonBranch: name of the muon branch to correct JES with muon [default: not specified]
- Output branches:
  - Tree: root tree name for the output branches.
  - To: name of the output branch(es) for jets

Output branches

- Several output branches could be specified. Cut on the number of jets could be applied only to the first branch. For all other branches jet selection parameters could be overwritten.

```
cafe.Run: ... JetControler(JCname) ...
JCname.To: BranchName1 BranchName2
JCname.pT: 20.0
BranchName2.pT: 15.0
```

- All intermediate branches are also available. They have names "*JCname_PRname_jets*", where "*JCname*" is a JetController name, "*Prname*" is a processor name. e.g. jets after JetRemoval Processor with JetController name "*mycontr*" saved in the branch "*mycontr_JetRemoval_jets*".

General configuration

- MonteCarlo: must be true for MC and false for data
- JetAlgo: jet reconstruction algorithm. Must be JCCB or JCCA
- JES: the type of jet energy scale correction to use. For the default selection must be "JES" or "JESMU".
- UseUniformRemapping: use uniform remapping (see descriptions of JSSR procedure in D0 Note 4914) [default: true]. For example, top group use "true" for top processes and "false" for all others.

Jet selection

- nJets: minimal number of jets to be selected. Applied only to the first output branch [default: 0]
- nJetsMax: maximal number of jets to be selected. Applied only to the first output branch [default: -1 (no selection)]
- pT: minimal pT cut on jets. Must be at least 13.0, because of JSSR [default: 15.0]
- pTJES: minimal pT cut on jet JES corrected PT [default:15.0]
- Eta or detEta (detector eta)
- There is also possibility to apply dR matching with any TLorentzVector branch.

Jet ID correction

- The JetCorr processor use corrections saved in the *spc format from the jetid_eff packages. Following options used by default and hence could be omitted in config file.
  - JetCorrPath: path to the *spc files [default: "jetid_eff/jetid/"]
  - EffName: [default: "MCJetIDScaleSSR_finedeta"]
  - EffType: [default: "Binned"]
  - EffMethod: [default: "TagAndProbe"]
  - EffVarNames: [default: "deta pt"]
  - EffVersion: [default: 1]
  - ObjQuality: [default: "Standard"]
  - TriggerVersion: [default: "v8-v14"]

For Systematics Studies

- JES: "JESShiftedPos", "JESShiftedNeg", "JESMUShiftedPos", "JESMUShiftedNeg".
- RES: JSSR resolution: “NotShifted”, "ShiftedPos" or "ShiftedNeg".
- EFF: JSSR plateau efficiency. Use "NotShifted", "ShiftedPos" or "ShiftedNeg". Currently ShiftedPos or ShiftedNeg will return exactly the same results, because the Jet ID group recommends to use the negative variation for both positive and negative systematics errors estimation
- Only one parameter "JES" or "RES" or "EFF" could be shifted at once.

Examples

- Simple example:
  - See JetController Wiki page or caf_util/config/test.config
- More sophisticated example: several output jets branches produced (for MET correction, 1 jet exclusive, 2 jets inclusive). JES corrected with muon is used. See caf_util/configs/test_jets.config

Luminosity reweighting
see CSG Wiki FAQ page

How to reweight luminosity profile?

- Tools to reweight MC according to the luminosity in data available in caf_mc_util
- Reweighting include 3 main steps
  - Measuring lumi profile in data
  - Measuring lumi profile in MC
  - Applying the data / MC difference as weight to MC events

Measuring profile in data

- The default luminosity profile for the RunIIa data set, available in *caf_mc_util/spc* directory. To measure profile for the specified set of triggers and run range, follow this instruction for luminosity calculation with dq_util tools, but instead of using getLuminosity script use getLuminosityProfile script.
- The output of this script is the root file with luminosity profile histograms for each trigger and run range. To produce the *spc file use executable SaveDataLumiProfile from *caf_mc_util/util/SaveDataLumiProfile.cpp*.
- Be sure to apply the same triggers and DQ definition as in the analysis.
- You should always use only one trigger per run range to produce the profile. In the case of trigger ORing the unprescaled triggers should be used or triggers with the smallest prescales.

*spc file, what is this?

- We use the *spc file for efficiency bookkeeping. These are 1d,2d or 3d histograms in ASCII format with associated meta data.
- These eff_util packages was modified to be able to keep all necessary information for the luminosity profile in teh same format.

```
LumiProfile_Data_1D_in...
DQDefinition : 2006-10-10.CALO.CFT.MUON.SMT.dqdef
DataSetName : EMMU_RunIIa
EffName : LumiProfile
EffType : Binned
EffVarNames : instlumi
ObjType : Data
BinEdgesX : 0 2 4 6 8 10 12 14 16 18 20 22 24 26 28 30 32 34 36 38 40 42 44 46 48 5
0 52 54 56 58 60 62 64 66 68 70 72 74 76 78 80 82 84 86 88 90 92 94 96 98 100 102
104 106 108 110 112 114 116 118 120 122 124 126 128 130 132 134 136 138 140 14
2 144 146 148 150 152 154 156 158 160 162 164 166 168 170 172 174 176 178 180 1
82 184 186 188 190 192 194 196 198 200 202 204 206 208 210 212 214 216 218 220
222 224 226 228 230 232 234 236 238 240 242 244 246 248 250 252 254 256 258 26
0 262 264 266 268 270 272 274 276 278 280 282 284 286 288 290 292 294 296 298 3
00 302 304 306 308 310 312 314 316 318 320 322 324 326 328 330 332 334 336 338
340 342 344 346 348 350 352 354 356 358 360 362 364 366 368 370 372 374 376 37
8 380 382 384 386 388 390 392 394 396 398 400
BinVal : 0 0.0192354 0.00117623 0.00117623
BinVal : 1 0.0887612 0.00191617 0.00191617
BinVal : 2 0.357551 0.00273777 0.00273777
BinVal : 3 1.11355 0.0105897 0.0105897
BinVal : 4 1.63425 0.0115015 0.0115015
BinVal : 5 2.98504 0.0212234 0.0212234
BinVal : 6 4.30311 0.0307326 0.0307326
BinVal : 7 5.52467 0.043489 0.043489
BinVal : 8 5.70519 0.0414964 0.0414964
BinVal : 9 5.72244 0.0479563 0.0479563
BinVal : 10 5.45486 0.0535635 0.0535635
BinVal : 11 4.95092 0.0460166 0.0460166
BinVal : 12 4.3629 0.0404403 0.0404403
u-----XEmacs: LumiProfile_Data_1D_instlumi_Binned.spc    (Fundamental)----L7--Top----
```

Measuring lumi profile for MC

- Create DQ definition **without** event quality.
- Run MeasureLumiProfile processor after bad run/lbns DQ selection.

```
cafe.Run: FindDuplicateEvents(dupl) Stat(eff)
# remove only bad runs and lbns and not bad event
+cafe.Run: CafeDataQualityProcessor(dq)
+cafe.Run: MeasureLumiProfile(lumi)

lumi.OutputDir: results
lumi.DQDefinition: %{dq.dqdef}

############ DATA QUALITY ##############################

dq.mode: MonteCarlo
dq.dqdef: 2006-10-10.CALO.CFT.MUON.SMT_noevtflags.dqdef
```

- Example available at caf_mc_config/lumi_measure.config

Reweighting MC

- You can use caf_mc_util::LumiReWeighting processor to reweight the CAF MC events according to the data /MC luminosity profiles ratio.
- You can run this processor at any position in the analysis chain, but the most convenient one is just after DQ processor
- See full example in caf_mc_util/config/lumi_reweighting.config

```
# MC profile in *spc format

lumi.MCPath: results
lumi.MCEffName: LumiProfile
lumi.MCDQDefinition: 2006-10-10.CALO.CFT.MUON.SMT_noevtflags
lumi.Profiles: 0lp_profile 1lp_profile

0lp_profile MCDataSetName: CAF-MCv3-35678-tmb_p17.09.06
0lp_profile FileMask: t+t_0lp

1lp_profile MCDataSetName: CAF-MCv3-35679-tmb_p17.09.06
1lp_profile FileMask: t+t

# Data profile in *spc format

lumi.DataPath: caf_mc_util/spc
lumi.DataSetName: RunIIa
lumi.DataEffName: LumiProfile
```

Some features

- The default behavior for this processor is not to give too match weight to one event. The maximum available scale weight for an event is 3. You can change this maximum weight or remove it.
- Luminosity reweighting do not change MC normalization.
- For some MC events the luminosity for the overlaid zero bias events is -1. At the moment these events are not taking into account in the lumi profile and weight 1.0 applied to them in the reweighting procedure.
- If the corresponding MC request does not have any events at high luminosity, the reweighing distribution will not have them either, so one should not expect that p17.09.01 MC will match to data profile after reweighing. In any case this tool could be useful to get some feeling how sensitive your analysis to the luminosity effects.

Issues

- Currently the luminosity profile should be measured for each request ids set used in analysis
  - Should we have a MC luminosity profile catalog for common MC request ids?
  - Should we have a tool to sum-up profiles for different MC ids together?

New in MET (ReComputeMET)

- Propagate tau energy correction to MET (only in p18 for now)
- Correct JES corrections in MET
  - Bug fixes: coarse hadronic corrections, JESMU corrections in MC
  - Jet smearing is now correctly propagated to MET, but it looks like MET is now oversmeared in Z->ee+jets and QCD events.
  - Addition to simplify the coherent treatment of the muon / JESMU corrections: remove internally all muons close to jets from MET correction branch while propagate JESMU corrections. Use the same matching procedure as ApplyJES uses. ⇒ The same muon branch can be used for the ApplyJES and ReCopmuteMET processors.

MET Significance

- Require to run MET selector before
- Configuration parameters :
  - .ElectronBranch: branch name with selected electrons [default: EMscone
  - .MuonBranch: branch name with selected muons [default: Muon]
  - .JetBranch: branch name with selected jets [default: JCCB]
  - .LeBobAlgo: jet algo name, should correspond to selectd jet branch [default: JCCB]
  - .metLcut: lower cut on this variable [default: 0]

b-quark fragmentation tool

- See Yvonne's presentation at simulation meeting Jan 10, 2007
- Describe the momentum fraction B-hadron keeps from b-quark with *Bowler* model (contains free parameters to be tuned to the data)
- Reweighting processor available in caf_mc_util to apply event depended weight which can be recalculated for the different model parameters ⇒ should be used for systematics study
- 2 pass approach
  - Measure the Z distribution in MC (bFragNormProc)
  - Apply the new event weight to all B hadron in the event
  - Could also be applied to D hadrons

Some more updates

- Tau
  - Tau selector has been updates (for example, phi crack selection has been added, add possibility to cut on NNe).
  - Add tau energy scale corrections
  - Propagate tau energy scale correction to the MET (only in p18 for the moment)
- Electron Selector: add several type of fiducial cuts, saving leading object to the branch
- Photon selector: contains the certified version of photon selector
- MuonSelector: could apply veto on muons from external branch. Save leading object to an another branch.
- Add processor to smear track. All parameters are copied from the muon one.

Documentation

- CAF web page
- caf_util, caf_mc_util packages
- The most recent list of the common tools packages available at http://www-d0.fnal.gov/Run2Physics/cs/caf/Packages.h
- Important announces and useful discussions is in the d0-caf-users@fnal.gov mailing list.