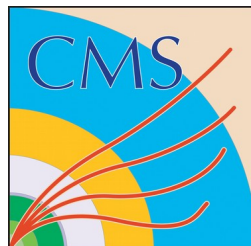中国科学院高能物理研究所
Institute of High Energy Physics
Chinese Academy of Sciences

IHEP Framework Tutorial

Duncan Leggat

2021-10-22

Scope of the code

Main goals

- Create a generic, highly customisable and easily configurable framework,
- Highly parallelisable to exploit Condor cluster.
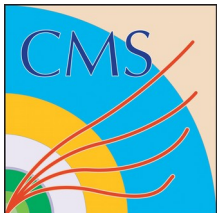
Inputs

- Ntuples,
  - BSM TNT/BOOM trees,
  - nanoAOD,
- Cut and object definitions.

Action on trees

- Extract particles and apply all necessary/relevant corrections,
- Apply customisable selection requirements,
- Plot customisable histograms,
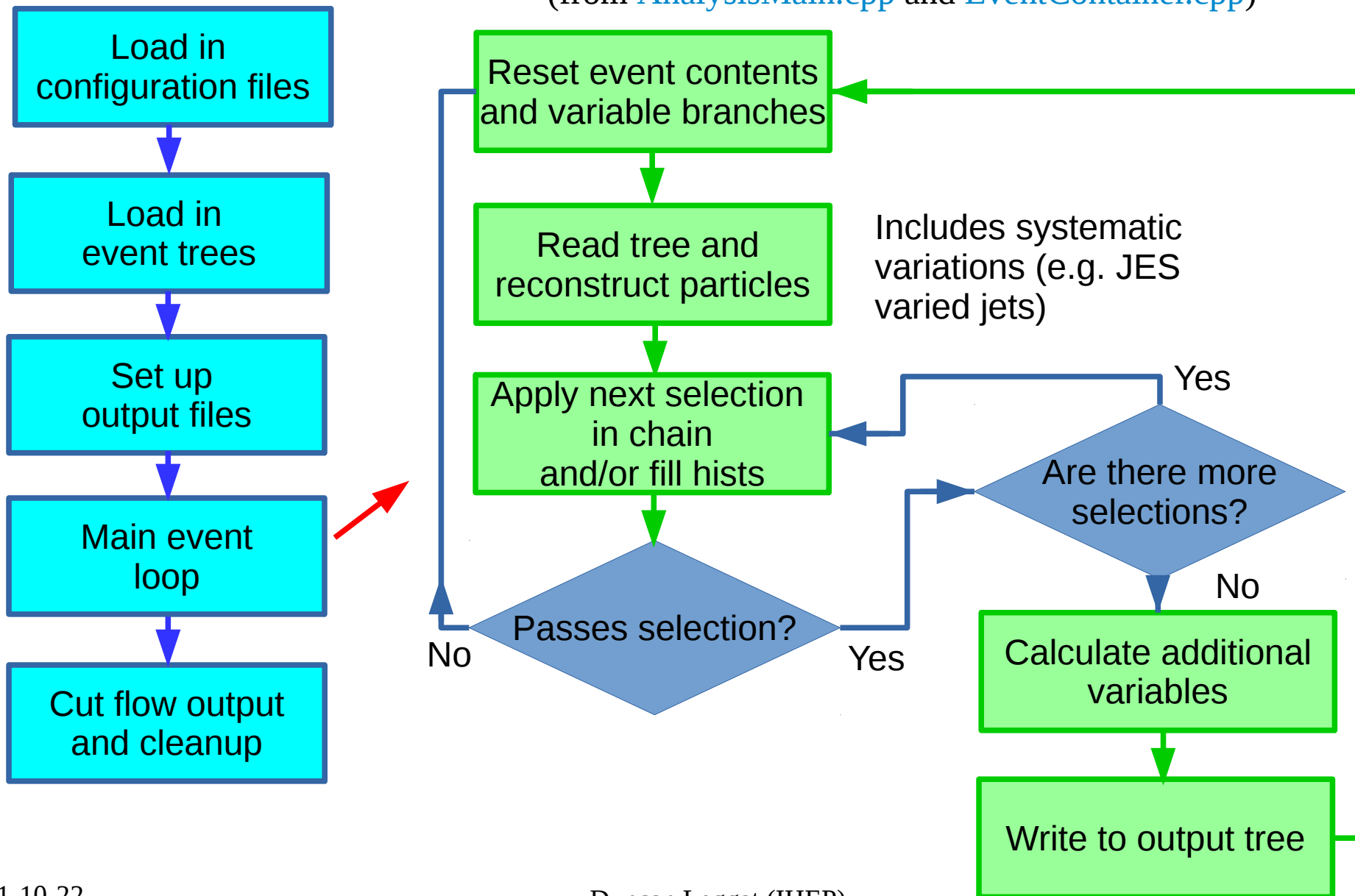- Calculate systematic uncertainties.

Outputs

- Skimmed output tree (including additional variables),
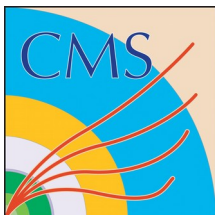- Histogram files.

Flow of the code

Main event loop
(from AnalysisMain.cpp and EventContainer.cpp)

Load in configuration files

Load in event trees

Set up output files

Main event loop

Cut flow output and cleanup

Reset event contents and variable branches

Read tree and reconstruct particles

Apply next selection in chain and/or fill hists

Includes systematic variations (e.g. JES varied jets)

Passes selection?

No

Yes

Are there more selections?

Yes

No

Calculate additional variables

Write to output tree

Github frontpage

Framework front page: https://github.com/leggat/tWIHEPFramework

What's in each folder?

SingleTopRootAnalysis → Headers for all classes

bin/Wt → Analysis executables

config → Configuration files

macros/wt-channel/L… → Object and cut definitions

src → Source code for all classes

tmva → TMVA code

utils → Utilities

.gitignore → Git ignore file

Makefile → Make file

README.md

SingleTop.Wt.LP.m…

newTest.sh → Example to run Wt_generic.x

Key:
Main code
Configurations
Utility Codes

Utility codes are largely focused on tW, but I am currently working on providing generic examples

Main code sub-directories

Main code directories found in SingleTopRootAnalysis (headers) and src (source)

📁 Base	➔ Classes that govern the central running of the code
📁 Cuts	➔ Event selections
📁 Histogramming	➔ Classes for the production of histograms
📁 Particles	➔ Definition of particles
📁 Trees	➔ Event tree (makeclass output)
📁 Vars	➔ Additional variables to store in output tree

How to set up the code

- Set up local directory

Instructions assume use of lxslc7.ihep.ac.cn

- Clone nanoAOD branch:
  - git clone --branch nanoAODDev https://github.com/leggat/tWIHEPFramework

- On lxslc7.ihep.ac.cn machines, set up environment with (can be set to an alias or default command in .bashrc etc.):
  - source /cvmfs/sft.cern.ch/lcg/views/LCG_93/x86_64-slc6-gcc62-opt/setup.sh

- Use command `make Nano' to build framework around the example bin/nano/nanoTest.cpp executable
  - `make clean' removes all

- Run the test command
  - `bash nanoTest.sh'

There should be a bunch of output finished by the following:

```
<CutFlowTable> GlobalCutFlow result:
------------------------------------------------------------------------------------------------
|  Cut Name               |        exposed        |         pass          |         cut          |
|                         |  count  |    yield    |  count  |    yield    |                      |
|      TightMuon.Number.Min | 398904 |     24687.7 |  98059 |     6068.77 | Tight Muon : N >= 1  |
|      TightMuon.Number.Max | 398904 |     24687.7 | 395218 |     24459.6 | Tight Muon : N <= 1  |
|      TightMuon.Number.All | 398904 |     24687.7 |  94373 |     5840.64 | Tight Muon : 1 <= N <= 1 |
|  TightElectron.Number.Min |  94373 |     5840.64 |  94373 |     5840.64 | Tight Electron : N >= 0 |
|  TightElectron.Number.Max |  94373 |     5840.64 |  88852 |     5498.95 | Tight Electron : N <= 0 |
|  TightElectron.Number.All |  94373 |     5840.64 |  88852 |     5498.95 | Tight Electron : 0 <= N <= 0 |
|           Jet.Number.Min |  88852 |     5498.95 |  70265 |     4348.62 | Jet : N >=2          |
|           Jet.Number.Max |  88852 |     5498.95 |  83029 |     5138.58 | Jet : N <= 4         |
|           Jet.Number.All |  88852 |     5498.95 |  64442 |     3988.25 | Jet : 2 <= N <= 4    |
------------------------------------------------------------------------------------------------

<CutListProcessor::PrintCutDetails> Total Efficiency: 3988.25/24687.7 = 0.161548

<AnalysisMain::Loop>
<AnalysisMain::Loop>              -------------------------------
<AnalysisMain::Loop>              |      Processing Summary      |
<AnalysisMain::Loop>              -------------------------------
<AnalysisMain::Loop>              |  # of events:       398904  |
<AnalysisMain::Loop>              |  Real time:         251.6 s |
<AnalysisMain::Loop>              |  Time per event:  0.0006308 s |
<AnalysisMain::Loop>              |  Events/second:       1585  |
<AnalysisMain::Loop>              -------------------------------
<AnalysisMain::Loop>
<AnalysisMain::Loop> Closing histogram file: hists/nanoAODTest_histFile.root
<AnalysisMain::Loop> Write skim file: skims/nanoAODTest_skimFile.root
<AnalysisMain::Loop> Closing skim file: skims/nanoAODTest_skimFile.root
<driver> Finished Loop over events in chain
<driver> Writing histograms to file: hists/nanoAODTest_histFile.root
<driver> Deleting chains
<driver> Sucessful Completion.
```

Example executable

A bunch of pre-amble, but the important bit starts at l.225

Two main vectors to be customised:

Cut list mystudy.AddCut()

- Includes selection requirements and histogramming classes,

- Performed sequentially on each event until the event passes or fails a cut.

Additional variable list mystudy.AddVars()

- Add additional variables to the output skim tree,

- Performed to all events that pass selection.

```
225    ////////////////////////////////////////////////////////////////////////
226    // Add Cuts and Histograms applicable to Fast and Full Analyses
227    ////////////////////////////////////////////////////////////////////////
228    // ******** Cuts and Histograms applied to all studies ********
229
230    mystudy.AddCut(new EventWeight(particlesObj,mystudy.GetTotalMCatNLOEvents(), mcStr, doPileup, dobWeight, useLeptonSFs, usebTagR
231
232    mystudy.AddCut(new HistogrammingMuon(particlesObj,"All"));  // make the muon plots, hopefully.
233    mystudy.AddCut(new HistogrammingMuon(particlesObj,"Tight",useInvertedIsolation));  // make the muon plots, hopefully.
234    mystudy.AddCut(new HistogrammingMuon(particlesObj,"Veto"));  // make the muon plots, hopefully.
235    mystudy.AddCut(new HistogrammingMuon(particlesObj,"UnIsolated"));  // make the muon plots, hopefully.
236    mystudy.AddCut(new CutPrimaryVertex(particlesObj));
237    mystudy.AddCut(new CutTriggerSelection(particlesObj, whichtrig));
238
239    mystudy.AddCut(new HistogrammingMET(particlesObj));
240    //mystudy.AddCut(new CutElectronTighterPt(particlesObj, "Tight"));
241    mystudy.AddCut(new CutMuonN(particlesObj, leptonTypeToSelect));     //require that lepton to be isolated, central, high pt
242
243    mystudy.AddCut(new CutMuonN(particlesObj, "Veto"));     //require that lepton to be isolated, central, high pt
244
245
246
247    mystudy.AddCut(new CutElectronN(particlesObj, leptonTypeToSelect)); //require that lepton to be isolated, central, high pt
248    mystudy.AddCut(new CutElectronN(particlesObj, "Veto")); //require that lepton to be isolated, central, high pt
249
250    mystudy.AddCut(new HistogrammingElectron(particlesObj,leptonTypeToSelect,useInvertedIsolation));  // make the muon plots, hopef
251    mystudy.AddCut(new HistogrammingElectron(particlesObj,"Veto"));  // make the muon plots, hopefully.
252
253    mystudy.AddCut(new HistogrammingMET(particlesObj));
254    mystudy.AddCut(new HistogrammingMtW(particlesObj,useInvertedIsolation));
255
256    mystudy.AddCut(new HistogrammingMuon(particlesObj,leptonTypeToSelect));  // make the muon plots, hopefully.
257
258    //mystudy.AddCut(new CutMuonTighterPt(particlesObj, "Tight")); //require that new Pt cut for leading and subleading muon
259    //if (isemu){
260    //  mystudy.AddCut(new CutEMuOverlap(particlesObj));
261    //}
262    //mystudy.AddCut(new CutJetPt1(particlesObj));
263    mystudy.AddCut(new CutJetN(particlesObj,nJets));
```

Link: https://github.com/leggat/tWIHEPFramework/blob/master/bin/Wt/Wt_generic.cpp

Example cut class

Each cut contains;

- BookHistogram()
  - Book histograms to fill when running cut
  - Also defines the value of the cut

- Apply()
  - Applies the cut to each event
  - Fill histograms
  - Returns true if pass, false if fail cut

Added into analysis executable

```
1   /********************************************************************
2    * CutMissingEt.hpp                                                 *
3    *                                                                  *
4    * Cuts on Missing Et                                               *
5    *                                                                  *
6    * Derived from HistoCut which is in turn derived from BaseCut      *
7    *                                                                  *
8    * Public Member Functions of CutMissingEt class                   *
9    *     CutMissingEt()                     -- Parameterized Constructor *
10   *     ~CutMissingEt()                    -- Destructor             *
11   *     BookHistogram()                    -- Book histograms        *
12   *     Apply()                            -- Apply cuts and fill histograms *
13   *     GetCutName()                       -- Returns "CutMissingEt" *
14   *                                                                  *
15   * Private Data Members of CutMissingEt                             *
16   *     myTH1F* _hMissingEtBefore          -- Hist of MissingEt before cuts *
17   *     myTH1F* _hMissingEtAfter           -- Hist of MissingEt after cuts *
18   *     Int_t _missingEtMin;               -- Minimum Missing Et     *
19   *     Int_t _missingEtMin;               -- Minimum Missing Et     *
20   *                                                                  *
21   * History                                                          *
22   *     15 Dec 2006 - Created by P. Ryan                             *
23   ********************************************************************/
```

```
236   mystudy.AddCut(new CutPrimaryVertex(particlesObj));
237   mystudy.AddCut(new CutTriggerSelection(particlesObj, whichtrig));
```

https://github.com/leggat/tWIHEPFramework/blob/master/src/Cuts/Other/CutMissingEt.cpp

Example histogram class

Histogram classes are treated as cuts but always return true

```
1   /****************************************************************************
2    * HistogrammingMET.hpp                                                    *
3    *                                                                         *
4    * Books and fills histograms                                             *
5    * Used for events passing cuts applied in other classes                   *
6    *                                                                         *
7    * Derived from HistoCut which is in turn derived from BaseCut             *
8    *                                                                         *
9    * Public Member Functions of AnalysisMain class                          *
10   *     HistogrammingMET()              -- Parameterized Constructor        *
11   *     ~HistogrammingMET()             -- Destructor                       *
12   *     BookHistogram()                  -- Book histograms                 *
13   *     Apply()                          -- Fill histograms only (No Cuts)  *
14   *     GetCutName()                     -- Returns "HistogrammingMET"      *
15   *                                                                         *
16   * Private Data Members of this class                                     *
17   *   - histograms of MET                                                   *
18   *                                                                         *
19   * History                                                                 *
20   *     14 Nov 2006 - Created by R. Schwienhorst for ATLAS                  *
21   *     20 Nov 2006 - Modified by Bernard Pope                              *
22   *     21 Mar 2007 - RS: Fill from event container, add sumET, mex, mey    *
23   *****************************************************************************/
```

Added into executable in the cut flow list chain:

```
271     mystudy.AddCut(new HistogrammingMET(particlesObj));
272     mystudy.AddCut(new HistogrammingMtW(particlesObj,useInvertedIsolation));
273     mystudy.AddCut(new HistogrammingJetAngular(particlesObj,useInvertedIsolation));
274     mystudy.AddCut(new HistogrammingJet(particlesObj));
275     mystudy.AddCut(new HistogrammingNPvtx(particlesObj));
```

Link: https://github.com/leggat/tWIHEPFramework/blob/master/src/Histogramming/Recon/HistogrammingMET.cpp

Example of additional variable class

Adds additional variables to the output tree

- Variables are defined in the constructor
  - float, int, vectors of floats and ints are currently supported types

- Filled in the FillBranches() method

Additional variable definition

```
15  WeightVars::WeightVars(Bool_t useIterFit){
16      _floatVars["EventWeight"] = 0.;
17      _floatVars["bWeight"] = 0.;
18      _floatVars["puWeight"] = 0.;
19      _floatVars["lepSF"] = 0.;
20      _floatVars["trigSF"] = 0.;
```

Now the output tree contains additional branches named after these strings

Variable filling

```
39  void WeightVars::FillBranches(EventContainer * evtObj){
40
41      //Fill the nominal event weight variables
42      _floatVars["EventWeight"] = evtObj->GetOutputEventWeight();
43      _floatVars["bWeight"] = evtObj->GetEventbTagReshape();
44      _floatVars["puWeight"] = evtObj->GetEventPileupWeight();
45      _floatVars["lepSF"] = evtObj->GetEventLepSFWeight();
46      _floatVars["trigSF"] = evtObj->GetEventTrigSFWeight();
```

Included in analysis executable in the additional variable chain:

```
289  mystudy.AddVars(new JESBDTVars());
290  mystudy.AddVars(new WeightVars(useIterFitbTag));
291  mystudy.AddVars(new ChannelFlag());
```

Link: https://github.com/leggat/tWIHEPFramework/blob/master/src/Vars/WeightVars.cpp

Duncan Leggat (IHEP)

Running the code

Once we've written our main analysis and cut/histogram/variable classes we need to compile the code

- Most technical bit: edit Makefile to run over your specific code
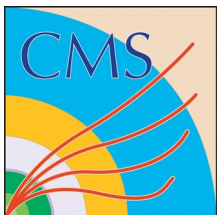
Now we are ready to run the analysis:

bin/myAnalysis.x -config config/overall/myAnalysis.config -inlist config/files/myAnalysisFiles.list -skimfile skimFileOutput/mySkim.root -hfile histFileOutput/myHists.root [additional options]

Necessary parts of execution command:
- Analysis executable
- -config   <config>   – contains links to further configuration files
- -inlist   <inlist>   – a list of files to run over
- -skimfile <skimFile> – what to call the output skim tree (optional)
- -hfile   <hfile>   – histogram file destination (optional)
- Additional opts   – Defined in analysis executable and AnalysisMain.cpp

Commands can also be run on Condor cluster
- Utility classes to help with this are available in the utils folder

Main Configuration File

Important information that the analysis needs

All configuration files are united into one common TEnv area accessible to the analysis code.

Each item is stored in a map locatable by its unique string identifier (e.g. Include.CutsFile)

The cut values to use in the analysis

Information of weights of MC samples

Physics objects defintions

Root file containing pt/eta dependent muon ID SF information

Example for tW analysis:
https://github.com/leggat/tWIHEPFramework/blob/master/config/overall/SingleTop.Wt.LP.mm1%2Bj.muonMSSmeardown.config

```
1   ###################################################################
2   #
3   # Configuration file for the single top analysis.
4   #
5   # Modifications:
6   ###################################################################
7   #
8   #
9   # Set the debug level: the higher, the more printout
10  DebugLevel: 10
11  #
12  # Mass set to MC value for this study.
13  # If we are optimizing for some top mass, set that mass here:
14  TopMass: 172.5
15  #
16  #
17  # Read in a file listing cuts we want to make (electron channel):
18  #Include.CutsFile: macros/wt-channel/LP/config/cuts/SingleTopCuts.Wt.DiElectron.onePlusJets.cuts
19  Include.CutsFile: macros/wt-channel/LP/config/cuts/SingleTopCuts.Wt.ElectronMuon.onePlusJets.cuts
20  #
21  #
22  # Read in a file listing the event weight:
23  Include.WeightsFile: config/weights/MCInformation.weights
24  #
25  #
26  # Read in a file listing the object ID definitions:
27  Include.ObjectIDFile: macros/wt-channel/LP/config/objects/SingleTopObjects.Wt.LP.noshift.config
28  #
29  # Output file with topological variables
30  Topology.SkimFile: topovars.root
31  #
32  #
33  # Events files
34  Include.MCTotEventNumberFile: config/weights/MCInformation.weights
35  #
36  #
37  # pile up vtx reweighting
38  #Include.pileUpReWeightingFile: config/weights/SingleTopWeights.PileUpReWeight.156.weights
39  #
40  #
41  #muonID SFs
42  Include.MuonIDSFsFile: config/weights/muon/MuonID_all.root
```

Examples of cut and object definition files

- Cut definitions are read in by the cut classes

- Object definitions are read by the particle classes

https://github.com/leggat/tWIHEPFramework/blob/master/macros/wt-channel/LP/config/objects/SingleTopObjects.Wt.LP.noshift.config

https://github.com/leggat/tWIHEPFramework/blob/master/macros/wt-channel/LP/config/cuts/SingleTopCuts.Wt.ElectronMuon.onePlusJets.cuts

File list and MC weighting

An example file list file:

```
Name: tW_top_nfh
Number: 500026_1
/publicfs/cms/data/TopQuark/cms13TeV/tWlJetSamples/mc/updatedEleJets/ST_tW_top_5f_NoFullyHadronicDecays_13TeV-powheg_TuneCUETP8M1/crab_tW_top_NFHext1updatedEleJets/180508_153925/0000/OutTree_5.root
/publicfs/cms/data/TopQuark/cms13TeV/tWlJetSamples/mc/updatedEleJets/ST_tW_top_5f_NoFullyHadronicDecays_13TeV-powheg_TuneCUETP8M1/crab_tW_top_NFHext1updatedEleJets/180508_153925/0000/OutTree_6.root
/publicfs/cms/data/TopQuark/cms13TeV/tWlJetSamples/mc/updatedEleJets/ST_tW_top_5f_NoFullyHadronicDecays_13TeV-powheg_TuneCUETP8M1/crab_tW_top_NFHext1updatedEleJets/180508_153925/0000/OutTree_12.root
/publicfs/cms/data/TopQuark/cms13TeV/tWlJetSamples/mc/updatedEleJets/ST_tW_top_5f_NoFullyHadronicDecays_13TeV-powheg_TuneCUETP8M1/crab_tW_top_NFHext1updatedEleJets/180508_153925/0000/OutTree_31.root
/publicfs/cms/data/TopQuark/cms13TeV/tWlJetSamples/mc/updatedEleJets/ST_tW_top_5f_NoFullyHadronicDecays_13TeV-powheg_TuneCUETP8M1/crab_tW_top_NFHext1updatedEleJets/180508_153925/0000/OutTree_7.root
/publicfs/cms/data/TopQuark/cms13TeV/tWlJetSamples/mc/updatedEleJets/ST_tW_top_5f_NoFullyHadronicDecays_13TeV-powheg_TuneCUETP8M1/crab_tW_top_NFHext1updatedEleJets/180508_153925/0000/OutTree_25.root
/publicfs/cms/data/TopQuark/cms13TeV/tWlJetSamples/mc/updatedEleJets/ST_tW_top_5f_NoFullyHadronicDecays_13TeV-powheg_TuneCUETP8M1/crab_tW_top_NFHext1updatedEleJets/180508_153925/0000/OutTree_2.root
/publicfs/cms/data/TopQuark/cms13TeV/tWlJetSamples/mc/updatedEleJets/ST_tW_top_5f_NoFullyHadronicDecays_13TeV-powheg_TuneCUETP8M1/crab_tW_top_NFHext1updatedEleJets/180508_153925/0000/OutTree_20.root
/publicfs/cms/data/TopQuark/cms13TeV/tWlJetSamples/mc/updatedEleJets/ST_tW_top_5f_NoFullyHadronicDecays_13TeV-powheg_TuneCUETP8M1/crab_tW_top_NFHext1updatedEleJets/180508_153925/0000/OutTree_23.root
/publicfs/cms/data/TopQuark/cms13TeV/tWlJetSamples/mc/updatedEleJets/ST_tW_top_5f_NoFullyHadronicDecays_13TeV-powheg_TuneCUETP8M1/crab_tW_top_NFHext1updatedEleJets/180508_153925/0000/OutTree_29.root
```

'Number' corresponds to a specific number in the config/weights/MCInformation.weights file:

Weight.Source.Number: <cross section>
Events.Source.Number: <nEvents in sample>

Plausible that additional settings could be added into this file (i.e. different configuration for run years?) but not included as of now.

```
#500026 = not fully hadronic tW
Weight.Source.500026: 19.559
Events.Source.500026: 11345619

#500027 = not fully hadronic tW_antitop
Weight.Source.500027: 19.559
Events.Source.500027: 11408144

#500028 = w+0jets
Weight.Source.500028: 49670.0
Events.Source.500028: 98083988
```

Cut flow output

At completion the framework outputs a cut flow of the processed trees

- A cut class can be linked the cut flow using

  GetCutFlowTable()->AddCutToFlow(cutName)

- And filled using:

  GetCutFlowTable()->[Pass|Fail]Cut(cutName)

```
<CutFlowJetTable> GlobalCutFlow result:
-----------------------------------------------------------------------------------------------
| Cut Name               | 0 jet | 1 jet | 2 jet | 3 jet | 4 jet | 5 jet | 6 jet | 7 jet | 8+jet | total  |
|                     PV | 25768 |131334 |259142 |255591 |146440 | 60764 | 21016 |  6409 |  2425 | 908889 |
|            MuonTrigger |  7631 | 38972 | 69302 | 60413 | 30588 | 11769 |  3971 |  1132 |   398 | 224176 |
|     TightMuon.Number.Min |  7388 | 36856 | 63229 | 51601 | 24027 |  8765 |  2840 |   791 |   255 | 195752 |
|     TightMuon.Number.Max |  6515 | 35026 | 67179 | 59566 | 30298 | 11686 |  3952 |  1125 |   395 | 215742 |
|     TightMuon.Number.All |  6272 | 32910 | 61106 | 50754 | 23737 |  8682 |  2821 |   784 |   252 | 187318 |
|      VetoMuon.Number.Min |  6272 | 32910 | 61106 | 50754 | 23737 |  8682 |  2821 |   784 |   252 | 187318 |
|      VetoMuon.Number.Max |  5218 | 26438 | 49180 | 39900 | 17709 |  6006 |  1884 |   495 |   145 | 146975 |
|      VetoMuon.Number.All |  5218 | 26438 | 49180 | 39900 | 17709 |  6006 |  1884 |   495 |   145 | 146975 |
|  TightElectron.Number.Min |  5218 | 26438 | 49180 | 39900 | 17709 |  6006 |  1884 |   495 |   145 | 146975 |
|  TightElectron.Number.Max |  5042 | 25907 | 48927 | 39810 | 17682 |  6002 |  1883 |   495 |   145 | 145893 |
|  TightElectron.Number.All |  5042 | 25907 | 48927 | 39810 | 17682 |  6002 |  1883 |   495 |   145 | 145893 |
|   VetoElectron.Number.Min |  5042 | 25907 | 48927 | 39810 | 17682 |  6002 |  1883 |   495 |   145 | 145893 |
|   VetoElectron.Number.Max |  4809 | 22908 | 40990 | 32616 | 13686 |  4395 |  1313 |   348 |    80 | 121145 |
|   VetoElectron.Number.All |  4809 | 22908 | 40990 | 32616 | 13686 |  4395 |  1313 |   348 |    80 | 121145 |
|         Jet.Number.Min |     0 |  1266 | 40990 | 32616 | 13686 |  4395 |  1313 |   348 |    80 |  94694 |
|         Jet.Number.Max |  4809 | 22908 | 40990 | 32616 | 13686 |   488 |     5 |     0 |     0 | 115502 |
|         Jet.Number.All |     0 |  1266 | 40990 | 32616 | 13686 |   488 |     5 |     0 |     0 |  89051 |

<CutFlowTable> GlobalCutFlow result:
-----------------------------------------------------------------------------------------------
| Cut Name               |      exposed      |       pass        |           cut           |
|                        | count  |  yield   | count  |  yield   |                         |
|                     PV | 925904 | 56240.1  | 908889 | 55383.7  | Primary Vertex Cut      |
|            MuonTrigger | 908889 | 55383.7  | 224176 | 13516.1  | Muon Trigger            |
|     TightMuon.Number.Min | 224176 | 13516.1  | 195752 | 11769.5  | Tight Muon : N >= 1     |
|     TightMuon.Number.Max | 224176 | 13516.1  | 215742 | 13024.7  | Tight Muon : N <= 1     |
|     TightMuon.Number.All | 224176 | 13516.1  | 187318 | 11278.1  | Tight Muon : 1 <= N <= 1 |
|      VetoMuon.Number.Min | 187318 | 11278.1  | 187318 | 11278.1  | Veto Muon : N >= 1      |
|      VetoMuon.Number.Max | 187318 | 11278.1  | 146975 | 9003.72  | Veto Muon : N <= 1      |
|      VetoMuon.Number.All | 187318 | 11278.1  | 146975 | 9003.72  | Veto Muon : 1 <= N <= 1 |
|  TightElectron.Number.Min | 146975 | 9003.72  | 146975 | 9003.72  | Tight Electron : N >= 0 |
|  TightElectron.Number.Max | 146975 | 9003.72  | 145893 | 8944.91  | Tight Electron : N <= 0 |
|  TightElectron.Number.All | 146975 | 9003.72  | 145893 | 8944.91  | Tight Electron : 0 <= N <= 0 |
|   VetoElectron.Number.Min | 145893 | 8944.91  | 145893 | 8944.91  | Veto Electron : N >= 0  |
|   VetoElectron.Number.Max | 145893 | 8944.91  | 121145 | 7418.47  | Veto Electron : N <= 0  |
|   VetoElectron.Number.All | 145893 | 8944.91  | 121145 | 7418.47  | Veto Electron : 0 <= N <= 0 |
|         Jet.Number.Min | 121145 | 7418.47  |  94694 | 5779.79  | Jet : N >=2             |
|         Jet.Number.Max | 121145 | 7418.47  | 115502 | 7076.55  | Jet : N <= 4            |
|         Jet.Number.All | 121145 | 7418.47  |  89051 | 5437.87  | Jet : 2 <= N <= 4       |
```

# Pileup how to

- Add command line option -PileUpWgt to executable command.

- Add lines to configuration file with data and MC PV distribution, and min bias up and down plots for systematic calculations,

  – Plot must be titled/named 'pileup' and be divided into integer bins

```
#pileup reweighting files
Include.dataPVFile: config/weights/dataPV.root
Include.mcPVFile: config/weights/mcPV80X.root
Include.minBiasUp: config/weights/minBiasUpPileupHistogram.root
Include.minBiasDown: config/weights/minBiasDownPileupHistogram.root
```

- See https://twiki.cern.ch/twiki/bin/view/CMS/PileupJSONFileforData for details on calculating PV distributions and syst variants
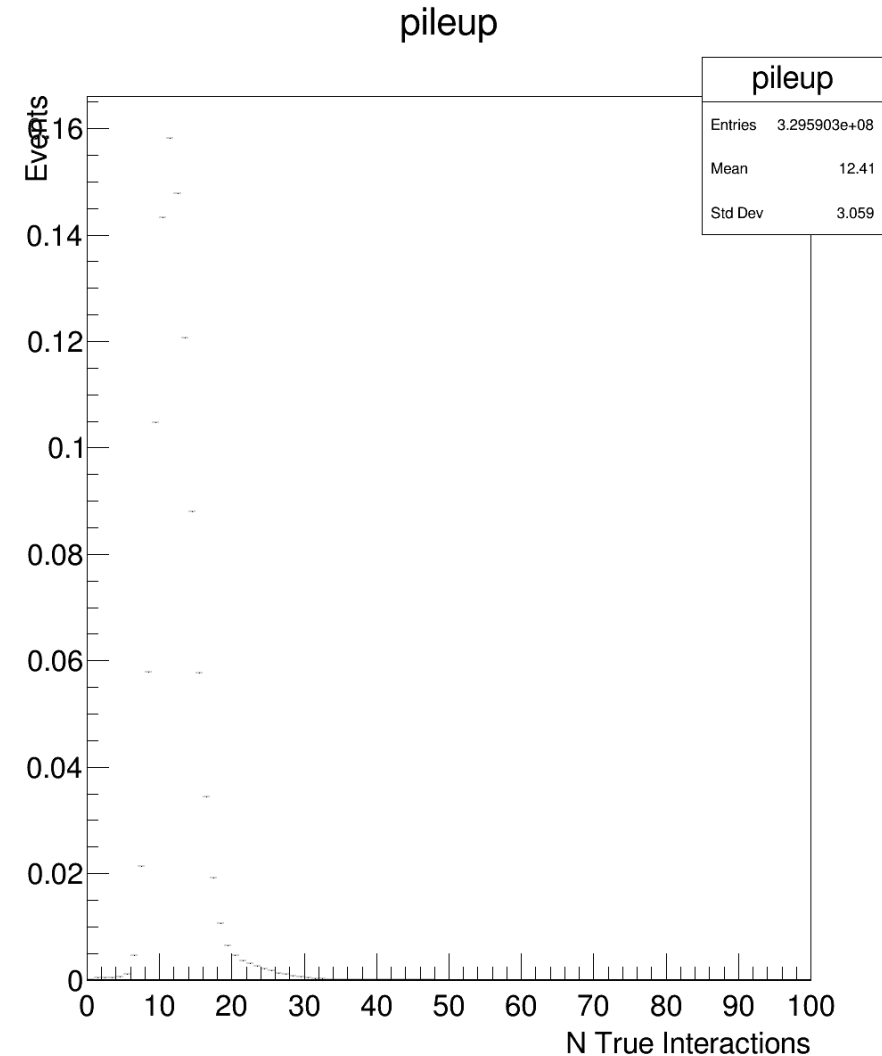
An example pileup distribution
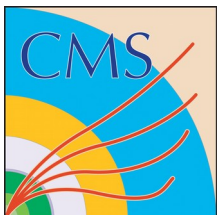
Output histogram file

Example of
output histogram
file

Duncan Leggat (IHEP)
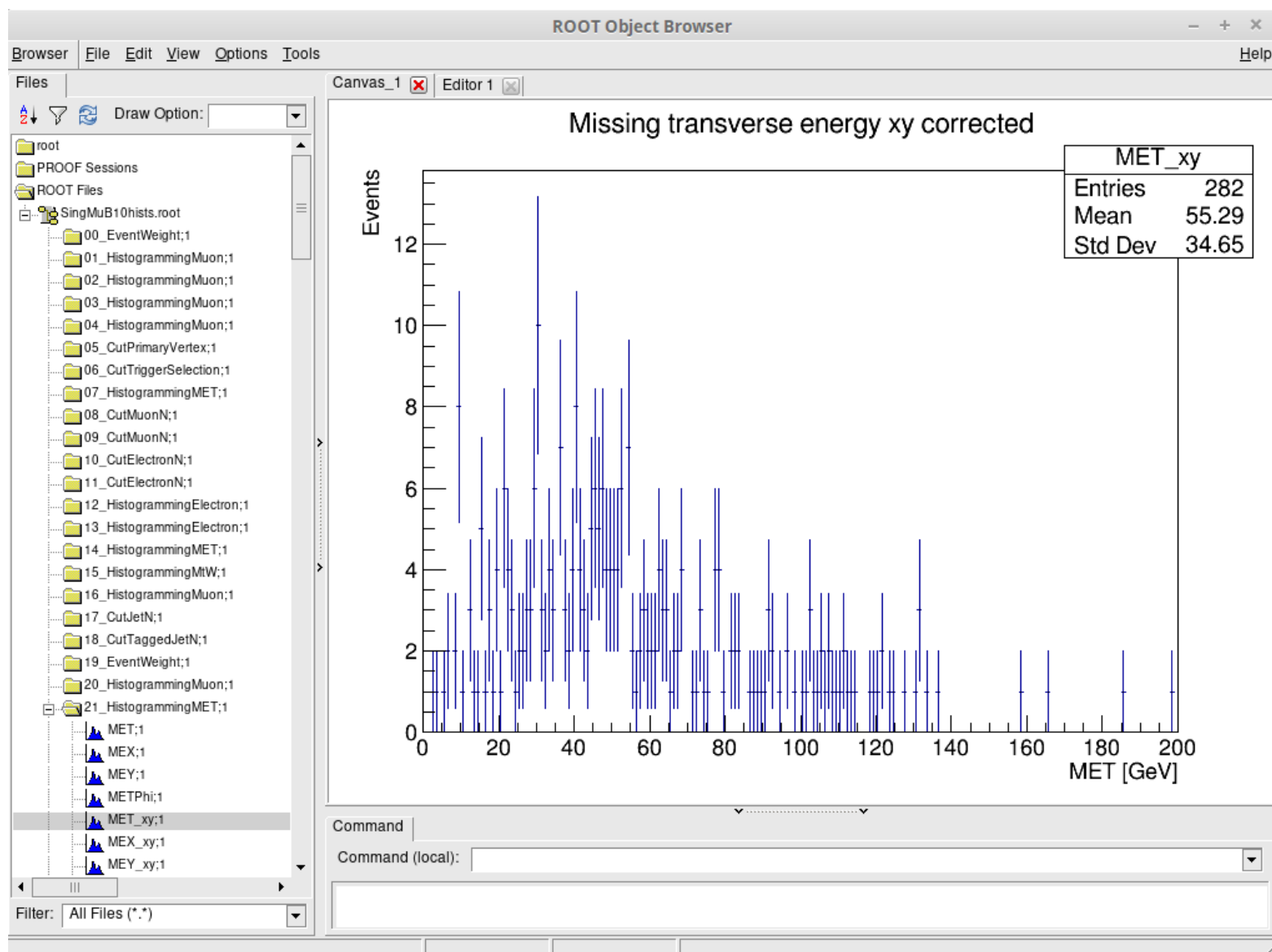
Thoughts on developments

Developments in progress

- Provide native nanoAOD support,

- Continue to provide generic examples of utility codes,

- Improve documentation based on your suggestions?

Potential developments

- Generic histogramming class with basic configuration options (similar to additional variable classes).