CDF/DOC/TOP/PUBLIC/6737
October 22, 2003

TopNtuple: a public analysis module for the Top Group for CDF Run II

Evelyn J. Thomson, Brian L. Winer, Richard E. Hughes, Phillip Koehn,
Valio Kostov, Radu Marginean Ben Kilminster Kevin Lannon
Ohio State University

Erik Brubaker *LBNL*

Koji Sato *University of Tsukuba*

Anyes Taffard *University of Illinos, Champaign-Urbana*

Salvatore Rappoccio *Harvard University*

Shabnaz Pashapour *University of Toronto*

Monica Tecchio *University of Michigan*

Chris Hill *University of California, Santa Barbara*

1 Introduction

TopNtuple is a ROOT tree used by over 90% of analyses in the Top group. It was originally created by the Ohio State University of group for their own use, but was made public as a useful tool for others. Since usage has grown so much, some brief documentation is now required. Hopefully, others will add to this documentation!

2 Layout

TopNtuple is a ROOT tree with several branches for different objects as shown in Figure 1. The folder is a branch. The leaves at this level are the number of objects in a particular branch or the description for a collection in a branch.

Each branch has several leaves containing variables relevant to that object. For example, the leaves on the jet branch are shown in Figure 2. Now you probably wanted to keep information for all the jets in an event, not just the first one. That's why each branch is a ROOT TClonesArray - this allows storage of many objects of the same type in a painless and efficient fashion. The number of jets is stored in the top level leaf *numJetObjs*. You can draw the jet E_T like this:

```
root[ ] TFile f("your_topntuple_root_file_here.root")
root[ ] TopTree->Draw("jet.Et","jet.CollType==3")
```

What is the meaning of *jet.CollType*? Well, you probably wanted to keep information for several different jet algorithms and cone sizes. In fact, several collections of jets with different jet algorithms and cone sizes are usually stored in the jet branch. *jet.CollType* is the most important variable in the jet branch since it allows you to pick only those jets in the collection you want You can find out what the different collections types are by drawing *jetCollDescr1*, *jetCollDescr2*, *jetCollDescr3*, etcetera. Likewise, the most important variables for the electron and muon branches are *electron.CollType* and *muon.ViewType*. The usual definitions are give in Table 1. Generally, a value of 1 means the object passes TopEventModule [1] tight cuts and a value of 2 means the object passes TopEventModule loose cuts. Therefore, 1 and 2 are subsets of 3 and if you just loop blindly over the jet, electron or muon branches without checking the collection type, then the same object will show up twice!

jet.CollType	Descriptor	Description
1	jetCollDescr1	WZTightJetView
2	jetCollDescr2	WZLooseJetView
3	jetCollDescr3	WZ_JETCLU0.4_ReclusterJetColl
4	jetCollDescr4	WZ_JETCLU0.7_ReclusterJetColl
5	jetCollDescr5	WZ_KTCLUS0.5_ReclusterJetColl
6	jetCollDescr6	WZ_MIDPOINT0.4_ReclusterJetColl
electron.CollType	Descriptor	Description
1	eleViewDescr1	WZTightEmObjectView
2	eleViewDescr2	WZLooseEmObjectView
3	eleViewDescr3	defEmObjects
4	eleViewDescr4	PhoenixEmObjectView
muon.ViewType	Descriptor	Description
1	muoViewDescr1	WZTightMuonView
2	muoViewDescr2	WZLooseMuonView
3	muoViewDescr3	defMuons
4	muoViewDescr4	defStublessMuons

Table 1: Usual values for different collections stored in electron and muon branches.

3 Example analysis code

Example analysis code is available from the TopNtuple web page [3]. There are three examples:

- Simple ROOT prompt commands.
- ROOT MakeClass version. Note that you should change the array sizes in .h file to somewhat larger values. ROOT tries to be efficient by making the size for the track branch the same as that in the event with the largest number of tracks. However, since you are probably intend to run over more files than the one you used to do MakeClass and the probability is high that an event in one of those other files will have more tracks, you should increase these numbers to larger values.
- Object-oriented version. You can pass a pointer to an entire object to a method. Link points to current Lepton+Jets example macro for selection of events.

4 Source code and definition of variables

There are a lot of variables in TopNtuple and I am not going to attempt to define them all here! The best place to look is in the source code, which is in these four files: TopMods/src/TopNtuple.cc and HighLevelObjects.cc, TopMods/TopMods/TopNtuple.hh and HighLevelObjects.hh.

5 How to make a TopNtuple

See also CAF attack instructions in next section. The tag for the current version of TopNtuple is given with the TopMods tag on the TopEventModule web page [2]. TopMods/test/topevent.tcl has an example talk-to for TopNtuple. This is reproduced below. Some important features are

- **TopNtuplefile:** Specifies location of your output TopNtuple.
- **eleUseCotParent & co:** Switches must be set to the same values used by TopEventModule so that variables in the TopNtuple are calculated exactly the same way.
- **eleList & co:** Description of object(s) to store in branch. Prefix *MyTop* must be set to the same value used by TopEventModule so that TopEventModule objects are stored.
- **getElectron & co:** Switch on/off filling of branch. Useful to reduce size of TopNtuple. You probably don't want the COT hit information! If you only want HEPG information for W, Z, top and their daughters, and bottom, leptons and neutrinos then set switch *getHepgAll* to 0.

```
module talk TopNtuple

   printLevel set 0
   makeNtuple set 1

   TopNtuplefile set /cdf/scratch/thomsone/topntuple_wztest.root

#For realistic MC with beam offset
     if {$beamoffsetMC ==1} then {
      BeamlineFromDB set false
      mcBeamXPos set -0.064
      mcBeamYPos set 0.310
      mcBeamXSlope set -0.00021
      mcBeamYSlope set 0.00031
      mcBeamXWidth set 0.0025
      mcBeamYWidth set 0.0025
     }
#For perfect MC with NO beam offset
     if {$beamoffsetMC ==2} then {
      BeamlineFromDB set false
     }
```

```
#Must be set to same values as for TopEventModule
#Below are values used for leptonIdCuts set 2 = Run II analysis cuts
        eleUseCotParent set true
        eleNumberOfTowers set 2
        eleBeamConstraint set true
        eleCemCorr set true
        muonBeamConstraint set true
        muonUseCotParent set true

#Describe collections/views you want to read in
#Replace prefix "MyTop" with value of newTopCollectionPrefix in TopEventModule talk-to

        eleList set MyTopTightEmObjectView
        eleList add MyTopLooseEmObjectView
        eleList add defEmObjects
        eleList add PhoenixEmObjectView


        muoList set MyTopTightMuonView
        muoList add MyTopLooseMuonView
        muoList add defMuons
        muoList add defStublessMuons

        jetList set MyTopTightJetView
        jetList add MyTopLooseJetView
        jetList add MyTop_JETCLU0.4_ReclusterJetColl
        jetList add MyTop_JETCLU0.7_ReclusterJetColl
        jetList add MyTop_KTCLUS0.5_ReclusterJetColl
        jetList add MyTop_MIDPOINT0.4_ReclusterJetColl

        metDescription set MyTopMetCorrected
        summaryDescription set MyTopSummary

        otrkList set defTracks
        otrkList add Phoenix_Tracking
        otrkList add PhoenixSI_Tracking

#
# describe blocks you want to fill
        getElectron set 1
        getMuon set 1
        getJet set 1
        getMet set 1
        getOfflTrack set 1

        getTopSummary set 1
```

```
        getTopFit set 1
        getSecvtxTrack set 1
        getTrackJetAssoc set 1

        getJetProbTrack set 1
        getJetProbTrackJetAssoc set 1

        getPrimaryVertex set 1
        getHepg set 1
        getHepgAll set 1
        getOBSP set 1
        getCotHit set 0
        getXFTHit   set 0
        getXFTPixel set 0
        getXFTTrack set 1

        getTrigger set 1
        getTriggerName set 1

    show
exit
```

6 Public TopNtuples

6.1 Data

TopNtuples for the high p_T lepton datasets are described on the official Top Group data page [4].

6.2 MC and CAF attack instructions

The Top Lepton+Jets group staged a highly successful CAF attack to make TopNtuples in release 4.11.1 of many MC samples for the summer conferences. The key was a common exe, tcl file and a standard script for use on the CAF. Instructions can be found here [5]. TopNtuples can be found on fcdfsgi2 in `/cdf/data60/top/TopNtuple_4111/mc`

```
PDFs    Wev2p   WevC0p  Wmv1p   WmvBB2p WmvCC2p WtvBB1p Zmm1p   bb2j     ttop2e
WW0p    Wev3p   WevC1p  Wmv2p   WmvC0p  Wtv1p   WtvBB2p Zmm2p   bb2j_old ttop3e
WW1p    Wev4p   WevC2p  Wmv3p   WmvC1p  Wtv2p   ZZ0p    ZmmCC1p ttop0e   ttop4e
WW2p    WevBB0p WevCC0p Wmv4p   WmvC2p  Wtv3p   Zee1p   Ztt1p   ttop0s   ttop5e
WZ0p    WevBB1p WevCC1p WmvBB0p WmvCC0p Wtv4p   Zee2p   Ztt2p   ttop1e   ttop6e
Wev1p   WevBB2p WevCC2p WmvBB1p WmvCC1p WtvBB0p ZeeCC1p ZttCC1p ttop1s   ttopei
ttopli
```

You can access these files remotely with rootd like this:
`root://fcdfsgi2.fnal.gov//cdf/data60/top/TopNtuple_4111/mc/ttopli/*`

References

[1] E.J. Thomson *et al, Top and Electroweak Event Classification Module for CDF Run II CDF-5947*

[2] TopMods web page
http://www-cdf.fnal.gov/internal/people/links/EvelynThomson/documentation2.html

[3] TopNtuple web page
http://www-cdf.fnal.gov/internal/people/links/EvelynThomson/documentation_topntuple.html

[4] Top Group data samples page
http://www-cdf.fnal.gov/internal/physics/top/topdata/newtopdata.html

[5] CAF attack instructions from K. Bloom
http://www-cdf.fnal.gov/internal/physics/top/RunIIWjets/tools/cafattack.txt

[6] Those with sharp eyes might notice there are two trees. Phillipe Canal said not to worry for three reasons. Firstly, the tree is not duplicated in his entirety, it is only the meta-data (description of the tree) and a limited amount of data (the default is 32K per end-branch) which is duplicated. Secondly, the reason why there is a duplicate is for data-integrity. Every once in a while (this is set by TTree::AutoSave) the tree saves its meta-data so that if the process happens to crash, you will lose only a limited amount of the actual data. Thirdly, you could get rid of the duplication but it is more hassle than it is worth, just always refer to the tree just by its name or double-click on the higher cycle number (i.e. the ;x) and everything will be fine :)

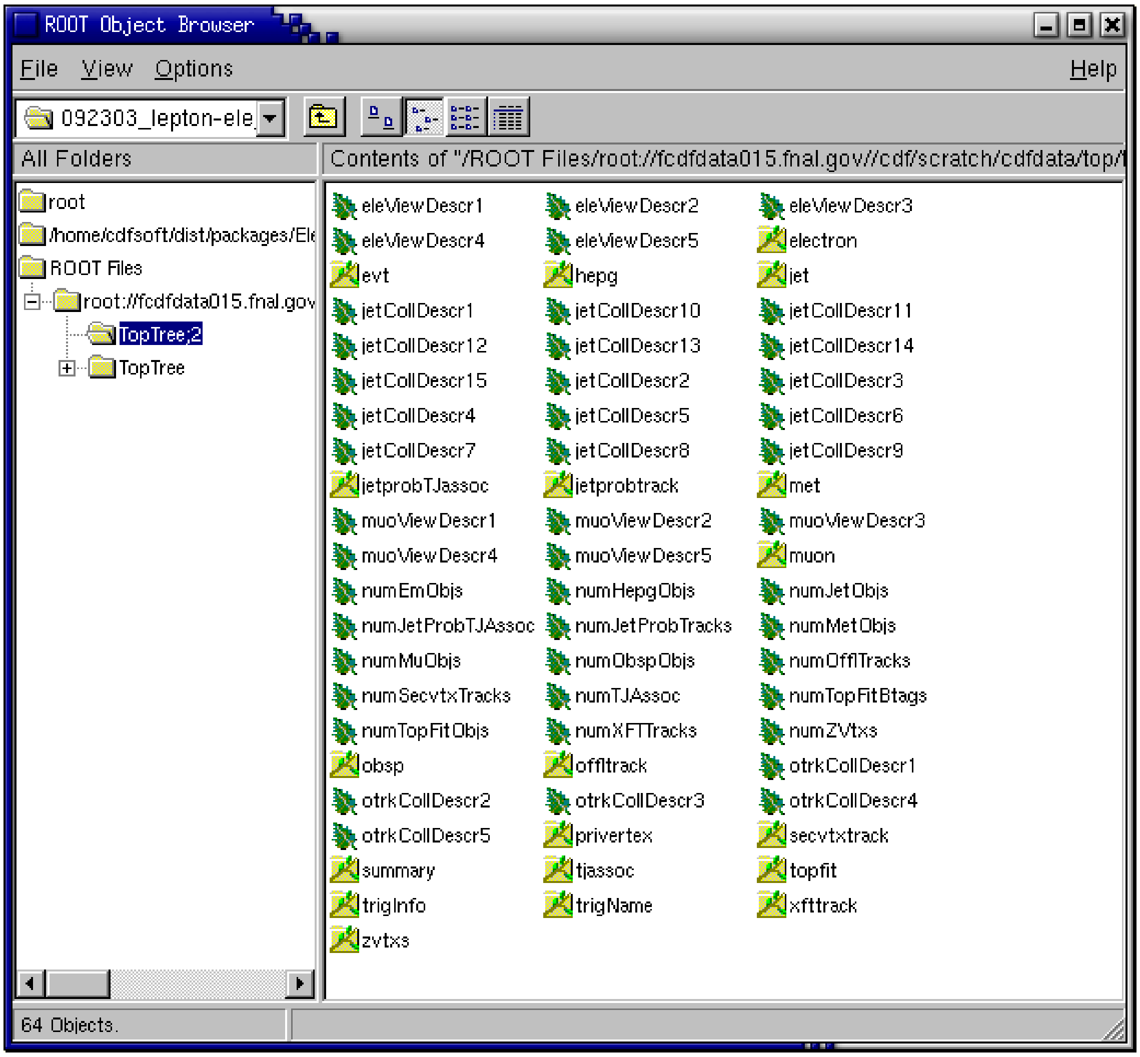

Figure 1: Top level of a TopNtuple tree[6].

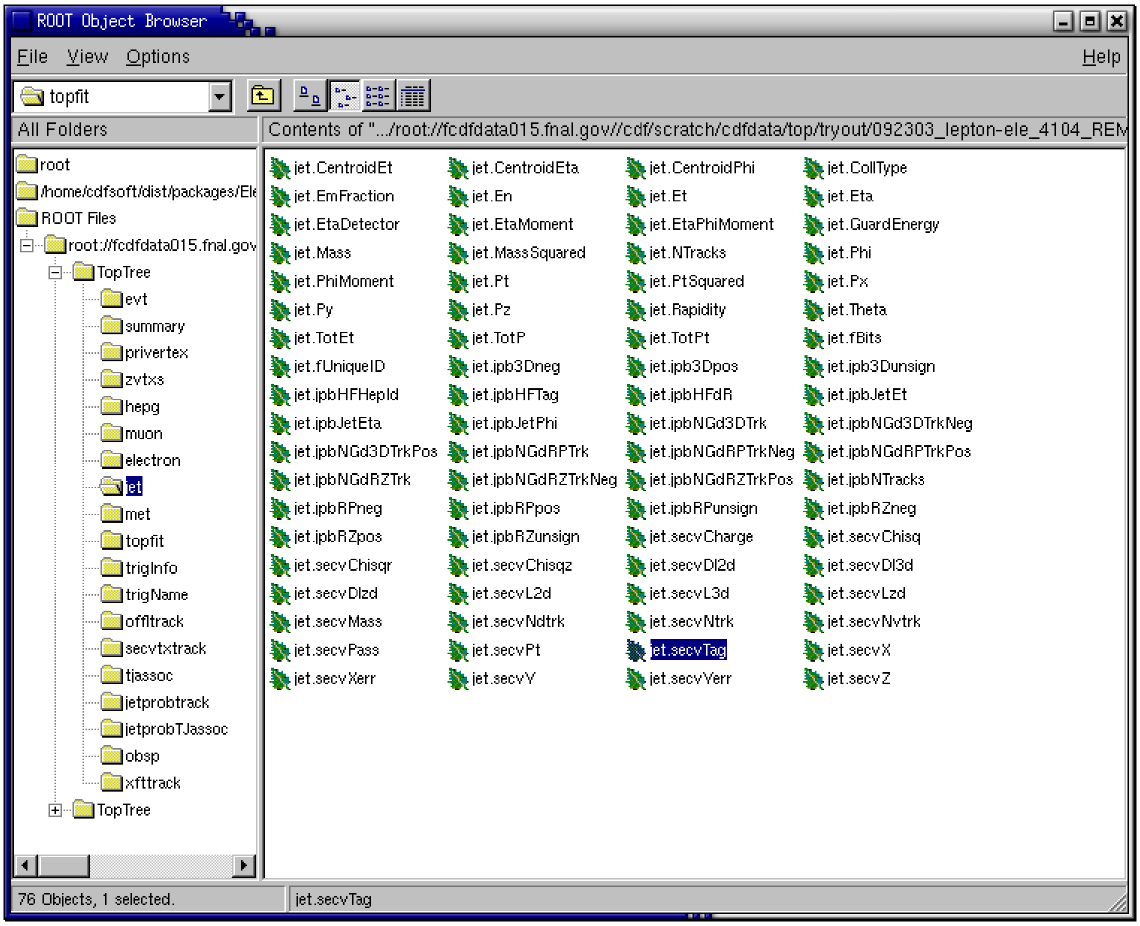

Figure 2: Leaves in the jet branch of a TopNtuple tree