

Table of Contents

Giulio's Notes.....	1
Basic Shell Commands.....	2
Killing programm.....	2
SERVERS.....	3
How to enter in lxplus.....	3
How to enter in galilinux.....	3
How to enter in gridui.....	3
How to enter in cmsanalysis.....	3
Change architecture.....	3
How to enter office computer from home.....	3
CMSSW release.....	4
compile and execute.....	4
Other commands.....	4
Coping in and out your PC.....	4
Screen Session.....	5
Server Analysis.....	5
X Section.....	5
Lxplus folder permission.....	6
Personal Webpage.....	6
Give permissions to other users.....	6
PyRoot.....	6
Load a macro from C.....	6
Array in PyRoot.....	6
Canvas in CMS style.....	6
Two Line Of Pearl.....	7
Nadezda's teaching.....	7
Some combine commands.....	7
Combine.....	8
Edit Analysis Notes.....	8
Notes SVN.....	8
Notes git.....	9
ROOT.....	9
Basic Commands.....	9
Basics.....	9
NTuple from a text file.....	10
Open and close root files.....	10
Directory managment.....	10
Read and write TVector.....	10
TTree.....	11
Other Tree Commands.....	11
Generating random numbers.....	11
TBranch.....	11
TChain.....	12
TCanvas.....	12
Canvas options examples.....	12
TPad.....	13
TEventList and TEntryList.....	14
Histogram.....	14
Axis.....	15
TLine, TText and TLatex.....	16
TLegend.....	17
Styles.....	18
Other Objects.....	19
Colors.....	20

Table of Contents

Giulio's Notes

Integral of a.....	20
Fit of a.....	20
Chi2 and KS probability.....	21
TESORO SEGRETO.....	21
CMSSW.....	21
Reset CPUs.....	21
Getting informations (Handle).....	21
Getting informations after 7_4 (Handle).....	22
Information about root files.....	23
Reading miniAOD from ROOT.....	25
Make and Fill a TTree.....	25
Heppy.....	25
Getting informations after 7_4 (Handle).....	26
Example.....	26
Batch System Pisa.....	26
Tip in order.....	26
What there should be in your executable file.....	27
CRAB.....	27
compile and execute.....	27
Check username.....	27
Job's status.....	27
Configuration file example.....	27
DAS.....	28
Examples.....	28
dasgoclient.....	28
GIT.....	29
Add Package.....	29
Add Package.....	29
Not empty directory.....	29
Create a Repository.....	29
Download a single folder from git repository.....	30
vi.....	30
Writing and erasing.....	30
Saving and quitting.....	30
Helpful links.....	30

Giulio's Notes

<ul style="list-style-type: none"> • Basic Shell Commands <ul style="list-style-type: none"> • Killing programmes • SERVERS <ul style="list-style-type: none"> ◆ How to enter in lxplus ◆ How to enter in galilinux ◆ How to enter in gridui ◆ How to enter in cmsanalysis ◆ How to enter office computer from home • CMSSW release • compile and execute • Other commands • Coping in and out your PC • Screen Session • Analysis on servers <ul style="list-style-type: none"> ◆ X Section <ul style="list-style-type: none"> ◇ Using ◇ GenXSecAnalyzer.cc • lxplus folder permission • Personal Webpage • Give permissions to other users • PyRoot • Load a macro from C • Array in PyRoot • Canvas in CMS style • Two Line Of Pearl • Nadezda's teaching <ul style="list-style-type: none"> • Some combine commands • Combine <ul style="list-style-type: none"> • ImpactPlot • Edit Analysis Notes • Notes SVN • Notes git 	<ul style="list-style-type: none"> • ROOT <ul style="list-style-type: none"> • Basic Commands <ul style="list-style-type: none"> ◆ Basics ◆ NTuple from a text file ◆ Open and close root files ◆ Directory management ◆ Read and write TVector • TTree <ul style="list-style-type: none"> ◆ Other Tree Commands ◆ Generating random numbers • TBranch • TCanvas <ul style="list-style-type: none"> ◆ Canvas options examples ◆ TPad • TEvent List and TEntryList • Histogram <ul style="list-style-type: none"> ◆ Axis • TLine, TText and TLatex • TLegend • Styles • Other Objects • Colors • Integral of a TH1D • Fit of a TH1D • Chi2 and KS probability 	<ul style="list-style-type: none"> • CMSSW <ul style="list-style-type: none"> • Reset CPUs • Getting informations (Handle) • Getting informations (Handle) • Information about root files • Reading miniAOD from ROOT • Make and fill a Tree • Heppy <ul style="list-style-type: none"> • Getting informations (Handle) • Heppy Example) • Batch System Pisa <ul style="list-style-type: none"> • Tip in oder • What there should be in your executable file • CRAB <ul style="list-style-type: none"> • compile and execute • Check username • Job's status • Configuration file example • DAS <ul style="list-style-type: none"> • Examples • dasgoclient • GIT <ul style="list-style-type: none"> • Add package <ul style="list-style-type: none"> ◆ Add package ◆ Not empty directory ◆ Create a Repository ◆ Download a single folder from git repository • VI <ul style="list-style-type: none"> • Writing and erasing • Saving and quitting • Helpful links
---	--	---

Basic Shell Commands

Some Shell Commands are here [↗](#) (italian page)

The ones I use most are

Command	what it does
Ctrl^T	open a new terminal window
Ctrl^d	close the terminal window
ls	list objects in the current directory
ls -a	list also hidden objects
ls -lh	list object and show their size
ls -l file.txt	show mode of file.txt
ls -ltra file*.root	show all the file file*.txt with options l, t, r, and a
ls -d path/*/*	show all the folders without showing their contents
cd name	call directory "name"
cd ..	call the previous directory
rm name	throw the file "name"
mkdir name	make a directory with name "name"
rmdir name	remove directory
rm -r name	remove directory and all its content
chmod 7777 name	change mode
pwd	show the present directory
gedit example &	keep line command on shell free
... > file.txt	put the output in a new file, name it file.txt
... grep BUG	show only the lines containing the word "BUG"
grep BUG file.txt	show only the lines containing the word "BUG" in file.txt
... grep -v BUG	show only the lines not containing the word "BUG"
... grep BUG -a100	show only the lines containing the word "BUG" and 99 lines after that one
... grep BUG -b100	show only the lines containing the word "BUG" and 99 lines before that one
cat file	show as output the file "file"
sdiff file1 file2 tail -n 50	place side by side the two files
clear	clean the shell

On Linux Systems: ALT Gr + ì = ~

Killing programmes

Command	what it does
jobs	show programs in background
kill %n	terminate n-th program in background
kill gdb	terminate all program in background named gdb
kill -9 gdb	
killall gdb	terminate all program in background named gdb
killall -9 gdb	
killall -9 -user yourUsername	kill all the running programmes and log out
ps -ux	show all my running programmes in all the machines
ps -aux	show all running programmes in all the machines

SERVERS

How to enter in lxplus

```
ssh -XY gimandor@lxplus.cern.ch
```

(Ctrl^D or exit to logout)

How to enter in galilinux

```
ssh -XY mandorli@galilinux.pi.infn.it
```

How to enter in gridui

```
ssh -XY mandorli@gridui1NOSPAMPLEASE.pi.infn.it
```

```
ssh -XY mandorli@gridui2NOSPAMPLEASE.pi.infn.it
```

Type in

- source /afs/pi.infn.it/grid_exp_sw/cms/scripts/setcms.(c)sh
- setenv SCRAM_ARCH slc6_amd64_gcc492

How to enter in cmsanalysis

Type in

```
ssh -f -N -XY -L2223:cmsanalysis:22 mandorli@galilinux.pi.infn.it
```

```
ssh -XY mandorli@localhost -p2223
```

Or

```
ssh -XY mandorli@galilinux.pi.infn.it
```

```
ssh -XY mandorli@cmsanalysis.pi.infn.it
```

For SL6:

```
ssh -p 2206 mandorli@cmsanalysis.pi.infn.it
```

Change architecture

```
setenv SCRAM_ARCH slc6_amd64_gcc491
```

```
export SCRAM_ARCH=slc6_amd64_gcc491
```

How to enter office computer from home

TO DO ONE TIME Install sshd

```
sudo apt install openssh-server
```

Look inet address with "ifconfig":

```
indirizzo inet:X.X.X.X .....
```

Look the pc name

```
pc-cms2@pc-cms2:~$ nslookup X.X.X.X
name = pccms62.pi.infn.it.
```

TO ACCESS

```
ssh mandorli@galilinux.pi.infn.it
ssh pc-cms2@pccms62.pi.infn.it
```

CMSSW release

Command	what it does
scram list	list the releases you can download
cmsrel CMSSW_8_X_X	download the release on your directory

compile and execute

Command	what it does
cmsenv	use cms environment
scram b	compile. Do it in src
scram b -j8	compiles with 8 cores
cmsRun cfg.py	execute the configuration file
cmsRun cfg.py >& logFile.txt &	execute the configuration file and put the output in logFile
python file.py	execute the python file

Other commands

Command	what it does
xrdcp root://cms-xrd-global.cern.ch//store/mc/**/MINIAODSIM/**/*.*.root . root://xrootd-cms.infn.it//store/.....	copy locally that root file string to use if you want to read DAS
hadd merged.root /gpfs/ddn/srm/cms/store/user/gimandor/fileNamePt*to*.root	merge that files in a new file named merged.root
hadd out.root `ls *.root grep -v badFile1.root grep -v badFile2.root`	merge that files except for badFile1 badFile2
top	show the status of the cores
scp *.png gimandor@lxplusNOSPAMPLEASE.cern.ch:/afs/cern.ch/user/g/gimandor/www/prova	copy that png in my page on lxplus http://gimandor.web.cern.ch/gimandor
df -h .	show the memory you have
du -h .	disk usage. It shows also files and subdirectories
du -h --max-depth=1	show the disk usage of only the current directory

Coping in and out your PC

In order to copy some files in your pc, use

- scp plotsDirectory_mu/*.png
gimandor@lxplusNOSPAMPLEASE.cern.ch:/afs/cern.ch/user/g/gimandor/www/Hmumu/Hmumu_0523/

How to enter office computer from home

- scp
mandorli@cmsanalysisNOSPAMPLEASE.pi.infn.it:mandorli/CMSSW_7_4_15/src/exampleFile.txt .
- scp exampleFile.txt
gimandor@lxplusNOSPAMPLEASE.cern.ch:/afs/cern.ch/work/g/gimandor/Folder

Screen Session

Command	what it does
screen	open a new screen session
Ctrl^a d	exit from the screen view but keep it to work
screen -r	enter again in the screen session
exit	close the screen session

Server Analysis

X Section

Using brilcalc

Only on lxplus:

```
export PATH=$HOME/.local/bin:/afs/cern.ch/cms/lumi/brilconda-1.0.3/bin:$PATH
brilcalc lumi -i Cert_yourCertificateName_JSON_v2.txt -u /pb --hltpath="HLT_PFJet200_v*"
```

This twiki page explai carefully how to use brilcalc

<https://twiki.cern.ch/twiki/bin/view/CMS/BrilcalcQuickStart>

Using.cc

<https://twiki.cern.ch/twiki/bin/viewauth/CMS/HowToGenXSecAnalyzer> explains how to do it

Otherwise you can use the analyzer:

<https://github.com/cms-sw/cmssw/blob/master/GeneratorInterface/Core/plugins/GenXSecAnalyzer.cc>

```
git cms-addpkg GeneratorInterface/Core
```

Make and run a config file like this:

config file Hide config file

```
import FWCore.ParameterSet.Config as cms
```

```
process = cms.Process("GenFilterEfficiency")
```

```
process.load("FWCore.MessageLogger.MessageLogger_cfi")
```

```
process.maxEvents = cms.untracked.PSet(
    input = cms.untracked.int32(-1)
)
```

```
process.source = cms.Source("PoolSource",
    fileName = cms.untracked.vstring('/store/mc/RunIISummer16MiniAODv2/DYJetsToLL_M-105To160_Tun
)
```

```
process.genXSecAnalyzer = cms.EDAnalyzer('GenXSecAnalyzer',
)
```

```
process.p = cms.Path(process.genXSecAnalyzer)
```

Lxplus folder permission

Personal Webpage

Permission:

```
fs setacl webDirectoryRoot webserver:afs read
afind webDirectoryRoot -t d -e "fs setacl -dir {} -acl webserver:afs read"
```

example path:

```
/afs/cern.ch/user/<letter>/<account>/www
```

Give permissions to other users

Permission:

```
fs listacl                                # enlist all the permission of all the users
fs setacl <directory> <username> <perms>
fs setacl directoryName lgianninni rw
```

PyRoot

There is an example of macro.C here [↗](#)

Load a macro from C

```
import ROOT
ROOT.gInterpreter.ProcessLine('TFile *f = TFile::Open("file.root")')
ROOT.gInterpreter.ProcessLine('TH1D *histo = (TH1D*)f->Get("histoName")')
ROOT.gInterpreter.ProcessLine('.L macro.C')
```

Array in PyRoot

```
from ROOT import *
from array import array
bin = [1,2,3,4,5]
histo= TH1D("histo", "histoTitle", len(binX)-1, array('d',binX))           #'d' stay for 'double'
```

Canvas in CMS style

```
import ROOT
ROOT.gInterpreter.ProcessLine('.L itIsNI.C')
gROOT.LoadMacro("tdrstyleTrigger.C")
setTDRStyle()
gROOT.LoadMacro("CMS_lumi.C")
gROOT.ProcessLine('lumi_7TeV="13 TeV, 3.1 fb^{-1}"')
```



```
gROOT.ProcessLine('extraText = "Simulation"')
```

```
CMS_lumi (canv)
```

Two Line Of Pearl

Introduce the variable 'event' and set it equal to the ninth element of the line when the eighth is 'Event'. the print event each line

```
cat log | awk '{if($8=="Event") { event=$9;} if($1=="stampaDentro") {print event " "$0; } }' > rati
```

print the first element minus 31303000 at the beginning of each line

```
cat new_200.txt | awk '{print $1-31303000 " "$0}' > new_200_v2.txt
```

Nadezda's teaching

Mount EOS ("eos" is the name of the folder, it could be anyone)

```
eosmount eos
```

Compile the code

```
g++ plotter_vbfzll.C -g -o plot `root-config --cflags --glibs` -lMLP -lXMLIO -lTMVA
g++ ./plotter_vbfzll.C LWTNN/src/parse_json.cc -g -o plot `root-config --cflags --glibs` -lMLP
```

Execute the code

```
./plot /afs/cern.ch/user/g/gimandor/private/eos/cms/store/group/phys_higgs/vbfHbb/VBFllskim/DYJet
```

or

```
source run_all.sh
```

Compile the code for plots

```
g++ analyzer_order_trig_stack_pas.cpp -I`root-config --incdir` `root-config --libs` -std=c++0x -o
```

Execute the code for plots

```
./nopas 0
```

Some combine commands

Command	what it does
-S 0	run without including systematics
-t -1	does not look at data. Show only expected results

Option ProfileLikelihood has been substituted with Significance in new version of combine, and MaxLikelihoodFit with FitDiagnostics.

```
combine -M ProfileLikelihood --significance dataCard.txt -t -1 --expectSignal=1 #old version
combine -M Significance --significance dataCard.txt -t -1 --expectSignal=1
combine -M MaxLikelihoodFit --preFitValue=1.0 --robustFit=1 dataCard.txt -n _mu -t -1 --expect
```

```
combine -M Asymptotic --run expected -v 2 dataCard.txt --rMin=-10 --rMax=10
#combine -M Asymptotic -t -1 --expectSignal=1 -n _mu $datacard #shows the data limit
combine -M Asymptotic --run expected dataCard.txt --rMin=-10 --rMax=10
combine -M ProfileLikelihood --X-rtd FITTER_NEW_CROSSING_ALGO --X-rtd FITTER_NEVER_GIVE_UP --s
combine -M ProfileLikelihood dataCard.txt --X-rtd FITTER_NEW_CROSSING_ALGO --X-rtd FITTER_NEVER_
```

If you want to merge two signal sources (see [here](#))

```
text2workspace.py -P HiggsAnalysis.CombinedLimit.PhysicsModel:multiSignalModel --PO verbose --PO
```

Combine

```
text2workspace.py -P HiggsAnalysis.CombinedLimit.PhysicsModel:multiSignalModel --PO verbose --PO
combine -M Significance --significance dataCard.root -t -1 --expectSignal=1
combine -M Significance --significance dataCard.root -t -1 --expectSignal=1 --toysFrequentist
combine -M AsymptoticLimits --run expected dataCard.root --rMin=-10 --rMax=10 -t -1
combine -M FitDiagnostics --expectSignal=1 -t -1 dataCard.root
```

You can find impact plot in the guide [here](#)

```
text2workspace.py -P HiggsAnalysis.CombinedLimit.PhysicsModel:multiSignalModel --PO verbose --PO
combineTool.py -M Impacts -d dataCard.root -m 125 --doInitialFit --robustFit 1 -t -1
combineTool.py -M Impacts -d dataCard.root -m 125 --robustFit 1 --doFits -t -1 # you can add
combineTool.py -M Impacts -d dataCard.root -m 125 -o impacts.json -t -1
plotImpacts.py -i impacts.json -o impacts
```

Run impacts after setting datacard name:

```
combineCards.py datacard1.txt datacard2.txt >> datacard.txt
DC=datacard
text2workspace.py $DC.txt
DC=$DC.root
combineTool.py -M Impacts -d $DC -m 125 --doInitialFit --robustFit 1 -t -1
combineTool.py -M Impacts -d $DC -m 125 --robustFit 1 --doFits -t -1 --parallel 16
combineTool.py -M Impacts -d $DC -m 125 -o impacts.json -t -1
plotImpacts.py -i impacts.json -o impacts
```

Edit Analysis Notes

Notes SVN

```
# SVN UPDATE AN

svn co -N svn+ssh://svn.cern.ch/repos/tdr2 AN
cd AN
svn update utils
svn update -N notes
cd notes
svn update AN-17-234 # (update with the number of the AN you want)

# TO BUILD PDF

eval `./tdr runtime -sh` #or csh
```

Some combine commands

NTuple from a text file

If you want to select and cut some rows:

```
cat log | grep ratio_ | perl -pe 's/ratio_._//g' > new.txt
```

To make the NTuple

```
TNtuple a("a","a","i:j:trueElement:meanElement:ratio:pt:eta:hitNumber:pixelHit:innerHit:outerHit")
a.ReadFile("new.txt")
a.Draw("trueElement", "ratio > 1")
```

Open and close root files

Command	what it does
TFile* f = TFile::Open("plot.root")	Open plot.root and f point to it
TFile* f = new TFile("plot.root")	make the pointer f to the root file, without open it
TFile f ("file.root")	Open "file.root" to read
TFile f ("file.root", "recreate")	to overwrite
TFile f ("file.root", "update")	to append
f->Close()	close the file f

Directory management

Command	what it does
f->mkdir("my_dir")	make a new directory named "my_dir"
f->cd()	change directory between open ones
f->cd("my_dir")	call directory "my_dir"
f->ls()	to see objects in f
gDirectory->pwd()	show the current path
gROOT->Reset()	close every file open and reset the variable to 0

Read and write TVector

Writing in pyRoot

```
f= TFile("ROOT/test.root", "recreate")
f.cd()
variable =TParameter(float)("variableName", 4.)

varVec = TVector (2)
varVec = -3.
varVec = 4.

variable.Write()
varVec.Write("varVecName")
f.Close()
```

Reading in C++

```
((TParameter<float>*) f->Get("variableName"))->GetVal()
TVector *vectorToRead = (TVector*) f->Get("folder/varVecName")
( (*vectorToRead) )[0]
( (*vectorToRead) )[1]
```

TTree

Command	what it does
TTree tree("name_tree", "tree_title")	make a TTree. Usually "tree_name" and the TTree name ("tree" in this case) are put the same
TTree * my_tree; f->GetObject("tree_name", my_tree)	make the TTree pointer "my_tree" point to the TTree "tree_name" in the file f
tree.Print()	show Branchs and number of entries
tree.Fill()	memorize the vatiabes in the Tree
tree.Scan("*")	show all leaves in colomn, if you don't put "*" it show only 8 branches
tree.Scan("leaf_list1:leaf_list2")	show the list of the leaves
tree.Draw("variable_name")	Draw the variable
tree.Write()	save the Tree in the open file root
tree.StartViewer()	open the root windows whit only the tree

Other Tree Commands

Command	what it does
tree.SetMaxTreeSize()	Set an upper limit on the tree memory
tree.AddFriend("tree_name", "root_file_name")	make the tree "tree_name" in root_file_name became friend of tree. It memorize the entries of tree_name
tree.GetEntry(n)	variable_name = (n+1)-th leaf in the branch
tree.GetEntries()	it is the number of entries in the Tree
tree.MakeCode("name_code_file.C")	make a file named "name_code_file.C" that read the tree variables
tree.MakeClass("CLASS_file_name")	make two files: "CLASS_file_name.C" and "CLASS_file_name.h". If execute it, it make a new class initialized with the branch, variable and the values of the tree
tree.GetListOfLeaves()->Print()	show the list of leaves of all branches

Generating random numbers

```

gRandom->Exp(tau)
gRandom->Integer(imax)
gRandom->Gaus(mean, sigma)
gRandom->Rndm() //uniform in (0,1]
gRandom->Uniform(x1) //uniform in (0,x1]
gRandom->Landau(mpv, sigma)
gRandom->Poisson(mean)
gRandom->Binomial(ntot, prob)

```

python code Hide python code

```

gRandom.Exp(tau)
gRandom.Integer(imax)
gRandom.Gaus(mean, sigma)
gRandom.Rndm() #uniform in (0,1]
gRandom.Uniform(x1) #uniform in (0,x1]
gRandom.Landau(mpv, sigma)
gRandom.Poisson(mean)
gRandom.Binomial(ntot, prob)

```

TBranch

Command	what it does
	make a TBranch

tree.Branch("name_branch", &address_variable, "name_leaf_list")	
tree.SetBranchAddress("branch_name", &variable_name)	Opposite of Fill: link the leaves in the branch with the variable "variable_name"
TBranch * tb = tree.GetBranch("Branch_name")	Like GetObject but with the Branch
tb->GetAddress(&struct_name)	Like SetBranchAddress but for structs or classes

```
float muon_pt = 10.;
tree.Branch("muon_pt", &muon_pt, "momentum of the muon/F") // float

std::vector<float> jet_pt;
tree.Branch("jet_pt", jet_pt) // vector

float jet_pt[30] = {10, 5, 3};
tree.Branch("jet_pt", &jet_pt, "momentum of the jets[30]/F") // array
```

▣ python code ▣ Hide python code

```
# nanoAOD postprocessor
self.out.branch("qqMass", "F");
self.out.branch("Jet_VBFselected", "F", 1, "nJet");
```

TChain

Command	what it does
TChain chain ("Tree_name")	make the chain with the tree "Tree_name"
chain.Print()	show entries
chain.Scan("leaf_list1:leaf_list2")	show the list of the leaves
chain.Draw("variable_name")	Draw the variable
chain.GetEntries()	it is the number of entries in the Chain
chain.Reset()	reset the chain

TCanvas

Command	what it does
TCanvas * canv = new TCanvas("canvas name", "canvas title", width, height)	make the canvas "canv"
canv = TCanvas("c", "c", width, height)	initialize the canvas
canv->Divide(n_column,n_row)	divide the canvas in nxm sub-Pads
canv->cd(n)	use the n-th pad
canv->Clear("-D")	clean the canvas keeping the division in pads
canv.SetLogy()	set log scale on y-axis
canv.SaveAs('plot.png')	save the plot
canv.Print('plot.png')	save the plot
tree.Draw("variable", "option")	show the plot in the open (or not) canvas

Canvas options examples

```
canv->Range(-1.744928, -17.19233, 8.755073, 153.8261);
canv->SetFillColor(0);
canv->SetBorderMode(0);
canv->SetBorderSize(2);
```

TBranch

```

canv->SetTopMargin(0.08289242);
canv->SetFrameBorderMode(0);
canv->SetFrameFillColor(3);
canv->SetFrameFillStyle(3002);
canv->SetFrameBorderMode(0);

tree.Draw("money", "money > 1000");
tree.Draw("money", "log(money)");
tree.Draw("money", "sqrt(money)"); //you can use math, you can call func
tree.Draw("money", "money > 1000 && money < 2000");
tree.Draw("money", "money > 1000 || money < 2000"); //you can use logical operator

// You can also define some Cuts and add them
TCut cut1 = "money > 1000" ;
TCut cut2 = "money < 2000" ;
tree.Draw("money", cut1 || cut2);
tree.Draw("money", "cut_name"); // if you cut something on the canvas (View

// You can use histogramms (it define the histogramm)
tree.Draw("money>>histo(100, -10, 10)", cut1 || cut2, "goff"); // with cuts
tree.Draw("money>>histo(100, -10, 10)", "", "goff"); // without cuts

```

[python code](#) [Hide python code](#)

```

canv.Range(-1.744928,-17.19233,8.755073,153.8261)
canv.SetFillColor(0)
canv.SetBorderMode(0)
canv.SetBorderSize(2)
canv.SetTopMargin(0.08289242)
canv.SetFrameBorderMode(0)
canv.SetFrameFillColor(3)
canv.SetFrameFillStyle(3002)
canv.SetFrameBorderMode(0)

tree.Draw("money", "money > 1000")
tree.Draw("money", "log(money) ")
tree.Draw("money", "sqrt(money) ") //you can use math, you c
tree.Draw("money", "money > 1000 && money < 2000")
tree.Draw("money", "money > 1000 || money < 2000") //you can use logical operator

// You can also define some Cuts and add them
TCut cut1 = "money > 1000"
TCut cut2 = "money < 2000"
tree.Draw("money", cut1 || cut2)
tree.Draw("money", "cut_name") # if you cut something on the canvas (View--

// You can use histogramms (it define the histogramm)
tree.Draw("money>>histo(100, -10, 10)", cut1 || cut2, "goff") # with cuts
tree.Draw("money>>histo(100, -10, 10)", "", "goff") # without cuts
tree.Draw("money>>+histo(100, -10, 10)", "", "goff") # add

```

TPad

```

TPad * Pad;
Pad = canv.GetPad(2);
Pad->SetLogx();
Pad->SetLogy();
gPad->SetLogz();

```

[python code](#) [Hide python code](#)

```
Pad = canv.GetPad(2)
Pad.SetLogx()
Pad.SetLogy()
gPad.SetLogz()
```

TEventList and TEntryList

```
//EVENT
tree.Draw(">>event_name", "money < 2500"); // make a EventList "event_name"
TEventList * myEvent;
gDirectory->GetObject("event_name", myEvent);
tree.SetEventList(myEvent); // make the my_event'selection in the
tree.Draw("money"); // draw faster

//ENTRY
tree.Draw(">>event_name", "money < 2500", "entrylist"); // make a EntryList "event_n
TEntryList * myEvent;
gDirectory->GetObject("event_name", myEvent);
tree.SetEntryList(myEvent); // make the my_event'selection in t
tree.Draw("money"); // draw faster
```

python code Hide python code

da scrivere

Command	what it does
EVENT	
tree.SetEventList(myEvent)	make the my_event'selection in the tree
tree.SetEventList(0)	to reset the selection
myEvent->GetEntry(n)	return the n-th index of the good event for the selection in myEvent
myEvent->GetN()	return the total number of selected events
ENTRY	
tree.SetEntryList(myEvent)	make the my_event'selection in the tree
tree.SetEntryList(0)	to reset the selection
myEntry->GetEntry(n)	return the n-th index of the good event for the selection in myEvent
myEntry->GetN()	return the total number of selected events

Histogram

Command	what it does
TH1F histo("histo_name", "histo_title", bin_number, xmin, xmax)	make the histogramm with floats. Usually histo=histo_name
histo.SetBit(TH1::kCanRebin)	set xmin and xmax as the minimum and maximum values. To do before filling
TH1D * histo2 = (TH1D*) histo.Clone()	clone the histogramm histo in histo2
histo.Sumw2()	Set Poissonian errors, they will be drawn
histo.SetTitle("histo_title")	Set the title of the histogramm
histo.FillRandom("gaus", 100)	fill 100 random gaussian variable in histo
histogram1.Draw()	plot the histogram
histogram2.Draw("SAME")	plot the histogram in the same canvas ("SAME" can also be "same")
histo.Scale(1.4)	multiply all the bins for 1.4
histo.Divide(histogram1,histogram2,1,1)	divide the two histograms bin to bin (only if they have the same binning)
histo.Divide(histogram1,histogram2,1,1,"B")	divide the two histograms bin to bin knowing it is a binomial error

<code>void TGraphAsymmErrors::BayesDivide(TH1 * pass, TH1 * total)</code>	divide the two histograms bin to bin with asymmetric errors
--	---

```
//Histogramm declaration
TH2D histogram("histo_name", "histo_title; xAxisName; yAxisName", xbin_number, xmin, xmax, ybin_number, ymin, ymax);
TH2D histogram("histo_name", "histo_title", xbin_number, xmin, xmax, ybin_number, ymin, ymax);
tree.Draw("y_variable:x_variable>>histo(100, -10, 10, 100, -10, 10)", "", "colz"); // with 2D histogram
tree.Draw("y_variable:x_variable>>histo(100, -10, 10, 100, -10, 10)", "", "surf"); // with 3D histogram
tree.Draw("y_variable:x_variable>>histo(100, -10, 10, 100, -10, 10)", "", "lego2"); // with 3D histogram

//Color
histogram->SetLineWidth(1);
histogram->SetLineColor(2); // red
histogram->SetFillColor(2); // red
histogram->SetFillStyle(3001);

//Axis label
histogram->GetXaxis()->SetTitle("This is the x axis");
histogram->GetYaxis()->SetTitle("This is the y axis");
histogram->GetXaxis()->SetTitleOffset(1.5); //the default is 1.

//Set Ranges
histogram->GetXaxis()->SetRangeUser(0, 20); // Set ranges for x axis. It works only for 1D histograms
histogram->GetXaxis()->SetRange(1, 20); // Set initial and final bin
histogram->GetXaxis()->SetLimits(0, 100); // Set initial and final X, it changes only for 2D histograms

//Maximum and minimum values
histo->SetMaximumValue(100); // Set maximum value on y
histo->SetMinimumValue(10); // Set minimum value on y
histo->SetMaximumValue(); // default
histo->SetMinimumValue();
```

[python code](#) [Hide python code](#)

```
# Histogramm declaration
histogramm = TH2D histo("histo_name", "histo_title; xAxisName; yAxisName", xbin_number, xmin, xmax, ybin_number, ymin, ymax);
histogramm = TH2D histo("histo_name", "histo_title", xbin_number, xmin, xmax, ybin_number, ymin, ymax);
tree.Draw("y_variable:x_variable>>histo(100, -10, 10, 100, -10, 10)", "", "colz") # with 2D histogram
tree.Draw("y_variable:x_variable>>histo(100, -10, 10, 100, -10, 10)", "", "surf") # with 3D histogram
tree.Draw("y_variable:x_variable>>histo(100, -10, 10, 100, -10, 10)", "", "lego2") # with 3D histogram

histogramm.SetLineWidth(1)
histogramm.SetLineColor(2) # red
histogramm.SetFillColor(2) # red
histogramm.SetFillStyle(3001)
```

Axis

```
//Axis label
histogram->GetXaxis()->SetTitle("This is the x axis");
histogram->GetYaxis()->SetTitle("This is the y axis");
histogram->GetXaxis()->SetTitleOffset(1.5); //the default is 1.

//Set Ranges
histogram->GetXaxis()->SetRangeUser(0, 20); // Set ranges for x axis. It works only for 1D histograms
histogram->GetXaxis()->SetRange(1, 20); // Set initial and final bin
histogram->GetXaxis()->SetLimits(0, 100); // Set initial and final X, it changes only for 2D histograms

//Maximum and minimum values
histo->SetMaximumValue(100); // Set maximum value on y
histo->SetMinimumValue(10); // Set minimum value on y
histo->SetMaximumValue(); // default
histo->SetMinimumValue();
```

```
// Renaming the bins
for(i==1; i <= binLenght; i++)
    histo->GetXaxis()->SetBinLabel(i, newLabelName[i])
```

python code Hide python code

```
# Axis label
histogram.GetXaxis().SetTitle("This is the x axis")
histogram.GetYaxis().SetTitle("This is the y axis")
histogram.GetXaxis().SetTitleOffset(1.5)           #the default is 1.

# Set Ranges
histogram.GetXaxis().SetRangeUser(0, 20)         # Set ranges for x axis. It works only for zoom
histogram.GetXaxis().SetRange(1, 20)           # Set initial and final bin
histogram.GetXaxis().SetLimits(0, 100)         # Set initial and final X, it changes only t

# Maximum and minimum values
histo.SetMaximumValue(100)                     # Set maximum value on y
histo.SetMinimumValue(10)                     # Set minimum value on y
histo.SetMaximumValue()                       # default
histo.SetMinimumValue()

# Renaming the bins
for i in range(len(binLenght)) :
    histo.GetXaxis().SetBinLabel(i+1, newBinlName[i])
```

TLine, TText and TLatex

```
float yt1 = 0.1;
TText *t1 = new TText(0.5,yt1,"ROOT");
t1->SetTextAlign(22);
t1->SetTextSize(0.05);
t1->Draw();
```

```
float yt2 = 0.8;
float xt2 = 0.2;
TLatex * t2 = new TLatex(xt2,yt2 ,"some text");
t2->SetNDC();
t2->SetTextAlign(22);
t2->SetTextSize(0.05);
```

```
TPaveStats *ptstats = new TPaveStats(0.6560386,0.1569665,0.8560386,0.345679,"brNDC");
ptstats->SetName("stats");
ptstats->SetBorderSize(1);
ptstats->SetFillColor(0);
ptstats->SetTextAlign(12);
ptstats->SetFont(42);
ptstats->SetOptFit(0);
ptstats->Draw();
```

```
TLine *line1 = new TLine(0.05,0.05,0.80,0.70);
line1->SetLineWidth(8);
line1->SetLineColor(2);
line1->Draw();
line1->DrawLine(0.6,0.1,0.9,0.9);
```

python code Hide python code

```
yt1 = 0.1
t1 = TText(0.5,yt1,"some text")
t1.SetTextAlign(22)
t1.SetTextSize(0.05)
```

```

t1.Draw()

yt2 = 0.2
xt2 = 0.7
t2 = new TLatex(xt2,yt2 , "some text")
t2.SetNDC()
t2.SetTextAlign(22)
t2.SetTextSize(0.05)

line1 = TLine(0.05,0.05,0.80,0.70)
line1.SetLineWidth(8)
line1.SetLineColor(2)
line1.Draw()
line1.DrawLine(0.6,0.1,0.9,0.9)

```

TLegend

```

TLegend* myLegend = new TLegend(0.4,0.5,0.7,0.7);
myLegend->SetHeader("The Legend Title");

// you can name it in the declaration
float Xbottom = 0.5;
float Ybottom = 0.5;
float Xtop = 0.9;
float Ytop = 0.9;
TLegend *myLegend=new TLegend(Xbottom, Ybottom, Xtop, Ytop, "The Legend Title");
myLegend->SetBorderSize(0);
myLegend->SetFillColor(0);
myLegend->SetFillStyle(0);
myLegend->SetTextFont(42);
myLegend->SetTextSize(0.04);
myLegend->SetBorderSize(0); // without border
myLegend->SetFillColorAlpha(1,0); // trasparent

//Add Entries and draw
myLegend->AddEntry(h1,"Histogram filled with random numbers","f"); // "f" = color filled
myLegend->AddEntry("f1","Function  $\text{abs}(\frac{\sin(x)}{x})$ ","l"); // "l" = color of the l
myLegend->AddEntry("gr","Graph with error bars","lep");
myLegend->Draw();

```

python code Hide python code

```

myLegend = TLegend(0.4,0.5,0.7,0.7)
myLegend.SetHeader("The Legend Title")

# you can name it in the declaration
Xbottom = 0.5
Ybottom = 0.5
Xtop = 0.9
Ytop = 0.9
myLegend= TLegend(Xbottom, Ybottom, Xtop, Ytop, "The Legend Title")
myLegend.SetBorderSize(0)
myLegend.SetFillColor(0)
myLegend.SetFillStyle(0)
myLegend.SetTextFont(42)
myLegend.SetTextSize(0.04)
myLegend.SetBorderSize(0) # without border
myLegend.SetFillColorAlpha(1,0) #trasparent

#Add Entries and draw
myLegend.AddEntry(h1,"Histogram filled with random numbers","f") # "f" = color filled
myLegend.AddEntry("f1","Function  $\text{abs}(\frac{\sin(x)}{x})$ ","l") # "l" = color of the l
myLegend.AddEntry("gr","Graph with error bars","lep")
myLegend.Draw()

```

Styles

The type of information printed in the histogram statistics box can be selected via the parameter mode. The parameter mode can be = iourmen (default = 0001111)

n = 1; name of histogram is printed

e = 1; number of entries printed

m = 1; mean value printed

r = 1; rms printed

u = 1; number of underflows printed

o = 1; number of overflows printed

i = 1; integral of bins printed

Example:

```
gStyle->SetOptStat(11);
```

print only name of histogram and number of entries.

```
gStyle->SetOptStat(1101);
```

displays the name of histogram, mean value and RMS.

⚠ never call SetOptStat(000111) but SetOptStat(1111), 0001111 will be taken as an octal number !!

```
gROOT.SetStyle("Plain");           // set plot style
gStyle->SetOptStat(0);              // suppress the statistics box
gStyle->SetOptTitle(0);             // suppress the histogram title
gStyle->SetPadTickX(1);             // to get the tick marks on the opposite side of the frame
gStyle->SetPadTickY(1);             // to get the tick marks on the opposite side of the frame
gStyle->SetStatY(0.9);              // Set y-position (fraction of pad size)
gStyle->SetStatX(0.8);              // Set x-position (fraction of pad size)
gStyle->SetStatW(0.2);              // Set width of stat-box (fraction of pad size)
gStyle->SetStatH(0.2);              // Set height of stat-box (fraction of pad size)

gStyle->SetTitleFont(42, "XYZ");     // set nicer fonts
gStyle->SetLabelFont(42, "XYZ");    // set nicer fonts
gStyle->SetPaintTextFormat("2.2f"); // set the text in the bins

gStyle->SetPalette(1);              // set default color in z axis. Each number is a different shad

gPad->SetGrid();                   // set the grid on the plot

// Log scale
gPad->SetLogx(kTRUE);
gPad->SetLogy(kTRUE);
```

python code Hide python code

```
gROOT.SetStyle("Plain")           # set plot style
gStyle.SetOptStat(0)              # suppress the statistics box
gStyle.SetOptTitle(0)            # suppress the histogram title
gStyle.SetPadTickX(1)            # to get the tick marks on the opposite side of the frame
gStyle.SetPadTickY(1)            # to get the tick marks on the opposite side of the frame
gStyle.SetStatY(0.9)              # Set y-position (fraction of pad size)
gStyle.SetStatX(0.8)              # Set x-position (fraction of pad size)
gStyle.SetStatW(0.2)              # Set width of stat-box (fraction of pad size)
gStyle.SetStatH(0.2)              # Set height of stat-box (fraction of pad size)

gStyle.SetTitleFont(42, "XYZ")    # set nicer fonts
gStyle.SetLabelFont(42, "XYZ")    # set nicer fonts
gStyle.SetPaintTextFormat("2.2f") # set the text in the bins

gStyle.SetPalette(1)              # set default color in z axis. Each number is a different shadi
```

```
gPad.SetGrid()           # set the grid on the plot

# Log scale
gPad.SetLogx(kTRUE)
gPad.SetLogy(kTRUE)
```

Other Objects

```
TLine line = new TLine(x1,y1,x2, y2);
line.SetLineWidth(8);
line.SetLineColor(2);
line.Draw();

//TStack
THStack superPositionHisto = new THStack("TStackName", "TStack title")
superPositionHisto->GetHistogram()->GetXaxis()->SetTitle("X title");
superPositionHisto->Add(histo1)
histo1->SetFillColor(2)
superPositionHisto->Add(histo2)
histo2->SetFillColor(3)
superPositionHisto->Draw()

// the following lines move the palette. Choose the values you need for the position. Do it after
gPad->Update();
TPaletteAxis *palette = (TPaletteAxis*)my2dhisto->GetListOfFunctions()->FindObject("palette");
palette->SetX1NDC(0.9);
palette->SetX2NDC(0.95);
palette->SetY1NDC(0.2);
palette->SetY2NDC(0.8);
gPad->Modified();
gPad->Update();

//logarithmic TH (never check in C)
std::vector<double> binX;
std::vector<double> binY;
// Fill binX and binY with a for cycle
TH2D * histo = new TH2D("histo", "", binX.size()-1, binX, binY.size()-1,binY)

//Adding overflow
histo->SetBinContent(histo->GetNbinsX(), histo->GetBinContent(histo->GetNbinsX()) + histo->GetBin
```

[python code](#) [Hide python code](#)

```
line = TLine(x1,y1,x2, y2)
line.SetLineWidth(8)
line.SetLineColor(2)
line.Draw()

#TStack
superPositionHisto = THStack("TStackName", "TStack title")
superPositionHisto.GetHistogram().GetXaxis().SetTitle("X title");
superPositionHisto.Add(histo1)
histo1.SetFillColor(2)
superPositionHisto.Add(histo2)
histo2.SetFillColor(3)
superPositionHisto.Draw()

#the following lines move the palette. Choose the values you need for the position. Do it after
gPad.Update()
palette = my2dHisto.GetListOfFunctions().FindObject("palette");
palette.SetX1NDC(0.88);
palette.SetX2NDC(0.93);
gPad.Modified();
```

```

gPad.Update();

#logarithmic TH2D
from ROOT import *
from array import array
binX = [1, 3, 10, 30, 100]          #do this with a for cycle
binY = [1, 3, 10, 30, 100]        #do this with a for cycle
histo = TH2D("histo", "", len(binX)-1, array('d',binX), len(binY)-1, array('d',binY))

#Adding overflow
histo.SetBinContent(GetNbinsX(), j, histo.GetBinContent(GetNbinsX(), j) + histo.GetBinContent(Get

```

Colors

From -10 to +4
 ROOT.kYellow,
 ROOT.kRed,
 ROOT.kMagenta,
 ROOT.kBlue,
 ROOT.kCyan,
 ROOT.kGreen,

From -9 to +10
 ROOT.kOrange,
 ROOT.kPink,
 ROOT.kViolet,
 ROOT.kAzure,
 ROOT.kTeal,
 ROOT.kSpring,

From 0 to +3
 ROOT.kGray,

ROOT.kBlack,
 ROOT.kWhite,

Integral of a

```

TH1F* h2 = (TH1F*)histo->Clone("h2")
h2->ComputeIntegral();
Double_t *integral = h2->GetIntegral();
h2->SetContent(integral);
h2->Draw("same");

```

python code Hide python code

```

h2 = histo.Clone("h2")
h2.ComputeIntegral()
integral = h2.GetIntegral()
h2.SetContent(integral)
h2.Draw("same")

```

Fit of a

```

TF1* turnonPt = new TF1("turnonPt", "(0.5+0.5*erf((x-[0])/[1]))*[3]+[2] ");

```

```
TF1 * FunFit = new TF1("Funzione", "(0.5+0.5*erf( (x-[0])/[1])) ", -2, 2); //fit in Rang
FunFit->SetParameter(0,2);
FunFit->SetParameter(1,2);
histo->Fit(Funzione, "R"); //Fit in range
histo->Draw();
FunFit->Draw("cont3") //there is "count", "count0",..., "count5"
```

Draw Options are here [↗](#)

python code Hide python code

```
turnonPt = TF1("turnonPt", "(0.5+0.5*erf( (x-[0])/[1]))*[3]+[2] ")

FunFit = TF1("Funzione", "(0.5+0.5*erf( (x-[0])/[1])) ", -2, 2) #fit in Range=[-2,2]
FunFit.SetParameter(0,2)
FunFit.SetParameter(1,2)
histo.Fit(Funzione, "R") #Fit in range
histo.Draw()
FunFit.Draw("cont3")
```

Chi2 and KS probability

```
double chi2 = hist_1->Chi2Test(hist_2, "CHI2/NDF");
double prob_chi2 = TMath::Prob(hist_1->Chi2Test(hist_2, "CHI2"), hist_1->GetNbinsX()-1);
double K_b = hist_1->KolmogorovTest(hist_2);
```

TESORO SEGRETO

Sotto il cumolino, 10 passi a sud dalla quercia più alta

CMSSW

How to generate analyzer, producer and other skeleton code is here.

Reset CPUs

```
scram b clean
scram b -r
```

Getting informations (Handle)

trigger informations Hide CMSSW code

```
edm::Handle<edm::TriggerResults> triggerBits;
edm::Handle<pat::TriggerObjectStandAloneCollection> triggerObjects;
edm::Handle<pat::PackedTriggerPrescales> triggerPrescales;

iEvent.getByToken(triggerBits_, triggerBits);
iEvent.getByToken(triggerObjects_, triggerObjects);
iEvent.getByToken(triggerPrescales_, triggerPrescales);

edm::EDGetTokenT<edm::TriggerResults> trigResultsToken;
trigResultsToken = consumes<edm::TriggerResults>(edm::InputTag("TriggerResults", "", "HLT"));
```

```
edm::Handle<edm::TriggerResults> trigResults;
iEvent.getByToken(trigResultsToken, trigResults);
const TriggerResults & triggerResults = *(trigResults.product());

const edm::TriggerNames& hltPathNames = iEvent.triggerNames(*trigResults);
// PRINT ALL TRIGGER NAMES
for (size_t it = 0 ; it < hltPathNames.size() ; ++it)
    std::cout << hltPathNames.triggerName(it) << std::endl;

std::string pathName="HLT_IsoTkMu24_v3";
bool passTrig=triggerResults.accept(hltPathNames.triggerIndex(pathName));
```

▣ Vertex informations ▣ Hide CMSSW code

```
Handle<reco::TrackCollection> tracks;
iEvent.getByLabel("generalTracks", tracks);

edm::ESHandle<GlobalTrackingGeometry> geometry;
iSetup.get<GlobalTrackingGeometryRecord>().get(geometry);

Handle<edmNew::DetSetVector<SiPixelCluster> > inputPixelClusters;
iEvent.getByLabel(pixelClusters_, inputPixelClusters);

Handle<edmNew::DetSetVector<SiPixelCluster> > inputPixelClustersIDEAL;
iEvent.getByLabel("IdealsiPixelClusters", inputPixelClustersIDEAL);

Handle<std::vector<reco::Vertex> > vertices;
iEvent.getByLabel(vertices_, vertices);

edm::ESHandle<PixelClusterParameterEstimator> pe;
iSetup.get<TkPixelCPERecord>().get(pixelCPE_ , pe );

#include "DataFormats/Candidate/interface/VertexCompositePtrCandidateFwd.h"
Handle<std::vector<reco::VertexCompositePtrCandidate>> SecondaryVerticesCollection;
iEvent.getByLabel("slimmedSecondaryVertices", SecondaryVerticesCollection);
```

▣ objects informations ▣ Hide CMSSW code

```
Handle<reco::TrackCollection> tracks;
iEvent.getByLabel("generalTracks", tracks);

Handle<std::vector<reco::CaloJet> > jets;
iEvent.getByLabel("ak5CaloJets", jets);

Handle<std::vector<pat::Jet>> jetsCollection;
iEvent.getByLabel("slimmedJets", jetsCollection);
std::vector<pat::Jet> jets = *jetsCollection.product();

Handle<std::vector<reco::GenJet>> jetsCollection;
iEvent.getByLabel("slimmedGenJets", jetsCollection);
std::vector<reco::GenJet> jets = *jetsCollection.product();

Handle<std::vector<reco::GenParticle>> particleCollection;
iEvent.getByLabel("prunedGenParticles", particleCollection);
std::vector<reco::GenParticle> particles = *particleCollection.product();
```

Getting informations after 7_4 (Handle)

▣ trigger informations ▣ Hide CMSSW code

```
edm::Handle<edm::TriggerResults> triggerBits;
edm::Handle<pat::TriggerObjectStandAloneCollection> triggerObjects;
edm::Handle<pat::PackedTriggerPrescales> triggerPrescales;

iEvent.getByToken(triggerBits_, triggerBits);
iEvent.getByToken(triggerObjects_, triggerObjects);
```

Getting informations (Handle)


```
iEvent.getByToken(triggerPrescales_, triggerPrescales);
```

▣ Vertex informations ▣ Hide CMSSW code

```
#include "DataFormats/Candidate/interface/VertexCompositePtrCandidateFwd.h"
edm::InputTag m_slimmedSecondaryVertices;
edm::EDGetTokenT<std::vector<reco::VertexCompositePtrCandidate> > slimmedSecondaryVerticesToken;

//or this two line (they have to be in the costructor)
m_slimmedSecondaryVertices = iConfig.getParameter<edm::InputTag>("slimmedSecondaryVertices");
slimmedSecondaryVerticesToken = consumes<std::vector<reco::VertexCompositePtrCandidate> >(m_slimmedSecondaryVertices);
//or the next line (they are equivalent) (it has to be in the costructor)
slimmedSecondaryVerticesToken = consumes<std::vector<reco::VertexCompositePtrCandidate> >(edm::InputTag("slimmedSecondaryVertices"));

Handle<std::vector<reco::VertexCompositePtrCandidate> > SecondaryVerticesCollection;
iEvent.getByToken(slimmedSecondaryVerticesToken, SecondaryVerticesCollection);
std::vector<reco::VertexCompositePtrCandidate> SecondaryVertices = *SecondaryVerticesCollection;

edm::InputTag m_offlineSlimmedPrimaryVertices;
edm::EDGetTokenT<std::vector<reco::Vertex> > offlineSlimmedPrimaryVerticesToken;

//or this two line (they have to be in the costructor)
m_offlineSlimmedPrimaryVertices = iConfig.getParameter<edm::InputTag>("offlineSlimmedPrimaryVertices");
offlineSlimmedPrimaryVerticesToken = consumes<std::vector<reco::Vertex> >(m_offlineSlimmedPrimaryVertices);
//or the next line (they are equivalent) (it has to be in the costructor)
offlineSlimmedPrimaryVerticesToken = consumes<std::vector<reco::Vertex> >(edm::InputTag("offlineSlimmedPrimaryVertices"));

Handle<std::vector<reco::Vertex> > vertices;
iEvent.getByToken(offlineSlimmedPrimaryVerticesToken, vertices);
const reco::Vertex & pv = (*vertices)[0];
```

▣ objects informations ▣ Hide CMSSW code

```
edm::InputTag m_slimmedGenJets;
edm::EDGetTokenT<std::vector<reco::GenJet> > slimmedGenJetsToken;

//or this two line (they have to be in the costructor)
m_slimmedGenJets = iConfig.getParameter<edm::InputTag>("slimmedGenJets");
slimmedGenJetsToken = consumes<std::vector<reco::GenJet> >(m_slimmedGenJets);
//or the next line (they are equivalent) (it has to be in the costructor)
slimmedGenJetsToken = consumes<std::vector<reco::GenJet> >(edm::InputTag("slimmedGenJets"));

Handle<std::vector<reco::GenJet>> jetsCollection;
iEvent.getByToken(slimmedGenJetsToken, jetsCollection);
std::vector<reco::GenJet> jets = *jetsCollection.product();

edm::InputTag m_prunedGenParticles;
edm::EDGetTokenT<std::vector<reco::GenParticle> > prunedGenParticlesToken;

//or this two line (they have to be in the costructor)
m_prunedGenParticles = iConfig.getParameter<edm::InputTag>("prunedGenParticles");
prunedGenParticlesToken = consumes<std::vector<reco::GenParticle> >(m_prunedGenParticles);
//or the next line (they are equivalent) (it has to be in the costructor)
prunedGenParticlesToken = consumes<std::vector<reco::GenParticle> >(edm::InputTag("prunedGenParticles"));

Handle<std::vector<reco::GenParticle>> particleCollection;
iEvent.getByToken(prunedGenParticlesToken, particleCollection);
std::vector<reco::GenParticle> particles = *particleCollection.product();
```

Information about root files

- ls -alrh file.root
- edmEventSize -v file.root

• edmDumpEventContent file.root

NANOAOB doc: https://cms-nanoaod-integration.web.cern.ch/integration/master/mc94X_doc.html

▣ edmEventSize -v MINIAOD 94X ▣ Hide

```
File TTToHadronic_94X_MINIAOD Events 66816
Branch Name | Average Uncompressed Size (Bytes/Event) | Average Compressed Size (Bytes/Event)
patPackedCandidates_packedPFCandidates__PAT. 128048 18601.1
patTriggerObjectStandAlones_slimmedPatTrigger__PAT. 71102.7 3971.92
patPackedGenParticles_packedGenParticles__PAT. 41805.5 3920.25
LHEEventProduct_externalLHEProducer__SIM. 15706.2 2706.68
patMuons_slimmedMuons__PAT. 8291.61 2462.17
patJets_slimmedJets__PAT. 24639.2 2328.35
recoGenParticles_prunedGenParticles__PAT. 12847.1 2324.92
patJets_slimmedJetsPuppi__PAT. 13812 1474.13
patElectrons_slimmedElectrons__PAT. 5337.97 996.794
recoVertexs_offlineSlimmedPrimaryVertices__PAT. 2652.58 902.886
patTaus_slimmedTaus__PAT. 13606.8 859.191
patJets_slimmedJetsAK8__PAT. 6228.59 690.302
patPhotons_slimmedPhotons__PAT. 3421.99 690.153
recoCaloClusters_reducedEgamma_reducedEBEEClusters_PAT. 1778.98 645.852
EcalRecHitsSorted_reducedEgamma_reducedEBRecHits_PAT. 1499.16 636.035
patTaus_slimmedTausBoosted__PAT. 7984.11 516.172
patCompositeCandidates_oniaPhotonCandidates_conversions_PAT. 1896.33 506.745
patPackedCandidates_lostTracks__PAT. 2270.01 497.287
patIsolatedTracks_isolatedTracks__PAT. 2192.58 493.502
recoGenJets_slimmedGenJets__PAT. 2846.62 478.976
recoSuperClusters_reducedEgamma_reducedSuperClusters_PAT. 1727.35 459.931
recoConversions_reducedEgamma_reducedConversions_PAT. 1976.46 458.166
recoCaloClusters_reducedEgamma_reducedESClusters_PAT. 1250.97 390.554
recoGenParticles_genPUProtons_genPUProtons_HLT. 2800.29 355.43
recoGsfTracks_reducedEgamma_reducedGsfTracks_PAT. 836.238 310.43
recoVertexCompositePtrCandidates_slimmedSecondaryVertices__PAT. 738.181 304.499
recoCaloJets_slimmedCaloJets__PAT. 1779.21 300.317
PileupSummaryInfos_slimmedAddPileupInfo__PAT. 1693.04 296.895
patMETs_slimmedMETs__PAT. 889.575 292.045
patMETs_slimmedMETsPuppi__PAT. 889.598 288.865
recoGenJets_slimmedGenJetsAK8__PAT. 2082.17 252.433
patMETs_slimmedMETsNoHF__PAT. 889.593 246.558
EcalRecHitsSorted_reducedEgamma_reducedEERecHits_PAT. 476.88 207.45
recoVertexCompositePtrCandidates_slimmedKshortVertices__PAT. 453.577 184.827
EcalRecHitsSorted_reducedEgamma_reducedESRecHits_PAT. 488.395 179.334
GenEventInfoProduct_generator__SIM. 304.421 133.052
recoDeDxHitInfos_isolatedTracks__PAT. 311.685 122.89
recoGenJets_slimmedGenJetsAK8SoftDropSubJets__PAT. 964.605 121.195
patJets_slimmedJetsAK8PFPuppiSoftDropPacked_SubJets_PAT. 1026.71 119.888
l1tTauBXVector_calostage2Digis_Tau_RECO. 1414.99 117.018
recoJetFlavourInfoMatchingCollection_slimmedGenJetsFlavourInfos__PAT. 1641.03 114.612
l1tJetBXVector_calostage2Digis_Jet_RECO. 1118.41 109.764
HcalNoiseSummary_hcalnoise__RECO. 470.704 108.427
floatedmValueMap_offlineSlimmedPrimaryVertices__PAT. 162.143 105.752
edmTriggerResults_TriggerResults__HLT. 1328.79 78.4765
l1tEGammaBXVector_calostage2Digis_EGamma_RECO. 648.3 73.6909
l1tEtSumBXVector_calostage2Digis_EtSum_RECO. 1369.87 56.7469
recoVertexCompositePtrCandidates_slimmedLambdaVertices__PAT. 138.059 51.5755
recoGenMETs_genMetTrue__SIM. 209.37 45.2902
recoGenMETs_genMetTrue__HLT. 209.37 45.2902
EventAuxiliary 133.123 44.5326
GlobalAlgBlkBXVector_gtStage2Digis__RECO. 1634.97 40.11
patPackedCandidates_lostTracks_eleTracks_PAT. 79.0923 24.4766
l1tMuonBXVector_gmtStage2Digis_Muon_RECO. 142.752 24.2695
recoConversions_reducedEgamma_reducedSingleLegConversions_PAT. 128.453 20.1273
recoGsfElectronCores_reducedEgamma_reducedGedGsfElectronCores_PAT. 222.428 16.5518
CSCDetIdCSCSegmentsOwnedRangeMap_slimmedMuons__PAT. 98.2064 15.8435
recoPhotonCores_reducedEgamma_reducedGedPhotonCores_PAT. 247.324 12.3706
DTChamberIdDTRecSegment4DsOwnedRangeMap_slimmedMuons__PAT. 119.437 12.1238
```

```

floatROOTMathCartesian3DROOTMathDefaultCoordinateSystemTagROOTMathPositionVector3D_genParticles_x
floatROOTMathCartesian3DROOTMathDefaultCoordinateSystemTagROOTMathPositionVector3D_genParticles_x
double_fixedGridRhoFastjetAll__RECO. 9.31653 7.76998
double_fixedGridRhoFastjetAllCalo__RECO. 9.32622 7.76543
double_fixedGridRhoFastjetCentral__RECO. 9.32622 7.75789
double_fixedGridRhoFastjetCentralCalo__RECO. 9.33592 7.75346
double_fixedGridRhoFastjetCentralNeutral__RECO. 9.3432 7.40716
double_fixedGridRhoFastjetCentralChargedPileUp__RECO. 9.35774 7.33802
recoDxDxHitInfosedmAssociation_isolatedTracks__PAT. 93.1052 5.35005
recoCSCHaloData_CSCHaloData__RECO. 105.724 5.14488
patPackedTriggerPrescales_patTrigger_llmax_PAT. 2536.34 4.9517
patPackedTriggerPrescales_patTrigger_llmin_PAT. 2536.34 4.9517
patPackedTriggerPrescales_patTrigger__PAT. 2536.29 4.902
float_genParticles_t0_HLT. 5.29222 4.2735
float_genParticles_t0_SIM. 5.29222 4.2735
double_fixedGridRhoAll__RECO. 9.29955 4.21775
GlobalExtBlkBXVector_gtStage2Digis__RECO. 1371.85 3.86547
recoBeamHaloSummary_BeamHaloSummary__RECO. 108.08 3.77809
EventSelections 90.0872 3.51603
recoBeamSpot_offlineBeamSpot__RECO. 335.433 2.95536
edmTriggerResults_TriggerResults__PAT. 126.375 2.87724
BranchListIndexes 26.261 2.7459
EventProductProvenance 16.0251 2.50902
edmTriggerResults_TriggerResults__RECO. 82.3585 2.12879
edmTriggerResults_TriggerResults__SIM. 80.3559 2.1081
patPhotons_slimmedOOTPhotons__PAT. 18.6087 2.0795
L1GlobalTriggerReadoutRecord_gtDigis__RECO. 73.3695 2.02498
recoCaloClusters_reducedEgamma_reducedOOTEBEClusters_PAT. 17.7531 1.73216
recoSuperClusters_reducedEgamma_reducedOOTSuperClusters_PAT. 17.6565 1.59989
recoCaloClusters_reducedEgamma_reducedOOTESClusters_PAT. 17.3822 1.53451
CTPPSLocalTrackLites_ctppsLocalTrackLiteProducer__RECO. 17.3798 1.53216
recoPhotonCores_reducedEgamma_reducedOOTPhotonCores_PAT. 17.5124 1.4862
LumiScalers_scalersRawToDigi__RECO. 17.3325 1.48476
Strings_slimmedPatTrigger_filterLabels_PAT. 15.343 1.46769
uint_bunchSpacingProducer__PAT. 5.30476 0.504325

```

Reading miniAOD from ROOT

```
Events->Scan("GenEventInfoProduct_generator__SIM.obj.weights_")
```

Make and Fill a TTree

```

edm::Service<TFileService> file;
TTree *tree;
tree=file->make<TTree>("tree", "tree");
int variableNameInTheCode;
double arrayNameInTheCode[100];
tree->Branch("variableNameInTheTree", &variableNameInTheCode, "variableNameInTheTree/I");
tree->Branch("arrayNameInTheTree", &arrayNameInTheCode, "arrayNameInTheTree/D");
tree->Fill();

```

Heppy

If you need to learn python you can look at this [link](#)

Getting informations after 7_4 (Handle)

Informations Hide python code

```
self.handles['name_you_choose'] = AutoHandle( 'packedPFCandidates', 'std::vector<pat::PackedCandi
self.handles['name_you_choose'] = AutoHandle( 'prunedGenParticles', 'std::vector<reco::GenParticl
self.handles['name_you_choose'] = AutoHandle( 'slimmedSecondaryVertices', 'std::vector<reco::Vert
self.handles['name_you_choose'] = AutoHandle( 'offlineSlimmedPrimaryVertices', 'std::vector<reco:
```

Example

example [↗](#)

python code Hide python code

example to put

Batch System Pisa

Command	what it does
<code>bsub -Is -R "select[defined(fai)]" -q fai /bin/bash -l</code>	connect to the fai server (you have to be in gridui)
<code>bsub -q local executable_file</code>	execute (there must be the environment setup)
<code>bjobs</code>	show the jobs executed with bsub
<code>bqueues</code>	show all the jobs running and pending in all the queues
<code>bkill job_number</code>	kill a bjob

Setup the environment:

- `source /afs/pi.infn.it/grid_exp_sw/cms/scripts/setcms.sh`
- `source /afs/pi.infn.it/grid_exp_sw/cms/scripts/setcms.csh`

Tip in order

connect to the server:

```
ssh -XY mandorli@gridui1.pi.infn.it
```

access the FAI facility:

```
bsub -Is -R "select[defined(fai)]" -q fai /bin/bash -l
```

call the directory you want:

```
/gpfs/ddn/cms/user/mandorli/Hmumu/CMSSW_8_0_25/src/skimming
```

setup the environment with one of:

```
source /afs/pi.infn.it/grid_exp_sw/cms/scripts/setcms.sh
source /afs/pi.infn.it/grid_exp_sw/cms/scripts/setcms.csh
```

send the job:

```
bsub -q local `pwd`/executable_file
```

Copying the certificate

```
cp /tmp/x509up_u9842 .
```

In order to see if there is any not saved file, you can use:

```
for i in `seq 0 1 165` ; do ls SingleMuon/skimmed_tree_${i}.root | grep file; done
```

What there should be in your executable file

```
source /afs/pi.infn.it/grid_exp_sw/cms/scripts/setcms.sh

export VO_CMS_SW_DIR=/cvmfs/cms.cern.ch
export SCRAM_ARCH=slc6_amd64_gcc530
source $VO_CMS_SW_DIR/cmsset_default.sh
#export CMSSW_GIT_REFERENCE=/cvmfs/cms.cern.ch/cmssw.git #if you need git

eval `scram runtime -sh` #cmsenv real name
```

CRAB

compile and execute

You can find instruction about how to get GRID certificate here

Command	what it does
source /cvmfs/cms.cern.ch/crab3/crab.csh	like cmsenv
voms-proxy-init --voms cms	create a proxy that last 24 hours
voms-proxy-init --voms cms --valid 168:00	create a proxy that last 168 hours
voms-proxy-info	provide some information about your proxy
crab submit -c file.py	launch the job, like cmsRun
crab kill directoryName	kill the job you launched

Instead of using 'crab submit' you can launch the job with: python file.py

You can find the output in /gpfs/ddn/srm/cms/store/user/gimandor/

Check username

To check your username (to check if everything is fine)

```
crab checkusername
```

Job's status

```
crab status name_directory
```

Configuration file example

[crab config file](#) [Hide file](#)

```
from CRABClient.UserUtilities import config, getUsernameFromSiteDB
config = config()
```

```
D = dict()
```

Tip in order

```
D['amcatnloFXFX'] = '/DYJetsToLL_M-105To160_TuneCUETP8M1_13TeV-amcatnloFXFX-pythia8/RunIISummer11
D['madgraphMLM'] = '/DYJetsToLL_M-105To160_TuneCUETP8M1_13TeV-madgraphMLM-pythia8/RunIISummer11
```

```
version='V01_'
config.General.requestName = 'TESTcrab_'+version
config.General.workArea = 'crab_projects'
config.General.transferOutputs = True
config.General.transferLogs = True

config.JobType.pluginName = 'Analysis'
config.JobType.psetName = 'cfg.py'

config.Data.inputDBS = 'global'
config.Data.splitting = 'FileBased'
config.Data.unitsPerJob = 20
config.Data.outLFNDirBase = '/store/user/gimandor/BBHeppy'+version+'/'
config.Data.publication = True
config.Data.outputDatasetTag = 'CRAB3_SingleMuon_check_kFactor'

config.section_("Site")
config.Site.storageSite = "T2_IT_Pisa"

if __name__ == '__main__':
    from CRABAPI.RawCommand import crabCommand

    for S in D.keys() :
        version_with_sample='V_18_'+S

        config.General.requestName = 'TESTcrab_'+version_with_sample
        config.Data.outLFNDirBase = '/store/user/gimandor/BBHeppy'+version_with_sample+'/'
        config.Data.inputDataset = D[S]

        crabCommand('submit', config = config)
```

DAS

DAS page is <https://cmsweb.cern.ch/das>

You can see sample version reference here

<https://twiki.cern.ch/twiki/bin/viewauth/CMS/PdmVAnalysisSummaryTable>

Examples

```
dataset= /QCD_Pt_*to*_TuneCUETP8M1_13TeV_pythia8/RunIIFall115MiniAODv2-PU25nsData2015v1_76X_mcRun2
dataset= /QCD_Pt_*to*_TuneCUETP8M1_13TeV_pythia8/RunIIFall115MiniAODv2-PU25nsData2015v1_76X_mcRun2
```

dasgoclient

```
dasgoclient -query="dataset status=* dataset=/DY*105*/*18NanoAODv6*/NANOAOBSIM"
dasgoclient -query="dataset=/DYJetsToLL_M-105To160_VBFFilter_TuneCUETP8M1_13TeV-amcatnloFXFX-pythia8"
```

GIT

I put this [?](#) page because it treats you like a child

Command	what it does
<code>git cms-addpkg SubSystemName/Package</code>	download that package from CMSSW
<code>git status</code>	Check the status of your repository
<code>git add name.py</code> <code>git commit -m "..."</code>	add to the files the one named name.py
<code>git commit -a -m "describe what you have done"</code>	write in the history that comment
<code>git merge</code>	merge your changes back to the master branch
<code>git push my-cmssw from-CMSSW_7_6_3:bbheppy</code>	save in gitHub your bbheppy's changes
<code>git push my-cmssw from-CMSSW_8_0_25:CovarianceTrack</code>	save in gitHub your covMatrix changes
<code>git pull ...</code>	load from gitHub the most up-to-date version
<code>git branch</code>	shows the branches
<code>git log</code>	
<code>git push my-cmssw</code> <code>name_local_branch:name_you_want_on_github</code>	push the cone in a branch of cmssw
<code>git rm --cached -r NameSubSystem/Package/test/file_name</code>	rm NameSubSystem/Package/test/file_name from the branch (you still have to push it)
<code>git diff CMSSW_10_2_0 --</code> <code>GeneratorInterface/GenFilters/src/VBFGenJetFilter.cc</code>	shows the defference of that file with that release
<code>git checkout CMSSW_10_2_0 --</code> <code>GeneratorInterface/GenFilters/src/VBFGenJetFilter.cc</code>	

Add Package

Add Package

```
git cms-addpkg DataFormats/TrackReco
git cms-addpkg HeavyFlavorAnalysis/RecoDecay
git cms-addpkg TrackingTools/TransientTrack
git cms-addpkg DataFormats/TrackReco
```

Not empty directory

```
mkdir old
mv * old
mv old ..
git cms-addpkg
mv ../old .
mv old/* .
```

Create a Repository

In GitHub create a new repository named "repositoryName", than tip in order:

```
git init
```

```
git status (optional)
git add fileYouWant.cc
git commit -m "description of the file"
git remote add origin https://github.com/gmandorl/repositoryName.git
git push -u origin master
```

Push an existing repository from the command line:

```
git remote add origin https://github.com/gmandorl/huu_main_A.git
git push -u origin master
```

Download a single folder from git repository

Andrea did the following commands

```
git fetch origin master:masteronline
git checkout masteronline -- prefire_maps #this command should be executed in the right folder
```

vi

A useful italian page about vi is available here [↗](#)

Writing and erasing

Command	what it does
i	insert the text on the left
a	add the text on the right
x	erase on the right like "canc"
dd	erase the whole line
exit	return to the normal status

Saving and quitting

Command	what it does
:q!	quit without saving
:w	save
Ctrl^ZZ	save and close

Helpful links

Links across the page:

- Shell Commands: http://guide.debianizzati.org/index.php/Guida_ai_comandi_da_terminale_-_Gestione_di_file_e_directory [↗](#)
- A macro for root: https://root.cern.ch/doc/master/cernstaff_8C.html [↗](#)
- python tutorial: <https://docs.python.org/2/tutorial/> [↗](#)
- vi tutorial: <http://www.science.unitn.it/~fiorella/guidelinux/ildp/guide/node80.htm> [↗](#)
- GRID certificate: <https://twiki.cern.ch/twiki/bin/view/CMSPublic/WorkBookStartingGrid#BasicGrid>
- genXSecAnalyzer tutorial: <https://twiki.cern.ch/twiki/bin/viewauth/CMS/HowToGenXSecAnalyzer>
- brilcalc how to start: <https://twiki.cern.ch/twiki/bin/view/CMS/BrilcalcQuickStart>

How to:

- Combine tutorial: <https://cms-hcomb.gitbooks.io/combine/content/>
- Workbook : <https://twiki.cern.ch/twiki/bin/view/CMSPublic/WorkBook>
- Skeleton code: <https://twiki.cern.ch/twiki/bin/view/CMSPublic/SWGuideSkeletonCodeGenerator>
- configuration files for cmsRun:
 - <https://twiki.cern.ch/twiki/bin/view/CMSPublic/SWGuidePoolInputSources>
- write a crab config file:
 - https://twiki.cern.ch/twiki/bin/view/CMSPublic/CRAB3ConfigurationFile#Predefined_CRAB_configuration_
- Check Lumi sections: <http://cms-service-lumi.web.cern.ch/cms-service-lumi/brilwsdoc.html#brilcalc>
- Example of tdrstyle.C: <https://ghm.web.cern.ch/ghm/plots/>

Check status:

- View the status of my crab jobs:
 - <https://monit-grafana.cern.ch/d/cmsTMGlobal/cms-tasks-monitoring-globalview?orgId=11&var-user=gimando>
- cmsanalysis:
 - http://farmsmon.pi.infn.it/ganglia-web/?c=User%20Interface&h=cmsanalysis.pi.infn.it&m=cpu_report&r=hou

Where to find:

- PPD string to search in all sample:
 - <https://twiki.cern.ch/twiki/bin/viewauth/CMS/PdmVAnalysisSummaryTable>
- MiniAOD reference: <https://twiki.cern.ch/twiki/bin/view/CMSPublic/WorkBookMiniAOD2015>
- DAS MiniAOD QCD here
- DAS nanoAOD with run and lumi selection here
- DAS DY_0J here
- What to write in the site above: \$ QCD
 - dataset=/QCD_Pt_300to470_TuneCUETP8M1_13TeV_pythia8/RunIIFall15MiniAODv2-PU25nsData2015v1
 - and run=1 and lumi=38036
- ZeroBias dataset=/ZeroBias*/Run2015D-16Dec2015-v1/MINIAOD
- CMS Data Aggregation System : <https://cmsweb.cern.ch/das>
- CMSSW Cross Reference : <http://cmslrx.fnal.gov/lxr/>
- mcm: <https://cms-pdmv.cern.ch/mcm/>
- MC samples Cross Sections:
 - <https://cms-gen-dev.cern.ch/xsdb/?columns=67108848¤tPage=0&pageSize=10&searchQuery=DAS%3>
- cuts for pile up jets:
 - https://twiki.cern.ch/twiki/bin/viewauth/CMS/JetID#Recommendations_for_13_TeV_data
- pdg ID: <https://twiki.cern.ch/twiki/bin/view/Main/PdgId>
- HEPPY code: <https://github.com/vhbb/cmssw/tree/vhbbHeppy80X/VHbbAnalysis/Heppy>
- C++ Classes:

pat::Jet Class Reference

https://cmssdt.cern.ch/SDT/doxygen/CMSSW_5_3_14/doc/html/d6/d00/classpat_1_1Jet.html

pat::PackedCandidate

https://cmssdt.cern.ch/SDT/doxygen/CMSSW_7_0_8/doc/html/d8/d79/classpat_1_1PackedCandidate.html

pat::PackedGenParticle

https://cmssdt.cern.ch/SDT/doxygen/CMSSW_7_4_5/doc/html/d0/dc9/classpat_1_1PackedGenParticle.html

reco::GenJet

https://cmssdt.cern.ch/SDT/doxygen/CMSSW_7_6_1/doc/html/d0/df9/classreco_1_1GenJet.html

reco::GenParticle

https://cmssdt.cern.ch/SDT/doxygen/CMSSW_5_3_9/doc/html/d5/dd4/classreco_1_1GenParticle.html

reco Namespace Reference

https://cmssdt.cern.ch/SDT/doxygen/CMSSW_7_5_0/doc/html/d1/d57/namespacereco.html

reco::VertexCompositePtrCandidate

https://cmsstdt.cern.ch/SDT/doxygen/CMSSW_7_5_0/doc/html/d3/dae/classreco_1_1VertexCompositePtrCandidate.html

reco::Vertex

https://cmsstdt.cern.ch/SDT/doxygen/CMSSW_4_4_2/doc/html/da/d95/classreco_1_1Vertex.html

reco::TemplatedSecondaryVertex< SV >

https://cmsstdt.cern.ch/SDT/doxygen/CMSSW_7_4_12_patch3/doc/html/d7/de8/classreco_1_1TemplatedSecondaryVertex.html

edm::HandleBase

https://cmsstdt.cern.ch/SDT/doxygen/CMSSW_7_0_8/doc/html/df/d10/classedm_1_1HandleBase.html#ae9431

edm::Ptr< T >

https://cmsstdt.cern.ch/SDT/doxygen/CMSSW_4_4_3/doc/html/dc/d1d/classedm_1_1Ptr.html

fastjet::PseudoJet

http://fastjet.fr/repo/doxygen-3.0.0/classfastjet_1_1PseudoJet.html

XYZTLorentzVector

https://root.cern.ch/root/html522/ROOT__Math__LorentzVector_ROOT__Math__PxPyPzE4D_double_.html

reco::Track

https://cmsstdt.cern.ch/SDT/doxygen/CMSSW_8_1_0_pre13/doc/html/dd/d5b/classreco_1_1Track.html

Example codes:

- Silvio's example code to create a TTree:
<https://github.com/silviodonato/cmssw/blob/11-ntuple/NtuplerL1/NtuplerL1/plugins/NtuplerL1Gen.cc>
- Silvio's example code:
<https://github.com/silviodonato/cmssw/blob/RemovePileUpDominatedEvents/RemovePileUpDominatedEvent>
- Silvio's example code to read trigger informations:
<https://github.com/vhbb/cmssw/blob/vhbbHeppy80X/PhysicsTools/Heppy/python/analyzers/core/TriggerObject>
- Cross Section:
<https://github.com/cms-steam/RateEstimate/blob/master/datasetCrossSections/datasetCrossSectionsSpring15.p>
- Andrea's Adaptive Vertex Finder:
https://github.com/cms-sw/cmssw/blob/CMSSW_7_4_X/RecoVertex/AdaptiveVertexFinder/plugins/VertexM
- Andrea's example of fastjet:
<https://github.com/vhbb/cmssw/blob/vhbbHeppy76X/VHbbAnalysis/Heppy/src/FastSoftActivity.cc>
- Example of declustering from fastjet:
http://fastjet.fr/repo/doxygen-3.1.3/12-boosted__higgs-old_8cc_source.html
- Andrea's BToD merger:
https://github.com/cms-sw/cmssw/blob/CMSSW_8_0_X/RecoBTag/SecondaryVertex/plugins/BtoCharmDeca
- Nuclear Interaction rejection code:
<https://github.com/dnowatsc/cmssw/blob/NI-rejection-730pre1/RecoBTag/SecondaryVertex/plugins/NuclearIn>

Unfolding pages:

- Unfolding Recommendations:
https://twiki.cern.ch/twiki/bin/view/CMS/TwikiSMP-GENRecommendations#Unfolding_How_to
- Recommendations on Unfolding: <https://twiki.cern.ch/twiki/bin/view/CMS/ScreeUnfolding>
- RooUnfold: <http://hepunix.rl.ac.uk/~adye/software/unfold/RooUnfold.html>

Other twiki pages:

- <https://twiki.cern.ch/twiki/bin/view/Sandbox/SilvioNotes>
- <https://twiki.cern.ch/twiki/bin/viewauth/CMS/SilvioRatesStep1>
- <https://twiki.cern.ch/twiki/bin/view/Sandbox/AgneseNotes>
- GiulioProgrammingLanguages:
<https://twiki.cern.ch/twiki/bin/view/Sandbox/GiulioProgrammingLanguages>
- VBF Hmumu page: <https://twiki.cern.ch/twiki/bin/viewauth/CMS/HiggsMuMuVBF>
- VBF Hmumu code: <https://github.com/arizzi/PisaHmumu>

- Hmumu sync page: <https://twiki.cern.ch/twiki/bin/viewauth/CMS/VBFHMuMuSync>

Others:

- Reference datase name: <https://twiki.cern.ch/twiki/bin/viewauth/CMS/PdmVAnalysisSummaryTable>
- tricky function for ROOT: <https://root.cern.ch/doc/master/classTTree.html> [↗](#)
- Trigger prescales 2015:
https://cmswbm.web.cern.ch/cmswbm/cmsdb/servlet/TriggerMode?KEY=l1_hlt_collisions2015/v339 [↗](#)
- git link to hmm nail: <https://github.com/arizzi/PisaHmm/tree/giulioDev> [↗](#)
- some andrea Plots : <https://www.pi.infn.it/~arizzi/tests/> [↗](#)

-- GiulioMandorli1 - 2015-11-17

This topic: Sandbox > GiulioNotes

Topic revision: r231 - 2020-01-14 - GiulioMandorli1



Copyright &© 2008-2020 by the contributing authors. All material on this collaboration platform is the property of the contributing authors.
Ideas, requests, problems regarding TWiki? Send feedback