Prepared by: Taimoor Khurshid & Muhammad Ahmad

Course material: From CMSDAS

Frame Work Lite

This is an interactive analysis tool integrated with CMSSW EDM (Event Data Model) Framework. It allows to automatically load the shared libraries defining CMSSW data formats and the tools provided, to easily access parts of the event in the EDM format within ROOT interactive sessions. It reads produced ROOT files, has full access to the class methods and there is no need to write full-blown framework modules. Thus having FWLite distribution locally on the desktop one can do CMS analysis outside the full CMSSW framework.

Exercise 7 - Remake the PATtuple in Exercise 6 and reduce its size by keeping only PAT Muon and Electron branches

In this exercise, we modify the config files used in Exercise 6. In order to reduce the size of the PATtuple we would only keep PAT Muons and PAT Electrons objects and drop the rest. The config files should now look like simple_PAT_MC_MuEle_cfg.py and simple_PAT_data_MuEle_cfg.py present in your home directory . Try to see the additional code lines in the config files compared to the corresponding ones in Exercise 6. To work with this modified config file and make the LITE PATtuple, execute the following steps:

Copy the script simple_PAT_MC_MuEle_cfg.py and simple_PAT_data_MuEle_cfg.py under CMSSW_5_3_13/src . Then execute the following commands:

```
cmsRun simple_PAT_MC_MuEle_cfg.py
```

This produces an output file called simple_PAT_MC_MuEle.root in your YOURWORKINGAREA/CMSSW_5_3_13/src area.

```
cmsRun simple_PAT_data_MuEle_cfg.py
```

This produces an output file called simple_PAT_data_MuEle.root in your YOURWORKINGAREA/CMSSW_5_3_13/src area.

On opening these two PATtuples one observes that only the selectedPatMuons and the selectedPatElectrons objects are retained.

Executing ls -altrh gives the size of these PATtuples.

To find the size of your PATtuple execute following Linux command:

```
ls -altrh simple_PAT_MC_MuEle.root
```

and

```
ls -altrh simple_PAT_data_MuEle.root
```

You may also try the following:

To know the size of each branch, use the edmEventSize utility as follows;

```
edmEventSize -v  simple_PAT_MC_MuEle.root
```

and

```
edmEventSize -v  simple_PAT_data_MuEle.root
```

To see what PAT objects there are, open the ROOT file as follows and browse to the PAT objects (for example patMuons_selectedPatMuons__PAT) as you did in Exercise 6:

Here is how you do it for the output file simple_PAT_MC_MuEle.root

```
root -l simple_PAT_MC_MuEle.root;
TBrowser b;
```

OR

```
root -l TFile f("simple_PAT_MC_MuEle.root") ;
TBrowser b;
```

To quit ROOT application, execute:

```
.q
```

QUESTION 7.1 - What is the size of the PATtuple simple_PAT_MC_MuEle.root and simple_PAT_data_MuEle.root?

QUESTION 7.2 - What is the mean eta of the muons for MC and data?

QUESTION 7.3 - Has the size of the output file compared to Exercise 6 reduced? Is the mean eta for muons for MC and data same as for the PATtuples created in Exrcise 6?

Exercise 8 - Use FWLite on the PATtuple created in Exercise 8 and make a Z Peak (applying `pt` and `eta` cuts)

FW Lite (pronounced "framework-light") is basically a ROOT session with CMS data format libraries loaded. CMS uses ROOT to persistify data objects. CMS data formats are thus "ROOT-aware"; that is, once the shared libraries containing the ROOT-friendly

description of CMS data formats are loaded into a ROOT session, these objects can be accessed and used directly from within ROOT like any other ROOT class!

In addition, CMS provides a couple of classes that greatly simplify the access to the collections of CMS data objects. Moreover, these classes (Event and Handle) have the same name as analogous ones in the Full Framework; this mnemonic trick helps in making the code to access CMS collections very similar between the FW Lite and the Full Framework.

In this exercise we will make a ZPeak using our data and MC sample. We will use the corresponding PAT tuples created in Exercsies 7. To read more about FWLite, have a look at Section 3.5 of Chapter 3 of Workbook(see link: https://twiki.cern.ch/twiki/bin/view/CMSPublic/WorkBook).

We will first make a ZPeak. We will loop over the reduced size selectePatMuons in the PATtuple and get the mass of oppositely charged muons. These are filled in a histogram that is written to an output ROOT file.

First make sure that you have the PAT tuples created in Exercise 7. They should be called simple_PAT_MC_MuEle.root and simple_PAT_data_MuEle.root.

1. First check these two packages from github.

Here are the links to get github setup:

Here are the instructions to do addpkg using Git:

```
git cms-addpkg PhysicsTools/FWLite
git cms-addpkg PhysicsTools/UtilAlgos
```

Recently some participants have reported errors with this exercise due to shutdown of cvs service. To fix this please add other packages also as required by FWLiteHistograms.cc

```
git cms-addpkg DataFormats/FWLite
git cms-addpkg DataFormats/Common
git cms-addpkg FWCore/FWLite
git cms-addpkg DataFormats/MuonReco
git cms-addpkg DataFormats/PatCandidates
```

Then to compile the packages, do

```
scram b
```

2. To make Z peak, we would be using the FWLite executable called FWLiteHistograms. The corresponding code should be in YOURWORKINGAREA/src/PhysicsTools/FWLite/bin/FWLiteHistograms.cc

With this executable we would be using the command line options. More about these can be learned from SWGuideCommandLineParsing (see link: https://twiki.cern.ch/twiki/bin/view/CMS/SWGuideCommandLineParsing)

To make ZPeak from this executable, using the MC PATTuple, run the following command:

```
FWLiteHistograms
inputFiles=/scratch/t/taimoor/LHC_SCHOOL_4/course_material/simple_PAT_M
C_MuEle.root outputFile=ZPeak_MC.root maxEvents=-1 outputEvery=10
```

You can see that you will get the following error:

```
terminate called after throwing an instance of 'cms::Exception'
  what():  An exception of category 'ProductNotFound' occurred.
Exception Message:
getByLabel: Found zero products matching all criteria
Looking for type: edm::Wrapper<std::vector<reco::Muon> >
Looking for module label: muons
Looking for productInstanceName:
The data is registered in the file but is not available for this event
```

This error occurs because your input files simple_PAT_MC_MuEle.root is a PATtuple and does not contain reco::Muon whose label is muons. It contains, however, selectedPatMuons (check yourself by opening the root file with ROOT browser). However, in the code FWLiteHistograms.cc there are lines that say:

```
using reco::Muon;
```

and

```
event.getByLabel(std::string("muons"), muons);
```

This means you need to switch from muons to selectedPatMuons

To implement this change, open the code YOURWORKINGAREA/src/PhysicsTools/FWLite/bin/FWLiteHistograms.cc. In this code, look at the line that says:

```
using reco::Muon;
```

and change it to

```
using pat::Muon;
```

and in this:

```
event.getByLabel(std::string("muons"), muons);
```

and change it to

```
event.getByLabel(std::string("selectedPatMuons"), muons);
```

To implement this change, you need to re-compile the code. To do this do:

```
rehash
scram b
cmsenv
```

Now again run the executable as follows:

```
FWLiteHistograms inputFiles=simple_PAT_MC_MuEle.root
outputFile=ZPeak_MC.root maxEvents=-1 outputEvery=10
```

You can see that now it runs successfully and you get a ROOT file with a histogram called ZPeak_MC.root. Open this ROOT file and see the Z mass peak histogram called mumuMass. Answer the following question.

QUESTION 8.1 - What is mean mass of the ZPeak for your MC PATtuple?

3. Now a little bit about the command that you executed.

In the command above, it is obvious that

simple_PAT_MC_MuEle.root is the input file ZPeak_MC.root is output file. maxEvents is the events you want to run over. You can change it any other number. The option -1 means running over all the events, which is 100 in this case. outputEvery means after how any events should the code report the number of event being processed. As you may have noticed, as you specified, when your executable runs, it says processing event: after every 10 events.

If you look at the code FWLiteHistograms.cc, it also contains the defaults corresponding to the above command line options. Answer the following question:

QUESTION 8.2 - What is the default name of the output file in the executable ?

Exercise 9 - Re-run the above executable with the data PATtuple

Re-run the above executbale with the data PATtuple file called simple_PAT_data_MuEle.root as follows:

```
FWLiteHistograms inputFiles=simple_PAT_data_MuEle.root
outputFile=ZPeak_data.root maxEvents=-1 outputEvery=100
```

This will create an output histogram ROOT file called ZPeak_data.root

Then answer the following question.

QUESTION 9 - What is mean mass of the ZPeak for your data PATtuple?

Exercise 10 - Fireworks - CMS Event Display

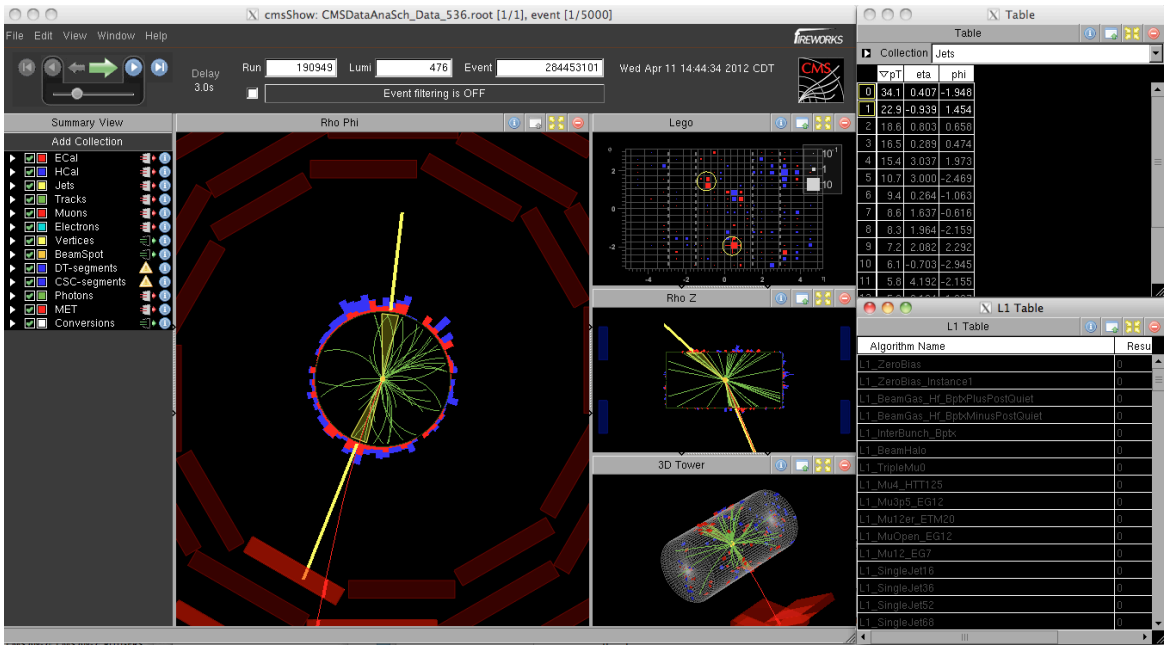
Fireworks is the CMS event-display project and cmsShow is the official name of the executable. Both names are used sometimes interchangeably. With this tool one can display events for physics. The core of Fireworks is built on top of the Event Data Model (EDM) and the light version of the software framework (FWLite). The Event Visualization Environment (EVE) of ROOT is used to manage 3D and 2D views, selection, and user-interaction with the graphics windows. Several EVE components were developed in collaboration between the Fireworks and ROOT teams. The event display operates using simple plugins, which are registered into the system to perform conversion from EDM collections into their visual representations. As a guiding principle, Fireworks shows only what is available in the EDM event-data, no reconstruction or result enhancement is performed internally. Visibility of collection elements can be filtered via a generic expression (PAT parser is used internally).

Please be aware that for any issues with fireworks display, first have a look at the twiki WorkBookFireworksHowToFix (see link: https://twiki.cern.ch/twiki/bin/view/CMS/WorkBookFireworksHowToFix. Also refer to the latest tutorial on fireworks is at http://indico.cern.ch/conferenceDisplay.py?confId=115595.

1. First we will look at the event display from YOURWORKINGAREA/CMSSW_5_3_13/src. Fireworks has been integrated into CMSSW since release CMSSW_3_1_0_pre6. After you login and do cmsenv, execute the following command from YOURWORKINGAREA/CMSSW_5_3_13/src: (We will look at the collision data and RelValZMM sample that you have used in earlier exercises.)

```
cmsShow /scratch/t/taimoor/LHC_SCHOOL_4/course_material/LHCSchool_Data_536.root
cmsShow /scratch/t/taimoor/LHC_SCHOOL_4/course_material/LHCSchool_RelValZMM536.root
```

It will pop a window like this:



This will open the Fireworks display window as shown in the snapshot above. This window has several parts that can be swapped or undocked for a separate view. Now do the following after the Fireworks windows open.

1. As you see the very first event displayed has an event number 284453101.

2. On the top left part that says "Summary View/Add Collection" uncheck all collections EXCEPT Muons. As you uncheck, notice how the different color-coded objects disappear from the main display sub-window that says "Rho Phi".

3. Now look at the small independent window on the top right that says Table on its title bar. In this window select Muons from the pull down menu. As you see, a row shows up there with details about the single muon object that you see (RED line) in the detector cut-out.

QUESTION 10.1 - What is pT of the only muon that you see in the first event?

In the "Summary View" panel on the left side of the main window, click on the little triangle button to the right of the "Tracks" row

QUESTION 10.2 - How many tracks does the first event have?

You can also open the RelValZMM.root file and display its events too.

Exercise 11 - Run Fireworks locally from Desktop

As you noticed, first accessing a remote file for cmsShow makes things run slowly. To overcome that you did the exercise as above. However, despite having the data and MC file in YOURWORKINGAREA/CMSSW_5_3_13/src, the display is still not fast enough as you are still probably logged into lxs or our local cluster remotely from your laptop. The display can be made the fastest possible if you have the fireworks executable and the data, MC ROOT files all locally on your laptop. In this exercise, we will first download fireworks locally and then run the display. We will also copy the ROOT files locally to the laptop/desktop.

1. First copy the ROOT files locally on your desktop. You can either copy files from the afs area OR copy the files with 100 events from YOURWORKINGAREA/CMSSW_5_3_13/src. We assume that you know how to do that. As an example, here is how you would copy the locally to a Macintosh.

From YOURWORKINGAREA/CMSSW_5_3_13/src,

```
scp
USERNAME@lxs.ncp.edu.pk:/scratch/t/taimoor/LHC_SCHOOL_4/course_material/LHCSc
hool_Data_536.root .
scp
USERNAME@lxs.ncp.edu.pk:/scratch/t/taimoor/LHC_SCHOOL_4/course_material/LHCSchoo
l_RelValZMM536.root .
```

Now we will get the fireworks executable locally. To do this, please have a look at the instructions in the twiki WorkBookFireworks (see link: https://twiki.cern.ch/twiki/bin/view/CMS/WorkBookFireworks) depending whether you have Linux/Windows/Macintosh (Note that, working with CMSSW_5_3_13, you should be able to see your data and MC files just fine).

For LINUX laptop

```
wget http://cern.ch/cmsshow/cmsShow-5.2-3.linux.tar.gz
```

For Macintosh laptop

```
curl -OL http://cern.ch/cmsshow/cmsShow-5.2-3.mac.tar.gz
```

Then do the following on LINUX

```
tar xzf cmsShow-5.2-3.linux.tar.gz cd cmsShow-5.2-3
```

OR the following on Macintosh

```
tar xzf cmsShow-5.2-3.mac.tar.gz cd cmsShow-5.2-3
```

Copy the LHCSchool_Data_536.root to directory cmsShow-5.2-3. To open the event display, execute the following:

```
./cmsShow LHCSchool_Data_536.root
```

QUESTION 11 - What is the size of the file called `cmsGeom10.root` in the directory cmsShow-5.2-3?