Table of Contents

Code setup

Initial setup

- Connect to pc2012 and obtain CERN kerberos ticket for SVN authentication:

```
$ ssh -XY pc2012.hep.manchester.ac.uk
$ kinit -f -r7d -A YOUR_CERN_USER_NAME@CERN.CH
```

- Commands to check out and compile code for the first time:

```
$ mkdir -p /pc2012-data3/${USER}/testarea/MPhysProject
$ cd /pc2012-data3/${USER}/testarea/MPhysProject

$ svn co svn+ssh://svn.cern.ch/reps/manc/PhysicsNtuple/PhysicsProject/trunk/macros/setup
$ source setup/setup_code_first_time.sh
$ ln -s PhysicsNtuple/PhysicsProject/macros/setup/setup_atlas_release_cvmfs.sh .
```

Setup on higgs.hep.manchester.ac.uk

- Replace "pc2012-data3" with "higgs-data4"
- Replace "pc2012 with "higgs"
- Use this data path when running code below: /higgs-data4/rustem/data/tth/v4_mc_data

Run analysis example

- Setup release:

```
$ cd /pc2012-data3/${USER}/testarea/MPhysProject
$ source setup_atlas_release_cvmfs.sh
```

- Run over 100k ttbar events to select simple analysis regions and make plots:

```
$ tag="410000"; python $TestArea/PhysicsNtuple/PhysicsProject/macros/runMPhys.py /pc2012-data
```

- old Ntuples:

```
$ tag="410000"; python $TestArea/PhysicsNtuple/PhysicsProject/macros/runMPhys.py /pc2012-data
```

- oldest Ntuples:

```
$ tag="410000"; python $TestArea/PhysicsNtuple/PhysicsProject/macros/runMPhys.py /pc2012-data
```

- To run on a different sample replace input file with a required process id:
  - ttbar: 410000.root
  - Z to ee: 361106.root
  - Z to mumu: 361107.root
  - Z to tautau: 361108.root
- List of all MC processes and cross-sections
- Brief description for some of available variables: docs/MPhys_variables.txt
- Print all available variables:

```
$ python $TestArea/PhysicsNtuple/PhysicsProject/macros/runMPhys.py /pc2012-data3/rustem/data/
```

- How to make plots
  - Double click on canvas to advance to advance to a next plot
  - Plotting command:

```
$ python macros/plotHists.py hist_410000.root --rkey=.*nocut -o plots -s -w
```

Analysis flow

- PhysicsProject - software package for this project

- Top level module is ReadNtuple class
  - Reads ROOT ntuple
  - Creates RecoEvent which contains raw reconstructed objects: muons, electrons, jets and hadronic tau candidates
  - Creates and executes children algorithms that are described next
  - It should not be necessary to modify this class

- PrepCand class creates candidate event
  - Select reconstructed objects using kinematic and particle identification information
  - Apply overlap removal between objects (eg electrons should not overlap with particle jets)
  - Create CandEvent to represent candidate event containing selected objects
  - It should not be necessary to modify this class

- PrepTopEvent class computes analysis variables
  - Code for new CandEvent variables should be added here

- PlotCand class plots variables stored with candidate event in a selected analysis region
  - Select candidate events with desired properties, for example: "2 leptons" OR "1 lepton and 1 hadronic tau"
  - Plot event variables defined in histogram files
  - It should not be necessary to modify this class

- Analysis job is configured in python module python/PhysicsProjectMPhys.py
  - Mapping between variables in ROOT ntuples and variables used by this analysis code
  - Define flow of algorithms that will be executed for every event
  - Configuration for object selection to create CandEvent
  - Configuration for event selection to plot CandEvent variables
  - Changes to object and event selection should go here

How to create and plot a new event variable

- Main analysis package is PhysicsProject - this is where all work should be done
  - Use "NJet" and "Mll" variables as an example.

- Add enum corresponding to the name of a new variable to these files:
  - PhysicsProject/CutVars.h
  - src/CutVars.cxx

- Compute value for new variables in PrepTopEvent class:
  - PrepTopEvent.cxx
  - New variables need to be added to CandEvent object

- Add histogram definition for candidate plots:
  - config/mphys/PlotCand.xml

Brief framework documentation

- Framework design: slides
- Python configuration for selections: PhysicsProjectMPhys.py
- Histogram definitions: histograms

How to update code after initial setup

- Check SVN status - make sure that there no conflicts (C):

```
$ source $TestArea/setup_atlas_release_cvmfs.sh status
```

- Update to server SVN:

```
$ source $TestArea/setup_atlas_release_cvmfs.sh update
```

- Recompile code (if no new files are added):

```
$ source $TestArea/setup_atlas_release_cvmfs.sh light-compile
```

- When new files are added, packages have to be rebuilt from scratch:

```
$ source $TestArea/setup_atlas_release_cvmfs.sh compile
```

How to download new ntuples from CERN

- Ntuples are stored on EOS mass storage at CERN
- Latest version is v4 with small changes with respect to previous version
- MC ntuples are organised by process id - described here
- First copy ntuples to tmp area on lxplus and then copy them to Manchester

```
$ ssh lxplus.cern.ch
$ eos ls root://eospublic.cern.ch//eos/escience/UniTexas/HSG8/multileptons_ntuple_run2/25ns_v
$ eos cp root://eospublic.cern.ch//eos/escience/UniTexas/HSG8/multileptons_ntuple_run2/25ns_v

$ scp lxplusNNNN.cern:/tmp/$USER/*root .
```

To-do items

Analysis at CERN

Introduction

- This section describes to how analysis jobs at CERN using LXBATCH batch system
- This uses python helper scripts which need to be checked out independently
- Batch jobs run this macro to make plots:
    - PhysicsProject/macros/run2Plot.py
- Python configuration for plotting jobs is defined here:
    - PhysicsProject/python/PhysicsProjectRun2Plot.py
- These jobs run over these input files:
    - run2/input_cern_tth_data_v04.txt
    - run2/input_cern_tth_siml_v04.txt
    - run2/input_cern_tth_siml_v04_priority2.txt
- Please make sure to have created this 100 GB work area using CERN IT website:

```
$ JOBPATH=/afs/cern.ch/work/${USER:0:1}/$USER/jobs
$ mkdir -p $JOBPATH
$ ls -al $JOBPATH
```

Batch commands

- LXBATCH is made of tens of thousand computers managed using Platform Load Sharing Facility (LSF) software
- Users can submit commands to these computers using special batch commands
- User priority is balanced by LSF software - more jobs you run, less priority you get
- These are simple batch commands

```
$ ssh -XY lxplus.cern.ch
$ bqueues -u $USER
$ bsub /bin/ls -al
$ bjobs
```

Initial code setup

- Setup usual PhysicsProject code on lxplus at this path: $HOME/testarea/MPhysProject
- Check out batch helper scripts and auxiliary data:

```
$ cd $TestArea/PhysicsNtuple/PhysicsProject
$ svn co svn+ssh://svn.cern.ch/reps/manc/PhysicsNtuple/PhysicsLight/trunk/macros/batch
$ svn co svn+ssh://svn.cern.ch/reps/manc/PhysicsNtuple/PhysicsLight/trunk/data/run2
$ svn co svn+ssh://svn.cern.ch/reps/atlasphys-hsg8/Physics/Higgs/HSG8/AnalysisCode/multilepto
$ rm XsectionInput_tth_run2/update_xsection.cxx
```

Submit analysis jobs

- Commands to submit analysis jobs ("--submit" actually submits jobs)

```
$ source macros/sub_batch_jobs.sh "" "-n 100000" "-first-plots"
$ source macros/sub_batch_jobs.sh "--submit" "-n 100000" "-first-plots"
$ bjobs
```

Merge data files

- Make sure that all jobs are completed: output of "bjobs" command should be empty
- Merge data files:

```
$ python batch/addFiles.py $JOBPATH/hists-2016-03-16-first-plots/plots-data_v04
```

Make plots

- Python command to make data and MC plots for "NJet" plots:

```
$ source macros/plot_stack.sh "$JOBPATH/hists-2016-03-16-first-plots" "--hkey=NJet.* --rkey=.*
```

Paper references

- ATLAS measurement of top quark branching ratios with 7 TeV data
- LEP WW Electroweak Measurements

ATLAS references

- Tau working group recommendations
- B-Tagging benchmarks

This topic: Main > PhysicsLightMPhysExample
Topic revision: r16 - 2016-05-26 - RustemOspanov