NTupleAnalysis Package - Rome

... following the path of Civilazation from Athena(s) to Rome.

Motivation and Introduction

The NTupleAnalysis package is not intended to compete with the EventView packages or even the EWPA framework. I developed it to be able to give an easy start to physics analysis based on ROOT-NTuples, i.e. that you can run your whole physics analysis outside Athena. This is just the fastest way to do analysis, since you can run over small sized files on high speed, even on your local machine. In later periods of LHC, such a local running is not be possible anymore and you have to do your analysis inside Athena. Hence I created a simple interface that you can run your NTuple-based analysis also in usual Athena job, without changing one line of code.

The heart of the package is an Event-object, which represent all information of one event, which the user needs. This event-Object can now simply be filled via a NTuple-File or via AODs, DPDs or even ESDs. The user will not know anything from that.

Obviously it is from high importance to produce the NTuple-Files from AODs, which are needed to run the NTuple-Analysis framework.

Installation

I have tested the package in Athena 14.2.0. So setup an Athena 14.2.0 release and go to your Athena directory, which I assume it is called "Athena/14.2.0" from now on

```
cd Athena/14.2.0
```

Then check out the following packages

```
cmt co PhysicsAnalysis/AnalysisCommon/InsituPerformance/NTupleAnalysisExample
cmt co PhysicsAnalysis/AnalysisCommon/InsituPerformance/NTupleAnalysisPackage
cmt co PhysicsAnalysis/AnalysisCommon/InsituPerformance/NTupleDumpingTool
```

Now compile each of the packages in the following order

```
cd ~/Athena/14.2.0/PhysicsAnalysis/AnalysisCommon/InsituPerformance/NTupleAnalysisPackage/cmt
cmt config
source setup.sh
gmake

cd ~/Athena/14.2.0/PhysicsAnalysis/AnalysisCommon/InsituPerformance/NTupleDumpingTool/cmt
cmt config
source setup.sh
gmake

cd ~/Athena/14.2.0/PhysicsAnalysis/AnalysisCommon/InsituPerformance/NTupleAnalysisExample/cmt
cmt config
source setup.sh
gmake
```

and now I hope everything went fine...

Dumping NTuples

Having compiled all the packages, go to

```
cd ~/Athena/14.2.0/PhysicsAnalysis/AnalysisCommon/InsituPerformance/NTupleAnalysisExample/share
```

and open the file "DemoAODDumper.py" and adjust it to your need. Escpecially, you should change

```
ServiceMgr.EventSelector.InputCollections = ["/scratch/mschott/AODs/v14/T1_McAtNlo_Jimmy.recon.AO
```

to a version 14 compatible AOD file. To specify, which collections should be dumped, you just change the following lines according to your needs

```
aodDumper.TruthContainerNames      = [ "SpclMC" ]
aodDumper.TrackContainerNames      = [ "TrackParticleCandidate", "MuonboyTrackParticles" ]
aodDumper.ClusterContainerNames    = [ "EMTopoCluster" ]
aodDumper.MuonContainerNames       = [ "StacoMuonCollection" ]
aodDumper.ElectronContainerNames   = [ "ElectronAODCollection" ]
aodDumper.PhotonContainerNames     = [ "PhotonAODCollection" ]
aodDumper.TauContainerNames        = [ "TauRecContainer" ]
aodDumper.JetContainerNames        = [ "Cone4H1TowerJets" ]
aodDumper.TriggerNames             = [ "L1_MU20", "L2_mu20", "EF_mu20" ]
aodDumper.TriggerDecisionNames     = [ "L1_MU20", "L2_mu20", "EF_mu20" ]
aodDumper.MissingETContainerNames  = [ "MET_RefFinal", "MET_Truth" ]
```

So if you also want to dump "MooreTrackParticles" just change

```
aodDumper.TrackContainerNames      = [ "TrackParticleCandidate", "MuonboyTrackParticles","MooreTr
```

Keep in mind that I only dump a fraction of information of each collection. Usually it is pt, eta, phi and the charge. An overview is given at this link (to be done). If you miss something, you can either change the code (The relevant classes are NTupleRootInterface in the NTupleAnalysisPackage and theNTupleDumpingTool) or just write me a mail...Now you can start the dumping via

```
athena.py DemoAODDumper.py
```

Creating your first Rome-Analysis

The first analysis is already prepared in

```
cd ~/Athena/14.2.0/PhysicsAnalysis/AnalysisCommon/InsituPerformance/NTupleAnalysisExample/UserAna
```

open "UserAnalysis_Zmumu.cxx" and have a look what it does. Acutally, everything is explained in the code. You should just note that the collections names in the NTuple-Event are the same as you specified in the Dumping JobOption File for Dumping.

Run the Analysis inside Athena

To run the analysis inside Athena, just go back to

```
cd ../share
```

and change the "DemoAODAnalysis.py"-File as you changed the "DemoAODDumper.py" file before (i.e. changing the your Input-File). Then just start the analysis via

```
athena.py DemoAODAnalysis.py
```

Run the Analysis outside Athena

So far so easy. Now we want to run the same code, i.e. "UserAnalysis_Zmumu.cxx" outside athena. To that, we change to

```
cd ../rome
```

and type

```
make
```

This will compile your analysis and the corresponding libraries and create a program called rome. If you start rome via

```
./rome
```

you will see

```
-------------------------------------------
| Starting Rome v1.0                      |
|  ... following the path of Civilization |
-------------------------------------------
Error: Arguments missing!
  Usage:   1. Argument: Name of Job-Option file
           2. Argument: Optional; number of events to be processed (No argument = all events)
  Example: ./rome DemoJob.txt 100
-------------------------------------------
```

This just means that you have forgotten to pass over the JobOption-like file, i.e.

```
./rome DemoJob.txt
```

the file "DemoJob.txt" contains the input and output-file configuration. Moreover it loads the file "RootLoaderConfig.txt". In this file it is specified, which collections are read from the NTuple file.

A more complicated Example: Tag&Probe Efficiency determination

To be done

-- MatthiasSchott - 28 May 2008

This topic: Main > NTupleAnalysis
Topic revision: r2 - 2008-08-08 - MatthiasSchott