Introduction to TAGs

ATLAS DESY NAF and FDR
Tutorial, May 13, 2008

OUTLINE:

- TAG Content
- Using TAG in Analysis

N. Vlasov

universitätbonn

Types of TAGs

There are two types of TAG available for analysis:

- *file based TAG*, which is created as a strict subset of AOD (one file is created for one AOD, ROOT format). TAG-files to be produced with AOD at T0 and replicated to each site (?) containing AOD copy (T2 and lower)

- *relational database* (MySQL, Oracle)

The contents of two are same while the way they point to AOD is different. In relational database, one event has one entry only while in file based TAG, one event can be duplicated if it exists in two streams

ROOT TAG file is producible from AOD, though TAG databases contain or are linked to sufficient navigational information to allow retrieval of event data at all production processing stages, i.e., AOD, ESD, Raw Data

Appling a first-level-cuts with TAG one can reduce strongly sample size. Actual scope:

- **Merging of streams removing non-unique events**
- **Removal of events in bad lumi blocks or those flagged as bad quality**
- **Skimming of events using object information in TAG such as missing Et**

TAG Content

- TAG data will be written during AOD production; 1k pro event for TAG (TAG size = 1% AOD size)
- TAG content sections:
  - **Collection information** and **global event quantities** reference to find event data (to AOD, ESD and RDO); Event Info record, number of tracks and vertexes, missing energy
  - **Trigger decisions -** trigger levels 1,2; Event Filter. Note: default trigger configuration is not appropriate for FDR
  - **Physics quantities** electrons, photons, muons, jets and τ-jets
  - Information specific to **physics and combined performance** physics group "yes-no" analysis decisions
  - **Data Quality** information is currently a straightforward copy of detector status flags. TAG contains an element StatusXYZ which takes value 0, 1, 2 or 3, corresponding to detector status UNKNOWN, RED, YELLOW or GREEN
- See : https://twiki.cern.ch/twiki/bin/view/Atlas/TagForEventSelection

TAG Content details (Electrons, Photons)

- `isEM` bits to select all (*loose*) electrons and photons with P_T > 7 GeV. No separation criteria for both

Attribute Definition	Attribute Name	Attribute Type
Total number of loose electrons	`NLooseElectron`	unsigned int
Loose electron P_T	`LooseElectronPt`	float
Loose electron η	`LooseElectronEta`	float
Loose electron φ	`LooseElectronPhi`	float
Loose electron Tightness	`LooseElectronTightness`	unsigned int

- The objects are ordered in falling P_T and signed: for instance `LoosElectronPt1` is negative it means it is an electron, positive - positron

Object	Number
Electrons	4
Photons	2
Muons	4
Taus	2
Jets	6

TAG Content details (Muons, Missing E_T)

- Loose muon – a stand-alone muon candidate from muon spectrometer

- By the default it is the `Muonboy` (container `MuonboyMuonSpectroOnlyTrackParticles` ?) AOD that is used to produce the Muon TAG but that may change depending on performance

Total number of loose muons	`NLooseMuon`	unsigned int
……..	`..`	…
Loose muon Tightness	`LooseMuonTightness`	unsigned int
Loose muon Isolation E_T	`LooseMuonIsolationEt`	float
Loose muon Track Isolation	`LooseMuonIsolationN`	unsigned int

- Missing E_T information from the container `MET_Final`

Missing Energy	`MET`	float
φ of Missing E_T	`METPhi`	float
Summed cell E_T	`SumET`	float

TAG Content details (Jets, τ-Jets)

- Defaults are to take `ParticleJet` information from the container `Cone4H1TowerParticleJets` and `TauJet` from `TauRecContainer` (?) in the AOD

Total number Jets	`NJet`	unsigned int
Total number b-tagged Jets	`NBJet`	unsigned int
Jet E_T	`JetET`	float
Jet η	`JetEta`	float
Jet φ	`JetPhi`	float
B-tag likelihood	`JetBLkh`	float
Summed E_T over Jets	`JetSumET`	float

Total number of tau jets	`NTauJet`	unsigned int
Tau Jet Pt	`TauJetPt`	float
Tau Jet η	`TauJetEta`	float
Tau Jet φ	`TauJetPhi`	float
Tau Jet number of tracks	`TauJetNTrk`	unsigned int
Tau Jet likelihood	`TauJetLikelhoodRatio`	float

Using TAG in Analysis

- To make a ROOT TAG query in Athena job:

```
EventSelector.InputCollections = [ “Tag.pool” ]
EventSelector.Query=“NLooseElectron>0 && NLooseElectron<3
&& abs(LooseElectronEta1)<2.5 && abs(LooseElectronEta2)<2.5
&& LooseElectronPt1>10000 && LooseElectronPt2>10000”
EventSelector.CollectionType = “ExplicitROOT”
```

- `PoolFileCatalog.xml` is needed to navigate from TAG to AOD (generated with command : `pool_insertFileToCatalog <file list> )`

- Job options for producing TAGs from AOD or merging AOD and building TAGs exist in `RecExCommon` :

```
aodtotag.py
aodtoaodtag.py
```

Using TAG in Analysis (2)

- One can make distributed analyses on relational data base with TAG queries. One not need to have AOD files locally
- **ELSSI** is a web skimming interface which allows to create AOD with only events passed TAG selection. Details are in the next talk