

Athenaの理解の現状

加茂直之

Kyoto ATLAS mtg
07/27/2012

今週やったこと

- (AthExHelloWorldをてきとうに書き換える)
- voの登録をする
- Athenaソフトウェアチェーンをひと通りやる
- diphoton.cxxが何してるか理解する
- LSFを使って、ジョブを並行して走らせてみる

Athenaソフトウェアチェーン

ここでは

Generation→Simulation→Digitization→Reconstruction→Analysis

という一連の処理をひと通りやる。

今回は $H \rightarrow \gamma \gamma$ で出てきた 2γ から H の不変質量を求めている。

Athenaソフトウェアチェーン

疑問点(大したことはないですが)

- \$.pyって\$ athena .py というのと同じ?
- CheckFile.pyというのは他の.rootでも一般的に使える?
- できた.rootの構造?(そもそもROOTが使えないのでよくわからない)
- X windowは出せる?

diphoton.cxx

C++もROOTもわかってないので、何を言ってるかよくわからなかった……。

→丁度、しょーたさんが夜中に帰ってきたので、ROOTのTTreeについて教えてもらいました!

(不変質量を求める部分はcだけの知識でわかりますが)

diphoton.cxx

```
for(int ph_i=0; ph_i<ph_n-1; ph_i++) {  
    // 4-momentum of the i-th photon  
    TLorentzVector p4_ph_i;  
    p4_ph_i.SetPtEtaPhiM(ph_Et->at(ph_i), ph_eta->at(ph_i), ph_phi->at(ph_i), 0.);  
    for(int ph_j=ph_i+1; ph_j<ph_n; ph_j++) {  
        // 4-momentum of the j-th photon  
        TLorentzVector p4_ph_j;  
        p4_ph_j.SetPtEtaPhiM(ph_Et->at(ph_j), ph_eta->at(ph_j), ph_phi->at(ph_j), 0.);
```

(ここまでですべての γ から2つ選んで、ローレンツベクトルを作る)

```
        // 4-momentum of di-photon  
        TLorentzVector p4_diphoton = p4_ph_i+p4_ph_j;  
(それぞれのローレンツベクトルをたす)  
        std::cout << "The invariant mass of diphoton ("  
            << ph_i << ", " << ph_j << ") is "  
            << p4_diphoton.M()/1000. << " GeV " << std::endl;  
        histo_Mass->Fill(p4_diphoton.M()/1000.);  
    }
```

(不変質量を計算する)

LSF

計算機に大量にジョブを渡したとき、適当に整理して、空いている計算機に効率良くジョブを分配してくれる機能

使い方: `bsub` (シェルスクリプト)

Athenaソフトウェアチェーンの最初の方の指示を
スクリプトに書いてやらせました。
そんだけしかしてないですが。

今後の予定

- AOD解析？
- RegularComputingTutorial？
- ROOTの使い方を少し勉強したほうがいい？