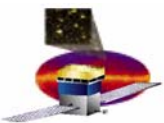


Introduction of the SVAC ntuple

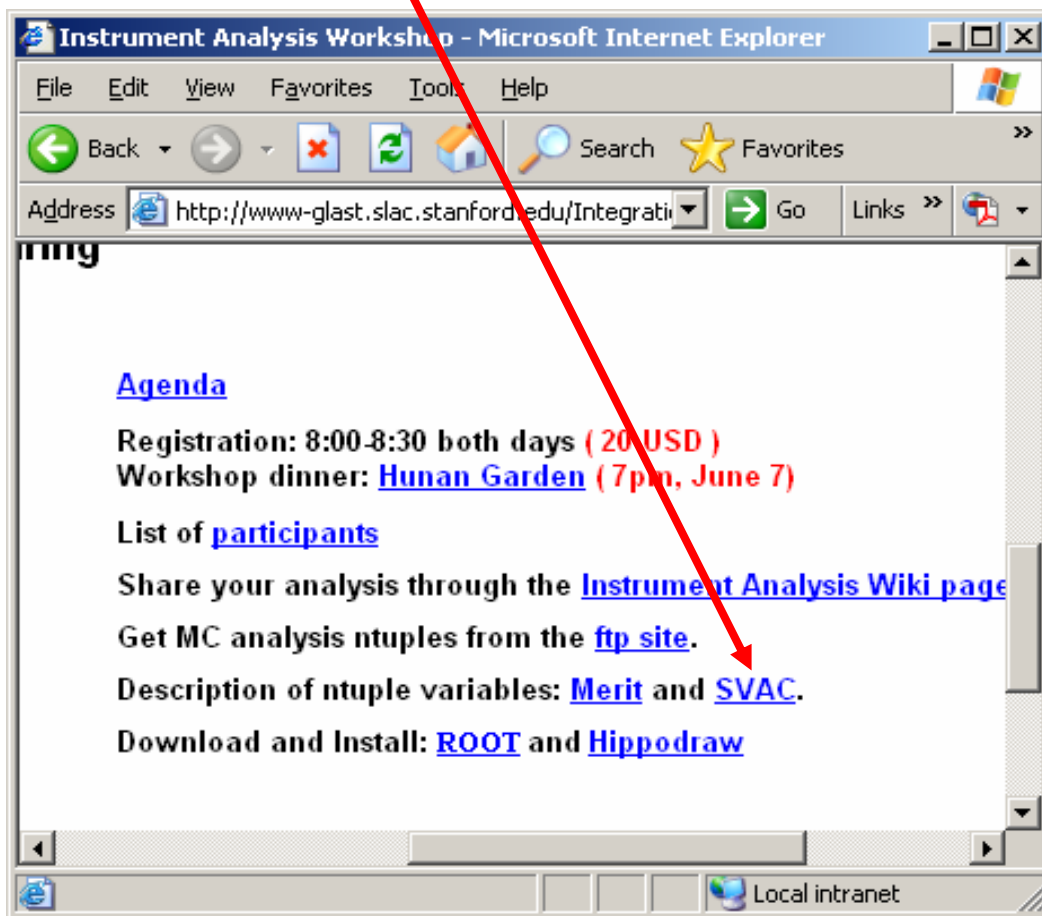
Xin Chen



Content of the ntuple

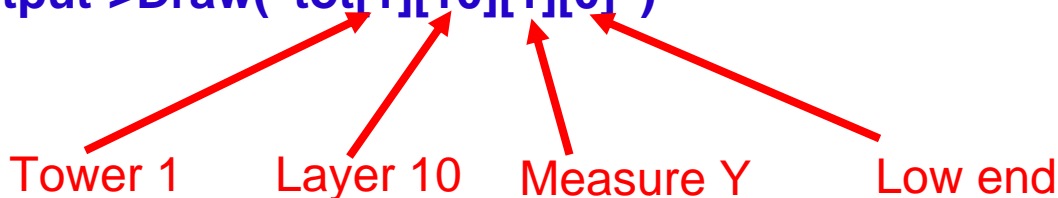
Detailed description of variables in the SVAC ntuple can be found at:

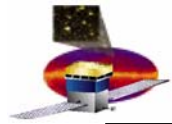
- <http://www.slac.stanford.edu/exp/glast/ground/software/SVAC/EngineeringModelRoot/main.html>



Rationales of creating another data file (1)

- Convenient for users with less experience in ROOT tree analyses since you can directly plot the ntuple variable.
- Compared with the merit ntuple, it not only contains reconstruction results but also some more basic and detailed info in the digi data.
 - For example, the **tot** variable is a 4 dimensional array storing TOT values for every tower, layer, view and end.
 - In ROOT,
 - Output->Draw(“tot[1][10][1][0]”)



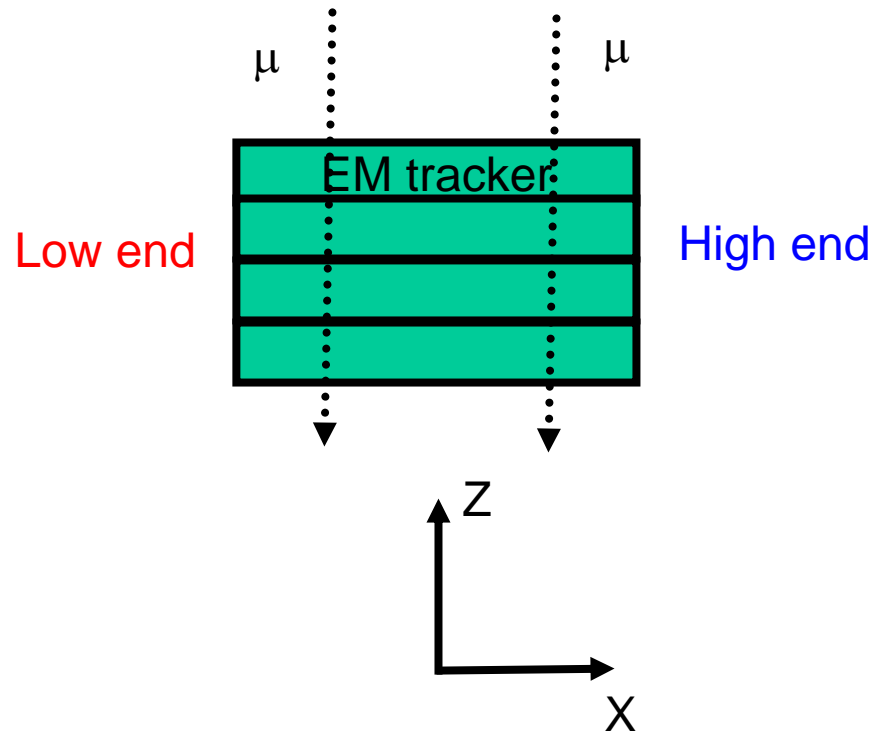


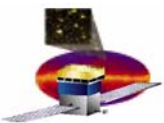
Rationales of creating another data file (2)

- **It contains variables not implemented in GlastRelease yet.**
 - For example, the **totCorrQ** variable is a 4 dimensional array. It stores corrected TOT values using quadratic formula for each strip
 - Used in EM1 data analyses. May go away when TOT calibration is implemented in GlastRelease.
- **It contains variables describing info from DAQ electronics and transform them from electronics space to detector space.**
 - For example, TkrTp contains trigger primitive data.
 - Would be useful to diagnose possible problems in the electronics

A example of EM1 data analysis

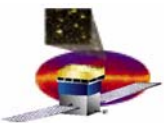
Trigger request can be sent from both end of Si plane, we need to verify trigger efficiency are the same nor matter which end sent the trigger request. **These info exist only in real data and are not simulated.**





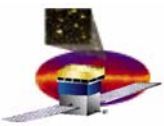
The TkrReq variable

- In the SVAC ntuple, **TkrReq[tower][layer][view][end]** is a **4 dimensional array variable**:
 - **TkrReq[0][2][0][0] = 1** means cable controller at tower 0, layer 2, measure X and low end sent the trigger request.
 - **TkrReq[0][2][0][0] = 0** means cable controller at tower 0, layer 2, measure X and low end did NOT send the trigger request.

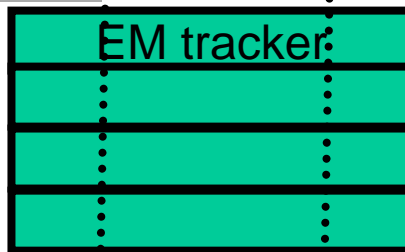
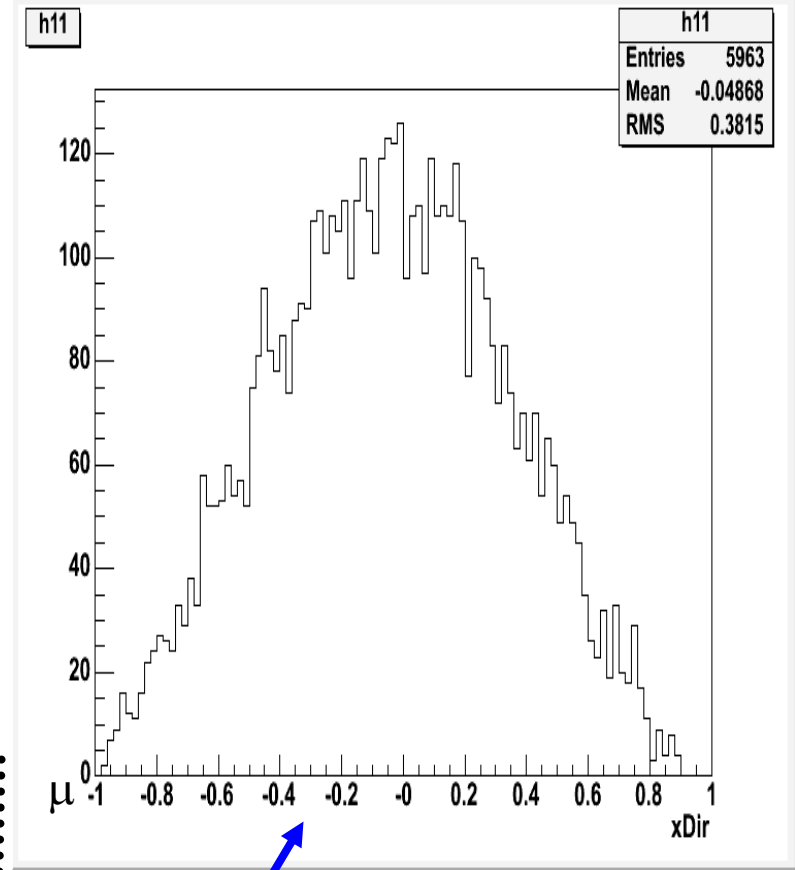
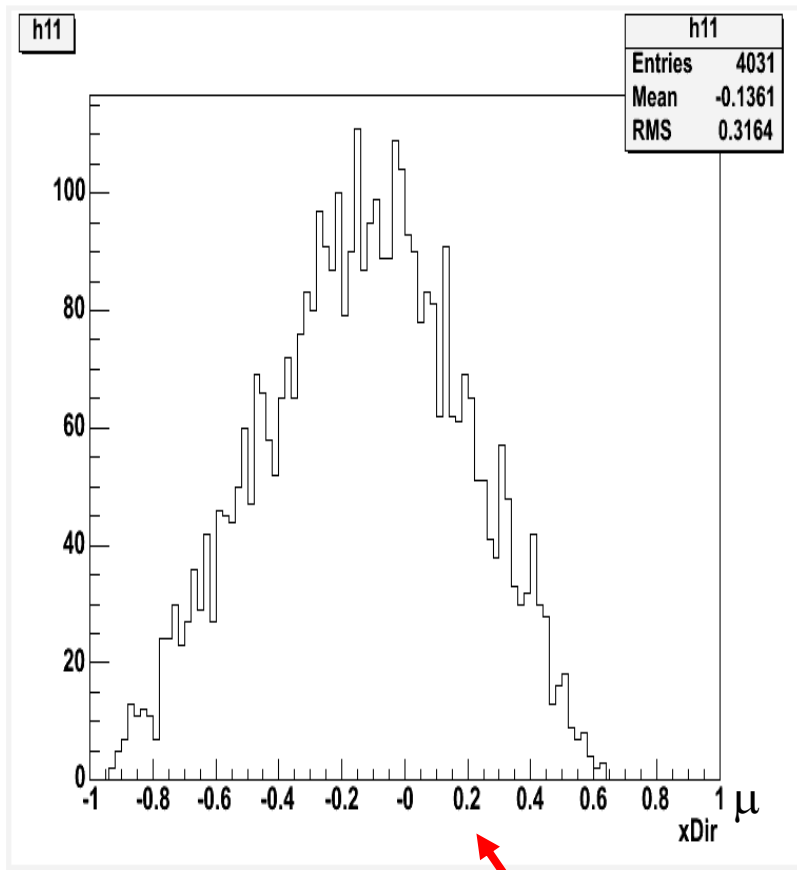


A root script example

- In the ROOT script:
 - Selects events with TKR trigger requests sent only from low end of Si plane
 - `TCut cut1("TkrReq[0][1][0][0]==1 && TkrReq[0][2][0][0]==1 && TkrReq[0][3][0][0]==1 && TkrReq[0][1][0][1]==0 && TkrReq[0][2][0][1]==0 && TkrReq[0][3][0][1]==0");`
 - Selects reconstructed events
 - `TCut cut3("TkrNumTracks>0");`
 - Draw distribution of reconstructed event direction along the x axis with cuts
 - `TH1F h11("h11", "h11", 100, -1, 1);`
 - `h11.SetXTitle("xDir");`
 - `Output->Project("h11", "VtxXDir", cut1&&cut3);`
 - `h11.Draw();`



Reconstructed event direction with different cuts



Low end

High end