Status Of OnboardFilter

Richard E. Hughes
Brian L. Winer
The Ohio State University
Overview

- Two ground software packages will be changed soon.
  - EbfWriter Package
  - OnBoardFilter Package
- The changes must be performed together.
EbfWriter Changes

- **Minor Changes**
  - Bring event contributions from LAT components into compliance with the hardware design
    - GEM Contribution Words
    - LATp Header words
  - Model Contribution (Tkr) Truncation
    - i.e. too many hits on a tracker layer end/tower.
  - ACD Veto Mapping Updated
  - Bug fixes.
EbfWriter Changes

- **Major Change**
  - Model event “truncation” for large events
  - Event contributions larger than ~4K get separated into pieces.
  - The new EbfWriter models this aspect of the hardware.
  - The Ebf format of the data is placed on the TDS in this fashion and the new OnBoardFilter properly receives the data.

- **Major Addition**
  - Testing the DAQ and Trigger System with the Testbed.
  - Requires files that represent the output of the Front-end Elec.
  - The new EbfWriter produces these files. (16 CAL, 16 TKR, 4 ACD)
DAQ/Trigger Testbed

Samples:
- Single Particle
- AllGamma Sample
- Background Sample
- Data Challenge 1

Integrity Testing
- Rate Testing
- Filter Testing

GLEAM/EbWriter

CAL/TKR/ACD Files

VxWorks Nodes

DAQ/Trigger Testbed

TEMs

GASU

OnBoardFilter

FrontEnd Simulators (FES)

Hardware Event

Software Event

Compare Predicted vs Observed

Richard Hughes

DCII Workshop

27-Jun-05; p.5
OnBoardFilter

- **Current version**
  - Has a wrapper (OnboardFilter.cxx) plus modified version of JJ’s actual c-code source
  - JJ’s code has undergone significant revision since this was done (~1 year ago)

- **New version**
  - Depends on 2 CMX packages: EFC (Event Filter Code) and EDS (Event Data Store), which are actual flight software
    - Dependency is through header files and libraries.
    - Libraries are compiled via CMX
    - These are accessed via External packages (like IdfReader)
  - OnboardFilter.cxx wrapper is modified to handle new call structure of EFC/EDS
  - Same output as previous, placed on TDS: result vector, energies, Projections, TKR hits, etc
OnboardFilter Validation

- The primary method we have used for validation is to compare to JJ’s filter code itself:
  - Create EBF file from some sample (e.g. AllGamma) using EbfWriter
  - Run JJ’s code (driver “filter” in EFC package) on this sample, and create summary table #1
    - The summary table contains a breakdown of which events were rejected by the filter
  - Run OnboardFilter code on AllGamma digi’s
    - Use EbfWriter to write EBF onto TDS
    - Create summary table #2
  - Compare the summary tables
Comparison of Old vs New

- Ran on subset of allGamma sample
- Total allGamma events filtered:
  - 446/1000: Old OnboardFilter
  - 450/1000: New OnboardFilter
- Plot to the right shows the inclusive vetoes by bits
  - A single event can contribute multiple times to this plot
  - Blue hatched: New OnboardFilter
  - Red hatched: Old OnboardFilter
Next Steps

- Need to validate OnboardFilter TDS
  - Use FilterAlg to do this
  - Compare Filtering of current version to previous
    - AllGamma
    - Background
  - Still some minor differences in filter results between FilterAlg and OnboardFilter

- Need to decide on versioning
  - How to link changes in flight software packages to new versions of OnboardFilter?
  - Currently flight filter code is grouped together: EDS + EFC
  - Propose: separate into two packages, then use the flight version numbers for each package. Will this work? □