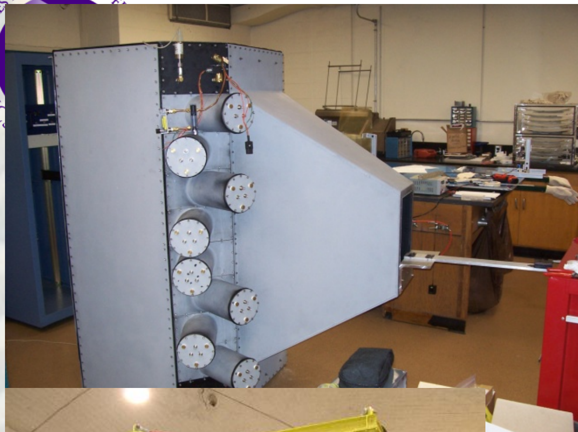
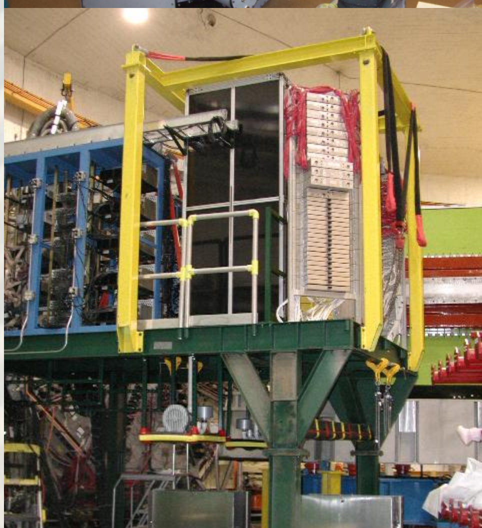
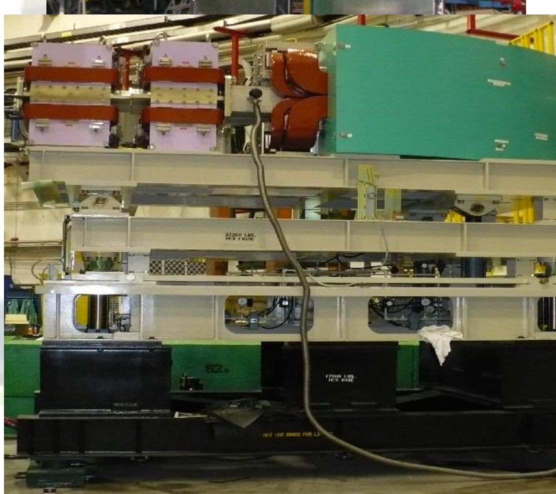


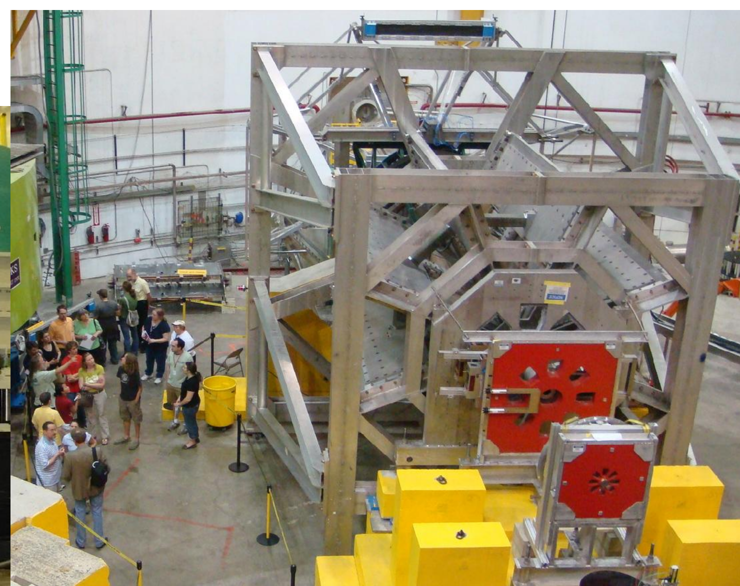
Hall C

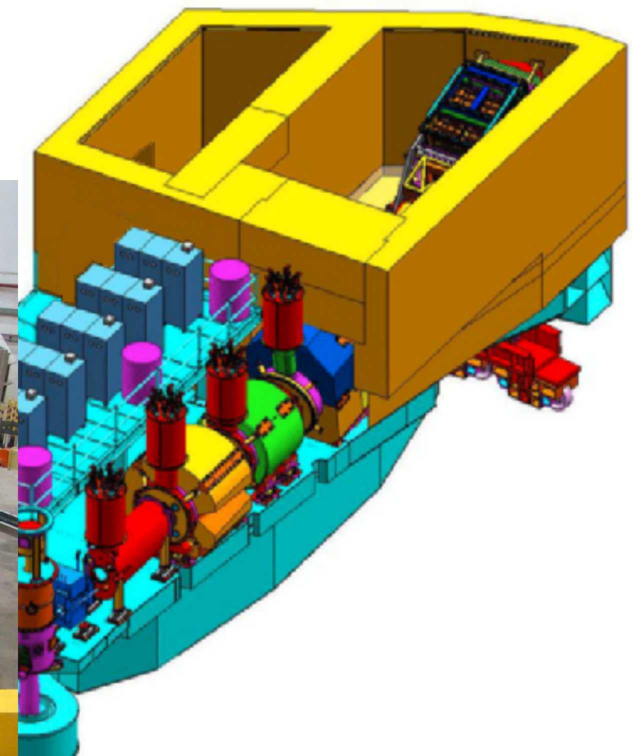
PM



Software Review







Outline

- ROOT/C++ analyzer
- Analysis algorithms (tracking, pid, etc.)
- Workflow
- Calibration
- Simulation
- Summary/Outlook

ROOT C++ Analyzer Status

- Built on top of (almost) unmodified Hall A PODD
- In publicly readable git repository
- Current features:
  - Reads Hall C style parameter files
  - Reads Hall C style hardware->detector mapping
- Builds ENGINE style raw hit lists
- Prints out hodoscope and drift chamber hit lists from HMS CODA files

(S)HMS Analysis Overview

- New C++/ROOT code will follow existing Engine approach:
  - Raw hit processing/decoding [need updates for new ADC/TDCs]
    - Option to dump information needed for detector calibration
  - Track-independent detector quantities
  - Tracking
  - Track-dependent reconstruction
  - Calculate efficiencies for each detector, reconstruction step
    - Robust algorithms which yield reliable measure of performance.
    - Not tuned to specific experiment (e.g. extreme rates or backgrounds may require modified approaches)
  - Calculate 'basic' physics quantities for each event
  - Dump all info. on luminosity, efficiencies, deadtimes, etc… for run
- Heavy emphasis on experiment-independent issues
  - Each experiment…
    - Must provide higher-level physics reconstruction
    - Must decide if they want to use more specialized efficiency calculations
    - Must determine efficiency of experiment-specific cuts
  - Most are fairly well standardized, with multiple 'default' options

(S)HMS Analysis Algorithms (I)

- *Current Fortran code identical for HMS and SOS, with detailed detector layout defined by position/geometry parameter files*
- *SHMS differs only by geometry; can use identical code*
  - *New code needed to implement additional functionality, e.g. w.r.t. multihit TDCs, flash ADCs, special benefits of quartz hodoscope.*
- *The initial version of the C++/ROOT Hall C code will NOT change any of the existing algorithms!*

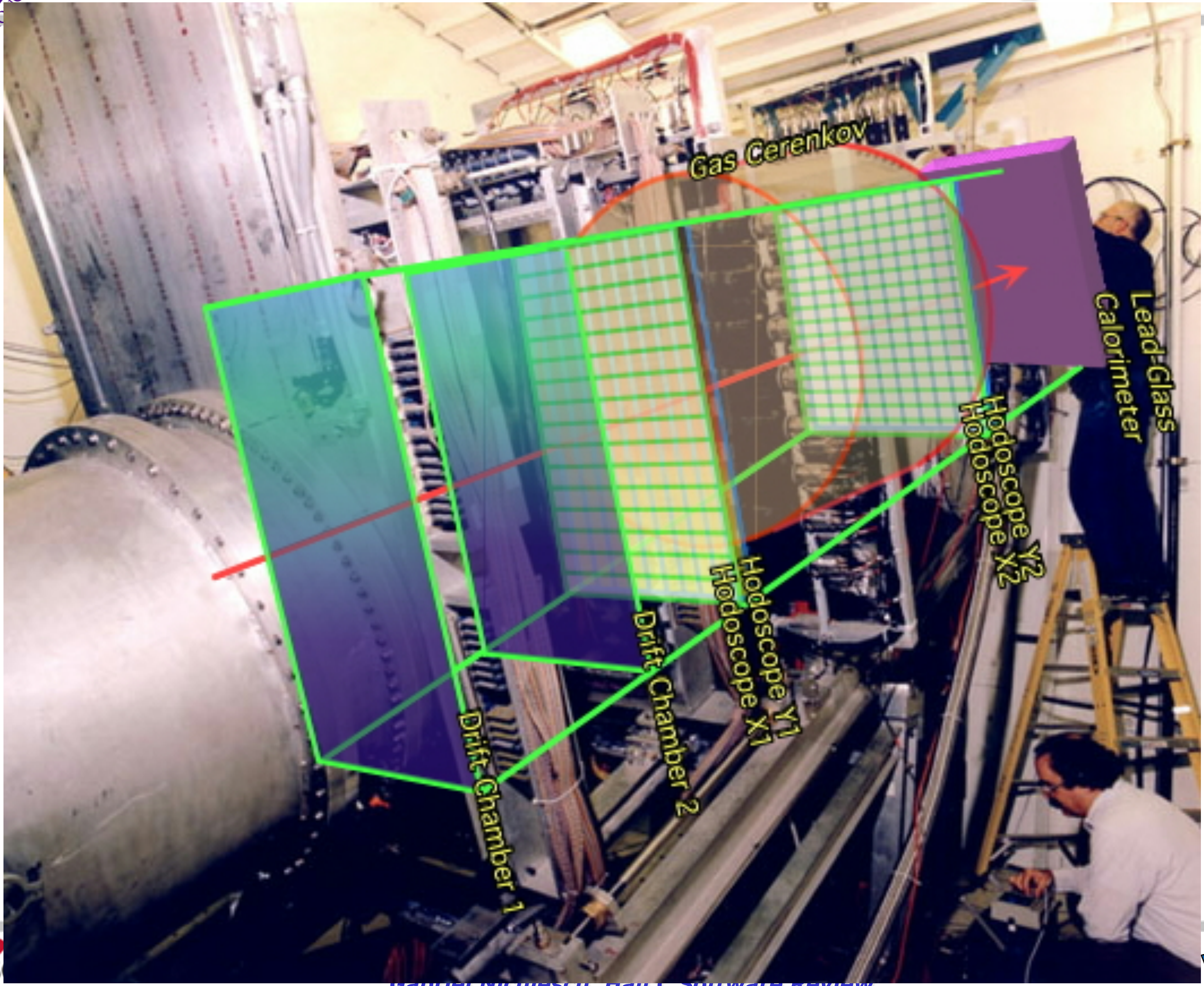
(S)HMS Analysis Algorithms (II)

- For each detector:
  - Calibration:
    - Options to dump information for diagnostic/detector calibration
    - Some detectors have built-in calibration options
  - "Detector reconstruction"
    - Tracking, TOF, npes, shower energy, etc
  - Efficiency Calculation
    - default efficiency calc. built in
    - Options for experiment-specific calculations
- For each spectrometer:
  - Track and particle identification (at the target)
    - Use all detector information
    - Use matrix elements
    - Options to apply various corrections (eloss, pointing, etc.)

(S)HMS Analysis Algorithms (III)

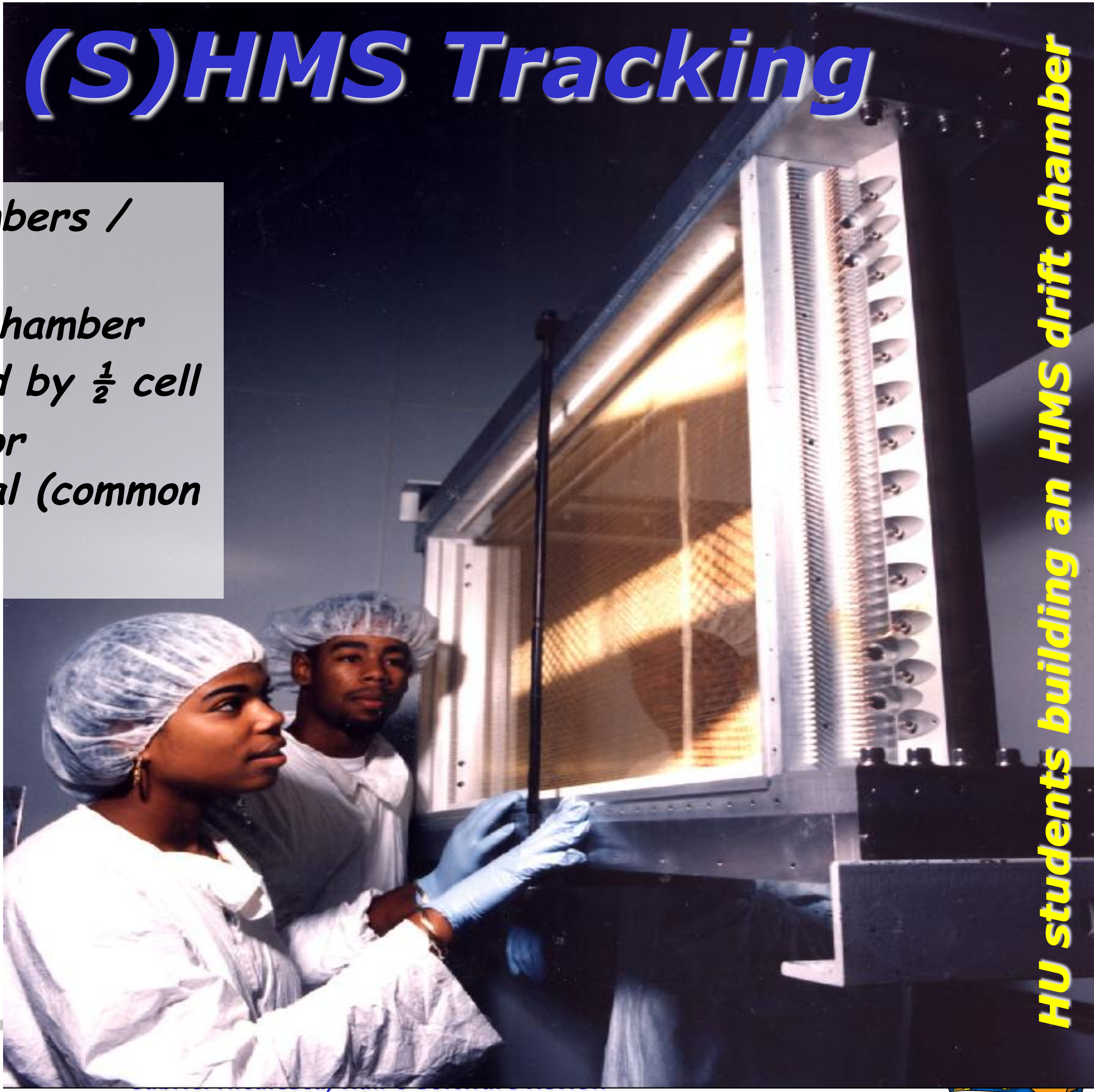
- *If a coincidence experiment:*
  - *Coincidence event reconstruction:*
    - *Very Experiment-specific!*
    - *The Fortran code provides subroutines for this (c_reconstruction, c_physics, etc.). Careful with +es and –es!*
    - *The default code only provides calculation for very basic quantities*
- *Historically some/most users just produce a coincidence ntuple and use their own coincidence reconstruction software (.f, .kumac, or both)*

HMS Detector Array



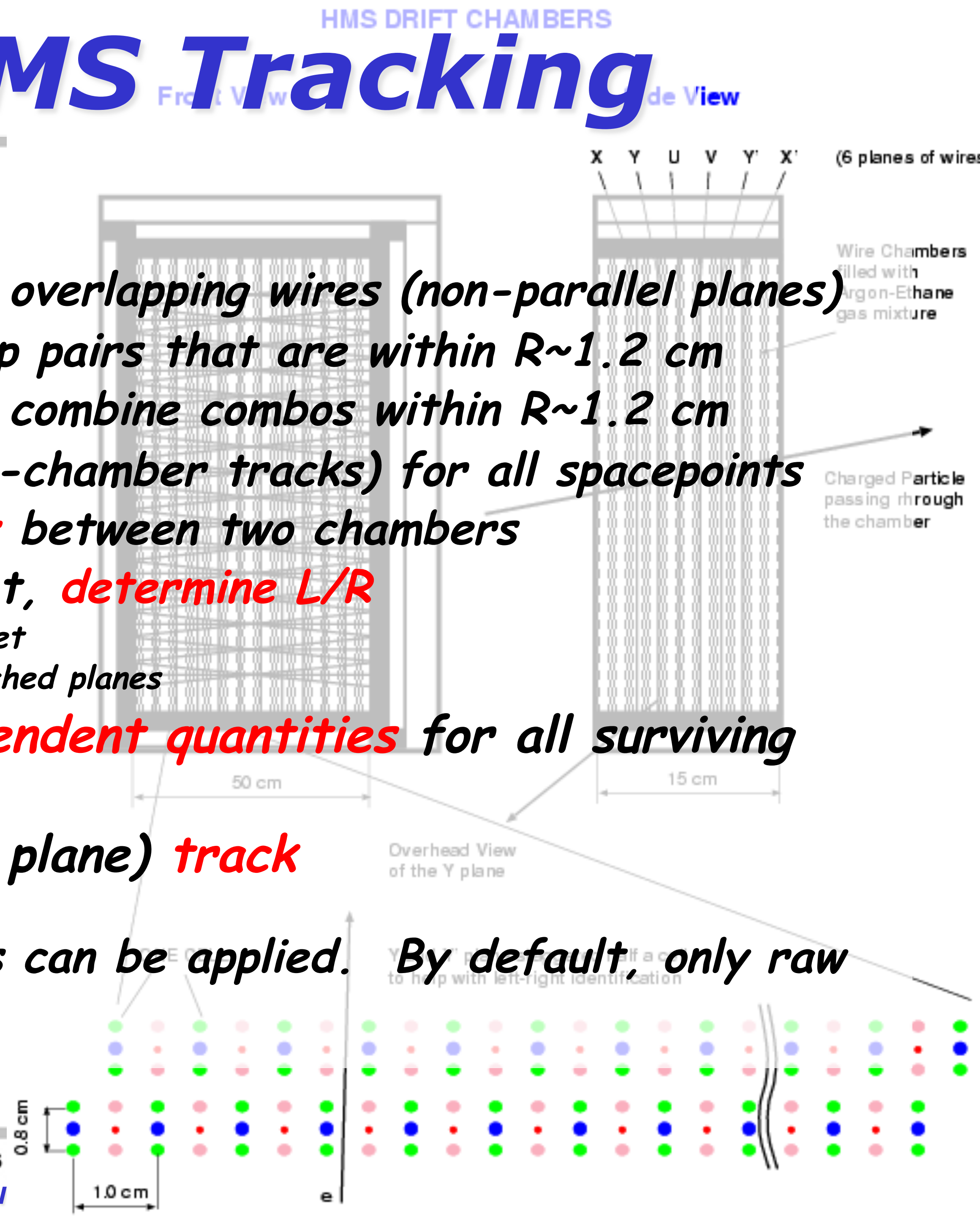
(S)HMS Tracking

- *Two drift chambers / spectrometer*
- *6 wire planes/chamber*
- *|| wires shifted by ½ cell*
- *Need scintillator hodoscope signal (common stop)*



HU students building an HMS drift chamber

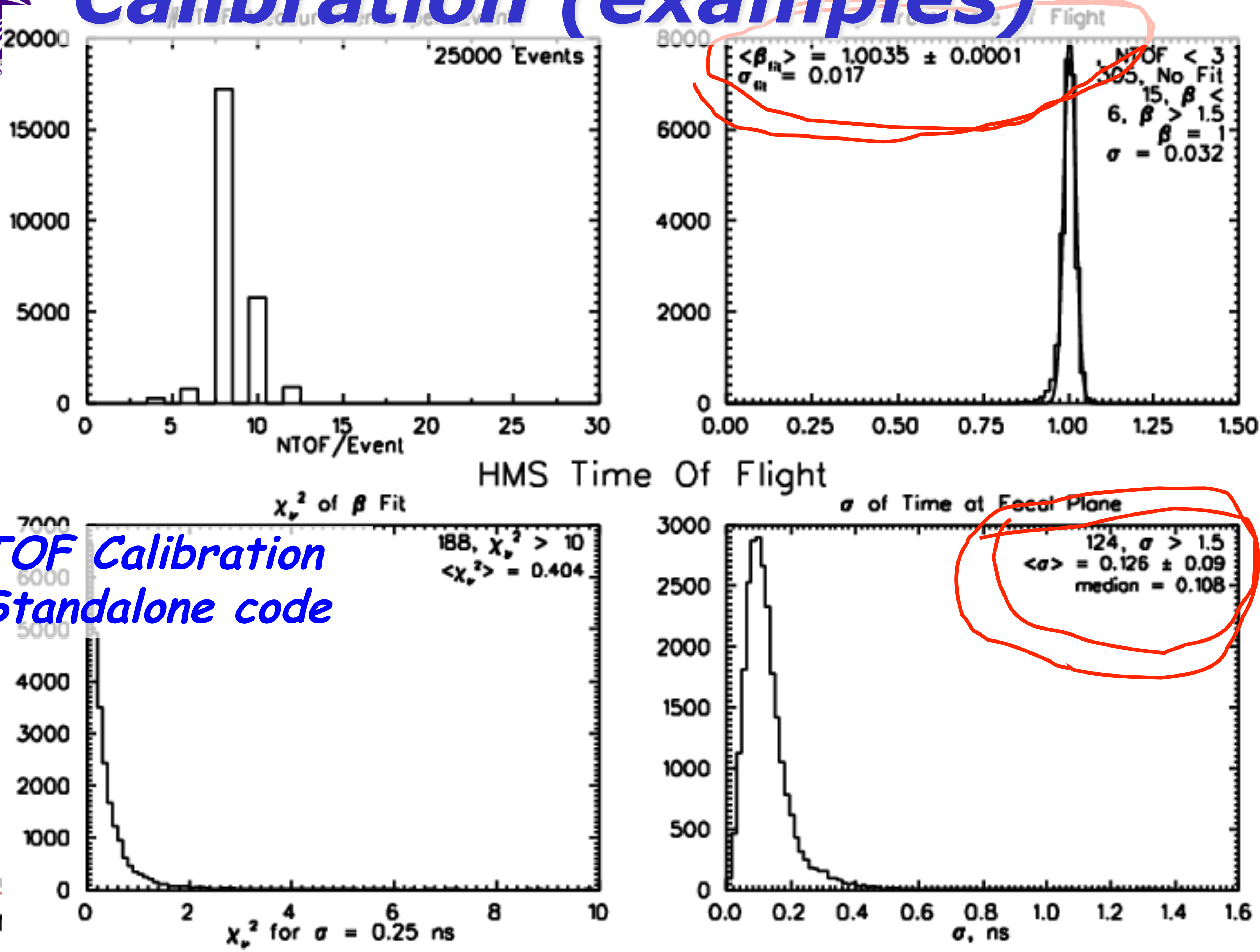
(S)HMS Tracking



- ***Tracking algorithm:***
  - *Identify pairs: hits in overlapping wires (non-parallel planes)*
  - *Identify combos: group pairs that are within R~1.2 cm*
  - *Generate spacepoints: combine combos within R~1.2 cm*
  - *Generate stubs (single-chamber tracks) for all spacepoints*
  - *Find tracks: link stubs between two chambers*
  - *Apply drift time offset, determine L/R*
    - *Between planes where offset*
    - *Best chisquared for unmatched planes*
  - *Calculate track-dependent quantities for all surviving tracks*
  - *Select 'final' (focal plane) track*

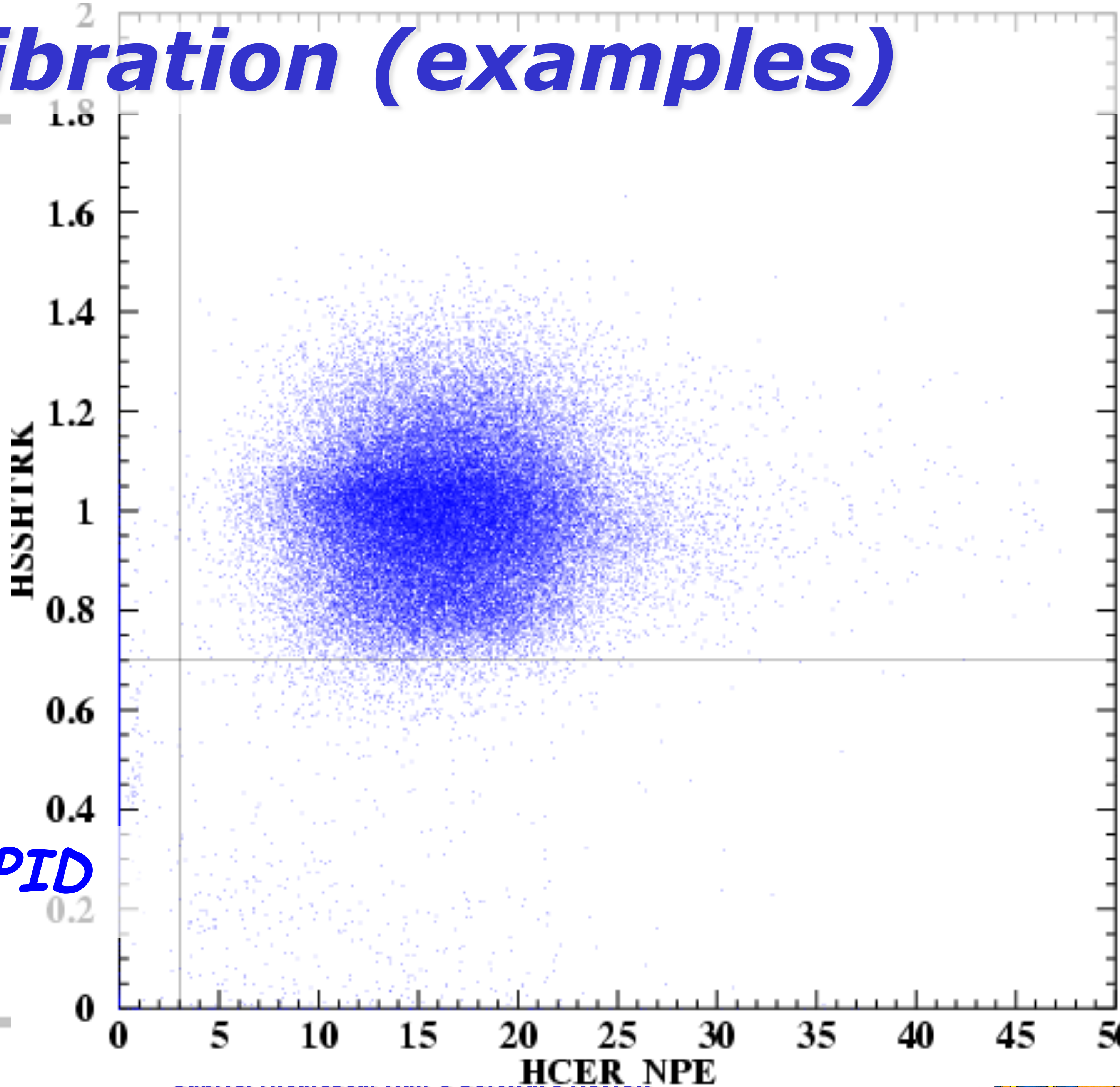
- *Various places where cuts can be applied. By default, only raw timing on DC TDC hits.*

Calibration (examples)



- *TOF Calibration*
- *Standalone code*

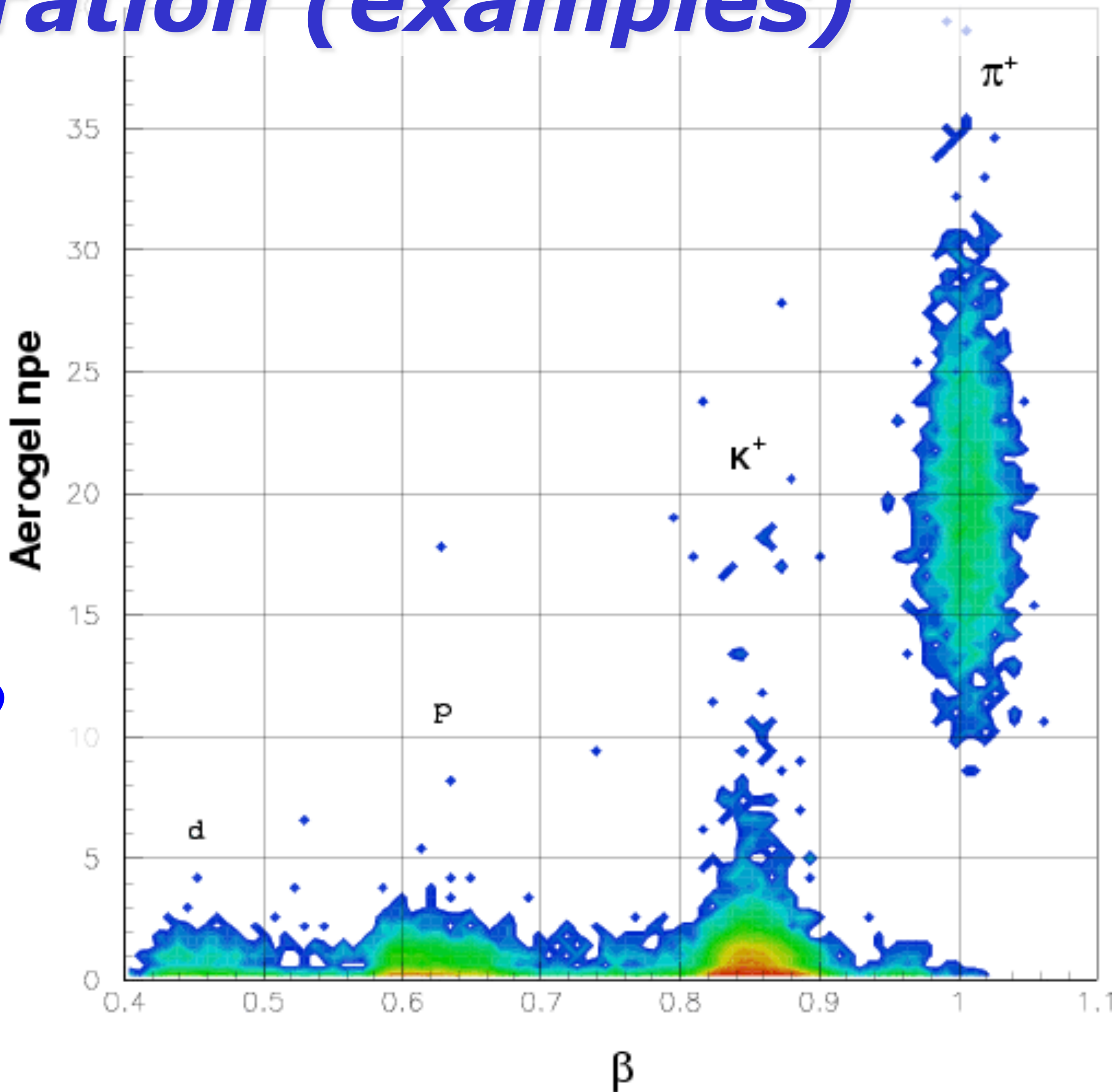
Calibration (examples)



- Electron PID

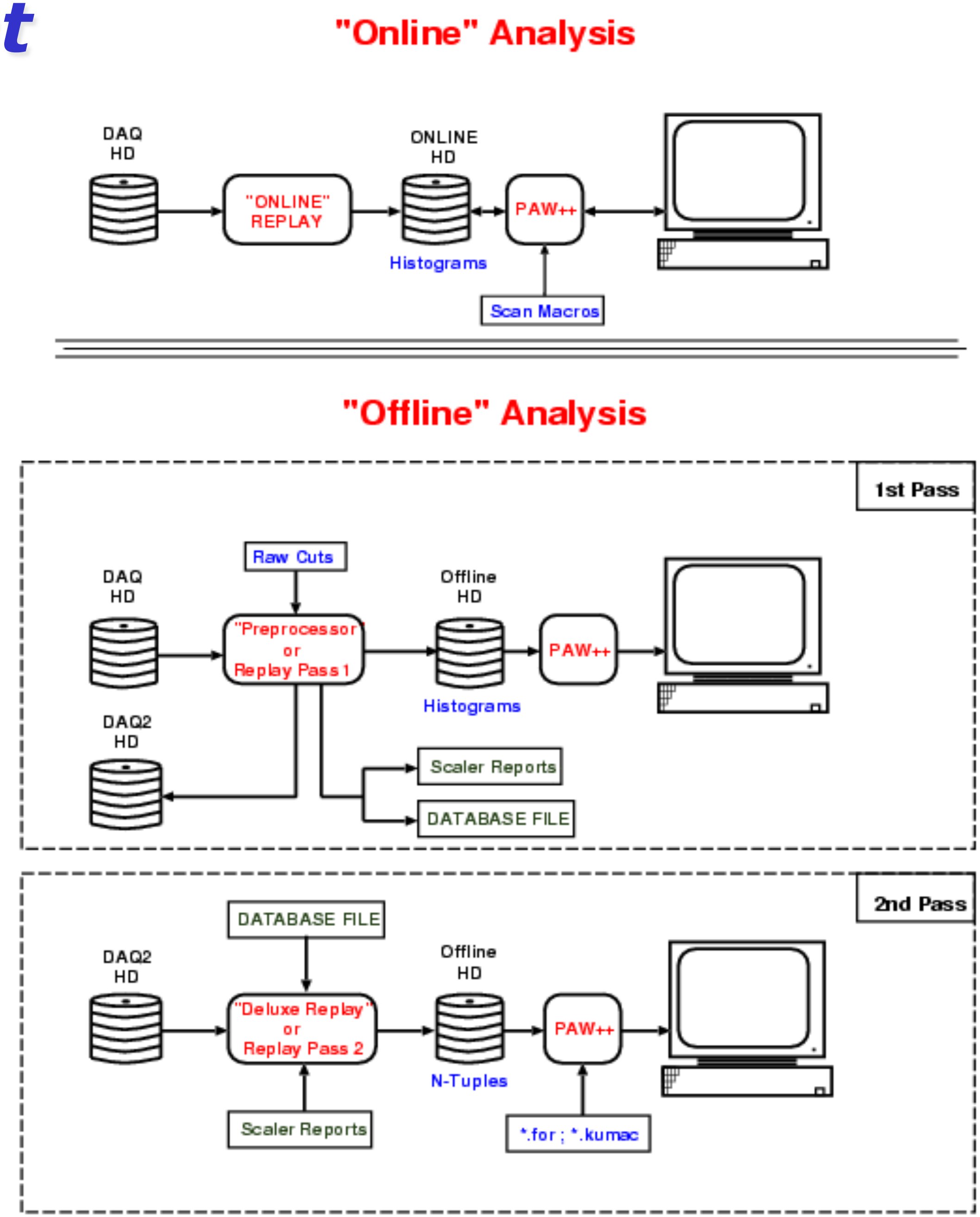
Calibration (examples)

- *Hadron PID*



Beyond single event reconstruction

- *Sample workflow for typical Hall C experiment*
- *Programs & scripts available to the user*
- *Most scripts will work “out of the box”*
- *Some will require user customization*



Beyond single event reconstruction

- *Internal efficiency calculations (detector, tracking…)*
  - *Experiment-specific modifications (e.g. tracking efficiency vs. particle type, high-background data, singles vs. (rare) coincidences)*
- *Scalar analysis*
  - *Defaults*
  - *Time charts (dump of E/x/y vs. t)*
  - *beam on/off*
- *Diagnostics*
  - *raw detector dumps*
  - *online monitoring*
- *Calibrations*
  - *Internal (including continuous)*
  - *Data dump for external calibrations*

Beyond single event reconstruction

- *Corrections (beam drift, FR, special[coin/cer block...])*
- *Cuts, normalized yields in arbitrary bins.*
- *Simulation: ratio method, include all 'physics' event-by-event, apply global corrections (mean eloss) as in analyzer*

- *At present, Engine has robust default estimates for efficiencies, calibrations, etc… with options for specific conditions (or user-provided experiment-specific changes).*
- *New framework: may want multiple (or selectable) versions to allow simpler selection of optimized approach. Fortunately this capability already exists in PODD.*

Fast Physics Monte Carlo: SIMC

- Standard Hall C (or A!) MC for coincidence reactions
  - HMS/SOS/HRS/SHMS optics models (COSY)
  - Aperture checks, detector FID cuts yield acceptance
  - Detailed implementation of radiative corrections, multiple scattering, ionization energy loss, particle decay
  - Simple prescriptions for some FSIs, Coulomb corrections
- Reactions implemented:
  - Elastic and quasielastic H(e,e'p), A(e,e'p)
  - Exclusive pion and kaon production
    - H(e,e'π^+)n, A(e,e'$\pi^{+/-}$) [quasifree or coherent]
    - H(e,e'K^+)Λ,Σ, A(e,e'$K^{+/-}$) [quasifree or coherent]
    - H(e,e'$\pi^{+/-}$)X, D(e,e'$\pi^{+/-}$)X [semi-inclusive]
    - H(e,e'$K^{+/-}$)X, D(e,e'$K^{+/-}$)X [semi-inclusive]
    - H(e,e'$\rho \rightarrow \pi^+\pi^-$)p, D(e,e'$\rho \rightarrow \pi^+\pi^-$) [diffractive ρ]

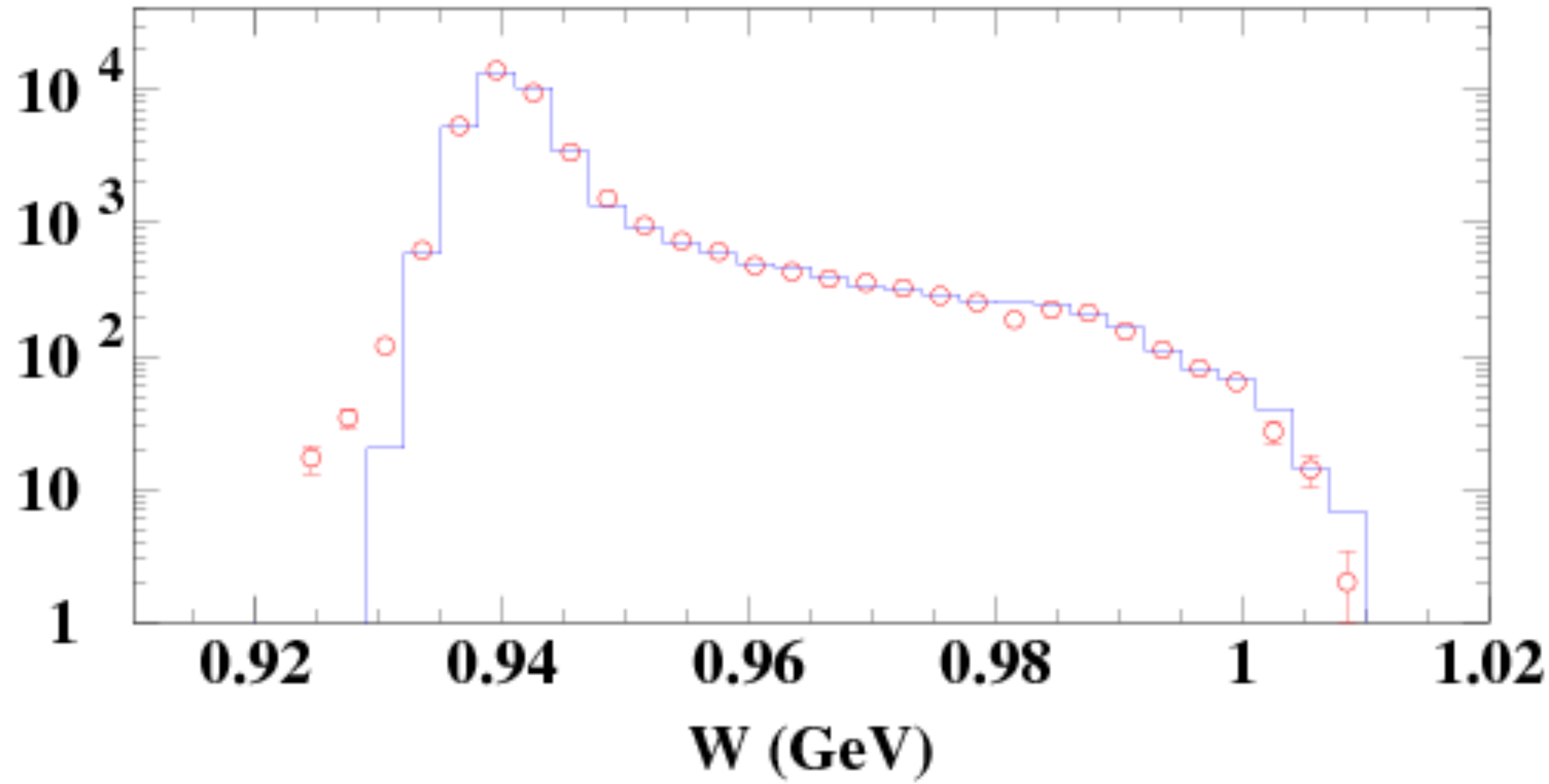
Fast Physics Monte Carlo: SIMC

- *SIMC is NOT a full detector response simulation a la GEANT*
  - *Detectors implemented via impact on acceptance*
  - *Efficiency corrections (generally) applied to data*
- *SIMC does NOT simulate a large class of processes simultaneously to generate backgrounds (e.g. like Pythia)*
- *SIMC is NOT a generic event generator:*
  - *Processes generated over a limited phase space, based on fully populating spectrometer acceptances*
- *SIMC is NOT hard to modify*
  - *Many non-standard or experiment-specific tests can easily be performed based on simulated information*

Data vs. SIMC

Hydrogen elastic:

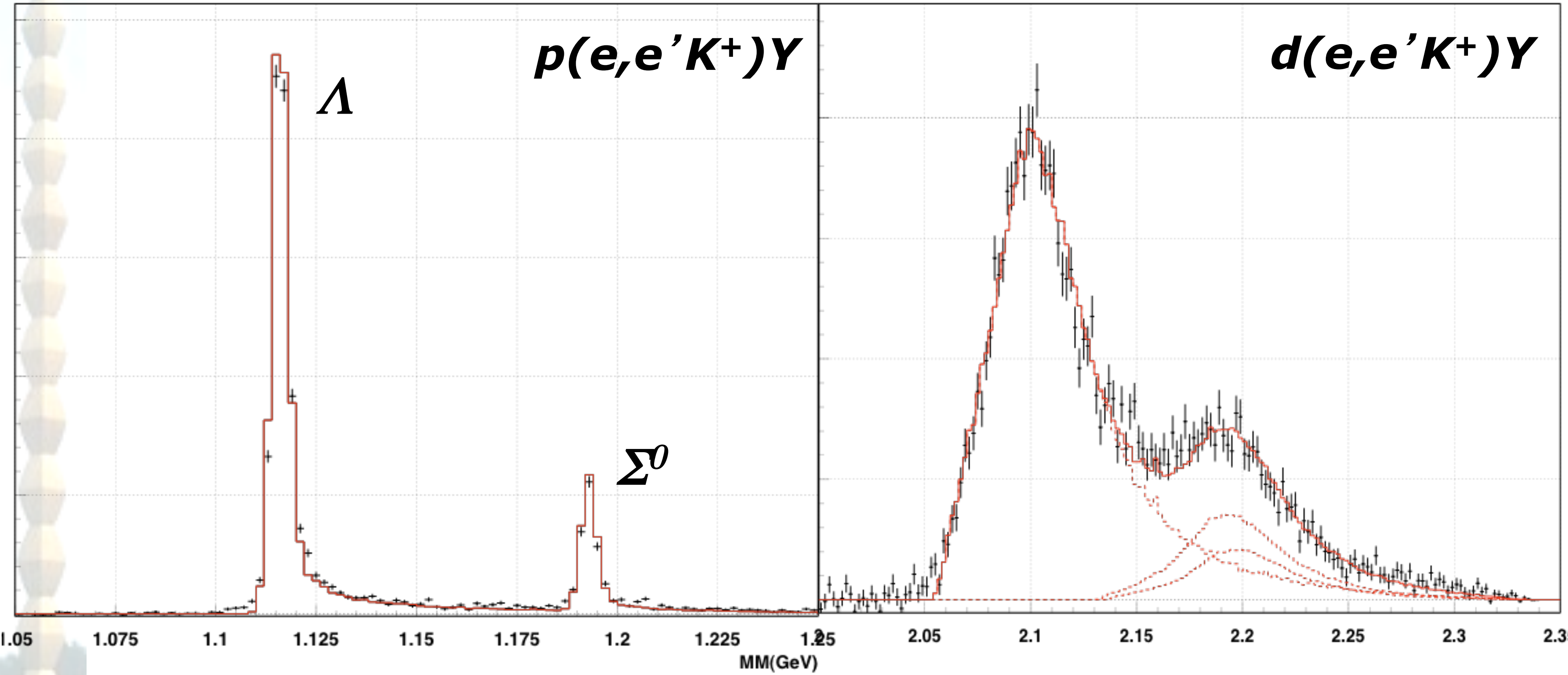
Radiative correction + fit to form factors as input
NO other free parameters!



Data vs. SIMC

Kaon electroproduction:

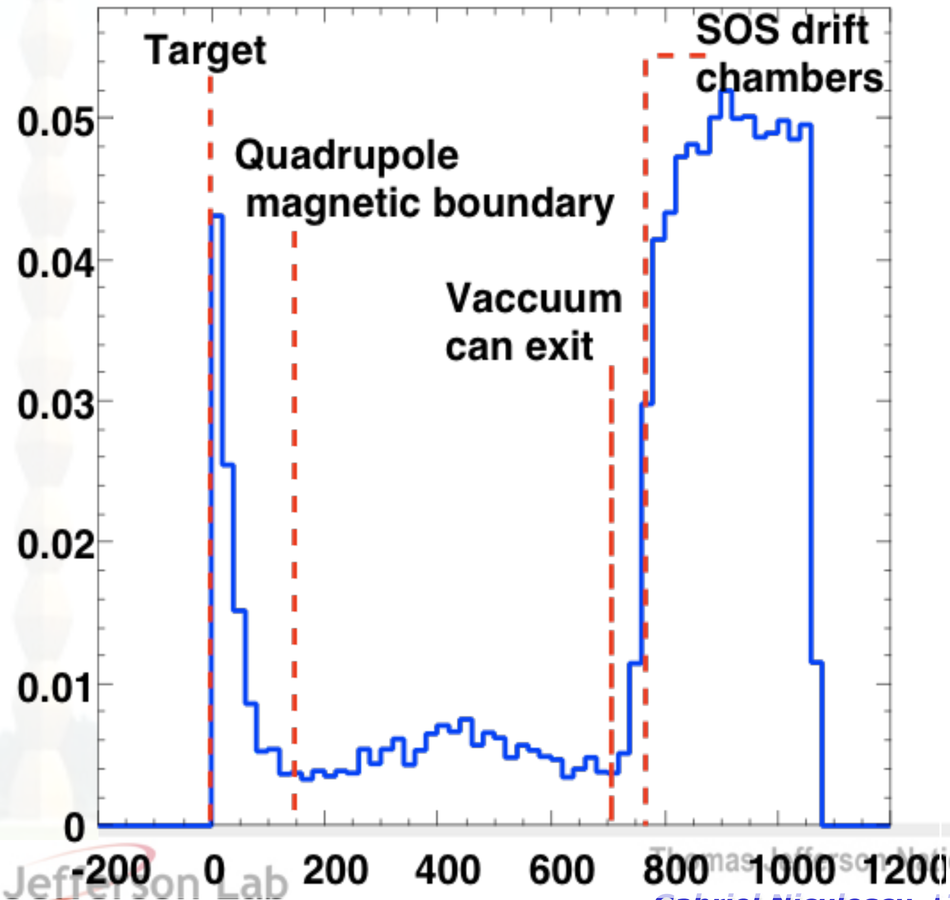
Hydrogen hyperon cross section model +
^{2}H (also $^{3,4}He$) spectral functions +
FSI + Kaon decay + RC + Norm. Factor for Λ, Σ peaks



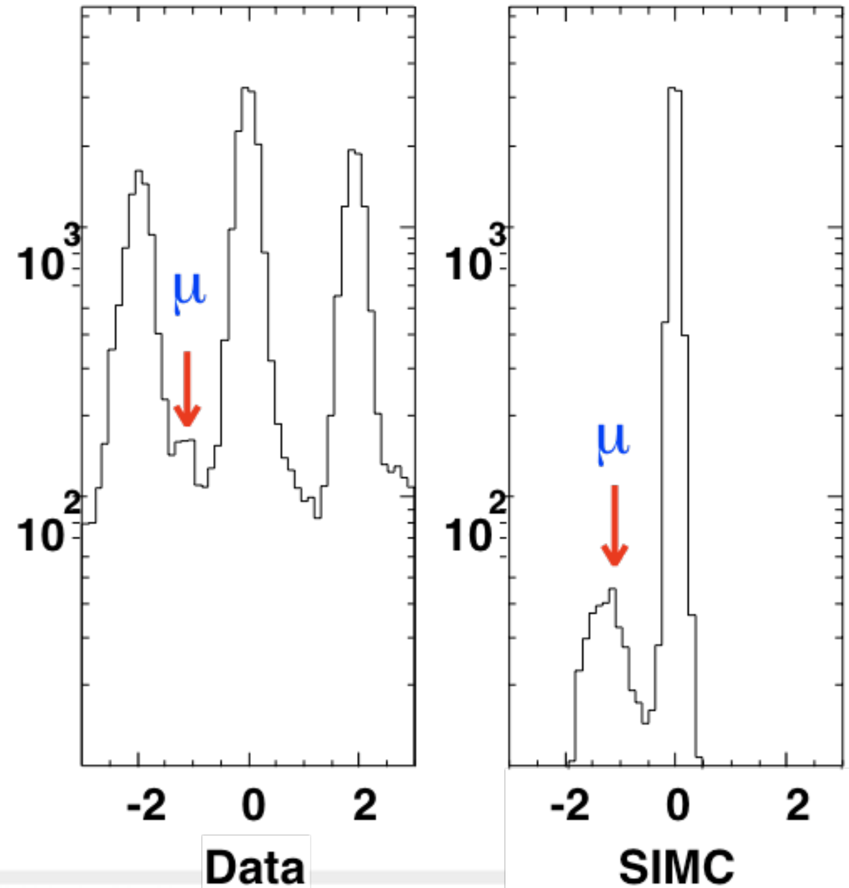
Pion electroproduction

Examining decay-at-target events

Simulated decay distance (cm) for accepted π−>μν decays



Coincidence time (ns)



Summary

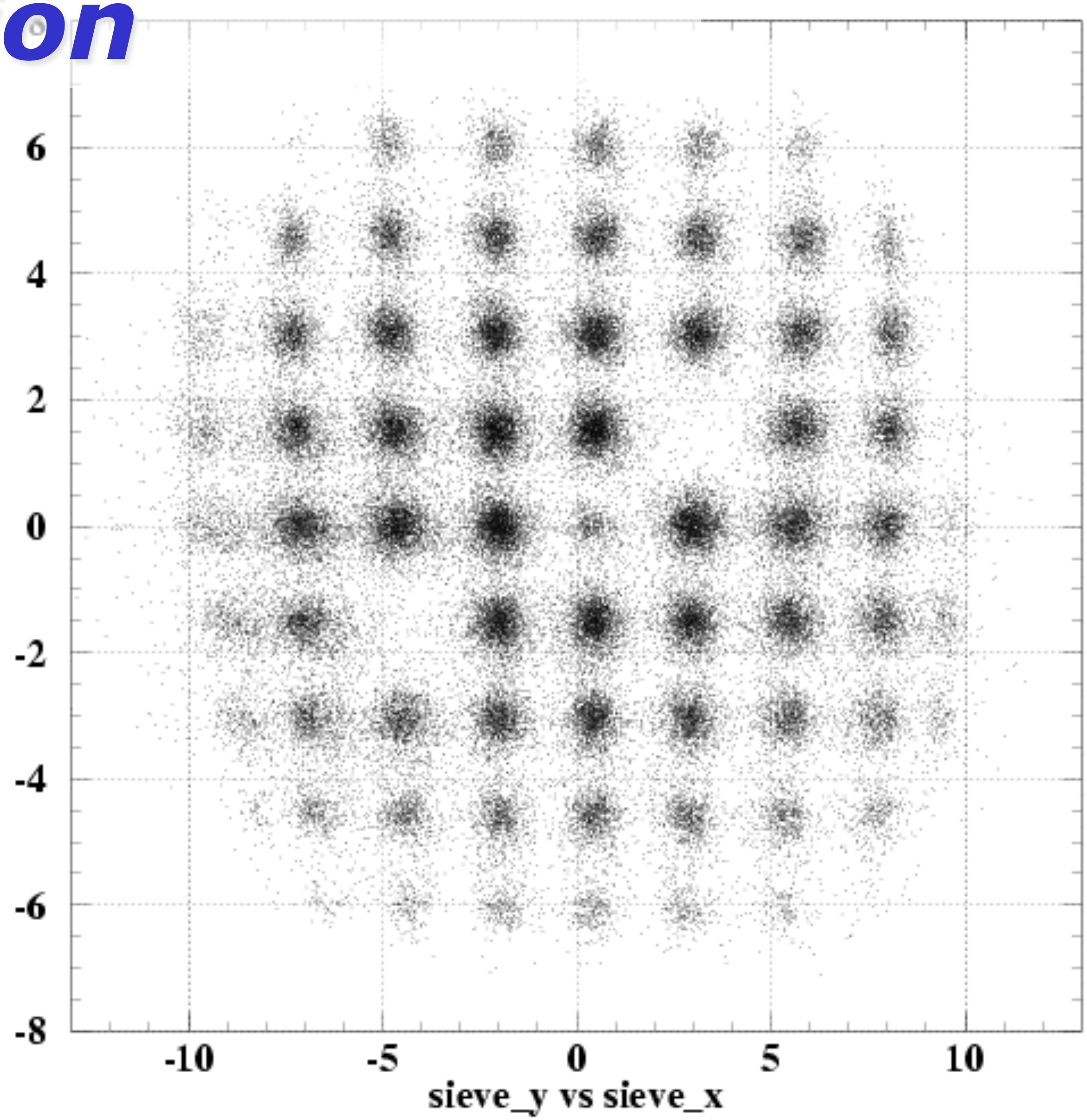
- ROOT/C++ analyzer
- Analysis algorithms (tracking, pid, etc.)
- Workflow
- Calibration
- Simulation

Back-up

- *Back-up from here on*

COSY matrix element optimization

- .



Jefferson Lab

SIMC Overview [probably backup]

- *Initialization*
  - *Choose reaction, final state (if appropriate)*
  - *Disable/enable implementation of (or correction for) raster, eloss …*
- *Event generation*
  - *Select vertex based on target size, position, raster size, beam spot size*
  - *Determine energy, angle generation that will populate 100% of the acceptance (accounting for radiation, energy loss, …)*
- *Physics Processes*
  - *Event-by-event multiple scattering, radiative corrections, particle decay, coulomb corrections*

Overview [probably backup]

- *Acceptance*
  - *Can apply geometric cuts or spectrometer model. Default spec. models include target/spec. offsets, model of magnetic elements, apertures at front, back, middle of magnets, collimators, detector active area*
- *Event Reconstruction*
  - *Tracks are fitted in the focal plane and reconstructed to the target. Apply average energy loss, fast raster corrections (consistent with data analysis). Calculate physics quantities for Ntuple.*