TKR Simulation and Reconstruction Overview

People

Simulation and Digitization

Reconstruction

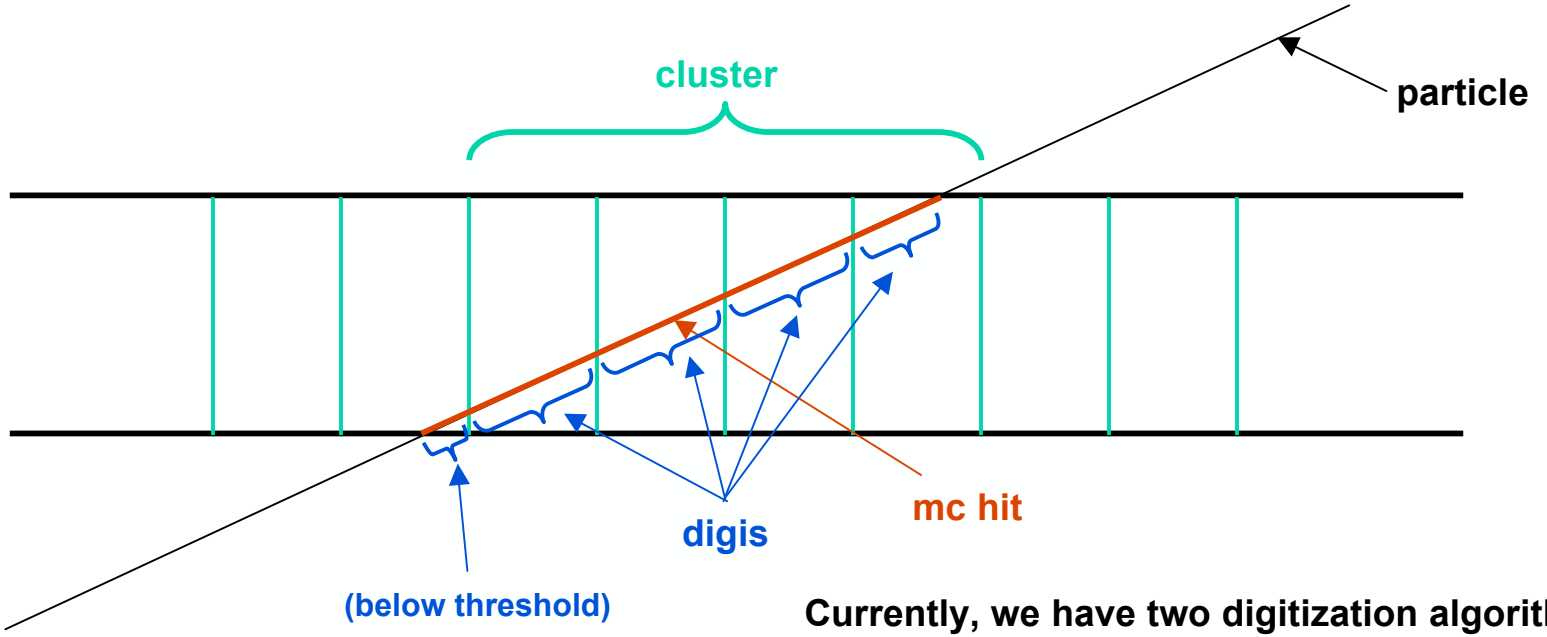
Status / Summary

Tracker Reconstruction Manpower
(Condensed from Delta-PDR)

- **TKR software team at SLAC**
  - Manpower
    - Tracy Usher
    - Leon Rochester
    - Hiro Tajima
  - Major Tasks
    - Track and Vertex Reconstruction
    - Geometry, calibration, Alignment, …
    - Support, Maintenance and Documentation
    - Analysis
- **TKR Software team at UCSC**
  - Manpower
    - Bill Atwood
    - Brian Baughman
  - Major Tasks
    - Track and Vertex Reconstruction
    - Analysis
- **TKR Software team at Pisa**
  - Manpower
    - Michael Kuss
    - Johann Cohen-Tanugi
  - Major Tasks
    - Vertex Finding and Fitting
    - Algorithm test package
- **TKR Software teams at Bari and Perugia**
  - Manpower
    - N.Giglietto (Bari)
    - M.Brigida (Bari)
    - C. Cecchi (Perugia)
    - M. Pepe (Perugia)
  - Major Tasks
    - Simulation and Digitization
    - ToT

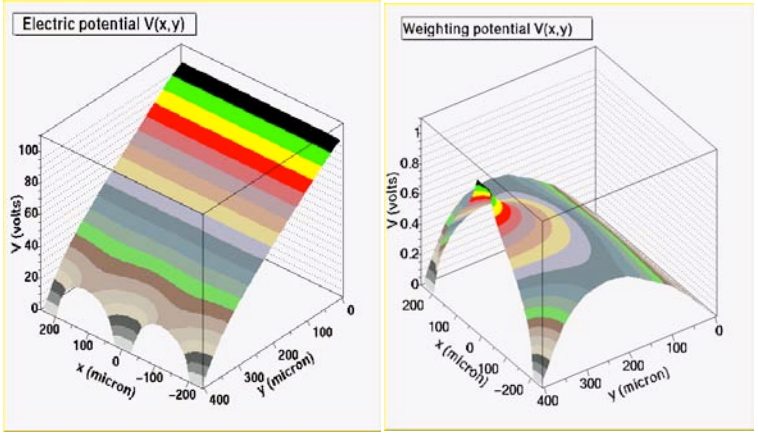
Simulation / Digitization

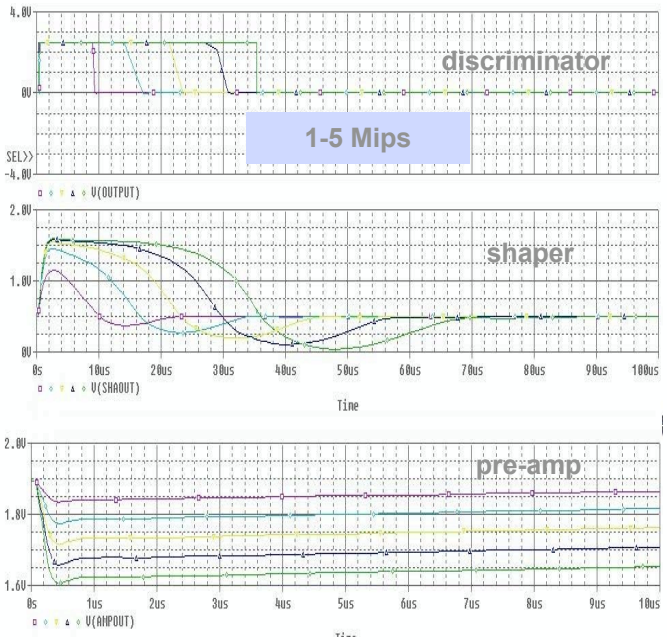
Geant4 treats the entire silicon plane as a unit. Energy is deposited with "landau" fluctuations. Digitization figures out which strips are hit.
Later, in the reconstruction phase, the clustering algorithm groups adjacent strips.



Currently, we have two digitization algorithms: In the simple digitization (SimpleDigiAlg), deposited energy is divided according to path length (no fluctuations). Time-over-threshold is linear in deposited energy.

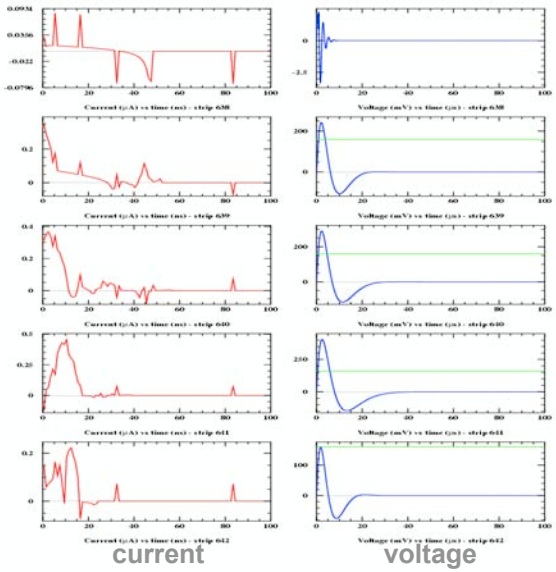
Bari Digitization





The second algorithm (BariDigiAlg) is a complete model:

- **electrostatics**
- **ionization clusters**
- **electron-hole drift**
- **electronic pulse-shaping**
- **electronic noise**
- **time over threshold**



Time above threshold (green line) for 5 adjacent strips

Simulation and Digitization Overview

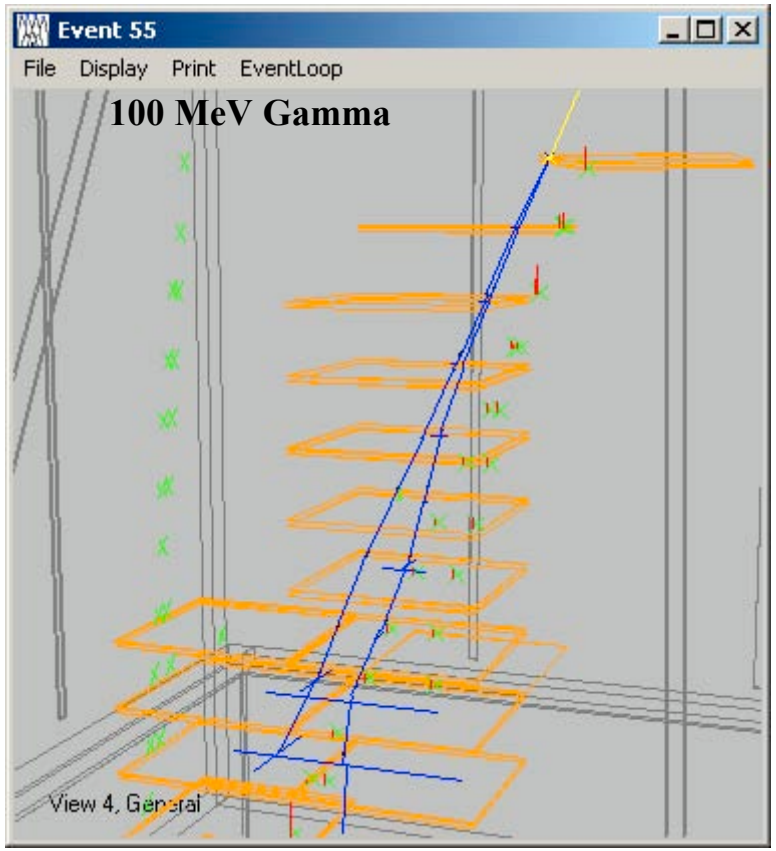
- **Simulation**
  - Some tunable parameters; see performance talk
- **Digitization: two algorithms are complementary**
  - Simple digitization is the default
    - Fast, but “simple”
    - Can be refined with results from the Bari digitization
    - Interface is most developed
      - standard random number generator
      - relational tables
      - random noise hits
  - Bari digitization
    - Gives more nuanced information
    - Currently very slow
    - Recently interfaced to Gleam; above features not yet in place
    - Is now being used to study ToT in Engineering Module (EM)

TkrRecon Reconstruction

The Problem

Basic Goals:

- Determine the incident direction of gamma rays converting within the tracker
- Provide help for rejecting backgrounds
- Augment the event energy determination



Challenges:

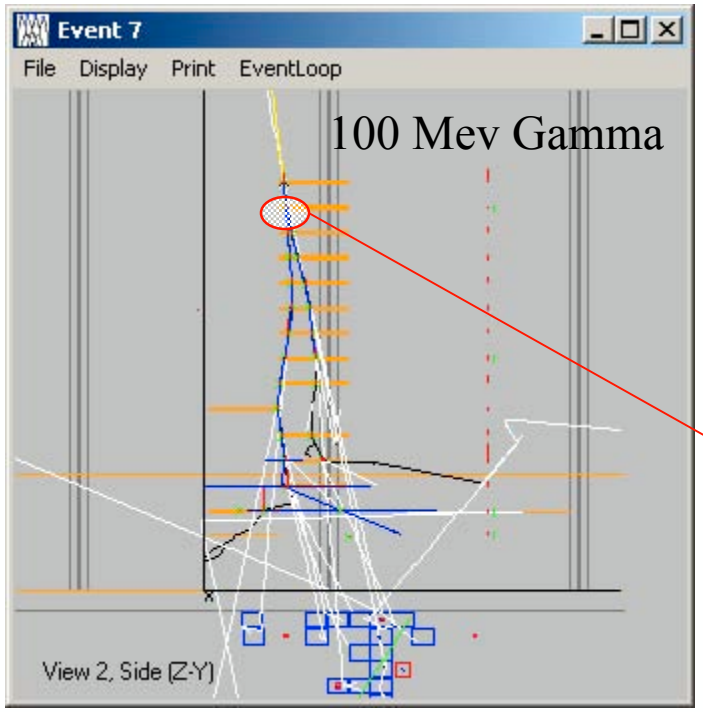
- Want to reconstruct Gammas across a wide energy range:
  - From less than 30 MeV
  - To greater than 100 GeV
- Silicon strips in x and y projections only
  - No stereo projections – ambiguities can arise in attempting to mate x and y projections to form 3D tracks.
- Don't know individual track energy
  - Cal returns total event energy, cannot "see" individual track energies
- Material in the Tracker creates special problems for tracking the electron and positron resulting from the gamma conversion:
  - Multiple Scattering
  - Production of secondaries from Bremsstrahlung
  - These processes occur primarily in the tungsten converters but also in the other materials comprising the tracker
  - Not all gammas convert in the Tungsten radiators…

TkrRecon Reconstruction Overview

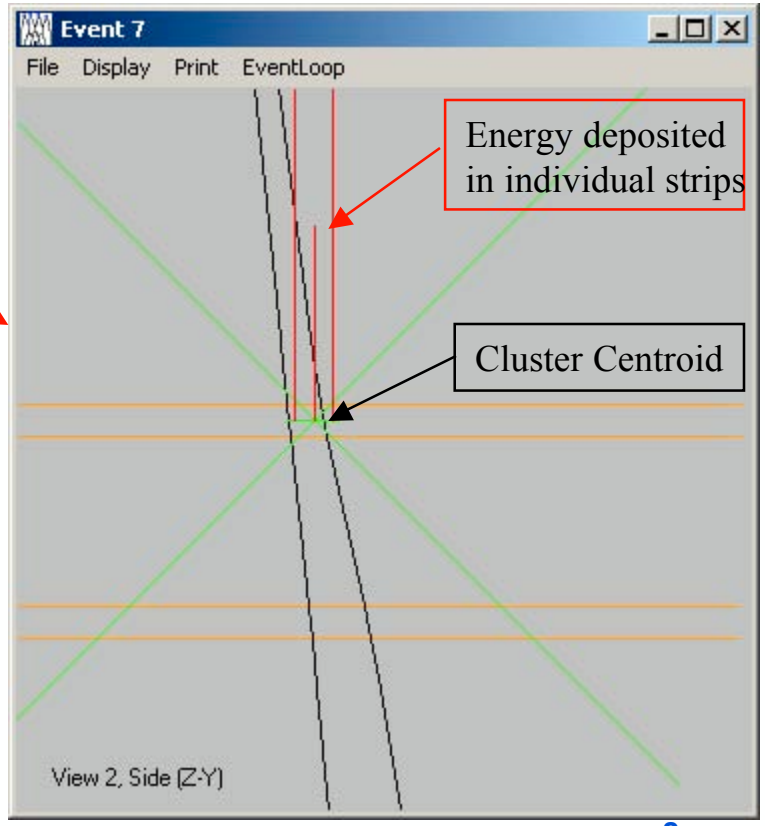
- **Basic goals for the reconstruction**
  - Determine the incident direction of gammas converting within the tracker
  - Provide help for rejecting Cosmic Ray backgrounds
  - Augment the event energy determination
- **Additional goals for the organization of the reconstruction code**
  - Interchangeability
    - Provide the ability to easily change a particular reconstruction algorithm
    - Allows for the development of alternate methods for solving the problem
  - Reduction in complexity
    - Break into smaller well defined tasks
      - Easier to understand each piece separately
      - Allows more people to be involved
  - Improve long term maintainability
    - Smaller pieces easier to understand for future code maintainers
    - Provide documentation to aid future code maintainers
  - Geometry independent
    - All geometry information obtained externally to the TkrRecon package
    - Provide for the ability to easily switch between various test modules

Step-by-Step Recon Overview

Step 1: Clustering



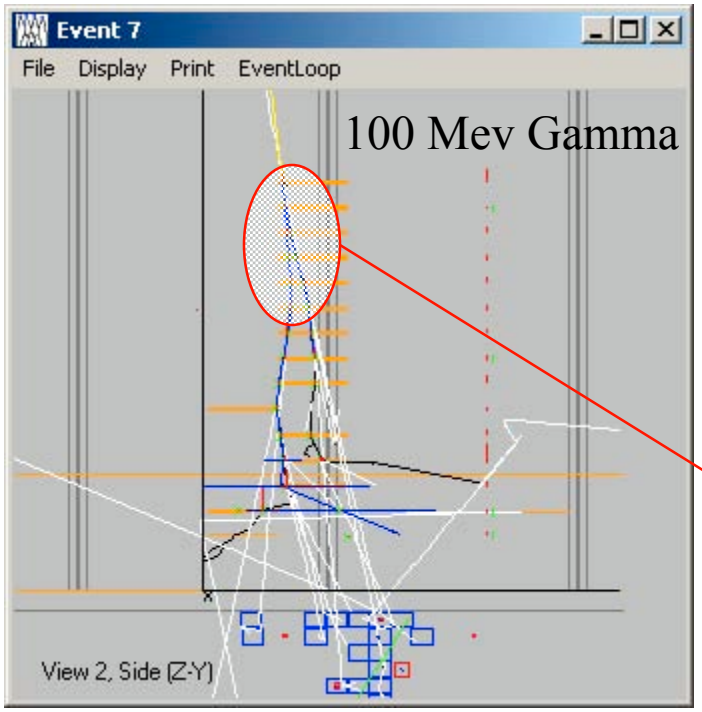
•Simulation deposits energy in silicon layers crossed by particles
•Digitization apportions energy to individual strips and then determines which are "hits"



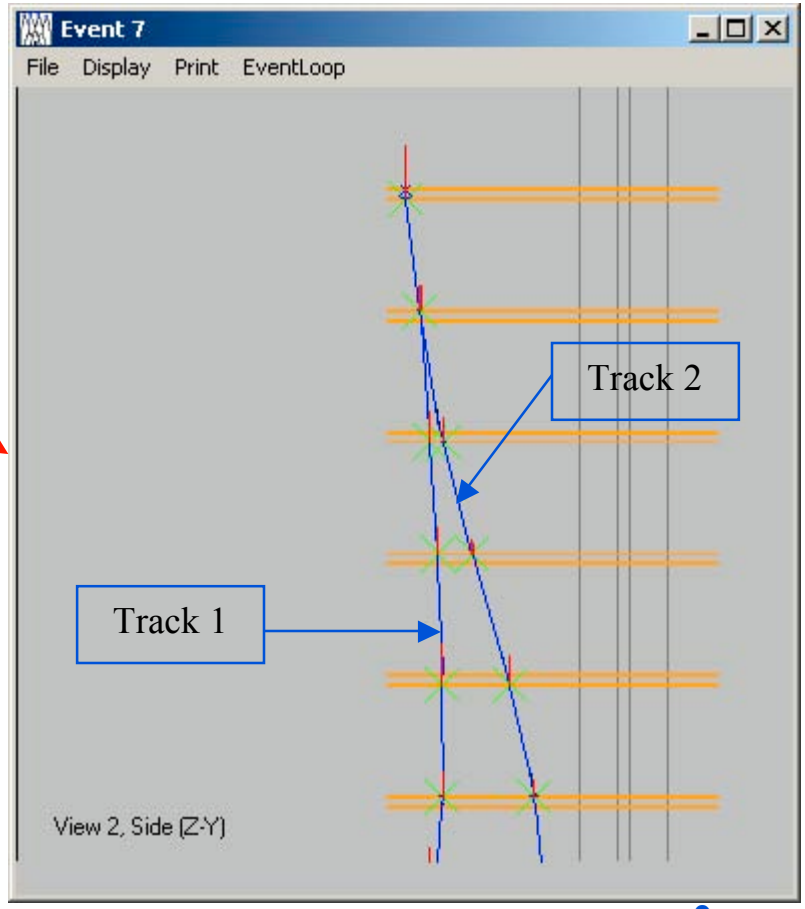
Recon Step 1: Clustering
- Adjacent hit strips combined to form centroid
- Strip ID's converted to position
- Also (coming soon):
  - Hot/dead strips
  - Alignment

Step-by-Step Recon Overview

Steps 2 & 3: Tracking Finding and Fitting



Recon Step 2: Track Finding - associate clusters to form candidate tracks

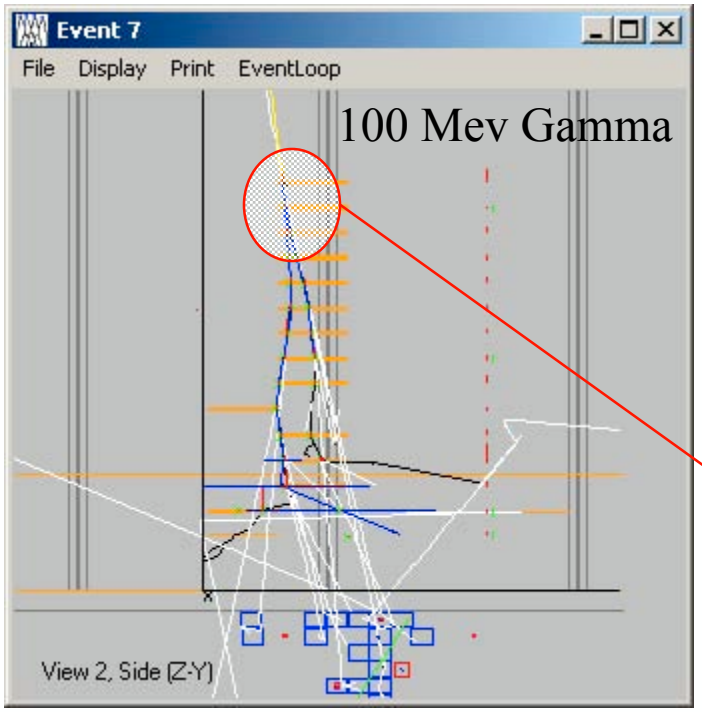


Recon Step 3: Track Fit - Perform fit to associated clusters (from track finding candidates) to obtain track parameters

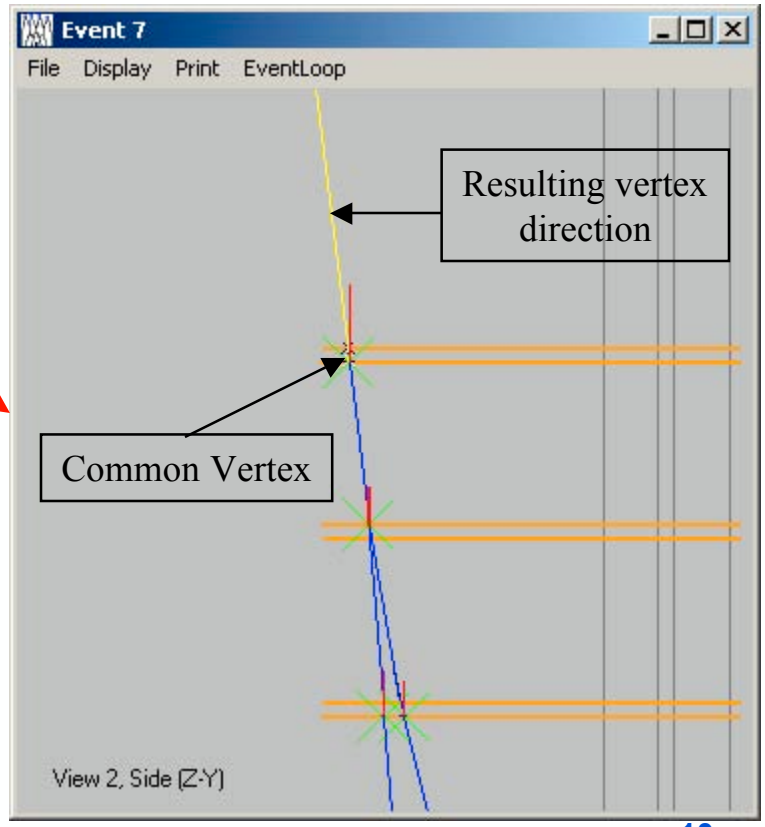
See Bill Atwood's talk following overview

Step-by-Step Recon Overview

Step 4: Vertexing



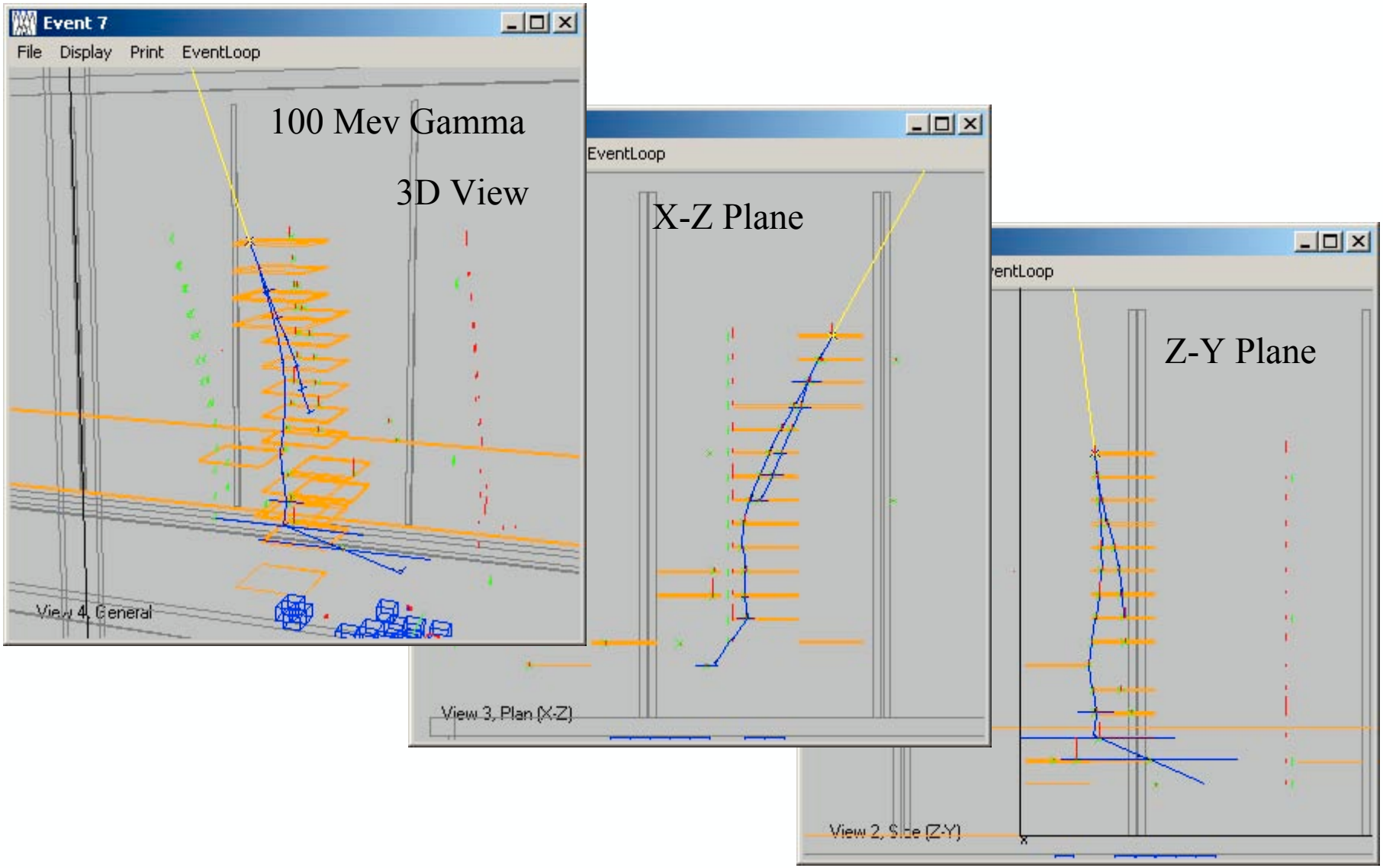
Recon Step 4: Vertexing – Find common intersection point of fit track pairs in event. Combine fit track parameters to get vertex direction.

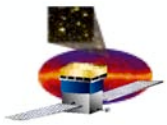


Again, see Bill Atwood's talk for more details

Step-by-Step Recon Overview

Final Product



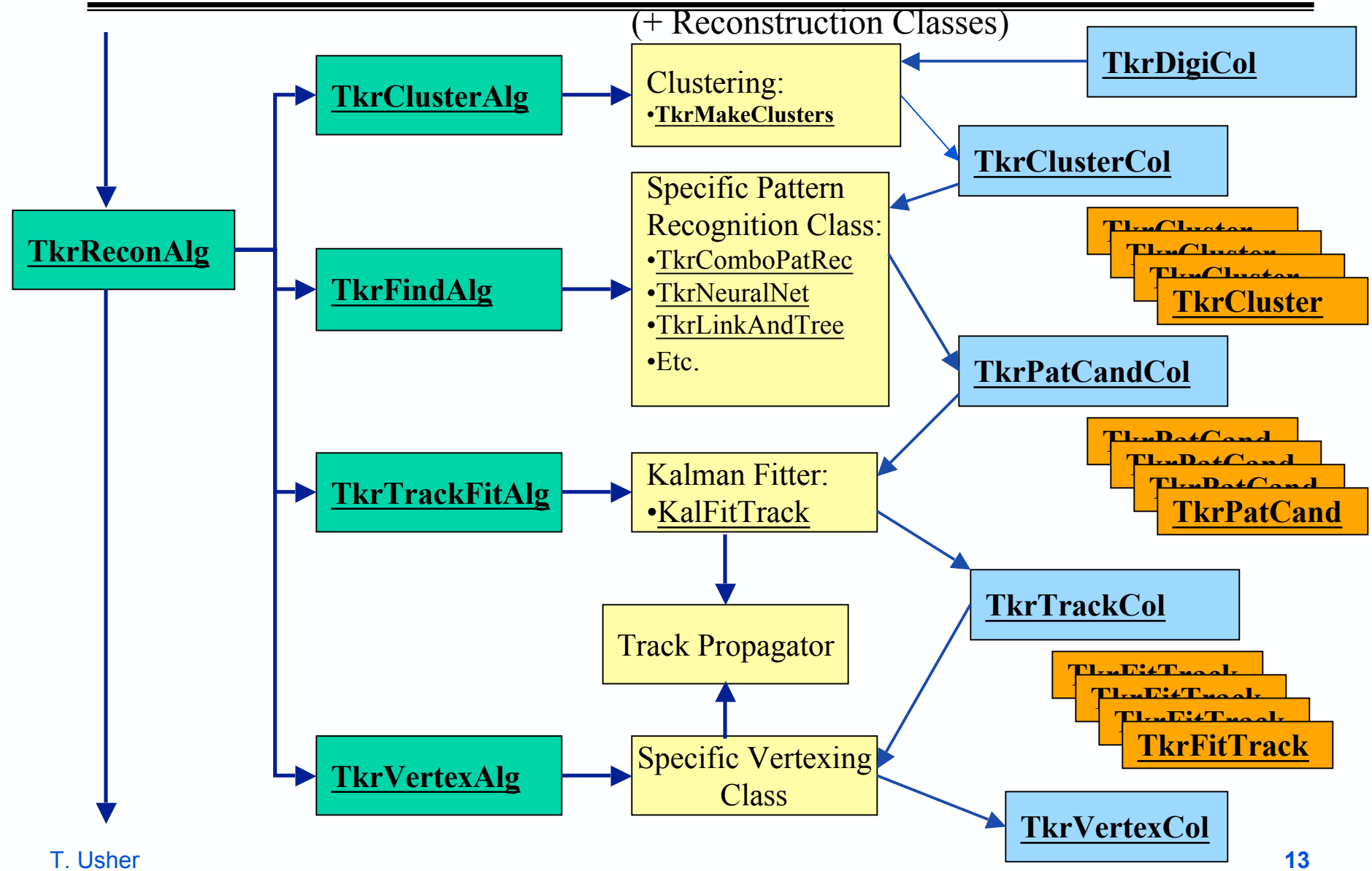


TkrRecon Reconstruction Overview

Code Organization

- **Organize the four main tasks into independent Gaudi "Algorithms"**
  - Each successive algorithm builds upon the work of the previous step
    - Clustering of hit strips → TkrClusterAlg
    - Track Finding → TkrFindAlg
    - Track Fitting → TkrTrackFitAlg
    - Vertex Finding and Fitting → TkrVertexAlg
  - Above implemented as Gaudi "SubAlgorithms" of a main driving algorithm
    - TkrReconAlg
- **All output stored in the Gaudi "Transient Data Store" (TDS)**
- **Algorithm Interchangeability achieved through the use of Gaudi "Tools"**
  - Particular reconstruction method implemented as a Gaudi "Tool"
  - SubAlgorithm then uses the right tool for the job
    - Can be selected at initialization
    - Can be changed "on the fly" during execution
- **Use Gaudi "Services" to provide necessary information**
  - Geometry (and alignment)
  - Reconstruction Constants
  - Calibration
  - Etc.

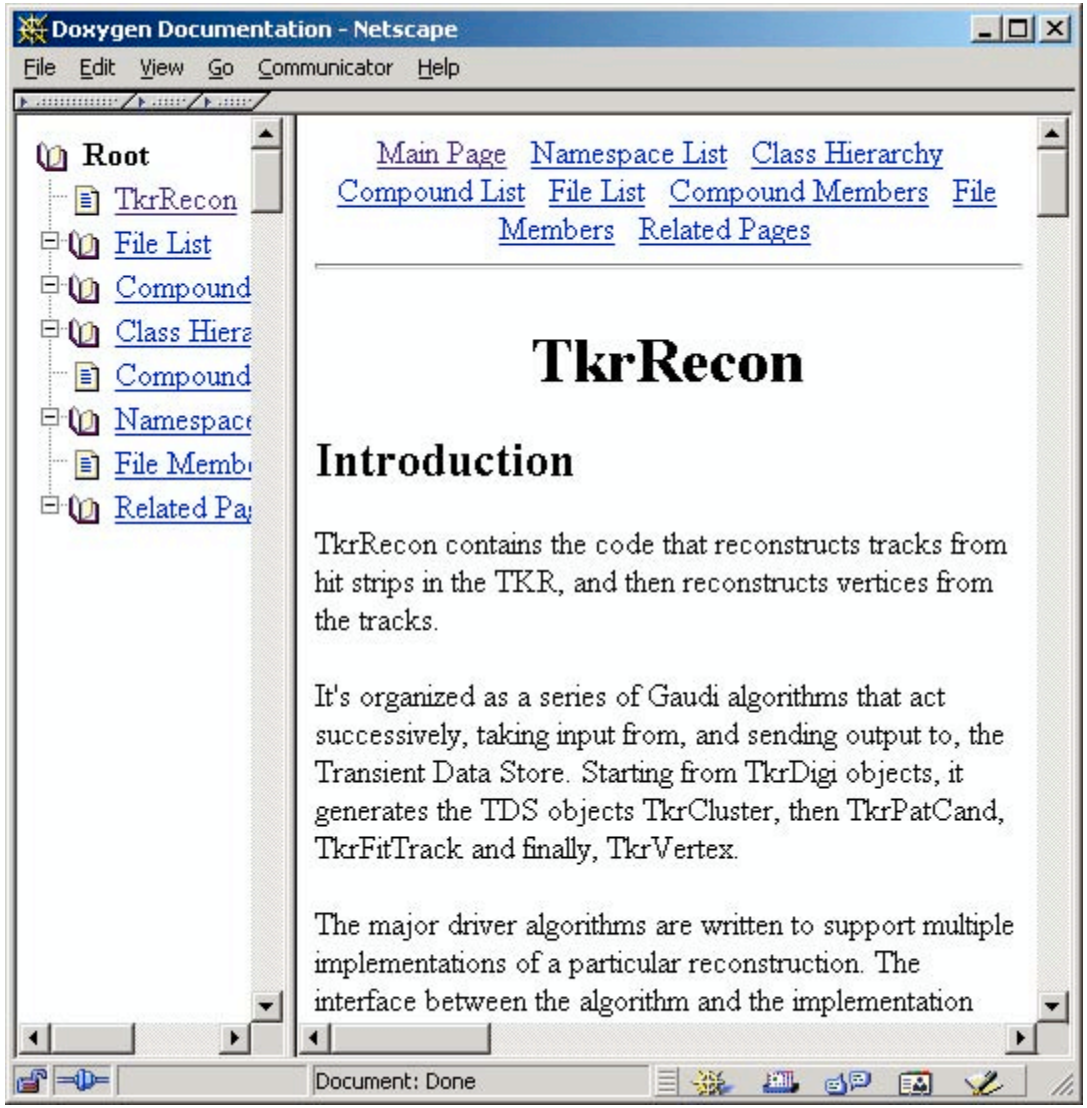
Tracker Reconstruction Diagram



TkrRecon Reconstruction

Code Documentation

- **Documentation Exists!!**
  - Doxygen comments inserted into code
  - Code and algorithm descriptions exist
  - Recon flow diagram added
  - etc.
- **Credit where credit is due**
  - Required by the Documenation Task Force
  - DTF has reviewed TkrRecon (twice!)
- **Go and see it yourself!**
  - Link at bottom of page
  - Or
    - Go to software web page
    - Follow link to DTF
    - Follow link to TkrRecon Review II



http://www-glast.slac.stanford.edu/software/core/documentation/reviews/TkrReconCodeReview_20020904.htm

TkrRecon Reconstruction

Summary

- **Since the PDR, TkrRecon has been successfully reorganized**
  - Reconstruction broken into smaller and easier to manage modules
  - Makes use of Gaudi Algorithms, Tools and Services to accomplish tasks
  - Geometry obtained from xml files via detModel
    - Currently only Full flight
  - Reconstruction constants separated into independent singleton object
    - Values can be modified in jobOptions file at initialization
- **Interchangeability feature has been demonstration**
  - Alternate track finding methods exist (but need more development)
  - Alternate vertex fitting method under development
- **Have completed two rounds of code documentation**
  - See Documentation Task Force page for TkrRecon
- **"Released" as part of the SAS September Release**
  - Default reconstruction the "Combo" recon
    - Again, see Bill Atwood's talk following this
- **Performance studies underway**
  - See final TkrRecon talk for brief survey of some current topics