Qscan

D. Lussi
-
Institute for Particle Physics, ETH Zurich

GLA2011 - Jyväskylä - 9. June

Collaborators (T32)

T. Maruyama, O. Araoka, T. Hasegawa, K. Kasami, N. Kimura, T. Kobayashi, M. Maki,K. Nishikawa, M. Tanaka,Y. Yamanoi,M. Yoshioka
High Energy Accelerator Research Organization (KEK), Tsukuba, Ibaraki, Japan

A. Badertscher, A. Curioni, S. di Luise, U. Degunda, L. Epprecht, L. Esposito, A. Gendotti, S. Horikawa, L. Knecht, C. Lazzaro, D. Lussi, A. Marchionni, A. Meregaglia, G. Natterer, F. Petrolo, F. Resnati, A. Rubbia, C. Strabel,T. Viant
Institute for Particle Physics, ETH Zürich, Switzerland

T. Mitani, Y. Nagasaka, J. Naganoma, A. Okamoto, H. Okamoto, K. Yorita
Waseda University, Tokyo, Japan

H. Naito, S. Narita
Iwate University, Morioka, Iwate, Japan

Outline

- **Introduction to Qscan**
- **Basic software structure**
- **Graphical User Interphase**
- **Geometry and (MC-) Data**
  - The ROOT geometry of T32
  - The virtual MC simulation for T32
- **Event reconstruction**
  - Noise subtraction
  - Pedestal subtraction
  - Hit finding and fitting
  - Clustering hits
  - Tracking
- **Conclusion**

Introduction: Qscan

Qscan is a general LAr-TPC software package, providing...

- **event visualization**
  - raw data displays (2D views, Fourier, Hough)
  - 3D display with full geometry (reconstructed tracks, MC tracks)
- **integrated MC simulation (VMC with ROOT geometry)**
- **event reconstruction tools**
  - noise reduction
  - hit finding, fitting
  - clustering and tracking algorithms

Moreover:

- Various detectors are already implemented.
- New detectors can be added in a simple way.

Introduction: History of Qscan

- **2000: Qscan implemented by the ETH Zurich group (A. Rubbia et al.) working on ICARUS 50L, T15 and T600 reconstruction and/or simulation**
- **2004-2007: Contribution from the University of Granada (A. Bueno et al.)**
- **2001-2007: Several publications / thesis**
  - J. Rico: *“First study with the stopping muon sample with the ICARUS T600 detector”*, PhD thesis (2002), ETH Zürich
  - S. Amoruso et al., *“Analysis of the liquid argon purity in the ICARUS T600 TPC”*, NIM A 516 (2004) 68-79
  - S. Amoruso et al., *“Study of electron recombination in liquid argon with the ICARUS TPC”*, NIM A 523 (2004) 275-286
  - F. Arneodo et al., *“Performance of a liquid argon time projection chamber exposed to the CERN West Area Neutrino Facility neutrino beam”*, Physical Review D 47 (2006)
  - A. Martinez, *“Study of Accelerator Neutrino interactions in a Liquid Argon TPC”*, PhD thesis (2007), Universidad de Granada
- **2010-2011: Revival**
  - new graphical user interphase (ROOT, openGL)
  - new Detectors: 3L LEM-TPC (see talk F. Resnati) and 250L/T32 (see talk T. Maruyama), new VMC, detector geometries, ...

Basic software structure (I)

basics

- code written in C++ (OO language)
- requirements: ROOT (compiled with pythia6), geant3-vmc
- subversion repository:

http://pcnometh4.cern.ch/viewvc/GLACIERrepos/RecoPackages/trunk/

software requirements

- The software has to be used for various different detectors.
- Adding a new detector should be straightforward.
- Common functionalities (example: MC tracking, noise generator,...) should be implemented only once

➡ each detector comes with separate classes which derive from general base classes (make use of C++ inheritance)

Basic software structure (II)

QscanApp
- read input info (command line or datacards)
- generates ROOT output file
- opens User Interphase
- loops over list of input files and opens new EventLooper for each new file

Qscan UI
(graphical or batch mode)

EventLooper
- reads the current input file
- Loads data class and geometry of the specific detector
- Loads next event and performs event reconstruction

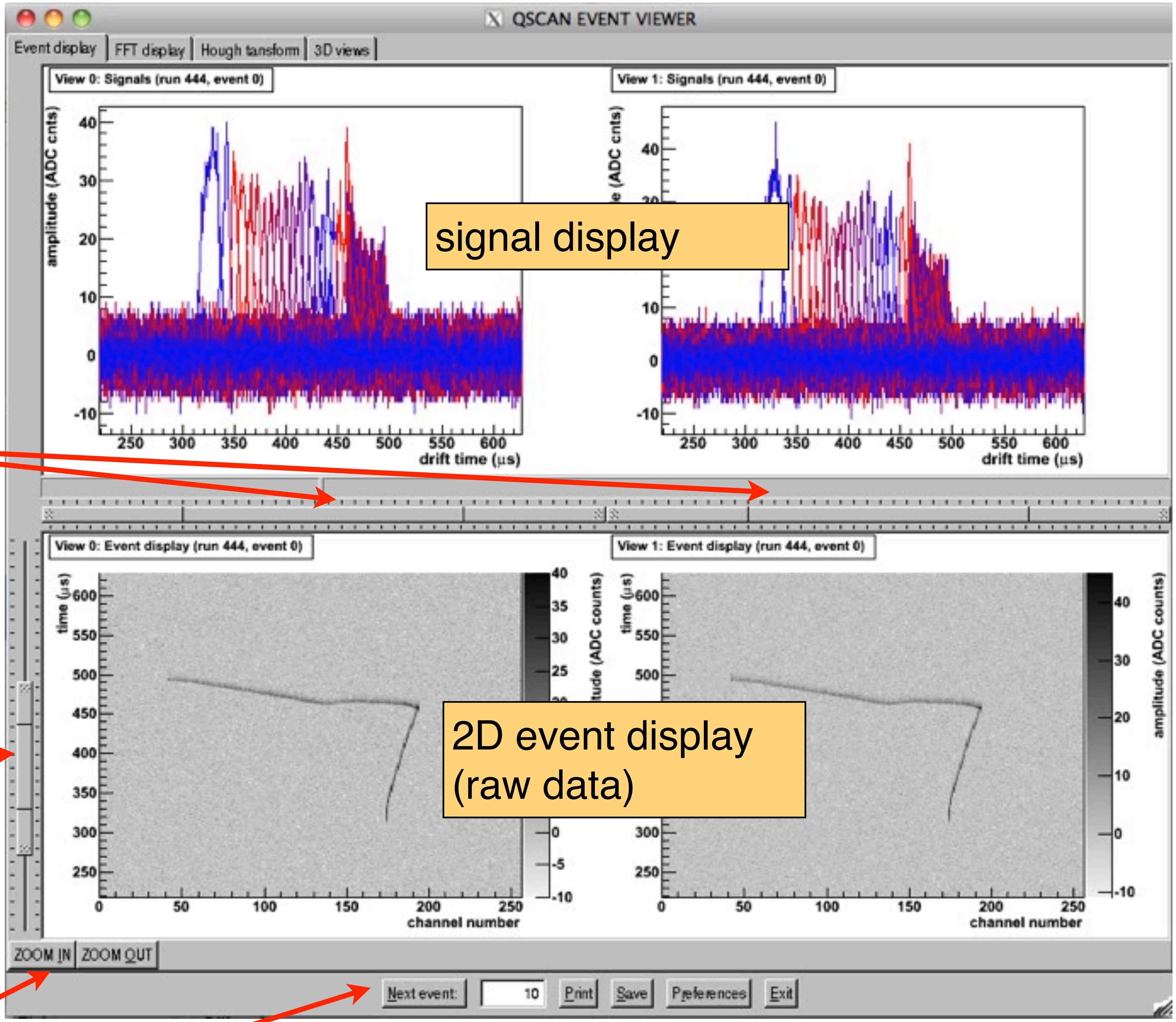
Data & geometry classes
- CAEN 3L and 250L
- ICARUS 50L, T15 and T600
- Virtual MC (VMC, MC250L,MC10T)

➡ Detector specific!

Reconstruction Classes
- BasicHits (hit finding, fitting)
- EventClusters
- EventTracks
- EventTracks3D, vertices,...

The Graphical User Interface (I)

other displays:
- -Event display
- -FFT display
- -Hough transform
- -3D views



signal display

wire range sliders

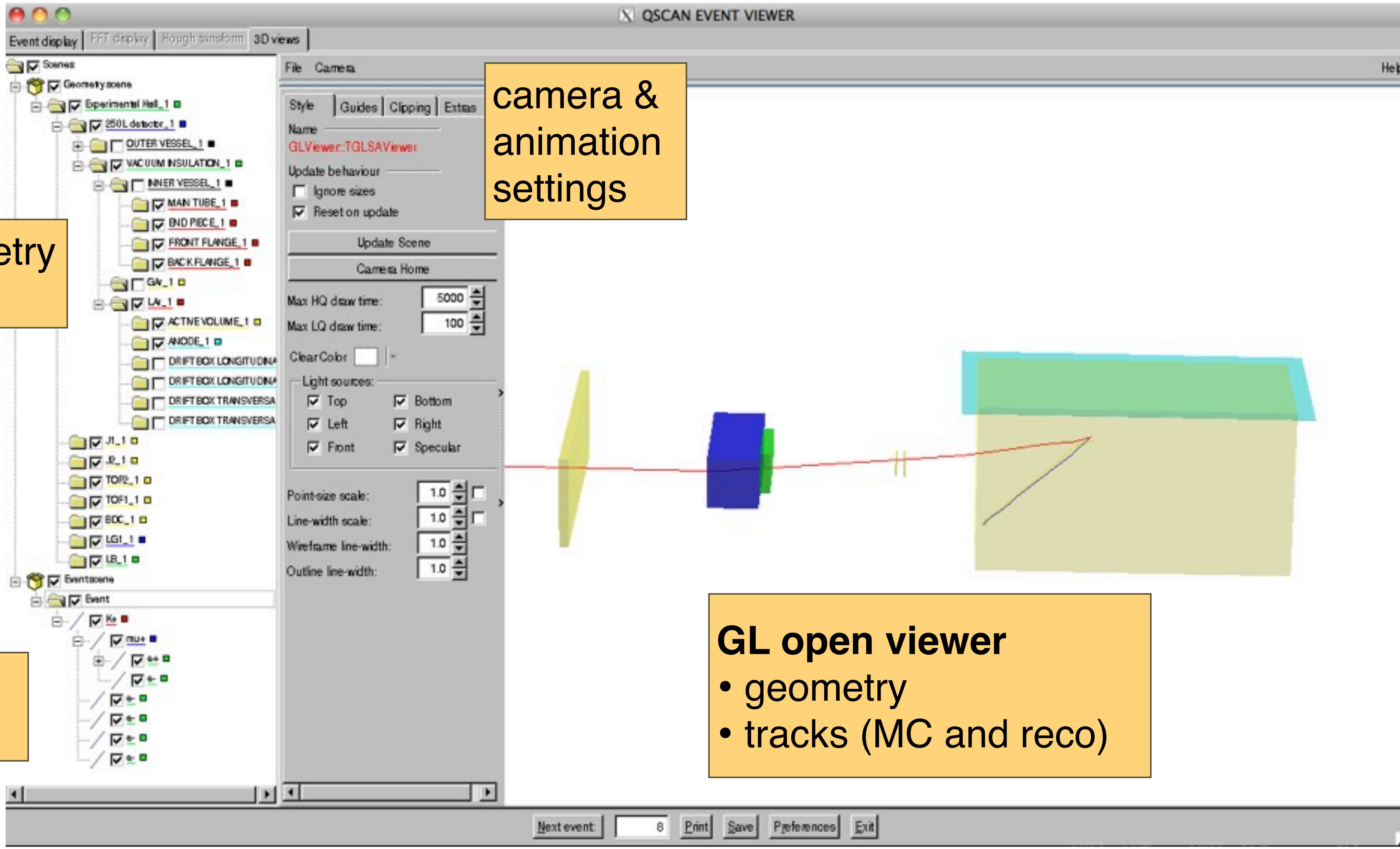
drift range slider

2D event display (raw data)

zoom in/out

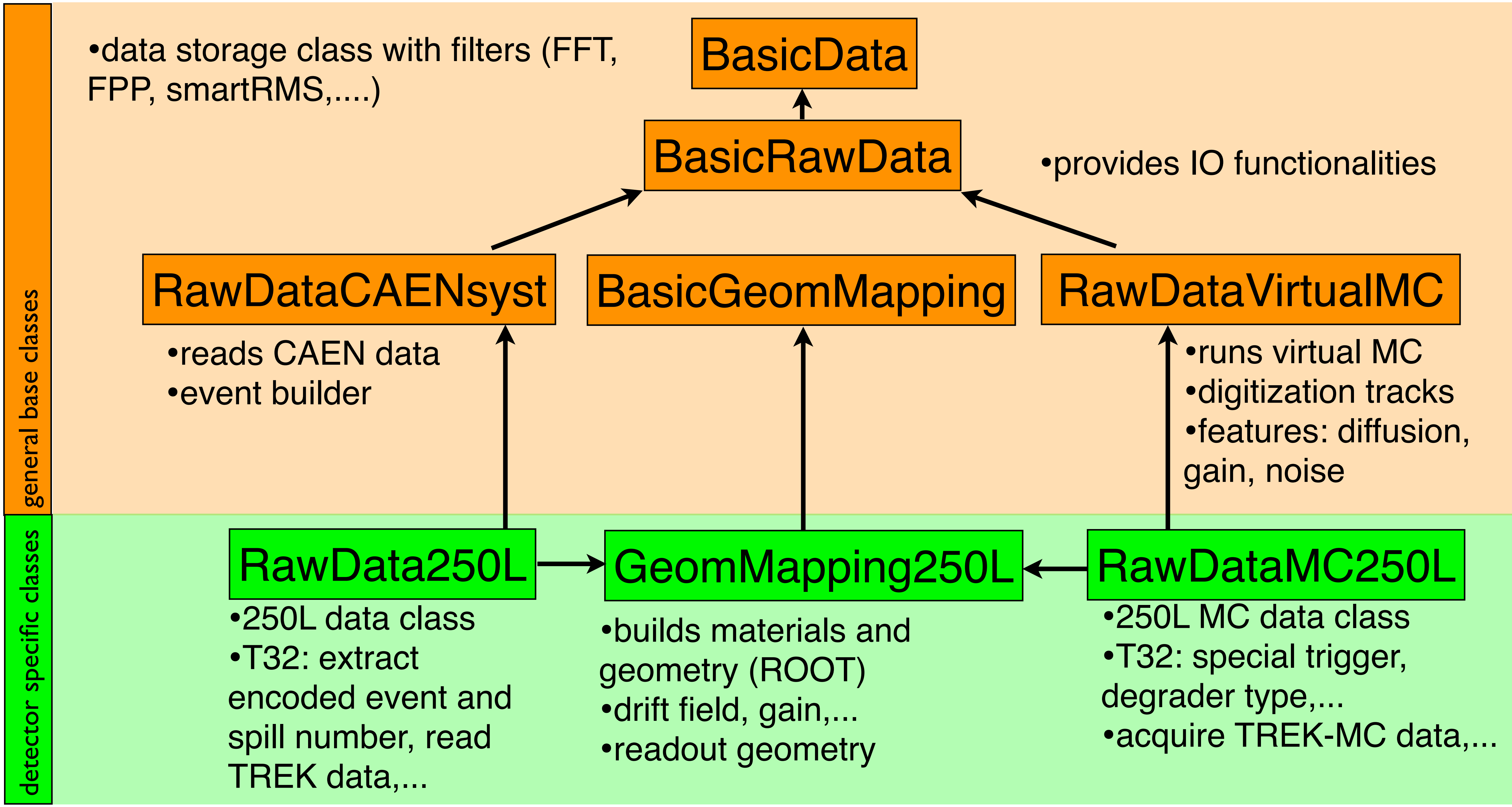
event control, Print, Save, Preferences, exit

The Graphical User Interface (II)



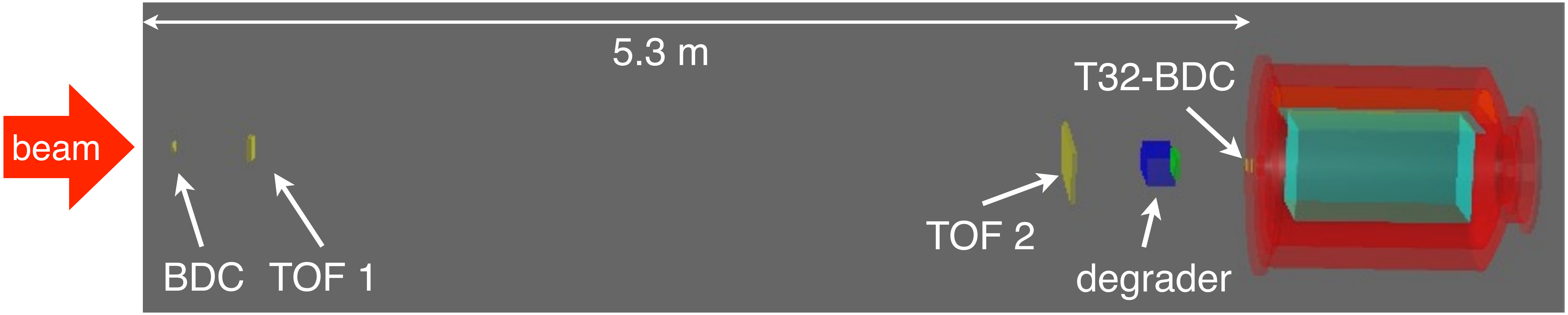
➡Based on the ROOT TEve package (see $ROOTSYS/tutorials/eve/...)

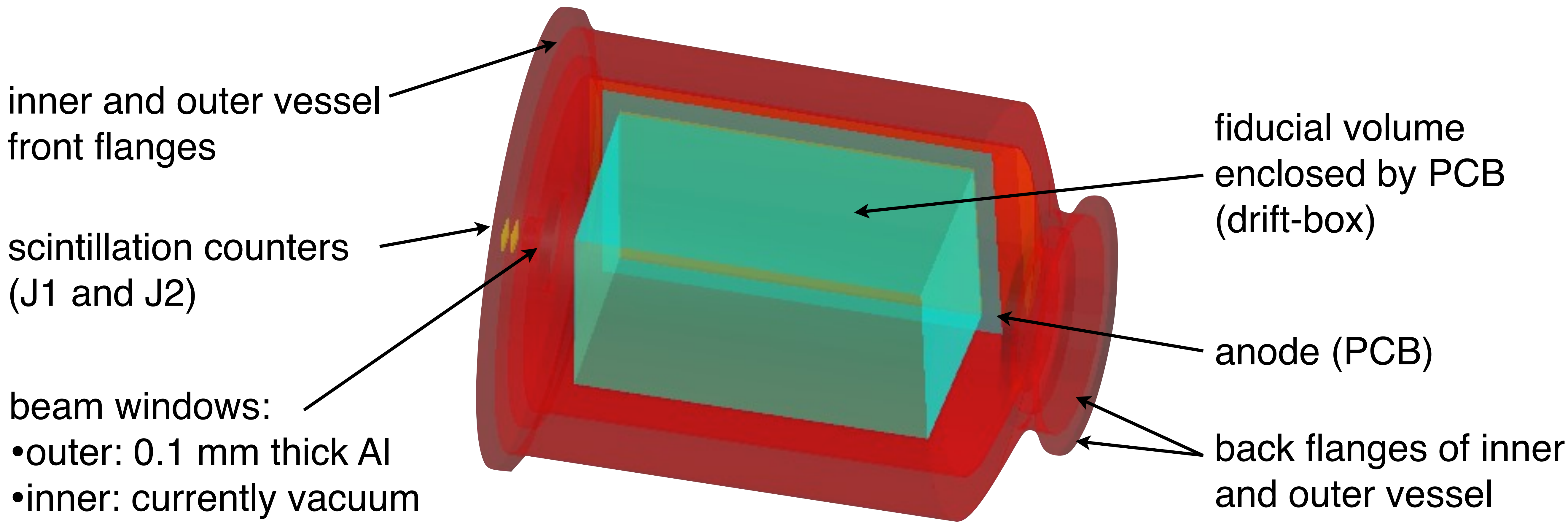
Geometry and (MC-) data



remark: The ROOT geometry (TGeo) defined in the geometry class of each detector is used by the MC event generator as well as by the data and reconstruction classes

The ROOT geometry of T32





The virtual MC simulation for T32 (I)

The ROOT VMC package allows to track particles in the created ROOT geometry using Geant (currently Geant3)

T32: In a first phase the T32 test beam experiment was using a single view readout in LAr (no gain) with a coarse pitch of 1 cm (see T. Maruyama's talk)
➡ MC simulation is needed to check agreement with data

Requirements from the T32 test beam experiment:

• varying beam momentum and particle composition (K/pi ratio adjusted,...)
• different degrader configurations (single LG, 2LG, LG&LB, no degrader,...)
• changing trigger (K+,e+,with or without CR data,...)
• HV cathode, drift field (inhomogeneous field map), calibration of the electronics,..

run info example

```
name    : PhysicsOct_52
ID      : 444
time    : Sat Sep 24 10:26:52 2061
---------------------------------------
trigger  = beam-kaons (J2+FC+BDC+TOF)
degrader = 1 lead glass and 1 lead brick after TOF2
HV cathode = -9000 V
number of events = 3189
beam momentum = 800
beam / cosmics ratio = 1 / 0
beam particles (e+|pi+|K+|p|others) = (0|0|0.98|0|0.02)
comment = -
```

The virtual MC simulation for T32 (II)

General VMC procedure (with trigger requirement):

1. Open a new file with option: “run:444” (run specific) or “pdg:K+,0.8” (general, arbitrary primary particle)

2. Define materials and detector geometry

3. Load the next event:
   1. Define the primary particle (T32 case: simulate proper beam composition)
   2. run the virtual MC
   3. Evaluate the MC tracks (fill structures, build trigger condition);
   4. Check trigger (T32-case: energy deposit in various upstream scintillation counters):
      - **no:** redefine the primary particle,...
      - **yes:** Acquire all the tracks in the fiducial volume

4. Digitize MC tracks (digitization, add noise, finalize event)

5. Save tracks, structures (T32 case: energy deposit, stopping point,...)

The virtual MC simulation for T32 (IV)

digitization of all MC tracks in the fiducial volume:

1. MC tracks contain the particle information and an array of points **(x,y,z)** as well as the corresponding deposited energy and the step-size **(edep,step)**

2. With a given electric field the recombination factor can be computed
   ➡ charge deposition **dQ** for each point **(x,y,z)**

3. The charge is drifted to the anode
   • more general: field map can be used if required
   • diffusion in LAr, purity losses, gain, diffusion in GAr,....

4. The charge induces signals with a known shape on all readout views.
   • mapping information (pitch, angle of strips,...) from GeomMapping250L
   • charge deposition is convoluted with the correct preamplifier response
   ➡ I(t) --> V(t) for each step, view and strip/wire.

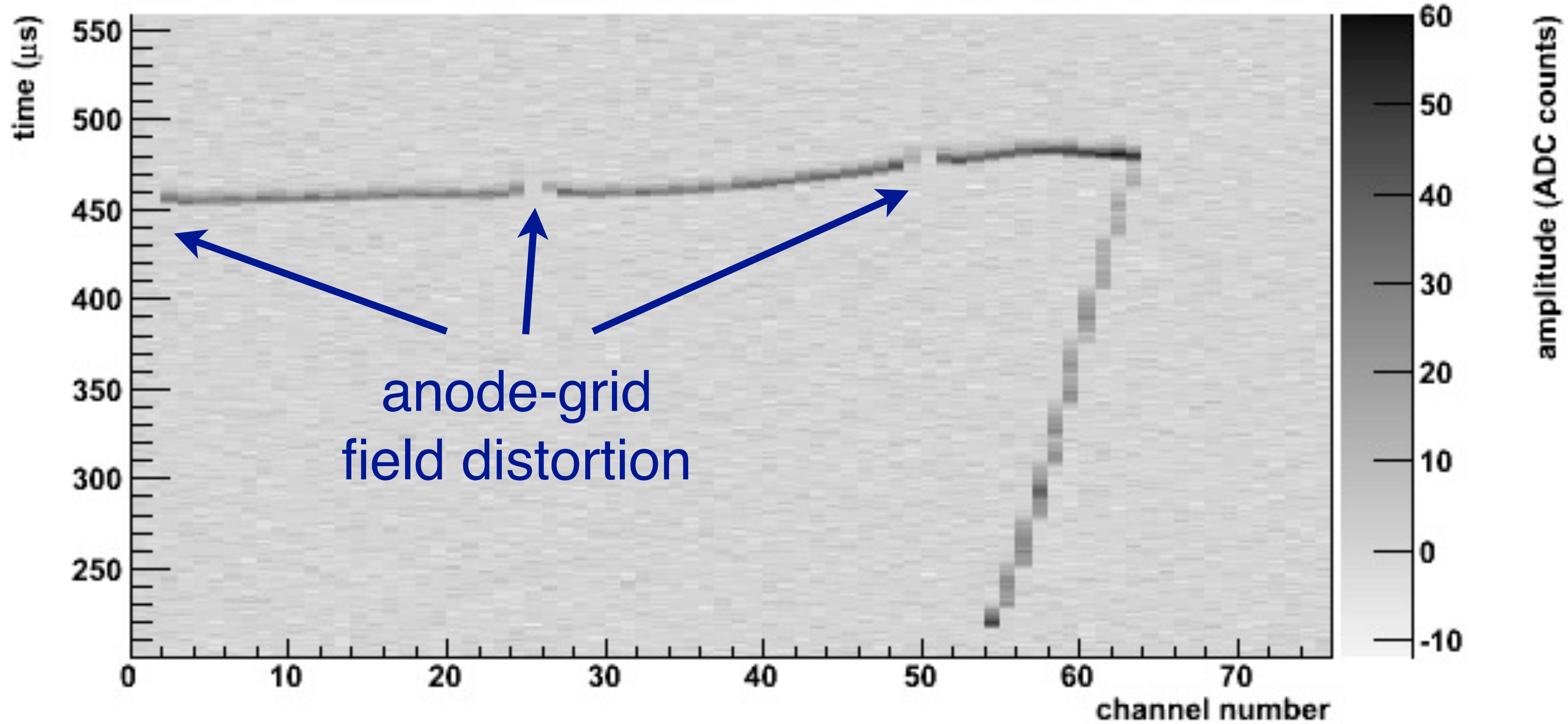
5. Finally noise can be added (currently only white noise is used)

The virtual MC simulation for T32 (V)

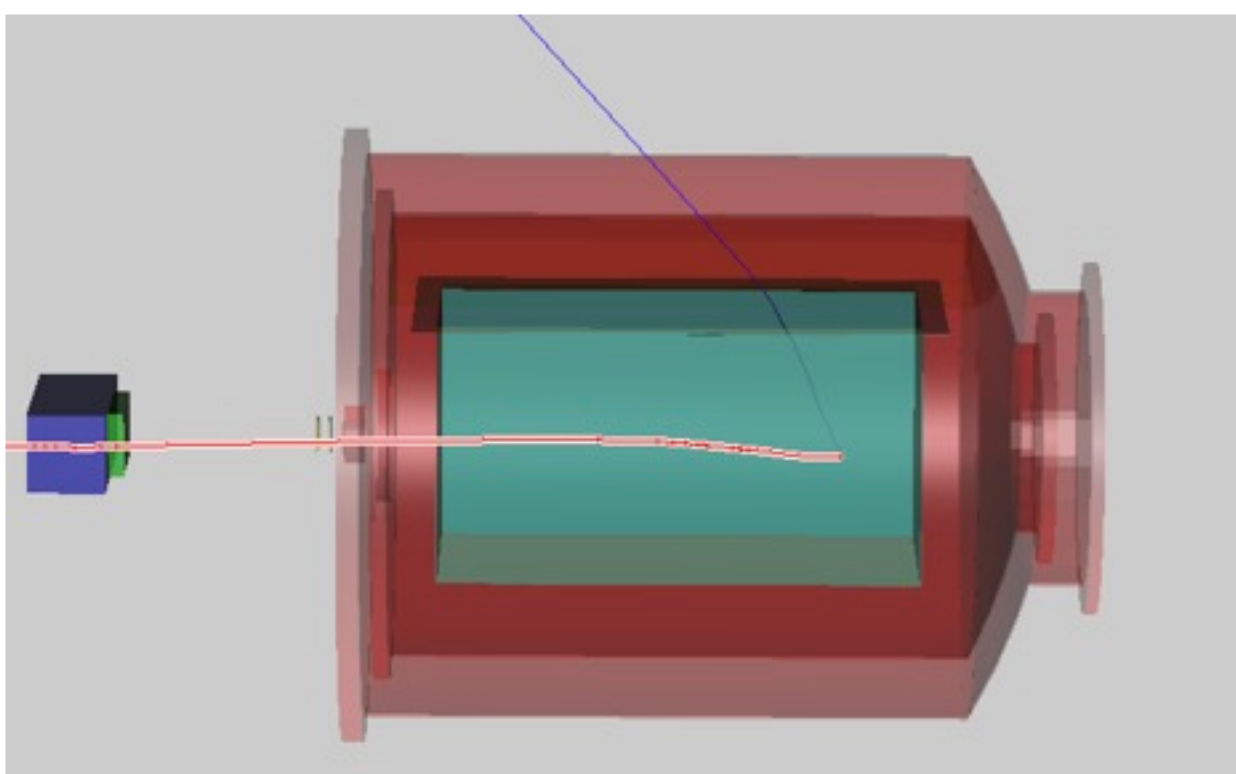
example event

- $K^+ \rightarrow \mu^+ \nu$ decay
- inhomogeneous drift field included (external finite element computation)
- realistic trigger and degrader configuration

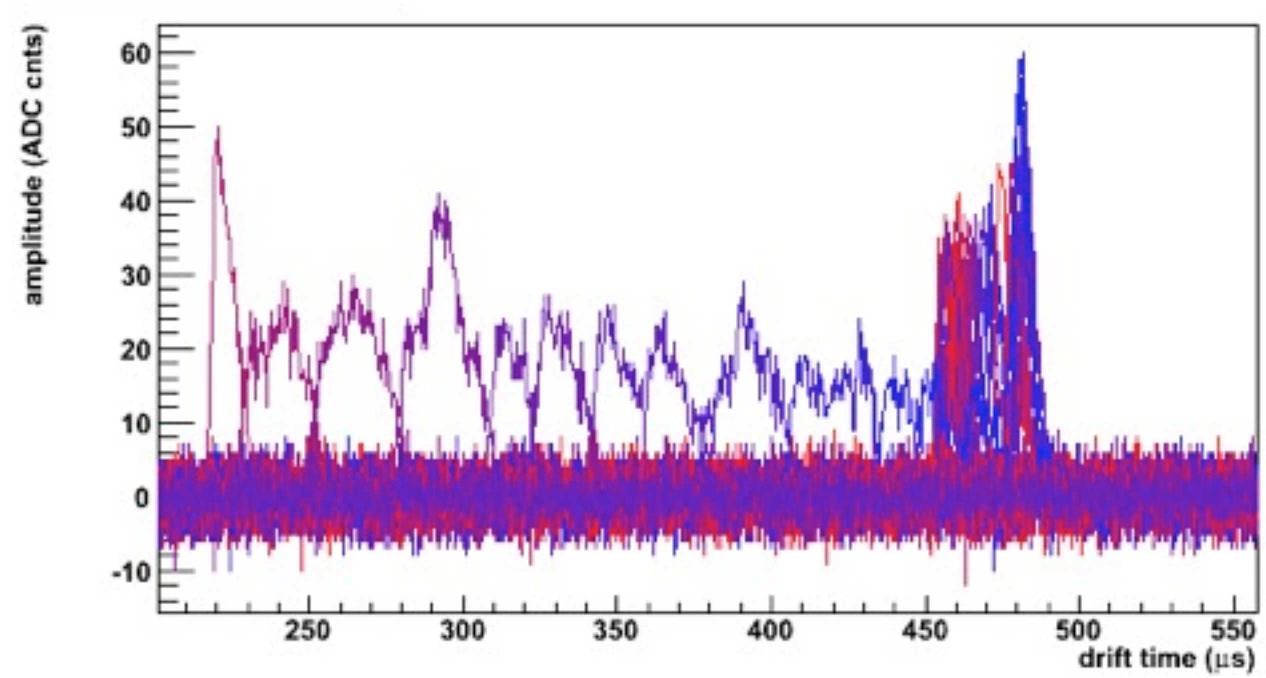
event display



3D display with MC tracks



waveforms



Event reconstruction

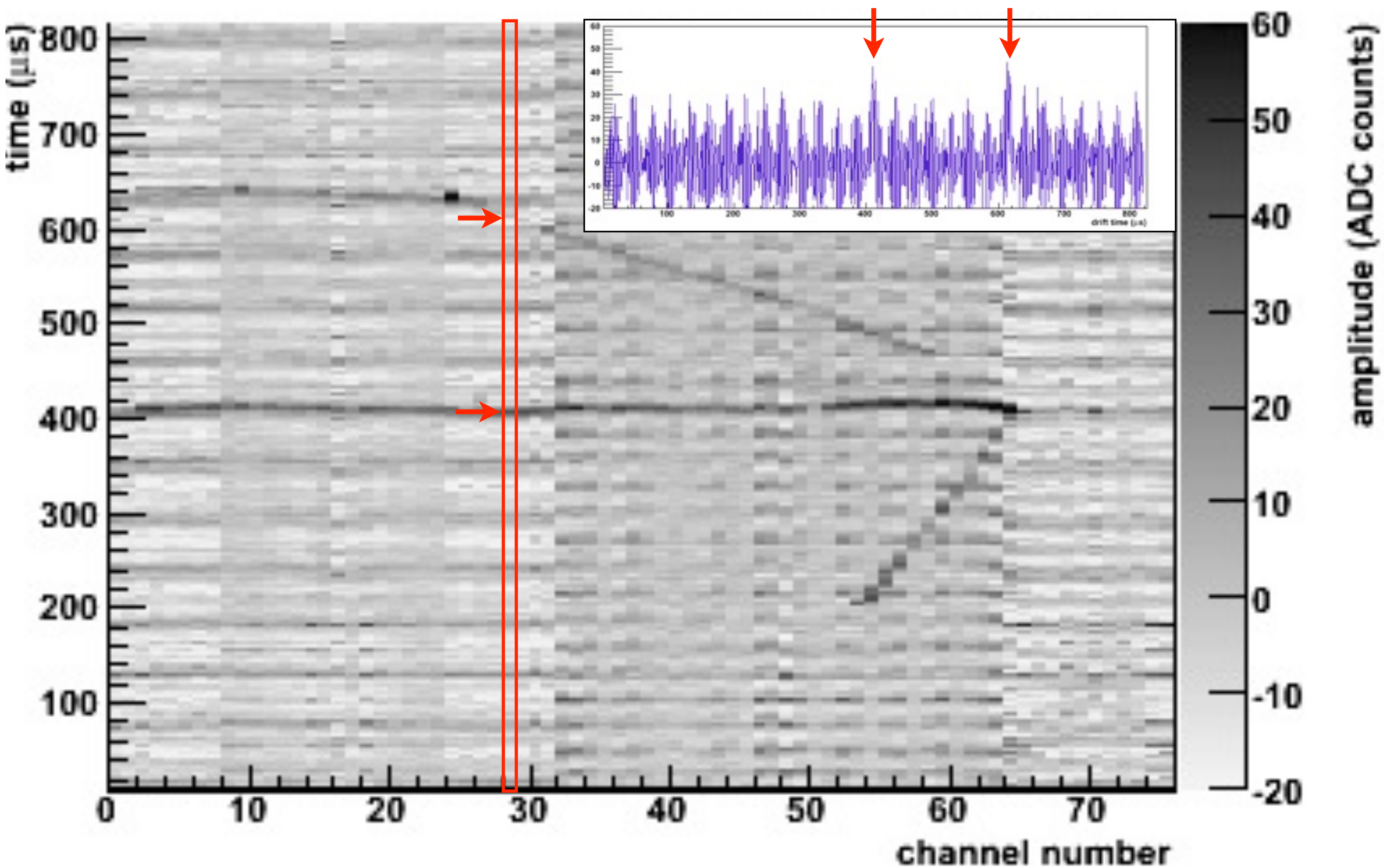
the “classic” procedure

1. Noise reduction
2. Pedestal subtraction
3. Hit finding, hit reconstruction, fitting
4. Clustering hits
5. 2D Track reconstruction (delta ray finder, multiple scattering)
6. 3D Track reconstruction (dE/dx, PID)

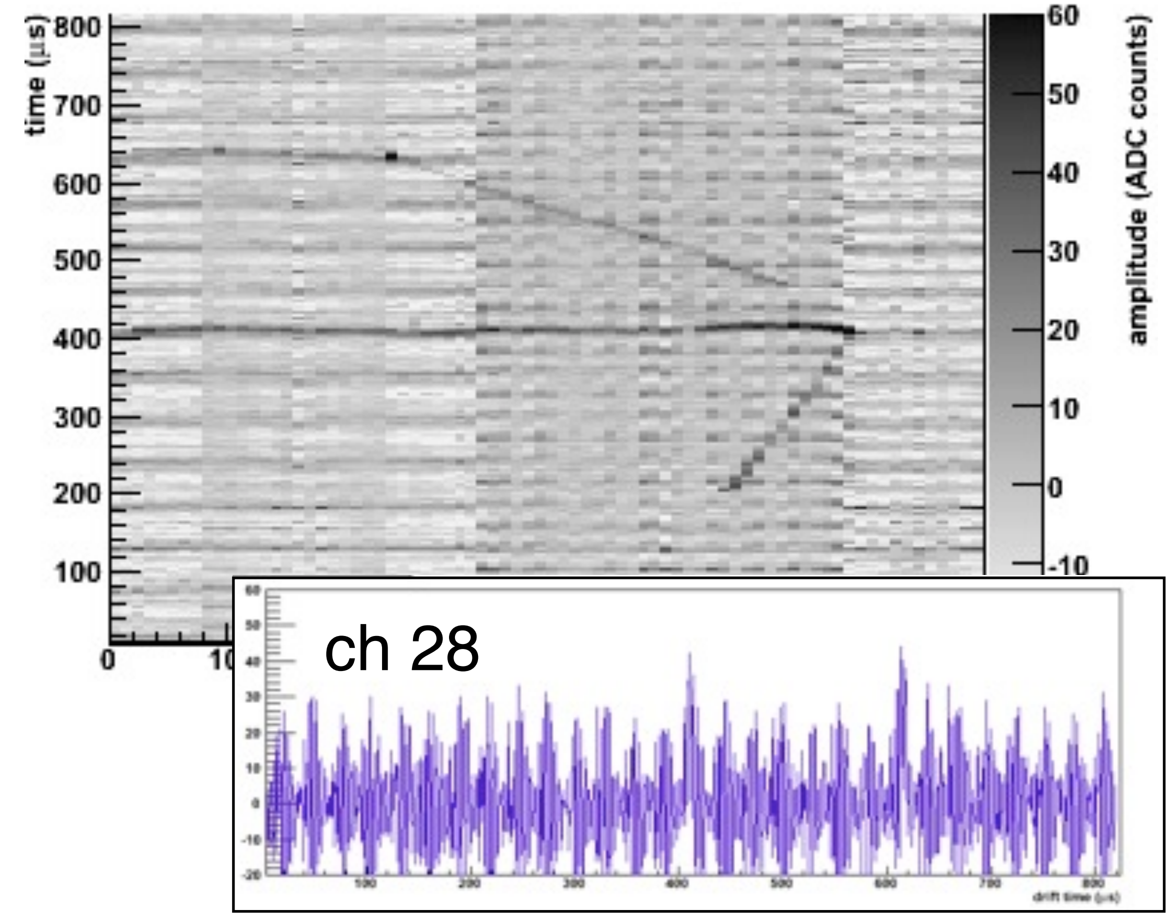
ref.:

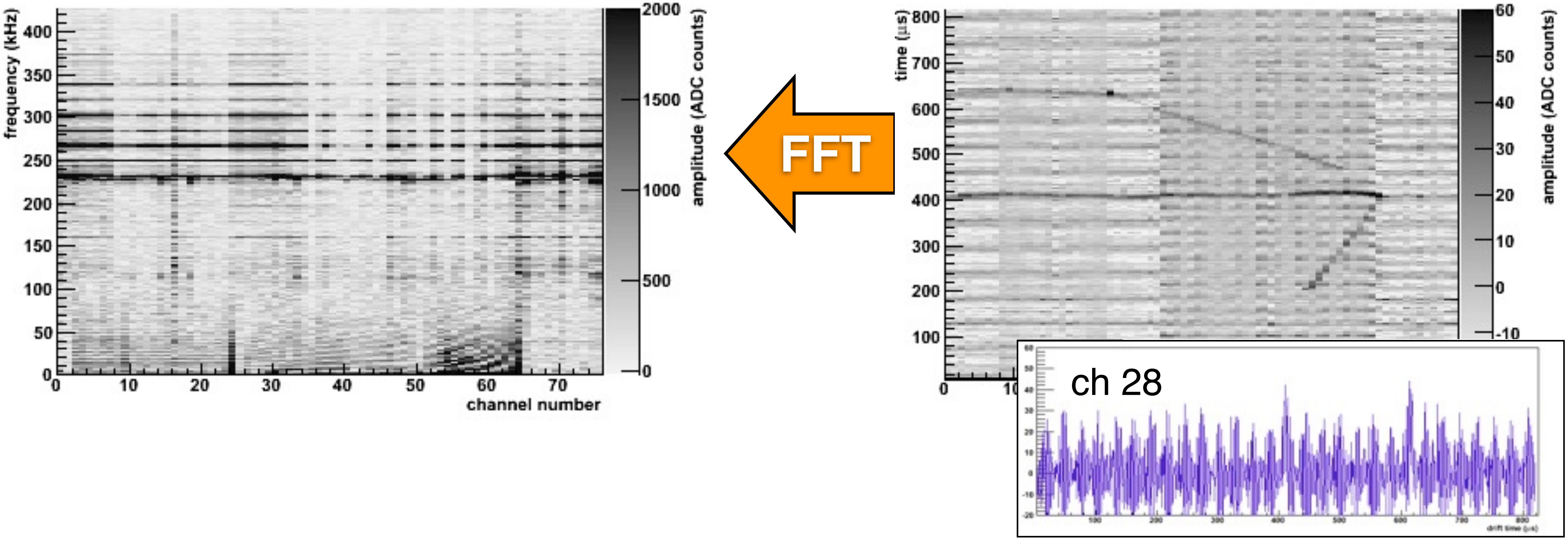
- J. Rico, “First study of the stopping muon sample with the ICARUS T600 detector”, PhD thesis (2002), ETHZ
- A. Martinez, “Study of accelerator neutrino interactions in a liquid argon TPC”, PhD thesis (2007), Universidad de Granada

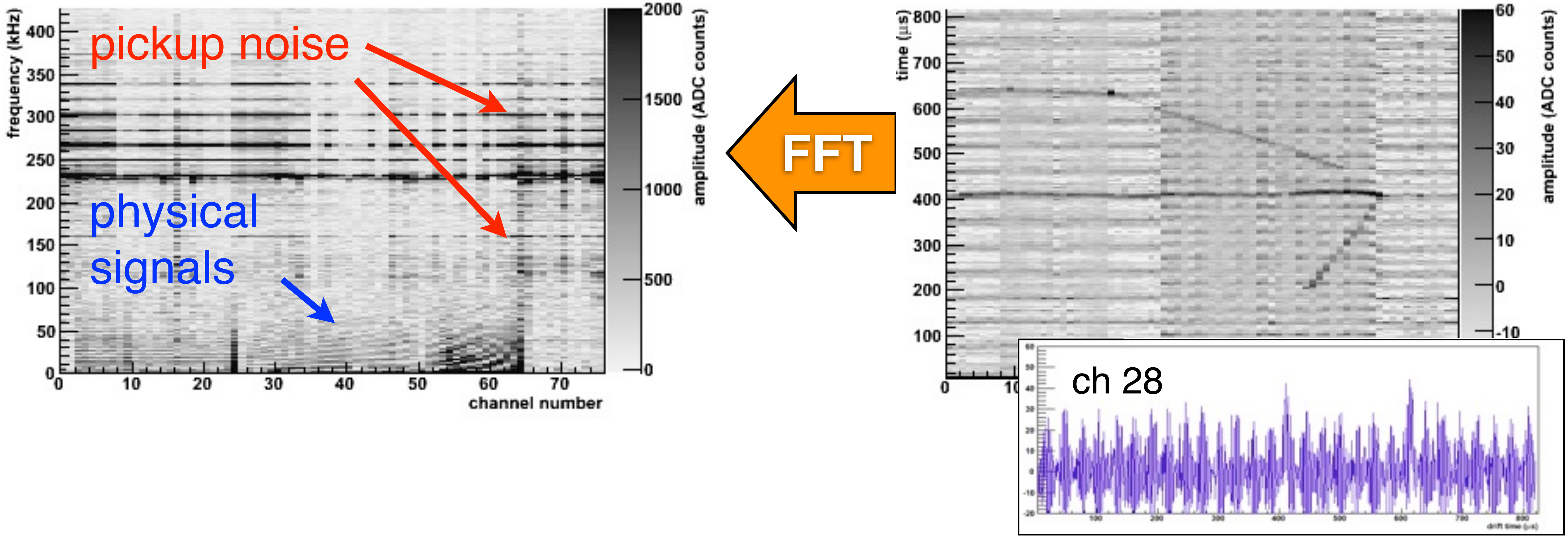
Noise reduction - T32 (250L)

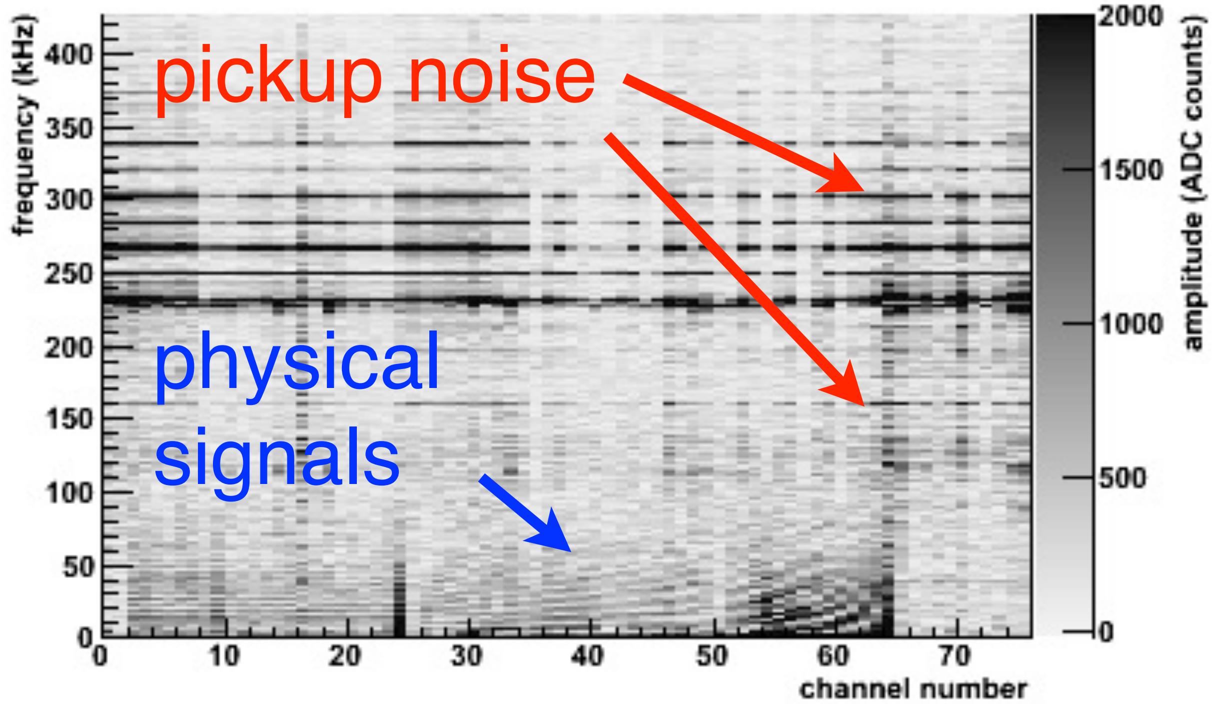


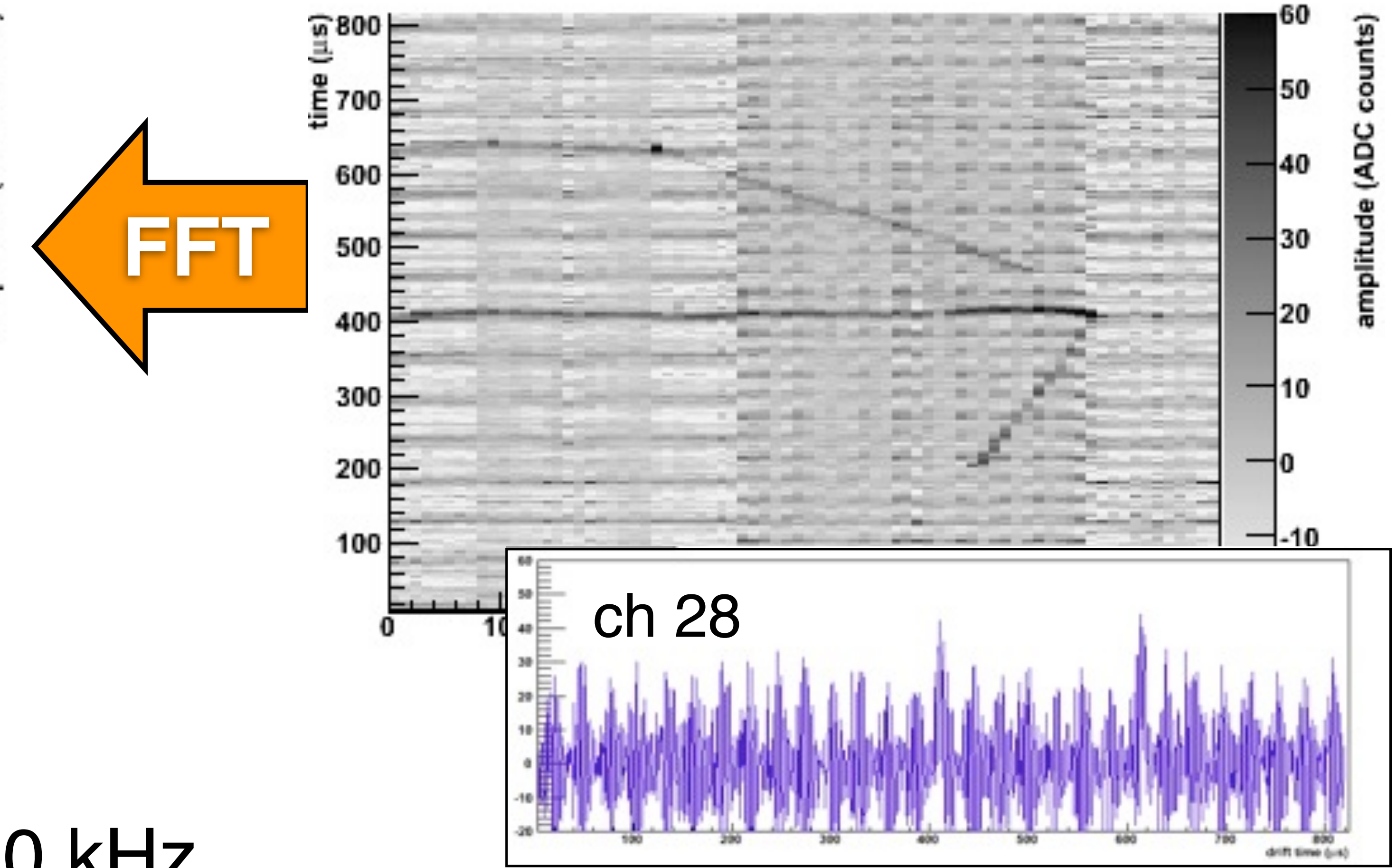
Raw, noisy event from T32: two decaying K+


time (µs)
amplitude (ADC counts)
ch 28
drift time (µs)

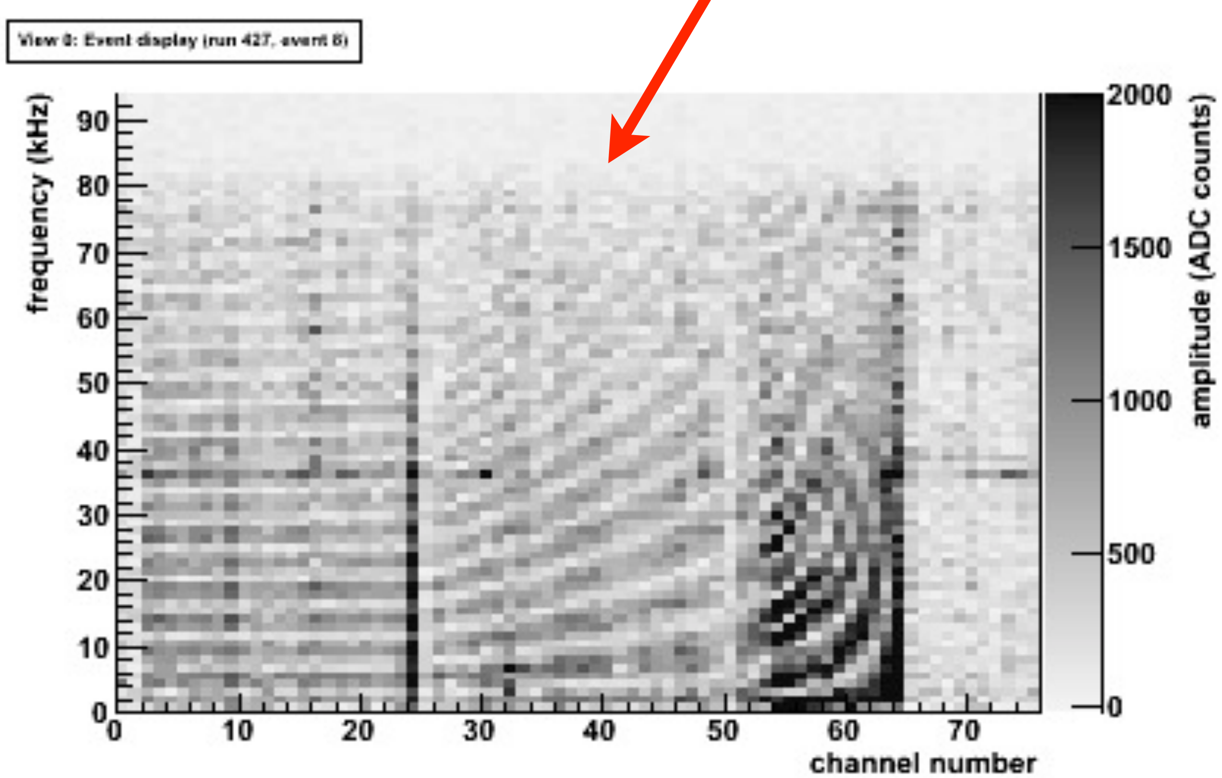

frequency (kHz)
channel number
amplitude (ADC counts)
FFT
time (μs)
amplitude (ADC counts)
ch 28

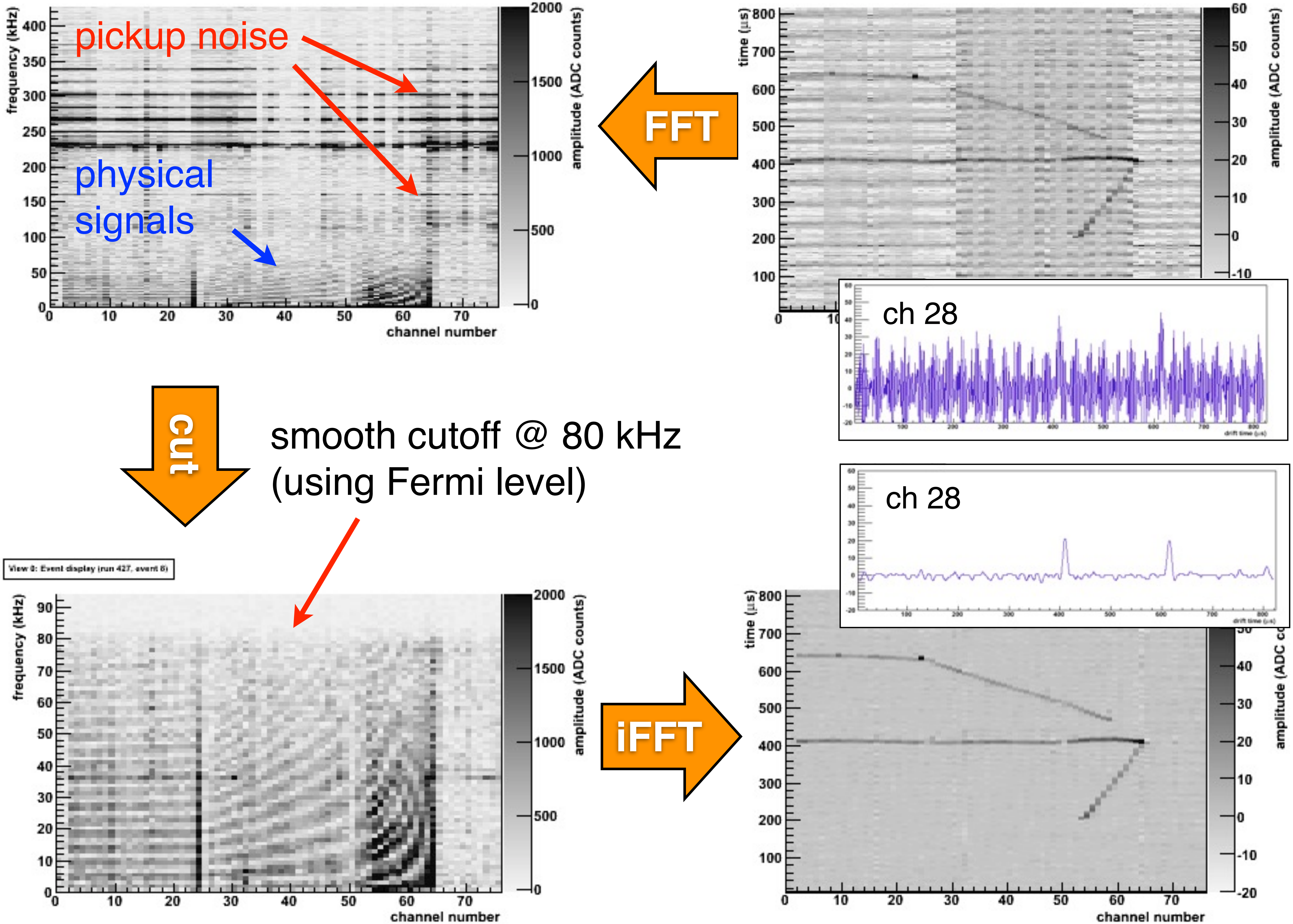

pickup noise
physical signals
frequency (kHz)
channel number
amplitude (ADC counts)
FFT
time (μs)
amplitude (ADC counts)
ch 28





smooth cutoff @ 80 kHz
(using Fermi level)

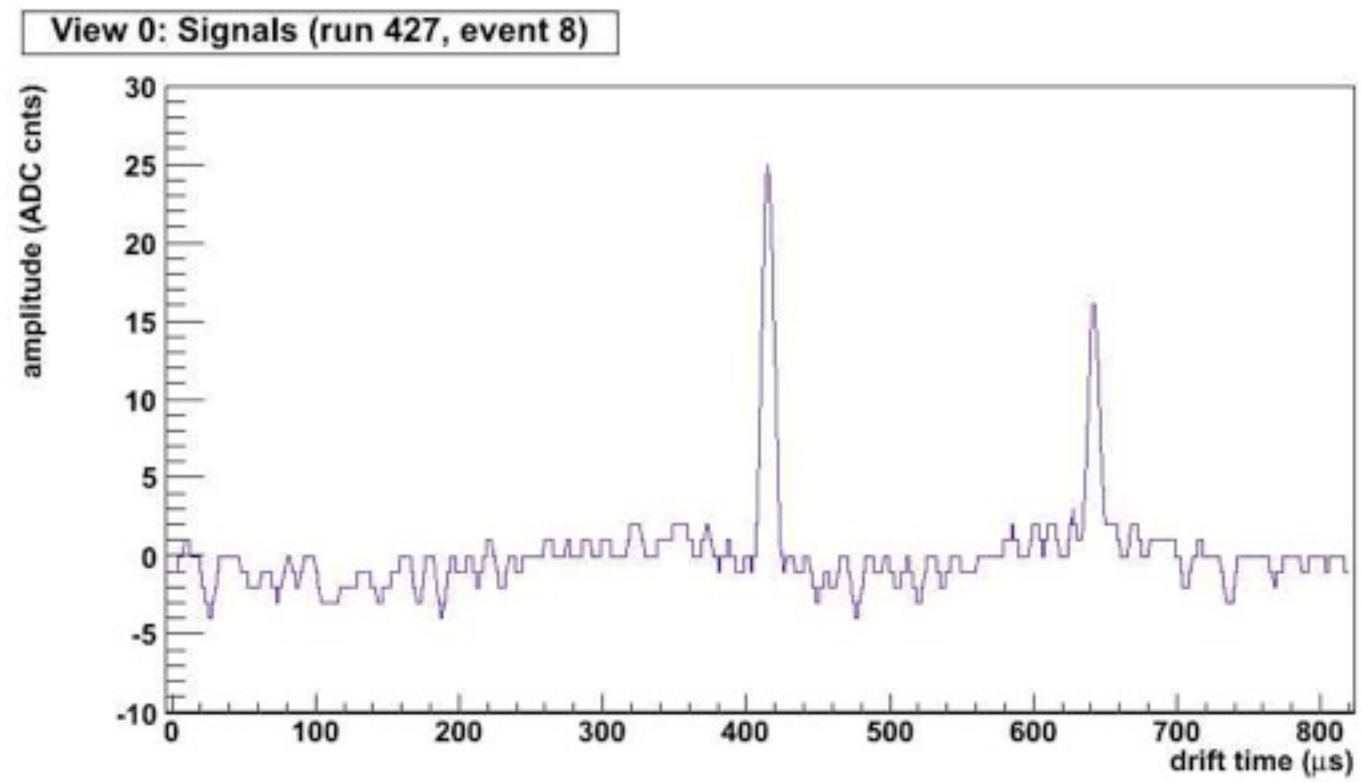



pickup noise
physical
signals
frequency (kHz)
channel number
amplitude (ADC counts)
FFT
time (µs)
ch 28
drift time (µs)
cut
smooth cutoff @ 80 kHz
(using Fermi level)
View 0: Event display (run 427, event 8)
iFFT

Pedestal subtraction

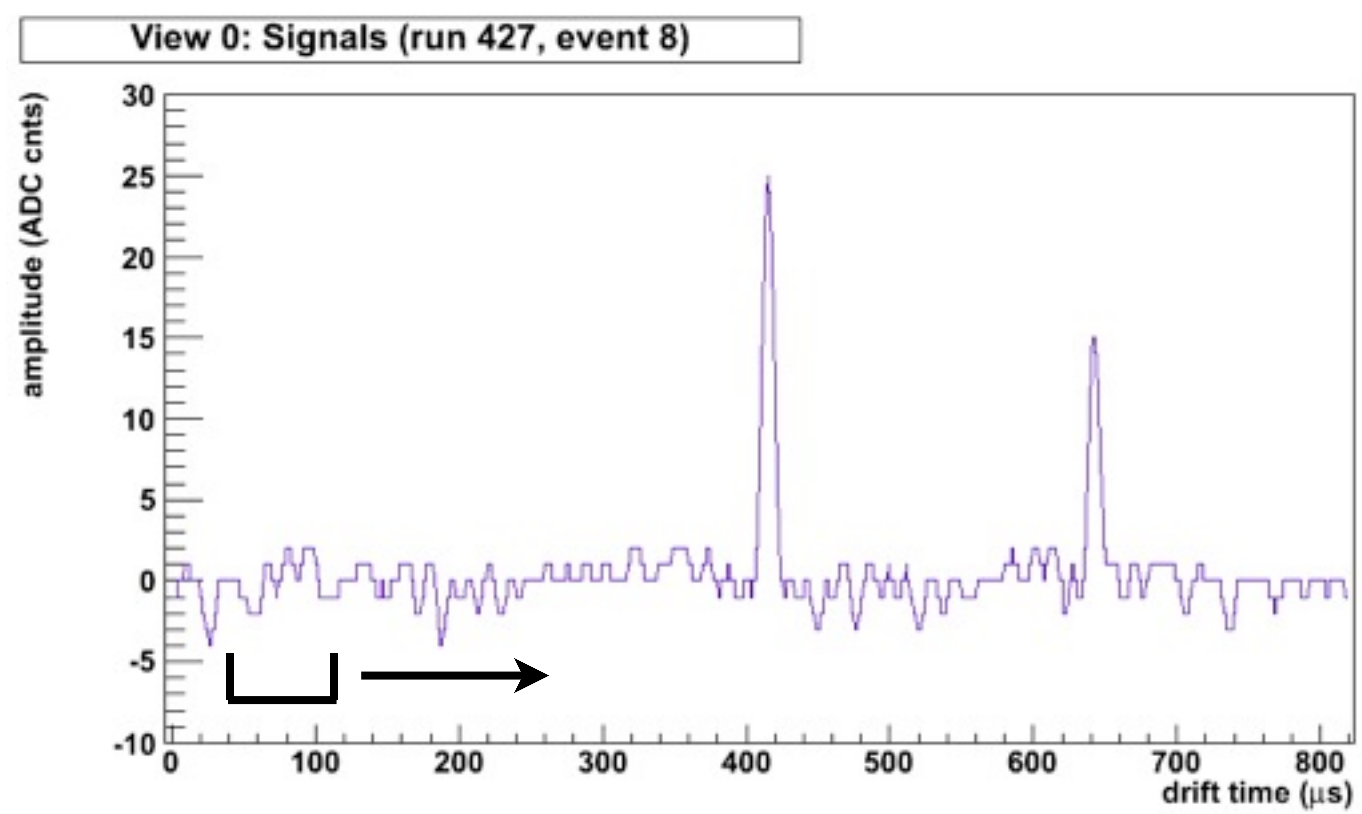
SmartRMS-filter
- iterative process:
  1. the algorithm computes the mean and RMS value of all samples.
  2. recompute the mean value, rejecting samples outside 3σ
  3. repeat step 2.

➡ biased method.
- not evolving with time



FPP-filter
- computes and subtracts MP value in a finite window

➡ unbiased method.
- as the window moves, low frequency fluctuations are subtracted (high-pass filter).



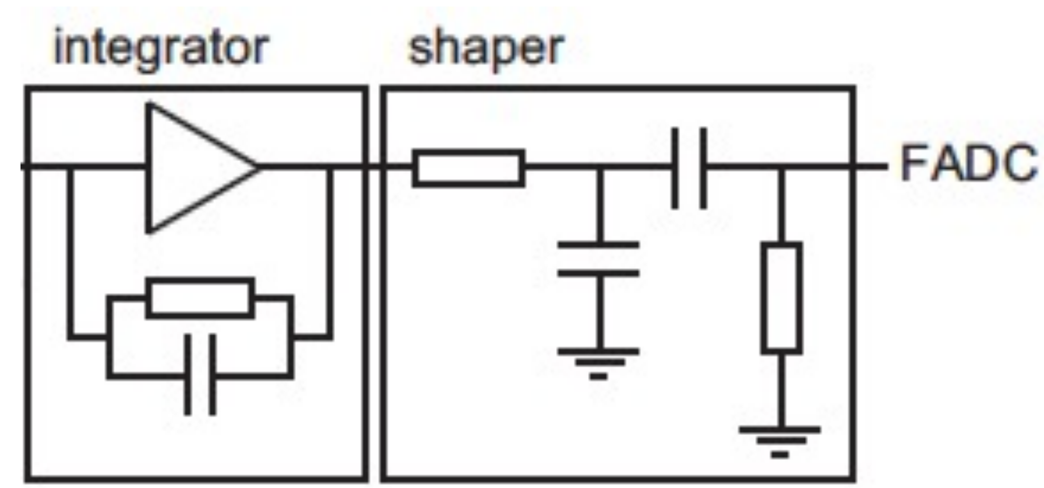
Hit finding and fitting

goal

Discriminate the signals coming from an ionizing particle from electronic noise

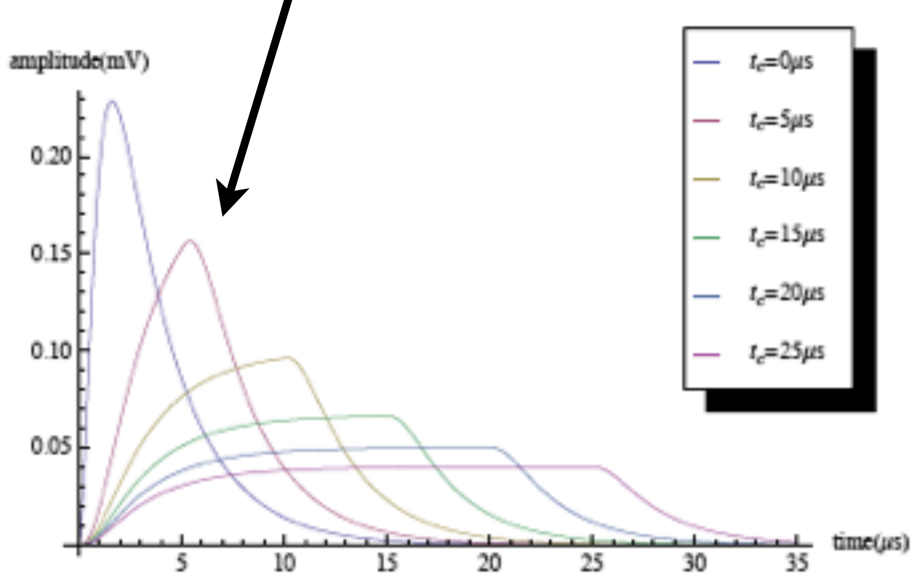
procedure

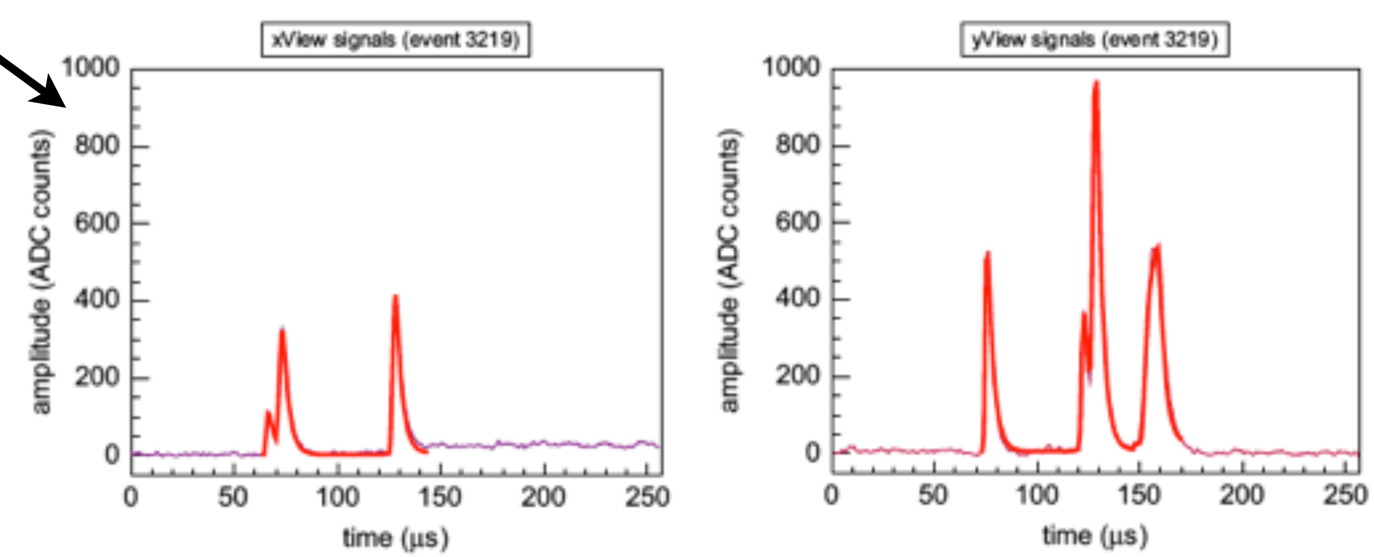
- search algorithm checks amplitude, width and fall time of any hit candidate
- 2nd. step: hits are further parametrized to extract the important physical information (integral, time, shape)



fitting-function for ETHZ preamp

- preamp response well known
- convolution with a square pulse of width Δt_c leads to excellent fit





ref: A.Badertscher et al., NIM A 641 (2011) 48-57

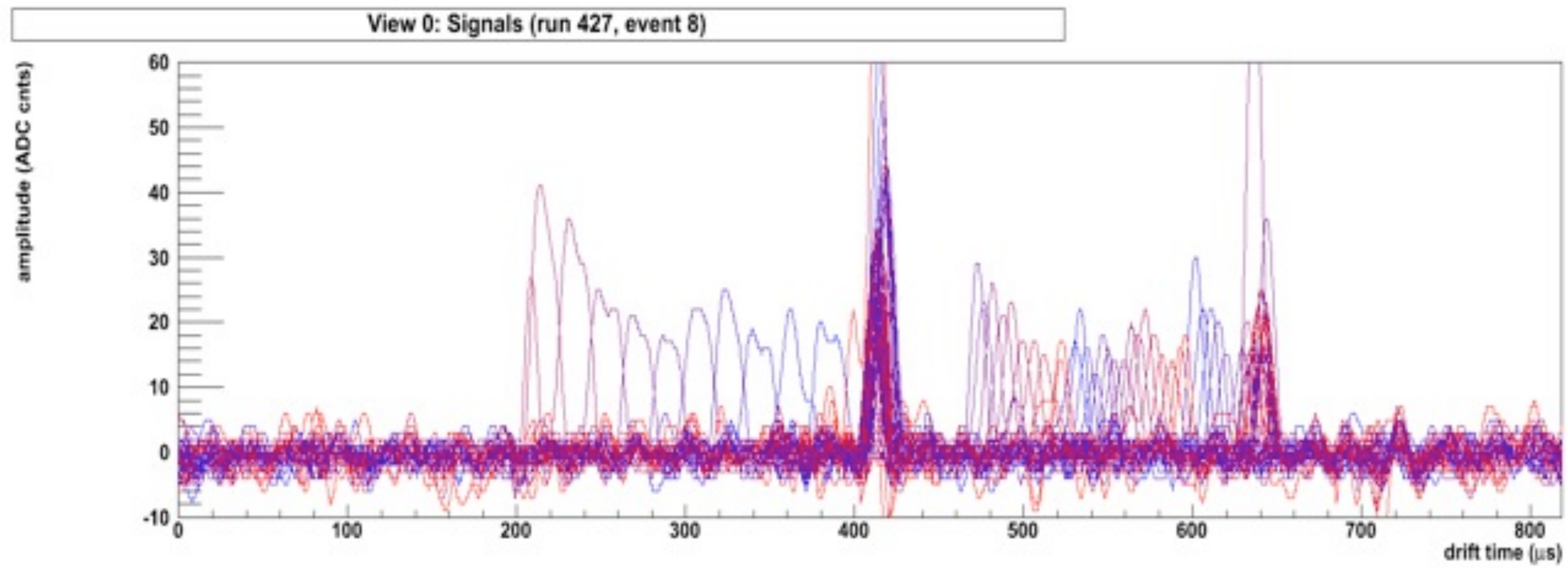
Clustering

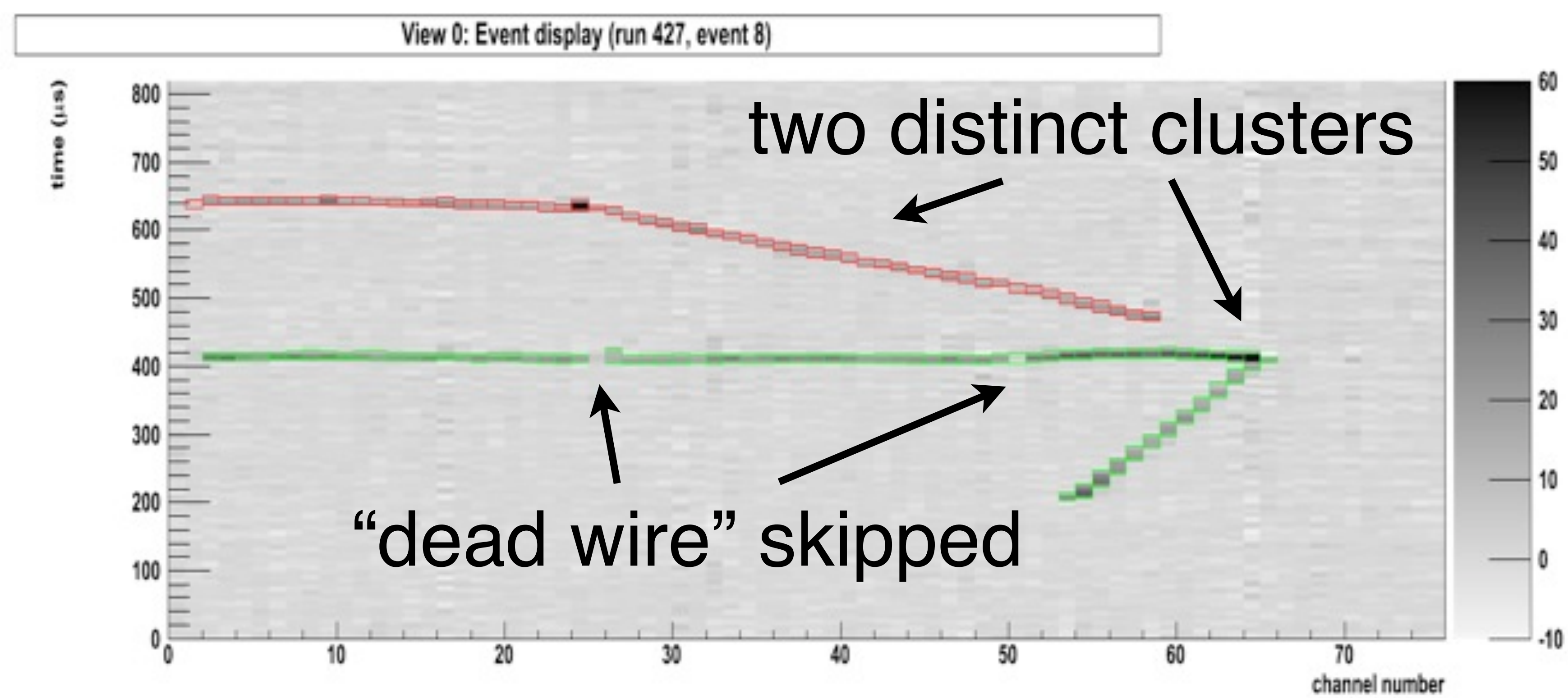
The algorithm groups adjacent hits in the wire-drift plane together.

Procedure

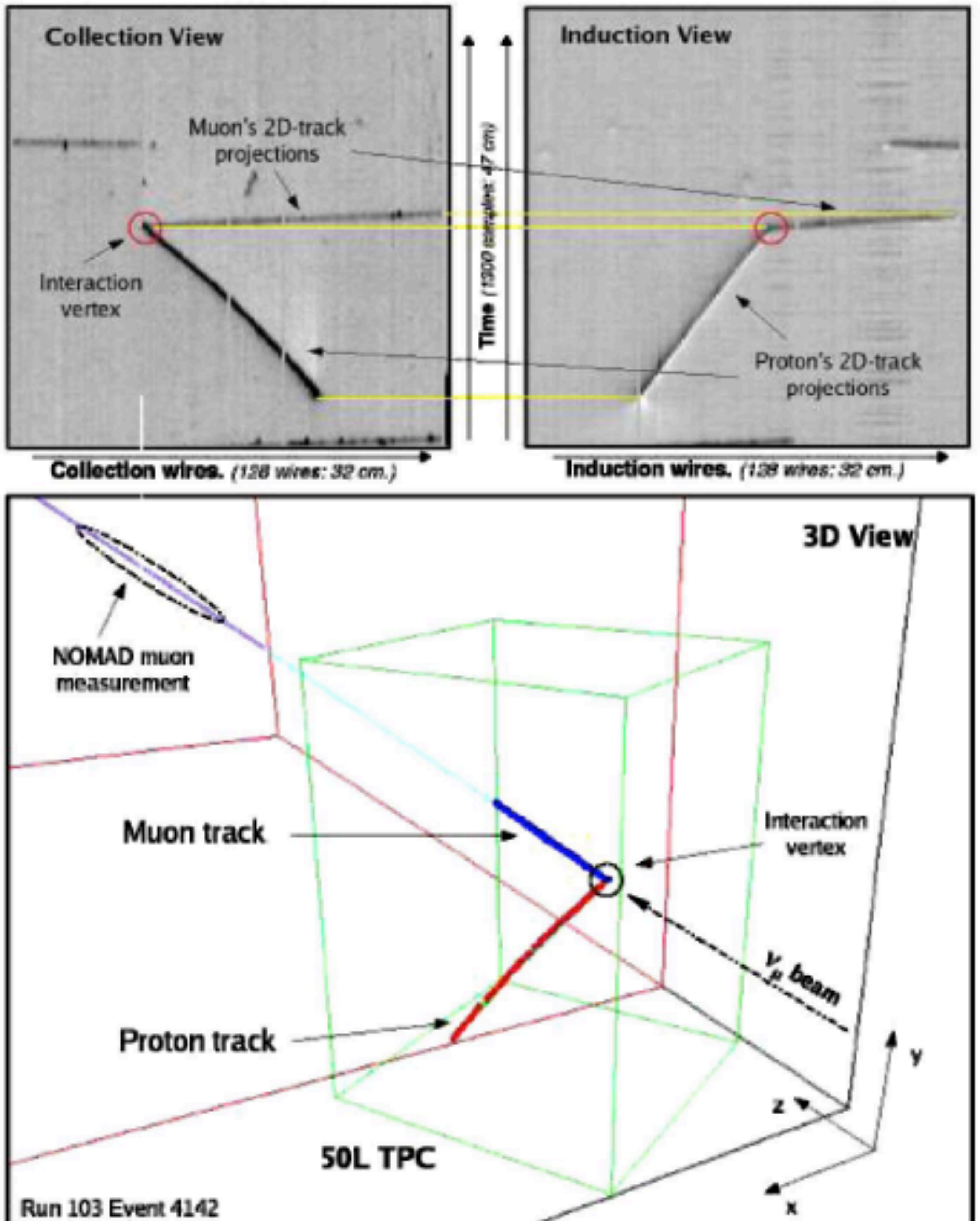
1. Building cluster
- Build links between close hits
- preliminary grouping of hits which are linked together.

2. Finalize cluster
- examine border region for adjacent hits (lower threshold) and add them to the cluster
- update clusters (i.e. merge clusters if necessary)





Track reconstruction - ICARUS 50L



2D tracking

- Divide cluster in long smooth track segments (for each view)
- Reconstruct interaction vertex

3D tracking

- Hits of different views with the same drift time (z-coordinate) are matched
- information from 2Dtracks is used: vertices and end points are matched first

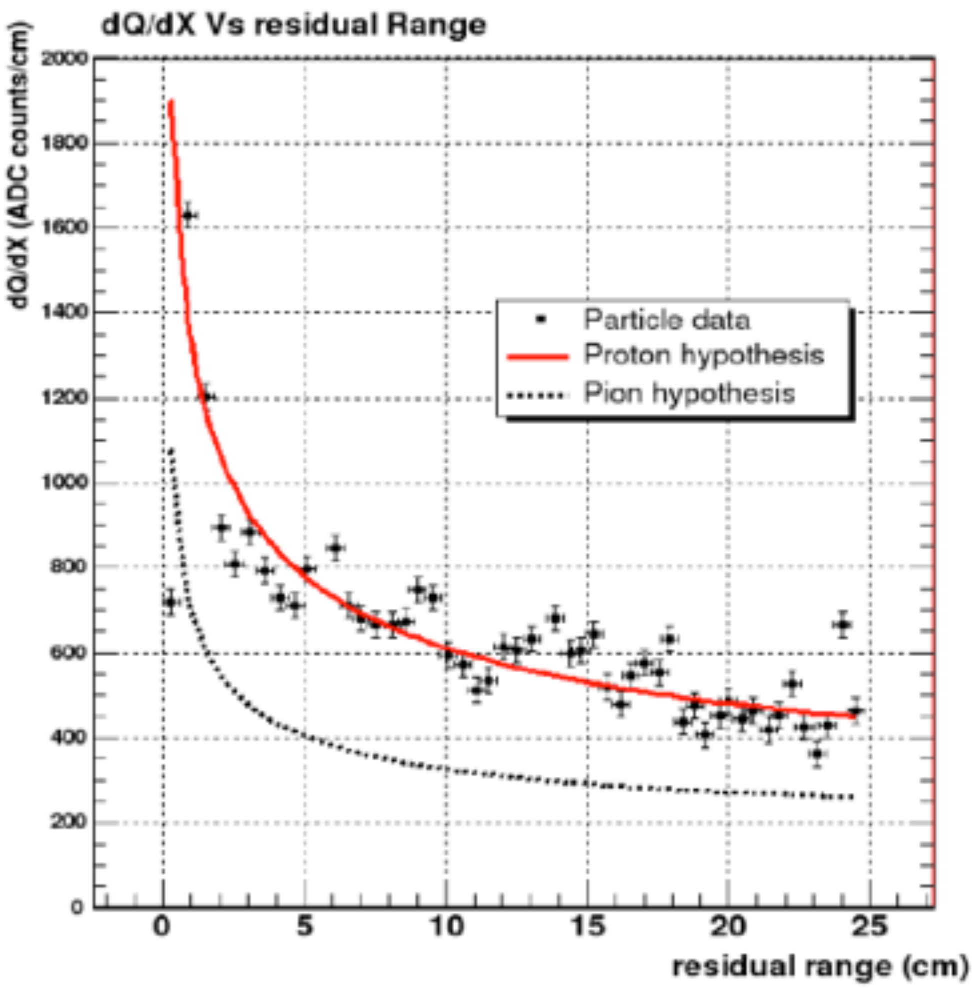
➡ Full information of dE/dx

other algorithms

Kalman filter, delta-ray finder,....

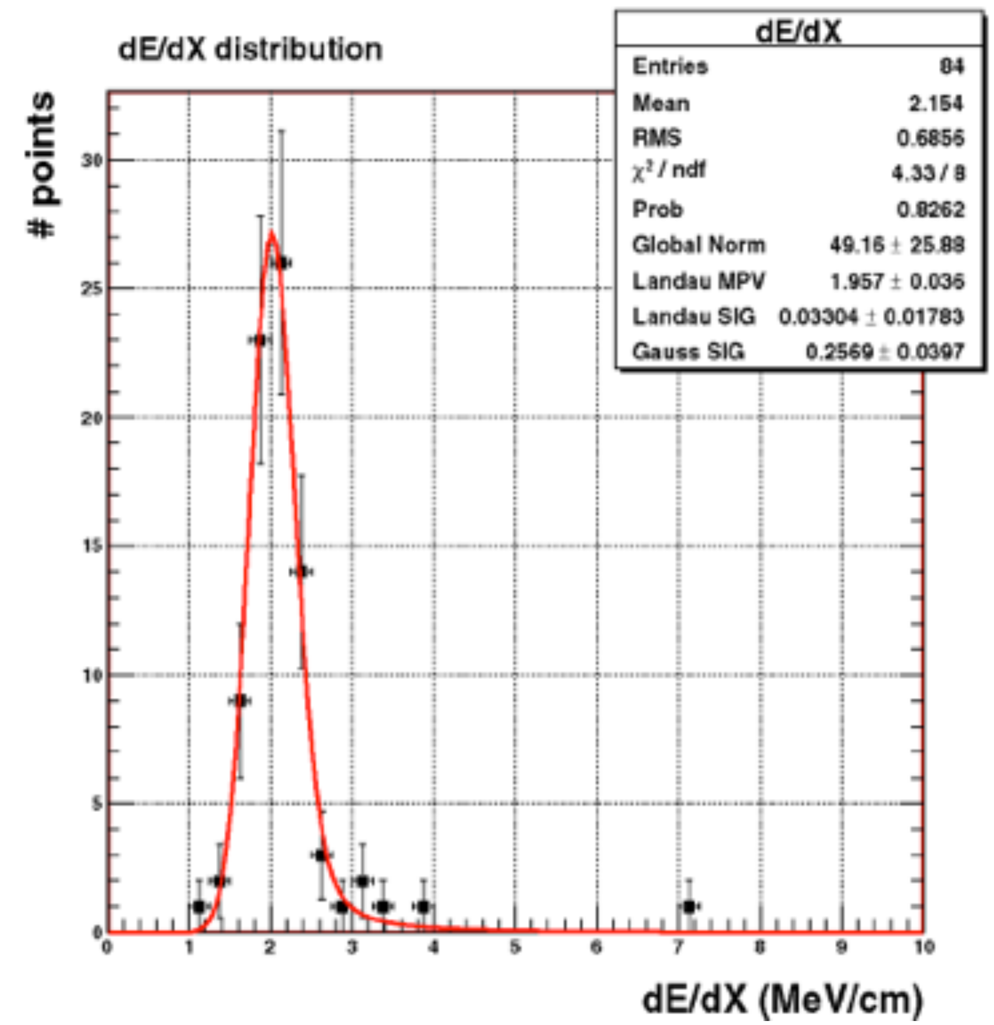
Track reconstruction - ICARUS 50L

dQ/dx proton:



dQ/dx vs residual range for protons and pions

dQ/dx muon:

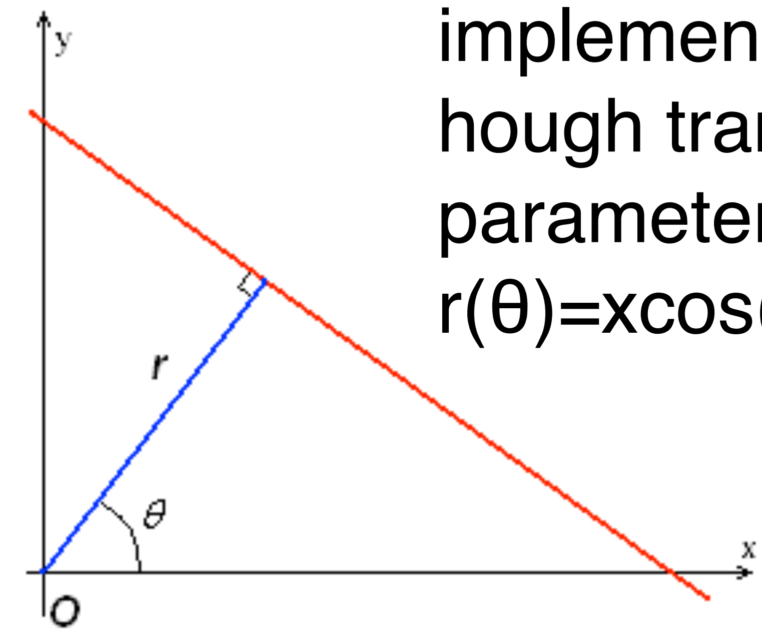


dE/dx of the escaping muon (fitted with a Gauss convoluted Landau distribution):

Reference: F.Arneodo et al., *"Performace of a LAr TPC exposed to the CERN WANF neutrino beam"*, Physical Review D 47 (2006)

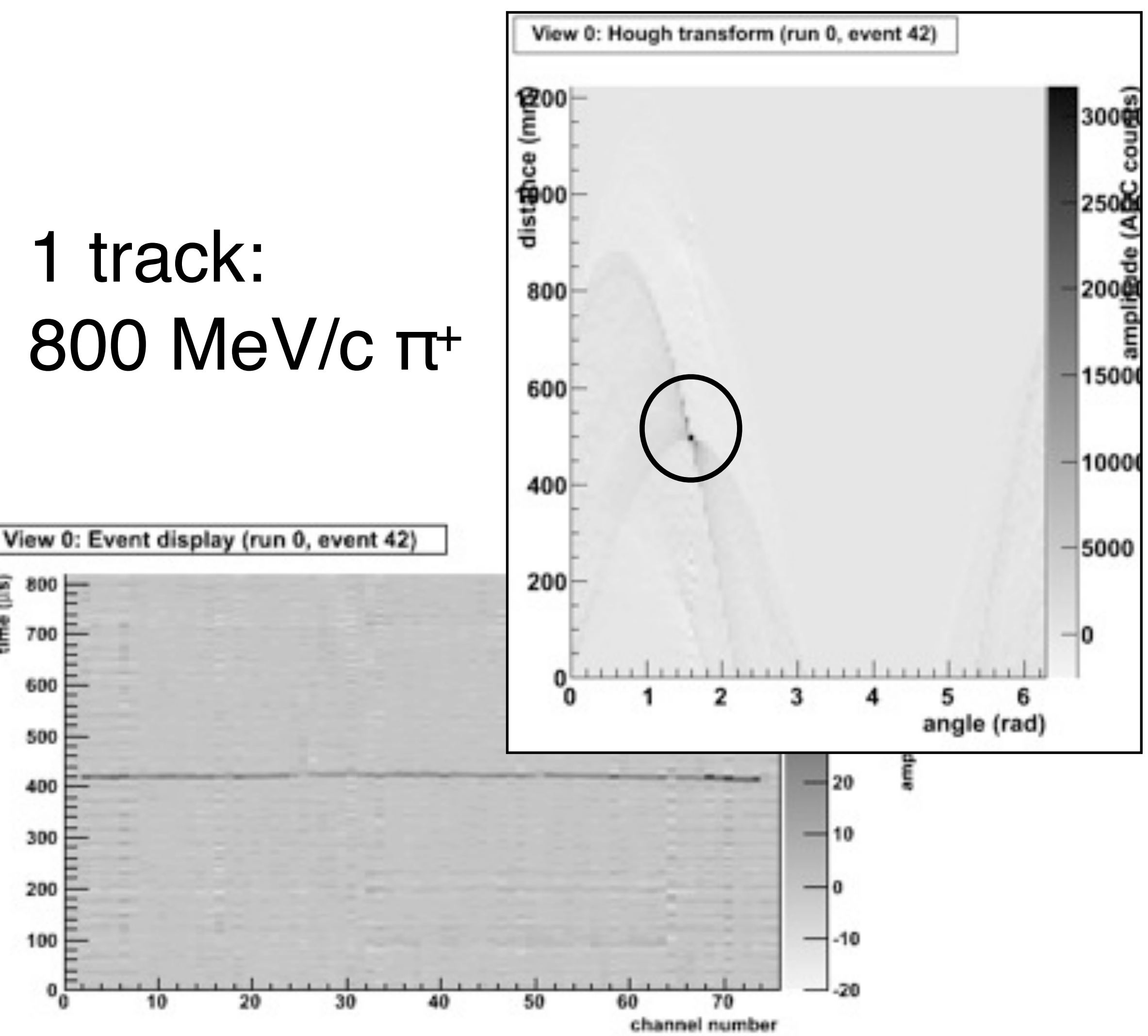
Alternative approach: Hough transform

we are often looking for straight tracks (hits lying on a straight line)
➡ it might make sense to work in the parameter space (Hough transform)

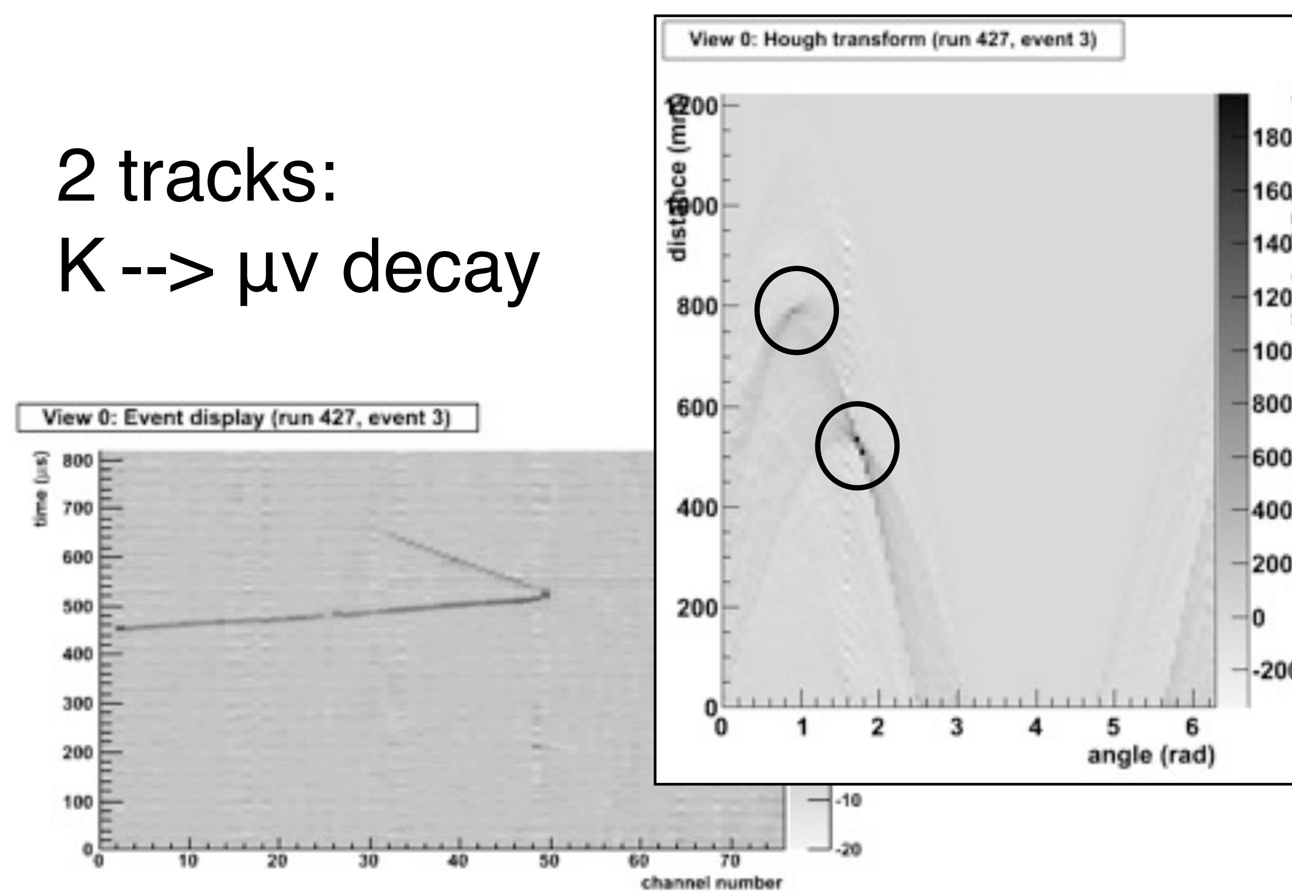


implementation:
hough transform with parameters (r and θ)
$r(\theta)=x\cos(\theta)+y\sin(\theta)$

1 track:
800 MeV/c π^+



2 tracks:
K --> µν decay



Conclusion

- Qscan is a general software package for LAr-TPCs including
  - a MC simulation package (Virtual MC)
  - Event reconstruction tools (noise subtraction, hit finding, clustering, 2Dtracking, 3Dtracking)
  - Viewer to display raw data (2D-display), MC tracks and the reconstruction (3D - display)
- It has successfully been used for analysis of the ICARUS 50L, 10T and T600
- Currently it is being used for the reconstruction of charge particle beam events from the T32 experiment. (Work is ongoing)