

MarlinTPC: Towards a common TPC software framework

Jason Abernathy¹, Klaus Dehmelt², Ralf Diener³, Christian Hansen¹,
Matthias Enno Janssen², Martin Killenberg⁴, Astrid Münnich⁵, Peter Wienemann⁴

¹University of Victoria

²DESY

³University of Hamburg

⁴University of Bonn

⁵RWTH Aachen

International Linear Collider Workshop 2007

May 29 – June 04, 2007

Hamburg, Germany

Status quo

- Many different TPC simulation, reconstruction and analysis packages have been developed
- Varying motivations for writing them
- Often similar codes, but they use different data formats, coordinate systems, units, etc.
- Most are monolithic programmes
- Exchanging code or data for cross-checks or to avoid double work can be time consuming and error-prone

Agreement

Held a TPC software workshop at DESY in June 2006,
trying to improve on the situation

Outcome of long discussions:

Proposal for an ILC TPC data stream

TIES BEHNKE^a, MAXIMILIEN CHEFDEVILLE^b,
FRANK GAEDE^a, CHRISTIAN HANSEN^c, MATTHIAS ENNO JANSSEN^a,
ALEXANDER KAOUKHER^d, MARTIN KILLENBERG^e,
JASON MCGEACHIE^c, ASTRID MÜNNICH^e, ADRIAN VOGEL^a,
MICHAEL WEBER^e, PETER WIENEMANN^f

^a*DESY*

^b*NIKHEF*

^c*University of Victoria*

^d*University of Rostock*

^e*RWTH Aachen*

^f*University of Freiburg*

Draft from July 3, 2006

Abstract

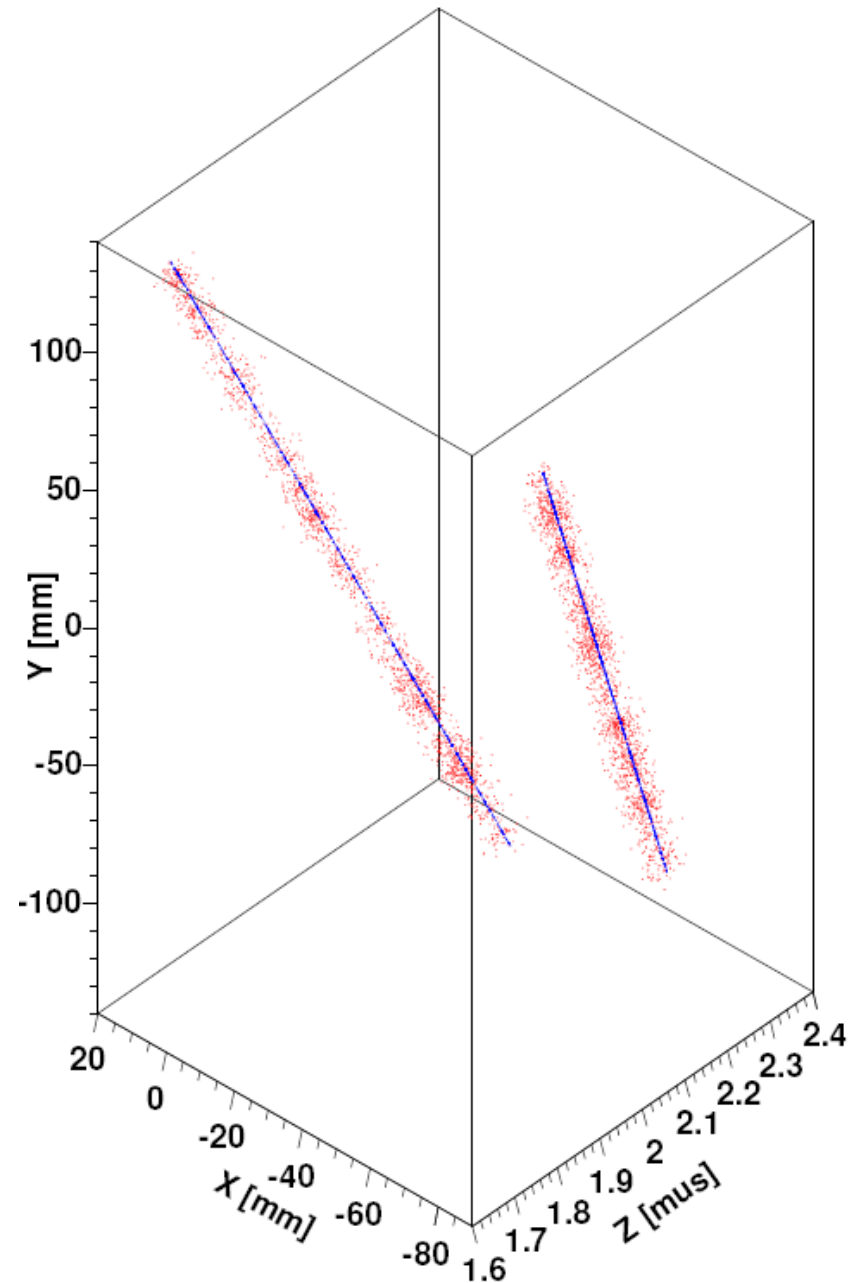
This document proposes a TPC data flow model for use during ILC detector R&D studies. It is based on LCIO data structures and Marlin as analysis and reconstruction framework.

MarlinTPC

- Goal: provide complete simulation, reconstruction and analysis framework using the data model described in the document
- Collection of existing and new algorithms in a single modular framework
- Modularity simplifies re-usability
- Based on ilcsoft tools widely used in ILC community:
 - LCIO: underlying model for transient and persistent data
 - MARLIN: modular analysis and reconstruction framework
 - GEAR: store and access geometry information
 - LCCD: conditions database package

Simulation

- Imported Astrid Münnich's GEMTPCSimulation package into MarlinTPC
- Performs
 - parametrised primary charge deposition
 - drift with diffusion
 - detailed parametrised simulation of amplification and charge transfer in GEM stack
 - digitisation (providing `lcio::TrackerRawData`)



Reconstruction

Reconstruction chain (from document):

Data structure	Processor name	input/output collection name
TrackerRawData		TPCRawData
	TrackerRawData2DataConverter	done
TrackerData		TPCConvertedRawData
	PedestalSubtractor	done
	ChannelByChannelCorrector	missing
	LinearityCorrector	missing
	TimeShiftCorrector	missing
TrackerData		TPCData
	PulseFinder	done
	ChannelMapper	done
	GainCorrector	missing
TrackerPulse		TPCPulses
	HitFinder	done
	HitPRFCorrector	under development
TrackerHit		TPCHits
	TrackFinder[Method]	done
Track		TPCSeedTracks
	TrackFitter[Method]	under development
Track		TPCTracks

Only the correction processors are still missing

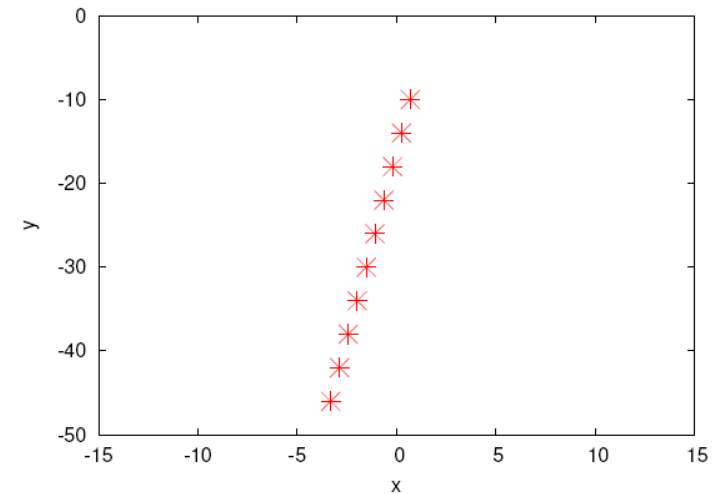
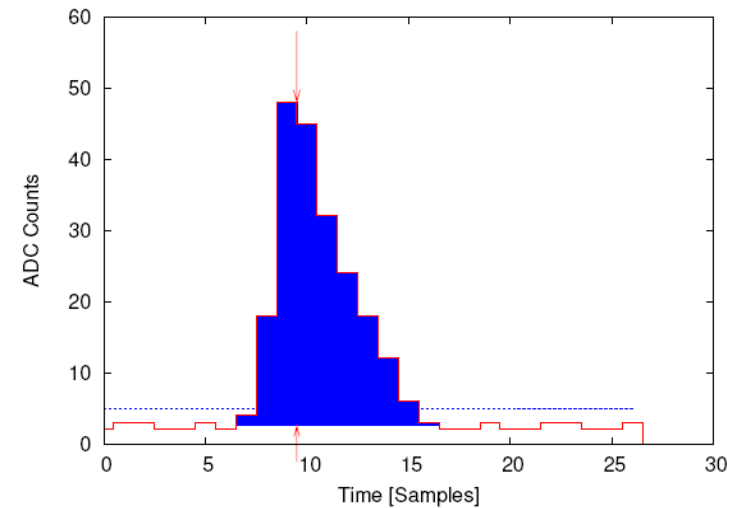
Pulse and hit finder

- **PulseFinder:**

- handles positive and negative polarity signals
- can be run on zero and non-zero suppressed data
- pre and post samples included in pulses
- threshold in terms of standard deviations from pedestal calculator
- no correct handling of double pulses yet

- **HitFinder:**

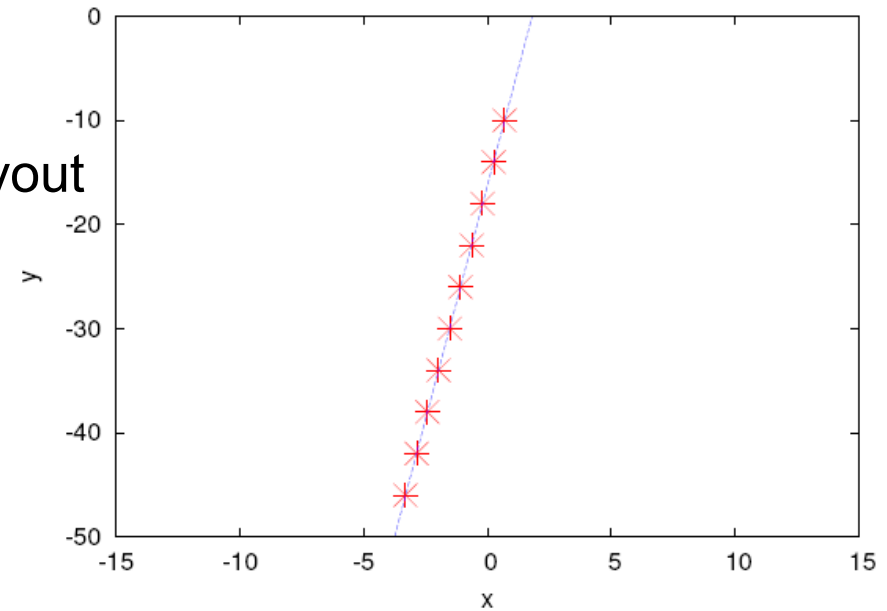
- still simple clustering algorithm without proper treatment of double tracks
- so far only RectangularPadRowLayout



Track finder

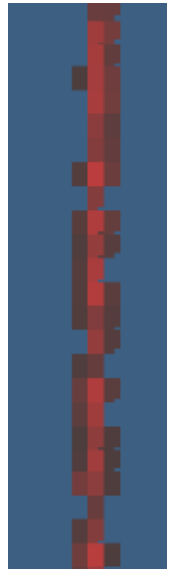
- **TrackFinderLinearRowBased:**

- simple hit based track finder assuming RectangularPadRowLayout
- presently only works for straight tracks (or small curvature)
- provides track parameters from χ^2 fit, can be used as seed for track fitter



In the pipeline

- Global likelihood based track fit processor
 - algorithm as developed at University of Victoria
- Analysis processor
 - providing all information as formulated in the conclusion of the first ILC TPC Analysis Jamboree in 2006:
resolution with geometric mean method, resolution with respect to reference track, bias plots, etc.
- Event display
 - extension of CED
- More sophisticated track finding algorithm
- Make all relevant processors work with `Rectangular-PadRowLayout` and `FixedPadSizeDiskLayout`
- Pad response function corrector and hit based track fitter
 - algorithm as developed at DESY



Summary and outlook

- Goal of MarlinTPC is to provide a common modular TPC simulation, reconstruction and analysis framework
- Most important processors of reconstruction chain are already available (though still partly with limited functionality)

Plans:

- Test MarlinTPC with real prototype data (not simulation)
- Include other readout technologies (TDCs, Timepix) in the data model and subsequently in MarlinTPC code. Where to switch to the conventional reconstruction chain?
- In the long run, merge code with full detector software (MarlinReco)

Further contributors or beta testers are highly welcome

More information

- **Homepage:**
<http://www-flc.desy.de/ilcsoft/ilcsoftware/marlintpc>
- **Mailing list:**
<https://lists.desy.de/sympa/info/ilcsoft-marlintpc>
- **User workbook:**
<https://twiki.cern.ch/twiki/bin/view/ILCTPC/MarlinTPC>
- **Subversion repository:**
<svn://pi.physik.uni-bonn.de/MarlinTPC/trunk>