TPC Testbeam Software

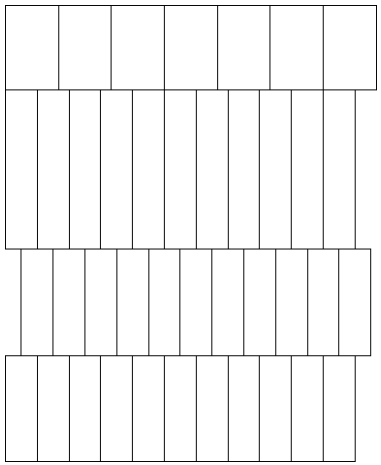
Martin Killenberg

Physikalisches Institut

universitätbonn

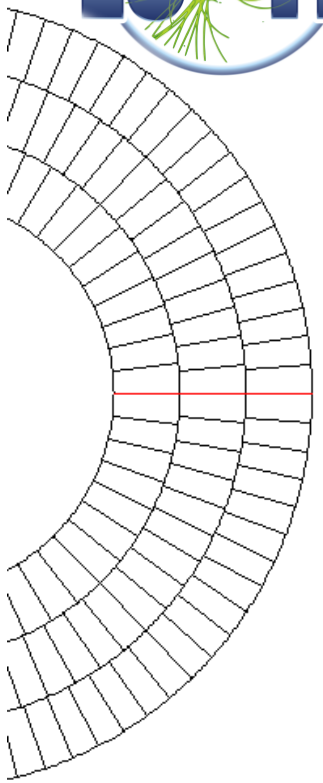
LCTW 2009
Orsay, 04.11.2009

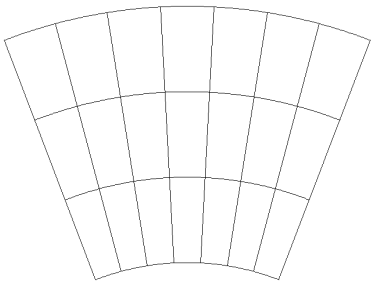
Pad Layouts in GEAR



- **RectangularPadRowLayout**
  - Cartesian Geometry
  - All pads in one row are equal
  - Pad size and number of pads may vary from row to row

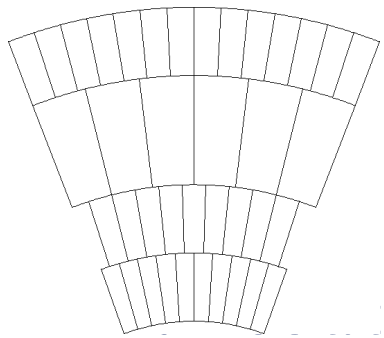
- **FixedPadSizeDiskLayout**
  - Polar Geometry
  - Complete circle
  - *All* pads have the same size





- **New: FixedPadAngleDiskLayout**
  - Polar Geometry
  - Segment of a circle
  - All pads have the same angle

- **New: VersatileDiskRowLayout**
  - Polar Geometry
  - Segment of a circle
  - All pads in one row are equal
  - Pad size and number of pads may vary from row to row



Multiple Modules in GEAR

A realistic TPC end plate (EUDET LP) consists of multiple modules
$\Rightarrow$ introduce TPCModule in GEAR

TPCModule

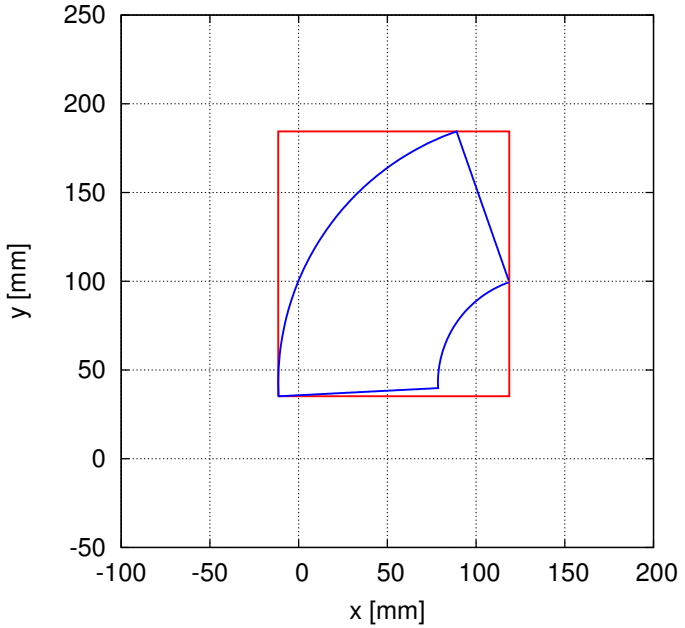
- derived from PadRowLayout2D
- contains a pad layout implementation

$\Rightarrow$ full backward compatibility and transparency in user code

Coordinates:

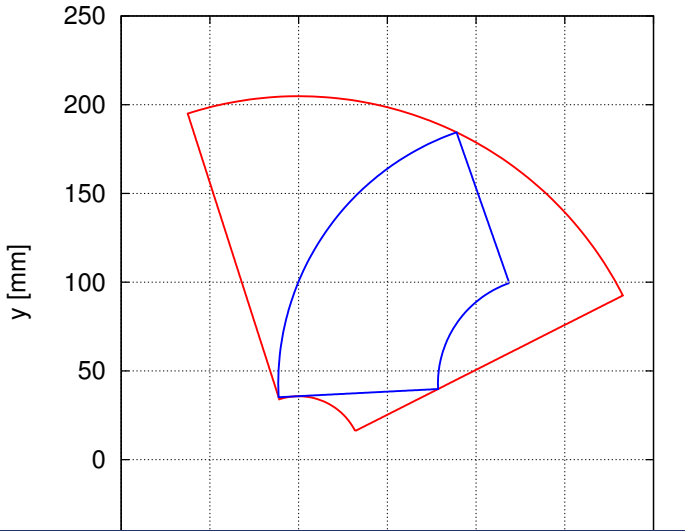
- Pad plane implementation provides local coordinates
- Module has offset and angle to global coordinate system
- Accessing the pad plane through the module automatically provides correct global coordinates

Example: Plane extent

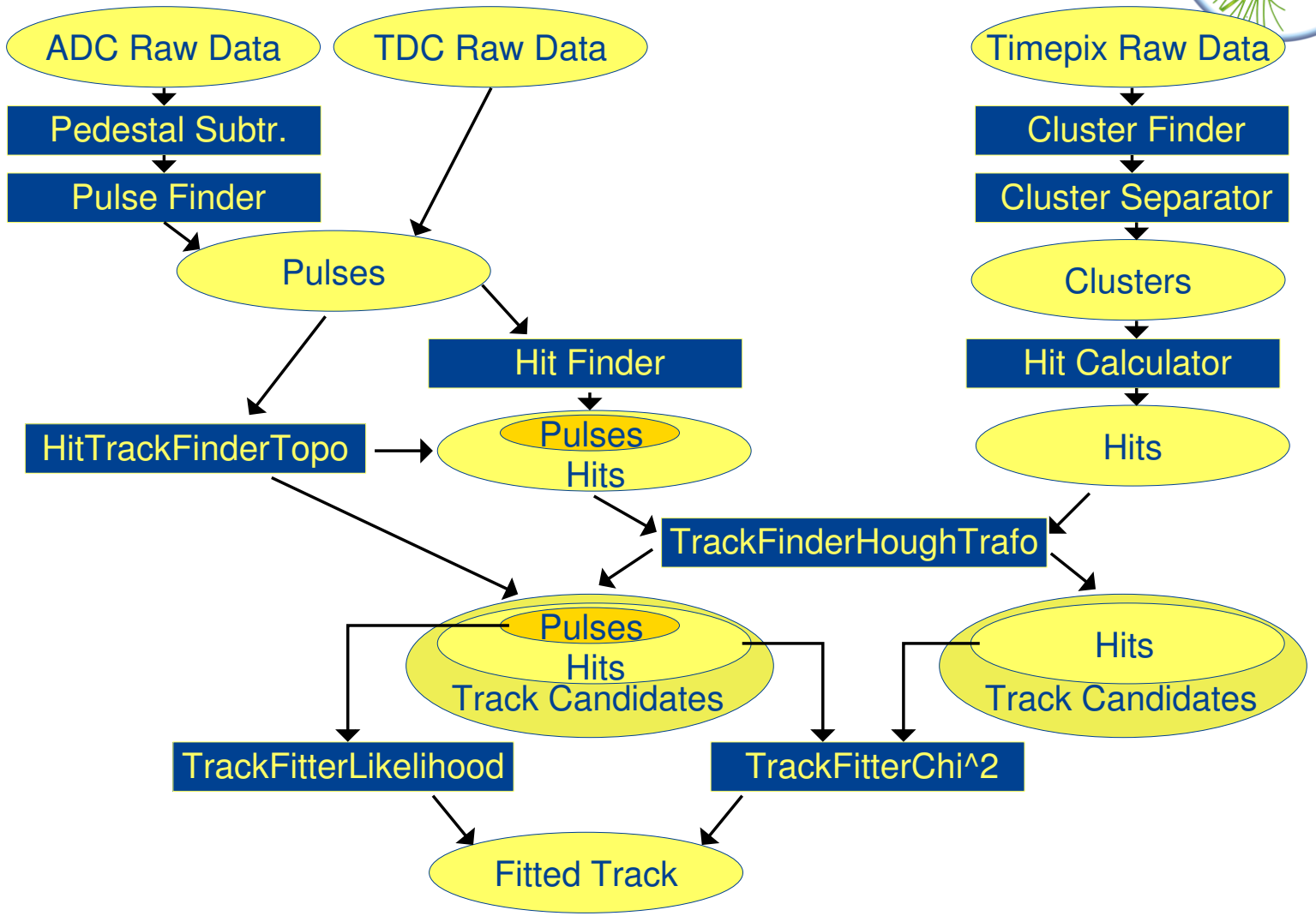


↑ global Cartesian coordinates

↓ global polar coordinates



MarlinTPC Reconstruction Overview



universitätbonn

MarlinTPC Reconstruction Status

Status last year:

- Chain is ready for non-modular version, but only tested with toy MC

Requirements for LP test beam:

- Multiple modules
  - Include gear multiple modules in processors (✓)
  - Add ModuleID to conditions data classes (e. g. pedestals) (✓)
  - Alignment (✓)
- Data taking with magnetic field
  - Helical track fit (✓)[1]
- Calibration (×)
  - Toy MC does not need calibration
  - Has to be developed with real data

[1] only simple fit without correct treatment of covariance matrix

Alignment

Alignment is possible:

- Calculate offsets manually
- Hard-code offsets in GEAR XML files

Goal: Apply alignment at run time using LCCD

- Should be transparent to existing reconstruction code
- Independency on pad plane implementation should not be broken

$\Rightarrow$ Keep the PadRowLayout2D interface

How to implement it?

- Store calibrated XML files in database?
- Extend functionality of TPCModule?
- Write a wrapper class?
  - Inherit from TPCModule?
  - Have an instance of TPCModule?

Conditions Data

- **L**inear **C**ollider **C**onditions **D**ata toolkit
- TPCCondData: LCIO / LCCD classes for the TPC

Data needed	TPCCondData	Available for test beam run				
		Japanese GEM ALTRO	Japanese GEM TDC	Micro-megas	Std. GEM	Timepix + GEM
Channel mapping	✓	✓	✓	✓	✓	
Channel quality	✓	×	×	×	×	×
Pedestals	✓	×	×	×	×	×
v_{drift} + Diffusion	✓	×	×	×	×	×
Gas conditions	✓	✓	✓	✓	✓	✓
E-field settings	✓	×	×	×	×	×
B-field settings	×					
HV settings	×					
Field maps	✓			×		
Electronics	×					
Calibration	×					

Conditions Data

- **L**inear **C**ollider **C**onditions **D**ata toolkit
- TPCCondData: LCIO / LCCD classes for the TPC

Data needed	TPCCondData	Available for test beam run				
		Japanese ALTRO	GEM TDC	Micro-megas	Std. GEM	Timepix + GEM
Channel mapping	✓	✓	✓	✓	✓	
Channel quality	✓	×	×	×	×	×
Pedestals	✓	×	×	×	×	×
v_{drift} + Diffusion	✓	×	×	×	×	×
Gas conditions	✓	✓	✓	✓	✓	✓
E-field settings	✓	×	×	×	×	×
B-field settings	×					
HV settings	×					
Field maps	✓					
Electronics	×					
Calibration	×					

We have to get a woking data base and conditions data bookkeeping

Extending LCCD Functionality

Two types of conditions data:

- Calibration data
  - Pedestals, mappings, gain factors
  - Collections / maps of many identical objects
  - Possibly large amount of data
  - Usually no DB access to individual data members

  $\Rightarrow$ Store as binary objects
  Current functionality of LCCD

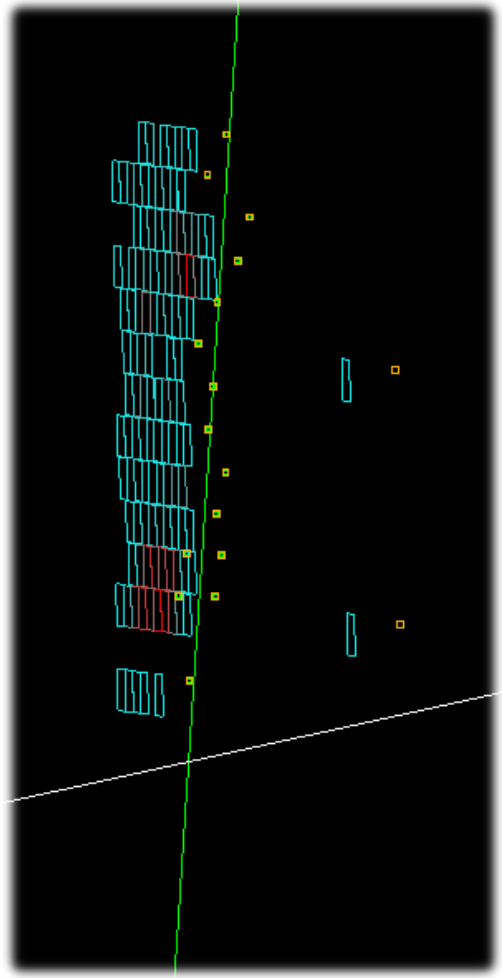
- Meta data
  - Gas mixture, beam energy, particle type, position of detector
  - No collection, indvidual data objects
  - Only few integer / float (/ string?) values
  - Query DB to search for runs / data sets

  $\Rightarrow$ Store data members directly in DB table

Event Display

For development of geometry descriptions and fist checks during data taking a graphical viewer is needed.

HepRepOutputProcessor produces HepRep XML file which can be displayed e. g. with Wired/JAS3



Event display shows

- TPC
- GEAR pad plane
- Charge on pads
- 3D hits
- Tracks

Disadvantages

- MarlinProcessor is offline software
- Events not browsable
- Graphical display has errors (depth)
- Very bad performance

universitätbonn

Conclusions

- GEAR
  - New pad layout classes for existing prototypes
  - Multiple modules incl. coordinate transformations
  - Implement allignment from conditions data
- MarlinTPC
  - All reconstruction processors are multi module capable
  - Performance has to be improved
  - Develop calibration tools with real data
  - EventDisplay: HepRepOutputProcessor + Wired/JAS3
- Conditions data
  - Define missing data classes for calibration
  - Set up a data base
  - Queryable conditons data