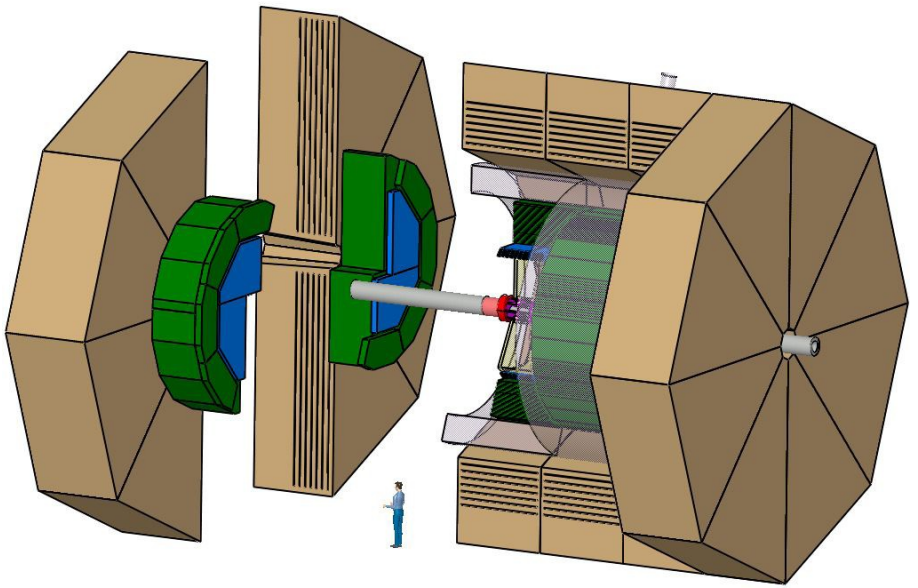
Marlin et al for full Reconstruction

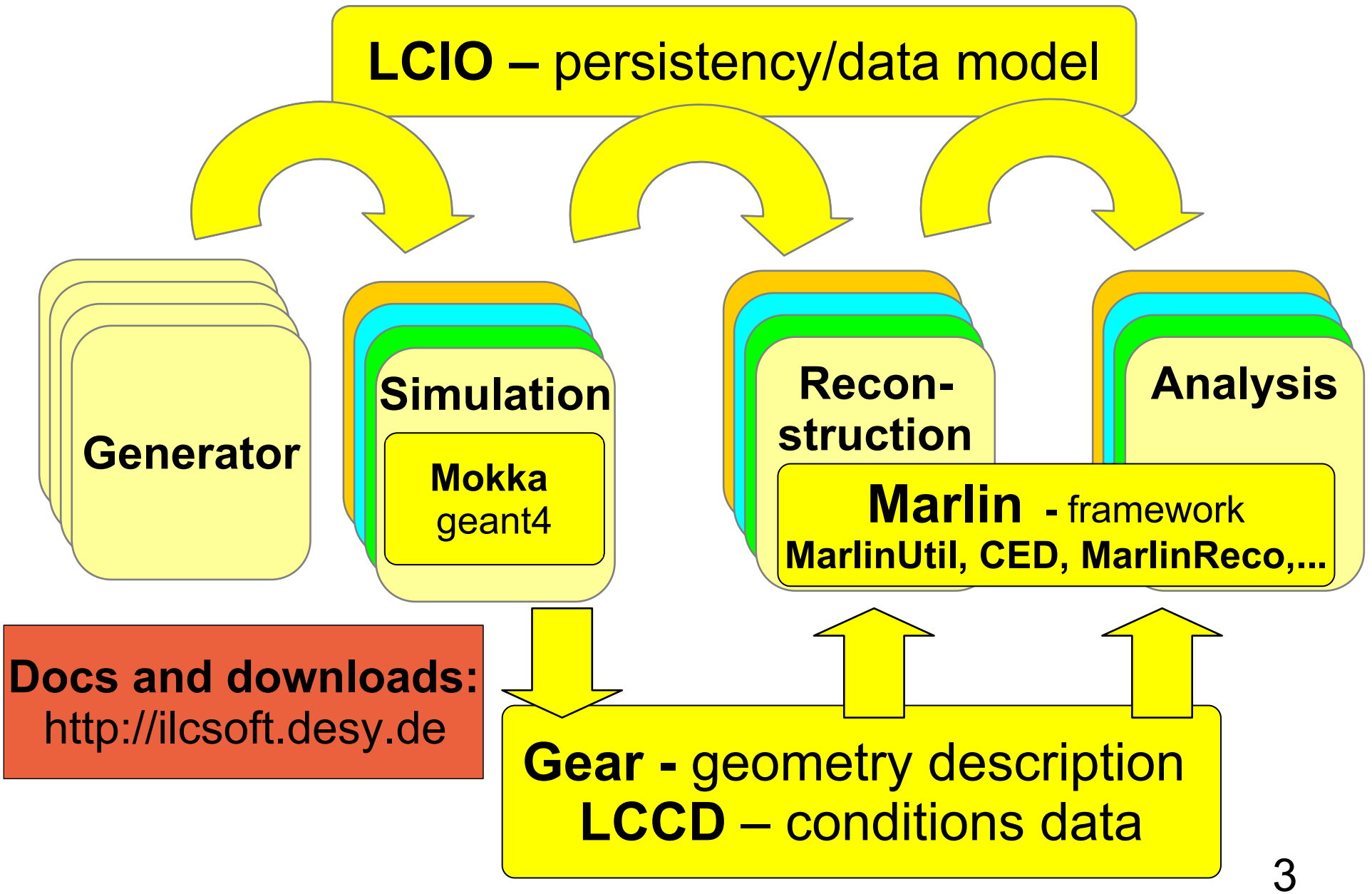
Frank Gaede
DESY
ILD Meeting, Zeuthen,
14-16 January 2008

Outline

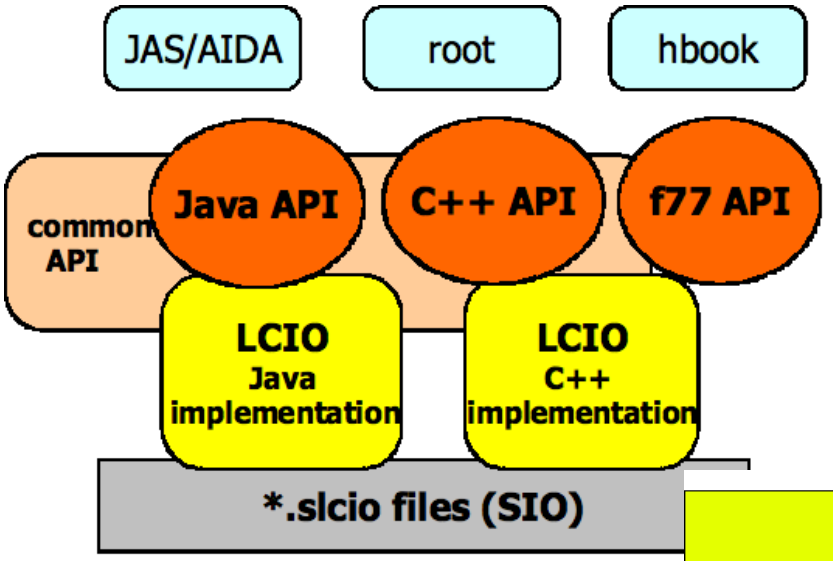
- overview core tools
  - LCIO, Mokka, Marlin, Gear, LCCD
  - status and features
- putting it all together
  - full reconstruction:
  - tracking
  - particle flow
- status of framework
- summary



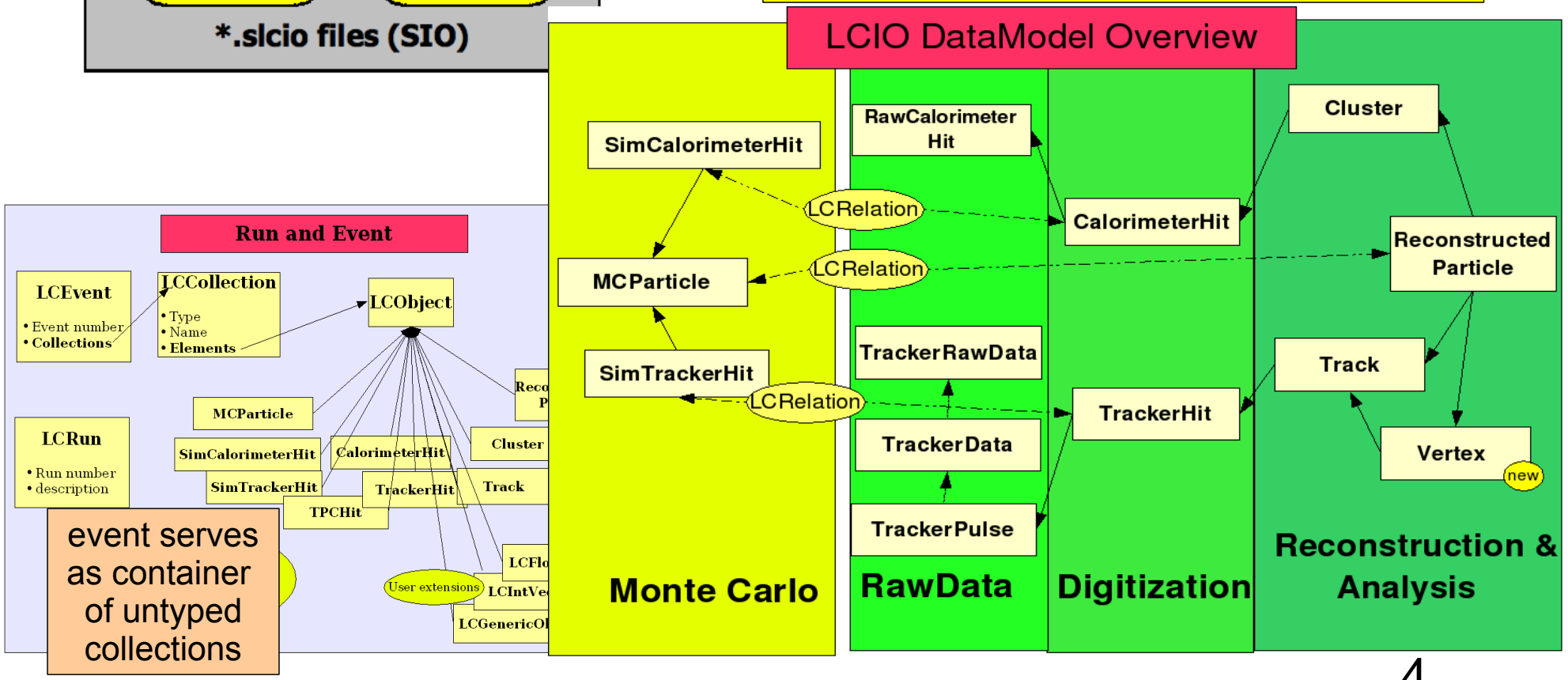
LDC sw-framework overview



LCIO: persistency & event data model



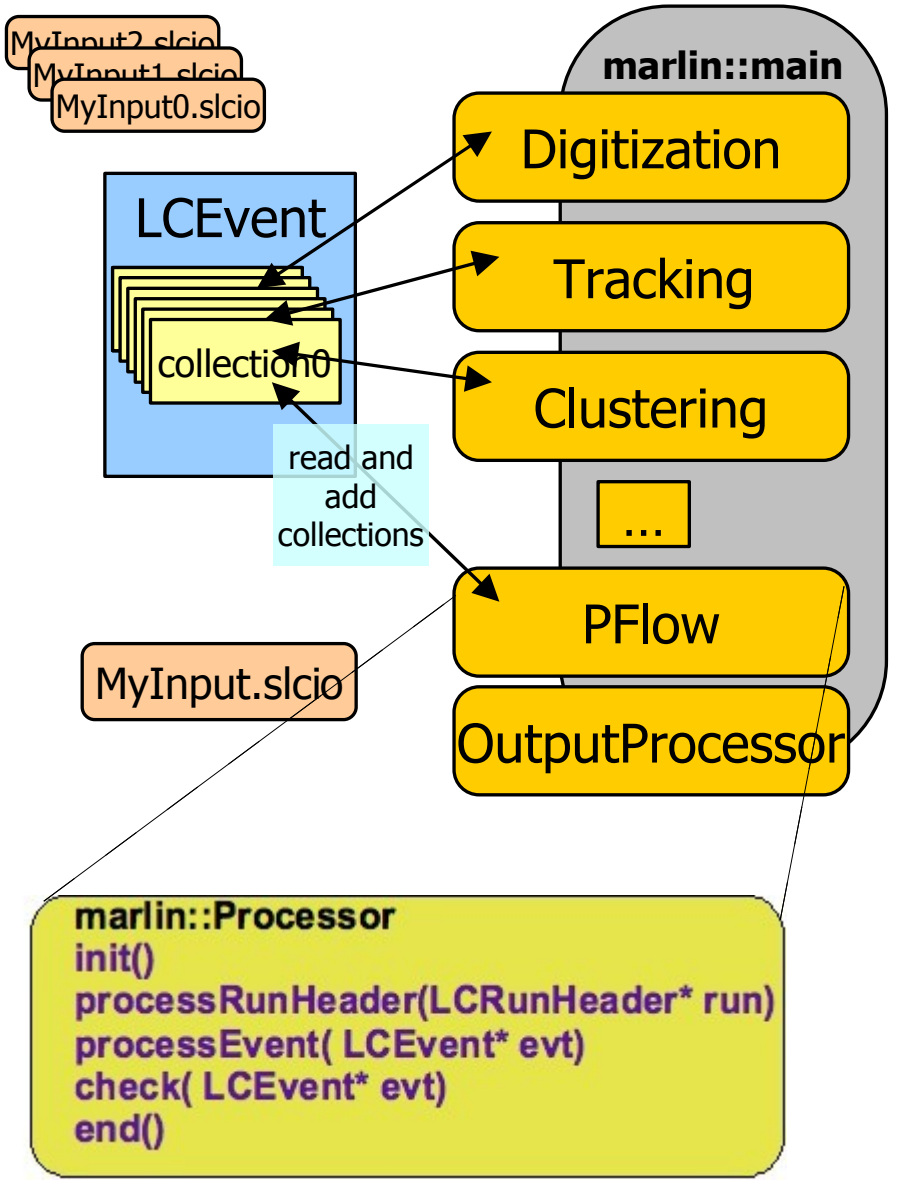
- DESY SLAC joined project
  (first presented at CHEP03)
- Java, C++ and f77 (!) API
- extensible data model
- now standard for
- ILC persistency & datamodel
- -> used in all detector concept
- studies



Marlin – application framework

Modular**A**nalysis & **R**econstruction for the **L I N**ear Collider

- modular C++ application framework for the analysis and reconstruction of ILC data
- **LCIO** as transient data model
- xml steering files:
  - fully configure application
  - order of modules/processors
  - parameters global + processor
- self documenting
  - parameters registered in user code
- consistency check of input/output collection types
- **Plug & Play** of modules



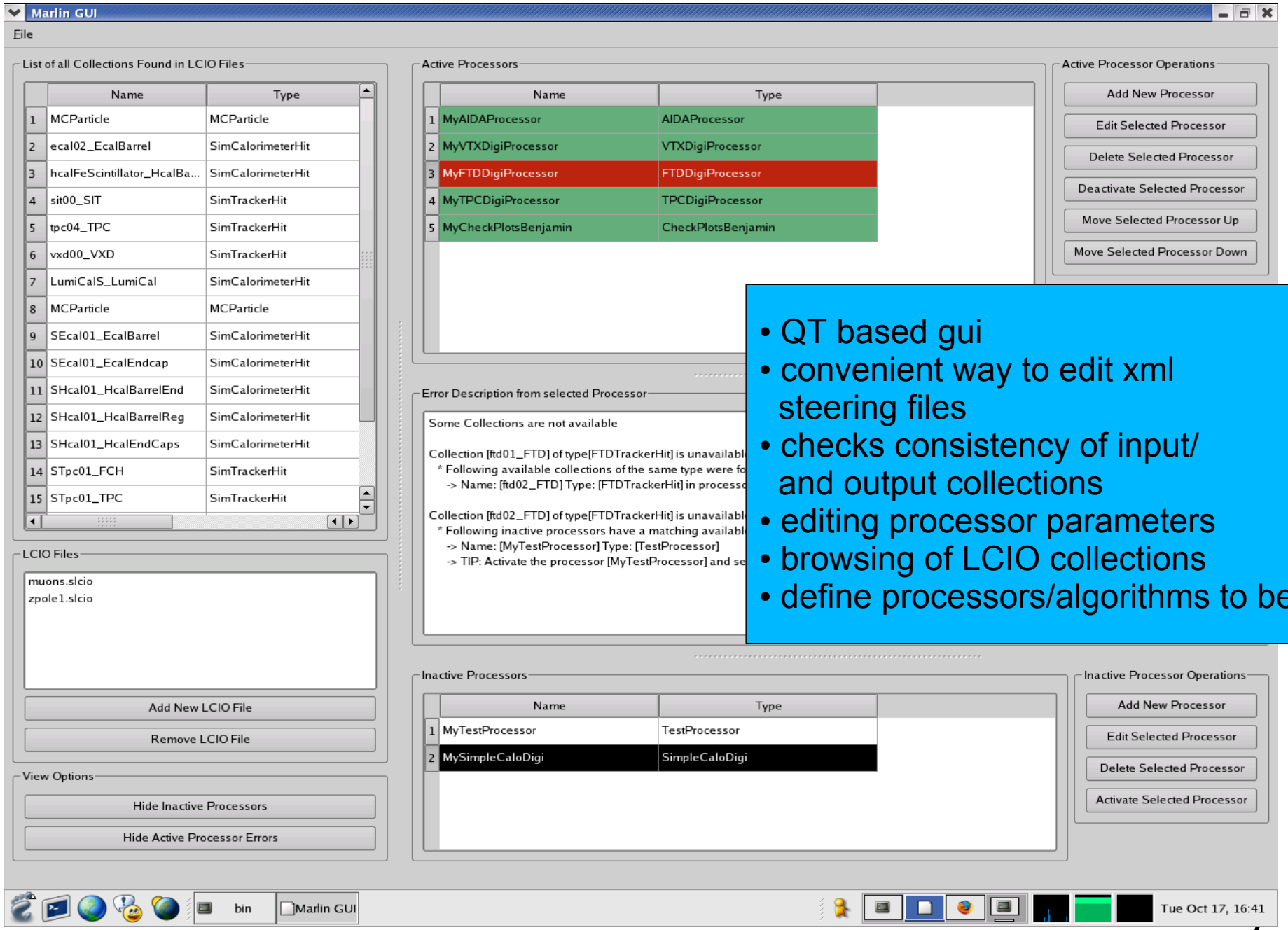
Marlin recent developments

- Marlin fully functional since 2005
  - -> focus on increasing user, i.e. developer convenience
- introduced new build system: CMake
  - ' easy configuration of build process and multi-platform support (Linux, MacOS, Windows)
  - also used for all other core tools
- switched to shared libraries and support for plugins
  - users can combine their binary from installed package libraries
- MarlinGUI,
  - flow charts
- new logging mechanism:

```
streamlog_out( DEBUG ) << " digitizing hit : "
                       << hit->getCellID() << std::endl ;
[ DEBUG "TrackDigitizer" ]  digitizing hit : 12345678
```

MarlinGUI

J.Engels, DESY



- QT based gui
- convenient way to edit xml steering files
- checks consistency of input/ and output collections
- editing processor parameters
- browsing of LCIO collections
- define processors/algorithms to be run

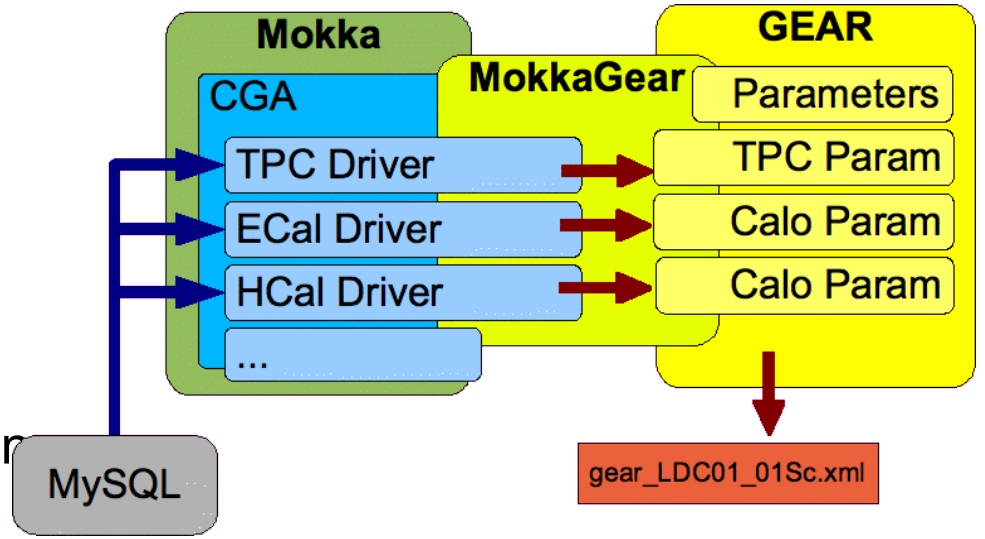
Marlin supporting packages

- the Marlin framework is completed by additional packages for
- description of detector geometry: **GEAR**
- conditions data: **LCCD**
- utility software , math libraries, ...
  - MarlinUtil – utility library
  - CED - event display
  - RAIDA – root AIDA implementation
  - CLHEP, gsl, cernlib,....

geometry for reconstruction

GEometry **A**PI for **R**econstruction

- high level abstract interface:
  - per subdetector type (Hcal,TPC,...) parameters/quantities for reco
    - geometry + some navigation
    - implementation uses xml files written from Mokka (simulation)
  - abstract interface for detailed geometry &materials:
    - point properties
    - path properties
    - implementation based on geant4



MokkaGear

- enforce only one source of geometry: the simulation program creates the geometry xml files used in reconstruction

(recently improved by K.Harder et al)

example – GEAR API VXD

Gear: gear::VXDParameters class Reference - Mozilla Firefox

File Edit View Go Bookmarks Tools Help

http://ilcsoft.desy.de/gear/v00-03/doc/html/classgear_1_1VXDParameters.html

simulation/geant4 LCIO Linux Conferences DESY IT Group LEO English/Ger... Google MyHome Ctime

virtual const **VXDLayerLayout** & **getVXDLayerLayout** () const=0
The layer layout in the Vertex.

virtual int **getVXDType** () const=0
The type of Vertex detector: VXDParameters.CCD, VXDParameters.CMOS or VXDPara

virtual double **getShellHalfLength** () const=0
The half length (z) of the support shell in mm (w/o gap).

virtual double **getShellGap** () const=0
The length of the gap in mm (gap position at z=0).

virtual double **getShellInnerRadius** () const=0
The inner radius of the support shell in mm.

virtual double **getShellOuterRadius** () const=0
The outer radius of the support shell in mm.

virtual double **getShellRadLength** () const=0
The radiation length in the support shell.

virtual bool **isPointInLadder** (**Point3D** p) const=0
returns whether a point is inside a ladder

virtual bool **isPointInSensitive** (**Point3D** p) const=0
returns wheter a point is inside a sensitive volume

virtual **Vector3D** **distanceToNearestLadder** (**Point3D** p) const=0
returns vector from point to nearest ladder

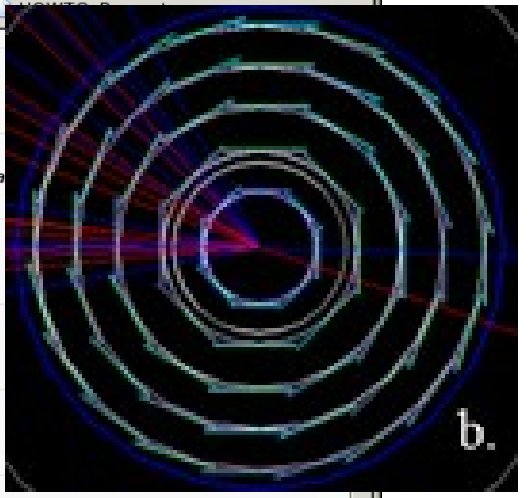
virtual **Vector3D** **distanceToNearestSensitive** (**Point3D** p) const=0
returns vector from point to nearest sensitive volume

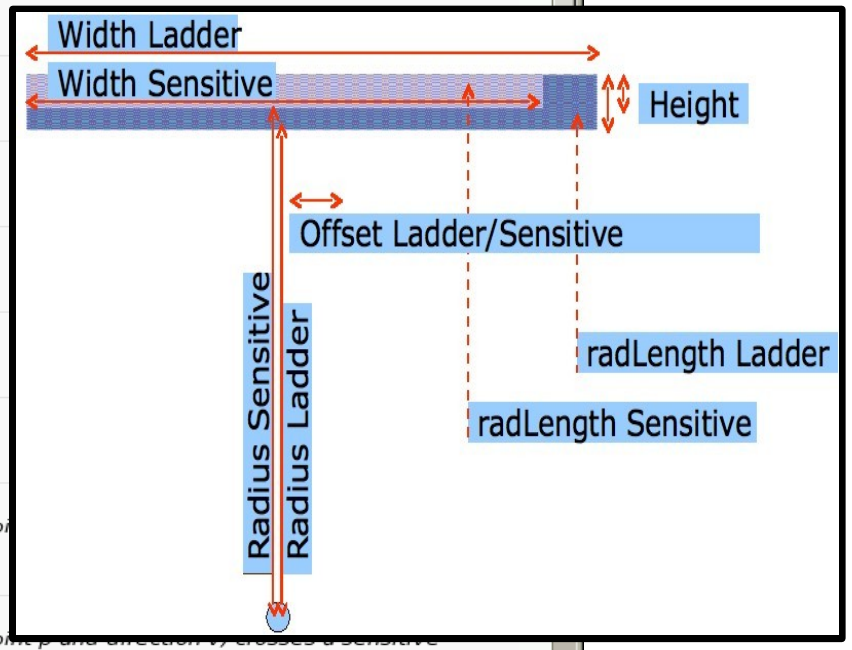
virtual **Vector3D** **intersectionLadder** (**Point3D** p, **Vector3D** v) const=0
returns the first point where a given strainght line (parameters po
is returned if no intersection can be found.

virtual **Vector3D** **intersectionSensitive** (**Point3D** p, **Vector3D** v) const=0
returns the first point where a given strainght line (parameters po
volume (0,0,0) is returned if no intersection can be found.

Find: VXD Find Next Find Previous Highlight Match case

Done

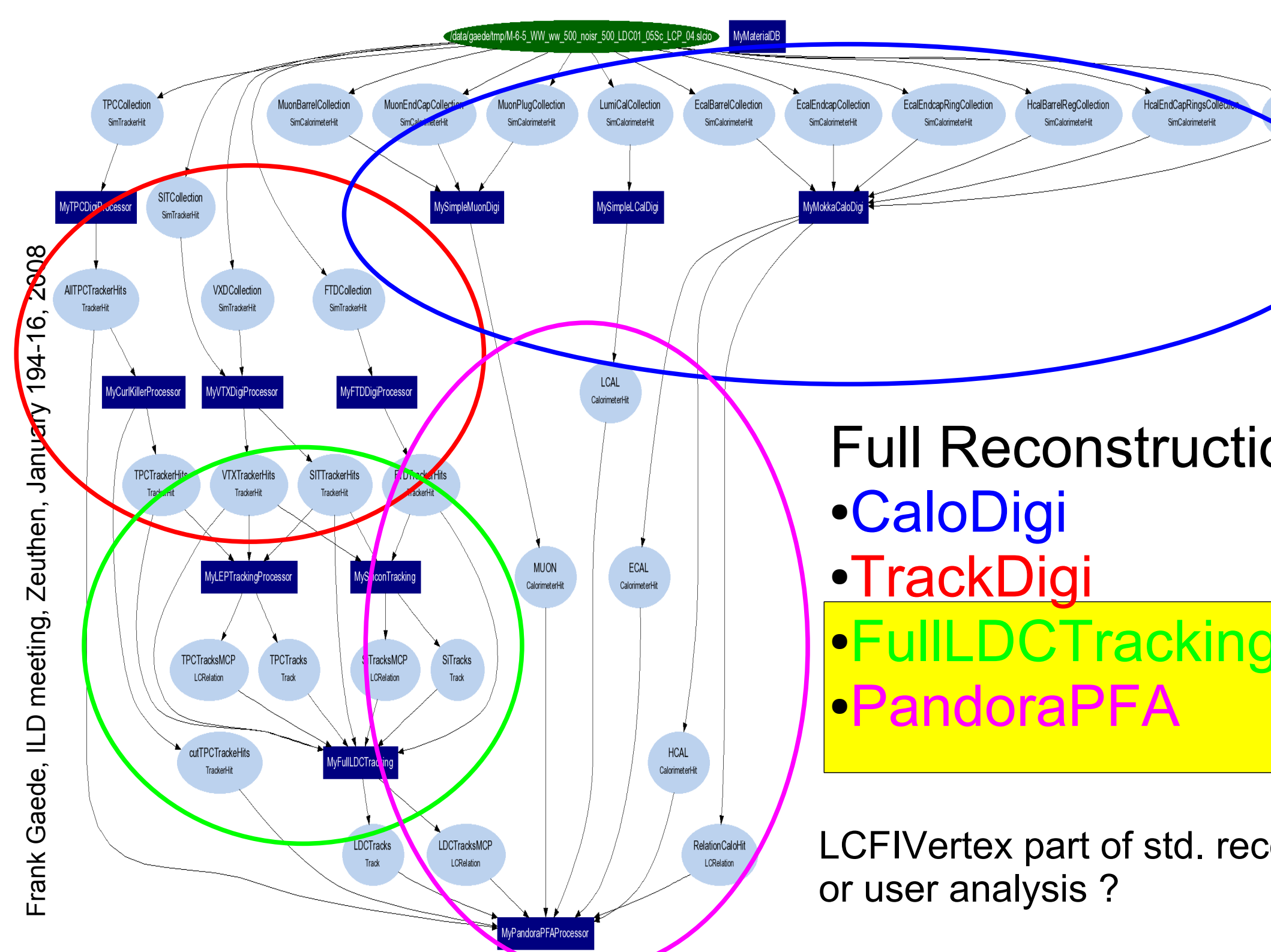




Applications of Marlin et al

- LDC detector optimization (MonteCarlo)
  - MarlinReco – full reconstruction suite
    - Digitization Calo,TPC, Silicon, PatternRecognition/Tracking, clustering, ParticleFlow algorithms: **Wolf**, **TrackBased**
  - PandoraPFA
    - ParticleFlow algorithm
  - LCFIVertex
    - ZVTop/ZVKin vertex finding and fitting algorithms
  - various physics analyses ...
- testbeams (Data & MonteCarlo)
  - Calice - calorimeter
  - MarlinTPC – TPC tracking
  - EUTelescope – pixel telescope for silicon tracking

using the same core framework for MC/offline and testbeam/online facilitates exchange of knowledge !



Frank Gaede, ILD meeting, Zeuthen, January 194-16, 2008

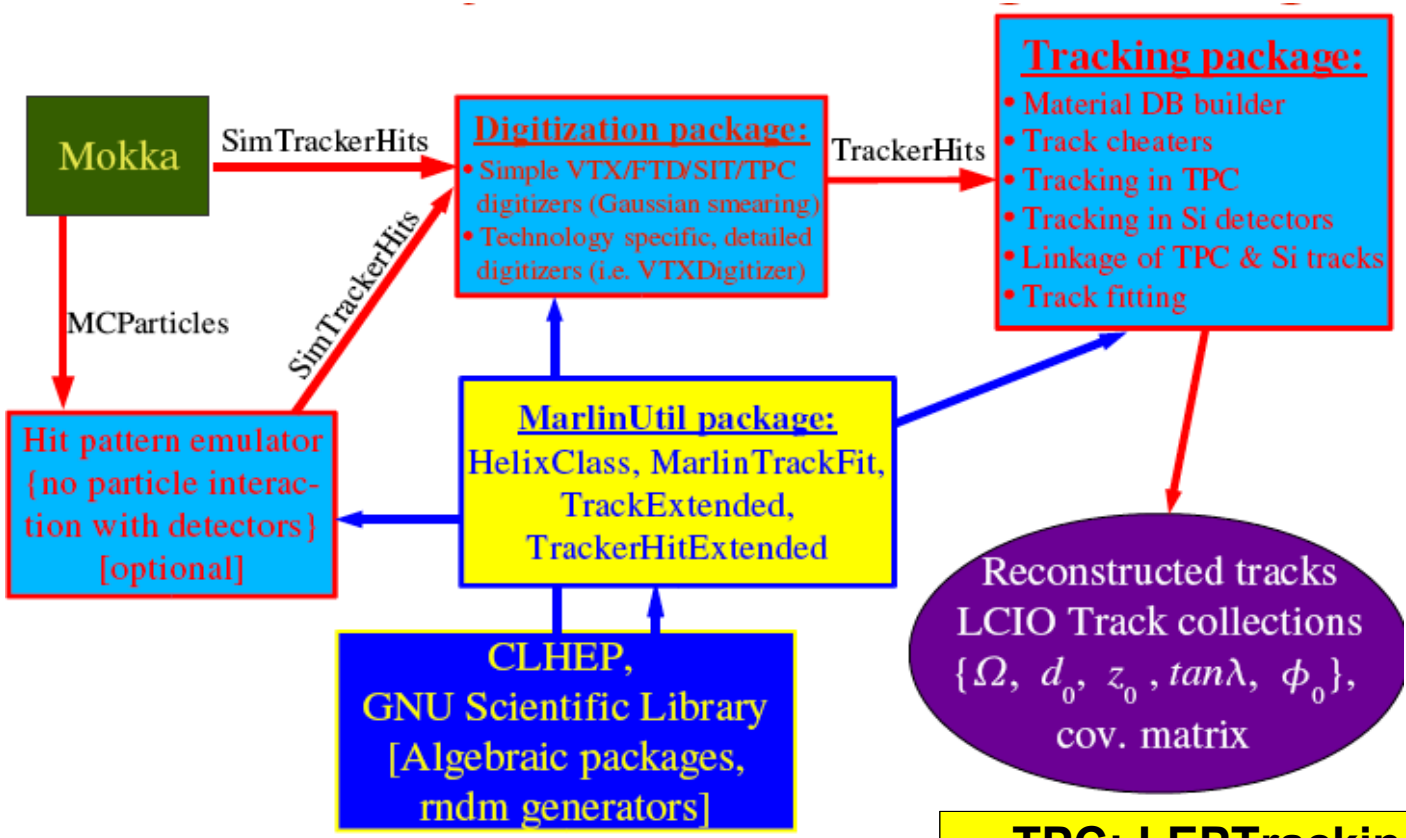
Full Reconstructio

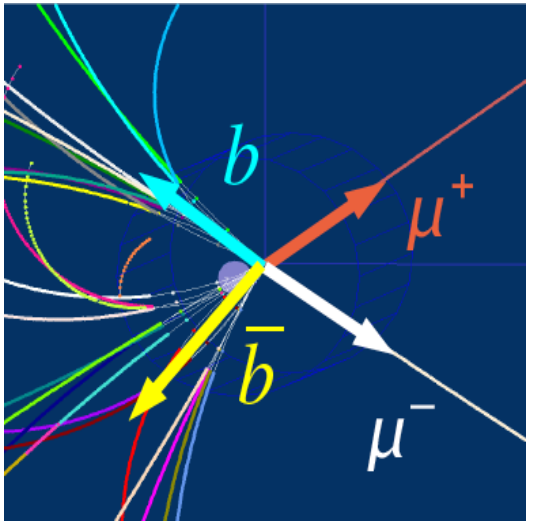
- CaloDigi
- TrackDigi
- FullLDCTracking
- PandoraPFA

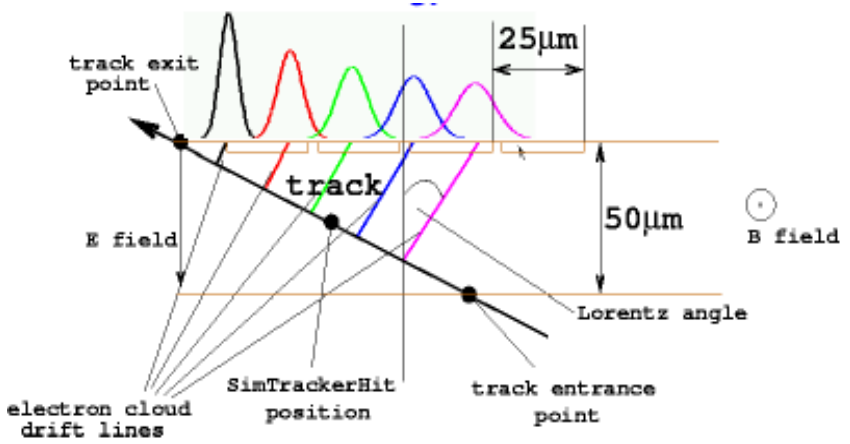
LCFIVertex part of std. rec
or user analysis ?

MarlinReco - FullLDCTracking

A.Raspereza (MPI)

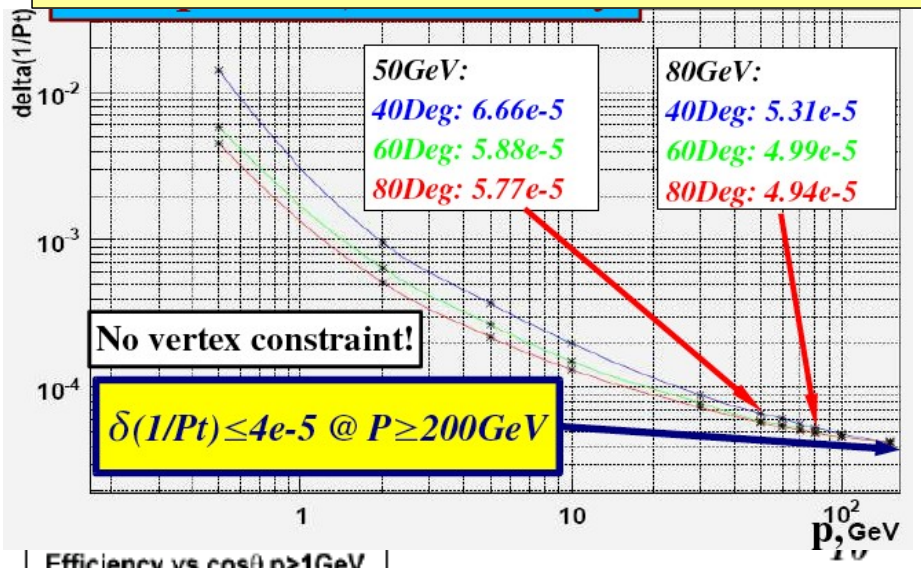




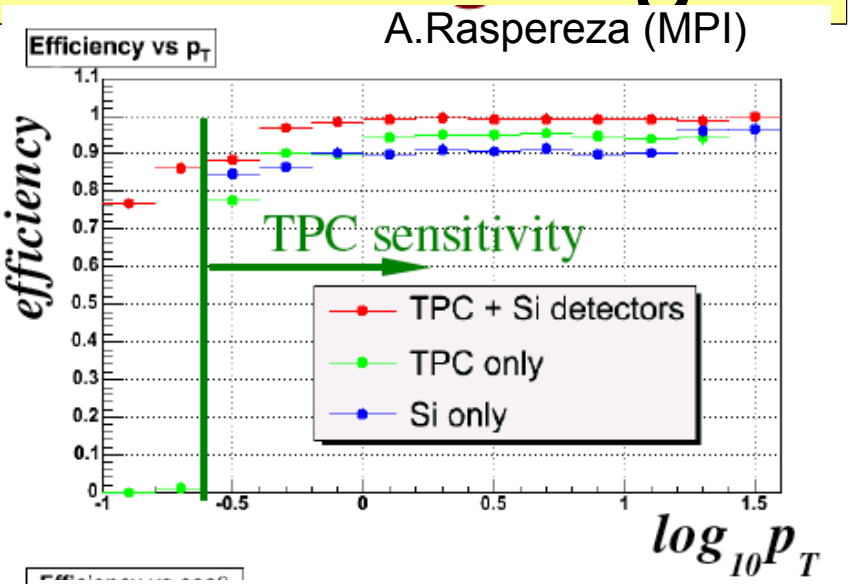


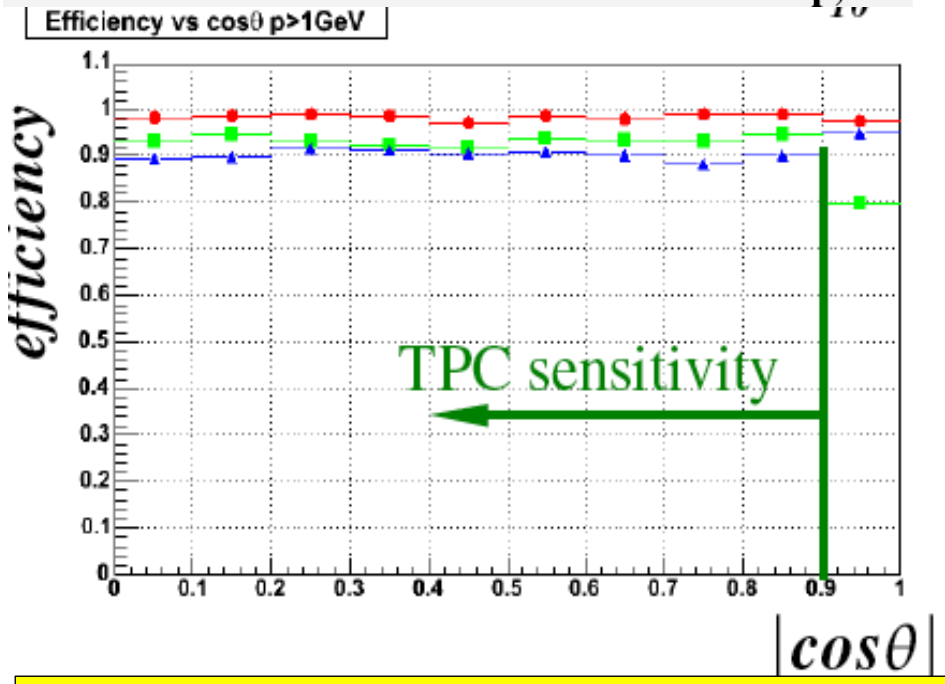
- **TPC: LEPTracking (wrapped LEP code)**
- **VXD, FTD, SIT:**
  - **detailed silicon digitization**
  - **standalone patrec + fitting**
- **LDCTracking:**
  - **combine tracks**
  - **find loopers**
  - **refit (Kalman Filter)**

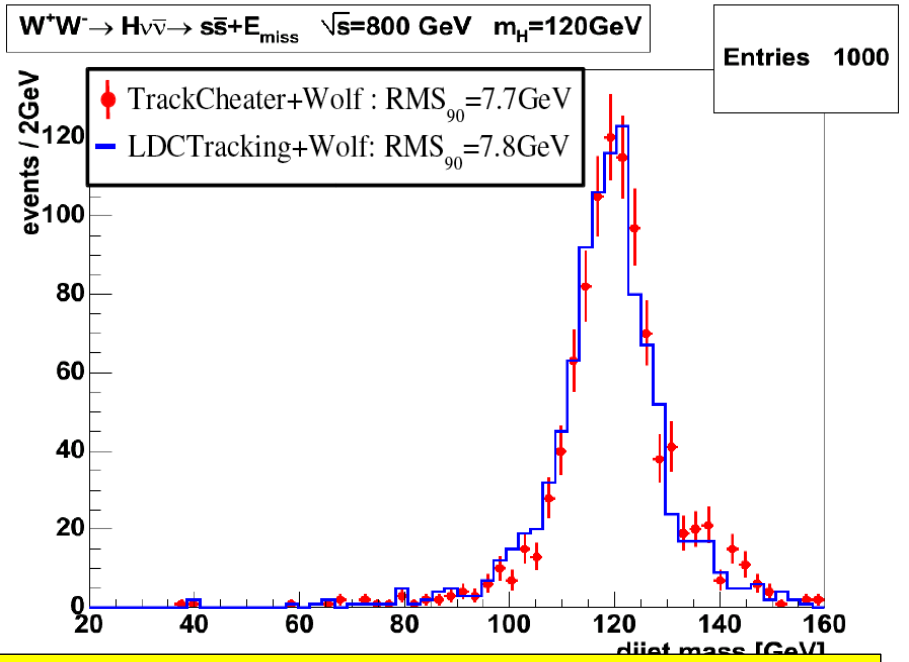
MarlinReco - FullLDCTracking



A.Raspereza (MPI)







- can now **use real tracking code** and PFA for detector optimization !
- improved recently: coherent Bfield desc., silicon ladders, fit options,...

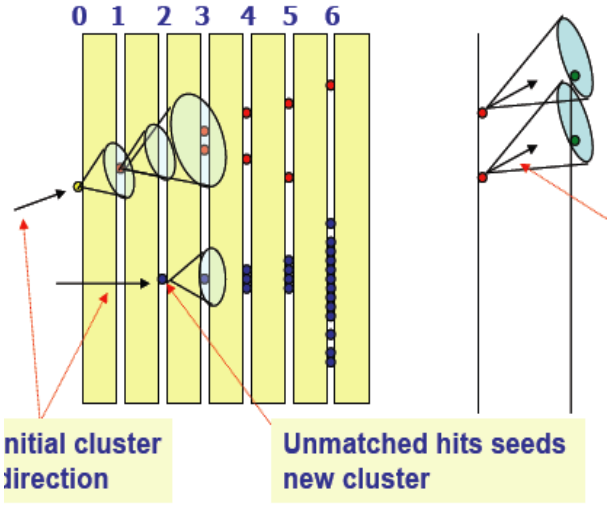
PandoraPFA

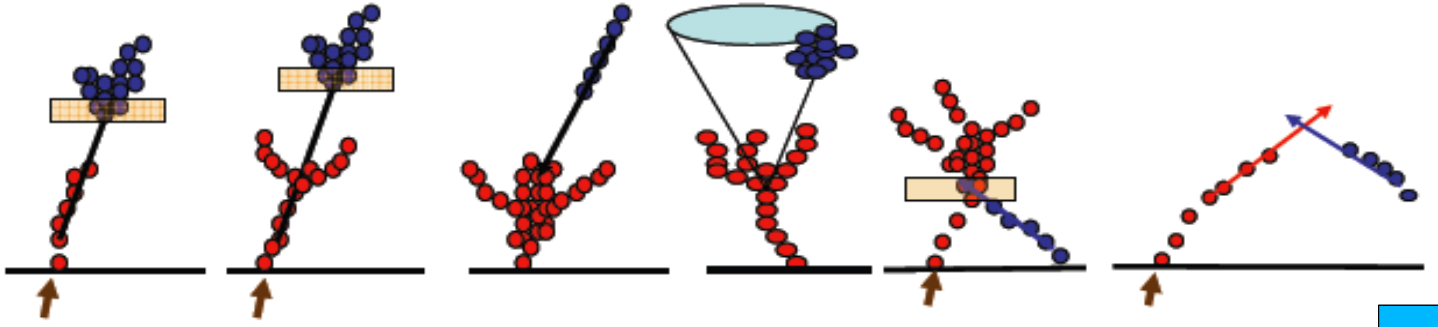
M.Thomson

i. Preparation (MIP hit ID, isolation, tracking)
ii. Loose clustering in ECAL and HCAL
iii. Topological linking of clearly associated clusters
iv. Courser grouping of clusters
v. Iterative reclustering
vi. Photon Recovery (NEW)

Order inter-changable

vii. Fragment Removal (NEW)
viii. Formation of final Particle Flow Objects (reconstructed particles) – not very sophisticated





If track momentum and cluster energy inconsistent : RECLUSTER

e.g.

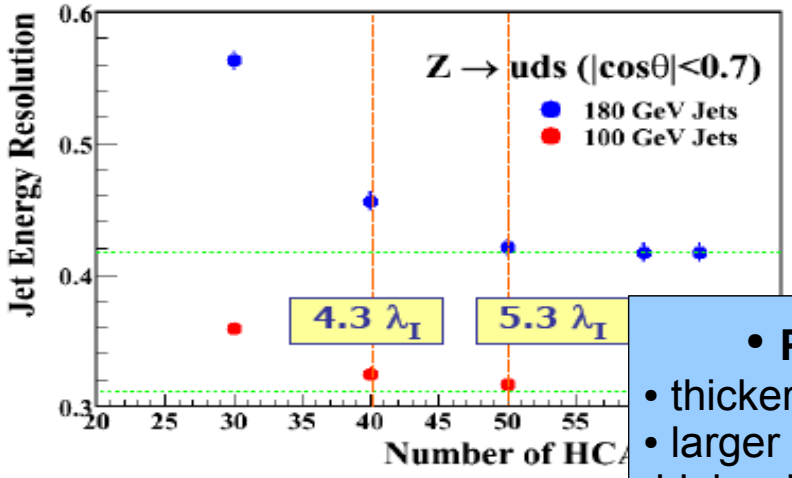
Pandora is the most sophisticated and best performing PFA to date

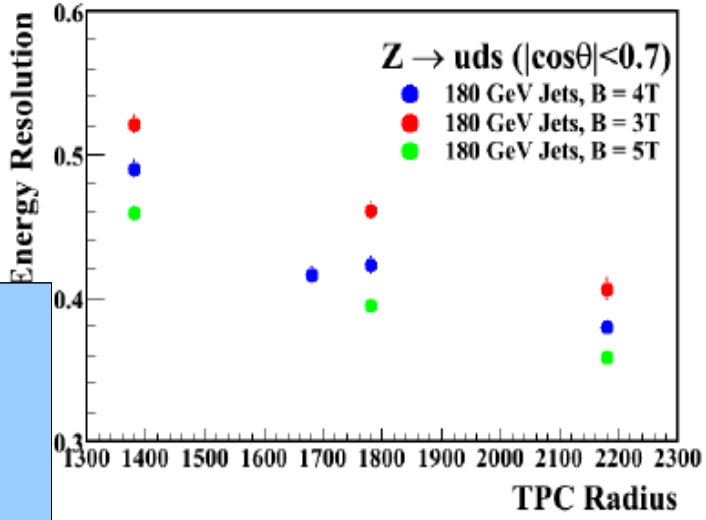
performance, PandoraPFA (M.Thomson,Cambridge)

E_{JET}	$\sigma_E/E = \alpha/\sqrt{E_{jj}}$ $\lvert\cos\theta\rvert<0.7$	σ_E/E_j
45 GeV	0.227	3.4 %
100 GeV	0.287	2.9 %
180 GeV	0.395	2.9 %
250 GeV	0.532	3.4 %

For jet energies < 100 GeV ILC "goal" reached !!!

'proof of concept' for PFA @ILC
-> use for detector optimization





- **PFA improves with:**
- thicker Hcal
- larger Tracking radius
  higher Bfield

-> use PFA for ILD detector optimization&physics analyses for LOI

Status of Marlin based Full Reco

- **new release ilcsoft v01-03 (xmas-07)**
  - all core tools + MarlinReco, PandoraPFA, ...
  - many improvements:
    - FullLDCTracking, event overlay, LCIO direct access, gear description, improved performance (PFA),...
- now put together a 'standard reconstruction' to be used for physics analyses
  - the tools are essentially there
  - -> need proper configuration & calibration
  - need lot of testing by experts
  - already some issues identified...

ILD optimization w/ full reco

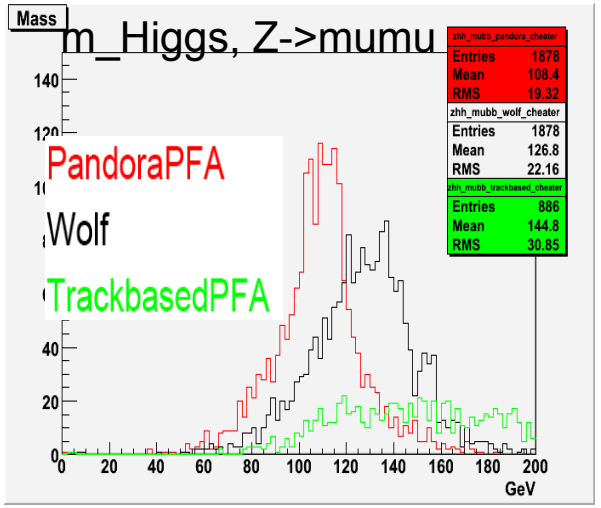
Sub-Detector	Parameter	GLD	LDC	GLD'	LDC'
TPC	R_{inner} (m)	0.45	0.30	0.45	0.30
	R_{outer} (m)	2.00	1.58	1.80	1.80
	Z_{max} (m)*	2.50	2.16	2.35	2.35
Barrel ECAL	R_{inner} (m)**	2.10	1.60	1.85	1.82
	Material	Sci/W	Si/W	Sci/W	Sci/W
Barrel HCAL	Material	Sci/W	Sci/Fe	Sci/Fe	Sci/Fe
Endcap ECAL	Z_{min} (m)***	2.80	2.30	2.55	2.55
Solenoid	B-field	3.0	4.0	3.50	3.50
VTX	Inner Layer (mm)	20	16	18	18

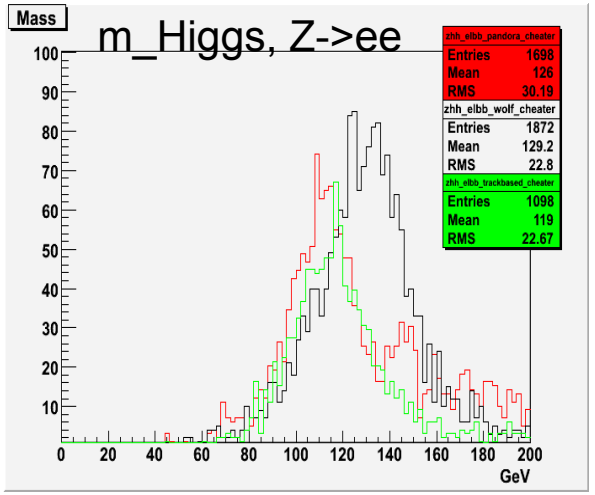
- use the full framework for Monte Carlo mass production:
  - two versions: LDC and LDC' initially
  - then more detector variations (scaling)
- Reconstruction code in Marlin has been written to be 'detector independent' (using geometry from gear only)
- however need to verify in detail at every parameter point
  - e.g. currently LEPTracking (TPC) breaks for #layer>200...

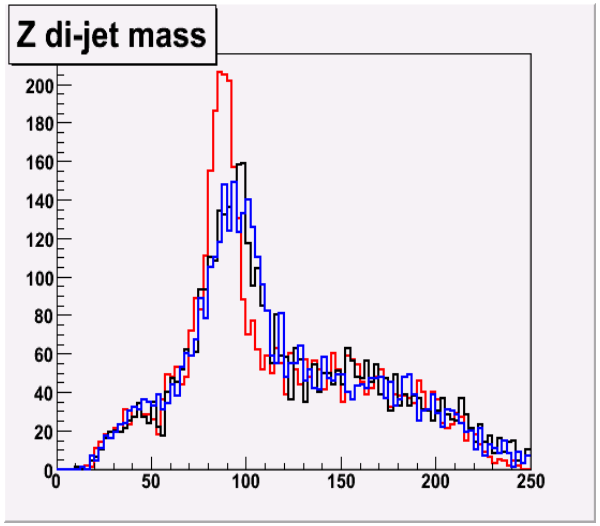
Initial MC production

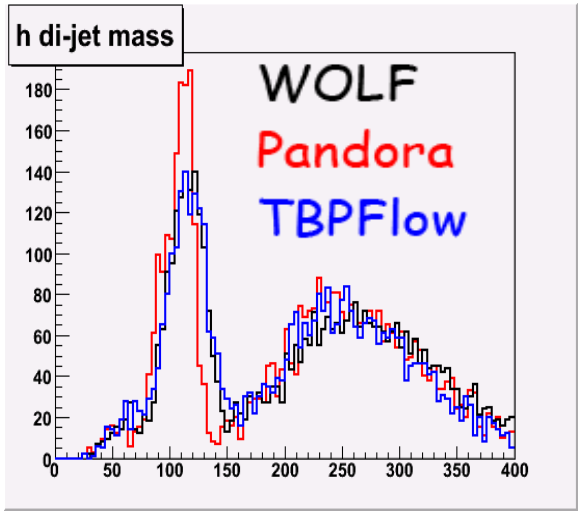
- plan: have central production of LDC/ILD MC files
- need to test the machinery
  - job submission scripts
  - data catalogue (data base)
  - reconstruction code
- start now with some simple events:
  - singles O(10k) of g, e, mu+-, pi+-, K_l, K_s
  - O(10k) Z->uds @ 90,250,500 GeV
  - O(10k) ZH @ 250, 500 GeV
  - both for LDC and LDCPrime
- -> use to put together and configure a std. reconstruction

other PFAs with Marlin









Michele Faucci Gianelli
ZHH analysis

SW- tools:
- Mokka
- MarlinReco/MarlinUtil
- TrackCheater/LDCTracking
- ***PandoraPFA***
- ***Wolf***
- ***TrackBasedPFA***

Katarzyna Wichmann
Higgstrahlung analysis

- modular frameworks allow comparison of different PFA Algorithms
- PandorPFA is the standard reconstruction but
- should also use other algorithms for cross check of detector optimization

Summary

- LDC (ILD) has a mature and easy to use software framework based on Marlin et al
- tools for full reconstruction are there:
  - high performance full Tracking algorithms
  - various PFA algorithms
- with PandoraPFA reached ILC goal for PFA !
- **can use full reconstruction for detector optimization and physics studies for LOI**

Outlook

- need to put everything together and create 'standard reconstruction'
- test and validate
- start 'massive' MonteCarlo production on the grid

additional material

ilc sw-installation

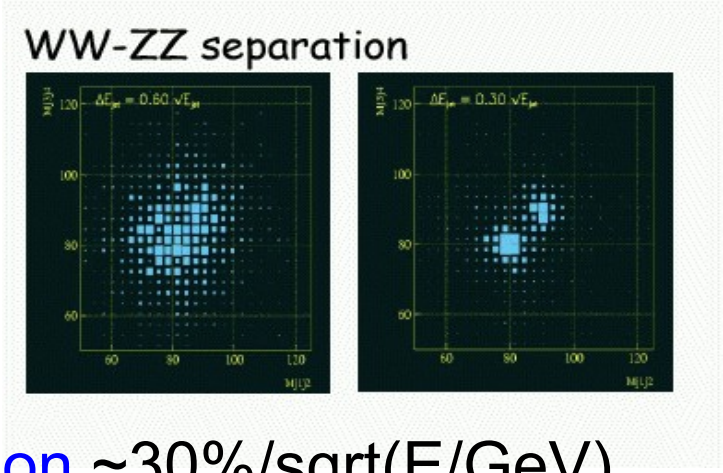
- ilc software requirements and complexity has grown
  - ~30 packages with sometimes optional dependencies
- tool to make installation and build process easier:
- **ilcinstall** (python)
  - script to install all of the LDC software in one go
  - **“start script – go to lunch – run application”**
  - fully configurable:
    - versions, dependencies/build options, links to existing packages/tools, e.g. root, CLHEP,...
  - used for reference installations in afs (SL3/SL4)
  - user can link their packages against these
    - even w/o installing any software on their computer

/afs/desy.de/group/it/ilcsoft/v01-03

released recently

Reconstruction @ the ILC

- general ILC detector features:
  - precision tracking
  - precision vertexing
  - high granularity in calorimeters
    - (Ecal ~1cm, Hcal ~1-5cm)
- important: very high jet-mass resolution ~30%/sqrt(E/GeV)



Particle Flow

- reconstruct all single particles
- use tracker for charged particles
- use Ecal for photons
- use Hcal for neutral hadrons

- dominant contribution (E<50 GeV):
  - Hcal resolution
  - **confusion term**

$$\sigma_{E_{jet}}^2 = \epsilon_{trk}^2 \sum_i E_{trk,i}^4 + \epsilon_{ECal}^2 E_{ECal} + \epsilon_{HCal}^2 E_{HCal} + \sigma_{confusion}^2$$

$$\epsilon_{trk} = \delta(1/p) \approx 5 \cdot 10^{-5}, \quad \epsilon_{ECal} = \frac{\delta E}{\sqrt{E}} \approx 0.1, \quad \epsilon_{HCal} \approx 0.5$$

- **PFA performance determines detector resolution**
- need sophisticated algorithms for **minimal confusion**

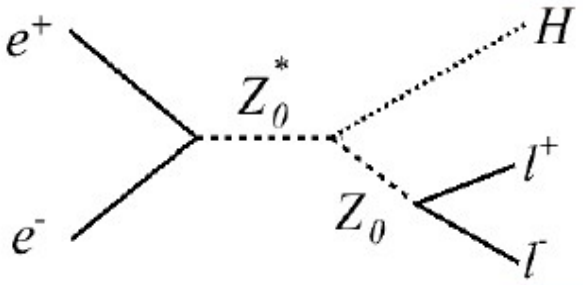
example: Higgs-Recoils analysis

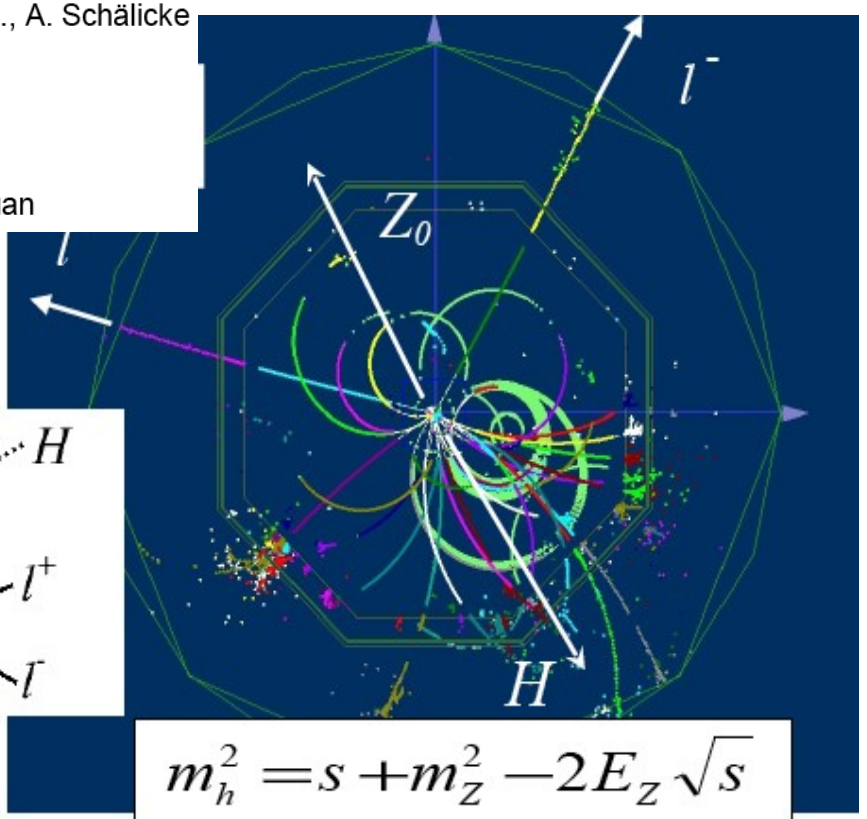
DESY (Zeuthen): W. Lohmann, M.O., A. Schälicke
(Hamburg): K. Wichmann

MPI (München): A. Raspereza

LAL (Orsay): H. Li, R. Pöschl, M. Ruan

- Higgs-Strahlung-Process:





- Higgs-Recoil-Mass

$$m_h^2 = s + m_Z^2 - 2E_Z\sqrt{s}$$

- Coupling Strength (model independent)

$$g_{ZZH}^2 \propto \sigma = N / L\varepsilon$$

Wolf / Pandora ...

⇓

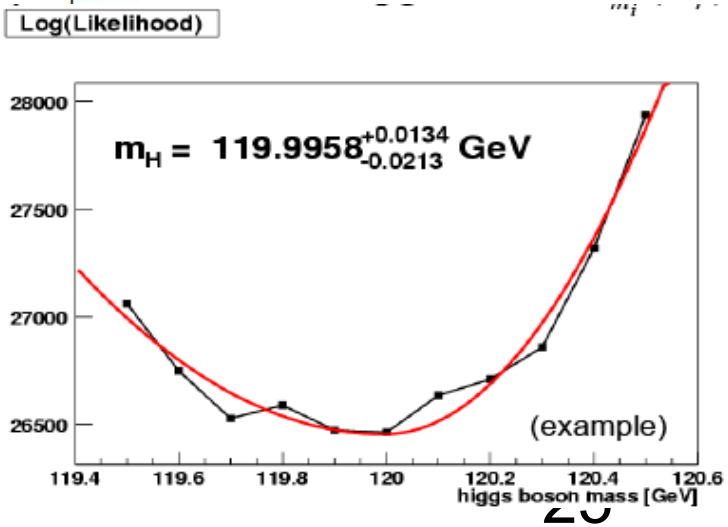
MOID / PFOID

⇓

HiggsRecoil

⇓

Analysis with ROOT

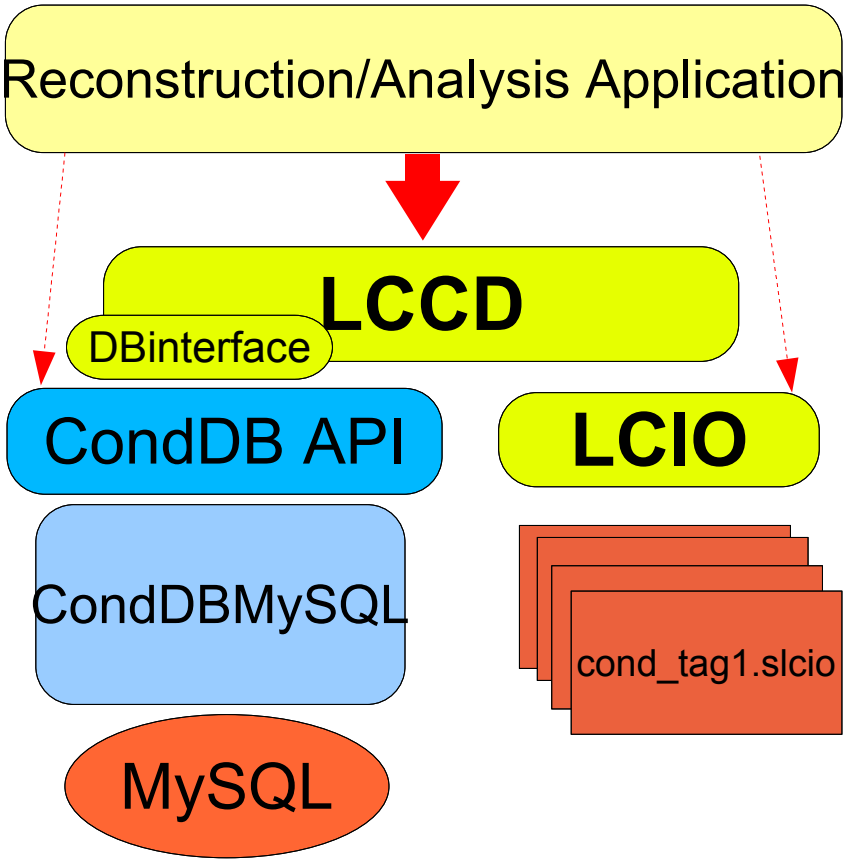


- particle ID code based on Log-Likelihood method developed
- can be used in other Marlin bases analysis

LCCD

Linear **C**ollider **C**onditions **D**ata Toolkit

- **Reading conditions data transparently (same user code):**
  - **from conditions database**
  - **from simple LCIO file**
  - **from LCIO data stream**
  - **from dedicated LCIO-DB file**
- Writing conditions data
- tag conditions data
- Browse the conditions database
  - through creation of LCIO files
    - vertically (all versions for timestamp)
    - horizontally (all versions for tag)



LCCD is used for the conditions data of the ongoing ILC testbeam studies