ILD/AIDA Tracking tools

Status and Plans

Ch. Rosemann

DESY

Linear Collider Software Meeting
31. January 2013

Changes

Personnel Changes in ilcsoft

- Steve Aplin left (tracking and more)
- Jan Engels left as well (installation, packaging, build system and more)
- Oleksandr Volynets joined as DESY fellow
- CR joined for tracking, installation, packaging, ...

⇒ re-organization

CR: new subtask coordinator of AIDA::WP2::Tracking

Introduction

History/Legacy of LoI times

- Model for simulation & reconstruction:
  mixture of simplified and realistic descriptions
- For track reconstruction a patchwork of code was used
- Lots of FORTRAN, lots of common blocks

DBD requirements

- Realistic and accurate detector models are a must
- Inclusion of beam-induced background

No way to continue with the old model

Outline of new tracking system

Rewrite

- Leave behind F77 tracking code (unmaintainable)
- Rewrite TPC pattern recognition
- Use KalTest by K. Fujii as fitting library (Kalman Filter)
- Stand alone reconstruction for inner silicon and forward trackers

Create a project: AIDA common tracking toolkit

More on this later

Current status: what we have

GEAR: Geometry description

Historically grown from API to toolkit

LCIO: Persistency

Clupatra: NN clustering PatRec

KalTest: Navigation and Fitting

Needs KalDet: hit classes and geometry

Abstract layer in between: MarlinTrk

Performance of what we have

The DBD

- Massive amount of simulated and reconstructed events
- In-depth physics analyses – in comparison to LOI analyses
- ilcsoft v01-16 stable and mature production release
- Consolidated software that provides good basis for further development

Construction sites

Working generic tracking package

- MarlinTrk is the first proof-of-concept prototype
- Interfaced to GEAR and KalTest

Geometry interfacing

- GEAR doesn't fulfill every requirement
- dd4hep is successor
- Technical details are being worked out

Silicon/VXD tracking still needs to be sorted out

Requirements of tracking from geometry

Tracking incorporates

- Multiple scattering – gaussian approximation
- Energy loss (depending on particle type)
- Navigation between sensitive layers
- Normal vectors of measurement layers at point of intersection

Needed for this

1. Amount of passive material (e.g. in X0) passed by trajectory
2. Exact crossing point of a trajectory with an active readout surface
3. Normal vector(s) on the measurement surface at the crossing point

What can be delivered by dd4hep?

Abstract interface: IMarlinTrk

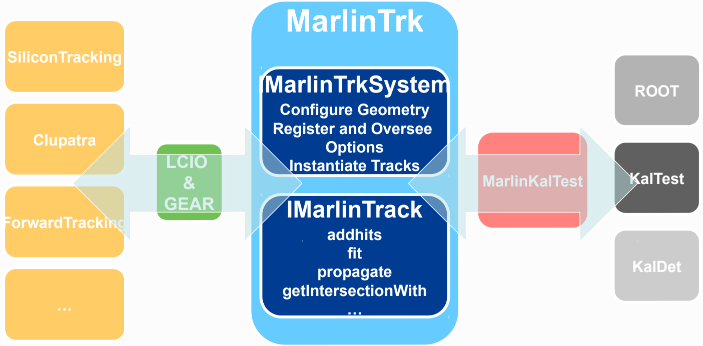
IMarlinTrk

- Interface class to track fitting/parameter manipulation in Marlin
- LCIO as persistency framework

IMarlinTrkSystem

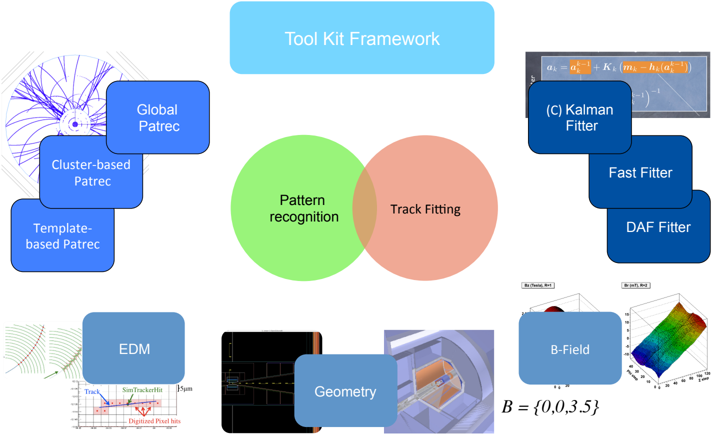
- Infrastructure management
- Interface to geometry (currently GEAR)
- Interface to persistent data
- Configuration control of fitting package

Abstract interface: MarlinTrk



(pic by Steve Aplin)

The goal: a common tracking toolkit



Tool Kit Framework

Global Patrec

Cluster-based Patrec

Template-based Patrec

Pattern recognition

Track Fitting

(C) Kalman Fitter

Fast Fitter

DAF Fitter

EDM

Geometry

B-Field

$B = \{0,0,3.5\}$

(pic by Steve Aplin)

Summary/Outlook

Summary

- ILD has implemented new C++ Tracking
- Pattern Recognition developed against MarlinTrk
- Clupatra, SiTracking, FwdTracking and FullLDCTracking
- Modular approach allows replacement of elements and extension to additional algorithms, e.g. AllSiliconTracking (SID)
- Eventual goal (within AIDA): generic tracking toolkit

Outlook

- Adopt DD4Hep soon
- Open for suggestions for improvements !