Software for ILD detector optimization

Status, plans and timeline

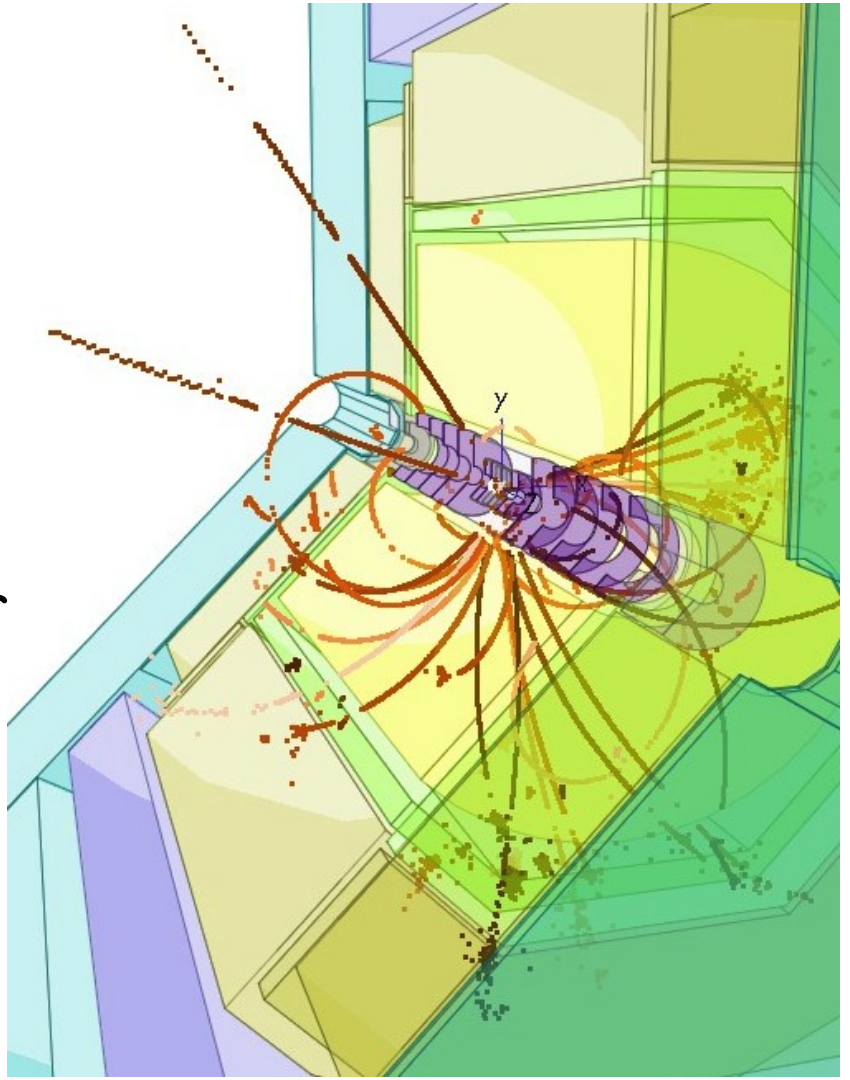
Frank Gaede, DESY

ILD Software and Analysis Meeting

June 25, 2014

Outline

- Introduction
- Recent developments and plans
  - DD4hep, DDSim, aidaTT
- Grid production
  - ILCDirac
- Towards a software timeline for ILD optimization
- Summary/Discussion



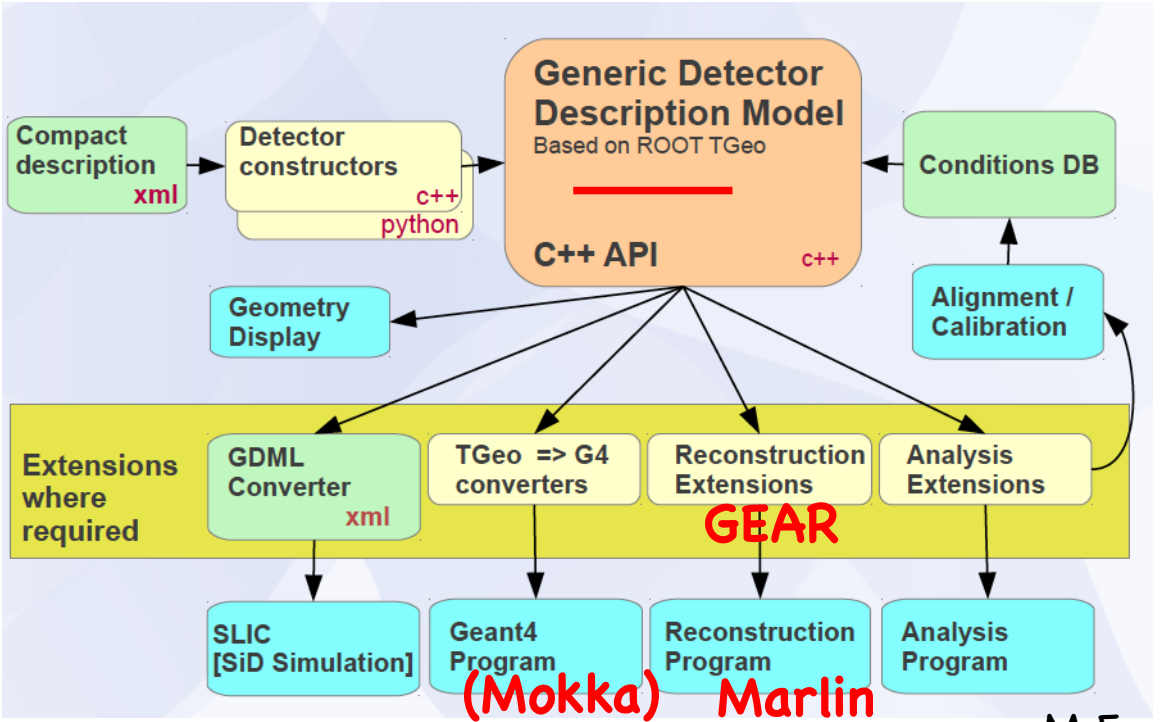
Introduction

- general agreement in LC community to move to (more) common software tools
- closeout of 2012 LC-Software Meeting:
  - a common simulation application based on the geometry description developed in AIDA WP2
  - a common C++ tracking package in the context of AIDA WP2
- at 2012 LC-Software Meeting discussion focused on the details of how these goals can be achieved
  - interface between geometry description and simulation
  - interface to reconstruction (tracking)
  - -> decision to develop prototypes to investigate options
- agreement to use DD4hep as the common geometry tool
- agreement to use SLIC as common simulation tool

  (see later this talk)

DD4hep - overview

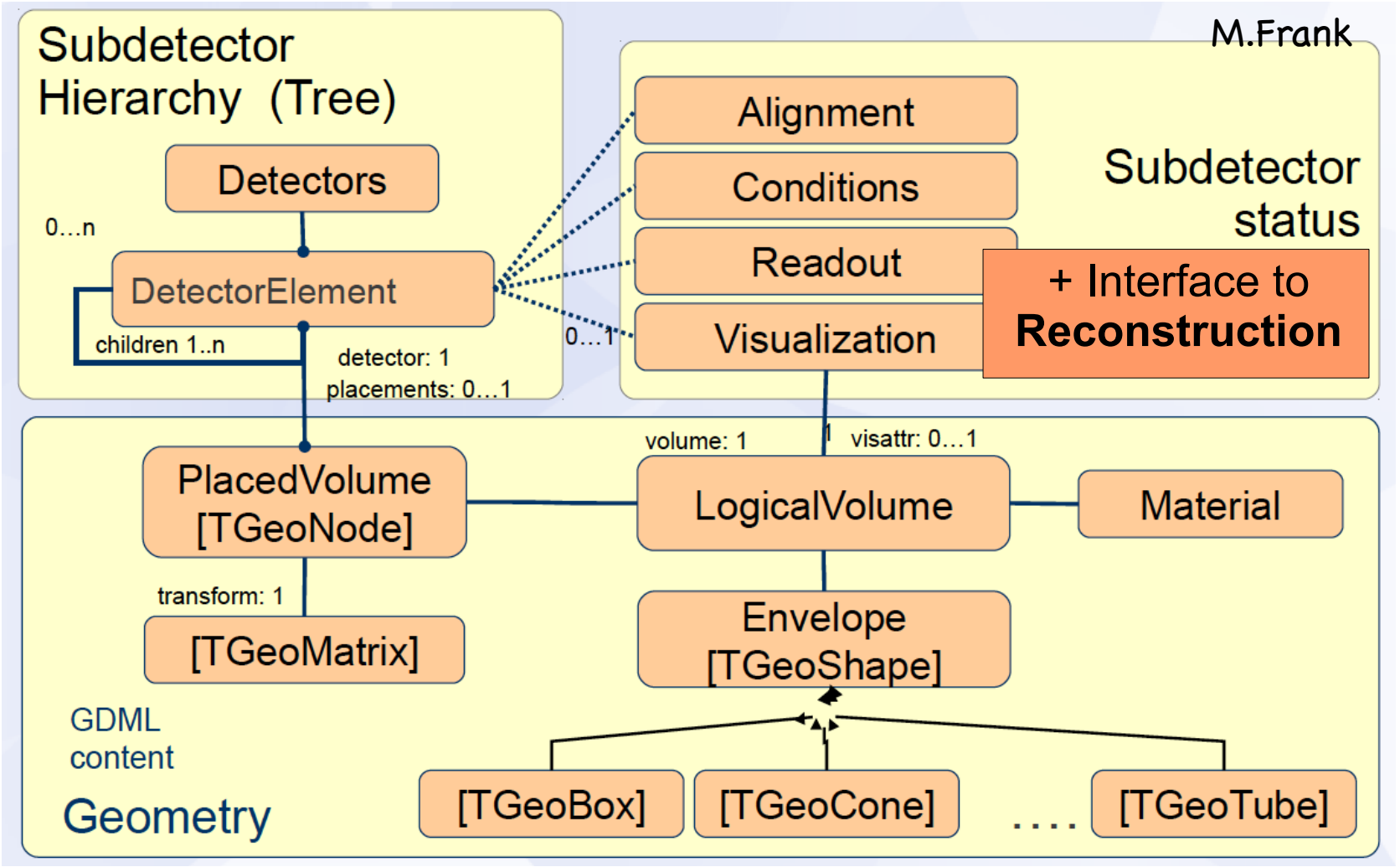
- goals for DD4hep:
  - full detector description
  - full experiment life cycle
  - consistent description
  - ease of use
- DD4hep is based on best concepts from existing geometry tools:
  - xml files with parameters and compact description
  - C++ drivers per sub detector
  - provide several interfaces to simulation, reconstruction and analysis programs



M.Frank

- **DD4hep** common detector geometry description
  - developed by CERN-SFT/LDC (+DESY) in AIDA WP2
- options for geant4 simulation:
  - use **SLIC** simulation via GDML export
  - write new simulation in DD4hep/DDG4

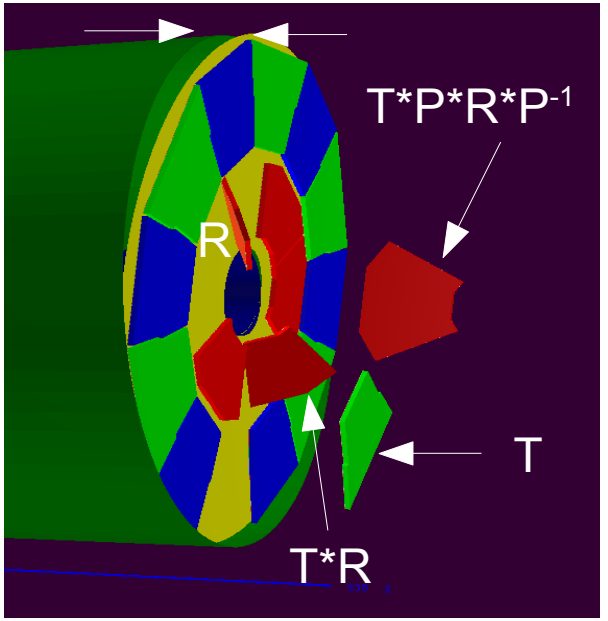
DD4hep geometry implementation



- DD4hep uses the TGeo geometry classes to instantiate geometry tree -> can use all TGeo features directly
- additional (user) code added to detector element class

DD4hep Alignment

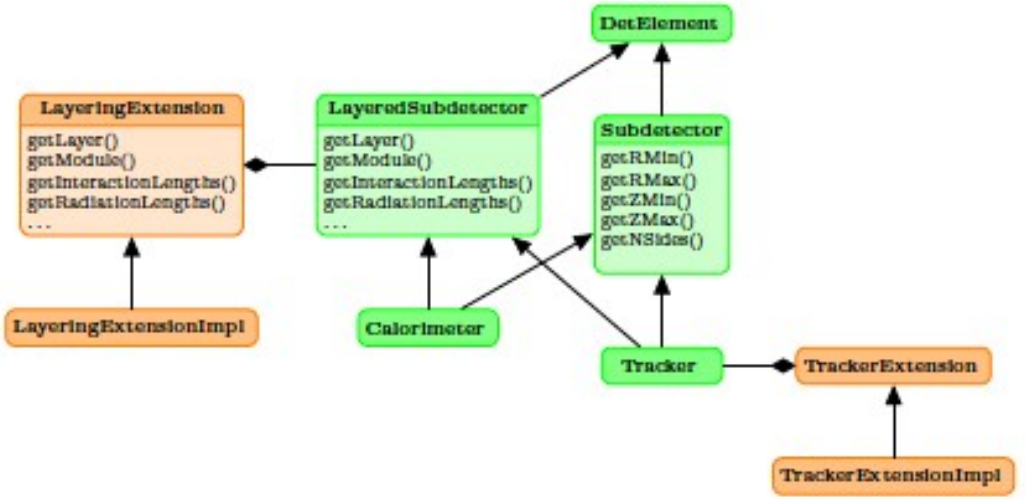
- describing mis-aligned detector components in simulation important for running experiments and test beams
- DD4hep has now a mechanism that allows delta mis-alignment conditions data to be applied to ideal geometry
- first implementation exists - experimental code so far
- could use this to develop an alignment strategy for ILD tracking system and study it with full simulation

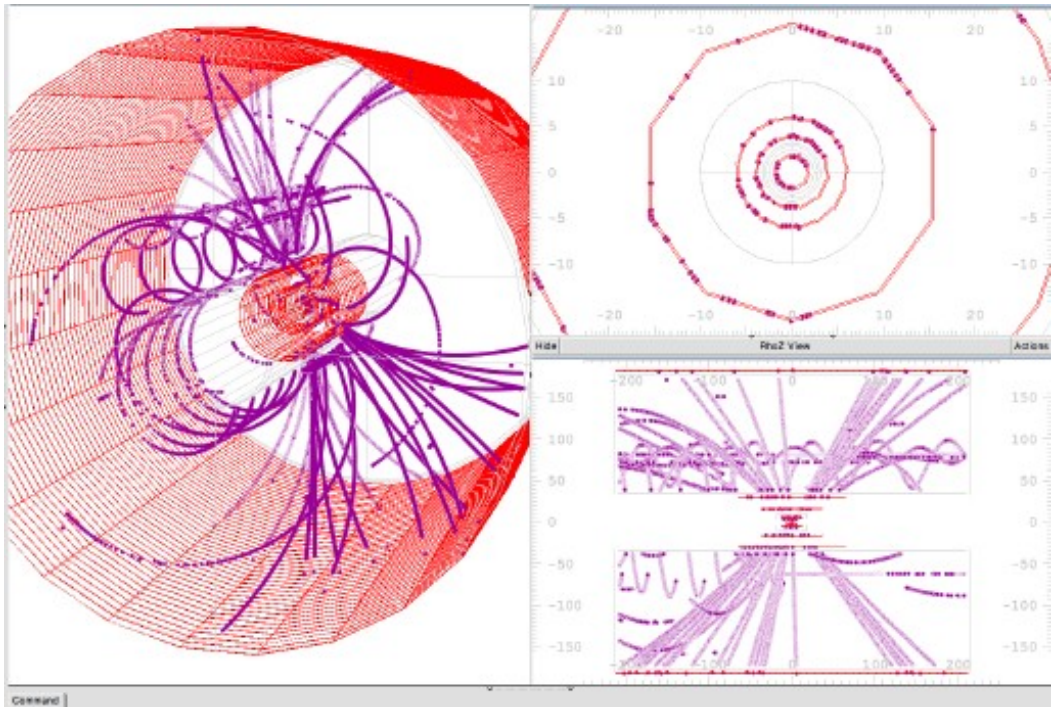


example: Aleph TPC
(misalignment exaggerated)

DD4hep interface to reconstruction

- extension mechanism is used to define interface for reconstruction
- calorimeters and trackers defined as LayeredSubdetectors
  - use to eventually replace GEAR
  - work in progress ...
- for tracking additional Surfaces provide:
  - u,v,normal and origin
  - inner and outer (averaged) material incl. thickness
- -> planes and cylinders allow for simple navigation in detector geometry for the tracking





DD4hep interface to SLIC

- compact geometry description in DD4hep inspired by the compact format used in org.lcsim
- export of detailed simulation geometry as LCDD files - extension of GDML as input to SLIC:
  - materials, solids, logical, placed and physical volumes
  - limits, regions fields
  - sensitive detector information via name lookup
- SLIC does not depend on DD4hep
  - common package DDSegmentation used as bridge
  - potential issue for more complex sensitive detectors

  currently under study ...
- SLIC is fully functional and used for SiD mass productions - however, limited man power available at SLAC to support binding to DD4hep

DDG4 - built in Geant4 gateway

- in memory conversion of TGeo geometry to Geant4 geometry
- plugin mechanism for
  - sensitive detectors, Geant4 user actions : stepping, tracking,...
  - input (generator files) and output (LCIO,...)
- configure mechanism with xml, python or CINT:
  - physics lists, limits, fields,...
- simple examples exist
  - very simplified sensitive detectors, creation of LCIO hits
- advantage:
  - full flexibility in sensitive detectors
  - can use extension code in simulation and reconstruction
  - supported by CERN for FCC and possibly CLICdp

- ILD needs to decide rather soon whether to use SLIC or DDG4 (or sth. in between) -> discuss at ILD meeting

porting Mokka models to DD4hep

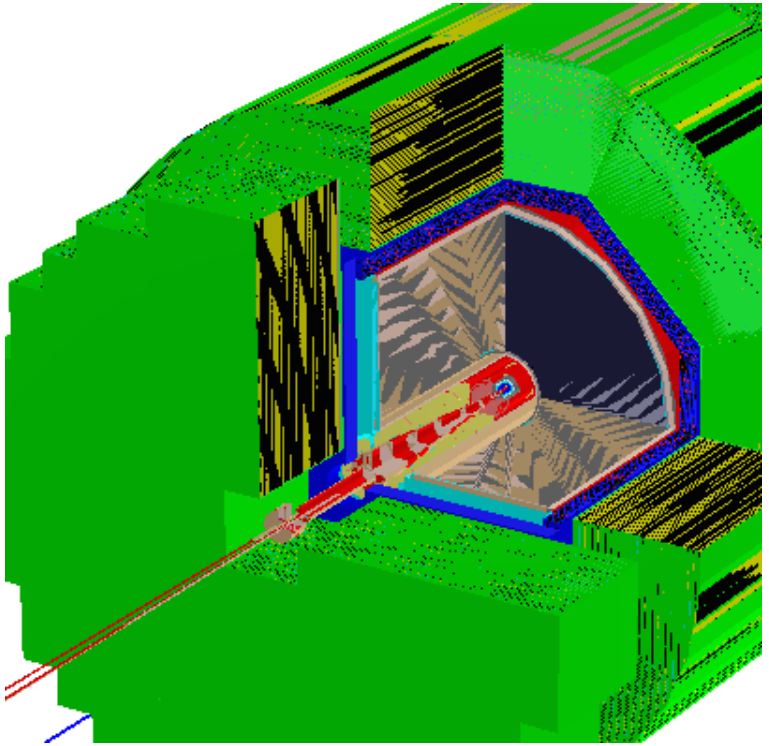
- ILD, CLIC, Calice and others use Mokka as simulation tool
- huge amount of man power has gone into developing and debugging detector simulation models
- -> this work needs to be preserved
- try to replace DB access with xml parsing of parameters and port code with as little changes as necessary:
  - G4Shape -> Shape
  - G4LogicalVolume -> Volume
  - PVPlacement -> PlacedVolume

https://svnsrv.desy.de/viewvc/ddsim/DDSim/

- created package DDSim to eventually preserve all current Mokka models - start with ILD_o1_v05
- scripts to create xml files with parameters from Mokka DB
  - per sub detector and global/model parameters
- simple wrapper class to transparently replace parameter access from DB to XML
- line-by-line port of geometry placements:
  - mostly straight forward
  - sometimes parameters in c'tors are (slightly) different

ILD_o1_v05 in DDSim

- started to port sub detectors from Mokka model ILD_o1_v05:
- VXD, FTD, SIT, TPC, SET, beam pipe (F.G)
- Ecal-barrel, Hcal-barrel, endcaps ongoing (Sh.Lu)
- Beamcal (A.Sailer)
- so far ported only one sensitive detector:
- TRKSiSD00

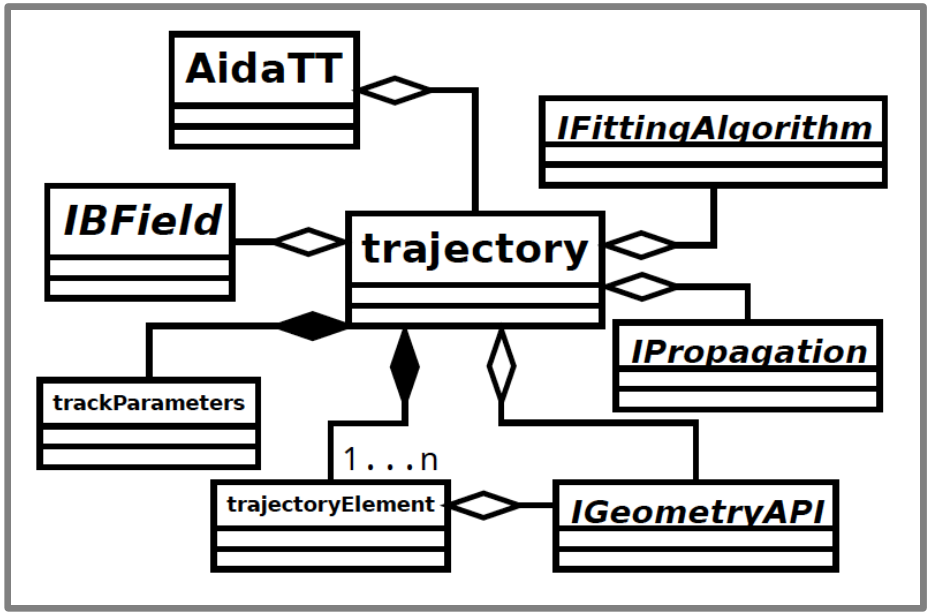


- To Do:
- reconstruction interface - GEAR ?
- implement missing sub-detectors
  - Coil, Yoke, Lcal, LHcal
- implement sensitive detectors
  - (SLIC vs. DDG4)
- test and validate everything
- => major effort for ILD
- => need dedicated experts for every sub-detector

aidaTT - overview

C.Rosemann

- tracking tool developed in AIDA WP2
- Design criteria:
  - completely modular
  - well defined API to reco frameworks
  - separation of data and algorithms
  - parallelization on track level
- features:
- transparently use Kalman-Filter or GeneralBrokenLines (needed e.g. for Millipede alignment tool)
- interface to DD4hep geometry
  - defined surfaces in detector (simulation) geometry model
  - navigation and material effects based on surfaces

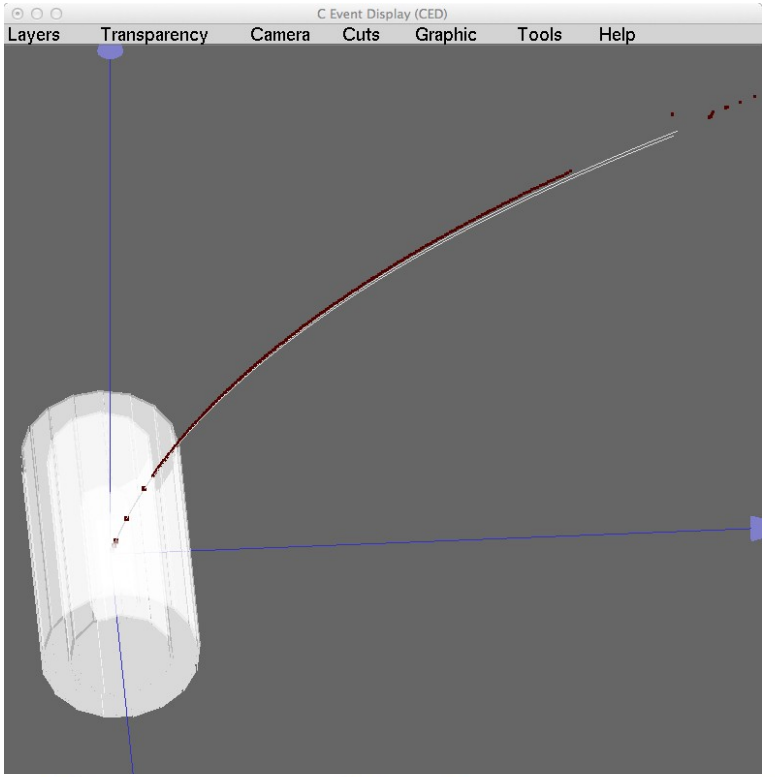


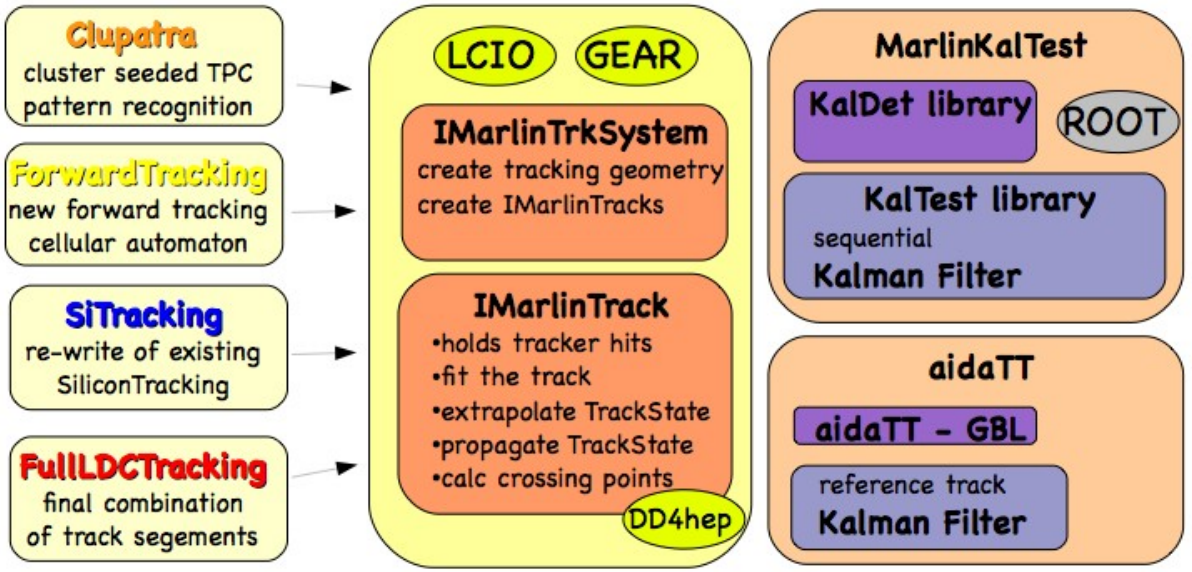
https://svnsrv.desy.de/desy/aidasoft/aidaTT/trunk

aidaTT Status

C.Rosemann

- implementation of GBL is basically ready
  - single muon track - fitted in VXD of DD4hep ILDExDet
  - uses reference track
- will need some debugging and further testing...
- next steps: implementation of MarlinTrk for pattern recognition and navigation...





ILD Monte Carlo production

- in transition phase from GridProd system (J.Engels) used for DBD mass production to ILCDirac - maintained by CLICdp group
- Shaojun has taken over role of ILD Monte Carlo production coordinator from Eduard
- ILCDirac is already used successfully by individual users for their specific physics sample production
- central Monte Carlo production is much more involved:
  - bookkeeping of samples to process, data catalogue, load sharing between Grid sites, ...
- ongoing task to adapt CLIC production scripts to ILD
  - will use new cvmfs software installations on the GRID
  - support from KEK for this task (T.Calancha)

Towards a timeline for software I

- ingredients and missing items needed for defining a timeline for ILD software development:
  - need to have first functional version of ILD_o1_v05 in DD4hep/DDSim
  - need functional interface to reconstruction (GEAR)
  - need testing and validation
  - define the ILD optimization models - how many (2-3) ?
    - smaller detector, technology options, geometry options, ...
  - implement these models
  - need testing and validation
  - define the physics benchmarks/data samples that need to be processed
    - 250 GeV, 350 GeV, 500 GeV full SM ?
  - finalize the Grid production infrastructure w/ ILCDirac
  - adapt reconstruction to new models
  - need testing and validation
  - estimate the CPU (and storage) needs
  - the actual Monte Carlo mass production
  - ...

Towards a timeline for software II

- a first rough estimate of the effort involved:

item:	estimated effort*	comment
first version of ILD_o1_v05 in DD4hep	1 pm	by ILD meeting ?
interface to reconstruction	1 pm	
testing and validation	2 pm	
define ILD optimization models	1 pm	start at ILD meeting
implement these models	3 pm	# models ?
testing and validation	3 pm	# models
define physics benchmarks	1 pm	
Grid production infrastructure	1 pm	
Grid Monte Carlo simulation	1 pm (3 months)	# channels/processes, # CPUs
adapt reconstruction (incl. testing)	2 pm	
Grid reconstruction	1 pm (1 month)	
Total	18 pm	

*pm: full time person month

the actual calendar time that is needed, depends on the number of (experienced) people that are available for this

Summary & Outlook

- ILD will use DD4hep geometry description for simulation and reconstruction
- a first version of ILD_o1_v05 in DD4hep is on its way
- need to decide which simulation application to use/develop
- to develop a time line for the software work for ILD optimization need to decide:
  - on number and layout of ILD optimization models
  - physics benchmarks and samples to be created w/ these models
  - need to implement, test and validate these models
  - identify people that can contribute to the effort
- => should have discussion and decisions at the ILD meeting in September
- -> can start the discussion already now ...