Summary of the WP2 Session

F. Gaede/DESY, P. Mato/CERN

AIDA 2nd Annual Meeting, Frascati, 10-12 April 2013

WP2 Goals and Tasks

- Goals
  - Develop core software tools that are useful for the HEP community at large and in particular for the next big planned projects: sLHC and Linear Collider (ILC/CLIC)
- Tasks
  - Task 2.1: Coordination and communication
  - Task 2.2: Geometry toolkit for HEP
  - Task 2.3: Reconstruction toolkit for HEP
- Subtask coordinators
  - Geometry - Gabriele Cosmo, CERN
  - Tracking - Steven Aplin, DESY --> Christoph Rosemann, DESY
  - Particle Flow - Mark Thomson, UCam
  - Alignment - Chris Parkes, UniGla
  - Pile-up - Lucia Silvestris, INFN

https://aidasoft.web.cern.ch

Reminder: Deliverables

Deliverable Number [61]	Deliverable Title	Partners (lead beneficiary)		e [62]	Dissemination level [63]	Delivery date [64]	
D2.1	Project web infrastructure to document software packages	CERN	5.00	O	PU	3	done
D2.2	Central code repositories and other infrastructure required for the software development	DESY	5.00	O	PP	4	done
D2.3	Software design for geometry toolkit including the interfaces for the reconstruction toolkits	CERN, DESY, LLR, UniGla, STFC			PU	12	done
D2.4	Software design for tracking toolkit	DESY, CERN, OeAW, KFKI				12	done
D2.5	Software design for PFA tools	Ucam, LLR, CERN,			PU	12	done
D2.6	Design for handling the pile-up in sLHC	INFN, NTU, KFKI			PU	30 ~~17~~	next
D2.7	Software toolkit for detector geometry, materials and detection technologies	CERN, DESY, LLR, UniGla, STFC			PU	38	
D2.8	Software toolkit with tracking algorithms	DESY, CERN, OeAW, KFKI				38	
D2.9	Particle Flow software tools	Ucam, LLR, CERN,			PU	38	
D2.10	Alignment tools software tools	UniGla	30.00	O	PU	38	
D2.11	Trigger simulation software tool	STFC	20.00	O	PU	38	

Reminder: Milestones

Milestone number [59]	Milestone name	Partners (lead beneficiary)		Comments	
MS10	Running first prototype of the particle flow algorithm.	Ucam,LLR,CERN	10	Application to LC detector (Task 2.3)	done
MS11	Running prototype of tracking toolkit including some algorithms	DESY	18	Application to ILD-TPC simulation (Task 2.2)	done
MS12	Running prototype of the geometry toolkit	CERN, DESY, LLR	26	Application to ILD detector simulation (Task 2.2)	next
MS13	Running prototype of the tracking code for the pile-up	INFN, NTU, KFKI	26	Application to sLHC simulation (Task 2.3)	next
MS14	Integration of tracking toolkit into LC softwa framework	DESY, CERN, OeAW	44	Validation of physics performance (Task 2.3)	
MS15	Application of PFA tools to sLHC detectors	Ucam, LLR	44	Demonstration of concept (Task 2.3)	
MS16	Application of alignment tools to sLHC	UniGla	44	Validation of performance (Task 2.3)	
MS17	Integration of pile-up tracking code in sLHC software frameworks	INFN, NTU, KFKI		Validation of tracking efficiency (Task 2.3)	

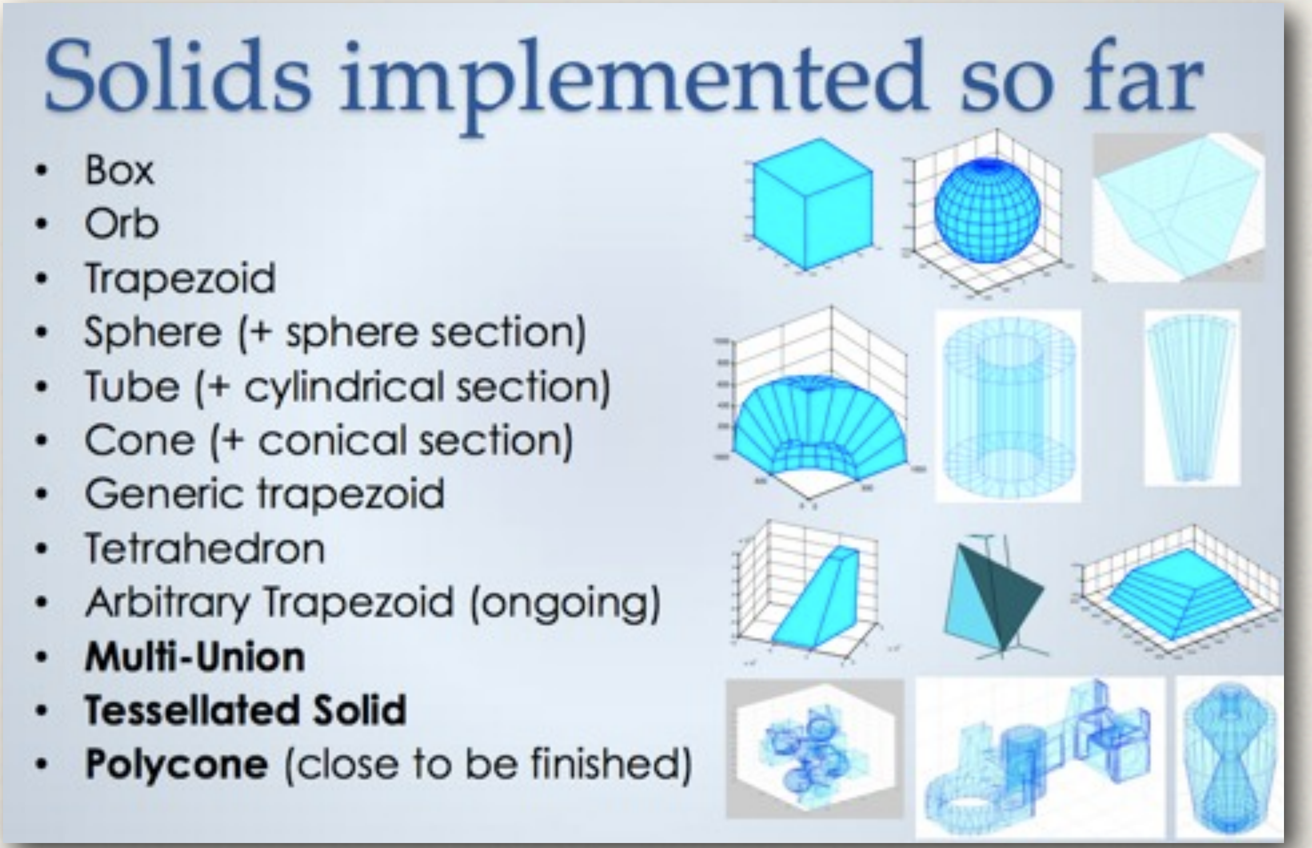
Agenda of WP2 Session

10 talks covering all tasks and sub-tasks of the WP2 work package

	Introduction	Frank-Dieter GAEDE	Aula Seminari, LNF Bldg 36	09:30 - 09:40
	Development status of USolids Library	Marek GAYER	Aula Seminari, LNF Bldg 36	09:40 - 10:00
10:00	Status of DD4Hep	Markus FRANK	Aula Seminari, LNF Bldg 36	10:00 - 10:20
	New Developments for Mokka	Vincent BOUDRY	Aula Seminari, LNF Bldg 36	10:20 - 10:40
	Status of Pile Up Task	Lucia SILVESTRIS	Aula Seminari, LNF Bldg 36	10:40 - 11:00
11:00				
	New Developments for Particle Flow	Prof. Mark Andrew THOMSON et al.	Aula Seminari, LNF Bldg 36	11:30 - 11:50
	Status of Tracking task	Christoph ROSEMANN	Aula Seminari, LNF Bldg 36	11:50 - 12:10
12:00	Status of (LHCb) Alignment	Christoph HOMBACH	Aula Seminari, LNF Bldg 36	12:10 - 12:30
	Clustering, energy loss, and vertexing tools for Si detectors	Ferenc SIKLER	Aula Seminari, LNF Bldg 36	12:30 - 12:50
	tkLayout - A Tracker Layout Modeling Tool"	Dr. Jelena ILIC	Aula Seminari, LNF Bldg 36	12:50 - 13:10
13:00	PFA activities at LLR	Manqi RUAN et al.	Aula Seminari, LNF Bldg 36	13:10 - 13:30

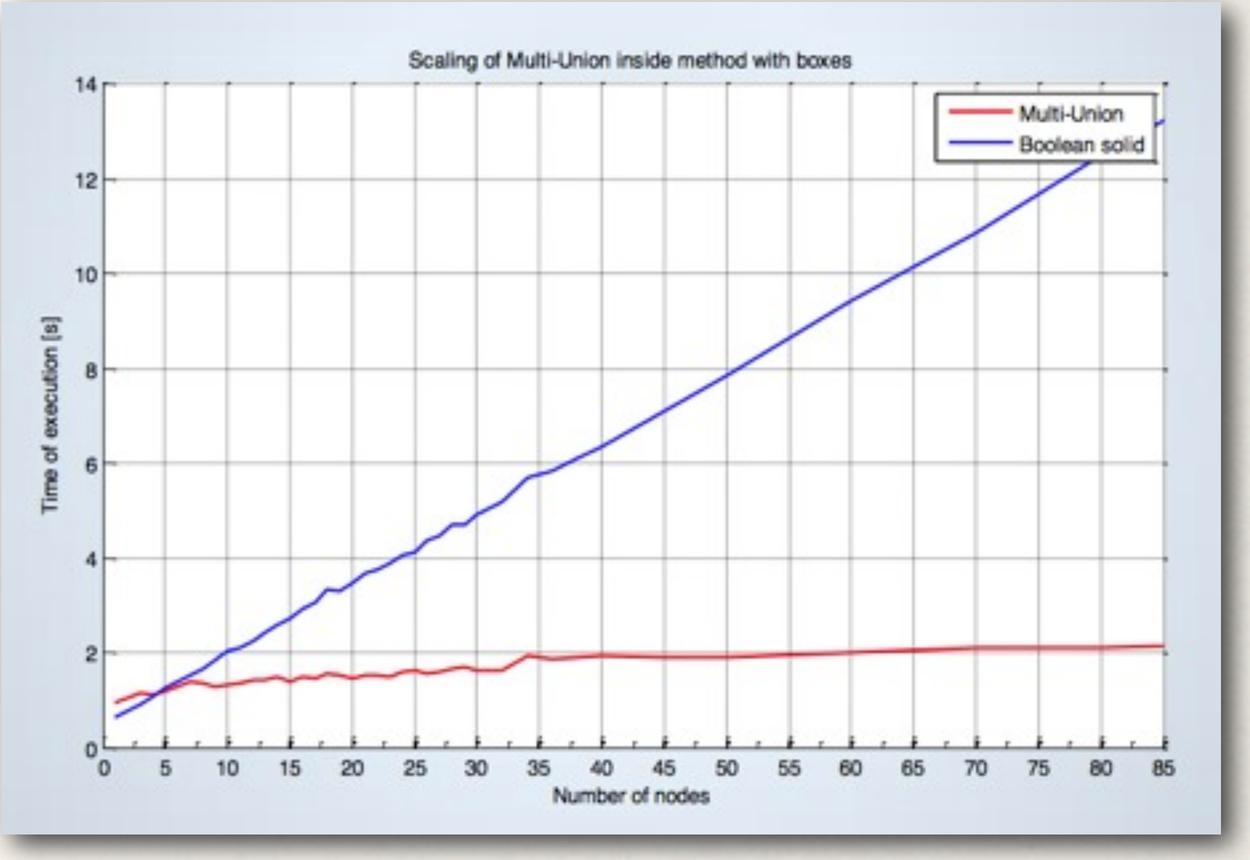
Unified Solids

- Goals
  - Optimize and guarantee better long-term maintenance of ROOT and Geant4 solids libraries (about 70-80% of code investment for the geometry modeler concerns solids)
  - Get the best of each implementation
- Huge work done evaluating CPU performance, precision and robustness (stress testing) of each basic 3D shape
- In particular:
  - Multi-union
  - Tessellated Solid
  - Polycone



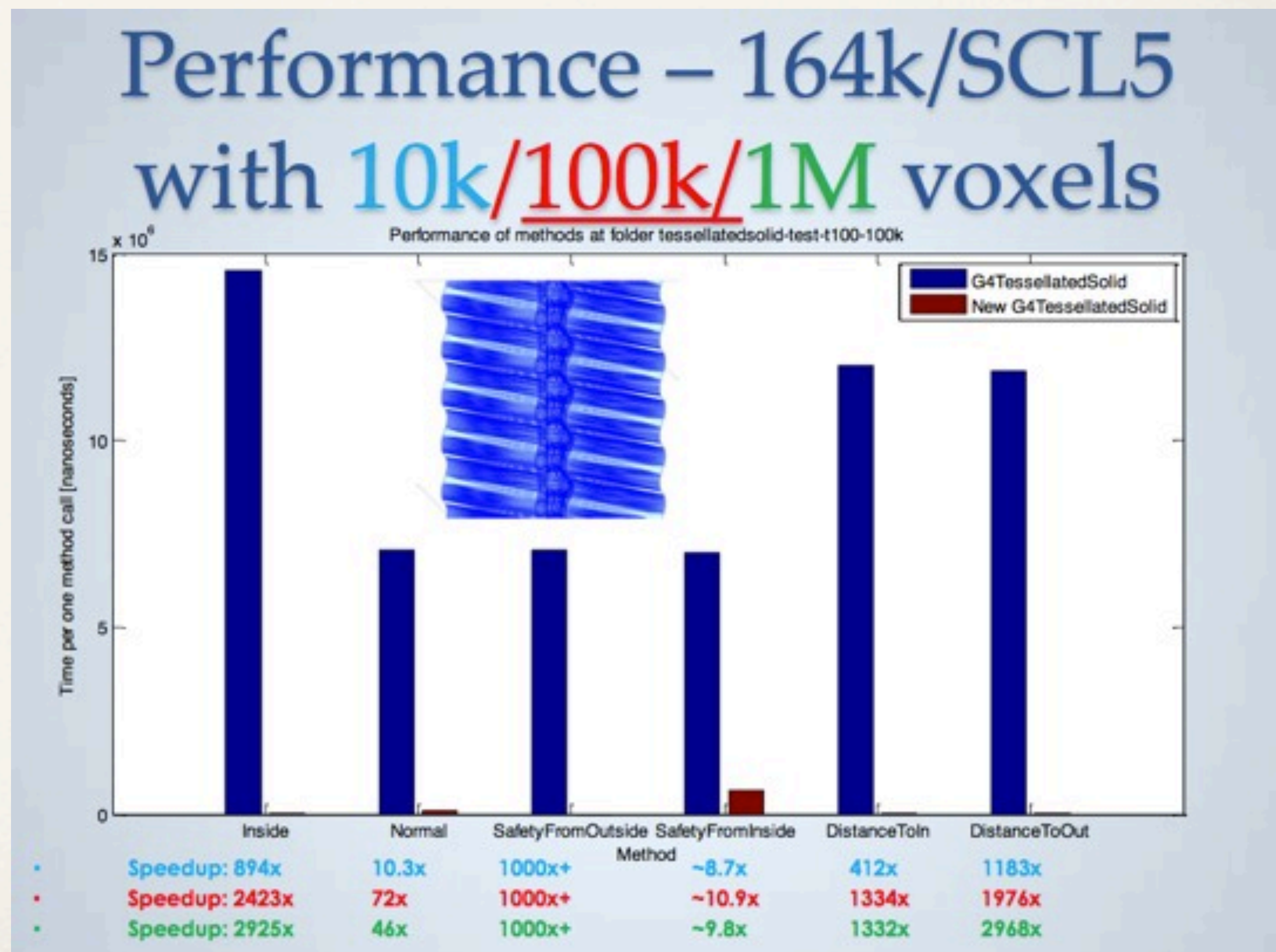
Multi-Union

- Existing CSG Boolean solids (ROOT and Geant4) are represented as binary trees
  - Linear performance
- Implemented a new solid as a union of many solids using voxelization technique to optimize speed and scalability
  - 3D space partition for fast localization of components
  - Aiming for a log(n) scalability
- Useful for complex composites made of many solids



Tessellated solids

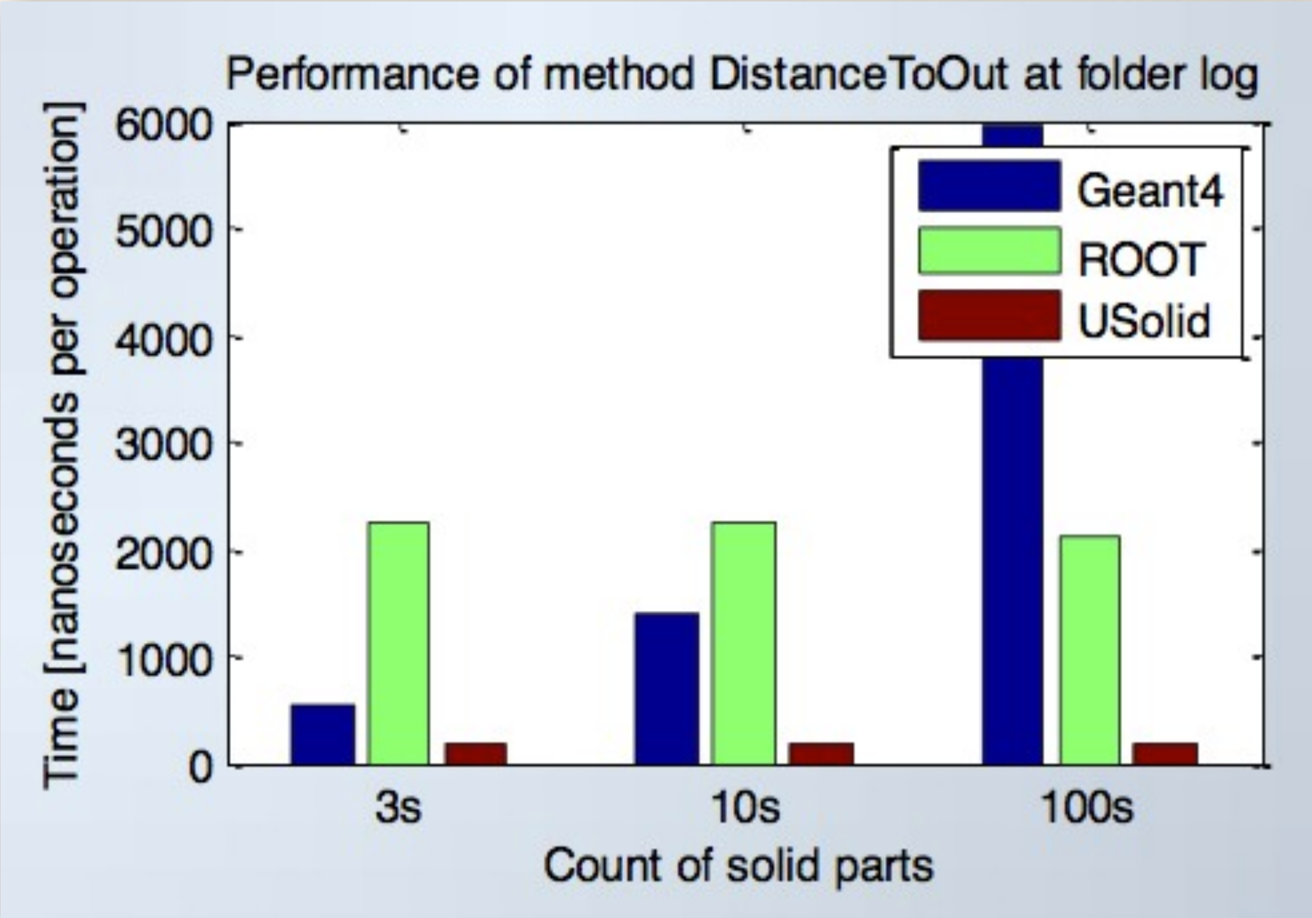
- Connected triangular and quadrangular facets forming solid
- Old implementation was slow, no spatial optimization
- We use spatial division of facets into 3D grid forming “voxels”
- Factor thousands faster for LHCb provided foil test case (164k facets)
- Implemented in Geant4 9.6 as G4TesselatedSolid
- Total memory save is about ~50%

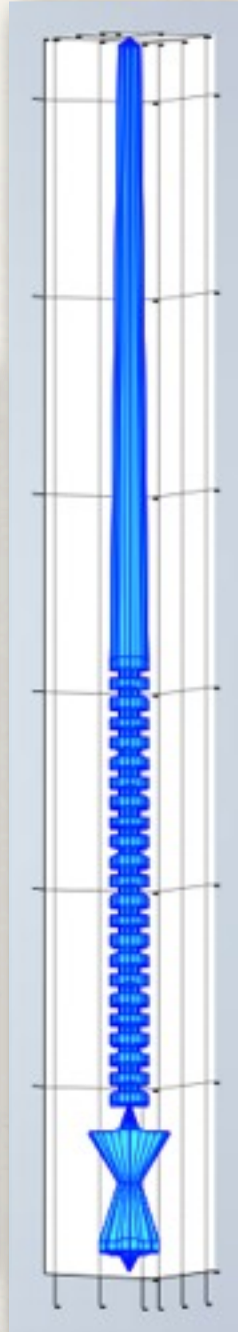


Polycone

- Polycone is an important solid, heavily used in most experimental setups
- Implementation based on composition of separate instances of cones, tubes (or their sections)
- Significant performance improvement over Geant4 & ROOT

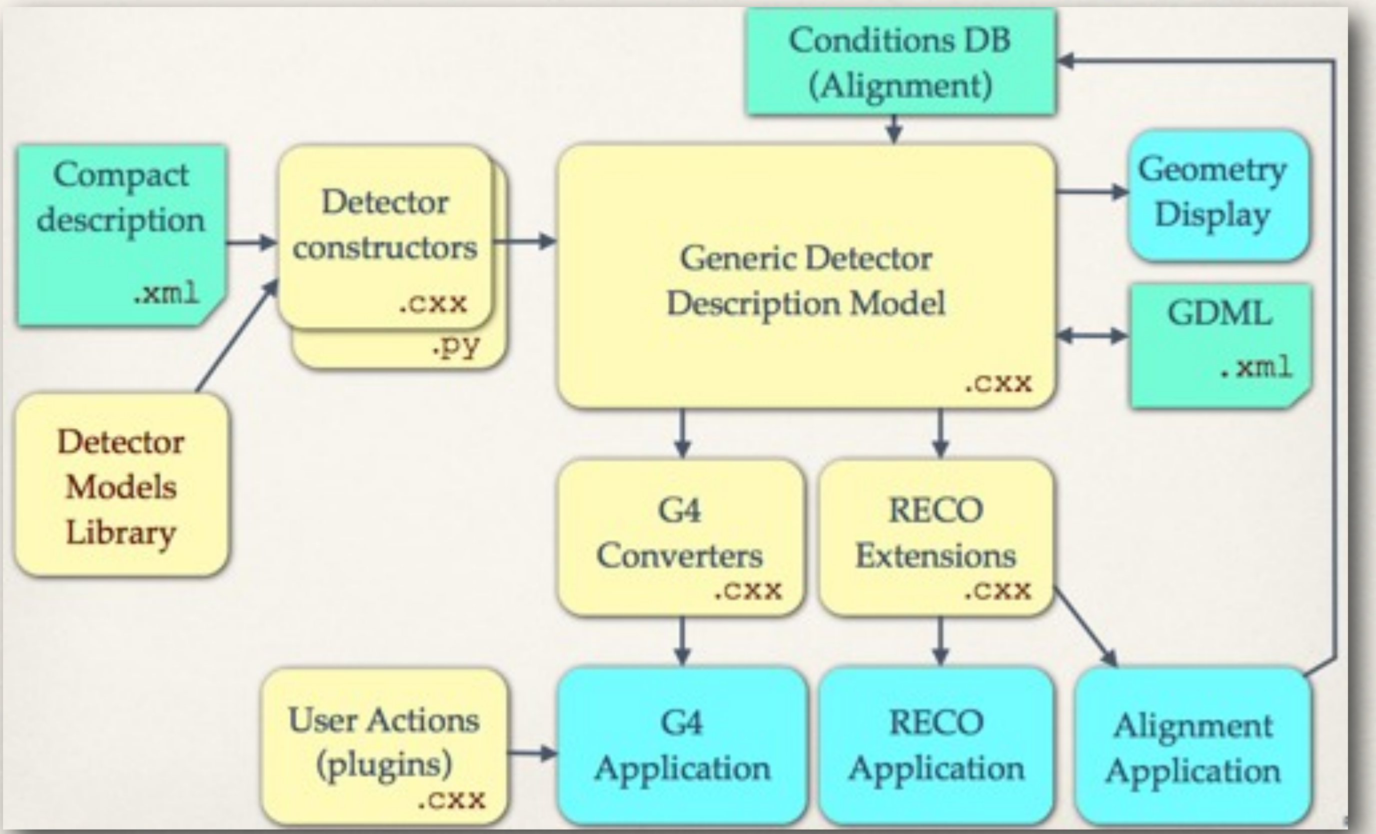
Work will continue with the remaining most critical solids
Change over





DD4Hep

- Development of detector description toolkit
  - **For the full experiment life cycle** (design, construction, operation)
  - **Consistent description with single source** (simulation, reconstruction, analysis)
  - **Full description** (geometry,. readout, alignment, etc.)
- Main components of the toolkit and design choices
  - Detector Element
  - Compact descrip. (XML)
  - Based on TGeo (ROOT)
  - Converters and adaptors
  - Data + Behaviour



DD4Hep: Reconstruction Extensions

- Client ‘extensions’ and specialized ‘views’
  - Customize a Detector Element object to support specific reconstruction questions and cached data
- Example: TPC (A.Muennich)
  - The “PadLayout” is retrieved from the detector element if present
- Adding an extension is possible anywhere
  - Extensions are not confined to detector constructor
  - Could also be somewhere in the reconstruction code

Geant4 Gateway

- Conversion of the Generic Geometry to Geant4 geometry. Two approaches:
  - **In Memory Translation to Geant4**
    - This processing chain was implemented
    - Requires additional development of *sensitive detectors*
  - **Generate GDML** (with lcdd extensions) to be compatible with SiD simulation application 'slic'
    - Approach taken by LCD at the last software workshop
    - Re-use treatment of *sensitive detectors*
    - Mokka (LCD simulation) would need to be migrated

Adaptation of Mokka to DD4Hep

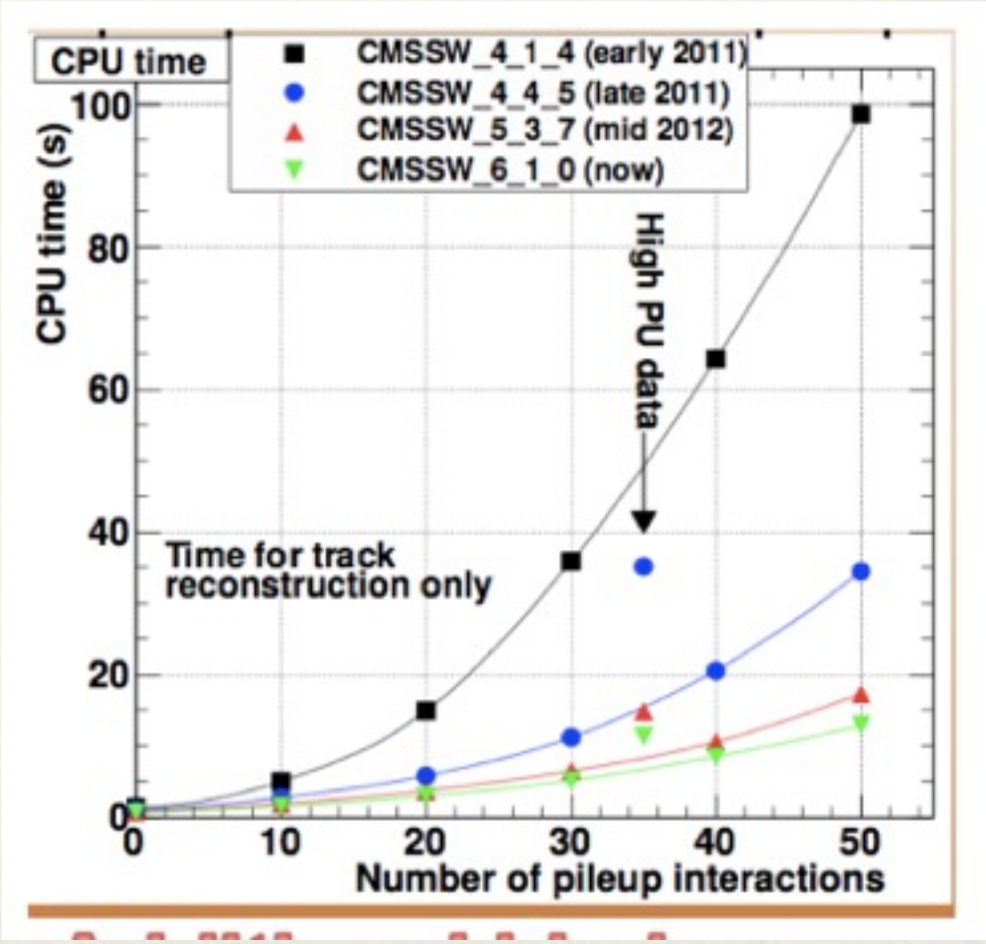
- Mokka detector parameters stored a MySQL DB
  - Many sub-detectors, versions and options (~323 versions in total)
  - Need to extracted and formatted in DD4Hep XML files
- Mokka 'drivers' to construct detectors in Geant4
  - Need to be translated into 'DD4Hep" detector constructors (Python or C++ code snippets)
  - Collaborative effort required (identify contact people)
- Review of sensitive detectors
  - Collaborative effort required (identify contact people)

Pile-up Sub-task

- Development of a toolkit to handle high multiplicity events
  - Main motivation:
    - pile up in sLHC will increase with respect to "nominal (2011) LHC"
    - requires a more efficient way to manage high particle multiplicity events (CPU and Memory)
  - Aiming for a set of generic and reusable algorithms/functions
- In CMS enters in **Mixing Module** (Simulation) and **Tracking Module** (Reconstruction)
  - So far the work has been based on CMSWS (CMS Framework)

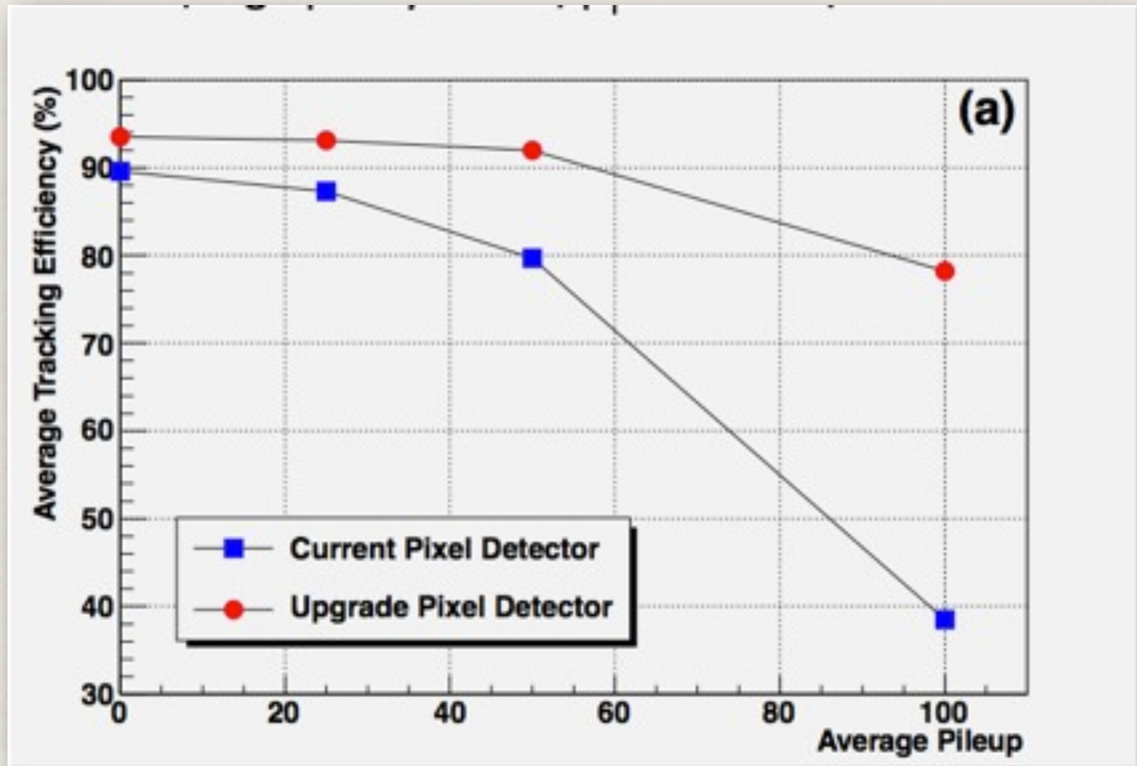
CMS Improvements 2011-2013

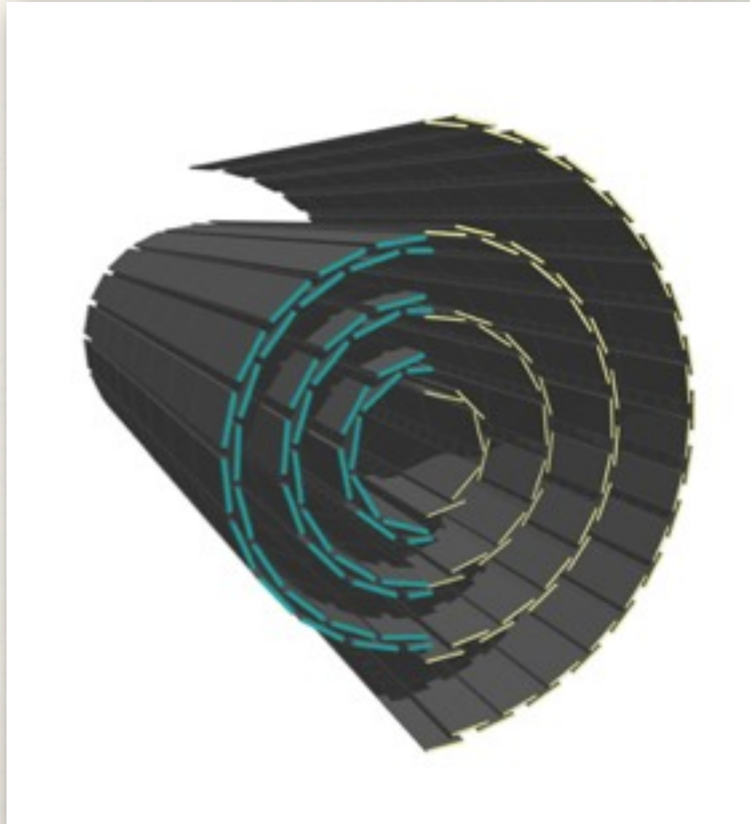
- Mixing Module
  - Mixing module superimpose pileup events to a signal event
  - Memory limited: max 300 events with 2GB/core
  - Keep the memory under 2 GB/core with 140 pile-up events and 5 bunch crossing (700 events)
- Tracking
  - CPU powerN increase with number of pileup interactions
  - Several optimization campaigns has bring the CPU to reasonable levels



CMS Strategies for 2015 and Phase 1

- **Generic improvements** as in 2011/2012 (smarter coding, compilers, seed cleaning) and iterative tracking tuning | tracking developers
- Tracking code reengineering; major redesign of the tracking code to implement **parallelization and vectorization**
- **New tracking algorithms** (Hough transform)
- Work done CMS Pixel @ Phase 1



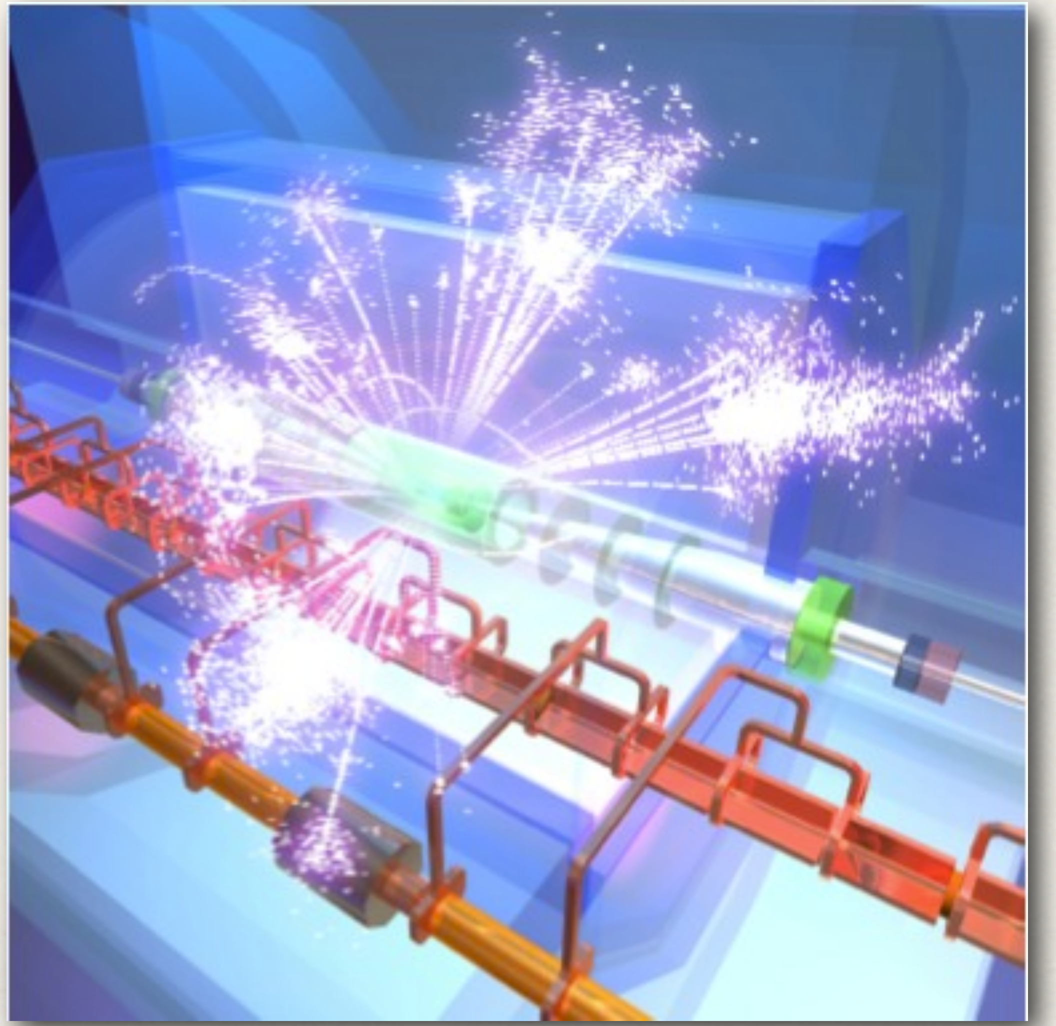


Future Activities

- New person fully dedicated to AIDA project will be hired (April-May 2013)
  - Fully involved in the tracking code developing for 2015 data taking and phase 1 and phase 2 preparation
  - Deliverable 2.6
- Identify components that need to be optimized and/or redesigned
- Start the iterative procedure for each component according to the priority list (from August 2013)
  - Integration into the CMSSW
  - Validation of the Physics performance

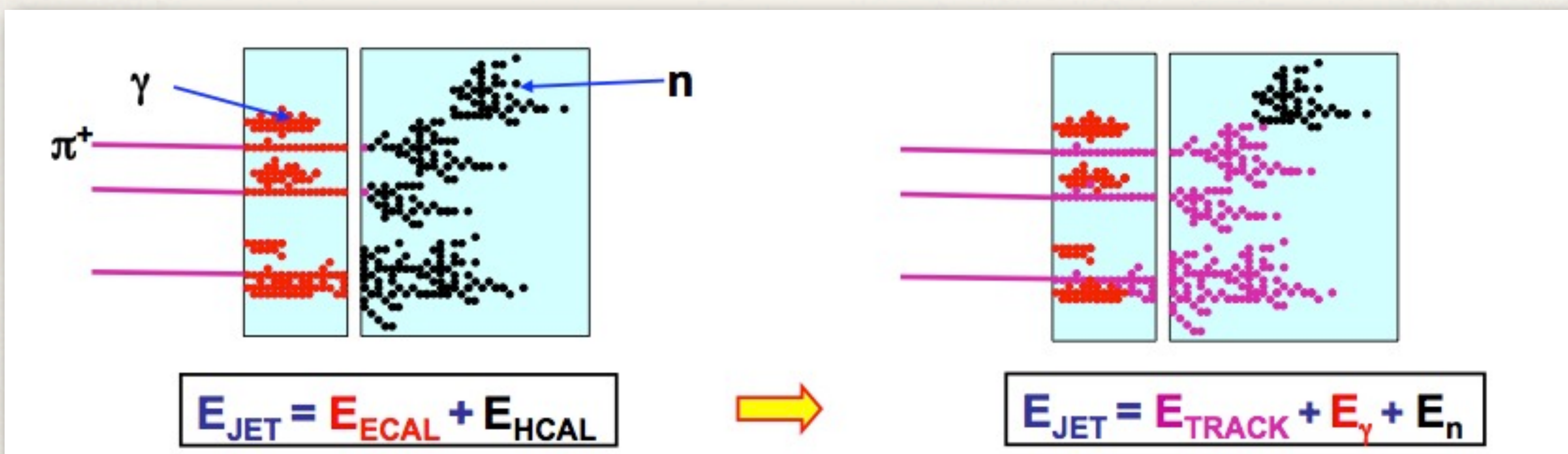
Particle Flow Calorimeter

- Cambridge
  - Framework
  - Algorithm development/optimisation
- CERN
  - extensive validation
  - performance benchmarking
- LLR
  - algorithm development



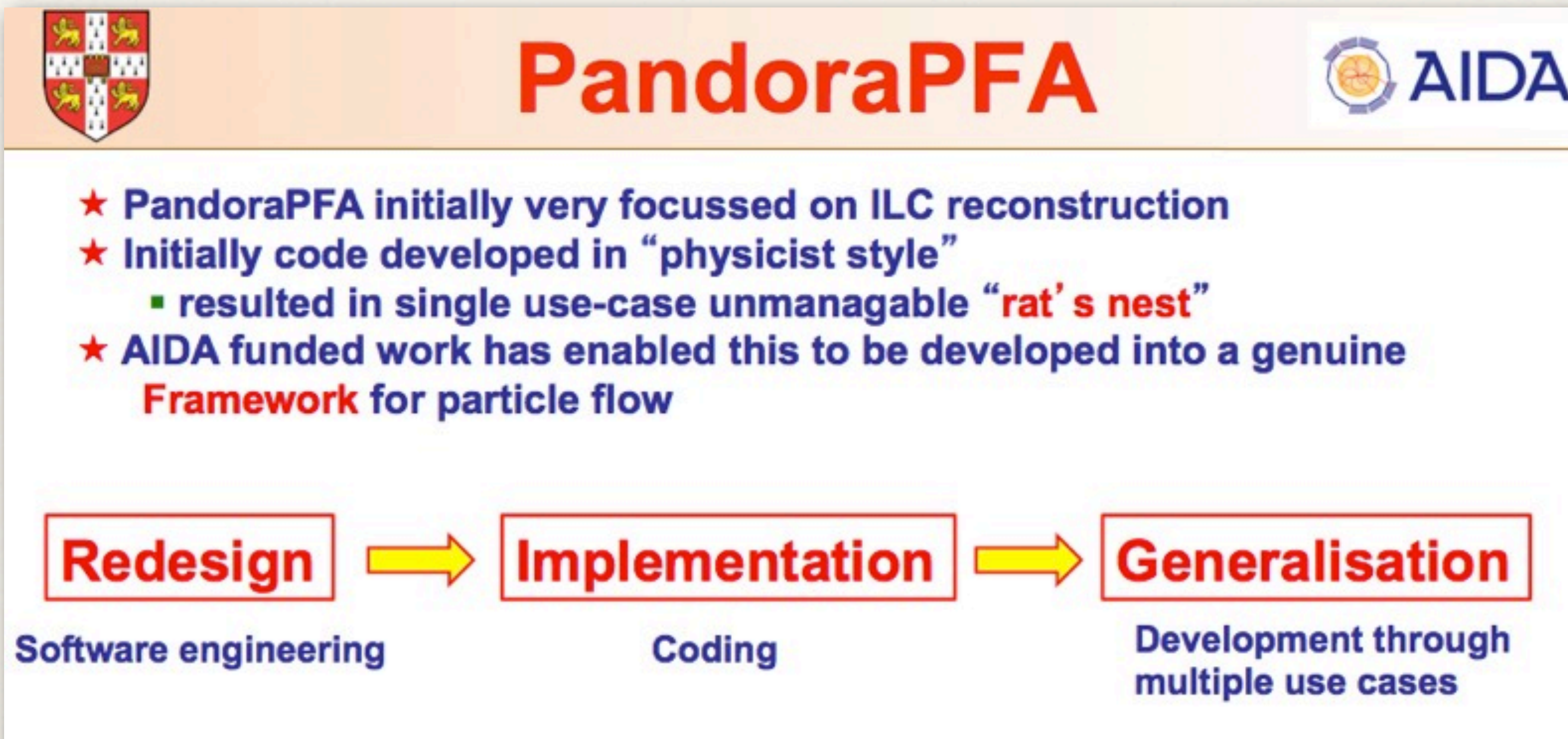
Particle Flow Ingredients

- Goal
  - Improved energy resolution
- Requires
  - Highly granular detector
  - Sophisticated reconstruction software



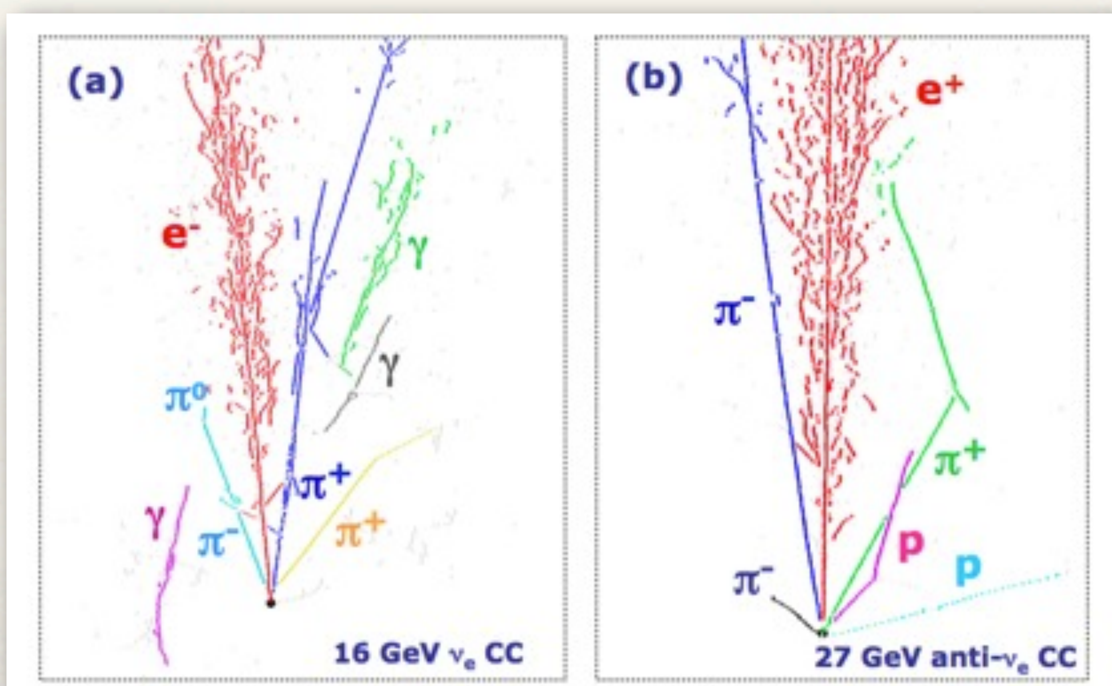
AIDA work

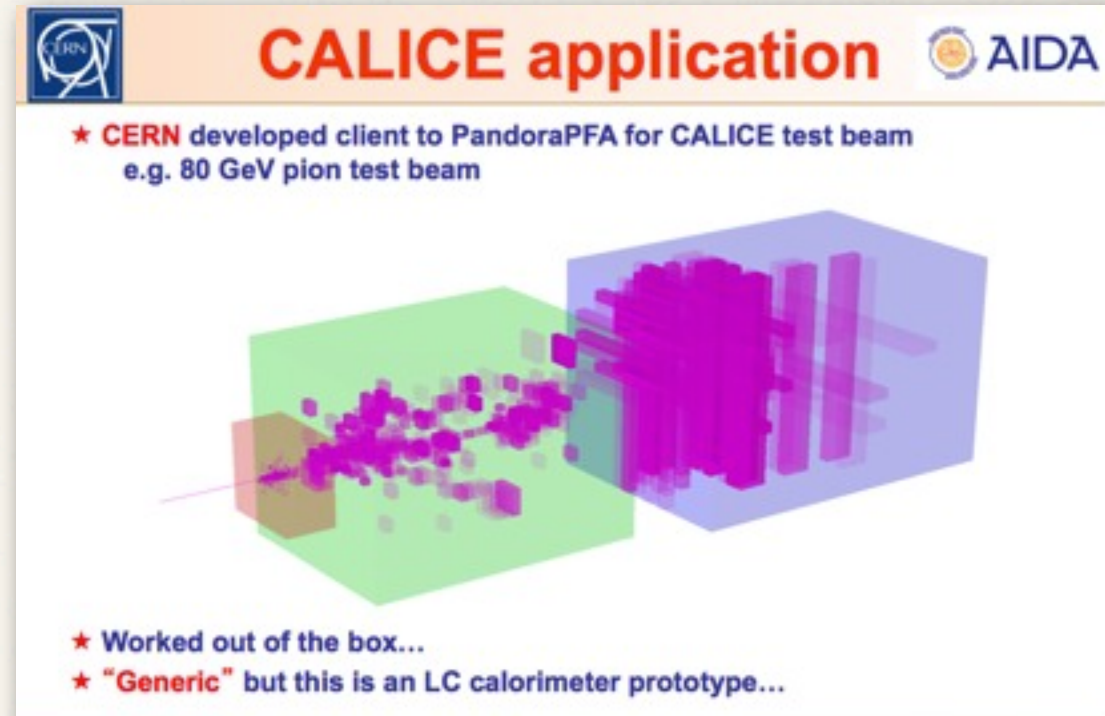
- PandoraPFA initially very focussed on ILC reconstruction
- Evolved to generic, fast implementations, easy adaptable (plugins), decoupled from geometry toolkit

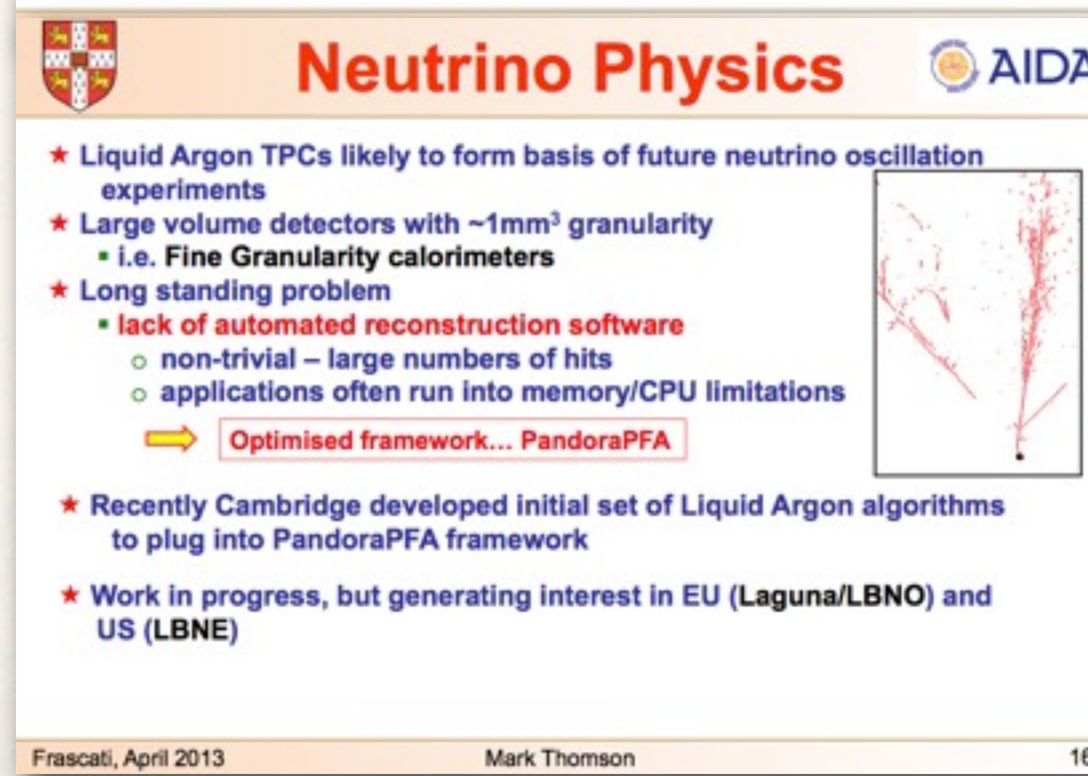


PandoraPFA Applications

- Generalization
  - Fine Granularity calorimetry – e.g. LC detectors
  - Coarse Granularity calorimetry – e.g. LHC detectors
  - Liquid Argon reconstruction – Neutrino physics

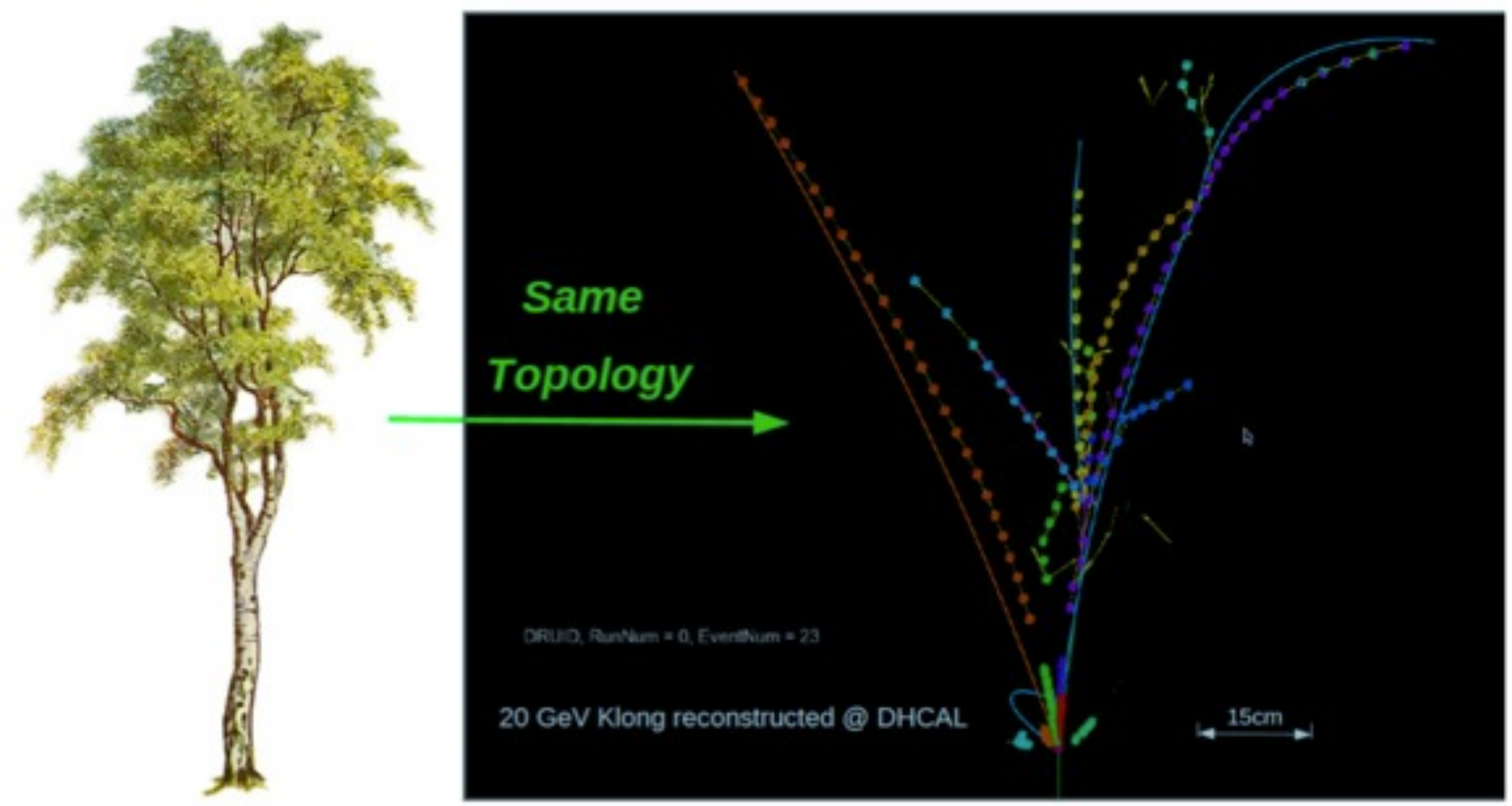






Alternative clustering algorithms

- Arbor: Shower/Jet reconstruction
  - LLR Work
  - Idea: Create local connectors, iterate until the final configuration satisfy a tree structure
  - Possible integration into PandoraPFA
  - Application to LHC

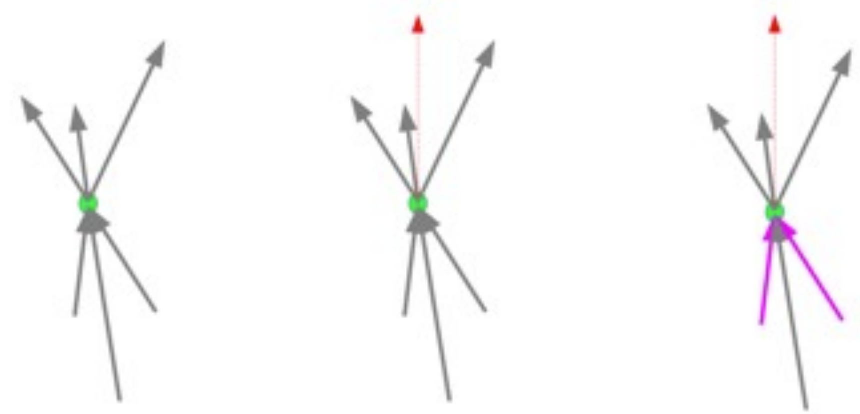


- Original idea from Henri Videau: Create local connectors, iterate until the final configuration satisfy a tree structure

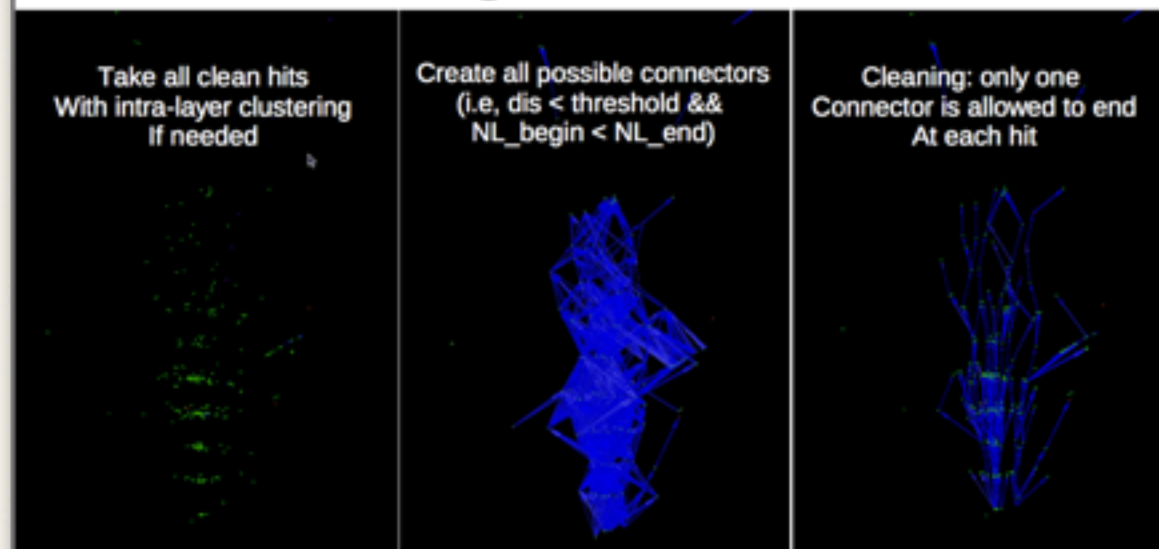
Four Algorithms

AIDA Algorithm - III

- Final Connector Set (as LCRelation)
  - Clean: keep at most one connect (with possible cut) end at given hit
    - Current order parameter : (Angle(link, ref_dir) + 0.02) * (link_length)
    - Order parameter can be defined according to a semi-local fit



AIDA Algorithm - 1



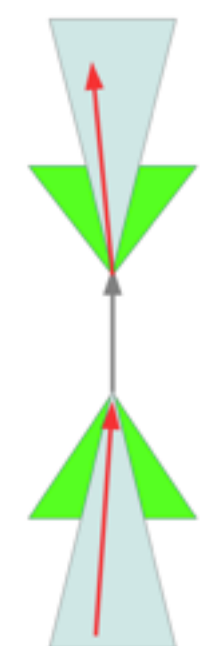
- Clean
  - If a hit is the end of many connectors, keep the one with extreme order
  - Current Order parameter: Ang(P1, P2)*sqrt(Dis(P1, P2))

AIDA Algorithm - IV

- Branch Set (As Cluster Collection)
  - Create all the possible branches
  - Loop the branches with length order, flag each hit (as belong to previous (longer) branches). End the branch at the flagged hits
  - An unique branch configuration from connections
  - Branch ~ daughter particle trajectory
- Link Branches to Bush
- Operational: Arbor operated independently for ECAL & HCAL.

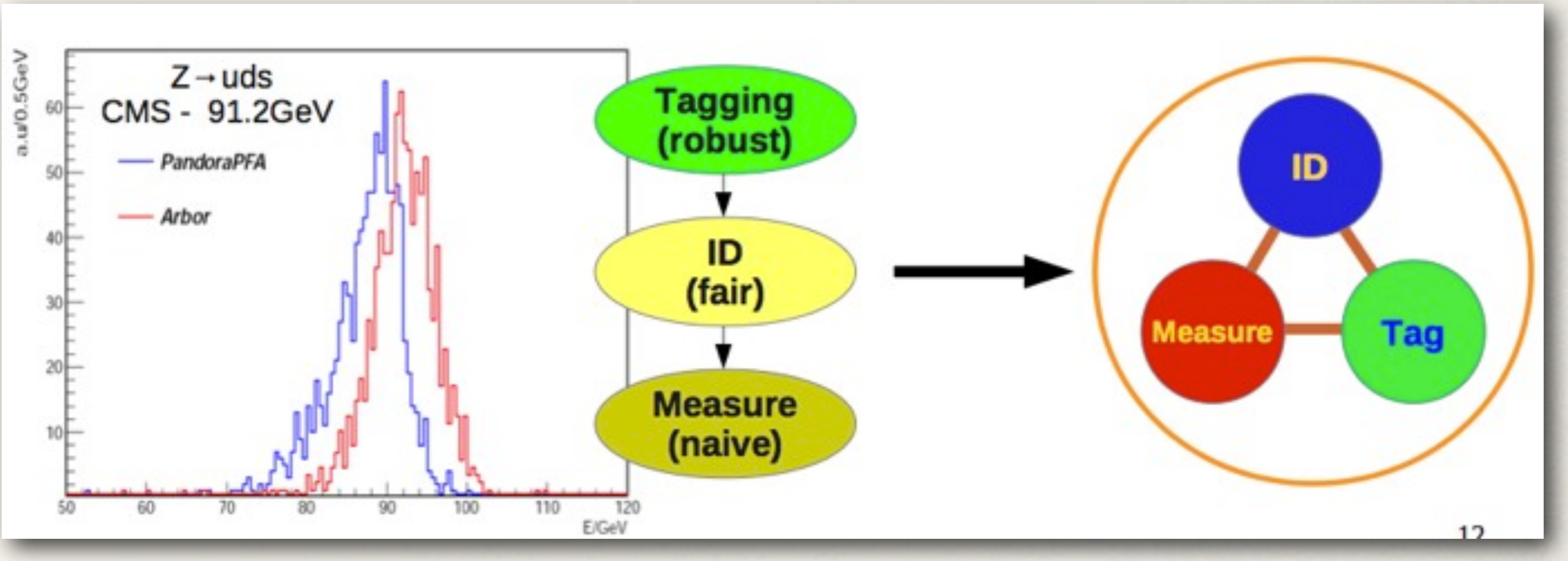
AIDA Algorithm - II

- Iteration the Connector Set (3 times)
  - Create new possible connector
    - If hit found in the explored region, add new connector
    - Definition of the region (function to be optimized)
  - For each hit, get reference direction vector (function to be optimized)
    - Current function: mean of outgoing and hit position (Normalized vectors)
    - If only one outgoing link: connector direction



Arbor Status

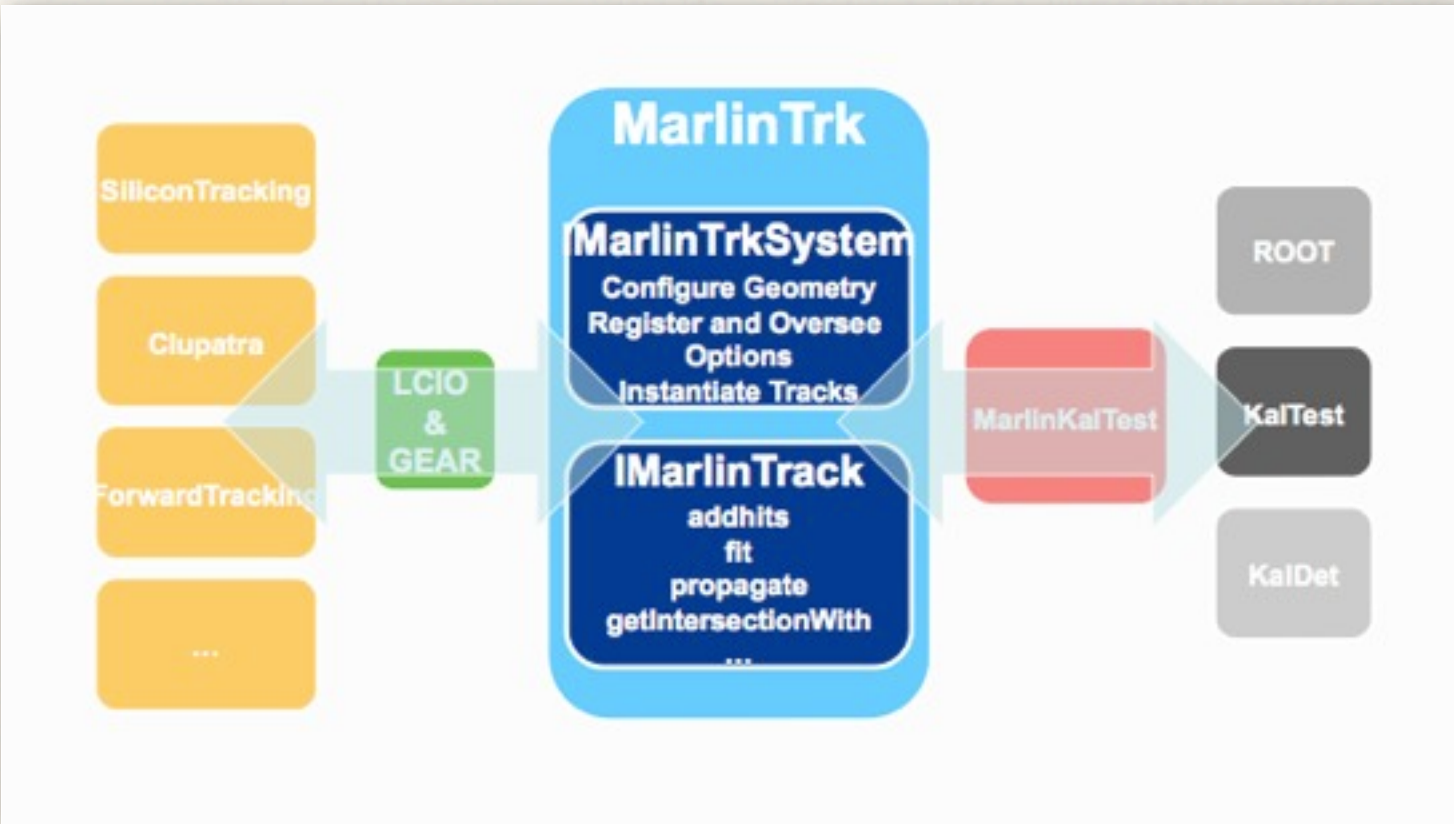
- Promising separation between nearby shower Jet E: ~ PandoraPFA
- Goal: precisely identify and measure every particle reaching calorimeter



- Stabilized core algorithm
- Successful tagging of sub shower structure
- Good separation between nearby shower
- Reasonable jet energy performance lots of optimization remains

Tracking Task

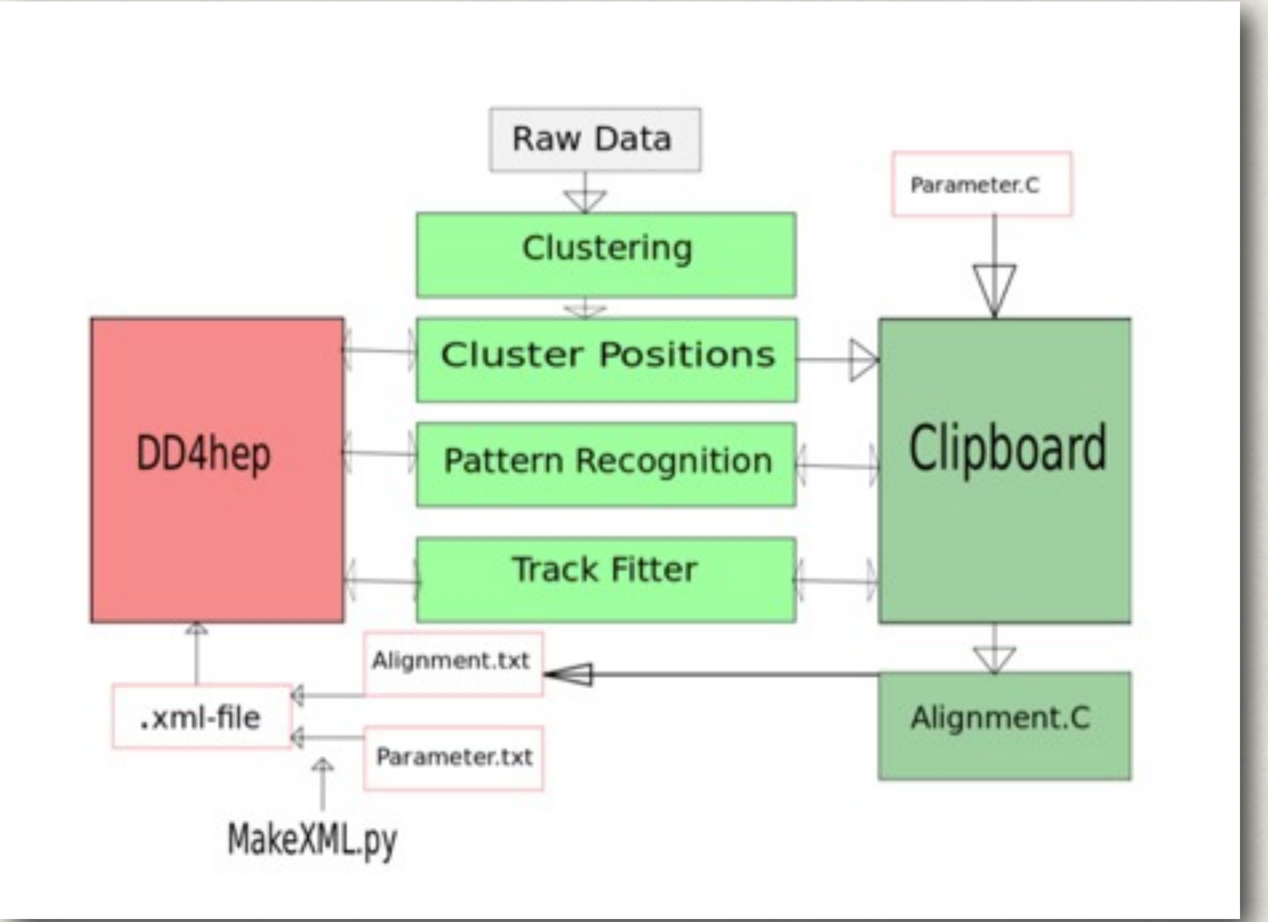
- Prototype in the context of ILD
- Current work related to geometry
  - Challenges:
    - Solve the problem(s): design interface and algorithms
    - Efficiency: accuracy vs. time consumption

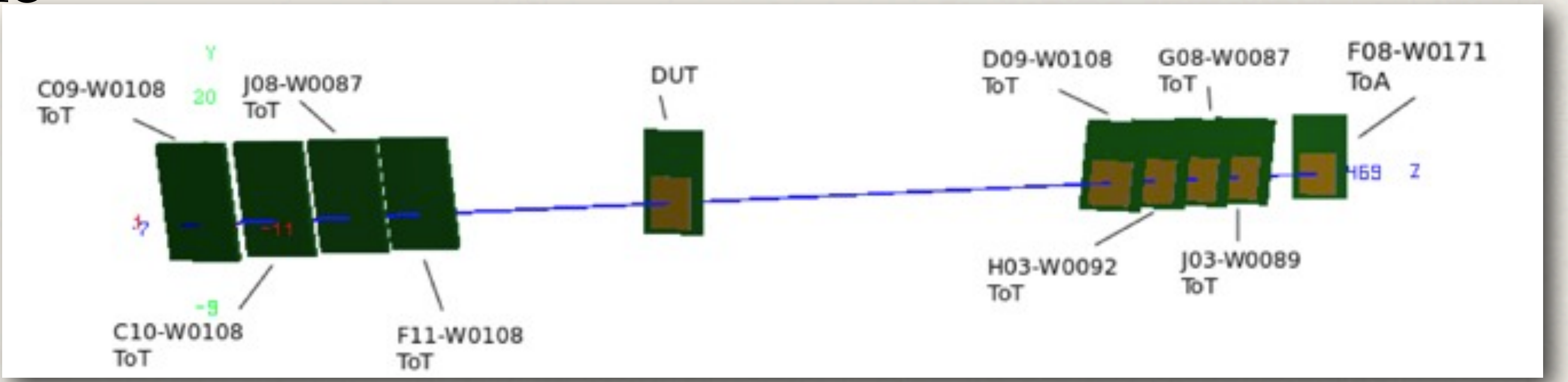


- In progress:
  - Exchange the algorithmic side for easier connection with geometry
  - Started working with GENFIT: generic tracking tool, used for Belle-2
  - Very large overlap with goals of tracking WP
  - Geometry is based/built with TGeo
  - Clear connection to DD4Hep

Alignment: Test-Beam Telescope

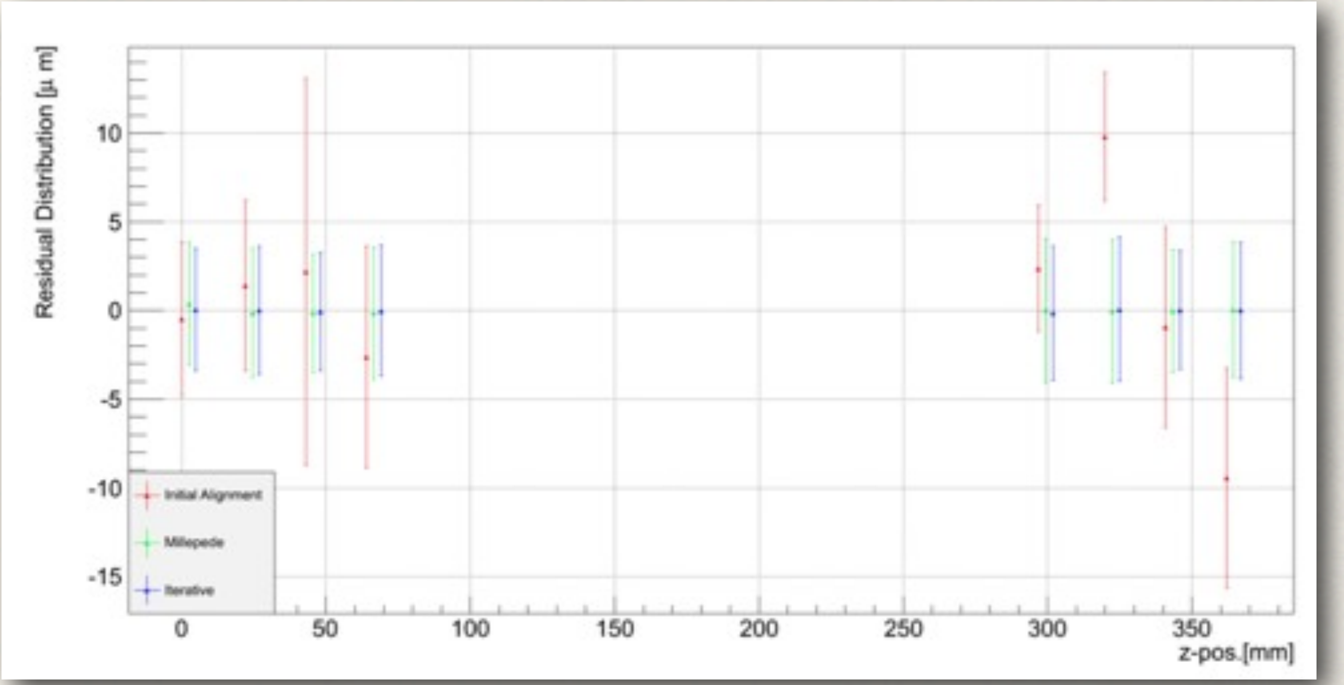
- Software components:
  - Raw data processing
    - Specific to telescope
  - DD4hep Geometry Package
    - **First real usage of it**
  - Millepede for Alignment
    - Minimization of χ2 via single matrix inversion





Test-Beam Telescope Results

- Alignment by combination of Millepede and iterative method
- Several angle-scans were performed for various devices with different thicknesses/ technologies at different bias voltages.



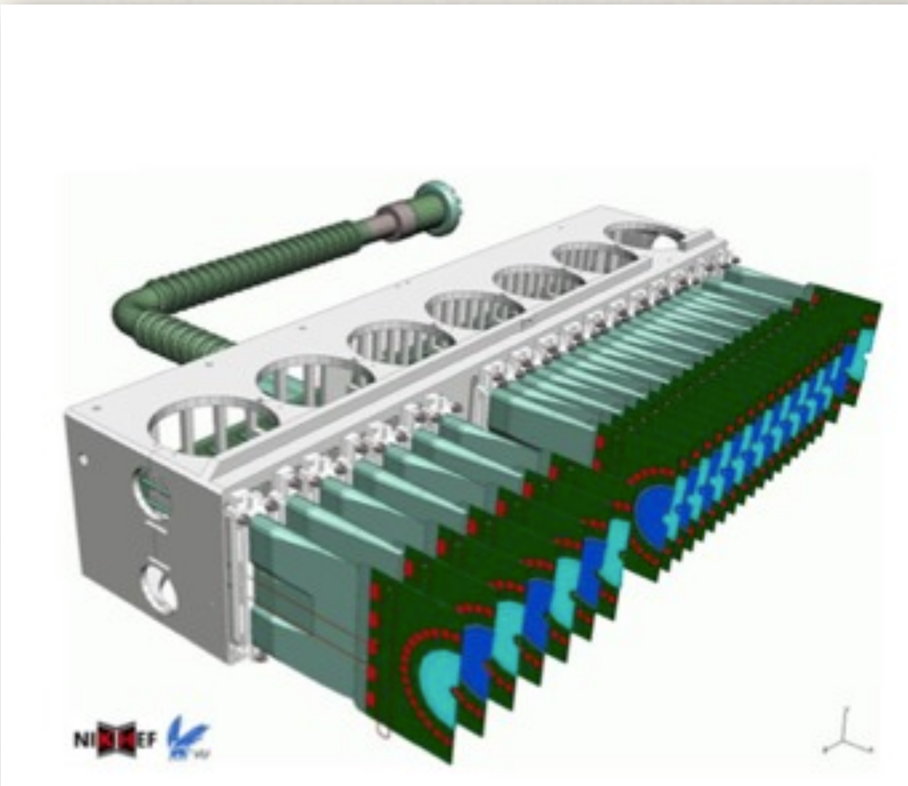
- Paper in preparation, including use of AIDA-work.

Comparison Millepede vs. Iterative

	Millepede	Iterative
Time for 1 iteration with $\approx$3500 tracks(on 1.6 GHz CPU)	3-4min	4-5min
Iterations needed for alignment	1	3-4
Pros	Fast and stable	Sensitive to every dof
Cons	No sensitivity to Δ_α, Δ_β	Time consuming carefull monitoring needed

LHCb Velo

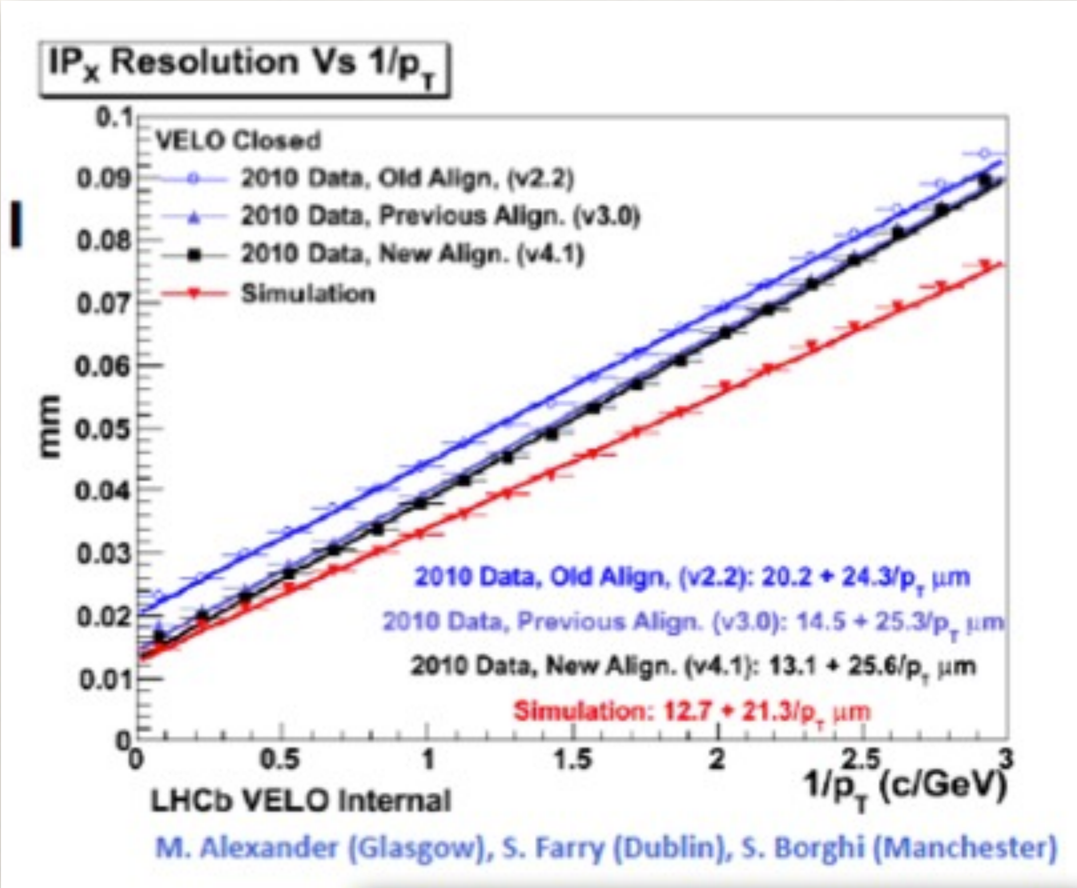
- Status of alignment
- Two alignment methods used:
  - Based on Millepede
  - Iterative procedure using Kalman track fits.
  - Sensor misalignments are known to be better than 4 μm.

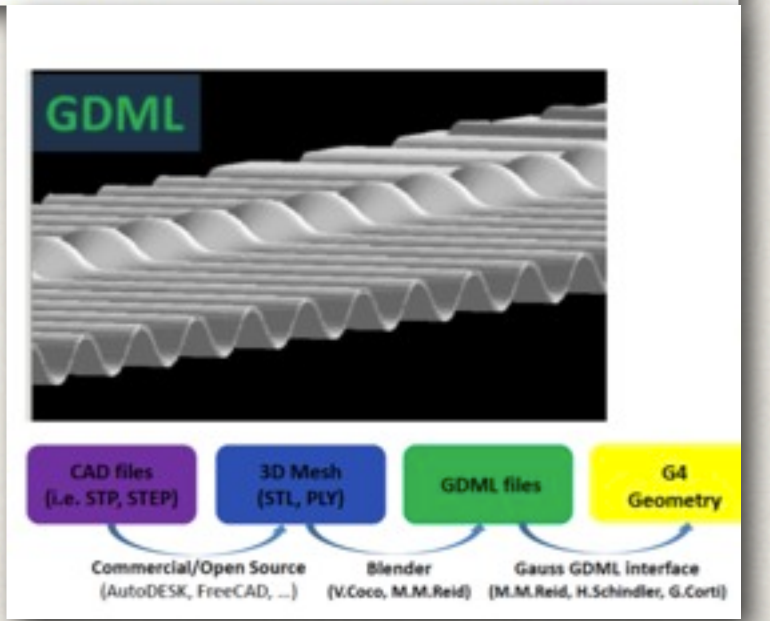


- Alignment of two detector-halves is carefully monitored after every run by comparing reconstructed primary vertex or track-residuals in overlap-region.
- In 2012: Misalignment in X less than 5μm. Difference between 2012 and 2013: ≈ 2μm.

LHCb Velo Material Description

- Discrepancy in IP-resolution between simulation and data.
- Alignment issues can be ruled out as data and MC agree at high momenta
- Reasons could be mass/material distributions or Geant4 multiple scattering
- Better description of the RF-Foil surrounds VELO sensors to separate them from primary vacuum
  - Using the new tessellated from USolids

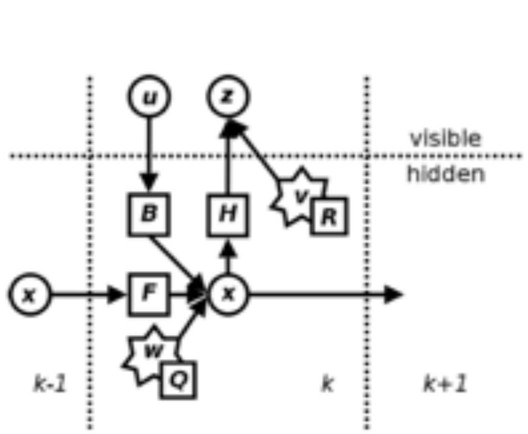


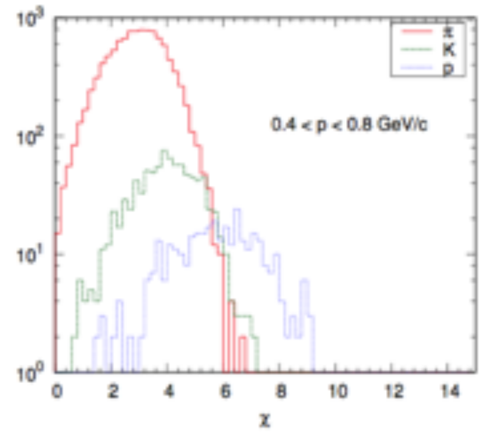


Clustering, Energy loss, and Vertexing Algorithms for Si detectors (Wigner RCP)

- Set of new algorithms for Si detectors
  - Kalman tracking (possibility to use track χ^2/ndf for PID)
  - Re-estimation of cluster position
  - Estimation of cluster deposit (fully corrected)
  - Estimation of most probable energy loss for tracks ("dE/dx")
  - Gain calibration of detector elements
  - Light event viewer in web browser (via LiveGraphics3D)
  - Configurable packages (eventSimulator, siEnergyLoss)
  - Control plots

Kalman tracking

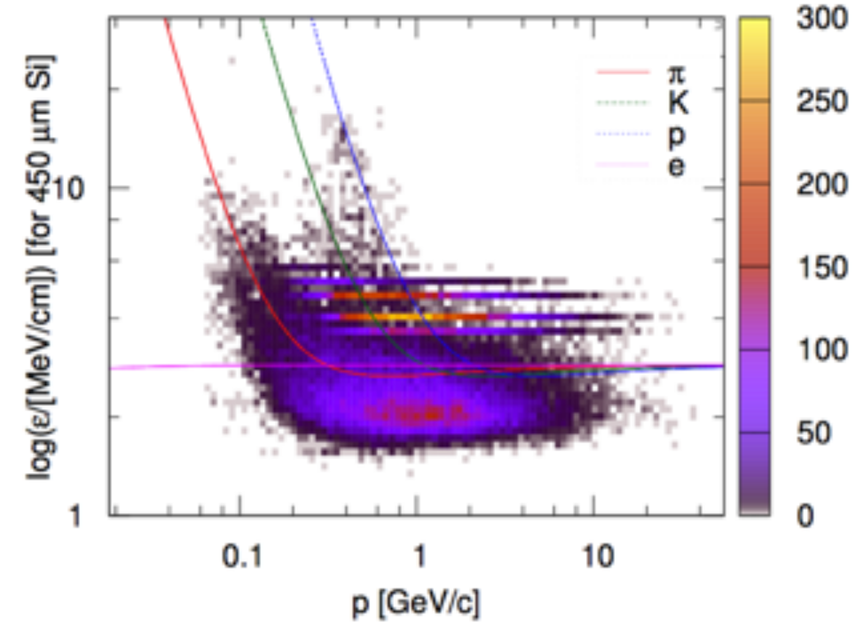




- Kalman filter
  - Known physics effects (F, Q), measurement uncertainties (V, R)
  - Prediction + filtering + smoothing
  - State vector: $x = (\kappa, \theta, \psi, r\phi, z)$, 5 dimensions
  - Measurement operator, covariance of noise

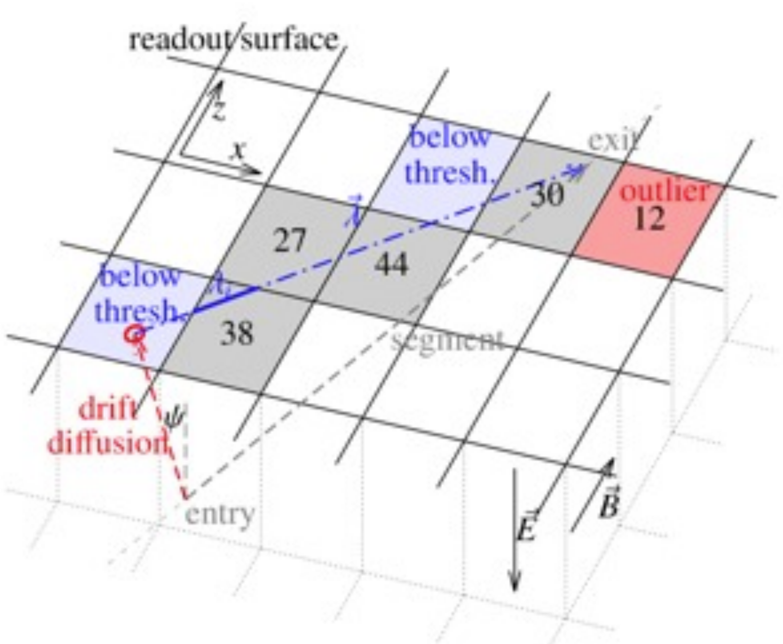
Possibility to use track-fit $\sqrt{\chi^2/\mathrm{ndf}}$ for enhancement

Estimation of most probable energy loss for tracks

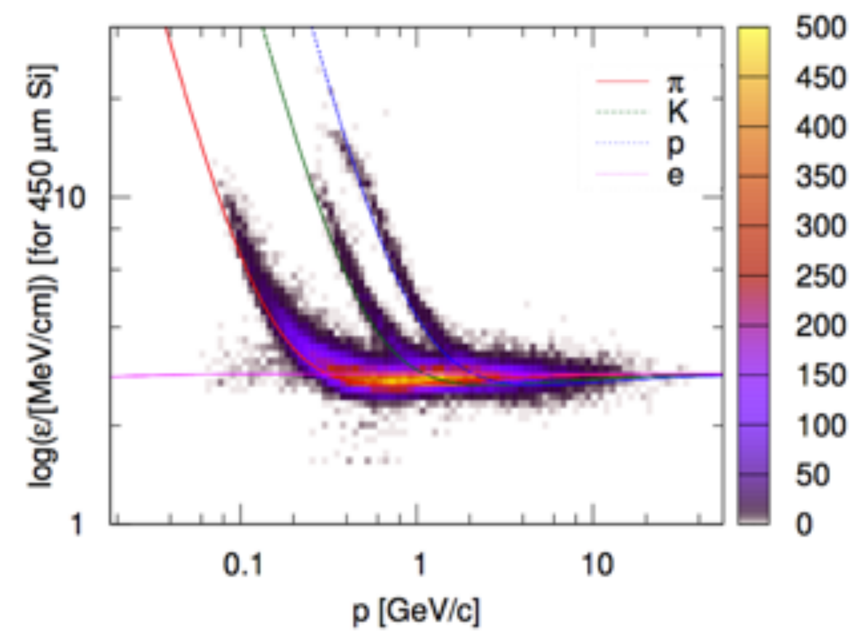


Most probable energy loss rate ε
over a reference path length of $l_0 = 450\ \mu\mathrm{m}$
Without calibration!

Clustering (pixels and strips)



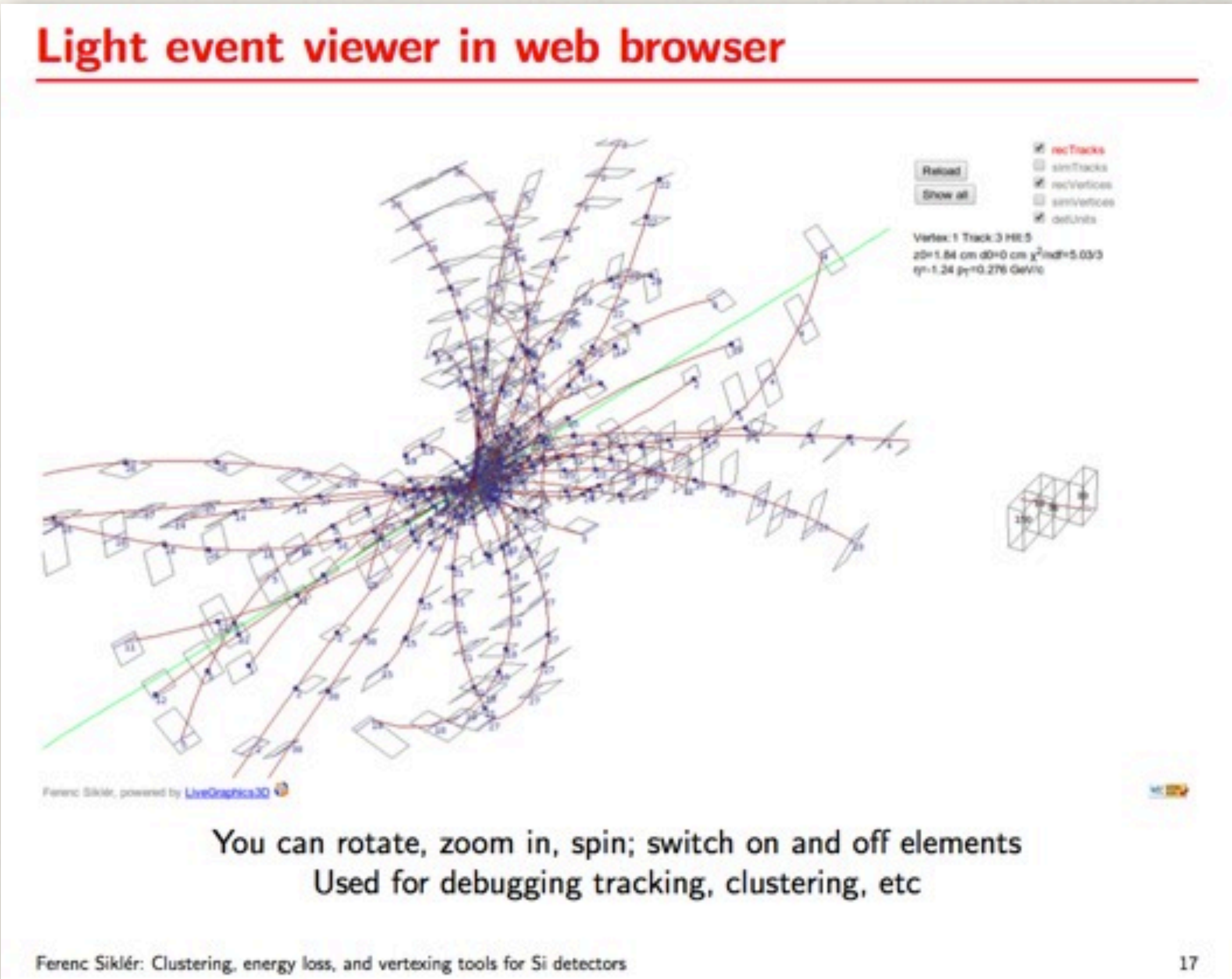
Estimation of most probable energy loss for tracks



Most probable energy loss rate ε
over a reference path length of $l_0 = 450\ \mu\mathrm{m}$
Well done

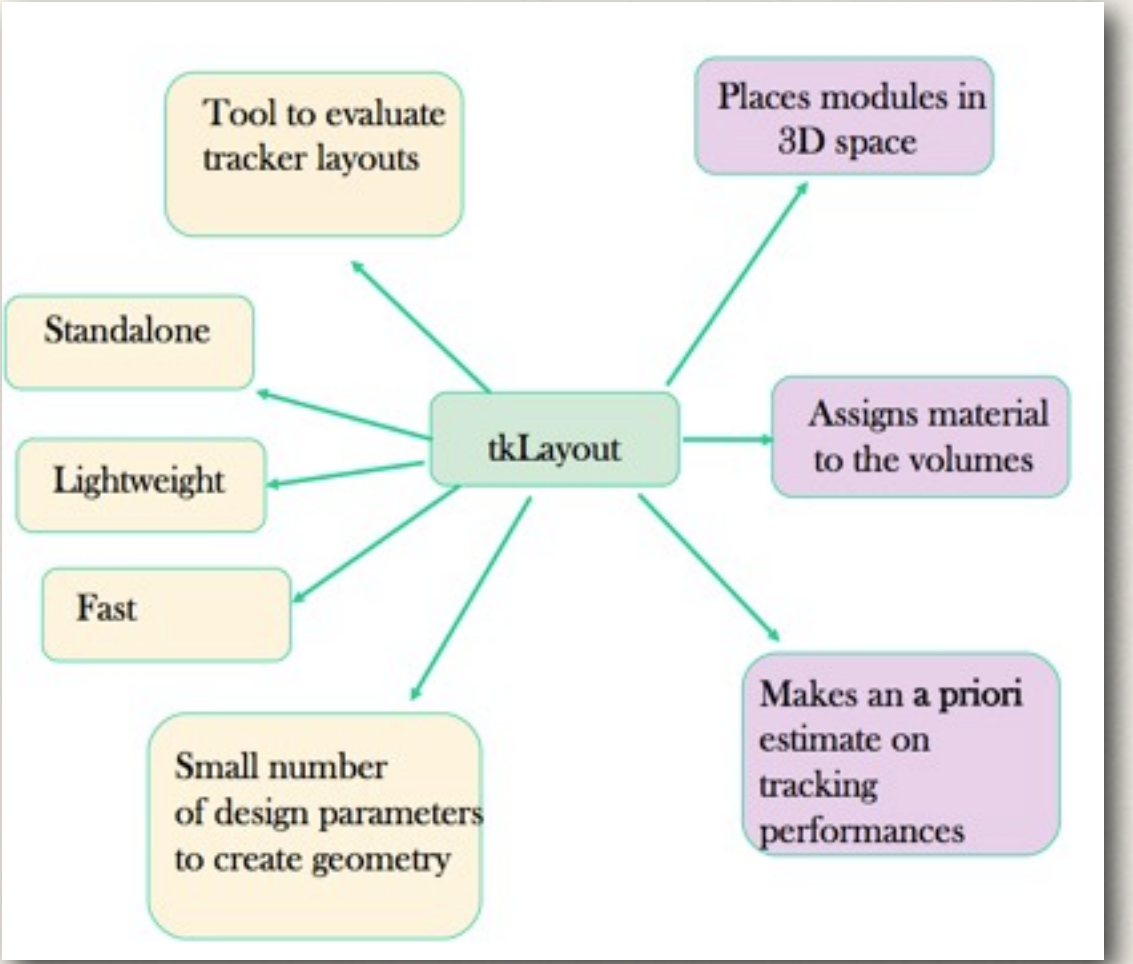
Status and Outlook

- Most of the packages are ready for use
- Methods used in CMS papers on particle spectra
- Next:
  - pattern recognition for tracking
  - use of cluster shape information for low fake rate pattern recognition
  - use refined cluster resolution and residuals for re-tracking
  - finish documentation

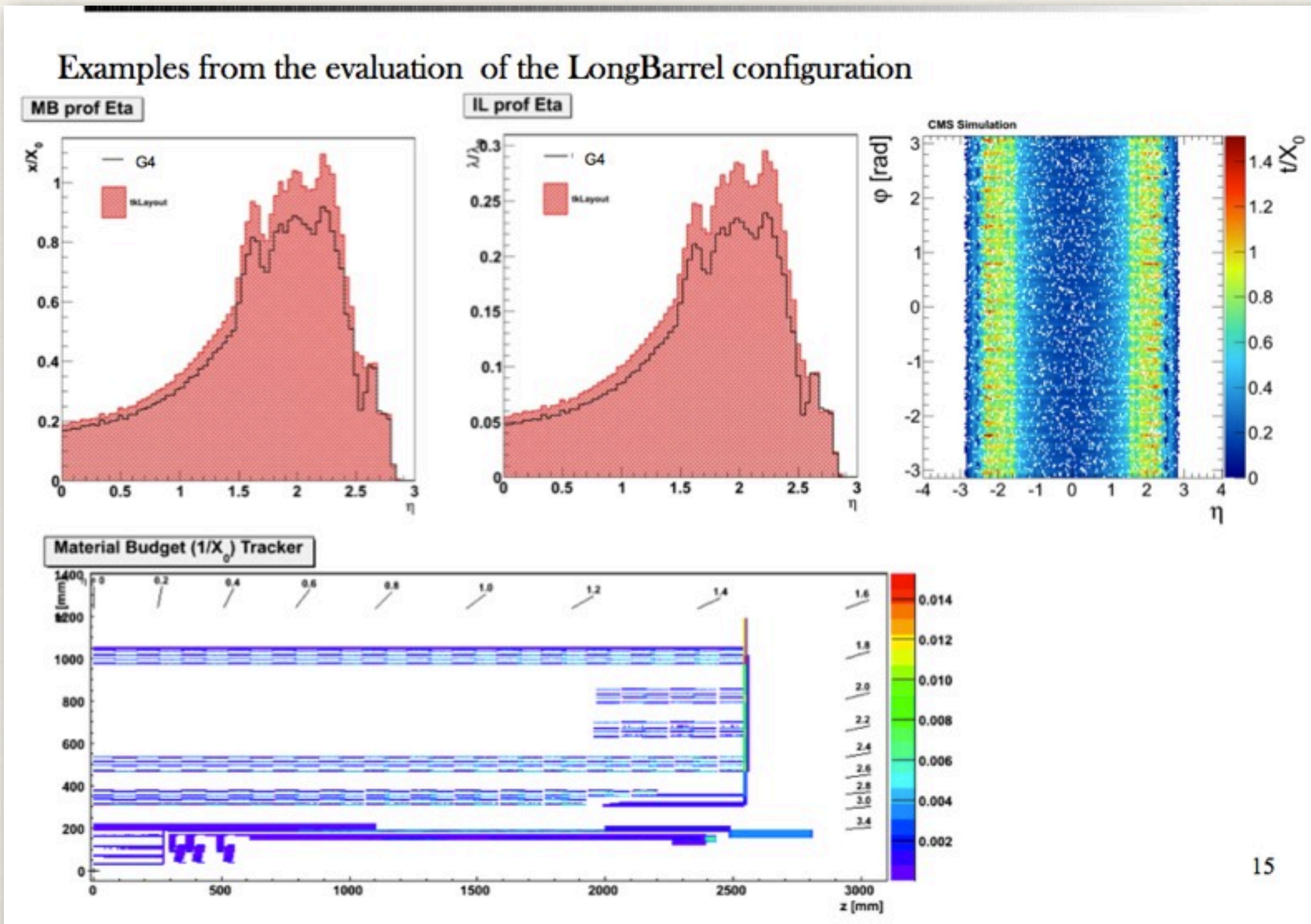


A Tracker Layout Modeling Tool

- How to evaluate performances of a (tracker) detector geometry without having to full simulate millions of events
- The **tkLayout** package
  - Compare different detector layouts
  - No Monte Carlo
  - Ingredients:
    - Error propagation
    - Sensor resolution (measurement error)
    - Multiple scattering (treated as a correlated a measurement error)



Comparing with G4 simulations



Conclusions

- Progress has been reported in all sub-tasks
  - Deliverables and Milestones achieved (with one exception)
- Started to see the use of geometry toolkit (DD4Hep) in other Sub-tasks
- Most of the work done is the context of a given experiment (LC, LHC)
  - This is fine but we need to make an effort to generalize and develop experiment independent software (PandoraPDF is an excellent example)
- Final goal is to deliver a set of generic and reusable libraries that can be used by the HEP community
  - Available in an open code repository
  - Proper documentation