# AIDA Alignment Package

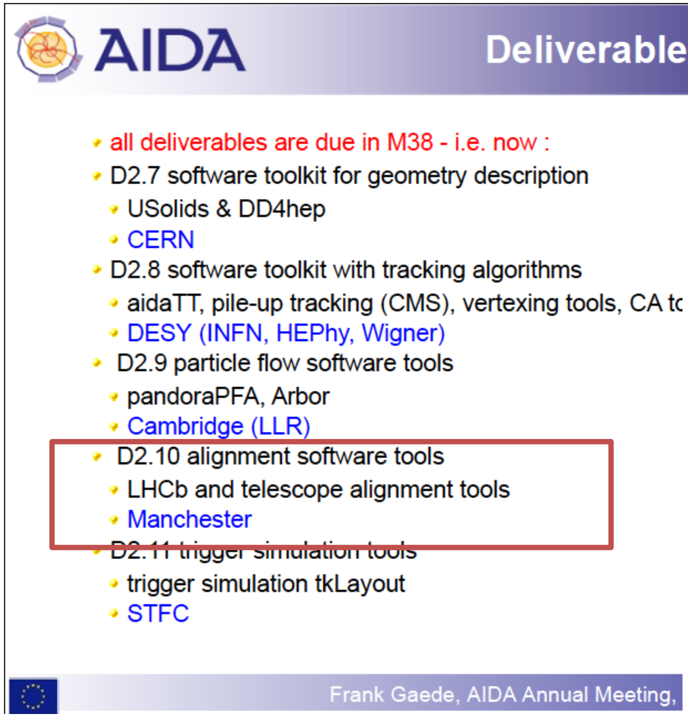
*AIDA Final meeting, CERN*
*10/12/2014*

Silvia Borghi, Christoph Hombach, Chris Parkes

**Outline:**

- Intro: Misalignment problem and strategies
- Testbeam: AIDA TimePix3 telescope alignment
- VELO & VELO upgrade: recent alignment & future plans

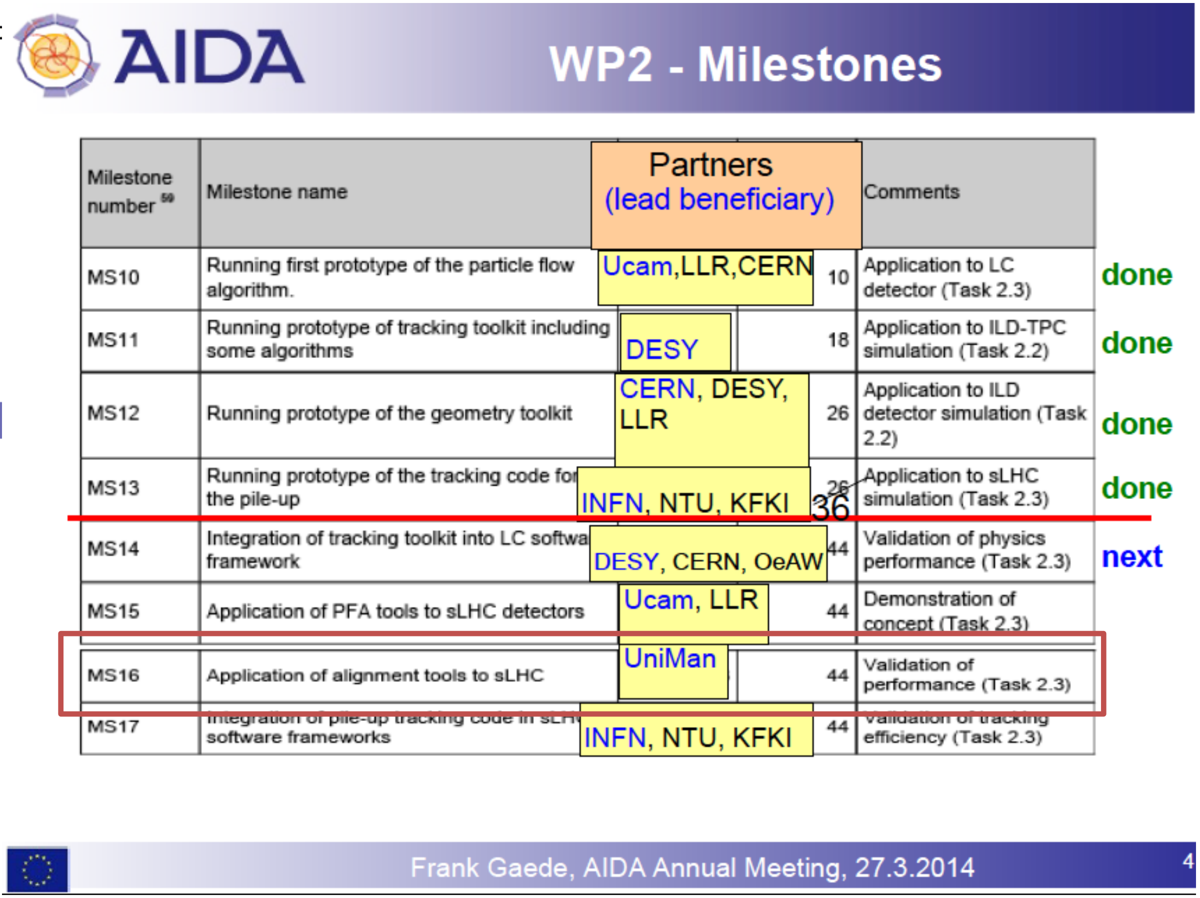
# Milestones and Deliverables:



## Deliverables

- all deliverables are due in M38 - i.e. now :
- D2.7 software toolkit for geometry description
  - USolids & DD4hep
  - CERN
- D2.8 software toolkit with tracking algorithms
  - aidaTT, pile-up tracking (CMS), vertexing tools, CA tc
  - DESY (INFN, HEPhy, Wigner)
- D2.9 particle flow software tools
  - pandoraPFA, Arbor
  - Cambridge (LLR)
- D2.10 alignment software tools
  - LHCb and telescope alignment tools
  - Manchester
- D2.11 trigger simulation tools
  - trigger simulation tkLayout
  - STFC

Frank Gaede, AIDA Annual Meeting,



## WP2 - Milestones

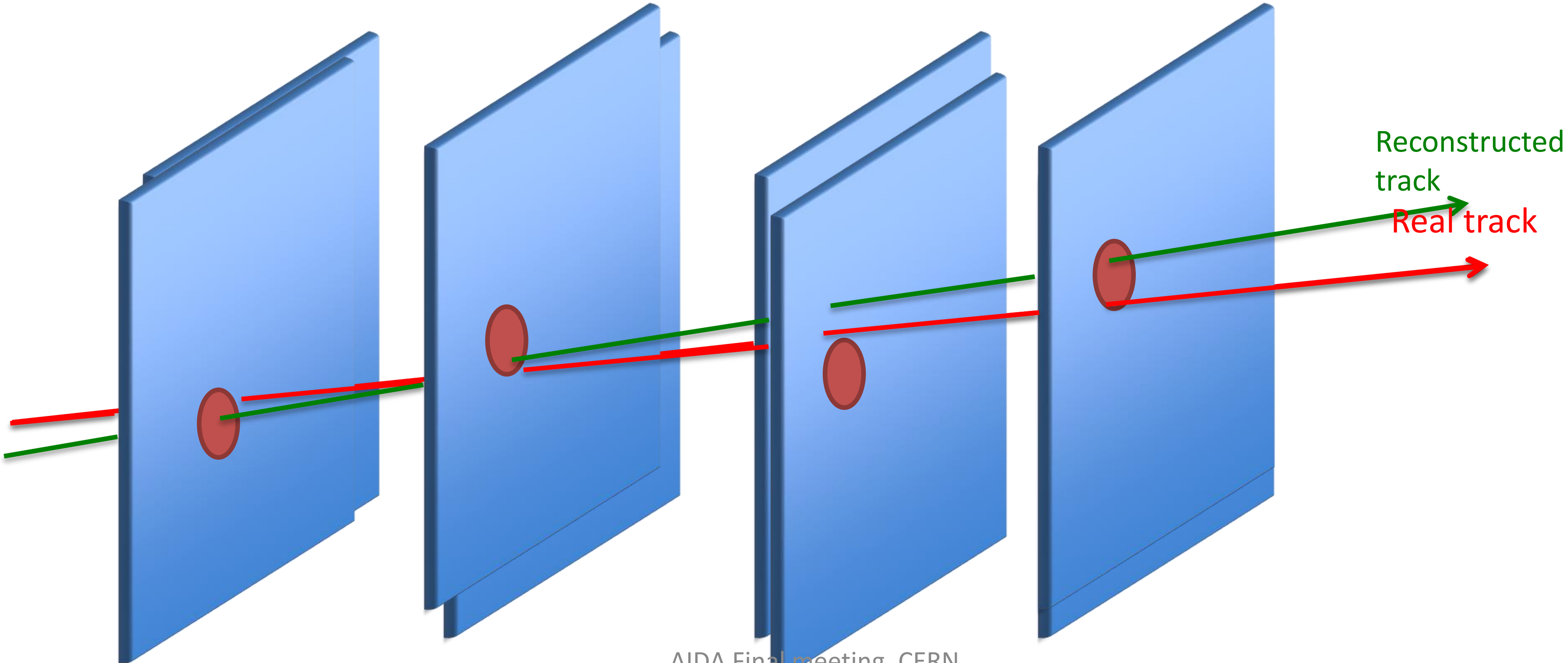
| Milestone number[59] | Milestone name | Partners (lead beneficiary) | | Comments | |
|---|---|---|---|---|---|
| MS10 | Running first prototype of the particle flow algorithm. | Ucam,LLR,CERN | 10 | Application to LC detector (Task 2.3) | done |
| MS11 | Running prototype of tracking toolkit including 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) | done |
| MS13 | Running prototype of the tracking code fo the pile-up | INFN, NTU, KFKI | ~~26~~ 36 | Application to sLHC simulation (Task 2.3) | done |
| MS14 | Integration of tracking toolkit into LC softwa framework | DESY, CERN, OeAW | 44 | Validation of physics performance (Task 2.3) | next |
| MS15 | Application of PFA tools to sLHC detectors | Ucam, LLR | 44 | Demonstration of concept (Task 2.3) | |
| MS16 | Application of alignment tools to sLHC | UniMan | 44 | Validation of performance (Task 2.3) | |
| MS17 | Integration of pile-up tracking code in sLH software frameworks | INFN, NTU, KFKI | 44 | Validation of tracking efficiency (Task 2.3) | |

Frank Gaede, AIDA Annual Meeting, 27.3.2014

- Introduction
  - Misalignment Problem & Strategies

# Alignment Introduction

- Track leaves hits on sensors
- **Misalignment problem:** Detector positions used in offline reconstruction do not correspond to the actual relative positions of the installed detector
    - Misplacements of det elements ->Hit positions are misplaced
- Reconstructed tracks are biased
    - Can lead to inefficient/wrong physical conclusions



# Solutions to the alignment problems

- **Assembly / survey measurements**
- Survey measurements of mounting positions
- Measurements during / after installation

- **Offline track based alignment algorithm**
  - Use track parameter to determine alignment parameter
  - Should be robust, stable and not too time consuming
  - Precision of alignment parameter should be known to an order of the detector resolution

- **Will focus on track based alignment**

# Alignment Overview

- **AIDA Alignment web page**

  http://aidasoft.web.cern.ch/node/31

  – Documents alignment papers /methods
    - Current/recent major particle physics experiments

AIDA Common Software Tools

Home » Documentation

Main menu

- Home
- Project
  - Deliverables and Milestones
  - Documents
  - Organization
- Tasks and Subtasks
  - Geometry
  - Tracking
  - Alignment
  - Particle Flow
  - Pile-up
- Meetings
- Packages

## Alignment

## Links

### Methods

Millepede webpage

Methods based on kalman: method1, method2

Experiments using:

- Methods based on Millepede: CDF, HERA-B, LHCb, CMS, ALICE, BELLE
- Residual minimisation: DELPHI, NOMAD, CMS
- Methods based on Kalman Filter: LHCb, CMS

# MILLEPEDE approach

Define:

$$\chi^2 = \sum_{k=0}^{n} \frac{\left(\overset{\text{Measured value}}{z_k} - \overset{\text{Reconstructed hit position}}{\mathbf{a}^T \mathbf{d}_k}\right)^2}{\sigma_k^2}$$

$\chi^2$ Is an explicit function of the **alignment parameters d.**

- At its minimum the corresponding alignment parameters represent the misalignment
- Minimisation can be written in matrix-form.

- MILLEPEDE (by V. Blobel) is a method, that can invert large matrices fast.
  - Alignment problem gets solved by **inverting a large matrix.**
  - Simultaneous fit of global and local parameters.

**A New Method for the High-Precision Alignment of Track Detectors,** Volker Blobel and Claus Kleinwort, Report DESY 02-077 (June 2002)

# BACH

- **BACH**: **B**asis of **A**lignment **CH**ain
- Standalone software, only depends on ROOT and BOOST

- Designed to test and verify alignment algorithms
  - Provides new users a development framework
  - Gives an example of a simple analysis chain and a full alignment algorithm
- Example based on telescope detector design
  - Simple geometry configurable

- Includes complete analysis chain:
- -> Simulation -> Clustering
  -> Pattern Recognition -> Track Fit -> **Alignment**
- Submitted to AIDA software package
- Documented in AIDA-NOTE-2014-001
- Source code available via svn:

  *svn co [https://svnsrv.desy.de/public/aidasoft/AIDAAlign/trunk/Tb](https://svnsrv.desy.de/public/aidasoft/AIDAAlign/trunk/Tb) Tb*

- Testbeam
  - AIDA TimePix3 Telescope

# TimePix3 Telescope

- LHCb testbeam programme motivated by the upgrade of the experiment
- First period in July/August 2014 at T9 beam line
- Second period in October/November 2014 at the SPS H8 beam line
- Telescope:
  - Two arms with four modules each
  - Each module is moveable
  - Moveable DUT centered between arms
  - Wide range of users
- TimePix3 ASIC key features:
  - high data rate (up to 10 million tracks/s)
  - availability of both deposited charge and timing information for each 55 x 55 µm pixel cell

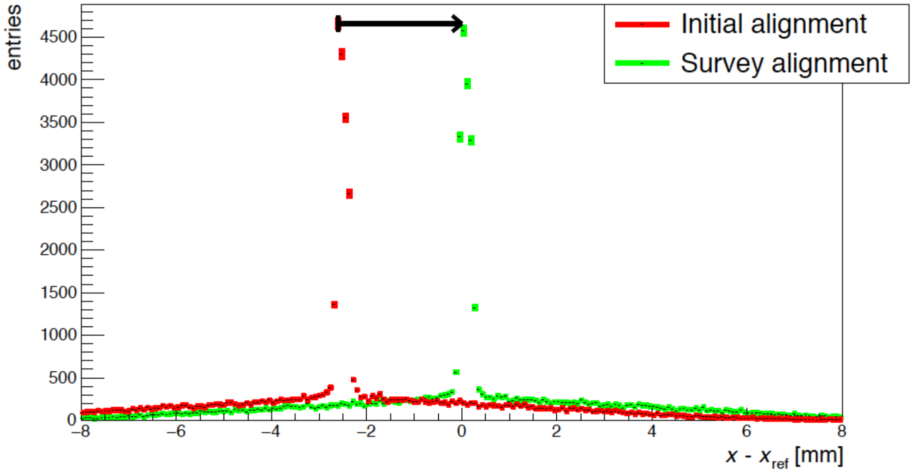
- Resulting in high spactial resolution and robust track reconstruction

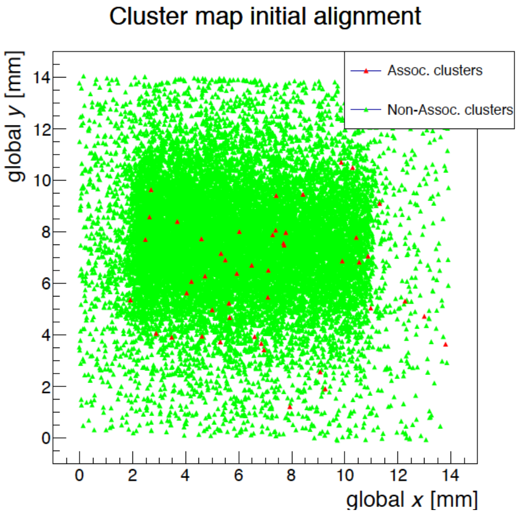
# Telescope software

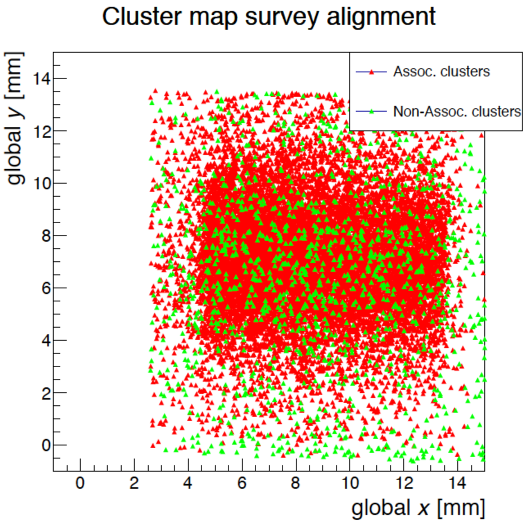
- Kepler: A new Gaudi-based software package to analyse testbeam results offline
- Part of LHCb-software
- Provides algorithms for
  - Event building
  - Clustering
  - Tracking
  - many user specific applications
- AIDA alignment package is fully integrated
- Alignment steps:
  - survey alignment
  - track-based alignment with Millepede
  - alignment of DUT
- Alignment works 'out of the box'
  - a simple alignment recipe allows the user to perform the alignment on a run-by-run basis
  - Takes O(~some minutes) to get the full alignment

# Rough alignment

- Usually the initial alignment is so poor , that it is impossible to reconstruct tracks
- A initial rough alignment can be done
- By looping over all clusters in a time-frame, the distribution of differences of x- and y- positions between a reference frame the module to align should peek at 0 (straight tracks)
- Rough alignment parameter are obtained at the maximum of the distribution
- After that the alignment should be good enough (<1mm) to reconstruct tracks.

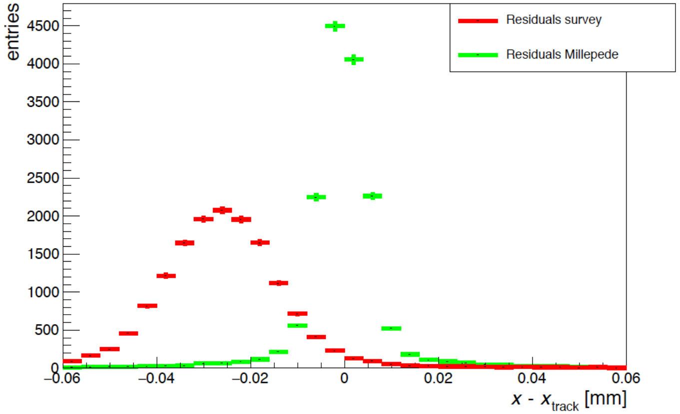


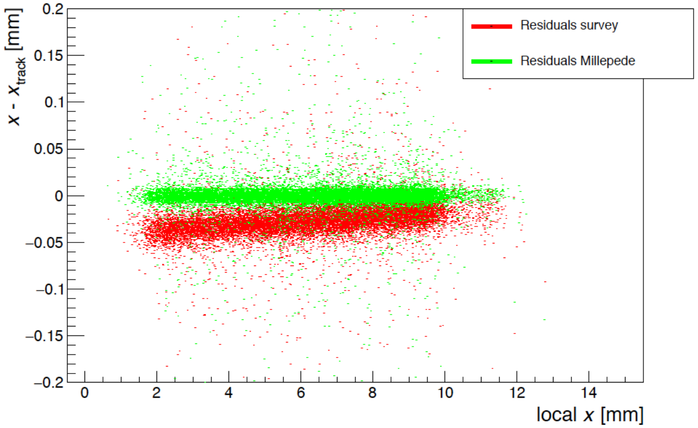


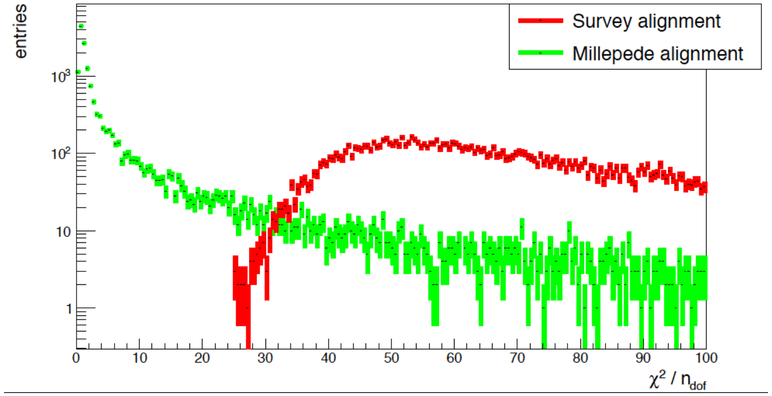


# Track-based alignment

- To reach a tracking precision of a few µm one has to align for the x- and y-translations and rotations around x, y and z.
- The MILLEPEDE approach minimises the track $\chi^2$ distribution
- A sensible set of constraints are applied:
  - One module is kept fixed
    - → defines global coordinate system and avoids global translations and rotations
  - The average translations and rotations are 0
    - → avoids global shearings or 'screwdriver'
- Telescope is perfectly aligned after this procedure!

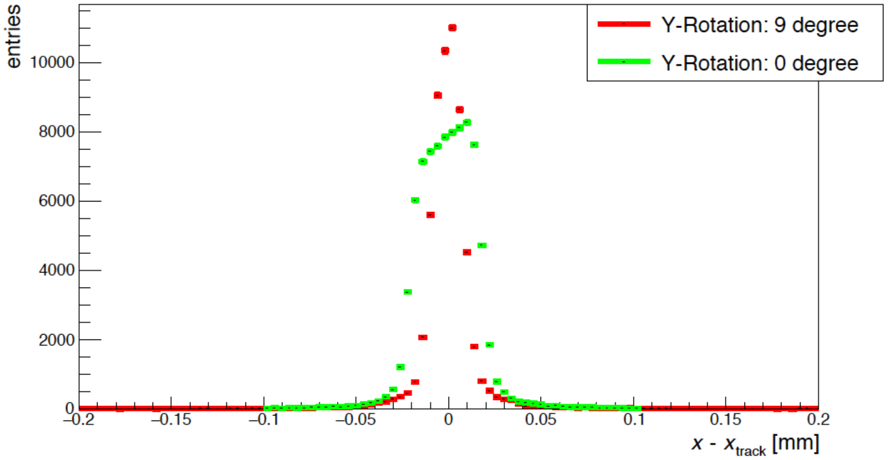


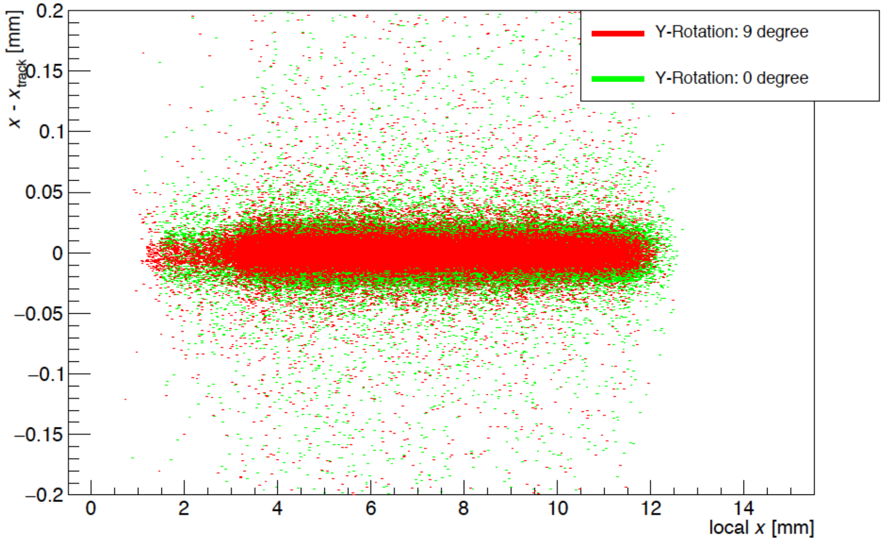




# DUT hit resolution

- The DUT is aligned separately from the telescope
- After the telescope is aligned, the DUT alignment is straightforward
- The DUT is aligned keeping the telescope-modules fixed
- Good telescope alignment guarantees
    - a good track-resolution
    - no further constraints need to be applied
- Resolution studies of the device can be e.g. for various angles wrt the beam



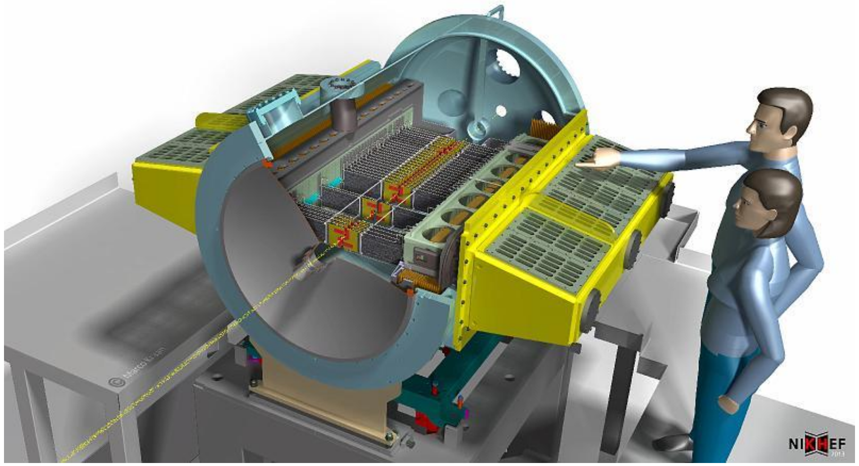


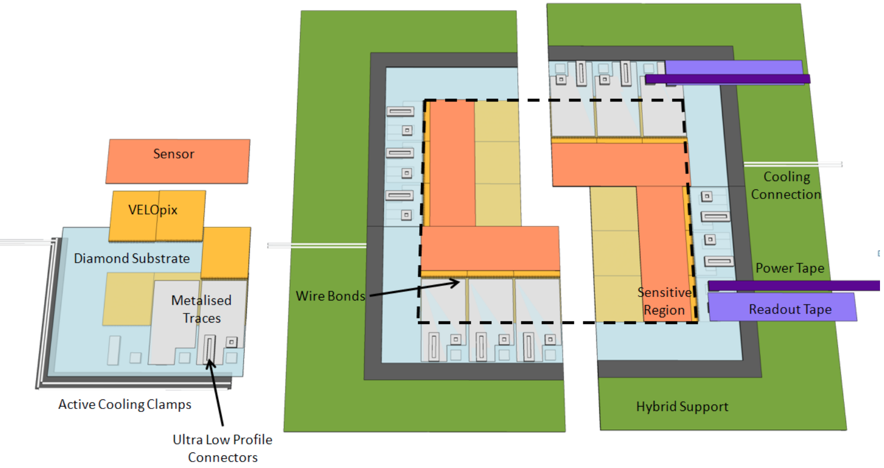
- LHCb VELO & VELO upgrade

  - Alignment

# VELO upgrade

- Designed to cope with LHC conditions after LS2 shutdown beginning in 2019
- Capable of 40MHz readout at a luminosity of $2 \text{ x } 10^{33}\text{cm}^{-2}\text{s}^{-1}$
- Consists of two moveable halves with 26 modules each
- Each module has an array of 4 sensors consisting of 3 TimePix chips
- Closest distance of approach to the LHC beams of just 5.1mm for the first sensitive pixel, 4 µm hit resolution and 30 fs time resolution

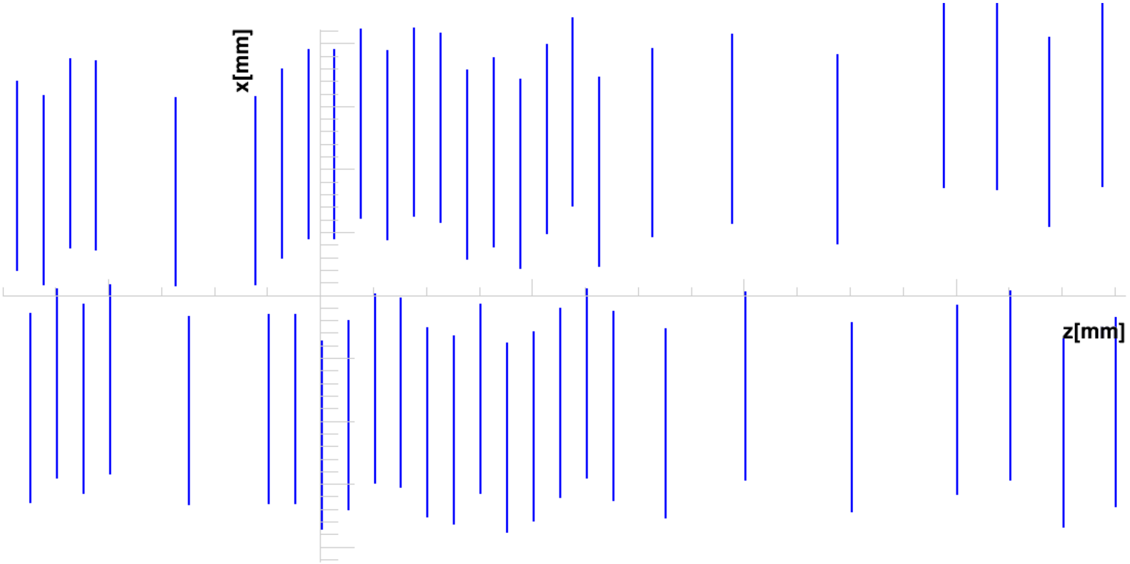




# VELO Upgrade alignment

- Similar alignment parameters as in VELO:
  - detector in global frame
  - detector halves
  - modules

- Alignment parameters are included in VELO upgrade software, some tuning is ongoing to perform the full alignment
- Should be ready after Christmas



# TED runs

- During LHC injection the beam is dumped on an absorber (TED) at the end of injection line
- The TED for beam 2 is located as ~340m before LHCb
- LHCb can reconstruct particles induced by the collision of beam 2 against the TED

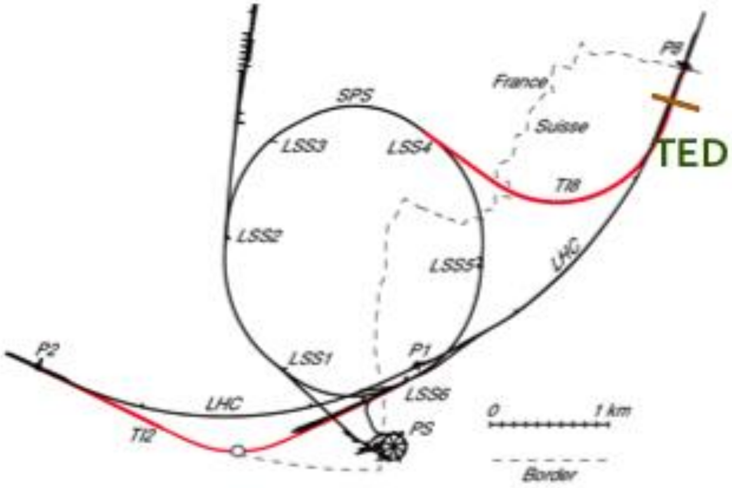
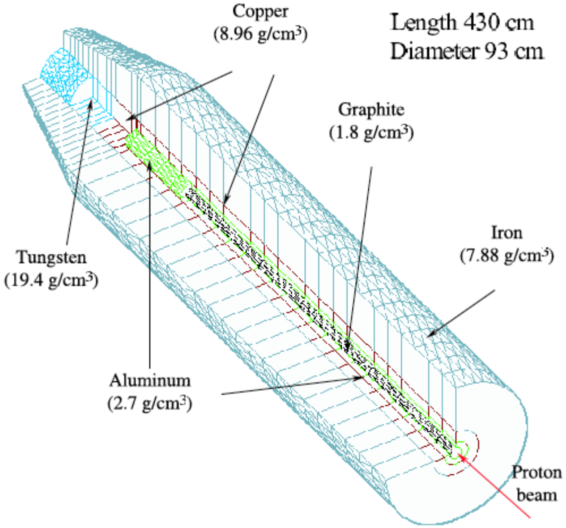
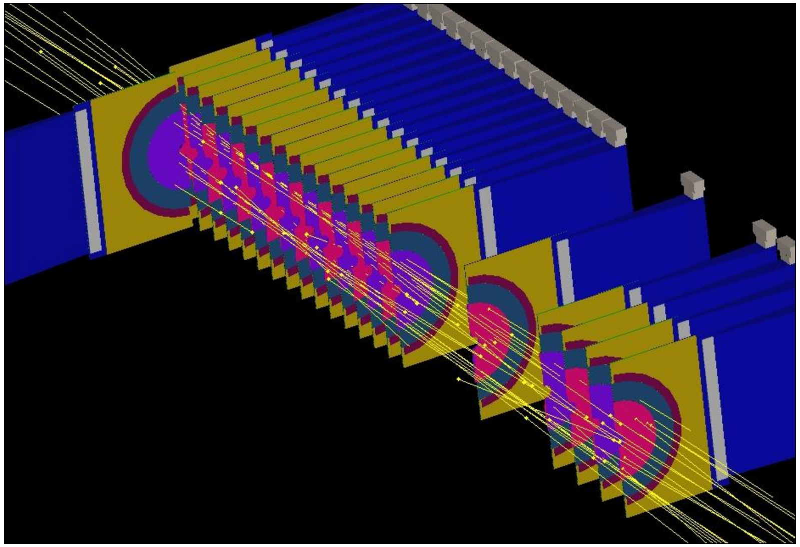


**Photo during installation**

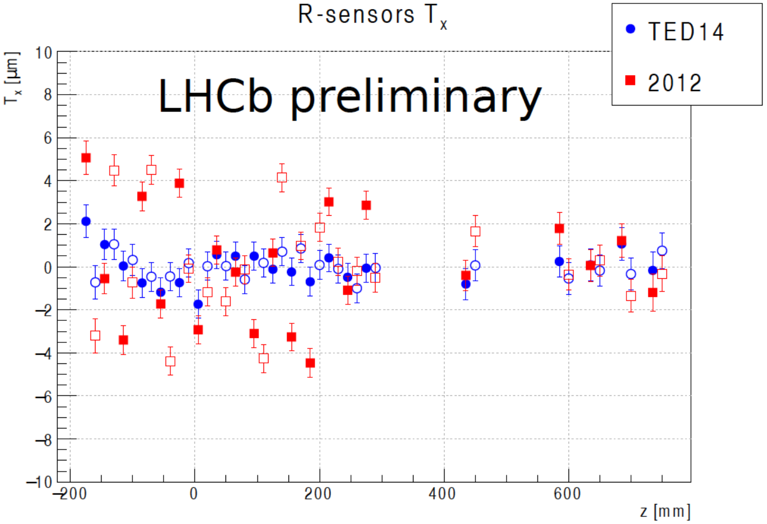


# TED data in the VELO

- Particle trajectories are almost parallel to the beam direction
- Very good sample for VELO alignment
- These kind of data was used in 2008 and 2009 for the detector commissioning



- LHC injection test in November 2014 provided a small data sample used to evaluate VELO alignment
- Observed small variations as expected:
  - ~3µm for Tx and Ty, ~30µrad for Rz
- Improvements observed in the monitoring plot based on residuals



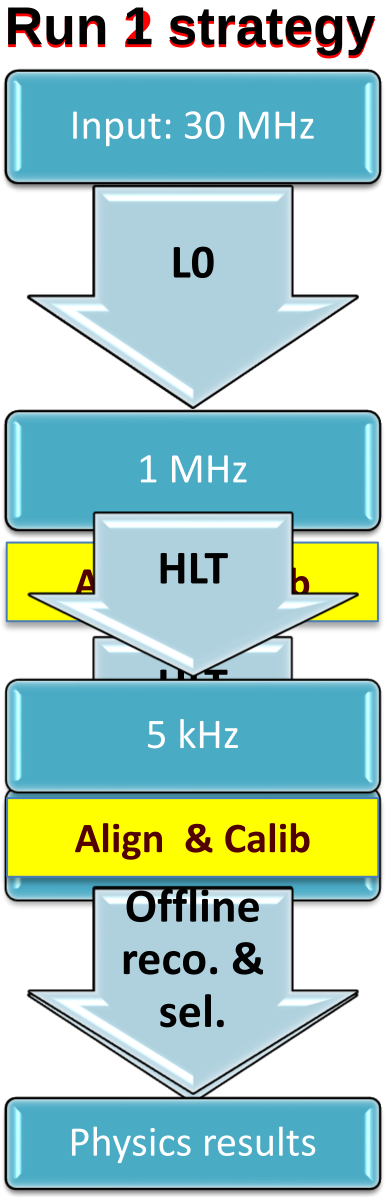
# Real-time alignment

## Run 1 strategy

- Run HLT with preliminary alignment (and calibration)
- Final alignment and calibration evaluated for the end of the year reprocessing data
- Final alignment and calibration for the data used for reprocessing during long TS

## Run 2 strategy

- Minimization of the online and offline differences
- Evaluation of the alignment before HLT
- Run HLT and offline with same alignment and calibration
    - Used for all physics results during run2
    - Data reprocessing foreseen only at the end of Run2

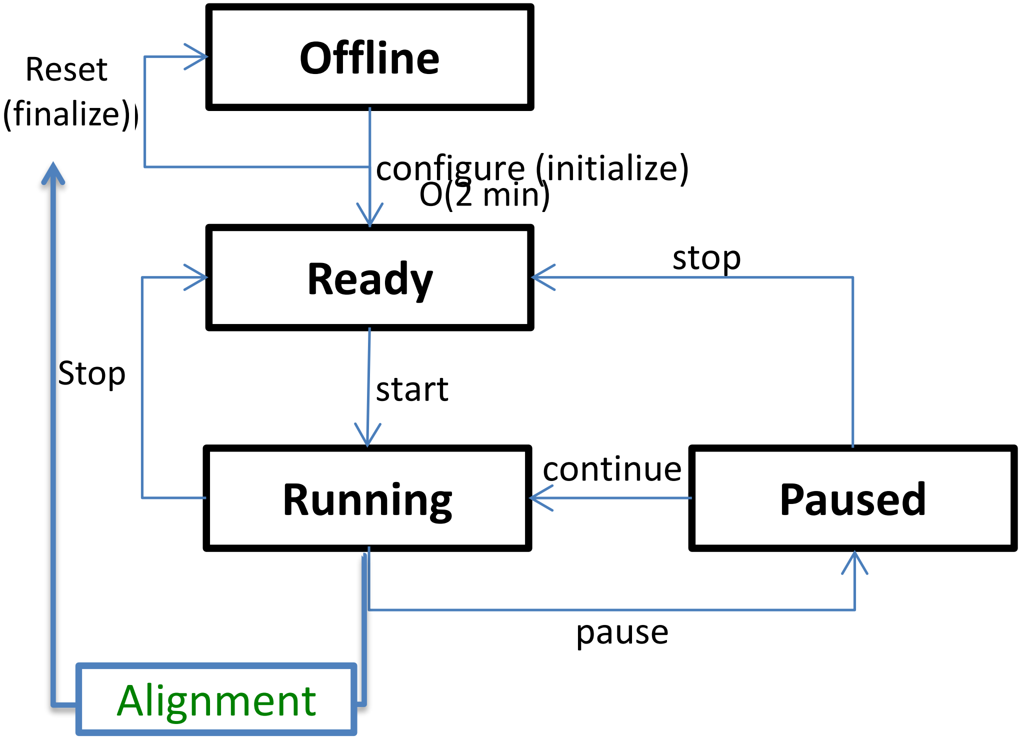


# Automatic procedure for real-time alignment

- **Procedure**
    1. Collect enough data with a dedicated selection for each alignment
    2. Run the alignment
    3. Compare old/new alignment
    4. Update the alignment constants if needed
- Alignment method based on a kalman filter [**NIM A600 (2009) 471, NIM A472 (2013) 48]**
- Implemented in the online system
- Parallelization on several nodes (up 1500) for the reconstruction
- Minimization performed in a single node
- Evaluation of the alignment in O(min)

**Job configuration**

parallelization on several nodes



Offline

Reset (finalize)

configure (initialize) O(2 min)

Ready

stop

Stop

start

Running

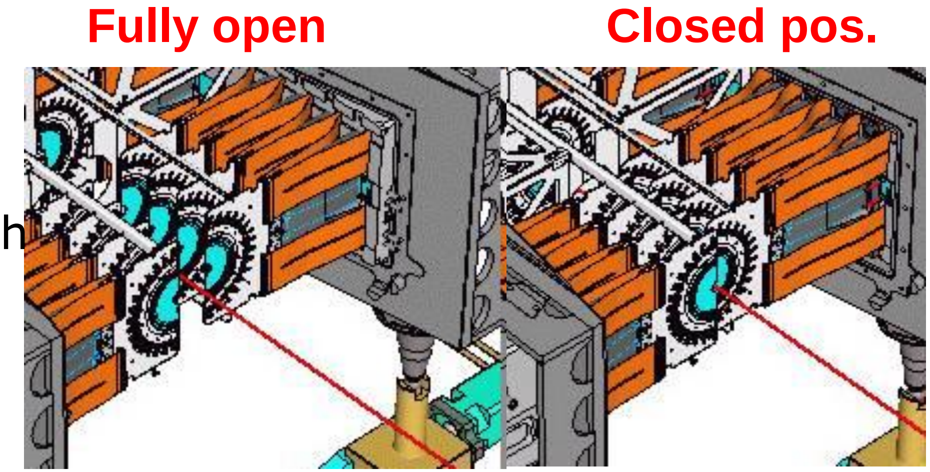
continue

Paused

pause

Alignment

# LHCb alignment



- VELO centred around the beam for each fill when the beam declared stable
- Stability of 2 half alignment
  - x: RMS 3.7 µm ; max var. ± 9 µm
  - y: RMS 2.5 µm ; max var. ± 6 µm

○ Alignment to be determined at the begin of each fill

- Tracker system align wrt VELO using mass constraint ($J/\psi$, $D^0$ mass)
- Preliminary studies show time variation over a period of about 2 weeks, partially due to magnet polarity switch

○ Alignment update expected each few weeks

| Number of alignment constants | |
|---|---|
| VELO | 86 |
| TT | 135 |
| IT | 64 |
| OT | 496 |

# Summary

- **AIDA Alignment Overview Web Page**
  - http://aidasoft.web.cern.ch/node/31

- **AIDA alignment package**
  - **BACH:** First implementation in telescope simulation
  - AIDA alignment package fully implemented in Kepler
  - →Proves alignment principle works in real-life situation

- **LHCb VELO alignment**
  - TED runs show improvements over 2012 alignment
  - Work on an automatic alignment procedure for Run II is ongoing

- **LHCb Velo Upgrade alignment**
  - Implementation of alignment in LHCb Upgrade software is ongoing

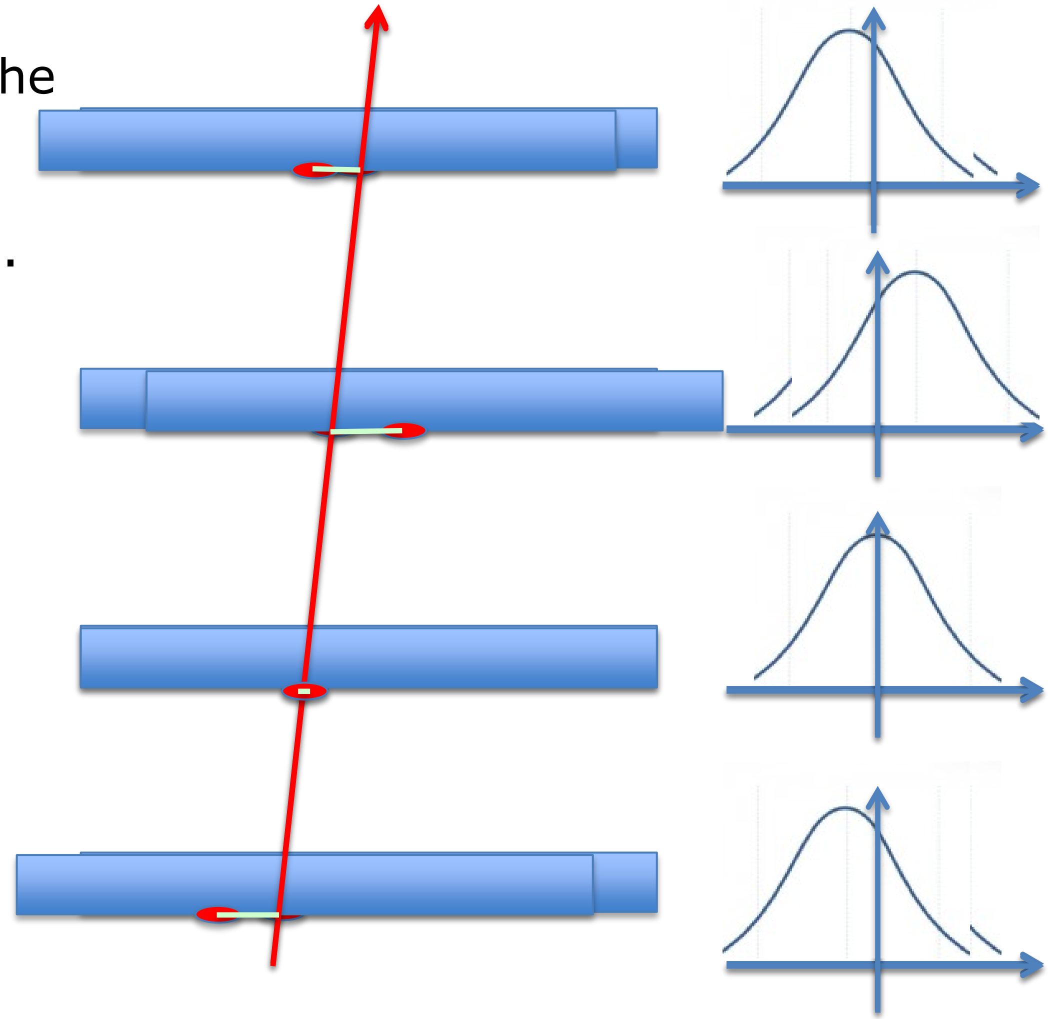
# Backup-slides

# Track based alignment

- In a perfect aligned detector the residual-distribution (distance between measured hit and track) is centered around zero.

- Distribution gets shifted (spread), when modules are misaligned.

- Residual depends on
  - *track* (local) parameter
  - *alignment* (global) parameter.



# Weak modes

- Detector deformations, that have **no impact on** $\chi^2$
- Solution is blind to multiple minima

For parallel tracks:

Linear transformations:

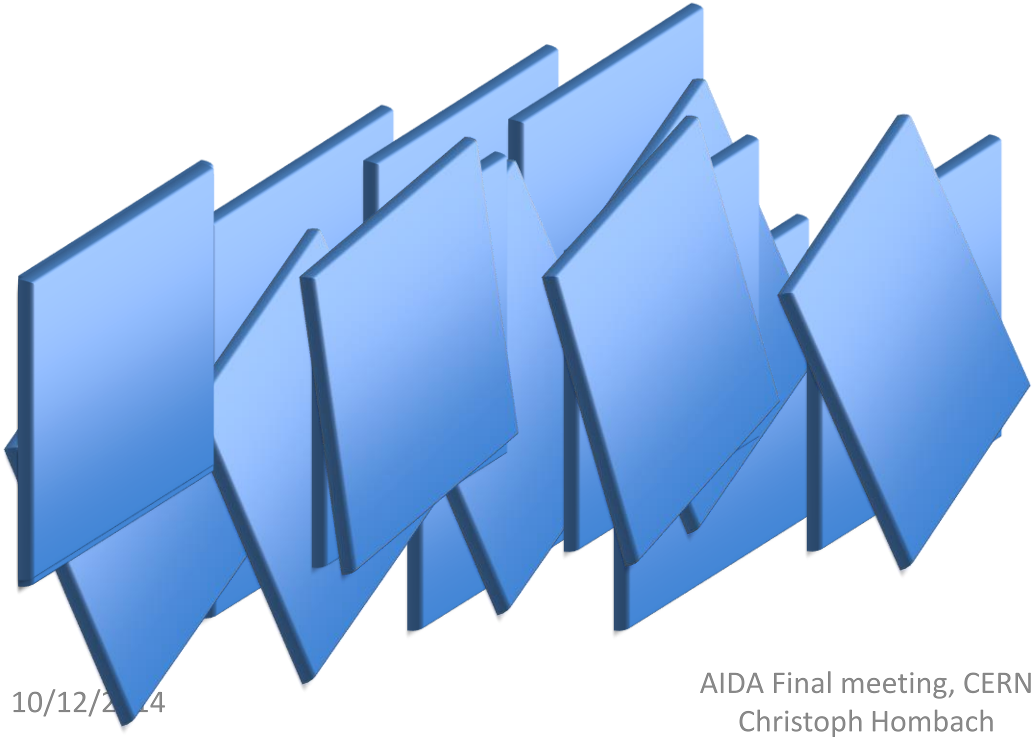
global Translation

global Rotation

Twist

Shearing

Scaling



# Dealing with weak mode

- Constraints, like fixing module position
- Constraint-equations
  - additional terms to $\chi^2$ that depend on alignment parameters
  - Like, set average translations to 0
- Use set of tracks with different characteristics
  - Parallel tracks, vertex tracks

cartoons