More Alignment …
Testbeam Preparations

Tadeáš Bilka
Charles University Prague

F2F Tracking Meeting Prague, Dec. 11. – 13. 2013

Outline

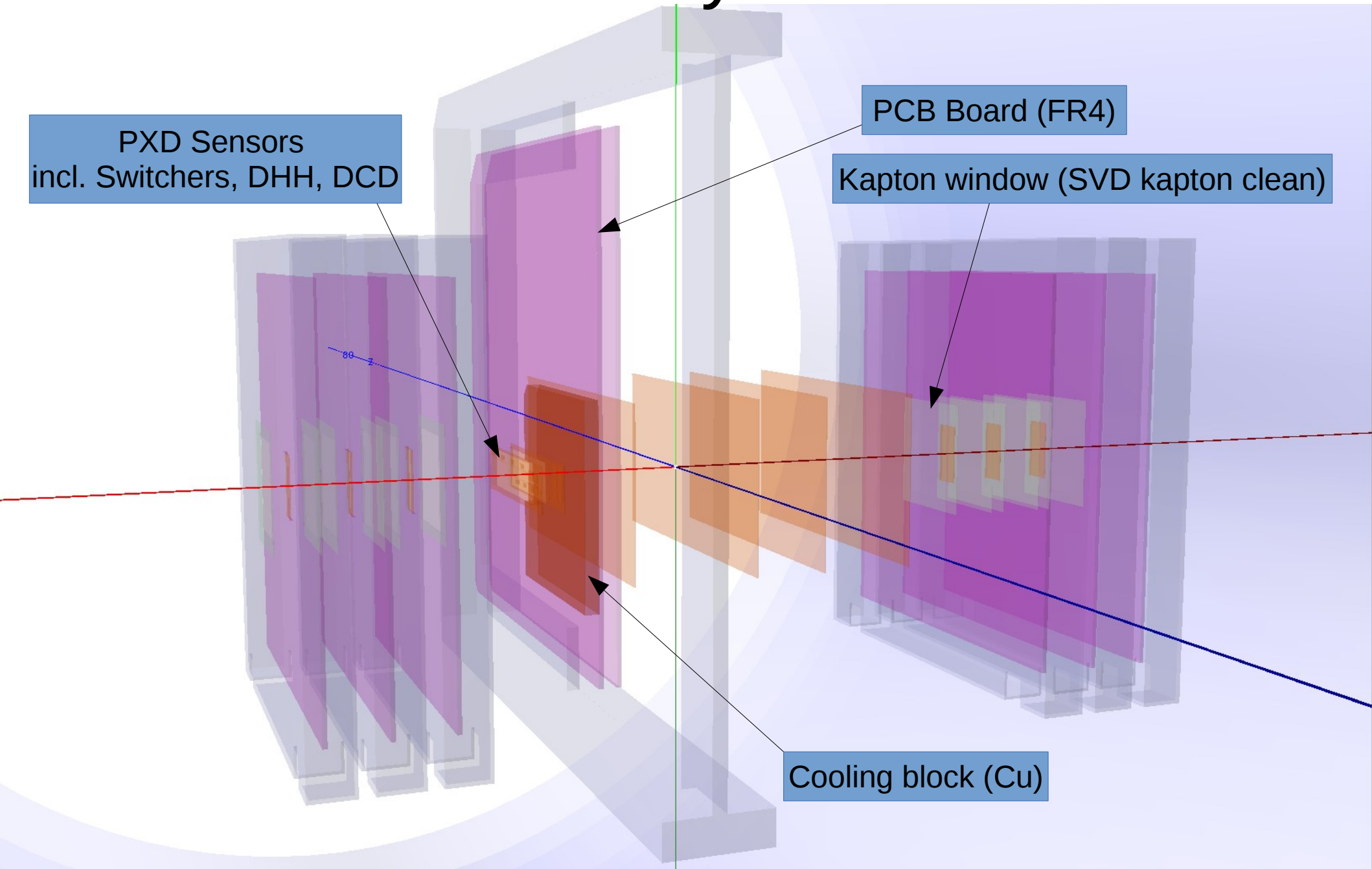
- Geometry for TB
- Beam Simulation in TB
- Telescope Integration
- General Broken Lines in GENFIT2
- Millepede II Alignment
- Conclusion & Discussion

Geometry for TB

- New more detailed geometry in svn
  - Updated geometry creator to support boolean solids
  - VXD setup according to drawings from Carlos
  - Structures far from beam not exact (PCB Frame, no SVD support) … still some dimensions/material unknown
  - „Baseline“ setup = centered along beam axis
- Telescopes = 6 x PXD matrix: 1152 x 576 pixels, 2 x 1cm

  - `FullTelescopeVXDTB.xml`
  - `FullVXDTB.xml` … telescopes deactivated

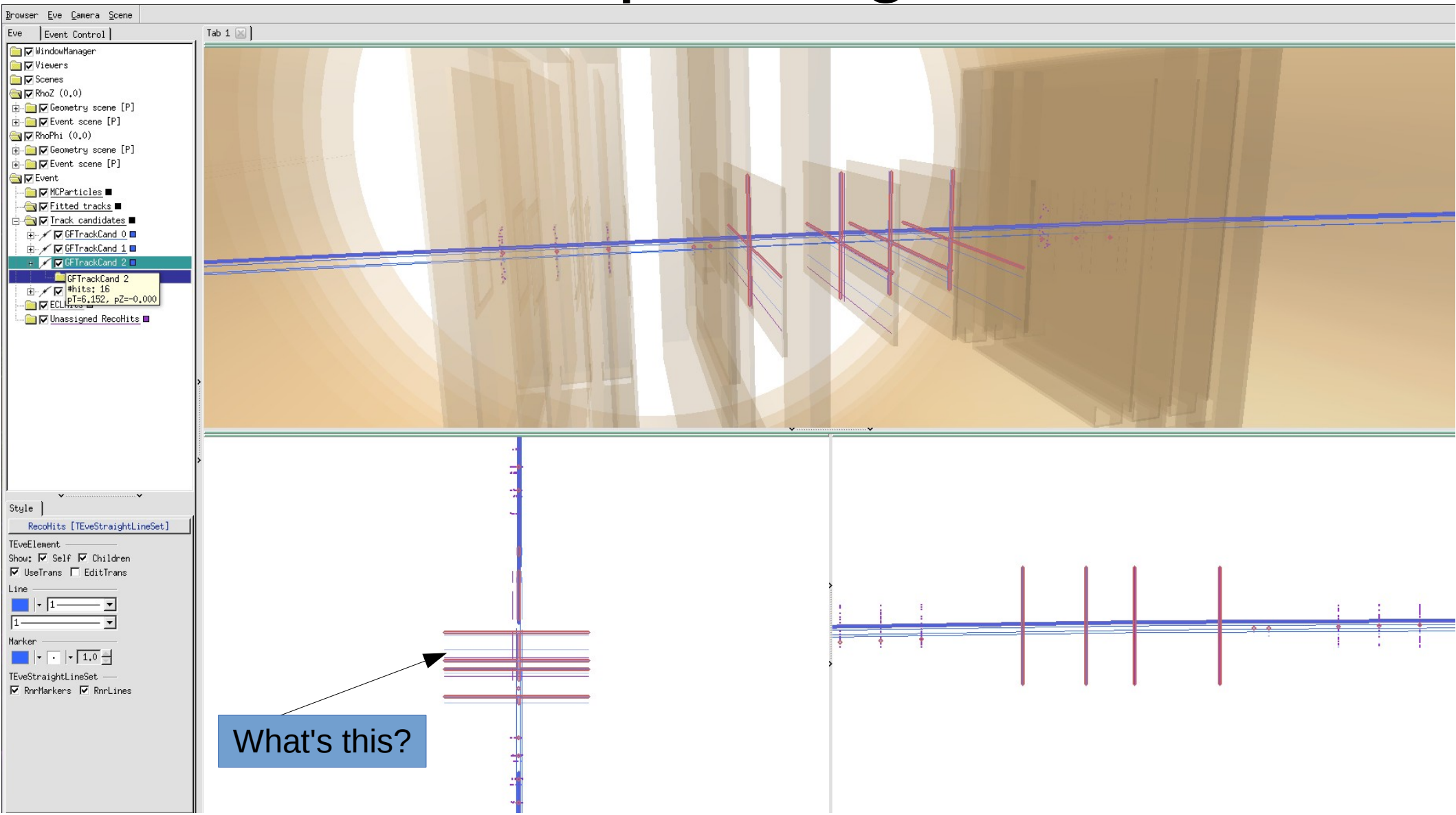
Geometry for TB



Telescope Integration

- **Idea**: Use PXD sensors as replacement
  - Matrix 1152 x 576 pixels
- Change telescope PXD to binary readout
  - Add `ChangePXDDigitsToBinary` module into path before PXDClusterizer
- Adjust cluster cache size in PXDClusterizer
  - `param('ClusterCacheSize', 576)`
- Example script `FullTelescopeVXDTB.py`

Telescope Integration



Beam Simulation in TB

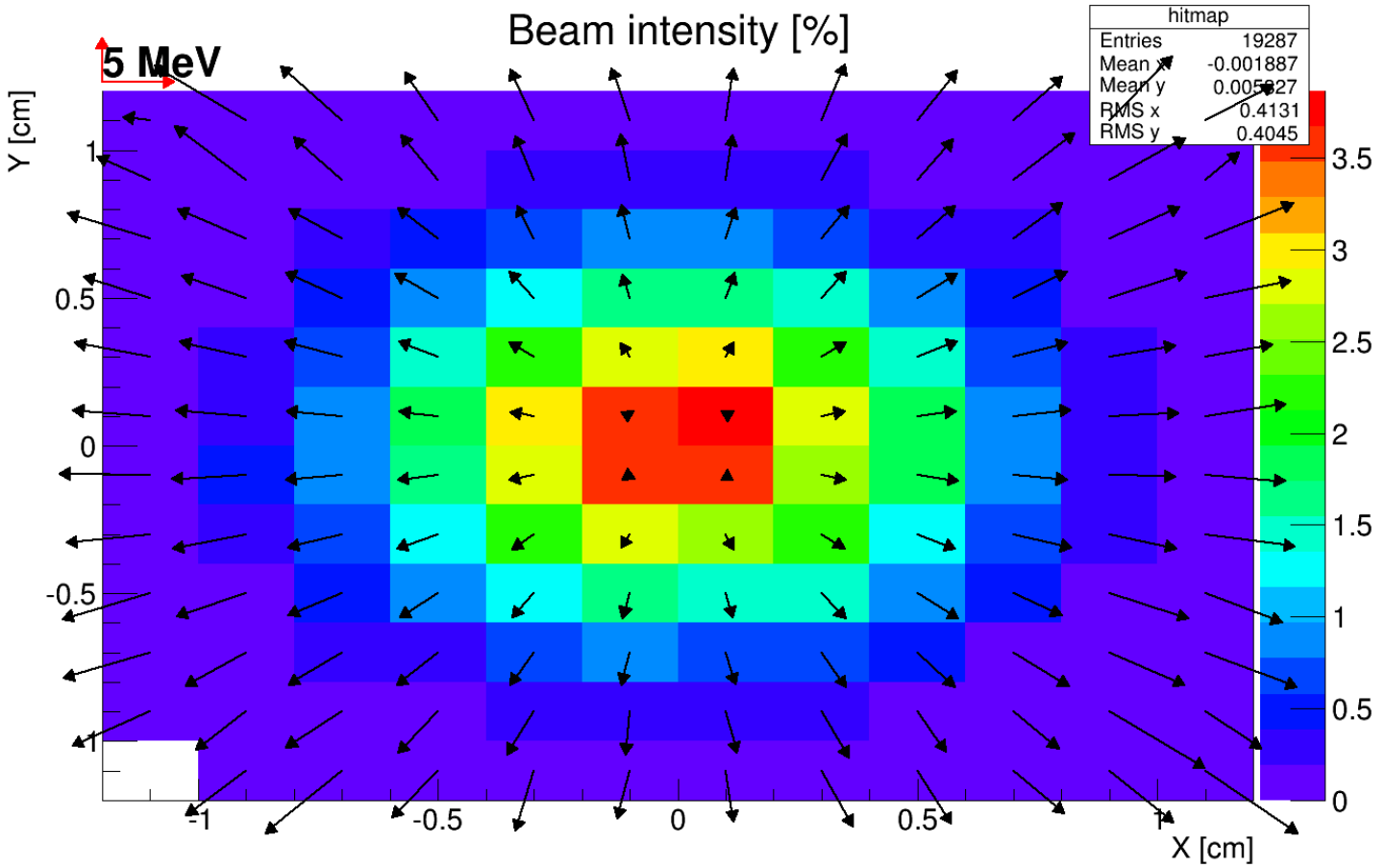
- Based on Benjamin's study. Beam properties important for VXDTrackFinder
- Simulation of beam propagation in 15m of air from collimator to our setup
  - + Investigation of influence of Wiena setup
  - Geometry in `BeamLine.xml`
- Simple module `BeamspotScan` in svn
  - Analyze TrueHits of special „scanning“ PXD sensor to get beam properties
  - Histogram average and RMS of momentum and slopes, average intensity in each bin
  - Output data for array of particle guns and for drawing of momentum vector field
  - To run it, use `BeamScan.py`

Beam Simulation in TB

- Example script BeamSimulation.py to simulate recorded beam properties. **Or real measured beam data (e.g. from Benjamin) can be loaded and simulated – just create the data file.**
- Creates a particle gun for each bin.
  - Loads `beamscan.dat` file (see it in svn)
    - Settings: number of bins in x/y, min x/y, max x/y (but x=z in TB)
    - Each bin (one line): (ρ, p, σ_p, φ, σ_φ, θ, σ_θ)
  - Intensity controlled by nTracks param
  - Uniform vertex generation in each bin
  - Normal momentum and angle generation
- Example ROOT macro drawField.C to draw vector field
  - please read comments and adjust scale and legend arrow if arrows too big/small

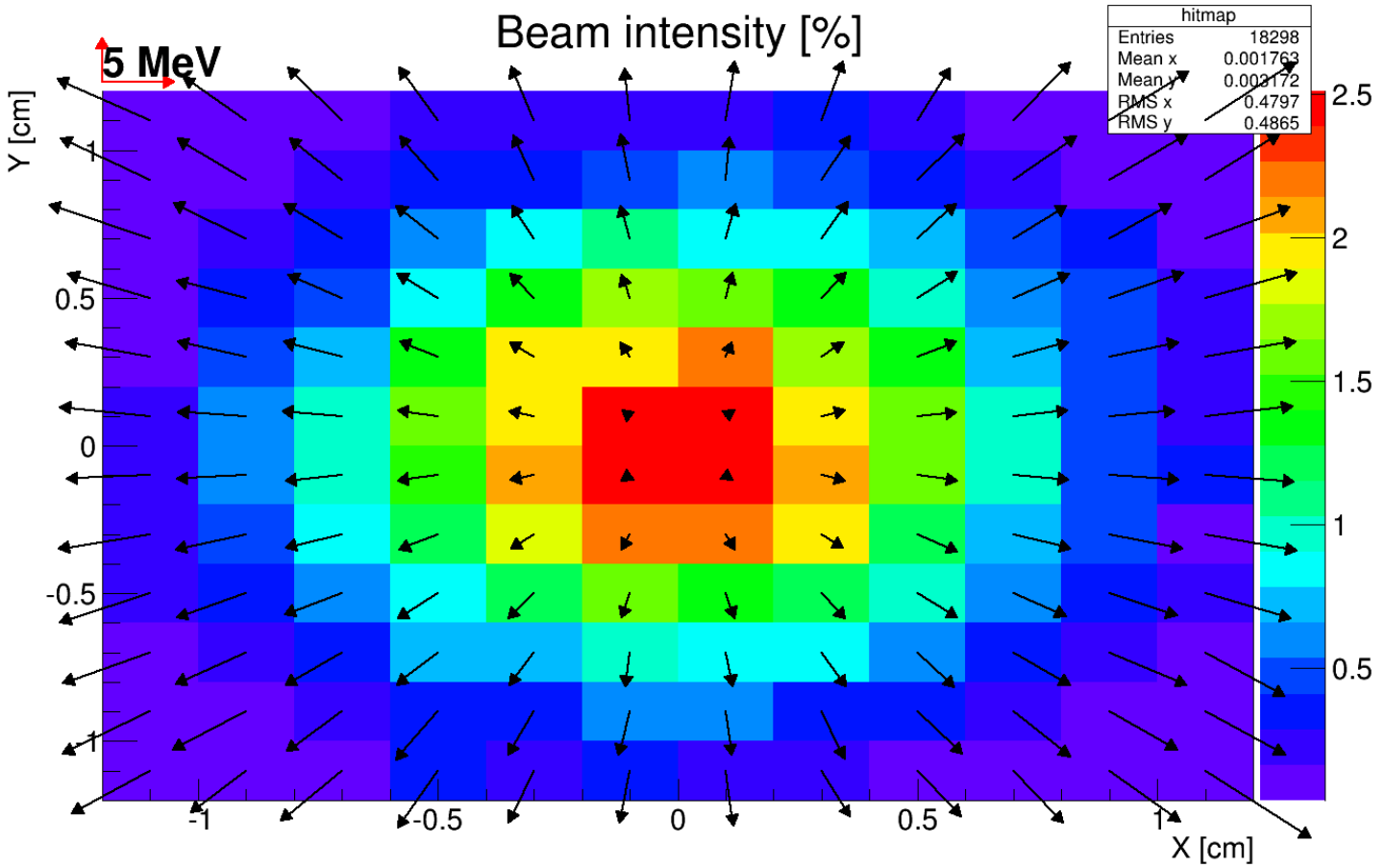
Beam Simulation in TB

Average Z/Y momentum after 15m air



Beam Simulation in TB

Average Z/Y momentum after 5m air + 6 x 320um Si + 10m air



General Broken Lines in GENFIT2

- Currently GBL inside GENFIT2, but to use it in basf2, you must change the Genfitter module
  - Simplified interface to GBL in place … but this works only for TrueHits
  - You have to call GBL from GenfitterModule.cc

```
454        if (fitSuccess) {
455          B2DEBUG(99, "       Chi2 of the fit: " << kfs->getChi2());
456          //B2DEBUG(99,"       Forward Chi2: "<<gfTrack.getForwardChi2());
457          B2DEBUG(99, "       NDF of the fit: " << kfs->getBackwardNdf());
458          //Calculate probability
459          double pValue = gfTrack.getFitStatus()->getPVal();
460          B2DEBUG(99, "       pValue of the fit: " << pValue);
461          // Everything succeeded: start GBL
462
463        m_gbl.processTrack(&gfTrack);
464        }
```

  - Also update GenfitterModule.h

```
 16 #include <genfit/DAF.h>
 17 #include <genfit/GFGbl.h>
104     std::string m_resolveWireHitAmbi;          /**< Determines
105     std::vector<double> m_beamSpot;            /**< The coordin
106   genfit::GFGbl m_gbl;
```

General Broken Lines in GENFIT2

- Update of the GBL interface to work with clusters prepared
- But since Friday, I still haven't finished all debugging
- I discovered a change in behavior of construction of track candidates
  - **Before**: SVD U and V clusters were 2 TrackPoints each with 1 raw 1D measurement
  - **Now**: SVD U and V clusters are on the same TrackPoint as 2 raw 1D mesurements

???

General Broken Lines in GENFIT2

- Old interface cannot take into account material outside sensors
- With dead material, fitting results not optimal (bad scatterig error estimation)
- My new interface (scattering point at each material boundary) not yet updated to work with clusters and to produce alignment data
- First alignment tests with old interface using clusters ongoing...

Millepede II Alignment

- First tests with clusters not too much succesfull
  - Millepede complains about bad quality of tracks (too many rejections)
  - Sometimes Millepede crashes on input data
  - Sometimes matrix of alignment parameters not with full rank (seems data for some alignment params missing)
  - I still don't have missalignment. Now testing only ideal alignment. If alignment finishes, with clusters, there is some shift in ideal geometry (?!) … but with some complaints from Millepede
- BUT: I used to have similar problems before and after some days of debugging I got rid of them. So there is a chance I can solve this:-)

Discussion

- Telescope integration
  - Do we want VXDTrackFinder for telescopes? Is it possible?
  - Are we going to rerun track finding offline with telescope data?
- Running GBL in basf2
  - A module for GBL fitting is prefered over current workaround.
  - What parameters for the module? GBL settings (outlier rejection settings), ? Additional GBL iteration, ? More material treatment strategies (e.g. attach air to next scatterer)
  - Store GBL track data / fit status in DataStore or just output data for alignment and some histograms?
- Running MP2 in basf2
  - Where should we check quality of tracks for alignment and using what criteria?
  - Should I write a module which will wrap MP2 alignment?
    - What settings should be available? Or just link the steering file?
  - What about the alignment chain? GBL fit → track selection → MP2 → Store alignment → Update RecoHits → Refit
    - I can write a module which will take MP2 results and tranform it into XML + Update recohits using VXDAlignmentManager
    - But we should somehow attach alignment data to each run (or maybe period of time) ... database?

Conclusion

- We might be able to align VXD almost „online“ using Millepede in basf2...
- But still some things have to be understood, some issues solved and quite some code written (but not too much :-)...

Thank you for your attention!

Backup

Geometry Questions

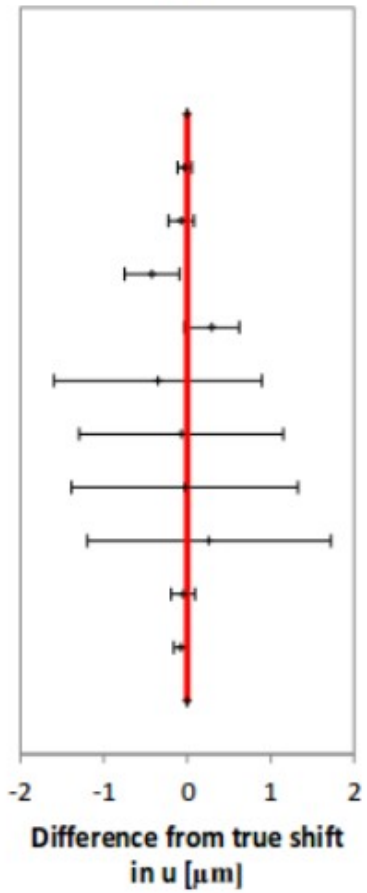
- If you are not here but know an answer, please mail to tadeas.bilka@gmail.com
  - What is the thickness of Cu coating on the PCB board?
  - What are the materials PCMAG layers are made of?
  - What are the dimensions (thickness) of EUDET sensor box and PCB board?
  - Can someone provide drawing of an SVD sensor?
  - How can I control the color in ROOT viewers?

Alignment / Misalignment Simulation

- How precisely misalignment can be reconstructed?
- B = const = 1T
- 15k tracks of primary electrons @ 6GeV/c
- True misalignment subtracted from Millepede estimation on plots for errors to be visible

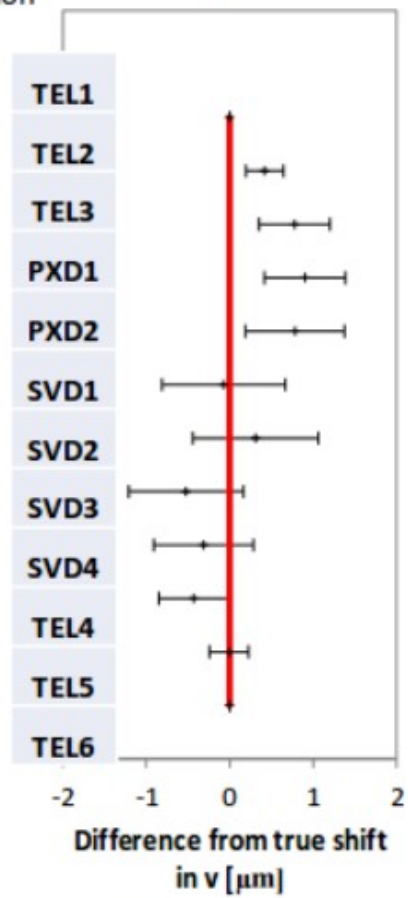
True Shift [μm]	MP2 Estimation [μm]
0	0,0 ± 0,0
10	10,0 ± 0,1
-62	-62,1 ± 0,2
8	7,6 ± 0,3
-62	-61,7 ± 0,3
-76	-76,4 ± 1,2
-26	-26,1 ± 1,2
237	237,0 ± 1,4
-22	-21,7 ± 1,5
-133	-133,1 ± 0,1
-81	-81,1 ± 0,1
0	0,0 ± 0,0

u



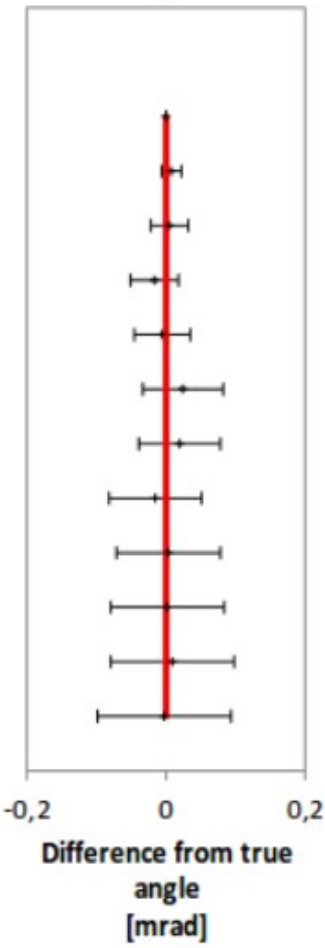
True Shift [μm]	MP2 Estimation [μm]	
0	0,0 ± 0,0	TEL1
41	41,4 ± 0,2	TEL2
-64	-63,2 ± 0,4	TEL3
38	38,9 ± 0,5	PXD1
26	26,8 ± 0,6	PXD2
-89	-89,1 ± 0,7	SVD1
-163	-162,7 ± 0,8	SVD2
-81	-81,5 ± 0,7	SVD3
32	31,7 ± 0,6	SVD4
90	89,6 ± 0,4	TEL4
-74	-74,0 ± 0,2	TEL5
0	0,0 ± 0,0	TEL6

v



True Angle [μm]	MP2 Estimation [mrad]
0	0,0 ± 0,0
0,5	0,5 ± 0,01
-0,84	-0,8 ± 0,03
1	1,0 ± 0,03
-1,79	-1,8 ± 0,04
1,75	1,8 ± 0,1
0,68	0,7 ± 0,1
-0,37	-0,4 ± 0,1
0,8	0,8 ± 0,1
1,2	1,2 ± 0,1
-1,8	-1,8 ± 0,1
0,4	0,4 ± 0,1

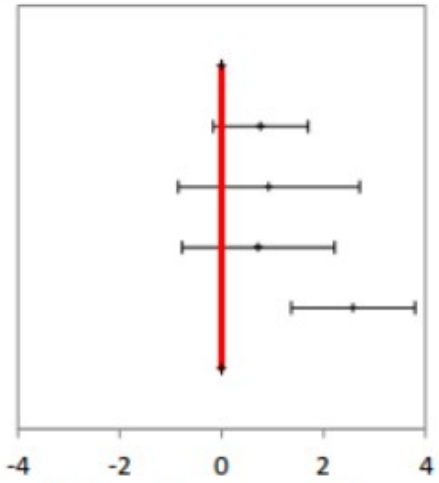
γ



Deviations from true shifts < 1 um, angles < 0.1 mrad

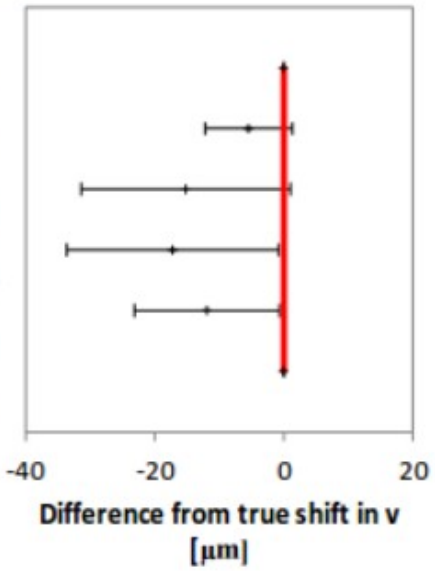
Alignment / Misalignment Simulation

u	True Shift [µm]	MP2 Estimation [µm]
PXD1	0	0,0 ± 0,0
PXD2	-62	-61,2 ± 0,9
SVD1	-76	-75,1 ± 1,8
SVD2	-26	-25,3 ± 1,5
SVD3	237	239,6 ± 1,2
SVD4	0	0,0 ± 0,0



- B = const = 1T
- 15199 Tracks
- 6 GeV/c, beamspot 2x2mm uniform
- Setup moved by 20 cm closer to gun
- Additional scatterer to have total 0.5 X0
- **Without Telescopes**

v	True Shift [µm]	MP2 Estimation [µm]
PXD1	0	0,0 ± 0,0
PXD2	26	20,6 ± 6,7
SVD1	-89	-104,2 ± 16,2
SVD2	-163	-180,2 ± 16,4
SVD3	-81	-92,9 ± 11,2
SVD4	0	0,0 ± 0,0



γ	True Angle [µm]	MP2 Estimation [µm]
PXD1	0	0,0 ± 0,0
PXD2	-1,79	-2,0 ± 0,3
SVD1	1,75	2,1 ± 1,0
SVD2	0,68	0,9 ± 1,2
SVD3	-0,37	-0,3 ± 1,5
SVD4	0,8	1,5 ± 1,7