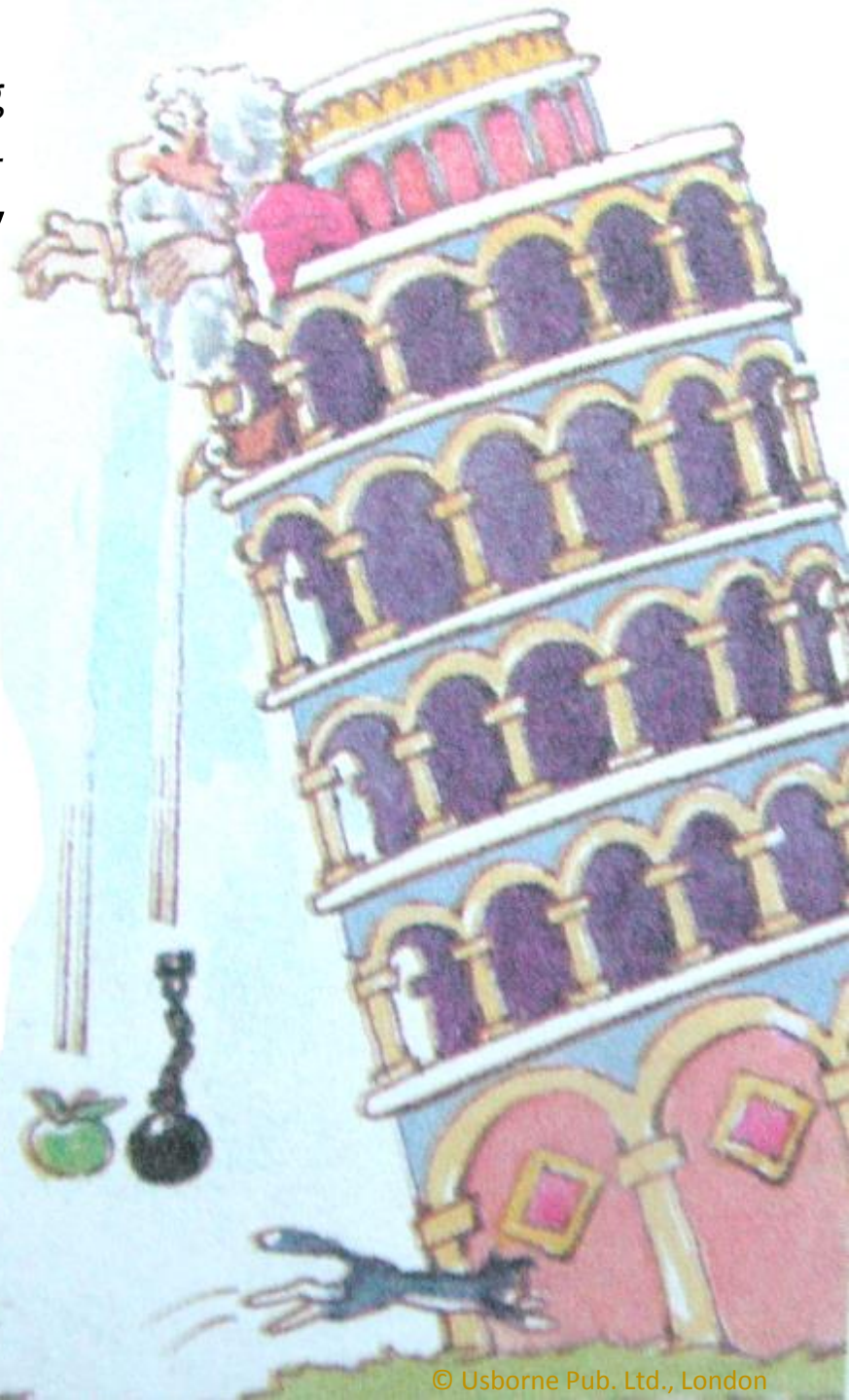


Belle II F2F Tracking Meeting
29 – 30 Sep 2014
Pisa, Italy

VXD Alignment

GBL Fitter in basf2

Tadeáš Bilka
Charles University in Prague



Overview

- ❑ Alignment in basf2: Overview
- ❑ What happened since last F2F?
- ❑ New Stuff: GblFitter
 - How it works?
 - How to use it?
 - Options
 - Some results
- ❑ TODOs & Plans

- ❑ Conclusion

Alignment and Fitting in basf2: Overview

- ❑ We use Millepede II for alignment
 - Linear least squares minimization w.r.t. all alignment and track parameters
 - https://www.wiki.terascale.de/index.php/Millepede_II
- ❑ Tracks have to be fitted with General Broken Lines (GBL) for alignment
 - Will produce Binary files = Millepede II input
 - <https://www.wiki.terascale.de/index.php/GeneralBrokenLines>
- ❑ Interface to GBL within GENFIT2
 - Inside GENFIT2: GBL + GBL interface which can convert `genfit::Track` to `GblTrajectory`
 - Experiment – independent: any geometry, sensor orientation, material distribution, combination of 1D (strip/wire) and 2D measurements (pixel or combined strips), non-homogeneous B field / no field
- ❑ In basf2: GBLfit Module
 - Starts as GenFitter Module by initialization and construction of `genfit::Track`
 - Compatible 1D SVD hits combined into 2D measurements
 - Fits the track
 - Continues as GenFitter: creates `TrackFitResult`, stores everything into `DataStore`

What happened since last F2F?

❑ Last F2F tracking meeting decisions:

- Make GBL a full and independent fitter (independent from Kalman stuff)
- Update PXD/SVD RecoHits with alignment interface (derivatives, labels)
- Solve dependency of alignment to TB package

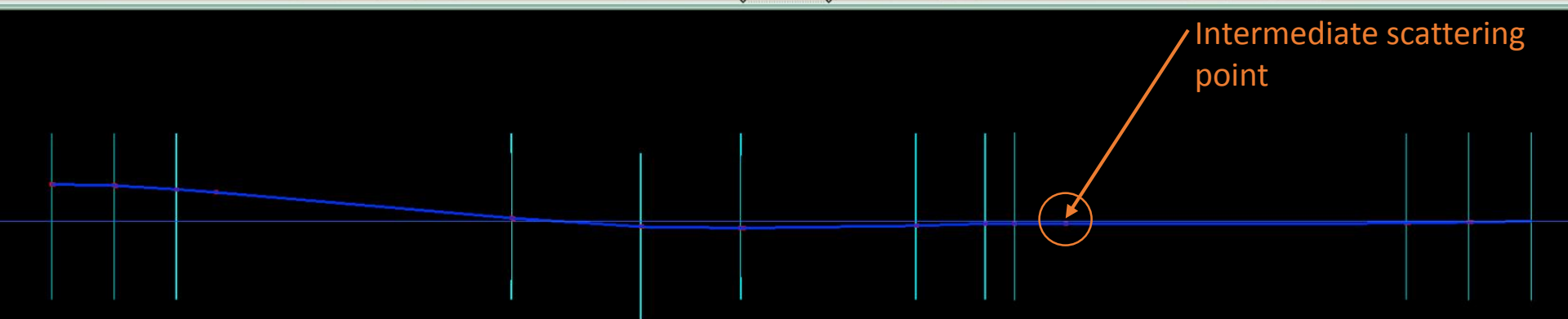
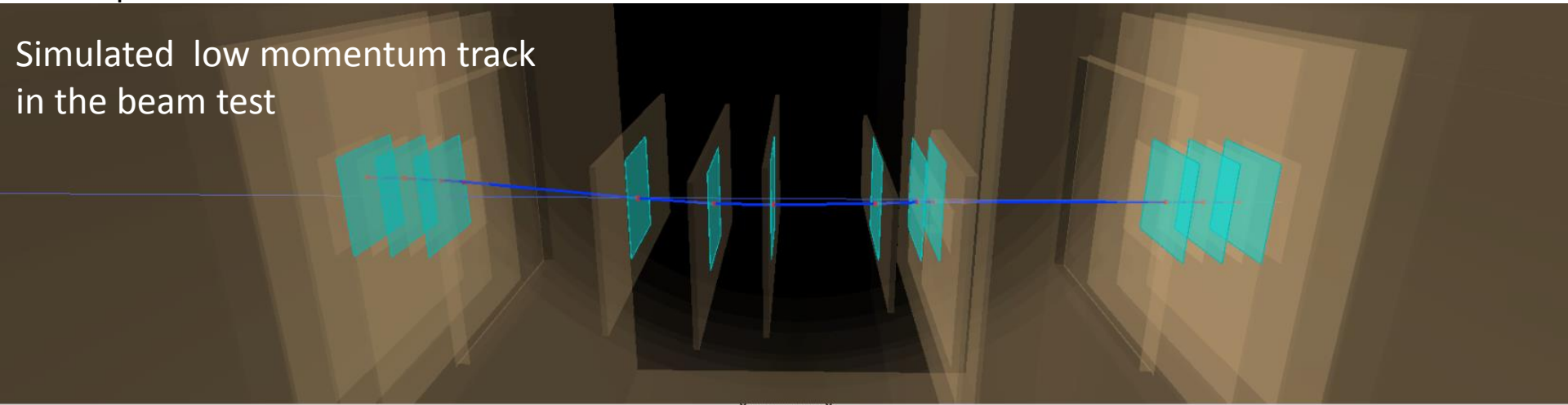
❑ The status now:

- New interface to GBL with 3 new classes:
 - GblFitter ... Independent full fitter
 - GblFitStatus ... simple class holding fit results (ndf, p-value)
 - ICalibrationParametersDerivatives ... interface for RecoHits supporting alignment
- RecoHits updated to provide alignment derivatives
- Independent full fitter?
 - No relation to Kalman stuff in GENFIT2
 - The track is updated with fit results and changes to „broken track“ – non-measurement scattering points added, track state changes at each point
 - The track is stored to DataStore – you should be able to just change the fitter from Kalman to GBL and everything should work the same, but:
 - Biased/unbiased → before/after kink
 - No getBackwardUpdate(), getForwardUpdate() ...
- Dependency to TB is automatically switched on if TB package is installed
 - Thanks to Thomas Kuhr

GblFitter: How it works?

- ❑ Initial track parameters from track finder propagated to get the reference (unbroken) trajectory
- ❑ From material distribution between measurements, positions and MS variances of scatterers are computed
- ❑ All info (residuals, Jacobians ...) passed to GBL and fitted
- ❑ Update of states with fit results

Simulated low momentum track
in the beam test



GblFitter: How to use it?

```
#include <genfit/GblFitter.h>
```

```
Track* track;
```

```
// ... track construction or retrieval from DataStore
```

```
GblFitter fitter;
```

```
// Process for all track representations
```

```
fitter.processTrack(track);
```

Getting the fitted states:

Don't forget the
intermediate
scatterering
points

There are 2 states:
Before (false) and after
(true) a kink, which
differ in slopes (u' , v')

```
int id = 1; // index of point on track (includes scatterers)
```

```
AbsTrackRep* rep = track->getCardinalRep(); // default track representation
```

```
TVectorD stateBeforeKink = track->getFittedState(id, rep, false).getState();
```

```
TVectorD stateAfterKink = track->getFittedState(id, rep, true).getState();
```

```
TMatrixDSym covarianceAfterKink = track->getFittedState(id, rep, true).getCov();
```

Getting the fit results:

```
double ndf = track->getFitStatus(rep)->getNdf();
```

```
double chi2 = track->getFitStatus(rep)->getChi2();
```

GblFitter: Options

Options of the fitter

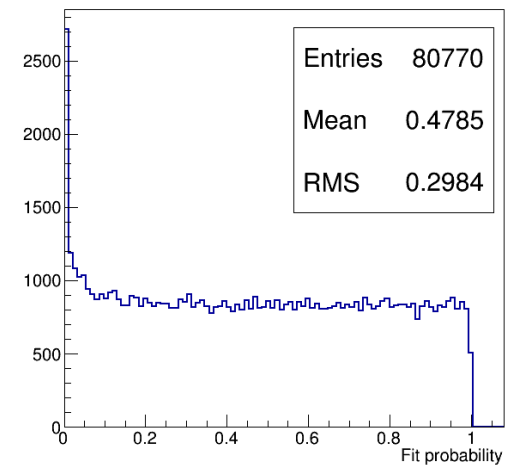
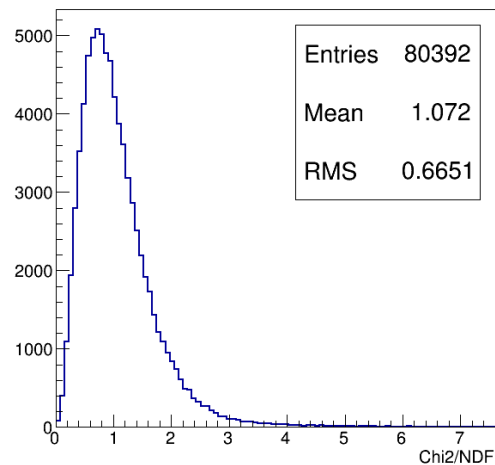
```
fitter.setOptions(  
    internalIterations,  
    enableScatterers,  
    enableIntermediateScatterer,  
    externalIterations,  
    recalcJacobians  
);
```

Can be set via GBLfit Module parameters

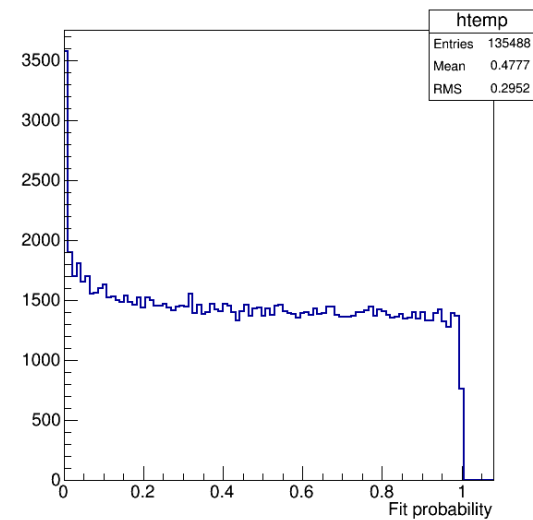
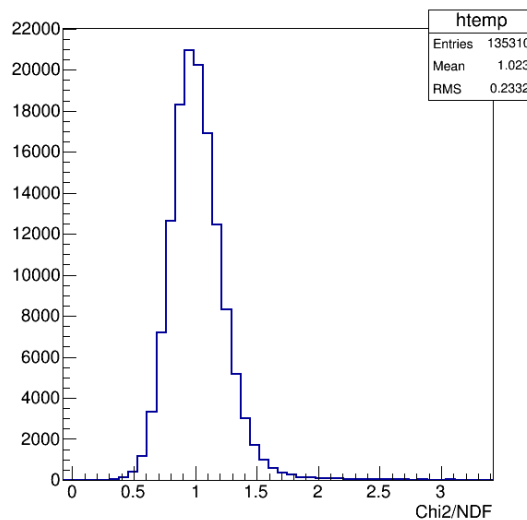
- **internalIterations** is a string setting the internal down-weighting for the GBL fit, usually an empty string meaning no down-weighting is sufficient, see the GBL manual.
- **enableScatterers** enables placing thin scatterers in the trajectory. If set to false, no multiple scattering effects will be added
- **enableIntermediateScatterer** switches between simulation of thick scatterers between measurements (true) and only thin scatterers at measurement planes (false). See the implementation bellow.
- **externalIterations** is integer defining number of times, the trajectory is fitted by GBL and updated with results. Usually one iteration is sufficient.
- **recalcJacobians** defines number of time Jacobians will recalculated and states re-extrapolated with results of the fit after external iteration. If set to zero, Jacobians are not recalculated. If set to i , the update is done after i^{th} external iteration.
- Hit resorting added (for CDC) - set by GBLfit module

GblFitter: Some results

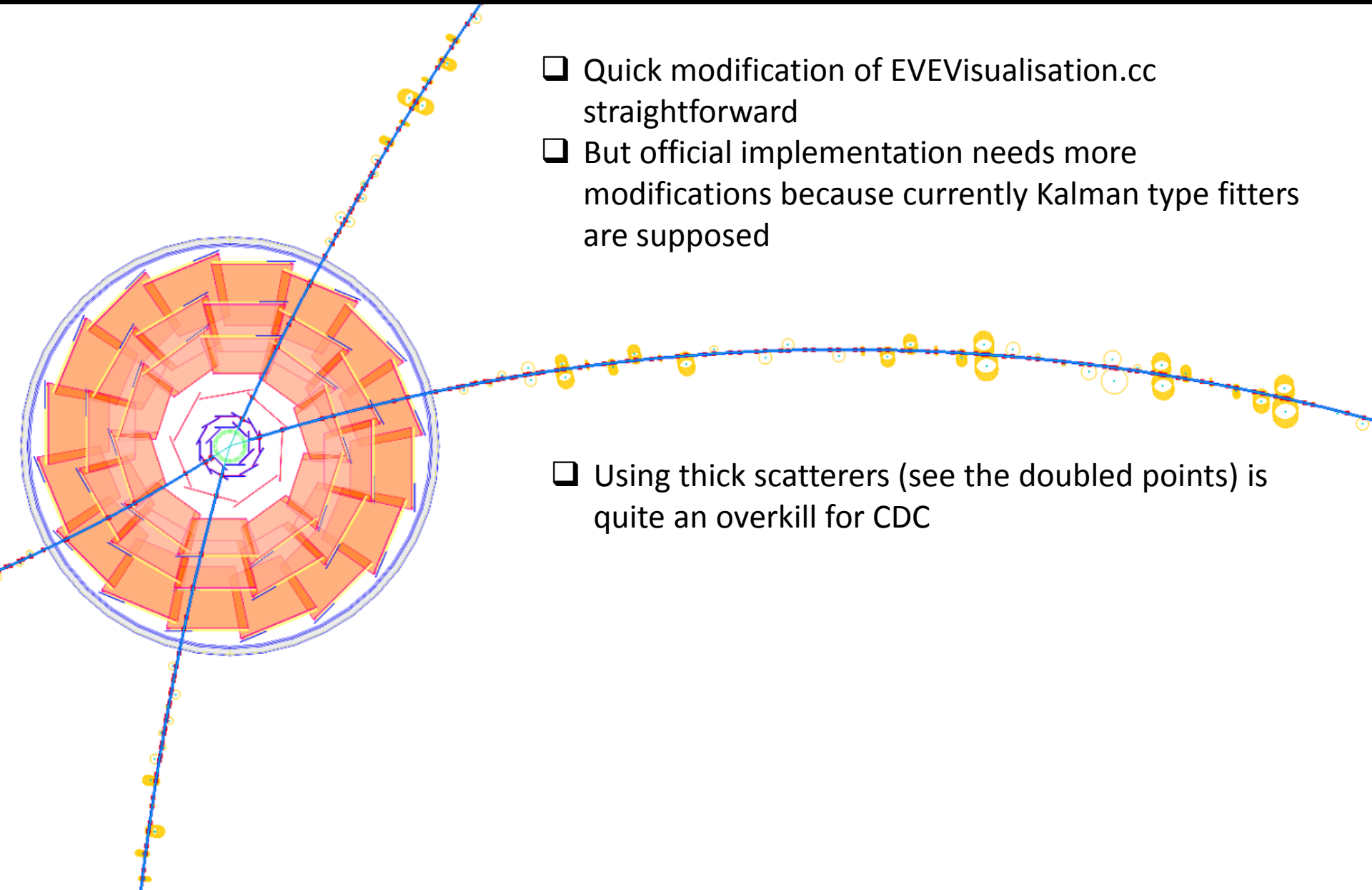
- Track fit with TrueHits in VXD only



- TrueHits in VXD+CDC



GBL Trajectories in (my) Event Display



- ❑ Quick modification of EVEVisualisation.cc straightforward
- ❑ But official implementation needs more modifications because currently Kalman type fitters are supposed

- ❑ Using thick scatterers (see the doubled points) is quite an overkill for CDC

TODOs & Plans

- ❑ Change in SVD hit handling in GENFIT
 - Only affects hit combination before track is passed to GblFitter
 - Small update of GBLfit module necessary
- ❑ Switch GBLfit Module to the new interface by default
- ❑ Update display package to show GBL trajectories
- ❑ Update of GBLfit module to the new way of working with relations

- ❑ Alignment plans
 - Start to work with misalignment simulation and perform MC studies with full Belle II
 - Start to implement alignment with multiple tracks (decays with vertex/mass constraint)
 - Start to think about CDC calibration/alignment together with VXD

Conclusion

- ❑ GBL is now independent fitter
 - You might try it in your analysis instead of Kalman

- ❑ This means the GBL fitting stuff is complete
 - Only some small updates for new features expected (+ bug-fixes, improvements in documentation etc.)

- ❑ The development will now focus on the calibration framework, mass/vertex constrained decays, playing with misalignment

Thank you for your attention!