

# Current Status

Thomas Madlener

Institute of High Energy Physics  
Austrian Academy of Sciences

28. Nov 2014



# What Happened Until Now?

'Achievements' of the last month(s)

- started to work with and develop *basf2*
- slowly understanding the working principle of Relations between objects and how to work with them
- start of development of *SpacePointTrackCand* class
- start of development of Converter Modules for converting *genfit::TrackCand* to *SpacePointTrackCand* and vice versa



# The SpacePointTrackCand class

A (container) class which allows grouping SpacePoints to Track Candidates

- storing SpacePoints as `std::vector<SpacePoint*>`
- storing additional information like
  - PDG code
  - charge (estimate)
  - state seed (as `TVectorD(6)`)
  - covariance seed for the state seed (as `TMatrixDSym(6)`)

- some simple getter and setter functions

- at the moment also a constructor

`SpacePointTrackCand(const genfit::TrackCand& )` **and a** `genfit::TrackCand getGenfitTrackCand()` **but this is already put into separate modules**



# Why do we need Converter Modules?

- FilterCalculator works with SpacePoints in the future (genfit::TrackCand to SpacePointTrackCand needed)
- SpacePointTrackCand to genfit::TrackCand needed for TrackFinder



# SPTC2GFTCConverterModule

from SpacePointTrackCand to genfit::TrackCand

- working with Relations between Clusters and SpacePoints
- Converting from `SpacePointTrackCand` to `genfit::TrackCand` simply by
  - getting all `Clusters` (SVD and PXD at the moment) that can be found via `getRelatedTo<ClusterType>()` or `getRelationsTo<ClusterType>()` from each `SpacePoint`
  - construct the `genfit::TrackCand` from the indices of the `Clusters`
  - if present also add additional information (PDG code, state seed, etc.)



# GFTC2SPTCConverterModule

from genfit::TrackCand to SpacePointTrackCand

## General Working Principle (pseudo code)

```
for all Cluster in genfit::TrackCand do
```

```
  if Cluster not marked as used then
```

```
    ClSpacePoints ←
```

```
Cluster.getRelationsFrom<SpacePoint>()
```

```
  SpacePoint ← choose appropriate SP from
```

```
ClSpacePoints
```

```
  TCSpacePoints ← add SpacePoint
```

```
  mark all Clusters of SpacePoint as used
```

```
  end if
```

```
end for
```

```
SpacePointTrackCand ← construct from
```

```
TCSpacePoints
```

```
SpacePointTrackCand ← add additional information
```



# Current Problems

How to choose the appropriate SpacePoint?

- No problem if relation between Cluster and SpacePoint is 'bijective' (e.g. PXD)
- No problem if there are no secondary (particle) hits in event (i.e. every Cluster has only one related SpacePoint)
- Problems arise, when a Cluster is related to more than one SpacePoint

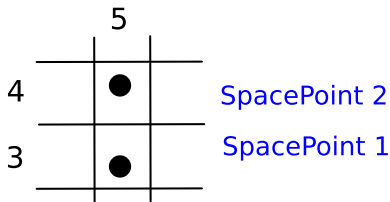


# Exemplary Problem

## An exemplary genfit::TrackCand

Considering only the problematic part of the genfit::TrackCand!

TrackCand has (SVD) hitIDs: 0, 1, 2,  $\underbrace{4, 5}_{\text{On Layer 5}}$ ,  $\underbrace{6, 7}_{\text{On Layer 6}}$ ,  $\underbrace{3}_{\text{On Layer 5}}$



Which SpacePoint to use?

Procedure:

- 1 Cluster 4  $\Rightarrow$  SpacePoint 2
- 2 SpacePoint 2  $\Rightarrow$  Clusters 4,5  $\Rightarrow$  **OK!**, mark as used
- 3 Proceed until
- 4 cluster 3  $\Rightarrow$  SpacePoint 1
- 5 SpacePoint 1  $\Rightarrow$  Clusters 3,5  $\Rightarrow$  **NOT OK!** 5 already marked used

