



# HEPHY

Institute of High Energy Physics

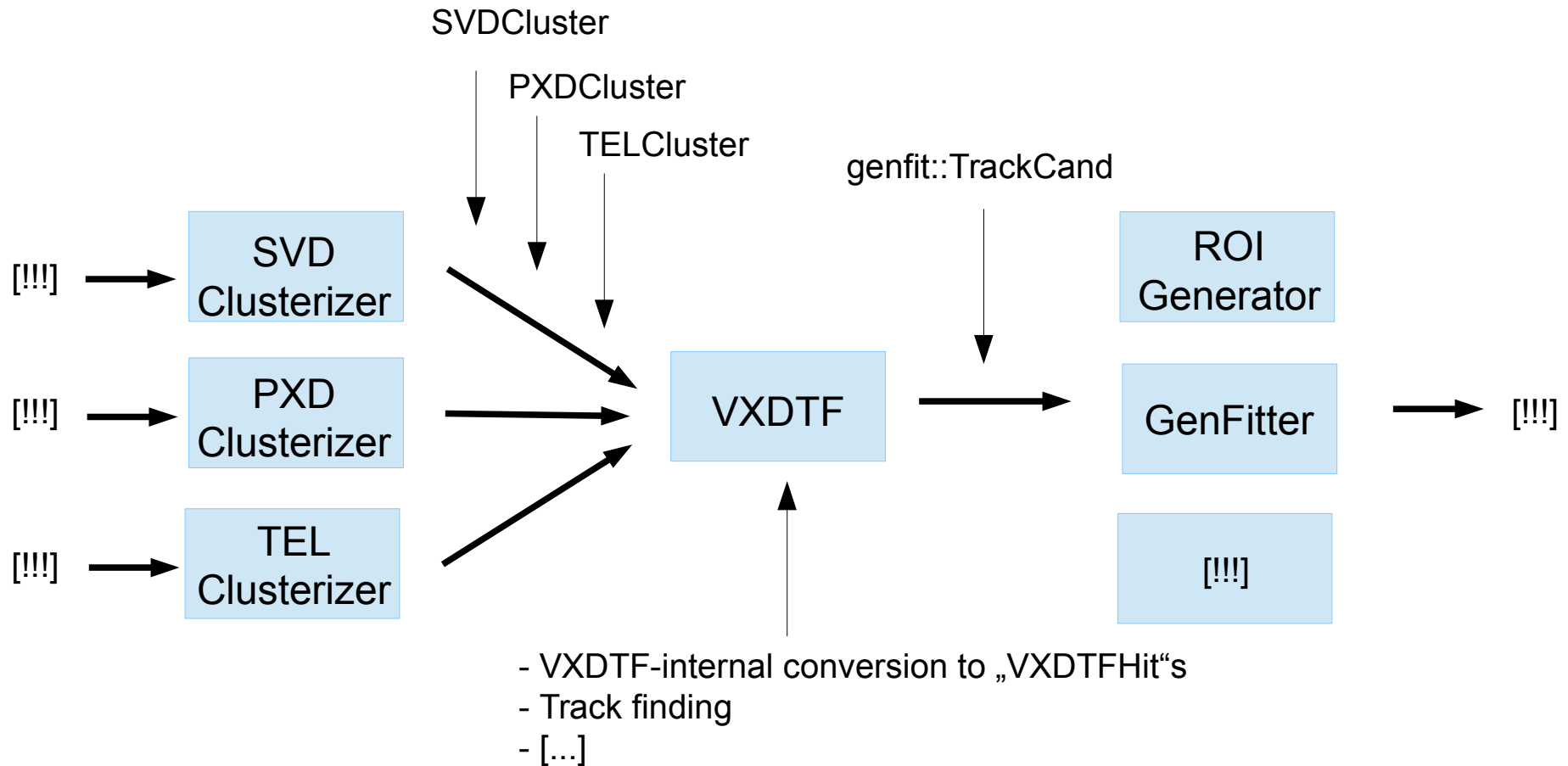
## VXD TrackFinder redesign – current approach

**Jakob Lettenbichler**

biweekly tracking meeting

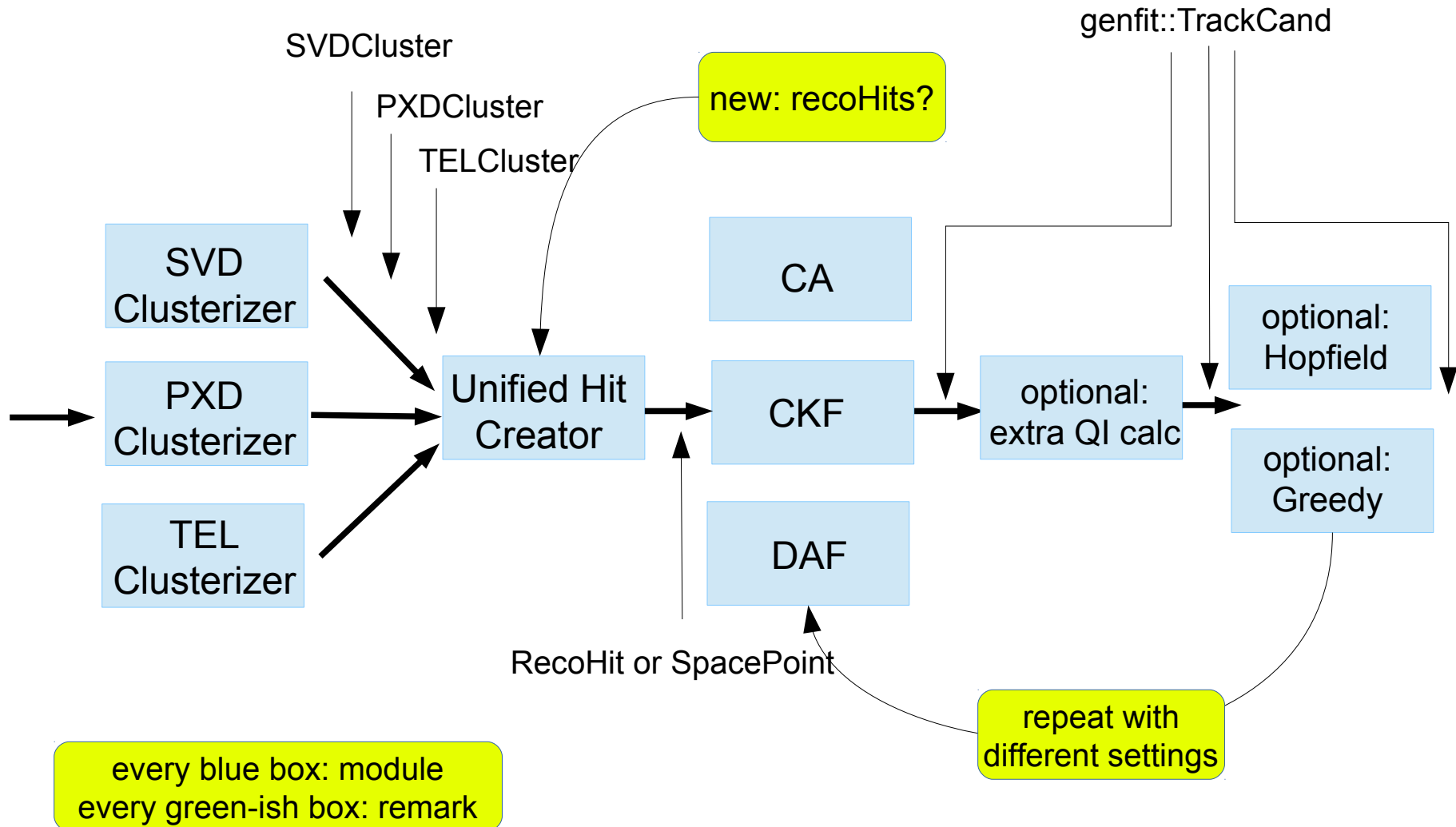
April 24th, 2014

### Current state of this part of the reconstruction chain

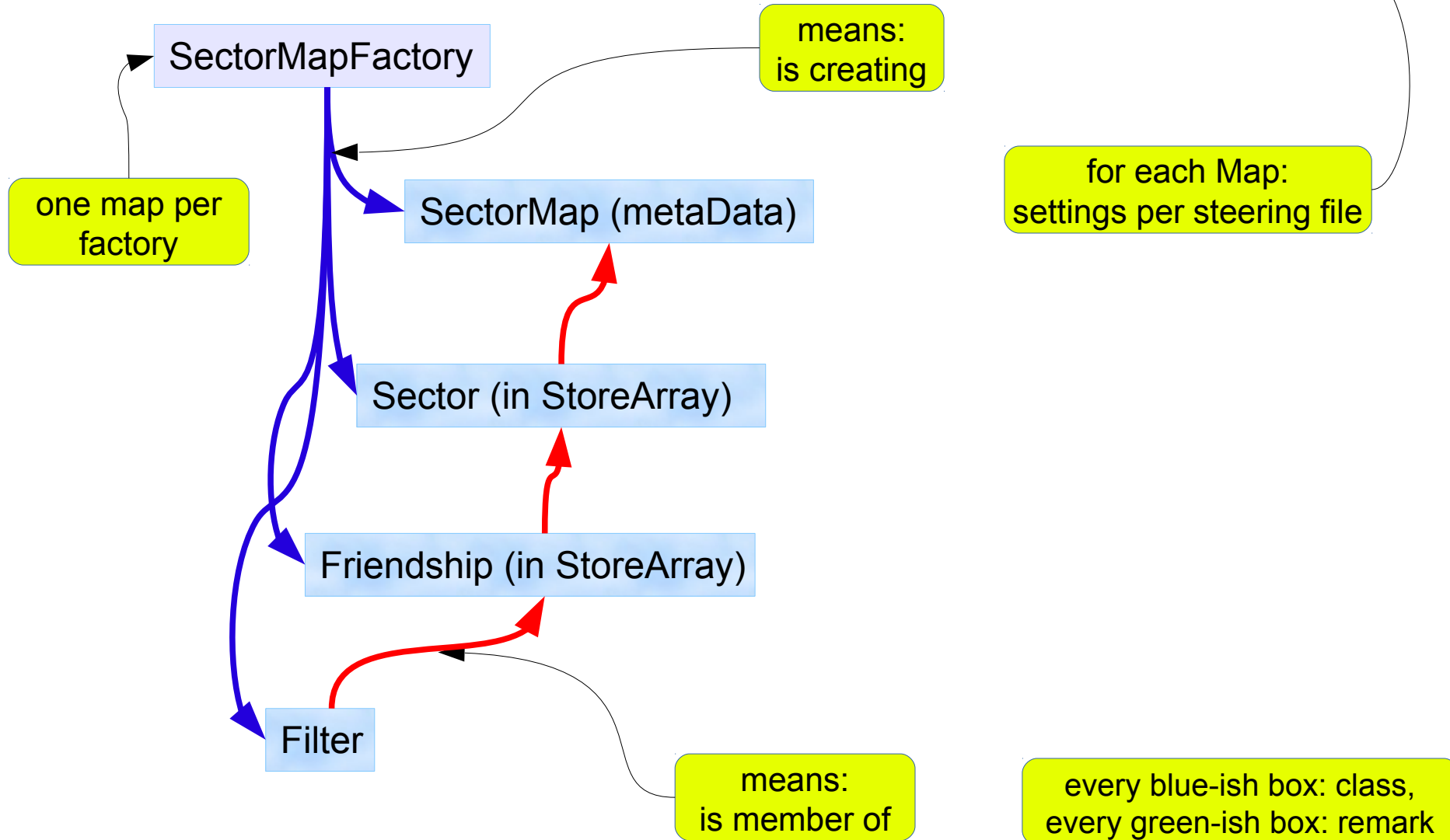


[!!!] : shortcut for „a lot of important stuff which is not part of this discussion“

### Future state of the trackFinder VXD-approach (event-part)



### SectorMapCreatorModule (does that during beginRun)



# what would be my wishlist for recoHits?

- !! detector independent interface (so far: SVD, SVD2D & PXDRecoHits)
- !! sensor-independent (->for global coordinates and error)
- !! local coordinates scaled between 0-1 (relative coordinates)
- ! some way to mark it as reserved for other TFs
- possibility to store infos about segments

Exclamation marks indicate that this would be really needed, the other stuff could be done internally too

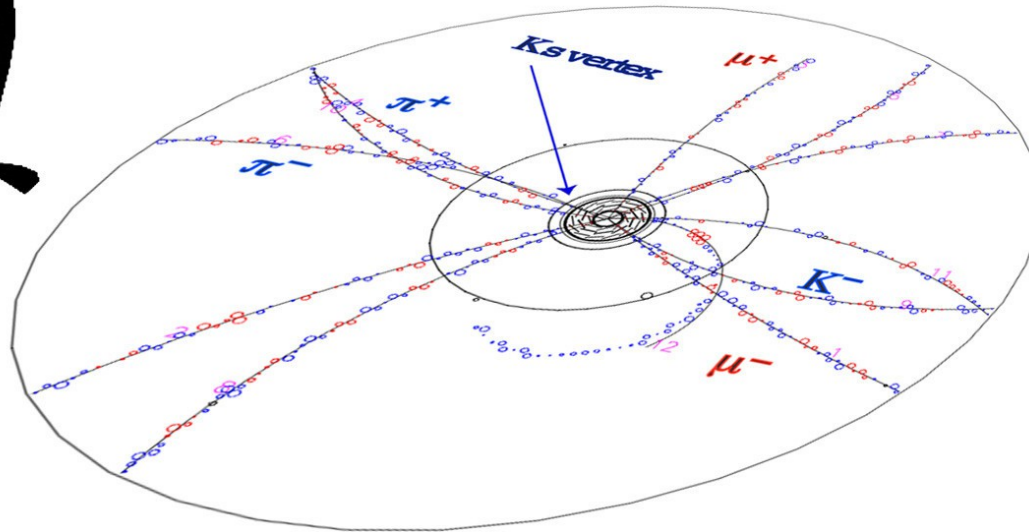
## details about the TF-design:

- currently working on the sectorMap, loaded once per run, work done by a separate module
- decoupling of static (once per run) and dynamic (per event) data
- old way of testing not good enough (e.g. 7 of ~37 classes had tests, complicated ones weren't tested)
- new design so every part can be tested





**Thank you!**

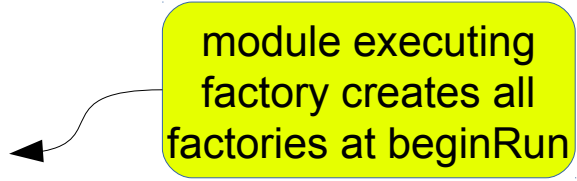


## Detailed sketch for the new sectorMap-approach

### ***SectorMapFactory***

- one factory per setup
- creates own StoreArray for its Sectors
- another StoreArray for SectorFriendship
- links them by relations and pointers(to storeArray-entries)
- creates storeObjPtr for metaData („SectorMap“)

module executing  
factory creates all  
factories at beginRun



### ***SectorMap :: storeObjPtr***

- carries metadata like name of StoreArrays
- sorted container of <Sector\*> for direct access
- defines origin/secMapCenter and sorting type

### ***Sector :: RelationArray***

- only static info like SectorFriendship
- form directed graph with other sectors (direction by secID or distance2Origin)
- container of <FriendRelations\*> for direct access
- carries ActiveSector\* (reset every event)
- carries segmentMaker called by ActivatedSector



## Detailed sketch for the new sectorMap-approach II

### ***SectorFriendship :: RelationArray***

- one Friendship allowed for each compatible combination with current sector
- combination can contain any number of sectors in chain (useful lengths: 1-3)
- Carries only the filters allowed for that combi & secMap

### ***Filter :: RelationArray***

- can be a filter for any number of hits (currently there are 2-X-hitfilters)
- Filter applied only for current Friendship → cutoffs only for local case
- creates/updates compatibilityTable for each possible hit-combi
- following filters only execute their stuff on combis which are still alive

### ***ActivatedSector (:: RelationArray?)***

- 1:1 relation to a sector of current sectorMap
- created once per event (lightweight, maybe not inheriting anything)
- container of <Hits\*> for direct access
- hits are passed to segmentMaker of Sector
- stores segments