HEPHY

Institute of High Energy Physics

Current Status of the VXD Track Finding Modules

Jakob Lettenbichler

F2F Meeting in Pisa

May 12th, 2014

Overview of the main topics of the last months

Testbeam Hamburg

- Track finding was possible
- Still some issues to deal with
- Top priority for the next weeks

VXD TF modules complete redesign

- VXDTF will be replaced by several modules
- Concentrating on test driven design
- Some basic decissions still unresolved

Bachelor students / Projects

- Stefan Ferstl, display interface, mostly completed
- Thomas Fabian, extensive Filter studies, ongoing
- Thomas Madlener, mainly Rudis student, but supporting work has to be done by me

University lectures

- Attending „Classification and discriminance analysis“
- Attending „Statistics“

Testbeam Hamburg Jan. 2014

- Extremely informative, effective and stressful weeks
- Main goal achieved, tracking and ROI finding was possible
- TF was fast enough and stayed below RAM threshold
- Many short term decisions multiplied work to be done
- Revealed issues in the TF which have to be fixed
- Topic still not closed

Open issues of testbeam related aspects

Telescope support

- On a first glance, simple task (partial TEL <-> PXD sw compatibility)
- Caused need for changing many smaller things
- Main problem: assuring that the CA stays a directed Graph without loops
- Still parked in a separate branch, needs to be merged with trunk

Thorough study of testbeam data

- Not done yet, only some runs tested locally
- Coupled with Tobias' (et al) project – real time conference
- Eventually Bachelor studend will do this

Some issues revealed by stressing the code

- Sometimes strange behavior of filters
- Bugs in the filters themselves
- Bug in producing cutoffs for secMap
- Baseline-TF is only working for very, very easy cases
- Code difficult to maintain
- Detailed tests for proving well defined behavior of the TF needed (essential)

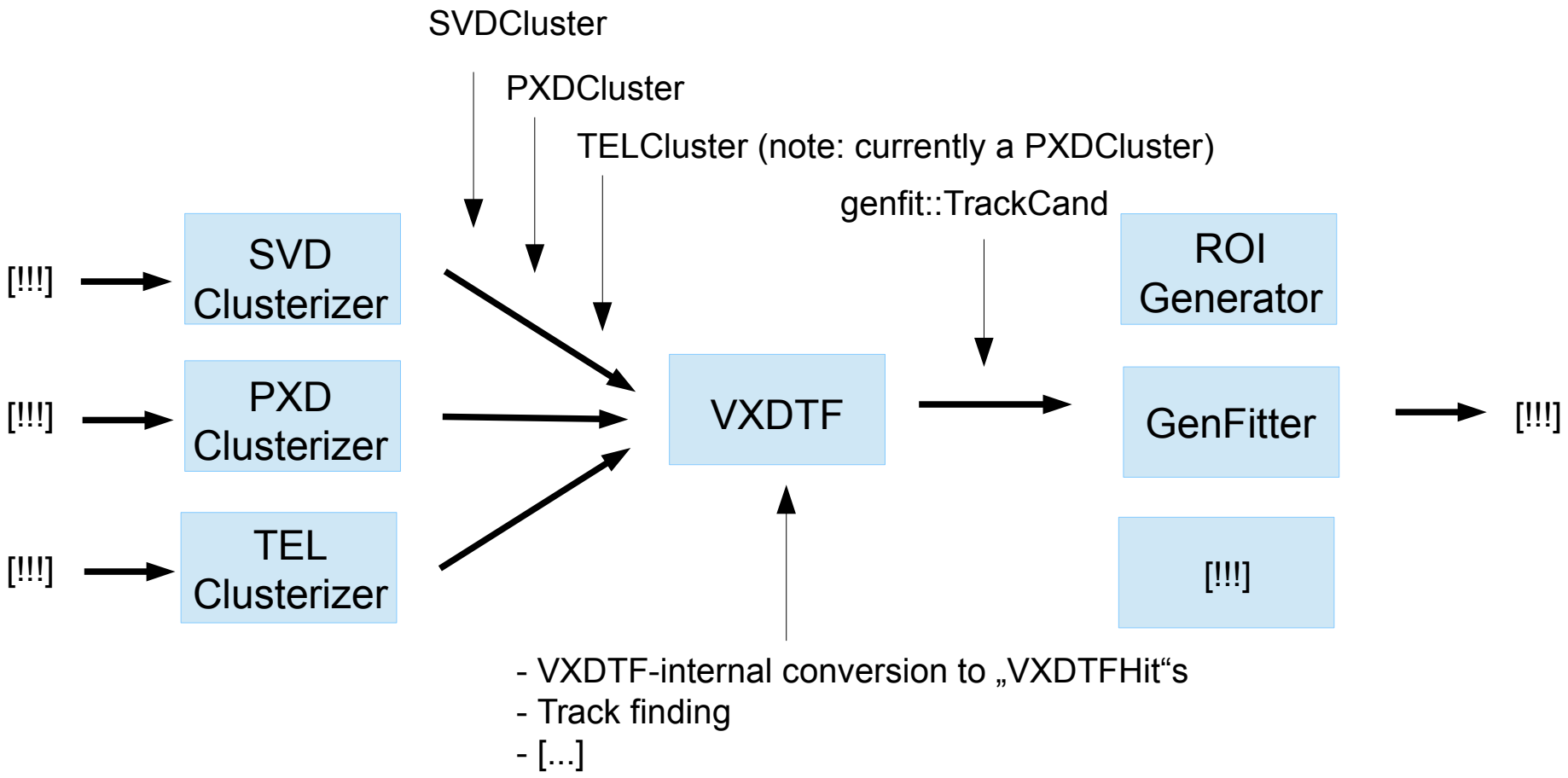
Testbeam stuff to do (ongoing):

- Merging code with current release
- Solving issues which surfaced within the last few days
  - Memory corruption resulting in a segfault
  - Creating new secMaps for everything (secMap container changed)
- Verifying Telescope support again (was working in March, still the case after merging?)
- Visual (Display) and statistical (DQM-plots) verification of VXDTF performance

VXDTF redesign

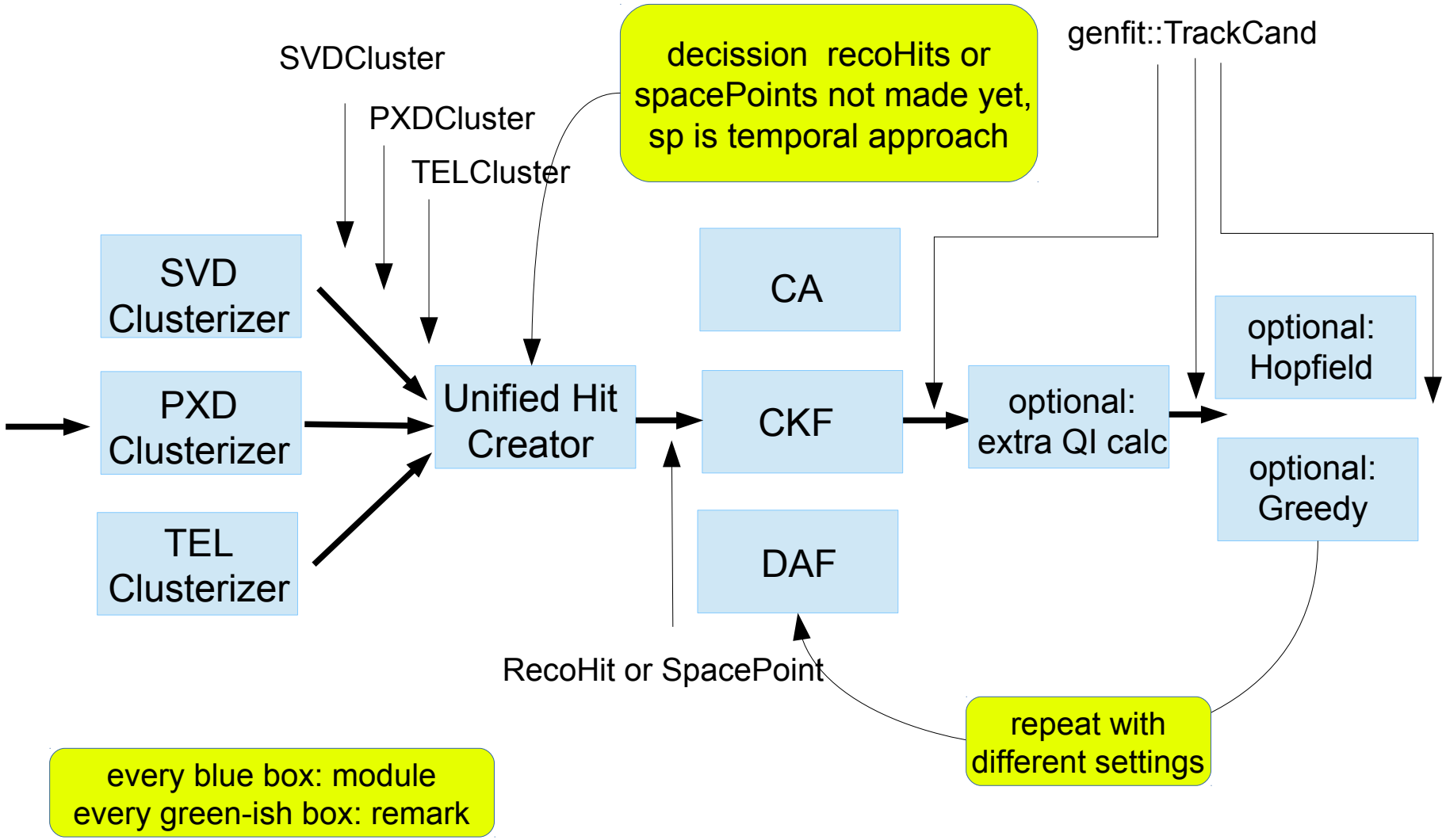
- Essential for further steps
  - Improvement of test-coverage (near 100% is the goal, so far it's 10% at best)
  - Splitting into several modules increases flexibility
  - Preparation for new TFs (CKF, DAF (both using genFit)
  - Increase maintainability of the Code
- (Of course) has to be done asap (after TB finish)

Current state of this part of the reconstruction chain

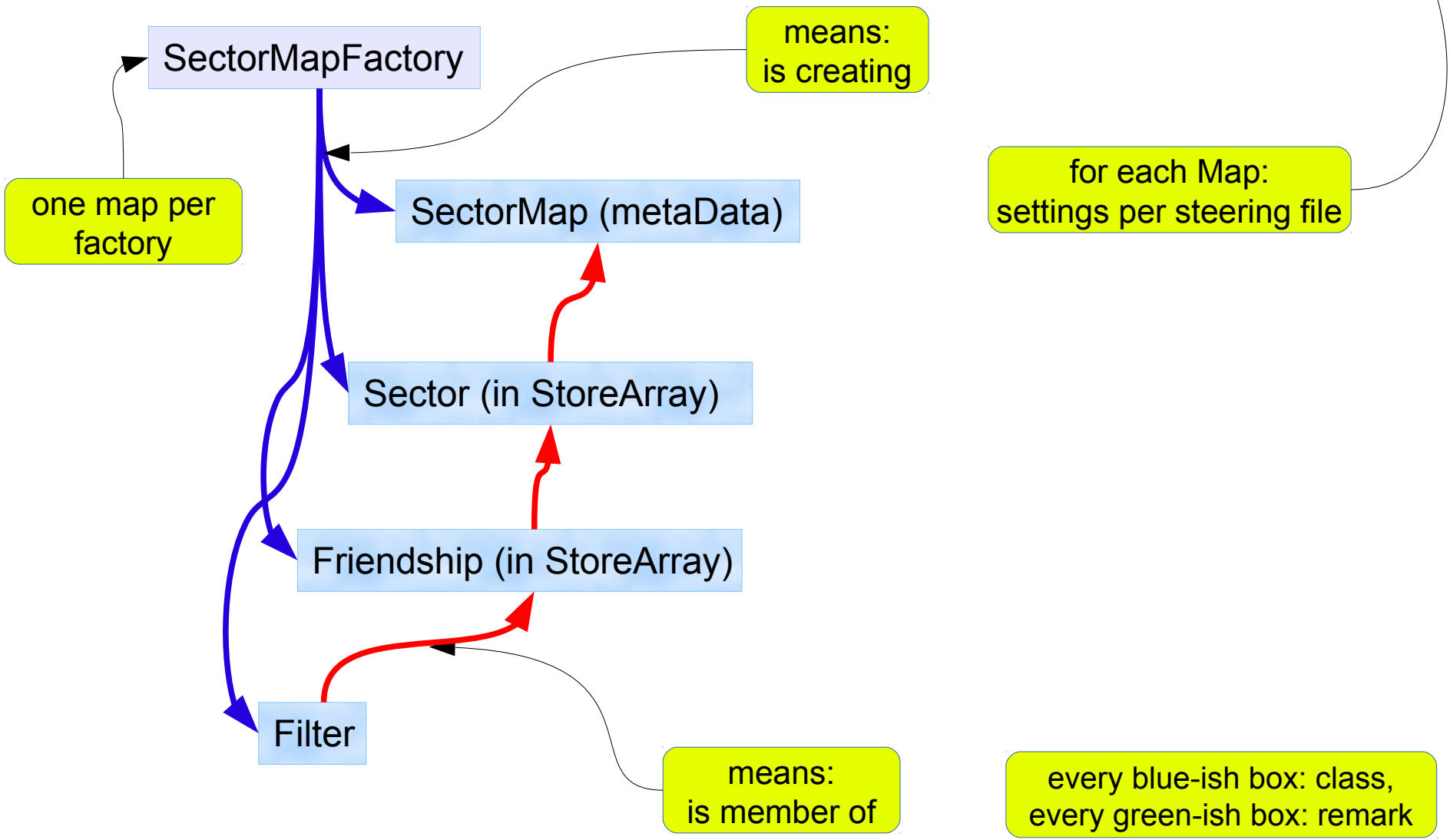


[!!!] : shortcut for „a lot of important stuff which is not part of this discussion“

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



Planned SectorMapCreatorModule (beginRun/endRun)



VXDTF redesign stuff to do (pending):

- RunWise:
  - SecMapCreator: some basic stuff is working, done: 25%
- EventWise:
  - SpacePointCreator: working for PXD, done 40%
  - CA-module: toDo: restructuring code of VXDTF, 0%
  - CKF: use genFit::processTrackPartially for CKF-implementation, 0%
  -DAF: genFit interface, 0%

Supervising Stefan Ferstl - project thesis

- Dealing with the issue of the VXDTF being a black box
- Covering everything and every step
- Output usable for detailed analysis or Display-module
- Analysis-output „Feature complete“ only bug-fixing left
- Display, awaiting comments from Christian Pulvermacher

Supervising Thomas Fabian – bachelor thesis

- Detailed study of filters used for the CA (e.g. Angle3D)
- Done in several steps
  - Check of every single filter implemented (tracking/tests)
  - Implementing improvements and three new filters
  - Using output provided by Stefan for analysis of „usefulness“ of each filter (doing correlations between filters, fake rate, …) - goal: reduce number of filters, chose best ones
- Currently starting with last point, analysis
- A very time-consuming supervising task

Suporting Thomas Madlener – bachelor thesis

- Study of using neuronal networks and a secMap for lowMomentum tracking (simplified matlab environment, but real Belle2-McData)
- Mostly supervised by Rudi, but I have to provide secMaps and mcData in their custom format, 95% done
- Kept me from working on previous topics but should be mostly done now

Outlook – May 2014

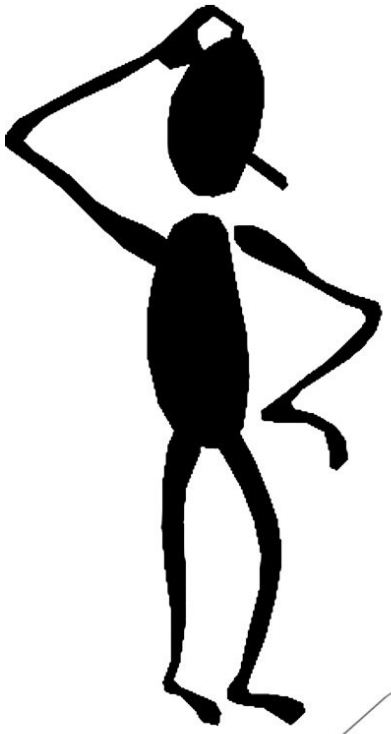
- F2F-Meeting, Pisa (now)
- DEPFET-Meeting, Seeon (Bavarian Metropolis – 25.-28.5.)
- Meantime: working on TB-stuff (see slides 3-5)
- Goal: not being the showstopper for Tobias' trip to the real time conference (26.-30.5.) in Nara, Japan

Outlook – June 2014

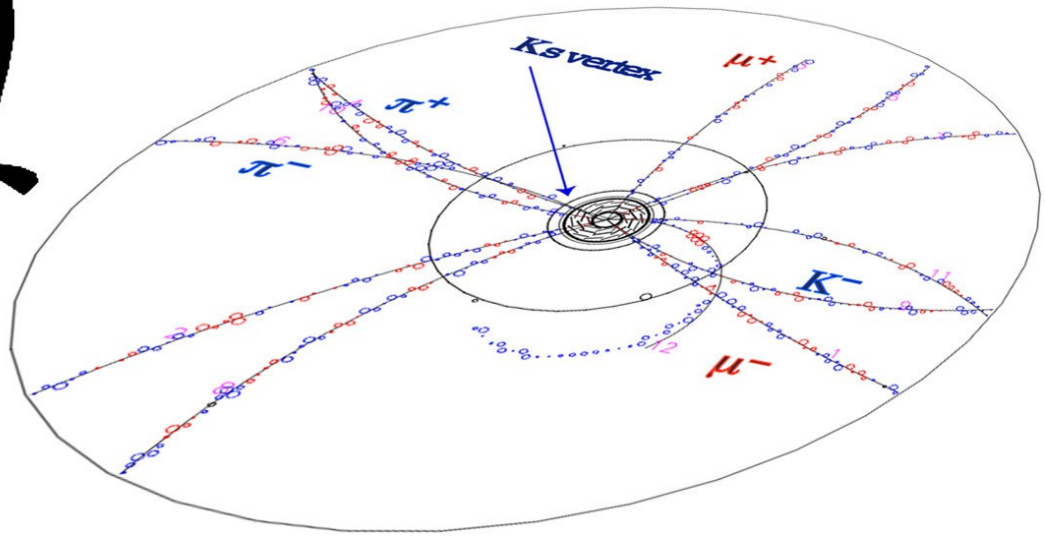
- Tidy up the code a bit
- Implement minor things of the redesign, write tests
- Prepare for:
- tCSC 2014, Split – 14.-22.6.
- Rest of the month – learning
- Exam in statistics – 30.6.

Outlook – Summer 2014

- July
  - Exam in classification and discriminant analysis – 7.7.
  - Holidays
- August, September: finishing redesign
- Autumn
  - Starting CKF and DAF implementation
  - TB preparation (?)



Thank you!



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

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