Some little studies

Idea

• On the basis, that probably the hot stuff happens where the CPU time is spend, I had a look into the timing.

basf2 -n 10000 createSimFile.py

```
42 # We might want to have particle gun(s) and EVTGen.
               43 # Still in that case the ParticleGun modules better come first:
               44 param pGun = {
Setup
               45
                      'pdgCodes': [13, -13],
                                                               # 13 = muon --> negatively charged!
                                                               # 20 tracks is a lot, but we don't use beam background in this sc
                      'nTracks': 20
                      'momentumGeneration': 'uniformPt',
               47
                      'momentumParams': [0.1, 0.15],
                                                               # 2 values: [min, max] in GeV
               48
                      'thetaGeneration': 'uniform'.
               49
               50
                      'thetaParams': [60., 85.],
                                                               # 2 values: [min, max] in degree
               51
                      'phiGeneration': 'uniform'.
               52
                      'phiParams': [0., 90.],
                                                               # [min, max] in degree
               53
                      'vertexGeneration': 'uniform'.
               54
                      'xVertexParams': [-0.1, 0.1],
                                                               # in cm...
               55
                      'yVertexParams': [-0.1, 0.1],
               56
                      'zVertexParams': [-0.5. 0.5].
               57 }
               58
               59 particlegun = register module('ParticleGun')
               60 particlegun.logging.log_level = LogLevel.WARNING
               61 particlegun.param(param pGun)
               62 main.add module(particlegun)
               63
               64 # Now we might want to add EvtGen:
               65 if False:
               66
                      evtgenInput = register module('EvtGenInput')
               67
                      evtgenInput.logging.log_level = LogLevel.WARNING
               68
                      main.add module(evtgenInput)
               69
               70 # Gearbox to access stuff from the data folders, and Geometry:
               71 gearbox = register_module('Gearbox')
               72 main.add module(gearbox)
               73
               74 geometry = register_module('Geometry')
               75 geometry.param('components', ['BeamPipe', 'MagneticFieldConstant4LimitedRSVD', # Important: look at B field!
               76
                                                'PXD', 'SVD'])
```

Name	Calls	1	VMemory(MB)	Time(s	Time(s)		Time(ms)/Call		
======================================	 10	1	 0	0.02	2	1.89	==== +-	0.32	
EventInfoPrinter	10	Ĩ.	0	0.00) j	0.00	+-	0.00	
Gearbox	10	Ì	0	0.00) j	0.00	+-	0.00	
Geometry	10	Ĩ.	0	0.00) İ	0.00	+-	0.00	
EventCounter	10	Ì	0	0.00) j	0.05	+-	0.09	
SectorMapBootstrap	10	Ĩ.	0	0.00) İ	0.00	+-	0.00	
SpacePointCreatorSVD	10	Ì	0	0.03	L j	1.18	+-	0.29	
SpacePointCreatorPXD	10		0	0.00) i	0.21	+-	0.09	
SpacePoint2TrueHitConnector	10	Ť	0	0.02	21	1.99	+-	0.81	
GFTC2SPTCConverter	10		0	0.02	2 İ	1.52	+-	0.45	
SPTCReferee	10	Ť	0	0.00) i	0.20	+-	0.07	
SpacePoint2TrueHitConnector	10		0	0.04	ŧİ	4.37	+-	0.65	
SecMapTrainerBase	10	Ì	Θİ	0.0	ιį	0.96	+-	0.27	
Total	10		0	0.13	3	12.70	+-	1.98	

basf2 -n 10 -i MyRootFile.root testVXDTFRelatedModules.py

basf2 -n 10 -i MyRootFile.root testVXDTFRelatedModules.py\										
Name	Calls	VMemory(MB)	Time(s)		ms)/Call					
RootInput EventInfoPrinter Gearbox Geometry EventCounter SectorMapBootstrap SpacePointCreatorSVD SpacePointCreatorPXD SpacePoint2TrueHitConnector GFTC2SPTCConverter SPTCReferee SpacePoint2TrueHitConnector RawSecMapMerger SegmentNetworkProducer IrackFinderVXDCettOMat SPTCvirtualIPRemover QualityEstimatorVXDRandom SPTCNetworkProducer TrackSetEvaluatorGreedy SVD0verlapChecker TrackSetEvaluatorGreedyDEV TrackFinderVXDAnalizer ====================================	10 10 10 10 10 10 10 10 10 10 10 10 10 1		0.02 0.00 0.00 0.00 0.00 0.01 0.01 0.00 0.02 0.01 0.00 0.05 0.00 0.05 0.00 0.05 0.000000		0.41 0.00 0.00 0.00 0.06 0.20 0.20 0.20 0.20	And bad finding efficiency, despite the same events are used for training! Segment Network Producer is slow, even when there are few friend relationships between sectors (as we have used only 10 events for the training), O(30) TrackCands				
	.===========			==============	========	1				

Note: There are 10,000 events in the file.

basf2 -i MyRootFile.root testVXDTFRelatedModules.py

Name	Calls	VMemory(MB)	Time(s)	Time(r	Time(ms)/Call		
RootInput	10001	 0	20.62	2.06 +-	15.31		
EventInfoPrinter	10000	0	0.02	0.00 +-	0.00		
Gearbox	10000	0	0.01	0.00 +-	0.00		
Geometry	10000	0	0.01	0.00 +-	0.00		
EventCounter	10000	0	0.29	0.03 +-	0.04		
SectorMapBootstrap	10000	0	0.02	0.00 +-	0.00		
SpacePointCreatorSVD	10000	0	13.04	1.30 +-	0.21		
SpacePointCreatorPXD	10000	0	1.79	0.18 +-	0.02		
SpacePoint2TrueHitConnector	10000	0	17.46	1.75 +-	0.21		
GFTC2SPTCConverter	10000	0	13.12	1.31 +-	0.14		
SPTCReferee	10000	0	1.86	0.19 +-	0.12		
SpacePoint2TrueHitConnector	10000	0	42.44	4.24 +-	0.67		
SecMapTrainerBase	10000	0	11.83	1.18 +-	1.17		
Total	10001	0	129.40	12.94 +-	15.43		

basf2 -n 10 -i MyRootFile.root testVXDTFRelatedModules.py\

4th event "breaks" my Computer;

basf2 -i MyRootFile.root -n 3 testVXDTFRelatedModules.py

Name Calls VMemory(MB) Time(s) Time(ms)/Call Segment Network Producer RootInput 3 0 0.01 1.92 +- 0.18 gets only factor 2 slower, if EventInfoPrinter 3 0 0.00 0.00 +- 0.00 there is a ton of friend Gearbox 3 0 0.00 0.00 +- 0.00 relations EventInfoPrinter 3 0 0.00 0.00 +- 0.00 relations EventCounter 3 0 0.00 0.00 +- 0.00 relations SectorMapBootstrap 3 0 0.00 0.26 +- 0.17 SectorMapBootstrap 3 0 0.00 1.33 +- 0.34 SpacePointCreatorSVD 3 0 0.00 1.33 +- 0.34 SpacePointZrueHitConnector 3 0 0.00 1.58 +- 0.17 SpacePointZTrueHitConnector 3 0 0.00 1.58 +- 0.17 SpacePointZTrueHitConnector 3 0 0.00 0.27 +- 0.80 SpacePointZTrueHitC	RootInput 3 0 0.01 1.92 - 0.18 gets only factor 2 slower, there is a ton of friend fiend 0.00 Gearbox 3 0 0.00 0.00 0.00 + 0.00 Geometry 3 0 0.00 0.00 + 0.00 relations EventCounter 3 0 0.00 0.00 + 0.00 relations SpacePointCreatorSVD 3 0 0.00 0.00 + 0.30 SVDOverlapChecker and SVDOverlapChecker and 0.00 SpacePointCreatorPXD 3 0 0.00 0.30 + 0.13 SVDOverlapChecker and SVDOverlapChecker and 0.00 SpacePoint2TrueHitConnector 3 0 0.00 0.30 + 0.13 SVDOverlapChecker and 0.00 SpacePoint2TrueHitConnector 3 0 0.00 0.27 + 0.80 0.00 0.27 + 0.80 0.00 0.13 Scale well with large numl 0.00 0.00 - 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00					
RootInput 3 0 0.01 1.92 +- 0.18 gets only factor 2 slower, if EventInfoPrinter 3 0 0.00 0.00 +- 0.00 there is a ton of friend Gearbox 3 0 0.00 0.00 +- 0.00 there is a ton of friend Geometry 3 0 0.00 0.00 +- 0.00 relations EventCounter 3 0 0.00 0.26 +- 0.17 SectorMapBootstrap 3 0 0.00 0.30 +- 0.13 SpacePointCreatorSVD 3 0 0.00 0.30 +- 0.13 SpacePointCreatorPXD 3 0 0.00 1.99 +- 0.26 TrackSetEvaluatorGreedyDEV GFTC2SPTCConverter 3 0 0.00 0.27 +- 0.08 cell with large number SpacePoint2TrueHitConnector 3 0 0.00 0.00 +- 0.00 (O(10,000)); SegmentNetworkProducer 3 0 0.00 0.00 +- 0.00 (O(10,000)); SegmentNetworkProducer 3 0 0.00 0.00 +- <	RootInput 3 0 0.01 1.92 +- 0.18 gets only factor 2 slower, there is a ton of friend field 0.00 Gearbox 3 0 0.00 0.00 +- 0.00 relations Geometry 3 0 0.00 0.00 +- 0.00 relations EventCounter 3 0 0.00 0.00 +- 0.00 relations SpacePointCreatorSVD 3 0 0.00 0.00 +- 0.00 relations SpacePointCreatorSVD 3 0 0.00 0.00 +- 0.00 relations SpacePointCreatorSVD 3 0 0.00 0.00 +- 0.00 New modules SpacePointCreatorSVD 3 0 0.00 1.33 +- 0.34 SVDOverlapChecker and SpacePoint2TrueHitConnector 3 0 0.00 1.58 +- 0.17 scale well with large number scale well with large number scale well with large number scale well with large number scale well with large number scale well with large number scale well with large number scale well with large number scale well with large number scale well with large number scale well with large number scale well with large number scale well with large number scale well with large number scale well with	Name	Calls	VMemory(MB)	Time(s)	Time(ms)/Call Segment Network Producer
TrackFindervxDAnalizer 3 0 0.01 2.42 +- 0.39	TrackFindervxDAnalizer 3 0 0.01 2.42 +- 0.39	RootInput EventInfoPrinter Gearbox Geometry EventCounter SectorMapBootstrap SpacePointCreatorSVD SpacePointCreatorPXD SpacePoint2TrueHitConnector GFTC2SPTCConverter SPTCReferee SpacePoint2TrueHitConnector RawSecMapMerger SegmentNetworkProducer TrackFinderVXDCellOMat SPTCvirtualIPRemover QualityEstimatorVXDRandom SPTCNetworkProducer TrackSetEvaluatorGreedy	3 3 3 3 3 3 3 3 3 3		0.01 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.01 0.00 0.01 0.00 0.01 0.00 0.01 0.00 0.01 0.00 0.00 49.25 96.15	1.92 + 0.18gets only factor 2 slower, if $0.00 + 0.00$ there is a ton of friend $0.00 + 0.00$ relations $0.26 + 0.17$ new modules $0.00 + 0.00$ New modules $1.33 + 0.34$ SVDOverlapChecker and $1.99 + 0.26$ TrackSetEvaluatorGreedyDEV $1.58 + 0.17$ scale well with large number $0.27 + 0.08$ of TrackCandidates $0.00 + 0.00$ (O(10,000)); $2559.42 + 260.50$ (O(10,000)); $0.58 + 0.26$ Let's see for Hopfield, but I'm $0.07 + 0.02$ entimistic. $16418.10 + 15926.02$ $32050.80 + 30802.89$
			3			
		=======================================				

isActive, Array Index: 11, QI: 0.825849 isActive, Array Index: 89, QI: 0.988963 isActive, Array Index: 2026, QI: 0.996657 isActive, Array Index: 4389, QI: 0.997194 isActive, Array Index: 4851, QI: 0.999532 isActive, Array Index: 6139, QI: 0.99964 isActive, Array Index: 6256, QI: 0.999548 isActive, Array Index: 6308, 0I: 0.773476 isActive, Array Index: 7341, QI: 0.958889 isActive, Array Index: 7389, QI: 0.995538 isActive, Array Index: 7447, QI: 0.999933 isActive, Array Index: 7499, QI: 0.473606 Array Index: 7447, QI: 0.999933 Array Index: 6139, QI: 0.99964 Array Index: 6256, QI: 0.999548 Array Index: 4851, QI: 0.999532 Array Index: 4389, QI: 0.997194 Array Index: 2026, QI: 0.996657 Array Index: 7389, QI: 0.995538 Array Index: 89, QI: 0.988963 Array Index: 7341, QI: 0.958889 Array Index: 11, QI: 0.825849 Array Index: 6308, QI: 0.773476 Array Index: 7499, QI: 0.473606

First set comes from Jakob's Greedy, second is the result of my Greedy of 3rd event.

They are the same (they differ with the fitter as QI estimator, as they act differently, if there are several candidates with the same QI value).

Conclusion of Study of differently trained secMaps

- Preliminary, possibly wrong: [Multi-Pass doesn't solve our problems, as even with sparse secMap, the SegmentNetworkProducer takes fairly long for a difficult event.]
- Timing strongly dependent on implementation of stuff → Stefano will not get a good intuition on the real timing with using the new trackFinding setup.

Look into SegmentNetworkProducerModule

- Found a swamp of code, that in some parts looks very optimized, but has as well tons of pitfalls regarding:
 - Branch misprediction ("If-statements"),
 - Lack of cache locality (Pointers or vectors of pointers, where vectors of the object itself could be),
 - debugging and supervising information, that isn't strictly needed, and
 - "Keep it save" checks on validity of data, where having non-valid data simply shouldn't occur (in this respect, the CDC code is as well not a good example).

Study the Timing of the event loop of SegmentNet...

Upt Opt	III 💙 DACK	* Forward * T Op * 70 Relative		o Parent V phorten lemplates Inscruction recti
Flat <u>P</u> ro	file		0 ×	× Belle2::SegmentNetworkProducerModule::event()
S <u>e</u> arch:			(No Grouping) 🗸	Types Callers All Callers Callee Map Source Code
Incl.	Self :	Called Function	Location	A # 1 Ir 1 Source (unknown)
	00.00 0.00 99.54 0.00 99.54 0.00 99.44 0.00 99.43 0.04 99.40 0.00	(0) ■ 0x000000000012d0 1 ■ 0x00000000040d3ac 1 ■ (below main) 1 ■ main 4 053 ■ <cycle 44=""> 1 ■ boost::python::exec_file(boo</cycle>	ld-2.19.so basf2 libc-2.19.so: libc-start.c basf2 libboost_python.so.1.61.0 libboost_python.so.1.61.0	1 There is no source available for the following function: 2 'Belle2::SegmentNetworkProducerModule::event()' 3 This is because no debug information is present. 4 Recompile source and redo the profile run. 5 The function is located in this ELF object: 6 'libvxdtFRedesign.so'
	99.40 0.00 99.40 0.00 98.58 0.00 98.58 0.00 98.58 0.00 98.58 0.00 64.66 0.00 64.60 0.00	 PyRun_FileExFlags boost::python::objects::caller process_overloads::non_void Belle2::Framework::process(b Belle2::EventProcessor::proc Belle2::EventProcessor::proc Belle2::EventProcessor::proc Belle2::EventProcessor::proc 	libpython3.5.so.1.0: pythonrun.c libframework_pybasf2.so libframework_pybasf2.so libframework.so libframework.so libframework.so libframework.so libframework.so	
	41.26 0.00 40.85 0.01 40.66 0.07 33.60 0.00 26.44 0.00	10 Belle2::SegmentNetworkPro 10 Belle2::SegmentNetworkPro 345 913 Belle2::StaticSector <belle2: 1 Belle2::EventProcessor::proc 1 Belle2::GeometryModule::init</belle2: 	libvxdtfRedesign.so libvxdtfRedesign.so libframework.so	
	26.44 0.00	1 📕 Belle2::geometry::Geometry	libgeometry.so	
	26.40 0.00 25.47 0.00 25.47 0.00 25.47 0.00 25.47 0.00 25.47 0.00 25.47 0.00 25.47 0.00 25.47 0.00	1 Belle2::geometry::Geometry 1 G4GeometryManager::CloseG 1 G4GeometryManager::BuildO 74 G4SmartVoxelHeader::G4Sma 74 G4SmartVoxelHeader::G4Sma 74 G4SmartVoxelHeader::BuildVo 74 G4SmartVoxelHeader::BuildVo	libG4geometry.so libG4geometry.so libG4geometry.so	In In per call Count Callee 40.85 9 707 539 10 Belle2::SegmentNetworkProducerModule::buildTrackNodeNetwork() (libxxdtFRedesign.so) 0.40 95 834 626 10 Belle2::SegmentNetworkProducerModule::buildSegmentNetwork() (libxxdtFRedesign.so) 0.01 2 145 11 Belle2::SegmentNetworkProducerModule::buildSegmentNetwork() (libxxdtFRedesign.so) 0.00 437 47 10 Belle2::SegmentNetworkProducerModule::buildActiveSectorNetwork(std::vector <selle2::segmentnetworkproducermodule::buildactivesectornetwork(std::vector<belle2::segmentnetworkproducermodule::buildactivesectornetwork(std::vector<belle2::segmentnetworkproducermodule::buildactivesectornetwork(std::vector<belle2::segmentnetworkproducermodule::buildactivesectornetwork(std::vector<belle2::segmentnetworkproducermodule::buildactivesectornetwork(std::vector<belle2::segmentnetworkproducermodule::buildactivesectornetwork(std::vector<belle2::segmentnetworkproducermodule::buildactivesectornetwork(std::vector<belle2::segmentnetworkproducermodule::buildactivesectornetwork(std::vector<belle2::segmentnetworkproducermodule::buildactivesectornetwork(std::vector<belle2::segmentnetworkproducermodule::buildactivesectornetwork(std::vector<belle2::segmentnetworkproducermodule::buildactivesectornetwork(std::vector<belle2::segmentnetworkproducermodule::buildactivesectornetwork(std::vector<belle2::segmentnetworkproducermodule::buildactivesectornetwork(std::vector<belle2::segmentnetworkproducermodule::buildactivesectornetwork(std::vector<belle2::segmentnetworkproducermodule::buildactivesectornetwork(std::vec< td=""></selle2::segmentnetworkproducermodule::buildactivesectornetwork(std::vector<belle2::segmentnetworkproducermodule::buildactivesectornetwork(std::vector<belle2::segmentnetworkproducermodule::buildactivesectornetwork(std::vector<belle2::segmentnetworkproducermodule::buildactivesectornetwork(std::vector<belle2::segmentnetworkproducermodule::buildactivesectornetwork(std::vector<belle2::segmentnetworkproducermodule::buildactivesectornetwork(std::vector<belle2::segmentnetworkproducermodule::buildactivesectornetwork(std::vector<belle2::segmentnetworkproducermodule::buildactivesectornetwork(std::vector<belle2::segmentnetworkproducermodule::buildactivesectornetwork(std::vector<belle2::segmentnetworkproducermodule::buildactivesectornetwork(std::vector<belle2::segmentnetworkproducermodule::buildactivesectornetwork(std::vector<belle2::segmentnetworkproducermodule::buildactivesectornetwork(std::vector<belle2::segmentnetworkproducermodule::buildactivesectornetwork(std::vector<belle2::segmentnetworkproducermodule::buildactivesectornetwork(std::vector<belle2::segmentnetworkproducermodule::buildactivesectornetwork(std::vec<>

Belle2::	SegmentNe	tworkProdu	ucerModule::	buildTrackNodeNet	work()							
Types	Callers	All Callers	Callee Map	S <u>o</u> urce Code								
# ilr	Sourc	ce (unknown)	i)									
1 2 3 4 5 6	'Bel This is Recor The fu	lle2::Segmer s because no mpile source	ntNetworkPro o debug inforr e and redo the ocated in this I	mation is present. e profile run.	tion: ITrackNodeNetwork	<o'< th=""><th></th><th></th><th></th><th></th><th></th><th></th></o'<>						
											300	
Ir	Irperca	Il Count	Callee									
40.6 0.1 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0	66 279 33 13 4 11 03 9 00 01 29 01 29 01 29 00 30 00 30 00 36 00 36 00 36 00 36 00 36 00 36 00 36 00 36	32 345 913 15 76 233 07 6 778 92 76 233 23 76 233 90 152 466 38 78 714 30 2 471 60 2 471 4 78 714 27 2 471	Belle2::Sta std::revers Belle2::Dird Belle2::Dird Belle2::Dird Belle2::Log Belle2::Log Belle2::Log Belle2::Log void std::ve	se_iterator <gnu_c rectedNodeNetwork rectedNode<belle2: c:normal_iterator gSystem::isLevelEnat rectedNodeNetwork c:normal_iterator- gSystem::Instance() ector<unsigned int,<="" td=""><td>pacePoint, Belle2::Fi cxx::normal_iterat k<belle2::tracknode, ::TrackNode, Belle2::\ ::TrackNode, Belle2::\ <belle2::directedno bled(Belle2::LogConf k<belle2::tracknode, <unsigned int*,="" std::<br="">(libframework.so) , std::allocator<unsig > (ld-2.19.so: dl-tram</unsig </unsigned></belle2::tracknode, </belle2::directedno </belle2::tracknode, </td><td>tor<belle2::dired e, Belle2::VoidMe :VoidMetaInFo>:: :VoidMetaInFo>:: ode<belle2::trac ifig::ELogLevel, in e, Belle2::VoidMe ::vector<unsigne< td=""><td>ctedNode<belle tainfo>::linkThe addOuterNode(addinnerNode() :kNode, Belle2:: ht, char const*) :tainfo>::addNo ed int, std::alloc</belle </td><td>e2::TrackNode eseEntries(Be (Belle2::Direct Belle2::Direct :VoidMetaInfo) const (libfra :de(Belle2::Tra :ator<unsigne< td=""><th>e, Belle2::Void elle2::TrackNo tedNode<bell >**, std::vect mework.so) ickNode&) (lib icd int> > > std</bell </th><th>MetaInfo>**, s de&, Belle2::Tr le2::TrackNode e2::TrackNode or<belle2::dir vxdtfRedesign ::find<gnu_c< th=""><th>std::vector<belle: rackNode&) (libvxo e, Belle2::VoidMet ;, Belle2::VoidMet ectedNode<belle n.so)</belle </belle: </th><td>2::Direct dtfRedesi taInfo>&) aInfo>&) (e2::TrackN</td></gnu_c<></belle2::dir </th></unsigne<></td></unsigne<></belle2::trac </belle2::dired </td></unsigned></belle2: </gnu_c 	pacePoint, Belle2::Fi cxx::normal_iterat k <belle2::tracknode, ::TrackNode, Belle2::\ ::TrackNode, Belle2::\ <belle2::directedno bled(Belle2::LogConf k<belle2::tracknode, <unsigned int*,="" std::<br="">(libframework.so) , std::allocator<unsig > (ld-2.19.so: dl-tram</unsig </unsigned></belle2::tracknode, </belle2::directedno </belle2::tracknode, 	tor <belle2::dired e, Belle2::VoidMe :VoidMetaInFo>:: :VoidMetaInFo>:: ode<belle2::trac ifig::ELogLevel, in e, Belle2::VoidMe ::vector<unsigne< td=""><td>ctedNode<belle tainfo>::linkThe addOuterNode(addinnerNode() :kNode, Belle2:: ht, char const*) :tainfo>::addNo ed int, std::alloc</belle </td><td>e2::TrackNode eseEntries(Be (Belle2::Direct Belle2::Direct :VoidMetaInfo) const (libfra :de(Belle2::Tra :ator<unsigne< td=""><th>e, Belle2::Void elle2::TrackNo tedNode<bell >**, std::vect mework.so) ickNode&) (lib icd int> > > std</bell </th><th>MetaInfo>**, s de&, Belle2::Tr le2::TrackNode e2::TrackNode or<belle2::dir vxdtfRedesign ::find<gnu_c< th=""><th>std::vector<belle: rackNode&) (libvxo e, Belle2::VoidMet ;, Belle2::VoidMet ectedNode<belle n.so)</belle </belle: </th><td>2::Direct dtfRedesi taInfo>&) aInfo>&) (e2::TrackN</td></gnu_c<></belle2::dir </th></unsigne<></td></unsigne<></belle2::trac </belle2::dired 	ctedNode <belle tainfo>::linkThe addOuterNode(addinnerNode() :kNode, Belle2:: ht, char const*) :tainfo>::addNo ed int, std::alloc</belle 	e2::TrackNode eseEntries(Be (Belle2::Direct Belle2::Direct :VoidMetaInfo) const (libfra :de(Belle2::Tra :ator <unsigne< td=""><th>e, Belle2::Void elle2::TrackNo tedNode<bell >**, std::vect mework.so) ickNode&) (lib icd int> > > std</bell </th><th>MetaInfo>**, s de&, Belle2::Tr le2::TrackNode e2::TrackNode or<belle2::dir vxdtfRedesign ::find<gnu_c< th=""><th>std::vector<belle: rackNode&) (libvxo e, Belle2::VoidMet ;, Belle2::VoidMet ectedNode<belle n.so)</belle </belle: </th><td>2::Direct dtfRedesi taInfo>&) aInfo>&) (e2::TrackN</td></gnu_c<></belle2::dir </th></unsigne<>	e, Belle2::Void elle2::TrackNo tedNode <bell >**, std::vect mework.so) ickNode&) (lib icd int> > > std</bell 	MetaInfo>**, s de&, Belle2::Tr le2::TrackNode e2::TrackNode or <belle2::dir vxdtfRedesign ::find<gnu_c< th=""><th>std::vector<belle: rackNode&) (libvxo e, Belle2::VoidMet ;, Belle2::VoidMet ectedNode<belle n.so)</belle </belle: </th><td>2::Direct dtfRedesi taInfo>&) aInfo>&) (e2::TrackN</td></gnu_c<></belle2::dir 	std::vector <belle: rackNode&) (libvxo e, Belle2::VoidMet ;, Belle2::VoidMet ectedNode<belle n.so)</belle </belle: 	2::Direct dtfRedesi taInfo>&) aInfo>&) (e2::TrackN

Belle2::StaticSector<Belle2::SpacePoint, Belle2::Filter<Belle2::OperatorAnd, Belle2::Fi...t(Belle2::FullSecID const&, Belle2::SpacePoint const&, Belle2::SpacePoint const&) const

Types C	Callers All Callers Callee Map Source Code		
# ∃lr	ESource (unknown)		
1 2 3 4 5 6	There is no source available for the following function: 'Belle2::StaticSector <belle2::spacepoint, belle2::filter<belle:<br="">This is because no debug information is present. Recompile source and redo the profile run. The function is located in this ELF object: 'libvxdtfRedesign.so'</belle2::spacepoint,>	2::OperatorAnd, Belle2::Filter <belle2::operatorand, belle2::filter<belle<="" th=""><th>2::OperatorAnd, Belle2::Filter<bell< th=""></bell<></th></belle2::operatorand,>	2::OperatorAnd, Belle2::Filter <bell< th=""></bell<>
lr i Ir	Ir per call : Count : Callee		
22.34 9.51 8.43 0.15 0.04 0.03	65 257 813 545 Belle2::SelectionVariable <belle2::spacepo 66 019 342 318 Trree::Fill() (libTree.so) 58 492 342 318 Belle2::createPurityInFosVec(std::vector< 217 1 627 090 std::map<std::string, bool*,="" std::less<std:<br="">200 471 227 std::map<std::string, double*,="" std::less<st<="" td=""><td>oint, double>::name() const (libvxdtfRedesign.so) <belle2::spacepoint const*="" const*,="" std::allocator<belle2::spacepoint=""> : ::string>, std::allocator<std::pair<std::string bool*="" const,=""> > >::at(std:: :td::string>, std::allocator<std::pair<std::string const,="" double*=""> > >::at(:or<char> const&) [clone.part.25] (libvxdtfRedesign.so)</char></std::pair<std::string></std::pair<std::string></belle2::spacepoint></td><td>string const&) (libvxdtfRedesign.so)</td></std::string,></std::string,></belle2::spacepo 	oint, double>::name() const (libvxdtfRedesign.so) <belle2::spacepoint const*="" const*,="" std::allocator<belle2::spacepoint=""> : ::string>, std::allocator<std::pair<std::string bool*="" const,=""> > >::at(std:: :td::string>, std::allocator<std::pair<std::string const,="" double*=""> > >::at(:or<char> const&) [clone.part.25] (libvxdtfRedesign.so)</char></std::pair<std::string></std::pair<std::string></belle2::spacepoint>	string const&) (libvxdtfRedesign.so)

OK, Problem is Observer setup

- Jakob told, me the problem are the Observers...
 - Next step: using VoidObservers and test the timing again.

```
diff --git a/tracking/modules/VXDTFHelperTools/src/RawSecMapMergerModule.cc b/tracking/modules/VXDTFHelperTools/src/RawSecMapMergerModule.cc
index 40b842a..0046637 100644
--- a/tracking/modules/VXDTFHelperTools/src/RawSecMapMergerModule.cc
+++ b/tracking/modules/VXDTFHelperTools/src/RawSecMapMergerModule.cc
@@ -457,18 +457,19 @@ template <class FilterType> void RawSecMapMergerModule::add2HitFilters(VXDTFFilt
   /// JKL Feb 2016 - big working example:
      VXDTFFilters<SpacePoint>::twoHitFilter t friendSectorsSegmentFilter =
-//
-//
          (filterCutsMap.at("Distance3DSquared").getMin() <= Distance3DSquared<SpacePoint>() <=</pre>
-11
-11
           filterCutsMap.at("Distance3DSquared").getMax()).observe(Void0bserver()) &&
-//
          (filterCutsMap.at("Distance2DXYSquared").getMin() <= Distance2DXYSquared<SpacePoint>() <=</pre>
           filterCutsMap.at("Distance2DXYSguared").getMax()).observe(VoidObserver()) &&
-//
          (filterCutsMap.at("Distance1DZ").getMin() <= Distance1DZ<SpacePoint>() <= filterCutsMap.at("Distance1DZ").getMax()).observe(</pre>
-11
-11
            VoidObserver()) &&
-11
          (filterCutsMap.at("SlopeRZ").getMin() <= SlopeRZ<SpacePoint>() <= filterCutsMap.at("SlopeRZ").getMax()).observe(VoidObserver()) &&</pre>
-//
          (filterCutsMap.at("Distance3DNormed").getMin() <= Distance3DNormed<SpacePoint>() <=</pre>
-11
           filterCutsMap.at("Distance3DNormed").getMax()).observe(Void0bserver())
-11
        );
   VXDTFFilters<SpacePoint>::twoHitFilter t friendSectorsSegmentFilter =
+
        (filterCutsMap.at("Distance3DSquared").getMin() <= Distance3DSquared<SpacePoint>() <=</pre>
         filterCutsMap.at("Distance3DSquared").getMax()).observe(VoidObserver()) &&
        (filterCutsMap.at("Distance2DXYSguared").getMin() <= Distance2DXYSguared<SpacePoint>() <=</pre>
         filterCutsMap.at("Distance2DXYSquared").getMax()).observe(VoidObserver()) &&
        (filterCutsMap.at("Distance1DZ").getMin() <= Distance1DZ<SpacePoint>() <= filterCutsMap.at("Distance1DZ").getMax()).observe(</pre>
          VoidObserver()) &&
        (filterCutsMap.at("SlopeRZ").getMin() <= SlopeRZ<SpacePoint>() <= filterCutsMap.at("SlopeRZ").getMax()).observe(VoidObserver()) &&</pre>
        (filterCutsMap.at("Distance3DNormed").getMin() <= Distance3DNormed<SpacePoint>() <=</pre>
         filterCutsMap.at("Distance3DNormed").getMax()).observe(VoidObserver())
      );
  VXDTFFilters<SpacePoint>::twoHitFilter t friendSectorsSegmentFilter =
   -484,6 +485,7 @@ template <class FilterType> void RawSecMapMergerModule::add2HitFilters(VXDTFFilt
          filterCutsMap.at("Distance3DNormed").getMax()).observe(ObserverCheckMCPurity())
       ).observe(ObserverCheckMCPurity())
     );
```

```
diff --git a/tracking/trackFindingVXD/environment/include/VXDTFFilters.h b/tracking/trackFindingVXD/environment/include/VXDTFFilters.h
index 2306387..11058f4 100644
--- a/tracking/trackFindingVXD/environment/include/VXDTFFilters.h
+++ b/tracking/trackFindingVXD/environment/include/VXDTFFilters.h
@@ -65,7 +65,7 @@ namespace Belle2 {
    /// minimal working 2-hits-example used for redesign of VXDTF.
       typedef decltype((0. <= Distance3DSquared<Belle2::SpacePoint>() <= 0.).observe(ObserverPrintResults())) twoHitFilter t;</pre>
11
+/***
    /// big working 2-hits-example used for redesign of VXDTF.
    typedef decltype(
@@ -76,15 +76,16 @@ namespace Belle2 {
        (0. <= Distance3DNormed<Belle2::SpacePoint>() <= 0.).observe(ObserverCheckMCPurity())</pre>
      ).observe(ObserverCheckMCPurity())
    ) twoHitFilter t;
                               ****
    // March9th2016: TODO: we want to use a big observer observing everything - Working title: Mega0bserver.
-// typedef decltype(
-// ((0. <= Distance3DSquared<Belle2::SpacePoint>() <= 0.).observe(Void0bserver())&&</pre>
-// (0. <= Distance2DXYSguared<Belle2::SpacePoint>() <= 0.).observe(Void0bserver())&&</pre>
-11
    (0. <= Distance1DZ<Belle2::SpacePoint>() <= 0.).observe(Void0bserver())&&</pre>
    (0. <= SlopeRZ<Belle2::SpacePoint>() <= 0.).observe(Void0bserver())&&</pre>
-//
      (0. <= Distance3DNormed<Belle2::SpacePoint>() <= 0.).observe(Void0bserver())).observe(Void0bserver())</pre>
-//
-// ) twoHitFilter t:
+ typedef decltype(
+ ((0. <= Distance3DSquared<Belle2::SpacePoint>() <= 0.).observe(Void0bserver())&&</p>
  (0. <= Distance2DXYSguared<Belle2::SpacePoint>() <= 0.).observe(VoidObserver())&&</pre>
+
+ (0. <= Distance1DZ<Belle2::SpacePoint>() <= 0.).observe(VoidObserver())&&</p>
+ (0. <= SlopeRZ<Belle2::SpacePoint>() <= 0.).observe(VoidObserver())&&</p>
+ (0. <= Distance3DNormed<Belle2::SpacePoint>() <= 0.).observe(Void0bserver())).observe(Void0bserver())</p>
+ ) twoHitFilter t:
```

```
/// minimal working example for 3-hits:
// typedef decltype((0. <= Angle3DSimple<point_t>() <= 0.).observe(0bserver3HitPrintResults())) threeHitFilter_t;</pre>
```

Name	Calls	VMemory(MB)	Time(s)	Time(ms)/Call
RootInput	10	 0	0.02	2.05 +-	0.38
EventInfoPrinter	10	0	0.00	0.00 +-	0.00
Gearbox	10	0	0.00	0.00 +-	0.00
Geometry	10	0	0.00	0.00 +-	0.00
EventCounter	10	0	0.00	0.13 +-	0.05 V
SectorMapBootstrap	10	0	0.00	0.00 +-	0.00 5
SpacePointCreatorSVD	10	0	0.01	1.04 +-	0.14 N
SpacePointCreatorPXD	10	0	0.00	0.19 +-	0.03 re
SpacePoint2TrueHitConnector	10	0	0.02	1.84 +-	0.21
GFTC2SPTCConverter	10	0	0.01	1.40 +-	0.18
SPTCReferee	10	0	0.00	0.22 +-	0.05
SpacePoint2TrueHitConnector	10	0	0.05	5.32 +-	0.89
RawSecMapMerger	10	0	0.00	0.00 +-	0.00
SegmentNetworkProducer	10	0	0.04	4.02 +-	1.75
TrackFinderVXDCell0Mat	10	0	0.03	2.67 +-	7.06
SPTCvirtualIPRemover	10	0	0.00	0.01 +-	0.00
QualityEstimatorVXDRandom	10	0	0.00	0.00 +-	0.00
SPTCNetworkProducer	10	0	0.00	0.08 +-	0.07
TrackSetEvaluatorGreedy	10	0	0.00	0.10 +-	0.10
SVD0verlapChecker	10	0	0.00	0.38 +-	0.04
TrackSetEvaluatorGreedyDEV	10	0	0.00	0.17 +-	0.04
TrackFinderVXDAnalizer	10	Οİ	0.03	2.80 +-	0.89
	10	 0	0.23	22.91 +-	7.76

Now using the fat SecMap again

basf2 -i MyRootFile.root testVXDTFRelatedModules.py

======================================				
Name	Calls	VMemory(MB)	Time(s)	Time(ms)/Call
RootInput	3	0	0.01	1.94 +- 0.20
EventInfoPrinter	3	0	0.00	0.00 +- 0.00
Gearbox	3	0	0.00	0.00 +- 0.00
Geometry	3	0	0.00	0.00 +- 0.00
EventCounter	3	0	0.00	0.19 +- 0.07
SectorMapBootstrap	3	0	0.00	0.00 +- 0.00
SpacePointCreatorSVD	3	0	0.00	0.97 +- 0.08
SpacePointCreatorPXD	3	0	0.00	0.20 +- 0.03
SpacePoint2TrueHitConnector	3	0	0.01	1.81 +- 0.18
GFTC2SPTCConverter	3	0	0.00	1.45 +- 0.27
SPTCReferee	3	0	0.00	0.25 +- 0.09
SpacePoint2TrueHitConnector	3	0	0.01	4.69 +- 0.35
RawSecMapMerger	3	0	0.00	0.00 +- 0.00
SegmentNetworkProducer	3	0	0.28	92.83 +- 53.59
TrackFinderVXDCell0Mat	3	0	0.12	40.29 +- 16.80
SPTCvirtualIPRemover	3	0	0.00	0.55 +- 0.24
QualityEstimatorVXDRandom	3	0	0.00	0.07 +- 0.02
SPTCNetworkProducer	3	0	47.30	15765.77 +-15135.49
TrackSetEvaluatorGreedy	3	0	93.29	31095.64 +-30119.97
SVD0verlapChecker	3	0	3.36	1120.39 +- 839.42
TrackSetEvaluatorGreedyDEV	3	0	0.10	32.29 +- 21.52
TrackFinderVXDAnalizer	3	0	0.01	2.61 +- 0.42
Total	3	0	144.53	48175.11 +-46135.34

OK, now we are back, that the OverlapChecker is the "problem". Or not, as this amount of TrackCandidates is rare, so we know, the redesign is fast enogh!

isActive, Array Index: 11, QI: 0.825849 isActive, Array Index: 89, QI: 0.988963 isActive, Array Index: 2026, QI: 0.996657 isActive, Array Index: 4389, QI: 0.997194 lsActive, Array Index: 4851, QI: 0.999532 IsActive, Array Index: 6139, QI: 0.99964 lsActive, Array Index: 6256, QI: 0.999548 isActive, Array Index: 6308, QI: 0.773476 isActive, Array Index: 7341, QI: 0.958889 IsActive, Array Index: 7389, QI: 0.995538 isActive, Array Index: 7447, QI: 0.999933 isActive, Array Index: 7499, QI: 0.473606 Array Index: 7447, QI: 0.999933 Array Index: 6139, QI: 0.99964 Array Index: 6256, QI: 0.999548 Array Index: 4851, QI: 0.999532 Array Index: 4389, QI: 0.997194 Array Index: 2026, QI: 0.996657 Array Index: 7389, QI: 0.995538 Array Index: 89, QI: 0.988963 Array Index: 7341, QI: 0.958889 Array Index: 11, QI: 0.825849 Array Index: 6308, QI: 0.773476 Array Index: 7499, QI: 0.473606



Iг		Inpericall	Count	Callee
	49.51	1 379 986 138	78	Belle2::EventProcessor::callEvent(Belle2::Module*) (libframework.so)
	0.00	473 137	1	Belle2::EventProcessor::processBeginRun(bool) (libframework.so)
	0.00	9 884	4	Belle2::RandomNumbers::initializeEvent() (libframework.so)
	0.00	38 4 4 2	1	cxa throw (libstdc++.so.6.0.21)
	0.00	306	78	Belle2::PathIterator::descendifNecessary() (libframework.so)
	0.00	196	79	dynamic_cast (libstdc++.so.6.0.21)
	0.00	1 378	10	dl runtime resolve <cycle 1=""> (ld-2.19.so: dl-trampoline.5)</cycle>
	0.00	7 228	1	Belle2::EventProcessor::StoppedBySignalException::StoppedBySignalException(int) (libframework.so)
	0.00	1 7 3 7	2	Belle2::ProcessStatistics::setCounters(double&, double&, double, double) (libframework.so)
	0.00	21	78	Belle2::Module::evalCondition() const (libframework.so)
	0.00	11	56	Belle2::EventMetaData::operator==(Belle2::EventMetaData const&) const (libframework.so)
	0.00	5	79	Belle2::EventMetaData::isEndOfData() const (libframework.so)
	0.00	50	4	Belle2::DBStore::updateEvent() (libframework.so)
	0.00	129	1	cxa_allocate_exception (libstdc++.so.6.0.21)
	0.00	16	4	std:/string:/assign(std:/string.const&) (libstdc++.so.6.0.21)
	0.00	12	4	TObject::operator=(TObject const&) (libCore.so)
	0.00	5	4	Belle2::RandomNumbers::useEventDependent() (libframework.so)
	0.00	4	4	Belle2::Environment::Instance() (libframework.so)
	0.00	4	4	Belle2::DBStore::Instance() (libFramework.so)
	0.00	3	1	Belle2::EventProcessor::processEndRun() (libframework.so)

Alternative Conclusion

- Pausing work for that long on a badly documented complicated piece of code, is not a great idea.
- Speed is probably a non-issue of the VXD Track finder.
 - $\circ \quad \rightarrow \text{Focus on efficiency}.$
 - Look into realistic situation, if then SVDOverlapChecker is still the problem, we can further think about not sorting...



basf2 -i MyRootFile.root -n 1000 testVXDTFRelatedModules.py

basf2 -i MyRootFile.root -n 10 testVXDTFRelatedModules.py

=======================================				===================
Name	Calls	VMemory(MB)	Time(s)	Time(ms)/Call
RootInput	10	0	0.02	2.00 +- 0.42
EventInfoPrinter	10	0	0.00	0.00 +- 0.00
Gearbox	10	0	0.00	0.00 +- 0.00
Geometry	10	0	0.00	0.00 +- 0.00
EventCounter	10	0	0.00	0.14 +- 0.04
SectorMapBootstrap	10	0	0.00	0.00 +- 0.00
SpacePointCreatorSVD	10	0	0.01	1.05 +- 0.14
SpacePointCreatorPXD	10	0	0.00	0.19 +- 0.03
SpacePoint2TrueHitConnector	10	0	0.02	1.77 +- 0.17
GFTC2SPTCConverter	10	0	0.01	1.36 +- 0.18
SPTCReferee	10	0	0.00	0.21 +- 0.05
SpacePoint2TrueHitConnector	10	0	0.05	5.02 +- 0.87
RawSecMapMerger	10	0	0.00	0.00 +- 0.00
SegmentNetworkProducer	10	0	21.26	2125.60 +- 678.76
TrackFinderVXDCell0Mat	10	0	0.07	7.40 +- 5.52
SPTCvirtualIPRemover	10	0	0.00	0.07 +- 0.06
QualityEstimatorVXDRandom	10	0	0.00	0.02 +- 0.01
SPTCNetworkProducer	10	0	2.20	220.46 +- 449.80
TrackSetEvaluatorGreedy	10	0	3.74	374.24 +- 785.13
SVD0verlapChecker	10	0	0.46	45.59 +- 77.04
TrackSetEvaluatorGreedyDEV	10	0	0.01	1.42 +- 1.89
TrackFinderVXDAnalizer	10	0	0.02	1.96 +- 0.32
Total	10	0	27.90	2790.22 +-1541.76
	.==============			

With circle Fitter instead of Random

						we should spe
Name	Calls	VMemory(MB)	Time(s)	Time(ms)/Call	
RootInput	10	0	0.02	2.11 +-	0.56	estimation!
EventInfoPrinter	10	0	0.00	0.00 +-	0.00	
Gearbox	10	0	0.00	0.00 +-	0.00	
Geometry	10	0	0.00	0.00 +-	0.00	
EventCounter	10	Θ	0.00	0.14 +-	0.03	
SectorMapBootstrap	10	0	0.00	0.00 +-	0.00	
SpacePointCreatorSVD	10	0	0.01	1.07 +-	0.16	
SpacePointCreatorPXD	10	0	0.00	0.19 +-	0.03	
SpacePoint2TrueHitConnector	10	Θ	0.02	1.78 +-	0.20	
GFTC2SPTCConverter	10	0	0.01	1.35 +-	0.18	
SPTCReferee	10	Θ	0.00	0.20 +-	0.06	
SpacePoint2TrueHitConnector	10	0	0.05	5.02 +-	0.85	
RawSecMapMerger	10	0	0.00	0.00 +-	0.00	
SegmentNetworkProducer	10	0	21.46	2146.11 +-	694.81	
TrackFinderVXDCell0Mat	10	0	0.07	7.40 +-	5.49	
SPTCvirtualIPRemover	10	0	0.00	0.08 +-	0.09	
QualityEstimatorVXDCircleFit	10	0	0.07	6.70 +-	14.67	
SPICNELWOIKPIODUCEI	10	U	2.18	[kg = NG 방법 방법 방법 NG NG = (1.11)		
TrackSetEvaluatorGreedy	10	0	3.78	377.67 +-	795.30	
SVD0verlapChecker	10	0	0.45			
TrackSetEvaluatorGreedyDEV	10	0	0.01	1.38 +-	1.85	
TrackFinderVXDAnalizer	10	0	0.02	2.02 +-	0.24	
Total	10	0	28.18	2818.15 +-	1543.33	

We should spend more time on the quality estimation!