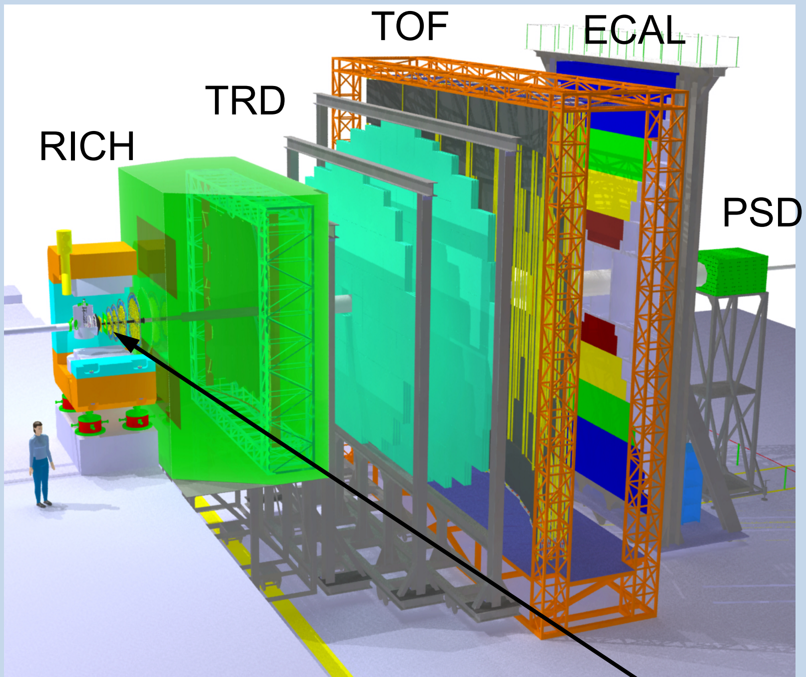
Challenges and Requirements For Online Data Processing in CBM

Volker Friese
GSI Darmstadt

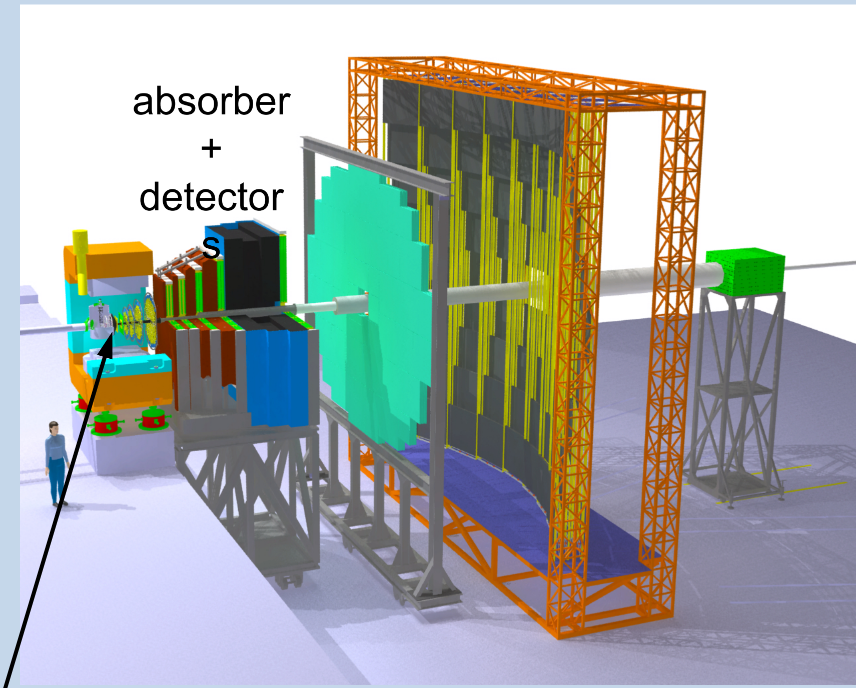
Workshop on Fair Event Reconstruction
GSI, 29 October 2012

CBM: experimental setup

Electron + Hadron setup

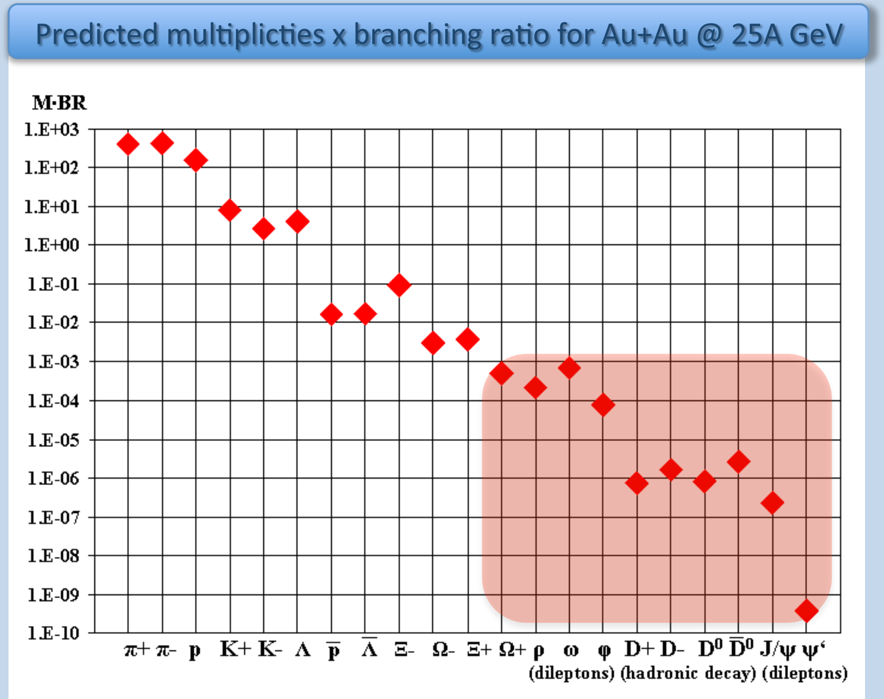


Muon setup



STS+MVD

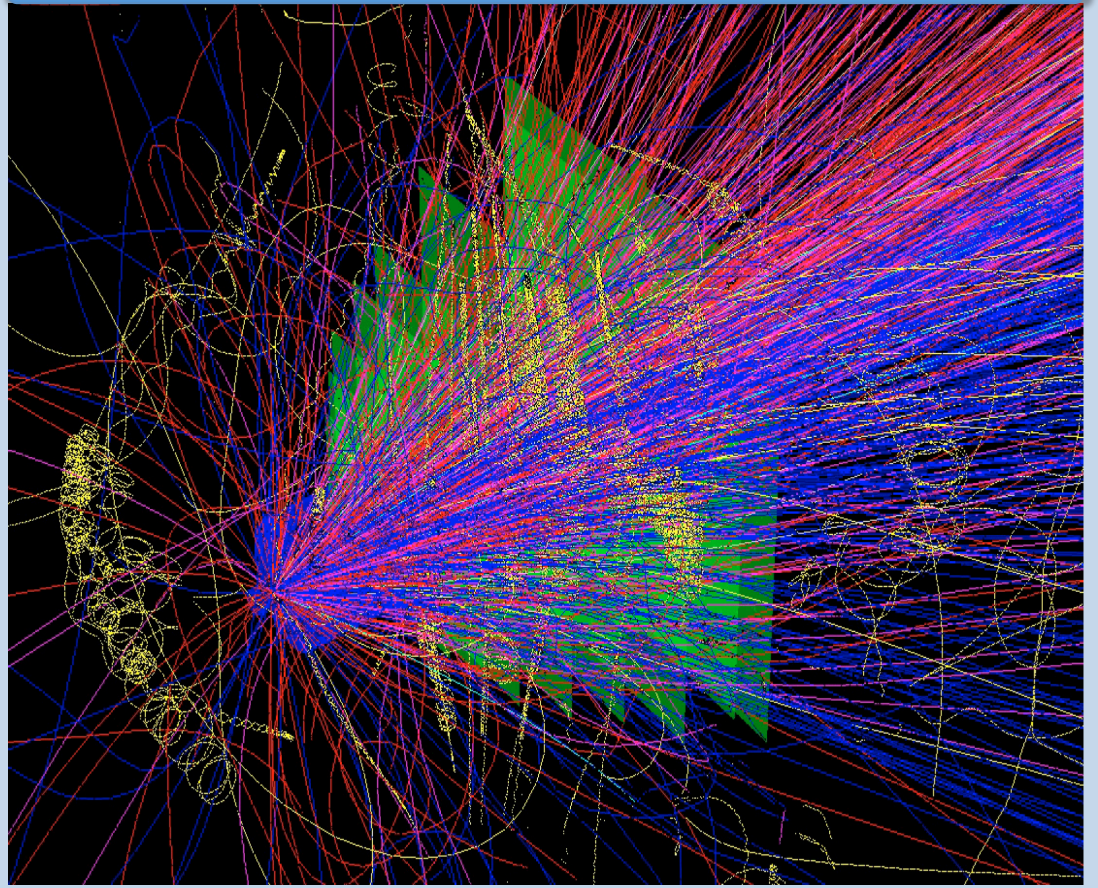
Observables



- The measurement of extremely rare probes requires very high interaction rates – up to 10 MHz
- Such rates drive the requirements on detectors, FEE and online computing

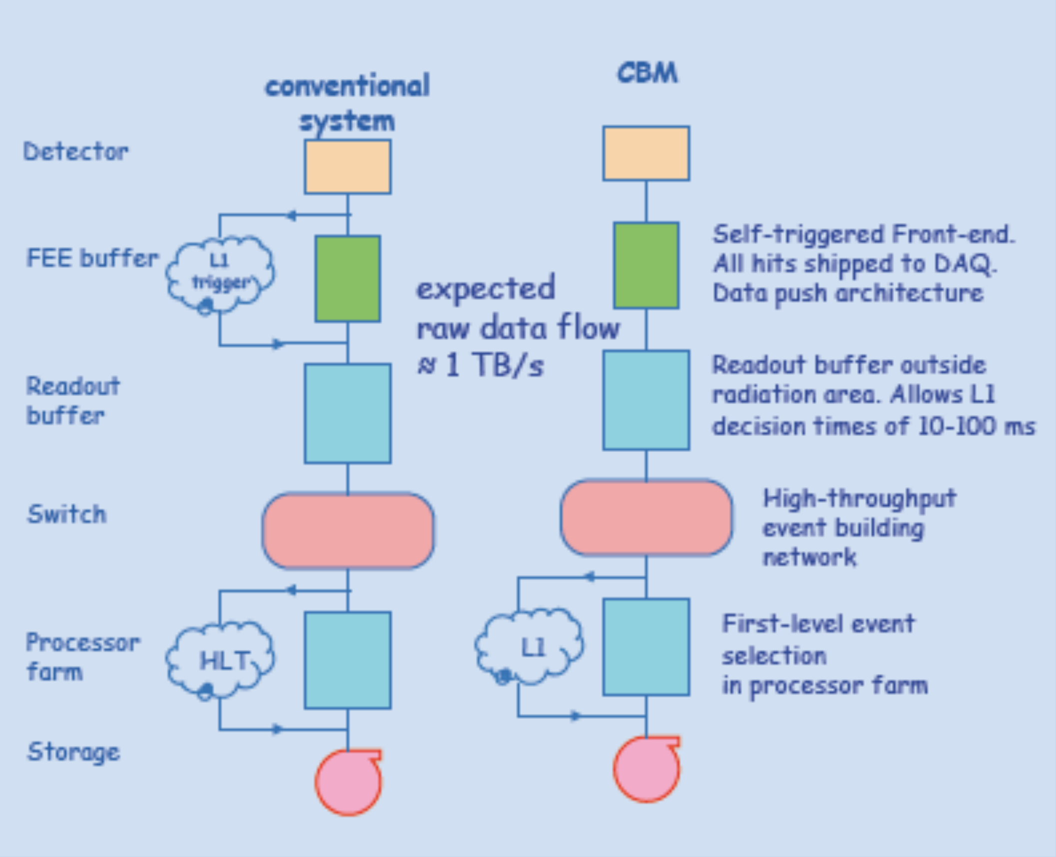
The challenge

Central Au+Au collision at 25A GeV., simulated in the CBM detector



- typical CBM event: about 700 charged tracks in the acceptance
- strong kinematical focusing in the fixed-target setup: high track densities
- up to 10^7 of such events per second
- to be reconstructed precisely, fast and with high efficiency

Free-streaming Data Acquisition



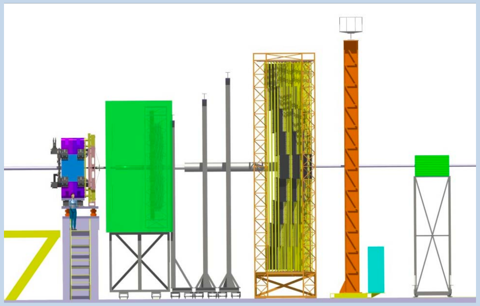
Trigger signatures are complicated and require (partial) event reconstruction

No conventional trigger, but self-triggered, autonomous front-end

Signals get time stamp from system clock and are shipped to DAQ

No a-priori association of signals to physical events!
„Event building" becomes non-trivial at high rates

CBM Online Computing

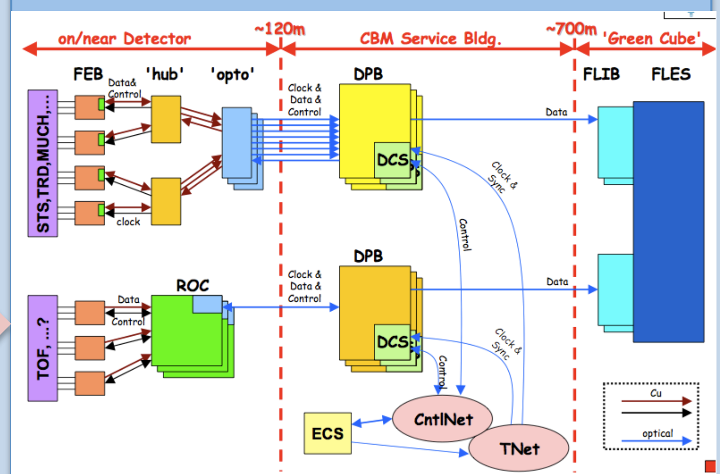


CBM FEE

Range of running modes:

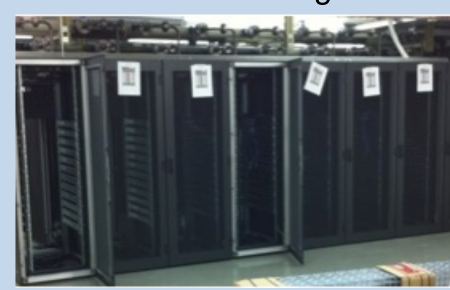
From triggerless (several kHZ) to extreme trigger (several MHz)

1 GB/s – 1 TB/s



1 GB/s

Mass Storage



Running Conditions

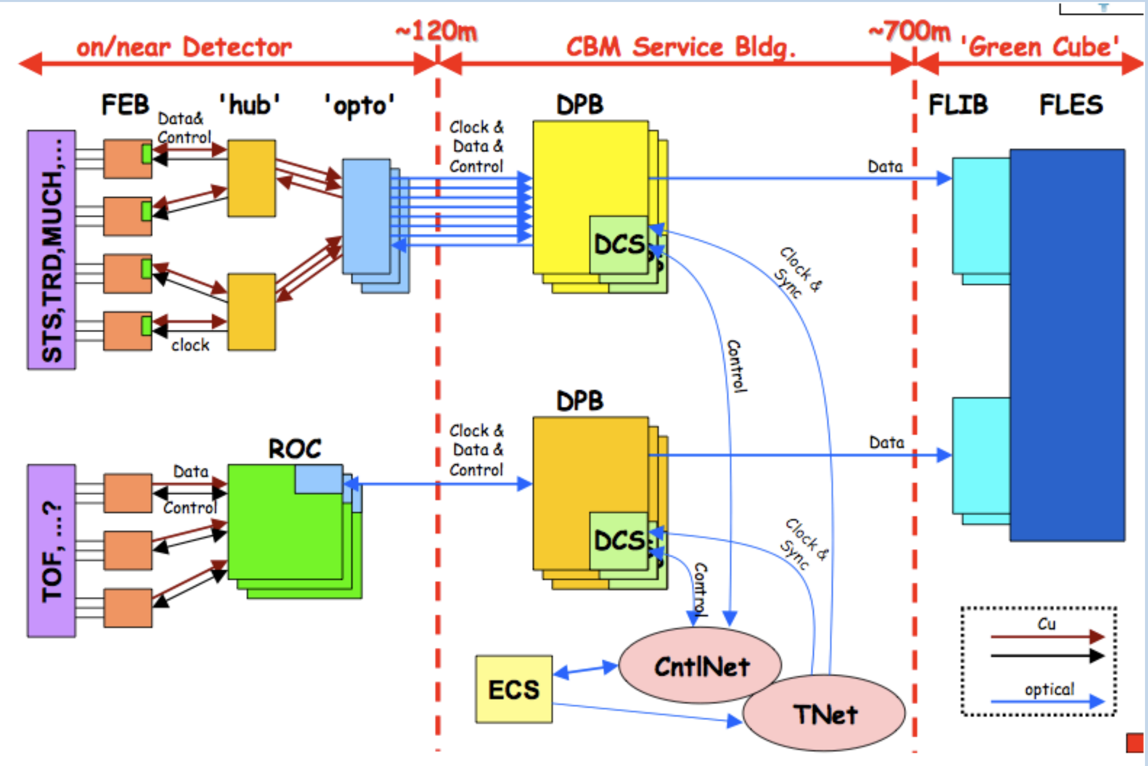
Condition	Interaction rate	limited by	Application
No Trigger	10^4/s	archival rate	bulk hadrons, low-mass di-electrons
Medium Trigger	10^5 /s – 10^6/s	MVD (speed, rad. tolerance)	multi-strange hyperons, open charm, low-mass di-muons
Max. Trigger	- 10^7/s (even more for p beam)	On-line event selection	charmonium

Detector, FEE and DAQ requirements are given by the most extreme case

Design goal: 10 MHz minimum bias interaction rate

Requires on-line data reduction by up to 1,000

Online Data Flow



- FPGA (DPB, FLIB): Data aggregation, pre-processing (e.g., cluster finding), time slice building
- CPU/GPU (FLES): (Partial) event reconstruction, data selection

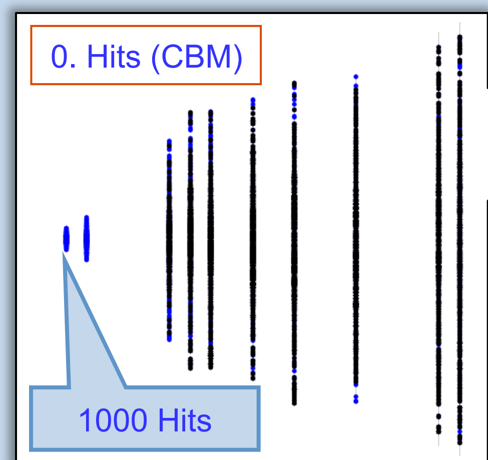
Towards the CBM Online Project

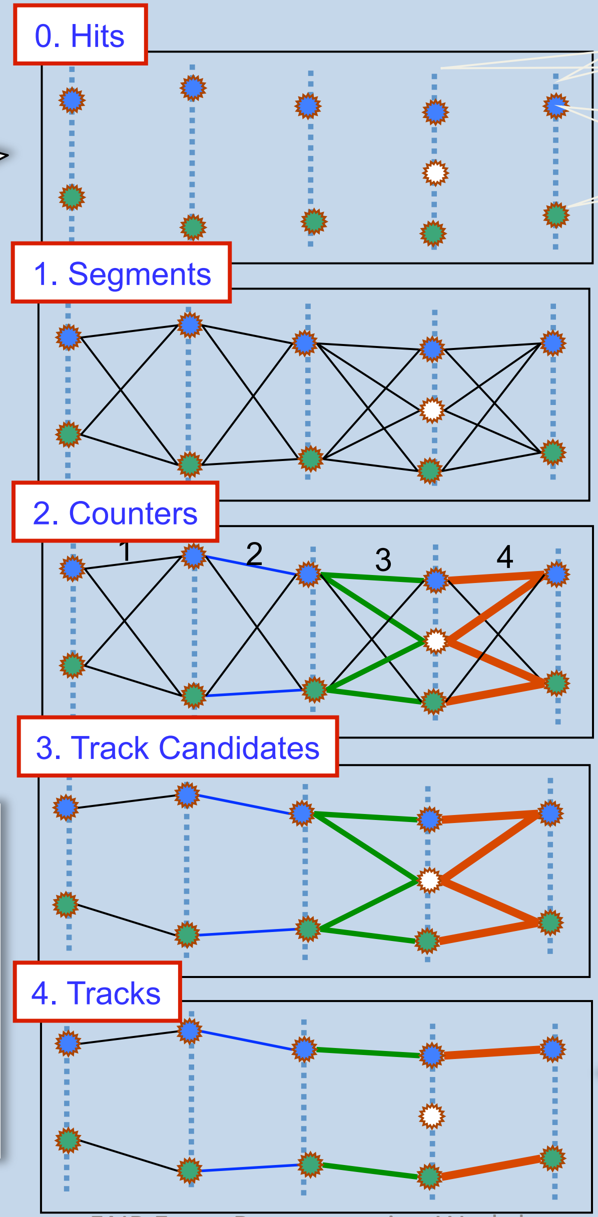
- DPB: Data transport / pre-processing — Linev
  - board design, firmware, interfaces to FEE
- FLIB: „Event“ building — de Cuveland
  - board design, firmware
- FLES architecture — de Cuveland
  - machines, network, infrastructural software
- FLES computing — Kisel
  - online reconstruction and analysis
- Online monitoring — Adamczewski-Musch
- Controls — Kolb
- Configuration — Ivanov
  - database(s)
- Software environment — Friese
  - simulations
  - alignment and calibration

Approaches to fast event reconstruction

Track finding in the STS – Cellular Automaton

Track finding: Wich hits in detector belong to the same track?



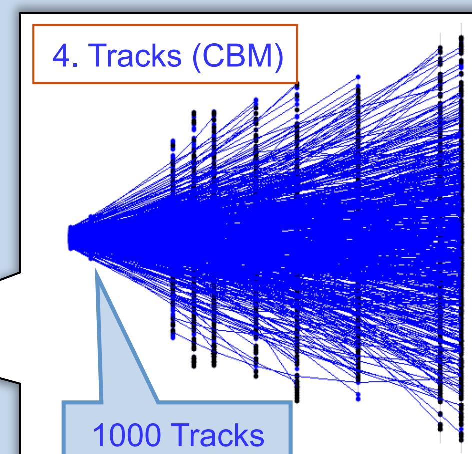


Cellular Automaton:
1. Build short track segments.
2. Connect according to the track model, estimate a possible position on a track.
3. Tree structures appear, collect segments into track candidates.
4. Select the best track candidates.

see talk by I. Kisel

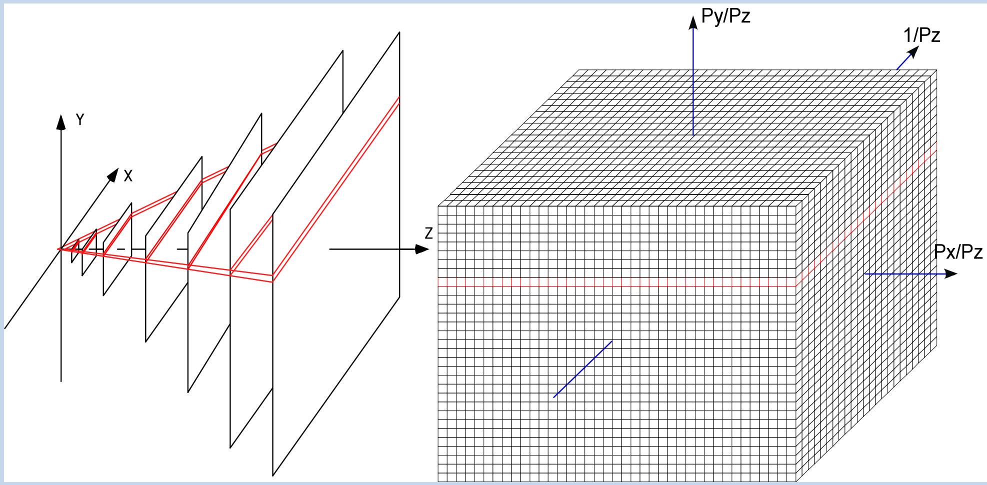
Cellular Automaton:
- local w.r.t. data
- intrinsically parallel
- extremely simple
- very fast

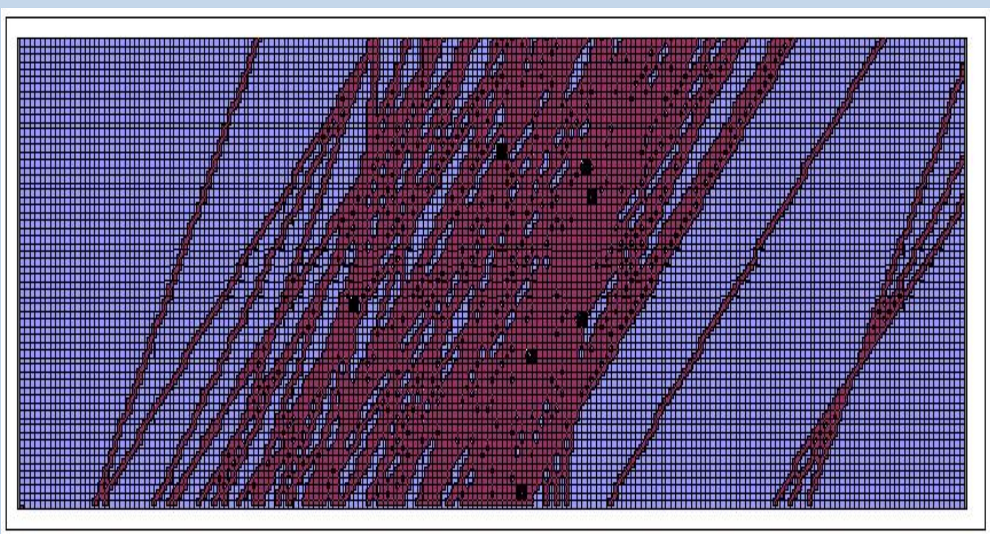
Perfect for many-core CPU/GPU !



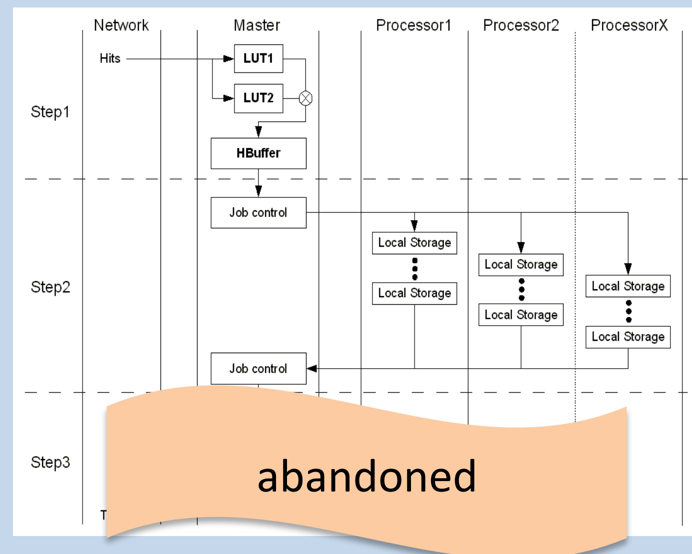
STS track reconstruction: Hough Tracker

3-D Hough transform

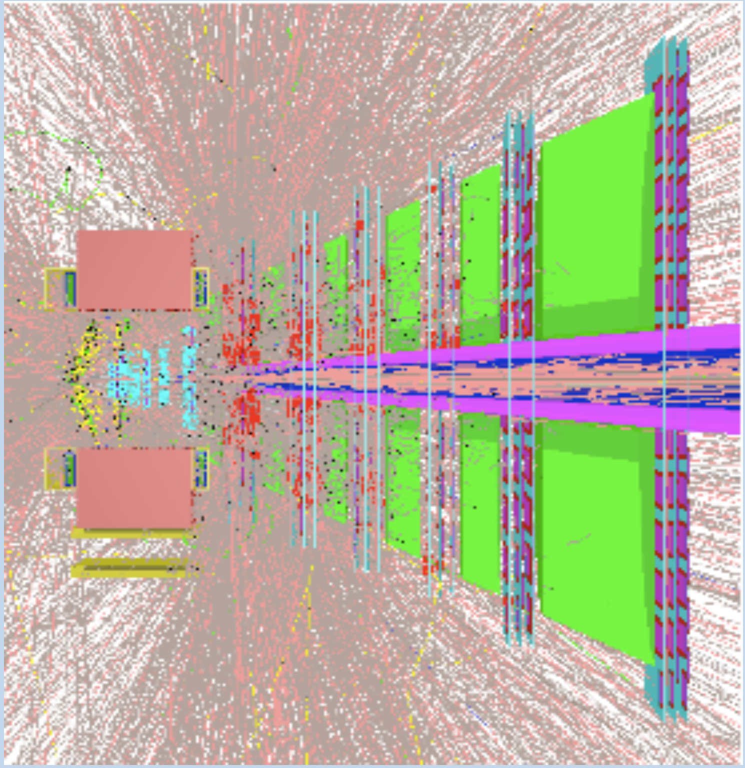


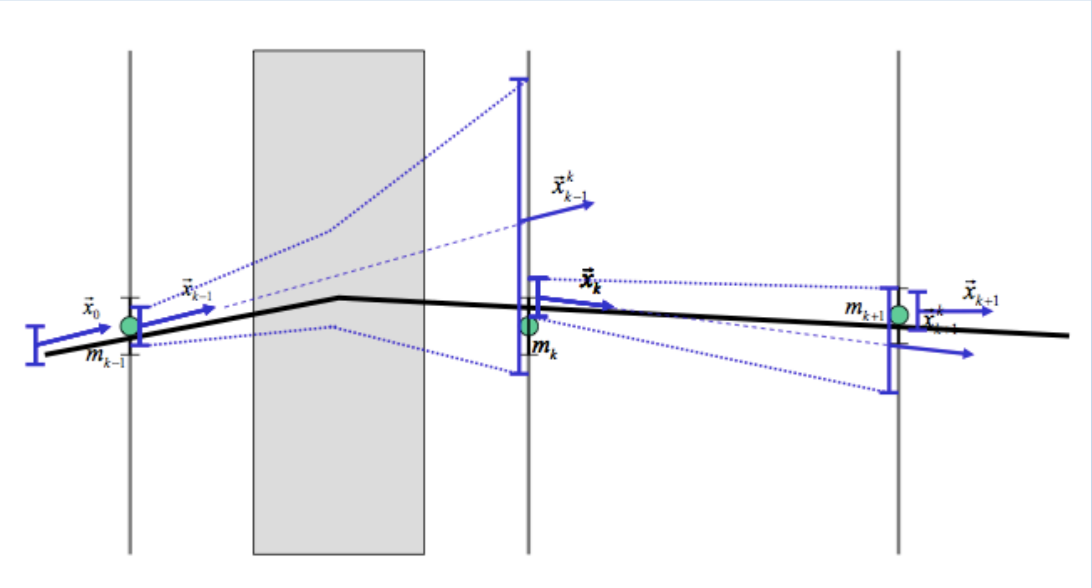


- 3-D Hough space is sliced into 2-D layers
- Histogramming operation ideally suited for FPGA
- Hough transform in non-homogeneous magnetic field is prepared offline by LUTs
- 2-D + 3-D peak finding in Hough space
- Current implementation in CELL-BE (Sony Playstation III) as prototype for FPGA array
- Reconstruction efficiency ≈ 90 % for p > 1 GeV



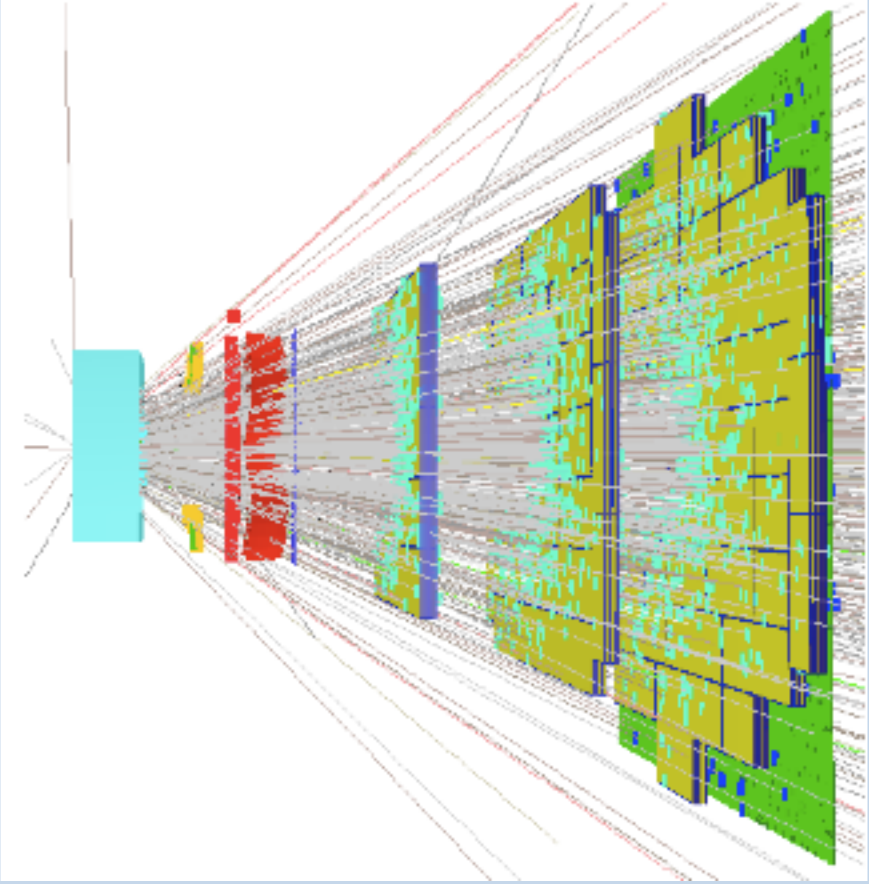
Tracking in the muon detector





- „Active absorber“ system: absorbers are interlayed with 6x3 detector layers
- Tracks from STS are used as seeds
- Track following with Kalman Filter
- Propgation with 4th order Runge-Kutta
- Hit association: nearest neighbour / branching / weighting

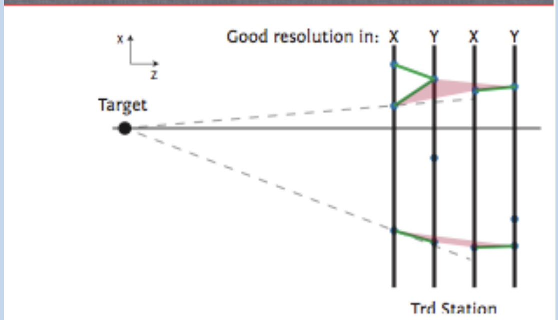
Tracking in the TRD



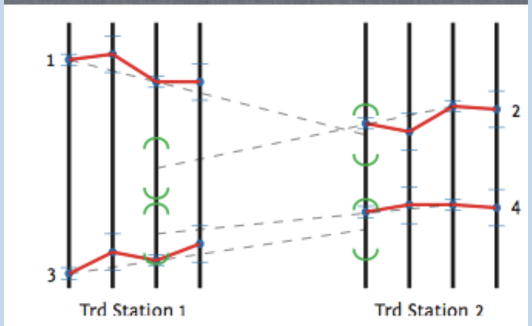
- 10 - 12 identical layers
- Track finding similar to tracking in the muon detector (track following + Kalman Filter)

Alternative: Standalone Track Finding in TRD

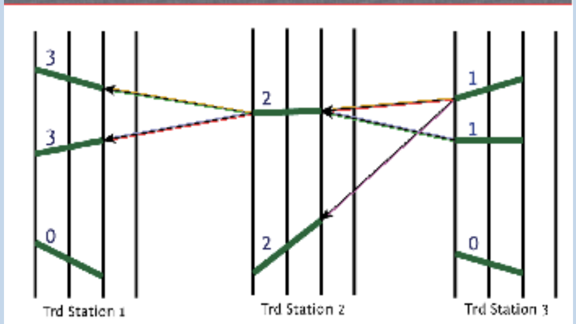
Tracklets Finding Step



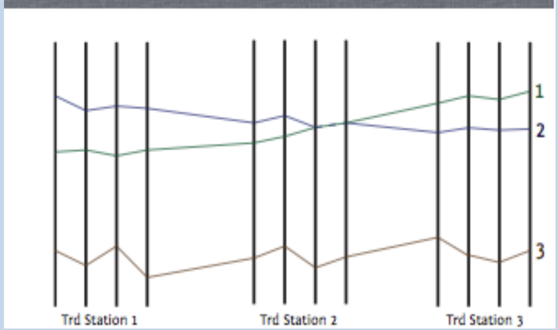
Neighbours Finding

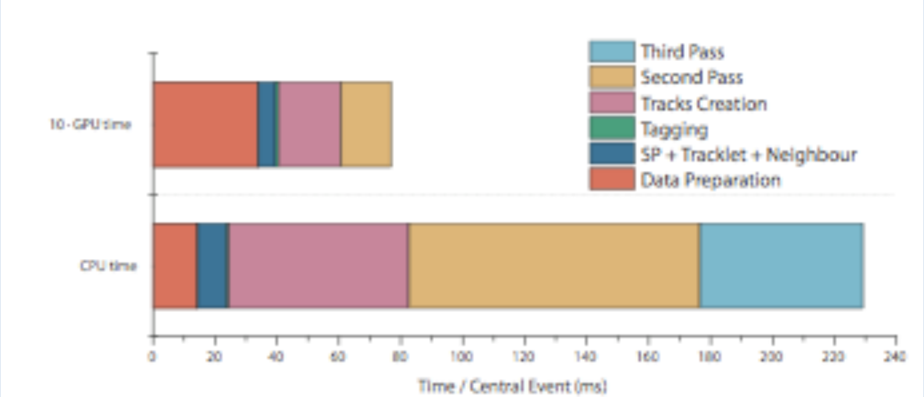


Tagging and Track Creation



Track Selection

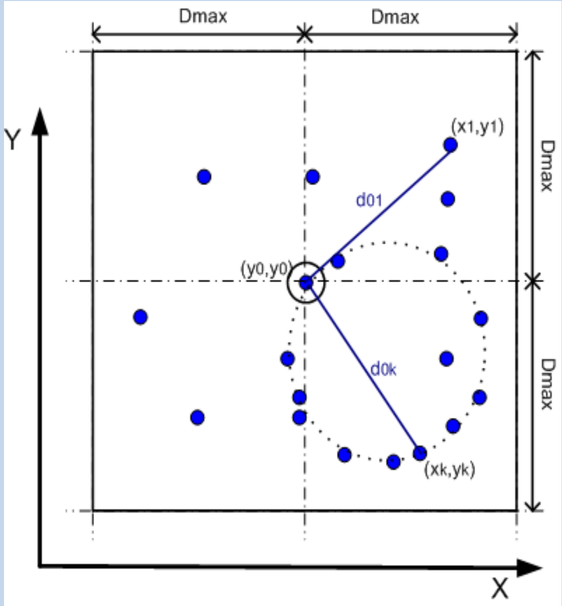


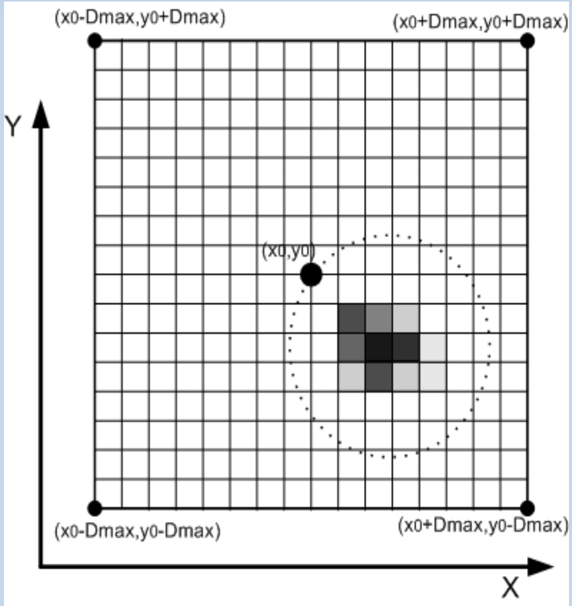


GPU time multiplied by factor of 10
Speedup of the whole algorithm: 30x
Speedup w/o data preprocessing: 50x

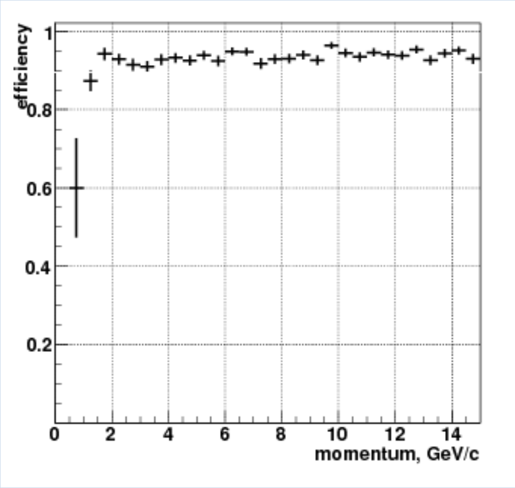
- CA algorithm similar to that in STS
- Implemented in CUDA on Tesla

Reconstruction of RICH rings





UrQMD, central Au+Au @ 25 AGeV + 5 e^+ + 5 e^-

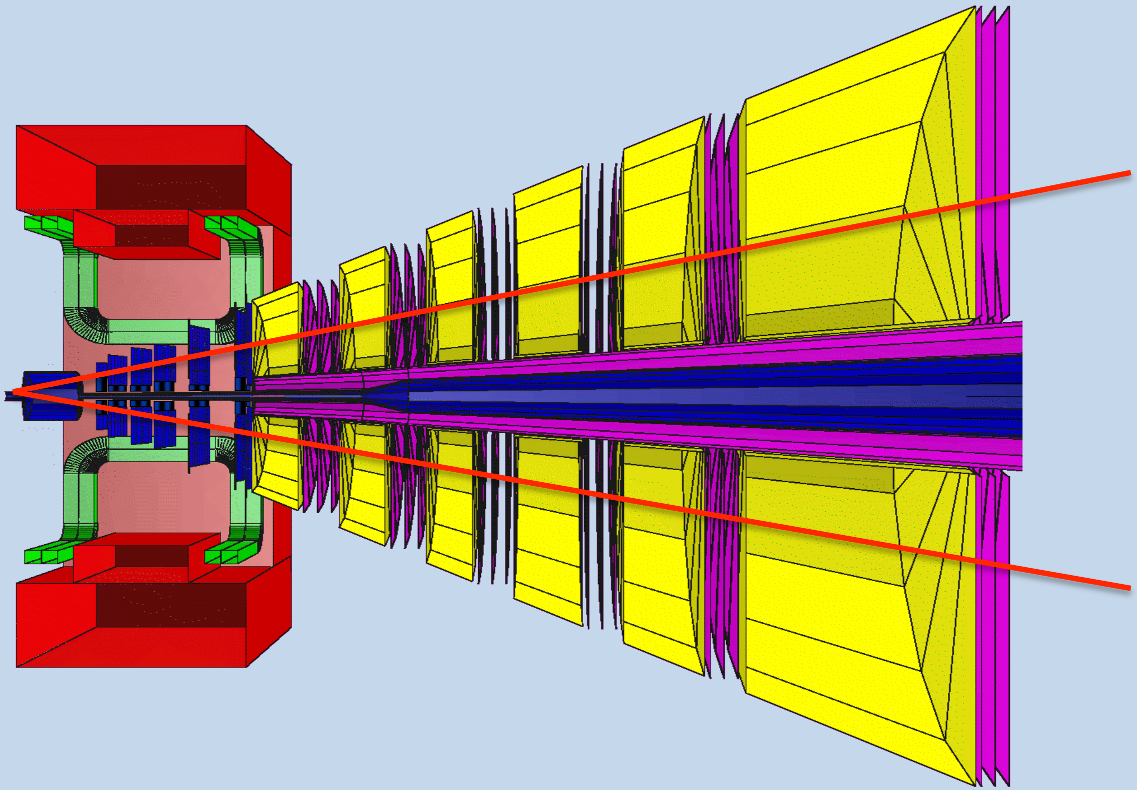


- Ring finding by localised Hough Transform (pre-selection of hits)
- Ellipse ring fitter
- Rejection of fake rings by quality criteria (ANN)
  - number of hits on ring, χ^2, largest angle
  - half axes, rotation angle
- Efficiency 92 %, fake rings 3.5 / event

Trigger Signatures

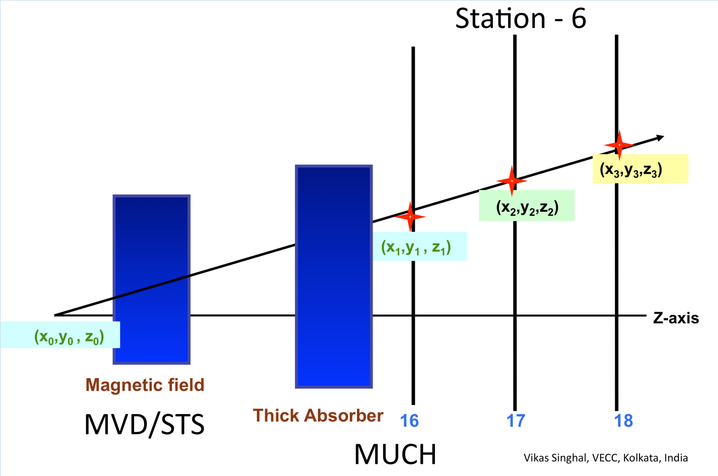
- Signatures vary qualitatively:
  - local and simple: J/ψ->$\mu^+\mu^-$
  - non-local and simple: J/ψ -> e^+e^-
  - non-local and complex: D,Ω->charged hadrons
- For maximal interaction rate, reconstruction in STS is always required (momentum information), but not necessarily of all tracks in STS

Trigger in the Muon System



Signature: Two main-vertex tracks after the last absorber

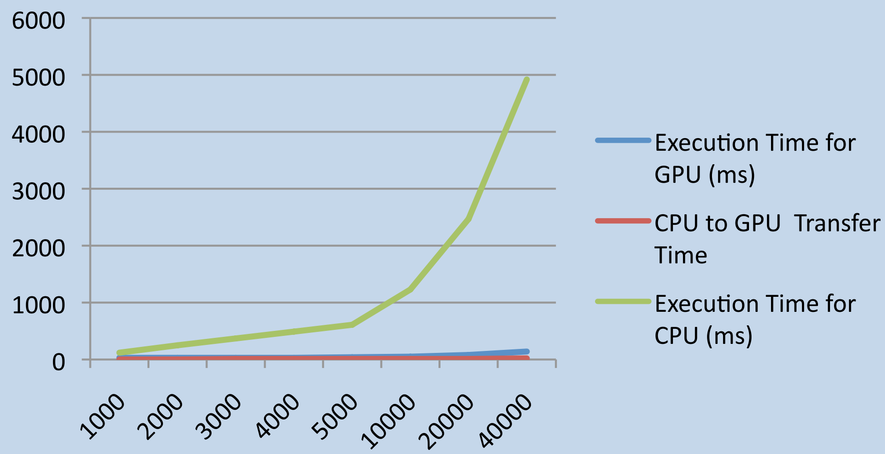
MUCH Trigger Implementation in CUDA



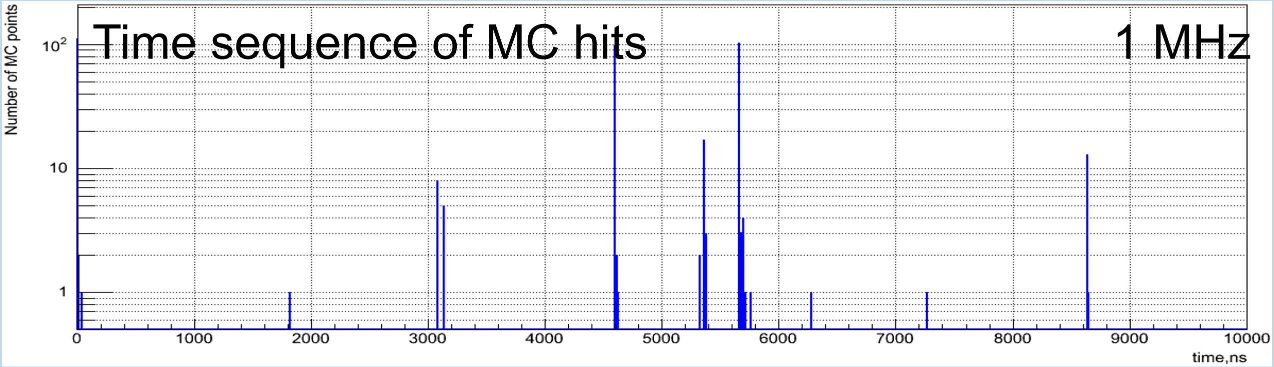
BSF:- Total Input Events / No of Background event survived after Threshold

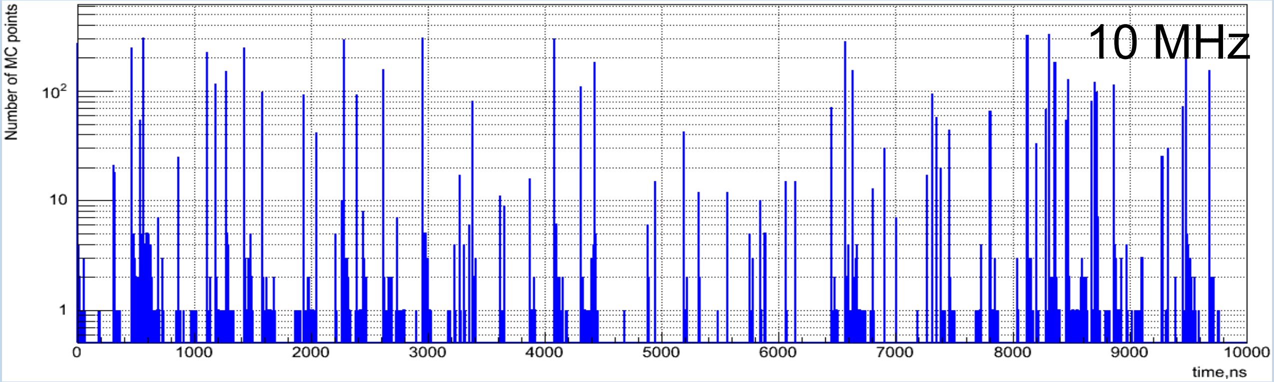
Threshold	Events Survived (1000 Events)	BSF
A	431	2.32
B	120	8.33
C	100	10
D	41	24.39

- Fit triplet by straight line and extrapolate backwards to target
- Implemented in CUDA and tested on NVIDIA Tesla



Treatment of Free-Streaming Data



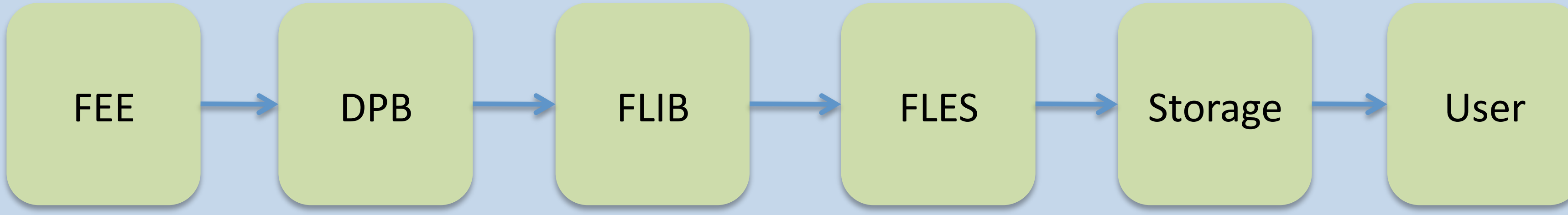


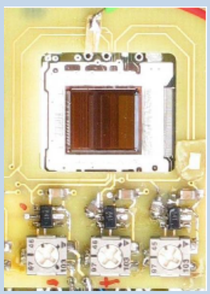
- Up to 1 MHZ, data can be „trivially“ pre-sorted into events; reconstruction then proceeds on event base (already developed)
- For higher rates, time-based reconstruction (4-d) is required
- Work in progress

Reconstruction Algorithms: Summary

- All current algorithms are developed for deployment on a many-core CPU/CPU environment
- Languages: C++ / OpenCL / CUDA
- Development of FPGA-suited reconstruction in STS (Hough Transform) is not continued
- Current algorithms work on the event-by-event level (valid up to 1 MHz interaction rate). Work on algorithms for higher rates are under development.

Deployment of FAIRROOT on FLES





FAIRROOT on FLES

- FLES software framework not yet decided
- FAIRROOT Cons:
  - overhead (t.b. specified)
  - parallelisation unclear
- FAIRROOT Pros:
  - configuration, services, I/O already available
  - straightforward consistence with offline computing
- Not necessarily a yes-or-no decision, but rather how much and where
  - data classes and algorithms probably simple C/C++; easily to be wrapped in ROOT for I/O
  - usage of FAIRROOT infrastructure

CBM Requests to GSI-IT

- Continuation and further development of FAIRROOT is indispensable
  - time-based data flow
  - deployment on many-core systems
  - GPU support (CUDA / OpenCL)
- Fair share for experiment-specific activities
- Possible participation in CBM-specific projects:
  - FLES software framework
  - FLES online reconstruction
  - data pre-processing: FPGA or software
  - alignment and calibration
  - databases