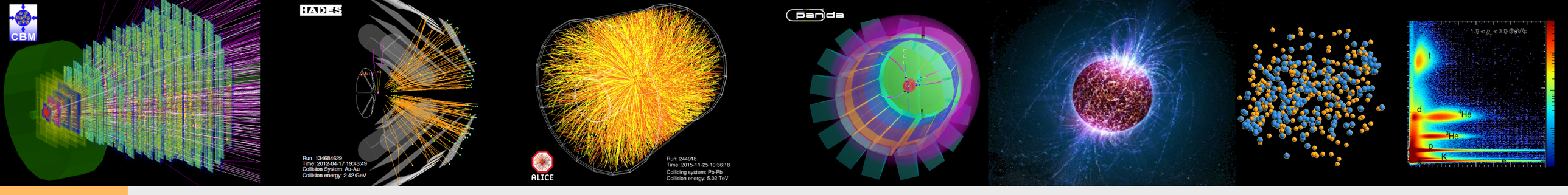
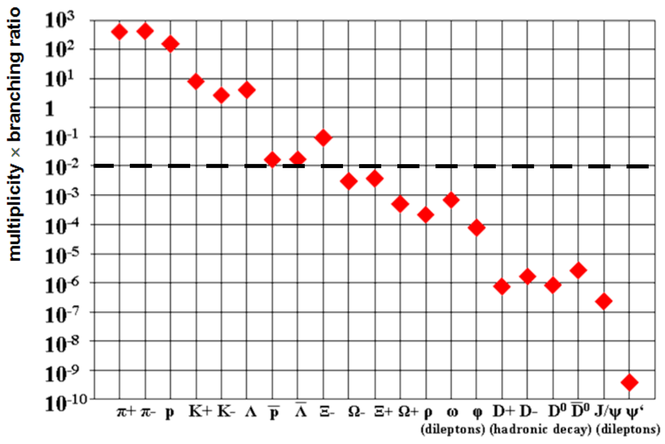
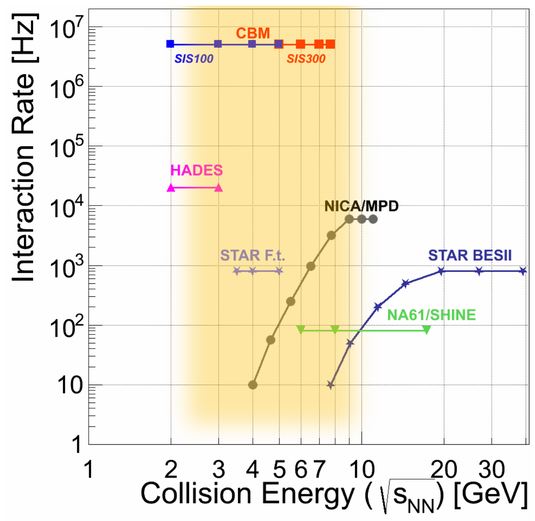


Time-based particle reconstruction and event selection in the CBM experiment

Novel Data-processing Concept for CBM

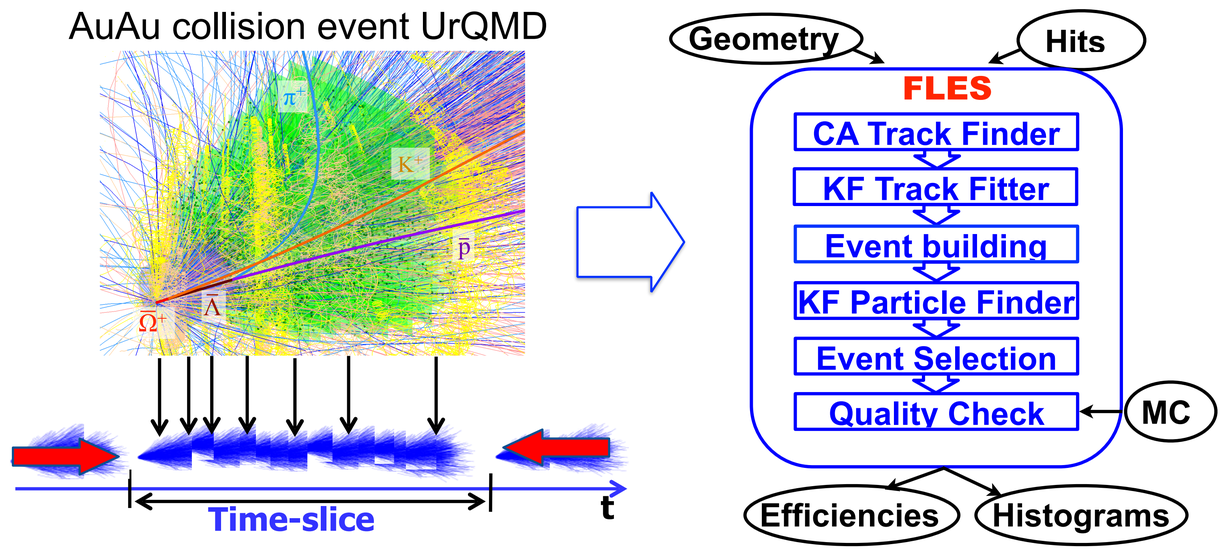




- CBM key observables include rare probes;
- unprecedented interaction rates of 10^5-10^7 collisions per second;
- complex observable signatures: no hardware trigger;
- free streaming data;

Rare prob measurements require novel data-processing concept.

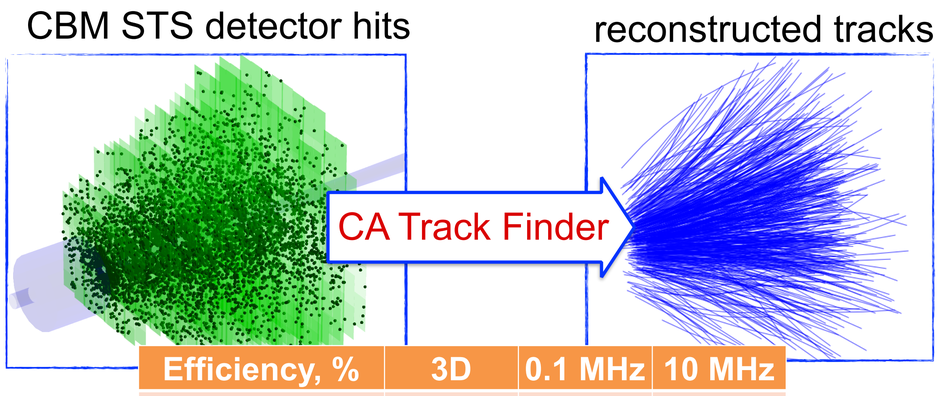
First Level Event Selection Package (FLES)



- continuous time-slices instead of individual collisions;
- time-based reconstruction online on a parallel FLES farm;
- need for highly parallelised algorithms;

Highly parallelized package with online reconstruction components.

4D Cellular Automaton Track Finder

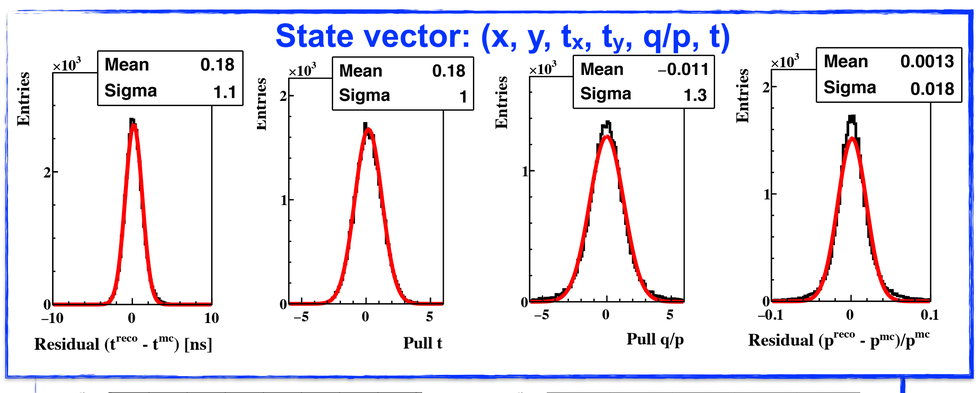


Efficiency, %	3D	0.1 MHz	10 MHz
All tracks	92.5	93.8	91.7
Ghost level	1.8	0.6	0.6

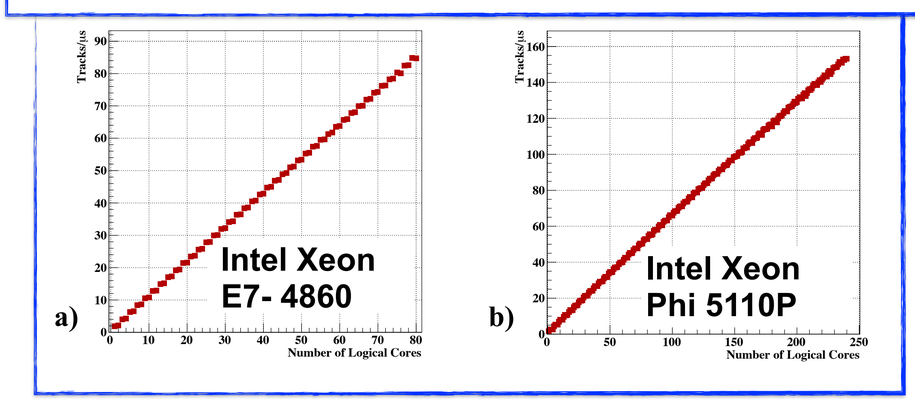
- efficient and stable track reconstruction;
- vectorised and parallel algorithm;
- leaner scalability on parallel computer systems;
- time-based performance comparable with event-based analysis;

Parallel and SIMD-ised algorithm allows to reconstruct tracks in time-slices.

4D Kalman Filter Track Fit



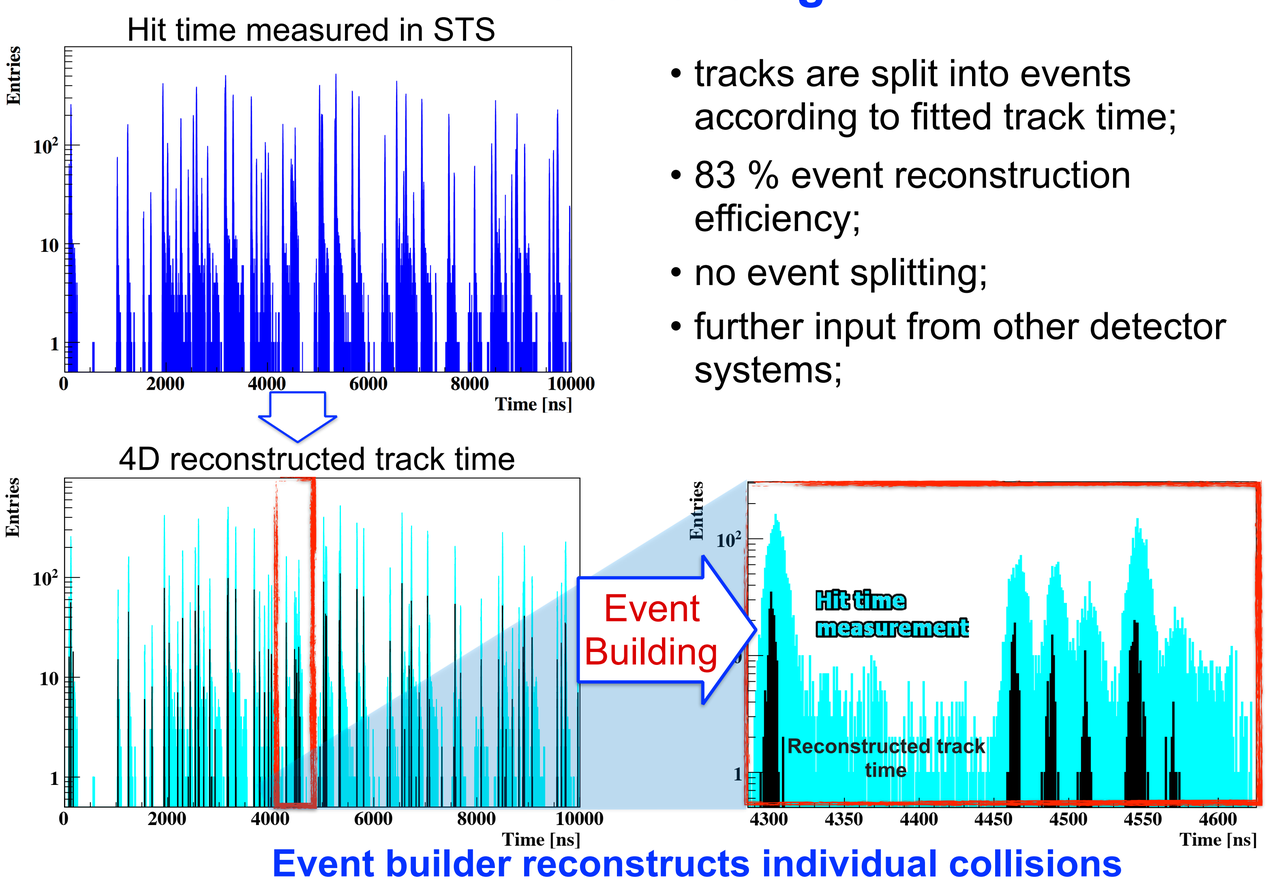
- unbiased track parameters;
- high resolution;
- correctly estimated errors;



- strong scalability on various computer systems;

Highly parallelized library to evaluate parameters with a high resolution.

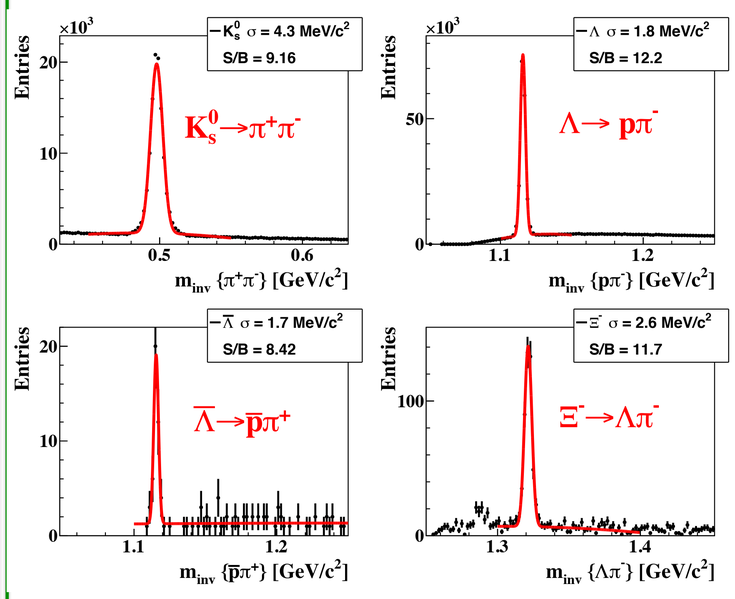
Event Building



- tracks are split into events according to fitted track time;
- 83 % event reconstruction efficiency;
- no event splitting;
- further input from other detector systems;

Event builder reconstructs individual collisions out of time-sliced data.

Short-lived Particle Reconstruction



Case	3D	10 MHz
K^0_S, %	22.9	21.2
Λ, %	21.9	19.6
Ξ-, %	7.8	6.3

- short-lived particles are of particular physics interest;
- particles are reconstructed via decay products;
- time-based particle reconstruction performance is comparable to the event-based procedure;
- work in progress: multi-vertex analysis and all detector sub-systems.

Reconstruction chain for free-streaming data processing in CBM is established.

Conclusions

- the CBM FLES package is capable of **free-streaming data processing**;
- FLES is **efficient, fast** and highly **parallelized**;

HELMHOLTZ