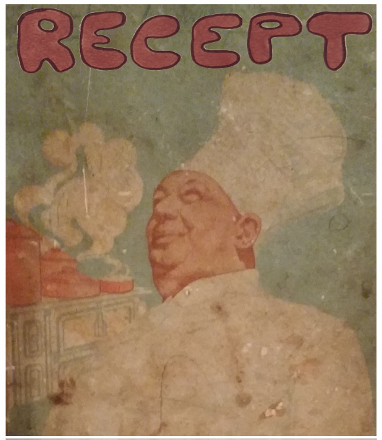
European Research Council
Established by the European Commission



Tracking for LHCb upgrade, using a full software trigger at 30MHz

Renato Quagliani

On behalf of the LHCb collaboration

LPNHE PARIS

Connecting the dots

20-22 March 2018

Seattle, Washington, USA

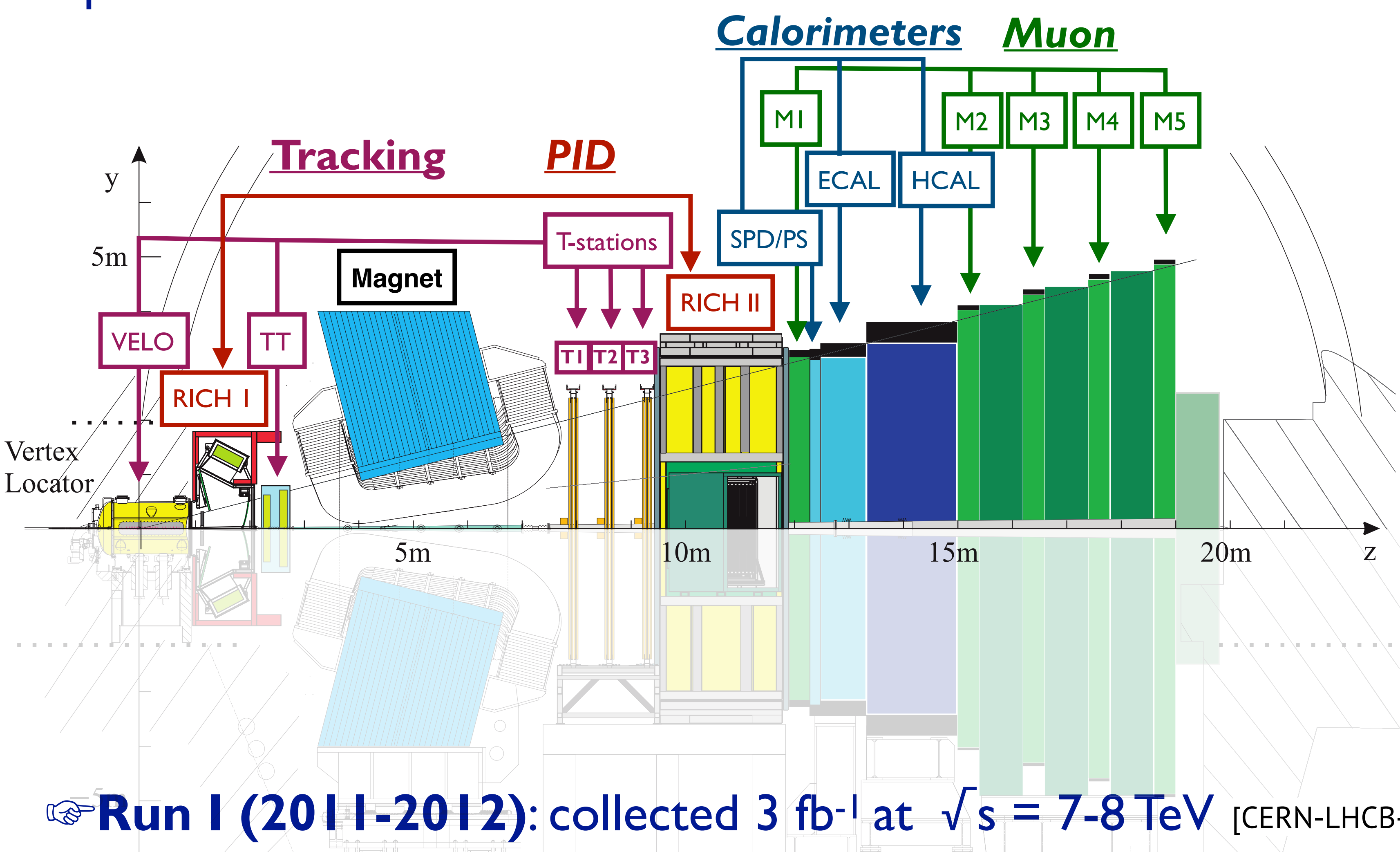
1. LHCb upgrade and the trigger strategy
2. Track reconstruction for the LHCb upgrade at collision rate
3. High-Throughput computing for the LHCb upgrade and current status
4. Conclusions and prospects

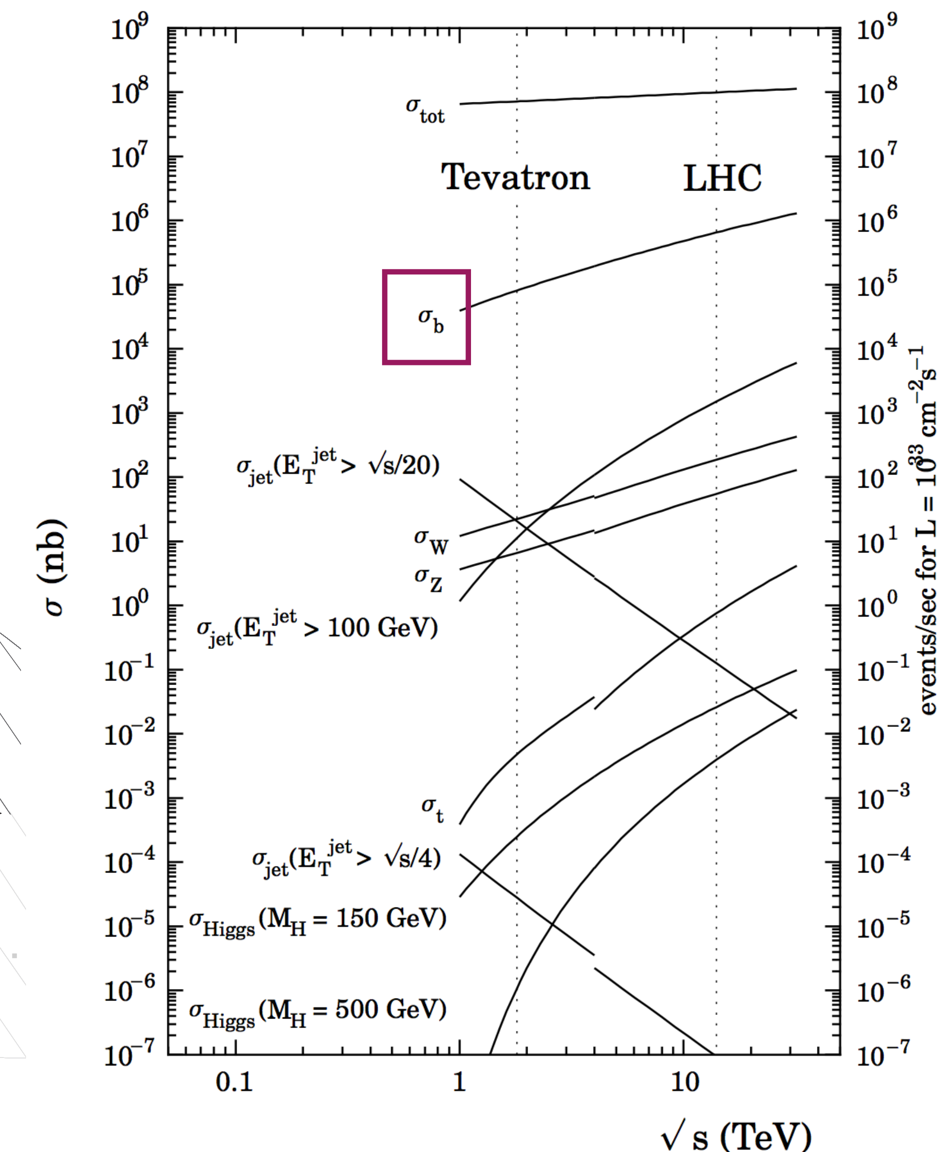
LHCb experiment : a dedicated flavour physics experiment

LHCb is a high precision experiment at CERN

☞**Large cross section of *b* and *c* quark production** in pp collisions.

☞**$b\bar{b}$ boosted in forward region**: LHCb single arm forward spectrometer.



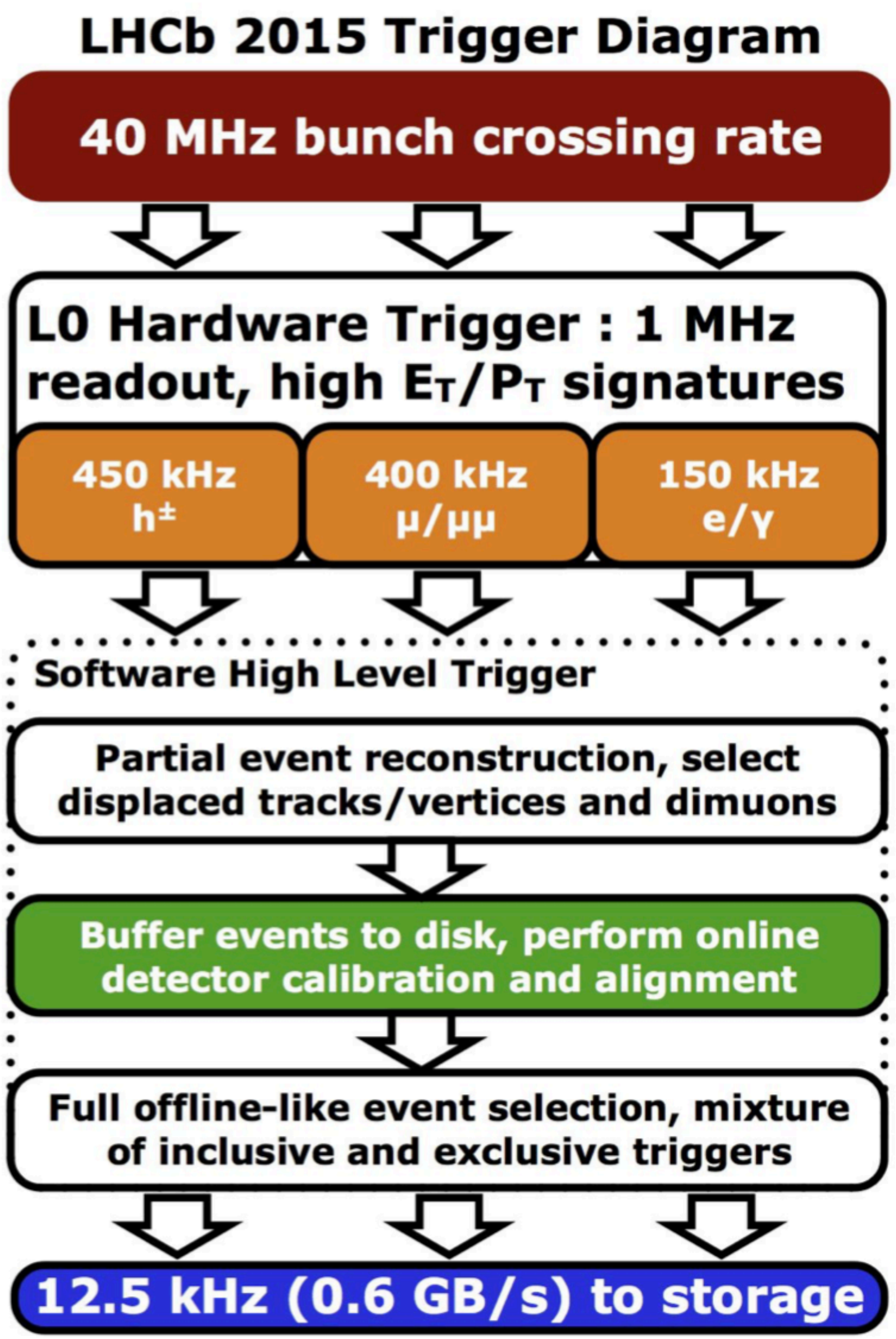


☞**Run I (2011-2012)**: collected 3 fb^{-1} at $\sqrt{s}$ = 7-8 TeV [CERN-LHCB-DP-2014-002]

☞**Run II (2015-2018)**: 4 times more $b\bar{b}$ expected at $\sqrt{s}$ = 13 TeV.

☞**Run III (2021 - 2025+)**: upgraded detector and 5 times current luminosity (5fb^{-1}/year).

Run II trigger strategy



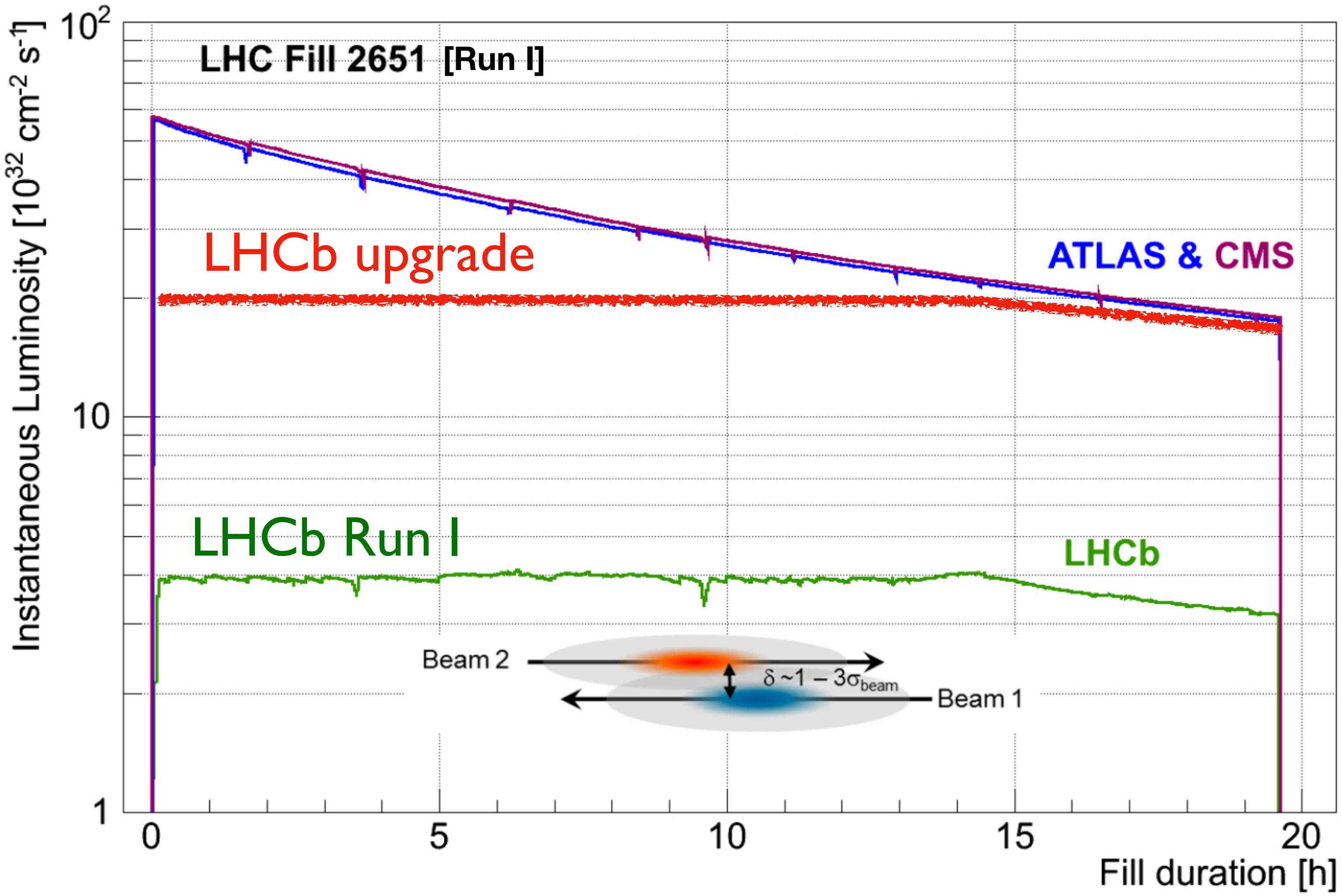
1. L0 Hardware trigger reduces the rate to 1 MHz (Front-End read-out limitation)

2. Partial event reconstruction (HLT1)
3. Buffer events to disk and
   real time alignment and calibration.
4. Full event reconstruction and offline-like event selection.

Store on disk only high level object for the decay mode of interest.

LHCb limitations and upgrade

LHCb in Run I (and Run II)



Run III target

$$\mathcal{L}_{inst} = 20 \times 10^{32} cm^{-2} s^{-1}$$

☞ **μ = 7.6** ☞ **50 fb^{-1}**

Run I and II

$$\mathcal{L}_{inst} = 4 \times 10^{32} cm^{-2} s^{-1}$$

☞ **μ = 1.1-1.8**

☞ **3 (Run I) + 5 (Run II) fb^{-1}**

- Huge gain in physics capabilities if able to run at larger luminosity **[CERN-LHCC-2011-001]**
- LHCb will further expand its physics program as GPD.

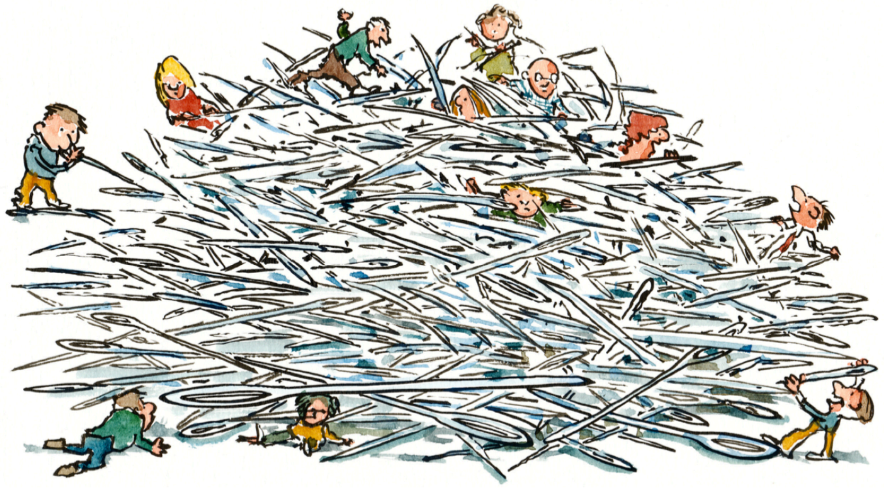
New detectors and trigger strategy.

The MHz signal era for LHCb

The rules of the games at 5 times larger luminosity are different

- ☞**24 % (2 %)** of events contains a **charm (beauty)** hadron.
- ☞We can store only 2-10 Gb/s offline.
- ☞**Separate signal/background** but also efficiently **separate signals from other signals.**
- ☞Trigger has to be **flexible and efficient.**

Triggers today



Triggers in the future

LHCb limitations and upgrade

At larger luminosity with current detector

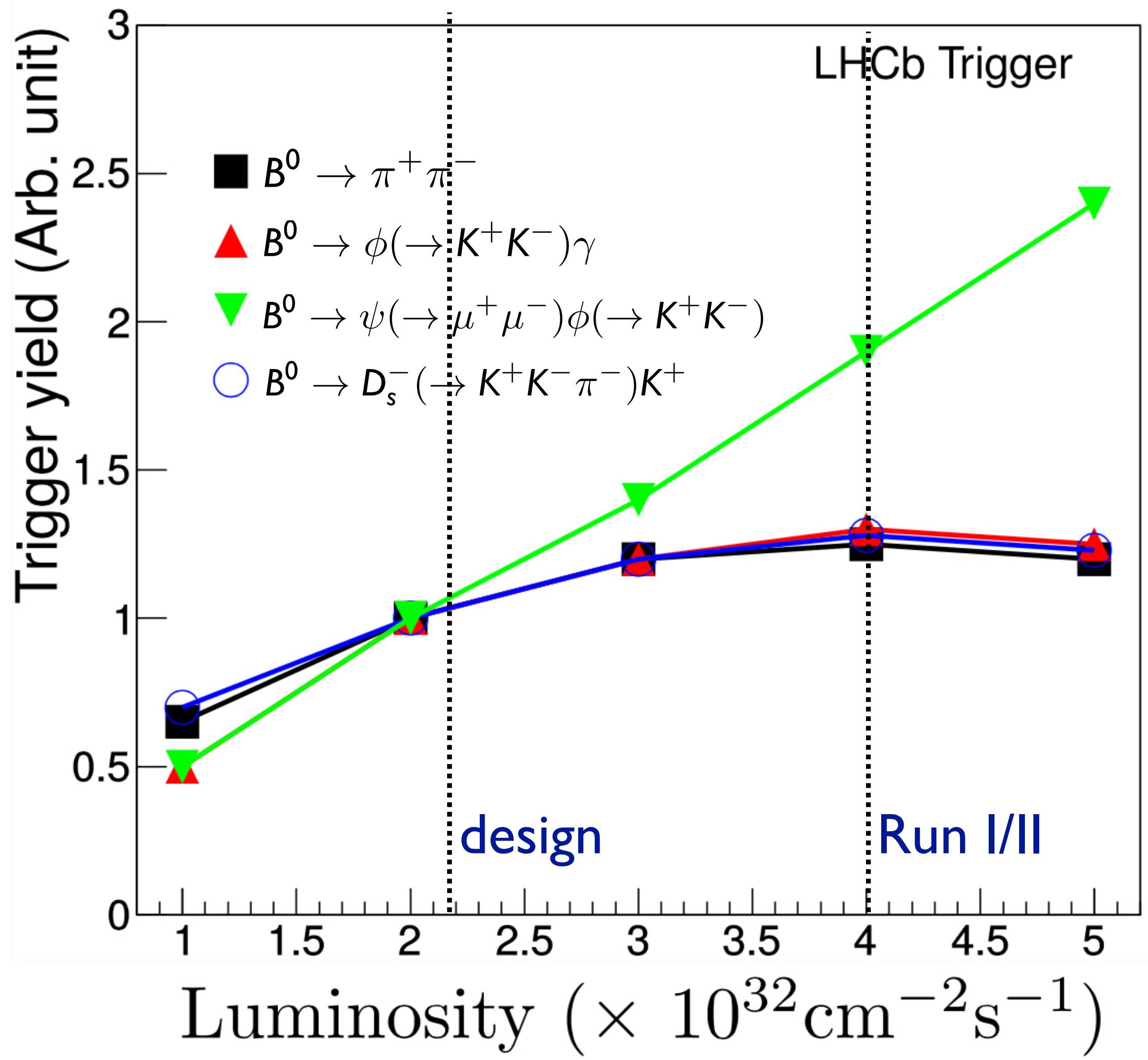
☞Too inefficient to keep 1 MHz FE read-out

Strategy for the upgrade

☞**Fully software based trigger** : handle efficiently the more complex event topology.

☞**Read out all detector at 40 MHz.**

☞**Replace the entire tracking system.**

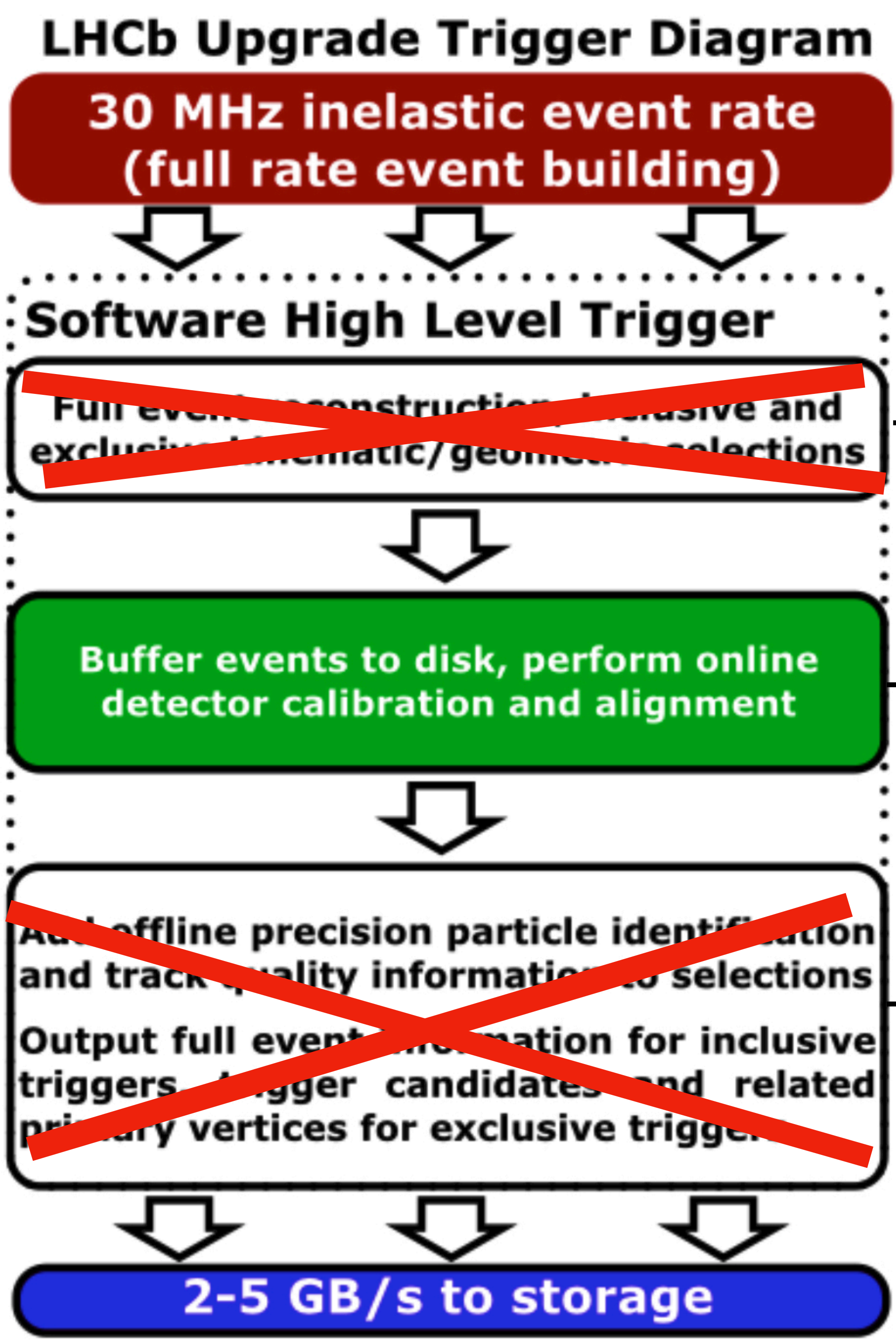


Scaling of trigger yields with current LHCb

[CERN-LHCC-2014-016]

LHCb trigger in Run III

Originally



The road to the upgrade…

From biannual upgrade review document [LHCb-PUB-2017-005]

Originally, full event reconstruction upfront.

CPU resources and budget are not unlimited and they put strong constraint.

Partial event reconstruction (HLT1)

- Data preparation for tracking
- Track reconstruction

Efficient events selection to reduce rate to 500-1000 kHz

Re-use Run II framework

Full event reconstruction (HLT II)

Best tracking performance, add particle identification information and offline quality selections.

Perform analysis directly on trigger output.

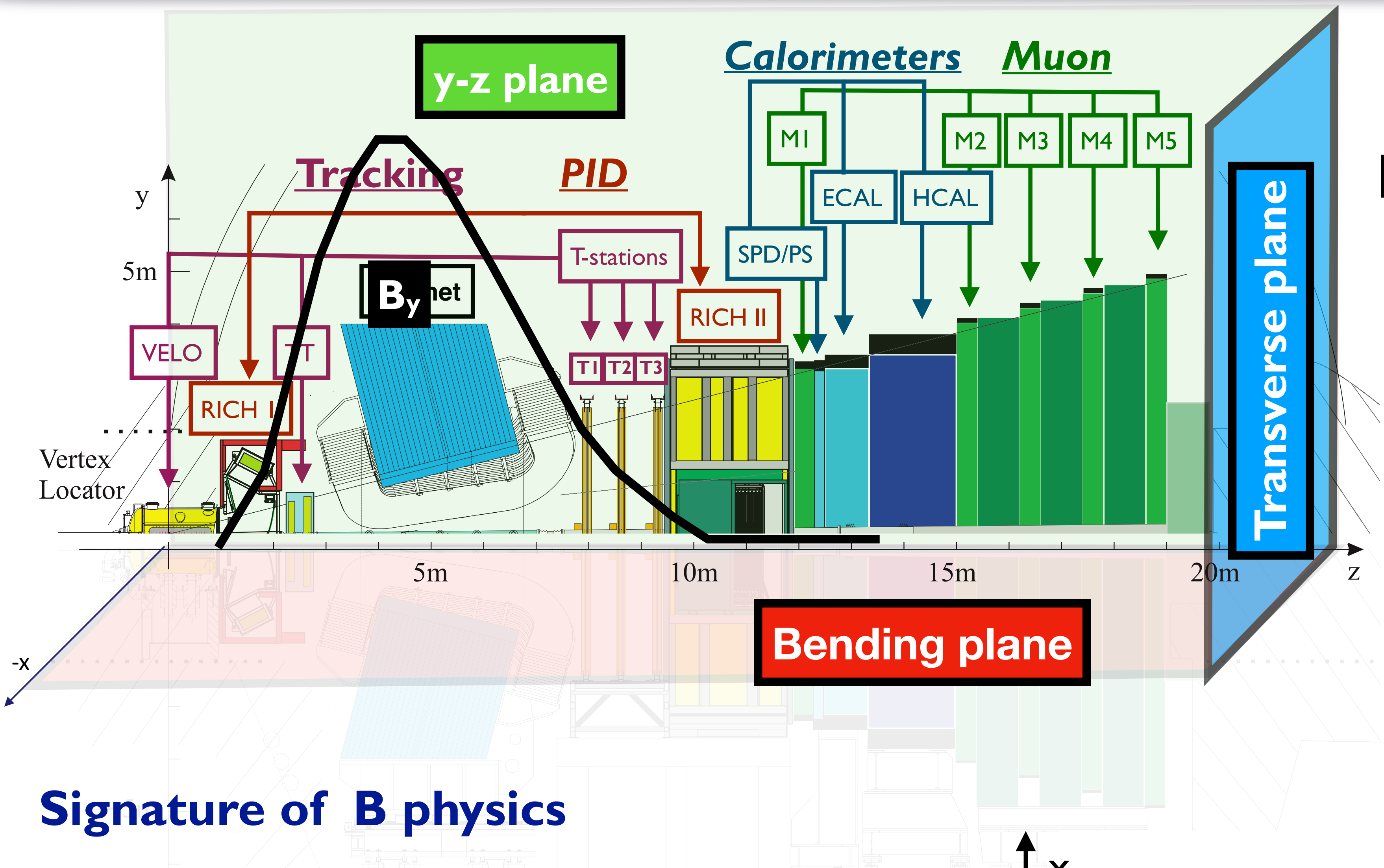
1. LHCb upgrade and the trigger strategy

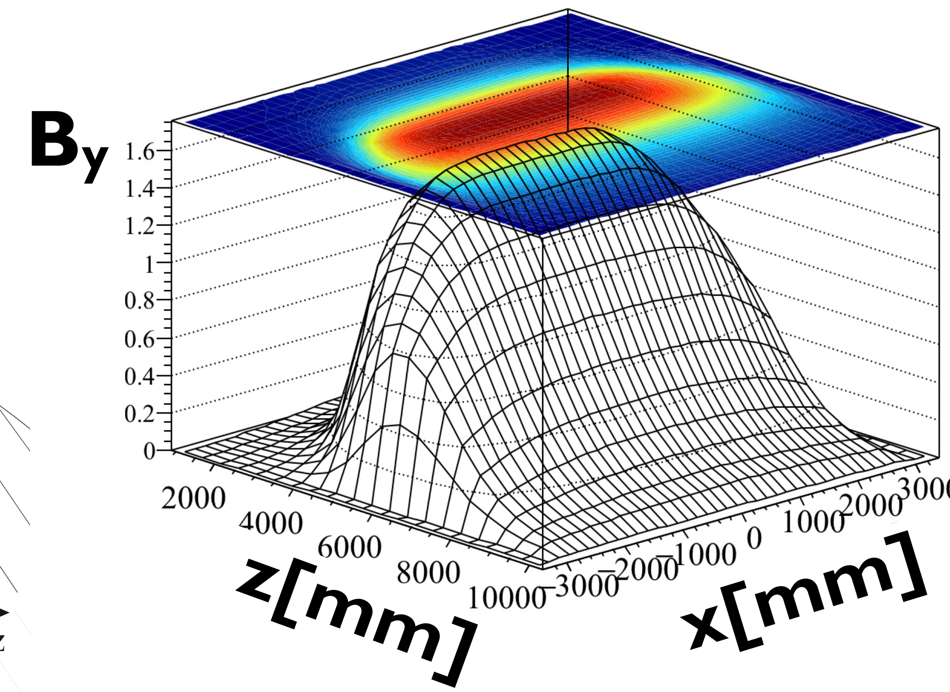
2. **Track reconstruction for the LHCb upgrade at collision rate**

3. High-Throughput computing for the LHCb upgrade and current status

4. Conclusions and prospects

LHCb B field and track behavior



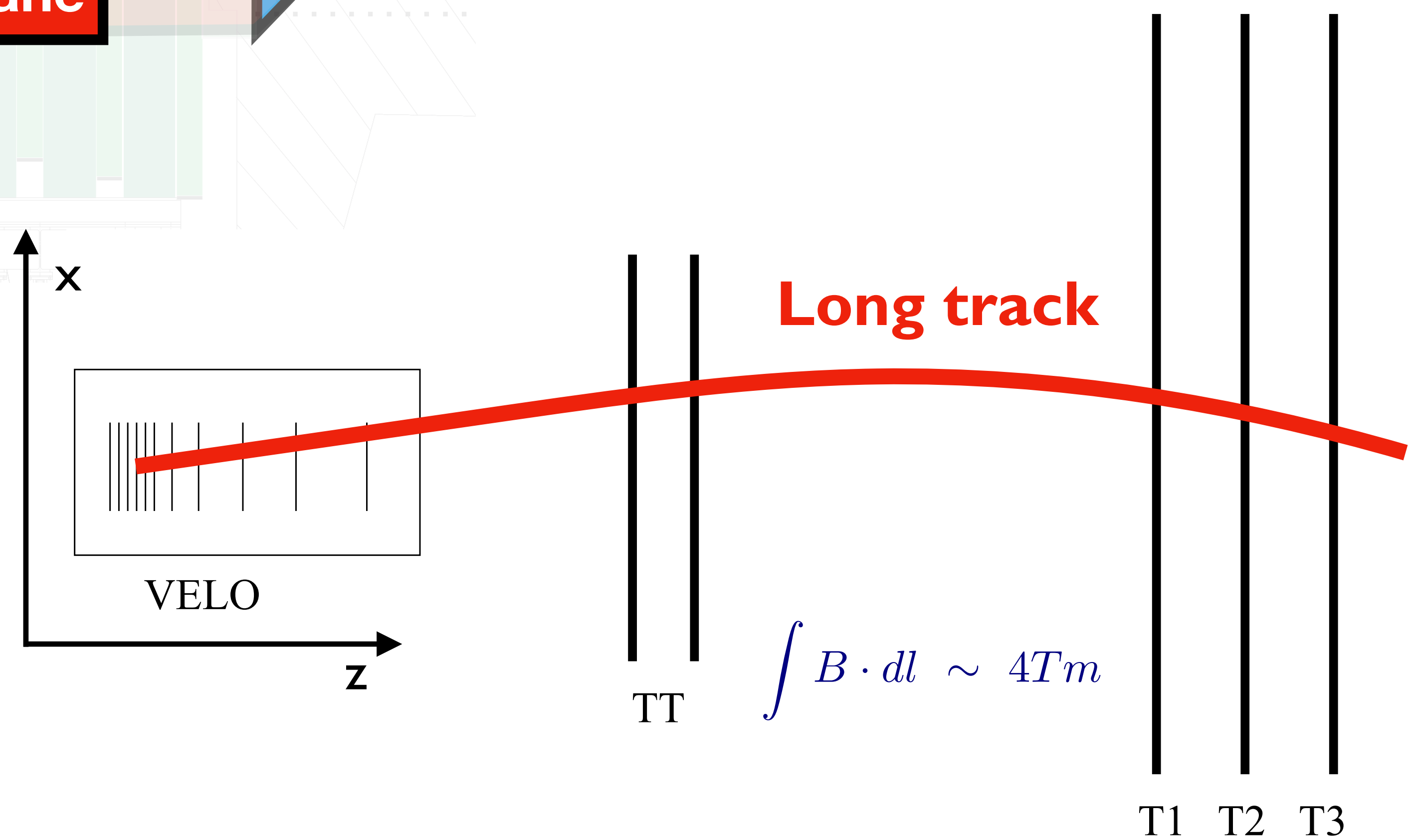


Signature of B physics

- ☞ High p_T (> 1 GeV/c) final states
- ☞ Displacement from primary vertex (PV)

Signature of charm prompt physics

- ☞ Moderate p_T (> 0.5 GeV/c) final states
- ☞ No displacement from PV

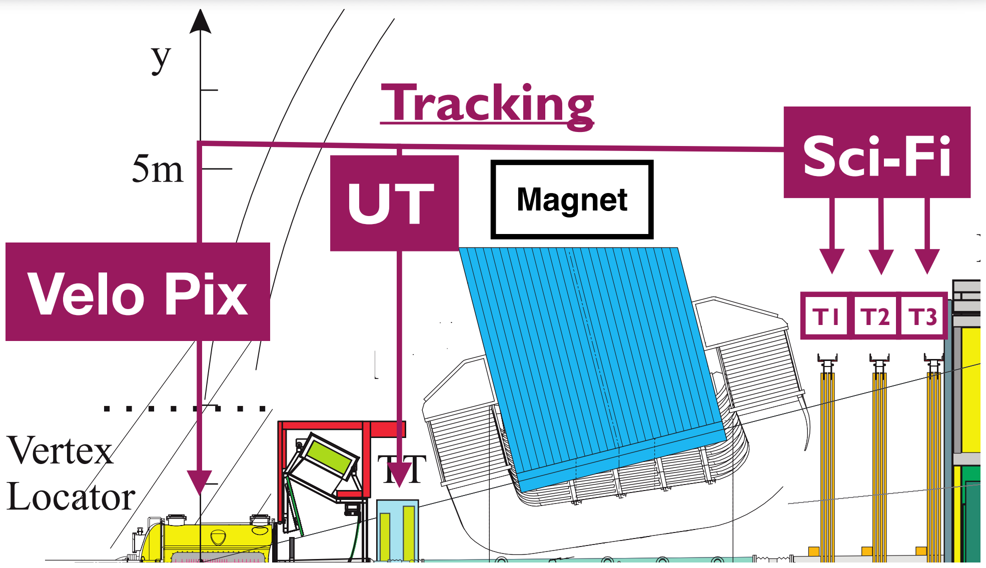


Partial reconstruction sequence : Velo tracking and PV finding

Velo Pix

LHCB-TDR-013

- **2-D pixel sensor** @-20°C: direct **x-y-z** measurement.
- **Tracking on raw data.**
- **5.1 mm distance** from interaction point (8.2 mm current VELO)

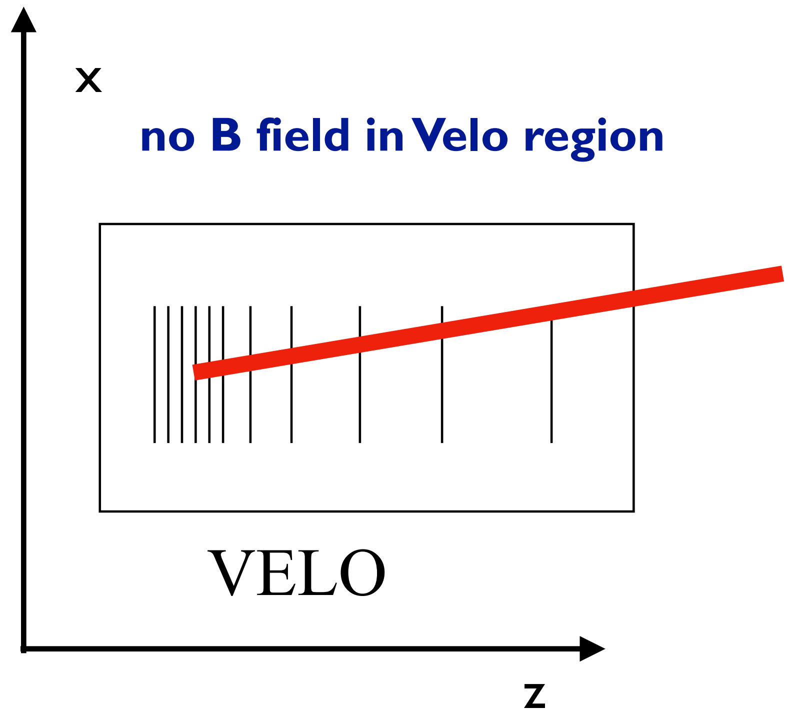


LHCB-TDR-013

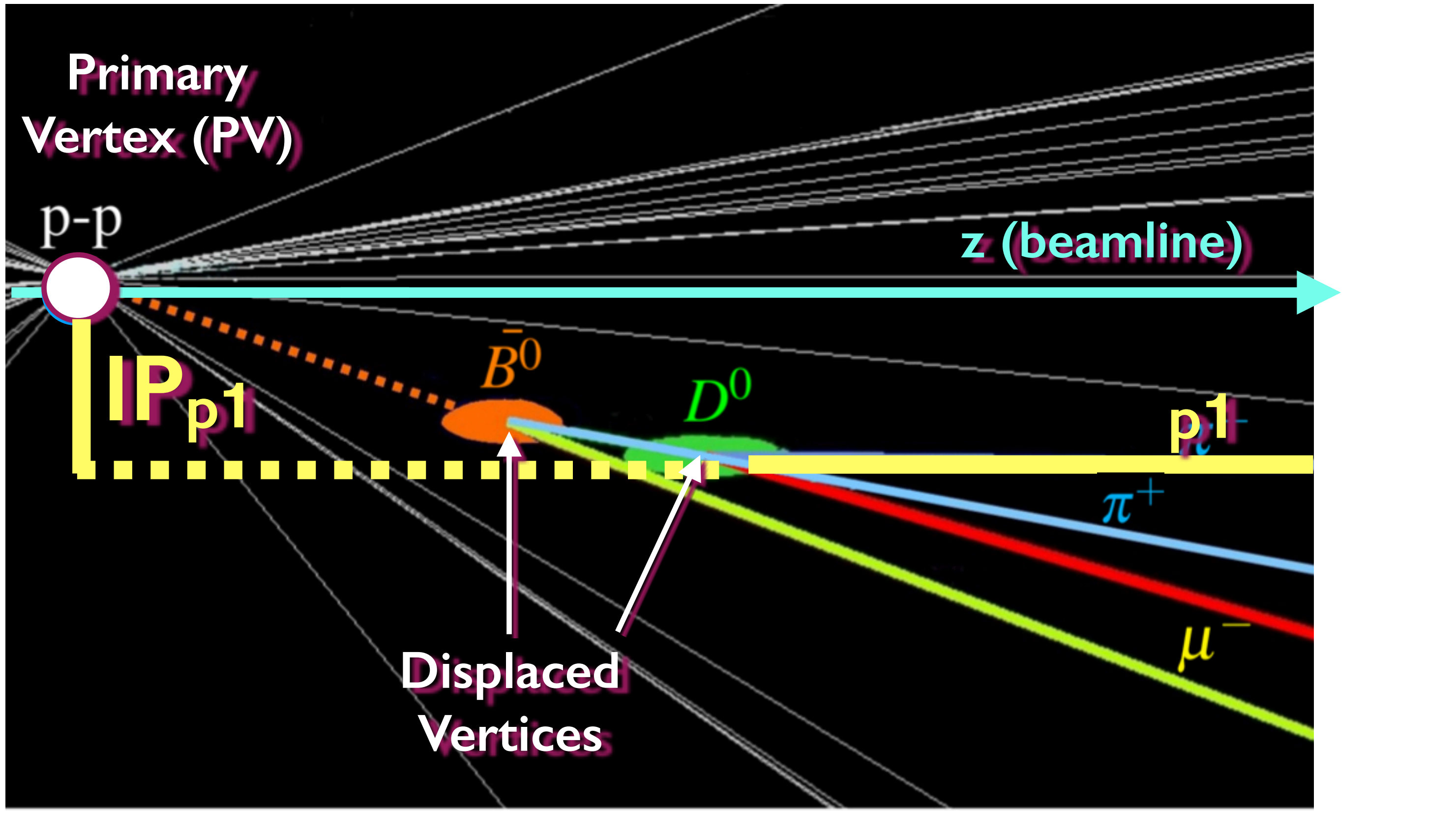
Velo tracking and PV reconstruction

Velo tracks used to find the PVs.

PV position used to identify displaced tracks in the event.



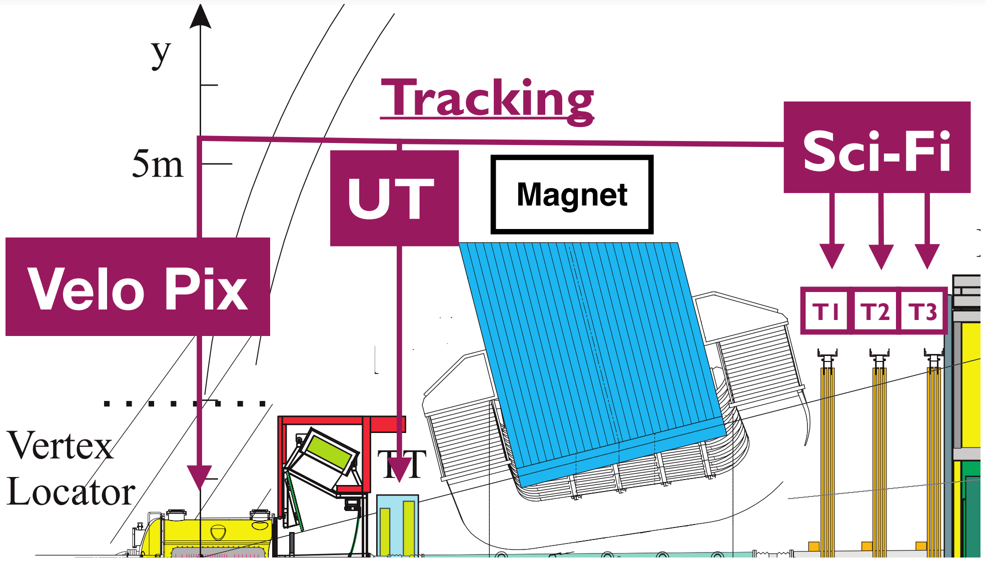
Velo tracking



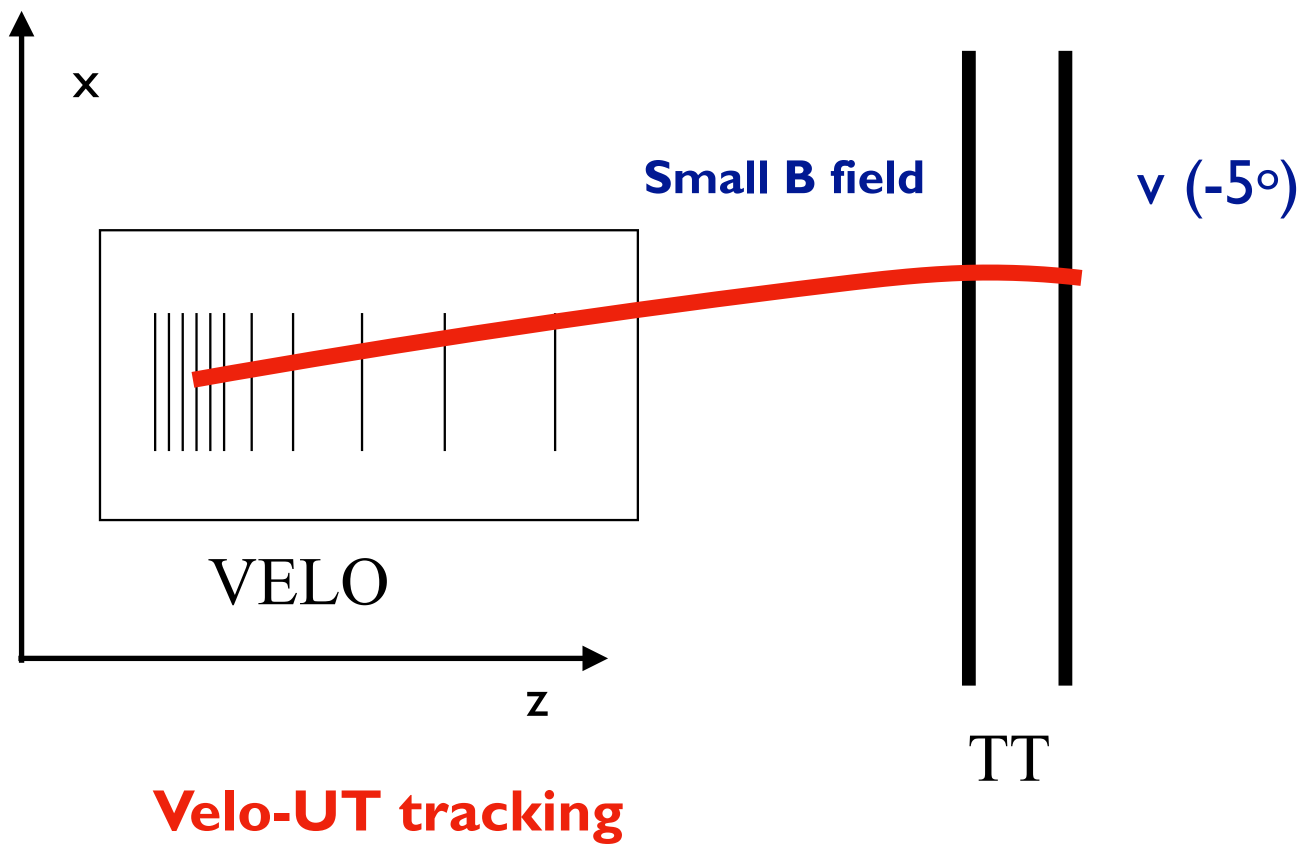
Partial reconstruction sequence : Velo-UT tracking

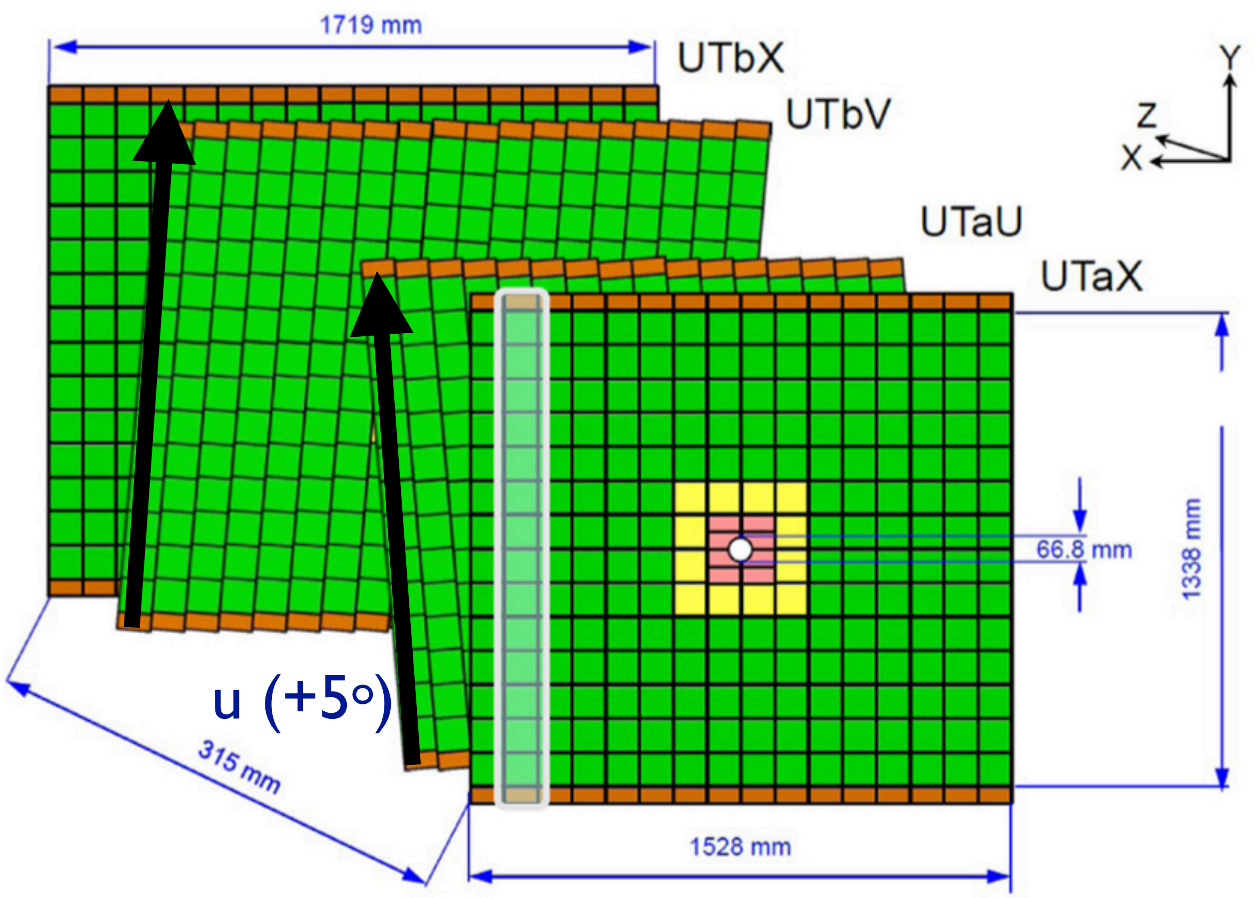
Upstream tracker (UT)

- **Larger acceptance** in central region
- **Reduced thickness**
- **Improved $\sigma_{x-z} \sim 50\ \mu m$**
- **Achievable $\Delta p/p$** ~ 15-30%



LHCB-TDR-015

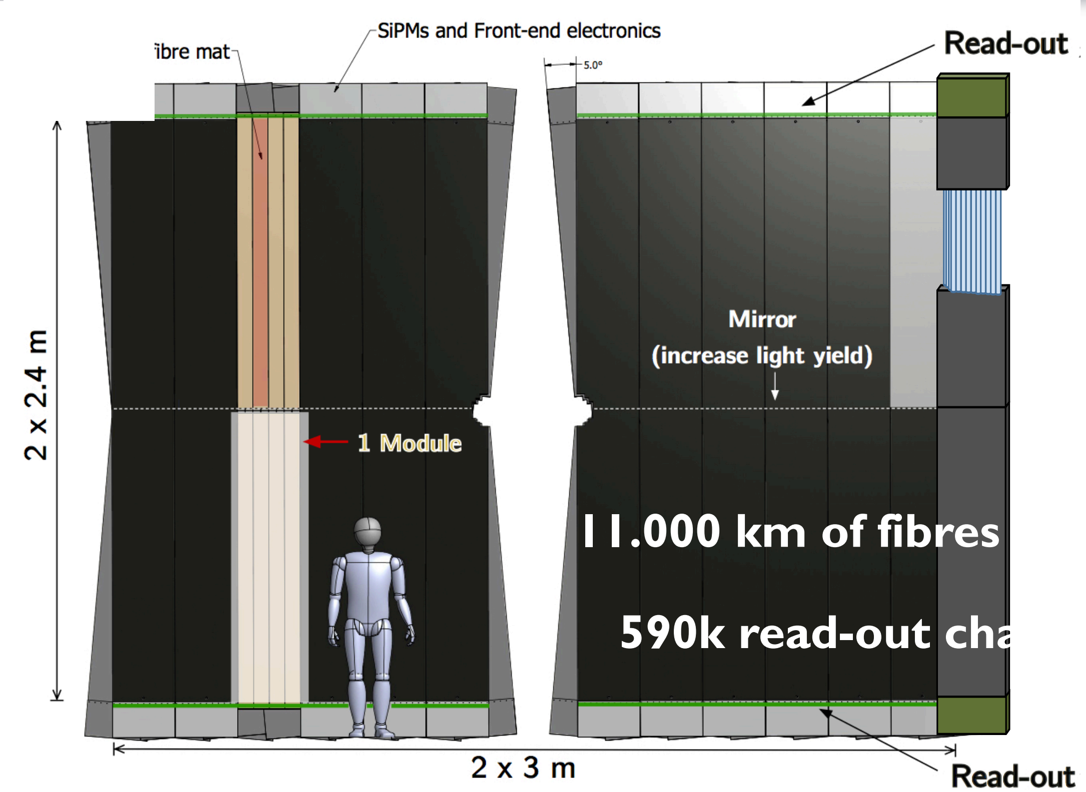




Partial reconstruction sequence : Forward tracking

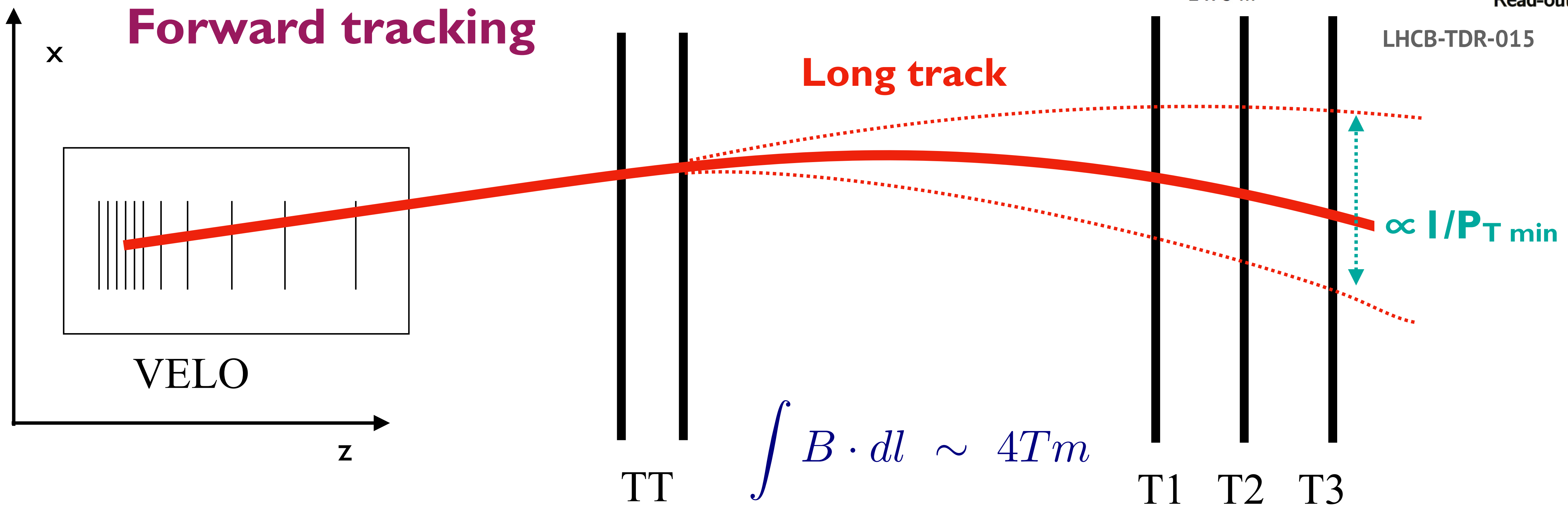
Scintillating Fibre Tracker (Sci-Fi)

- 3 stations x 4 planes (*x*/*u*/*v*/*x*) of 6 stacked 2.4 m long **scintillating fibres (ϕ=**250 μm)
- Read-out by Silicon-Photon multipliers (250 μm channel pitch)
- $\sigma_{x\text{-}z} \sim 100\ \mu m$



LHCB-TDR-015

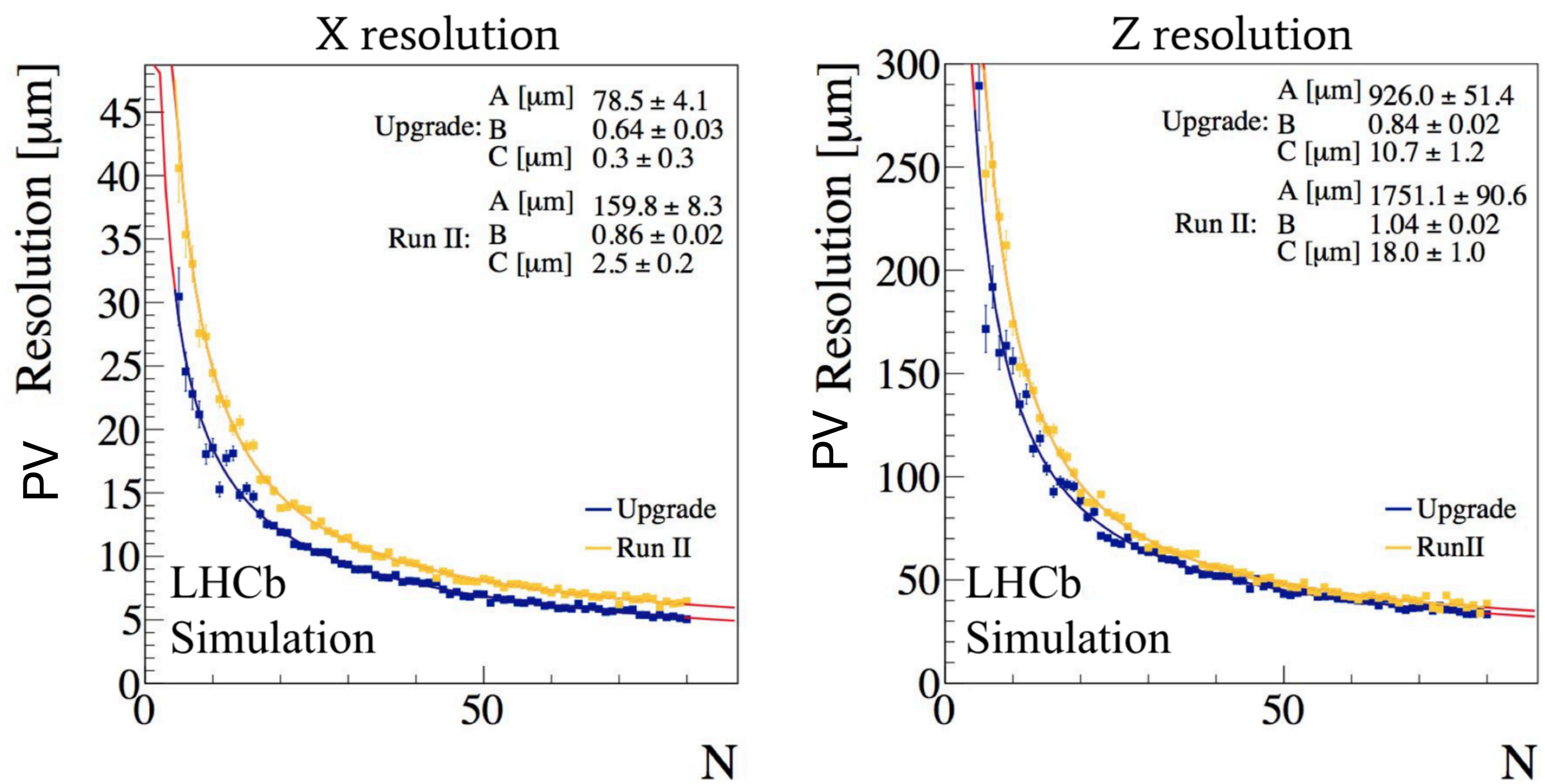
Forward tracking



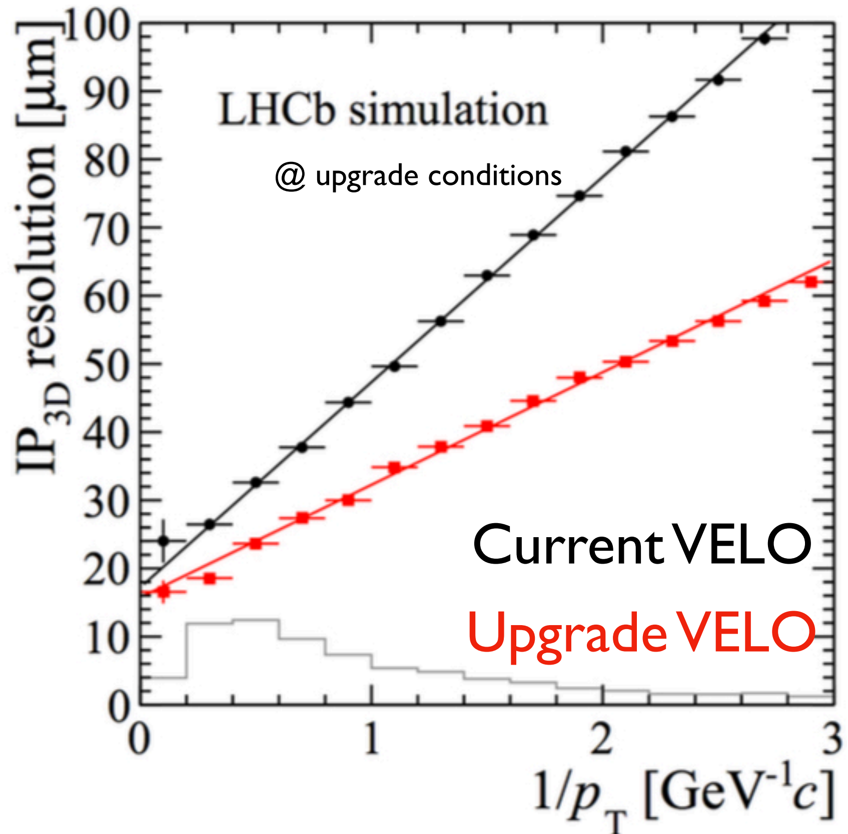
Find matching segments in SciFi according to transverse momentum tolerances

Partial reconstruction sequence : physics handles

[LHCb-PUB-2017-005]

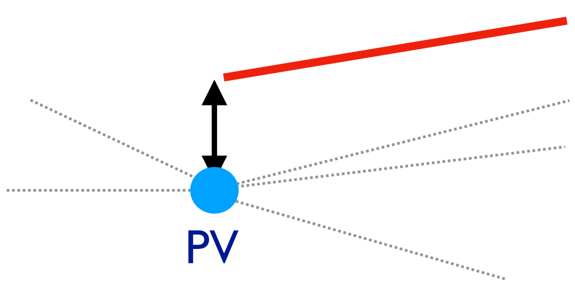


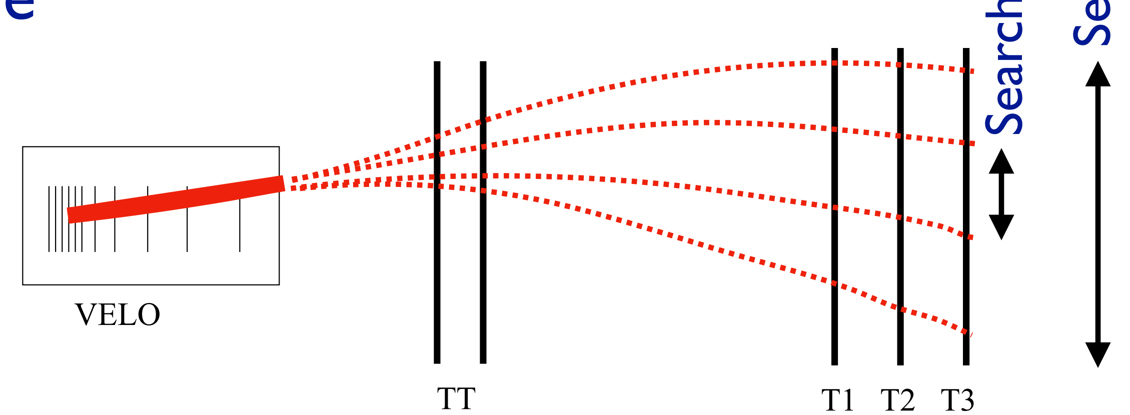
CERN-LHCC-2013-021



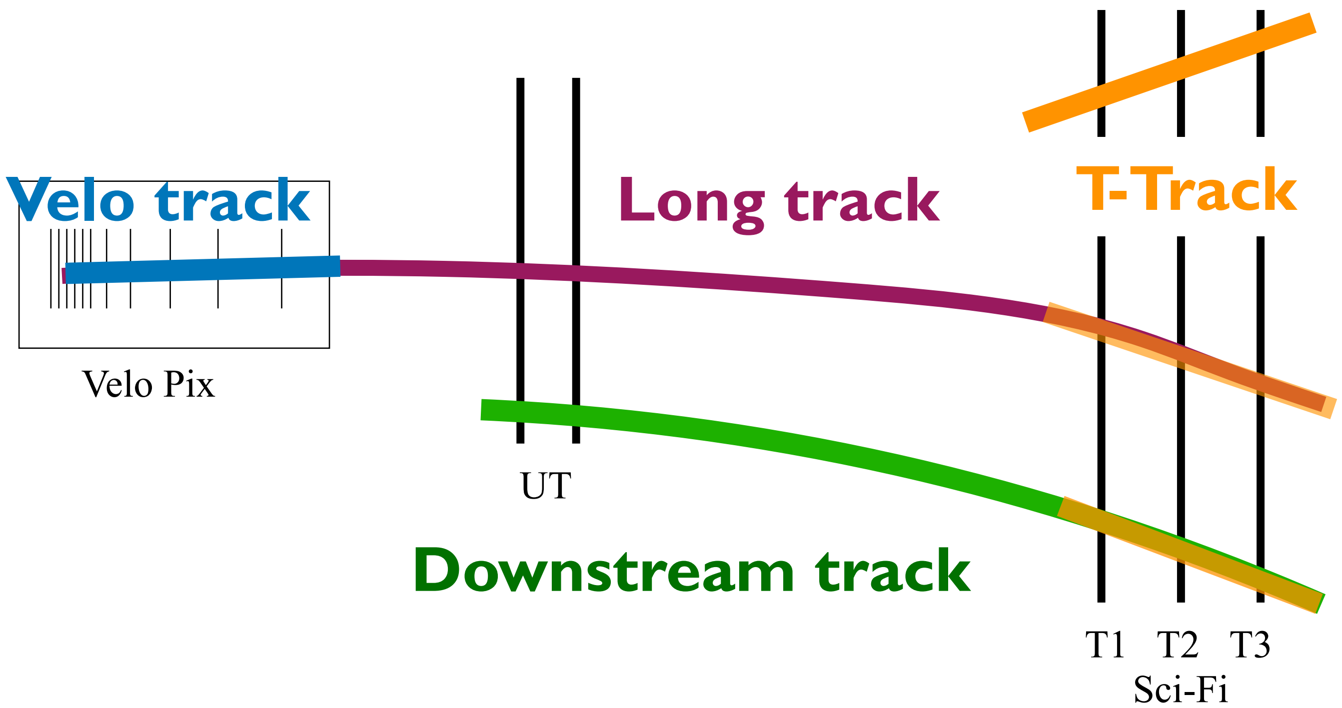
At the upgrade conditions

- ☞ 3-4 times more PVs and 2-3 times higher Track multiplicity.
- ☞ Tuning of reconstruction sequence has impact on the physics capabilities of the detector.
- ☞ To speed-up the partial reconstruction: select only displaced tracks and/or tight the p_T thresholds to reduce the search windows in UT and Sci-Fi.





Best track reconstruction for the LHCb upgrade



In the full event reconstruction

Daughters of b and c hadron decays (long)

T-Tracks seed the reconstruction of

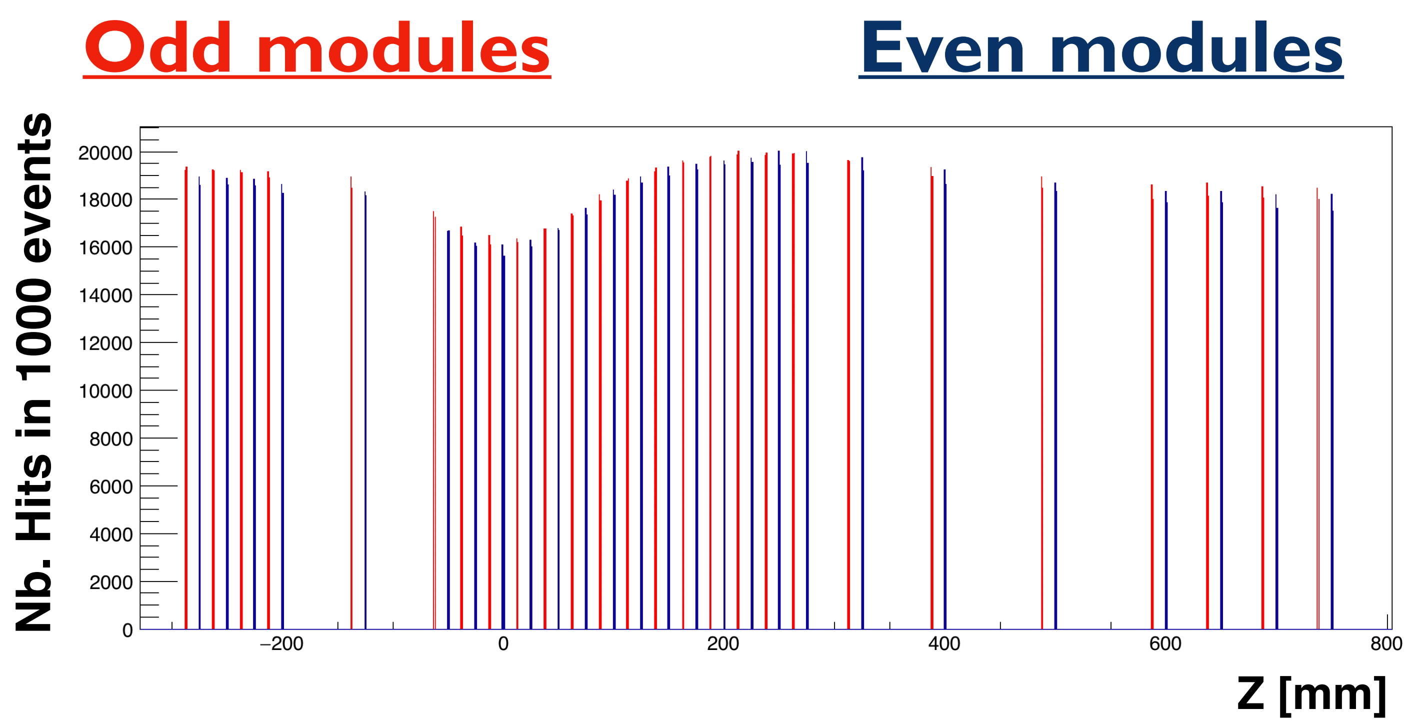
Daughters of long-lived particles such as K_s and Λ^0 (downstream)

- ☞Velo tracks reprocessed by forward tracking using very low p_T thresholds.
- ☞T-Tracks are searched by the seeding algorithm (see poster session).
- ☞Long tracks are also searched matching T-Tracks with Velo Tracks.
- ☞Downstream tracks are found using the T-Tracks segments.

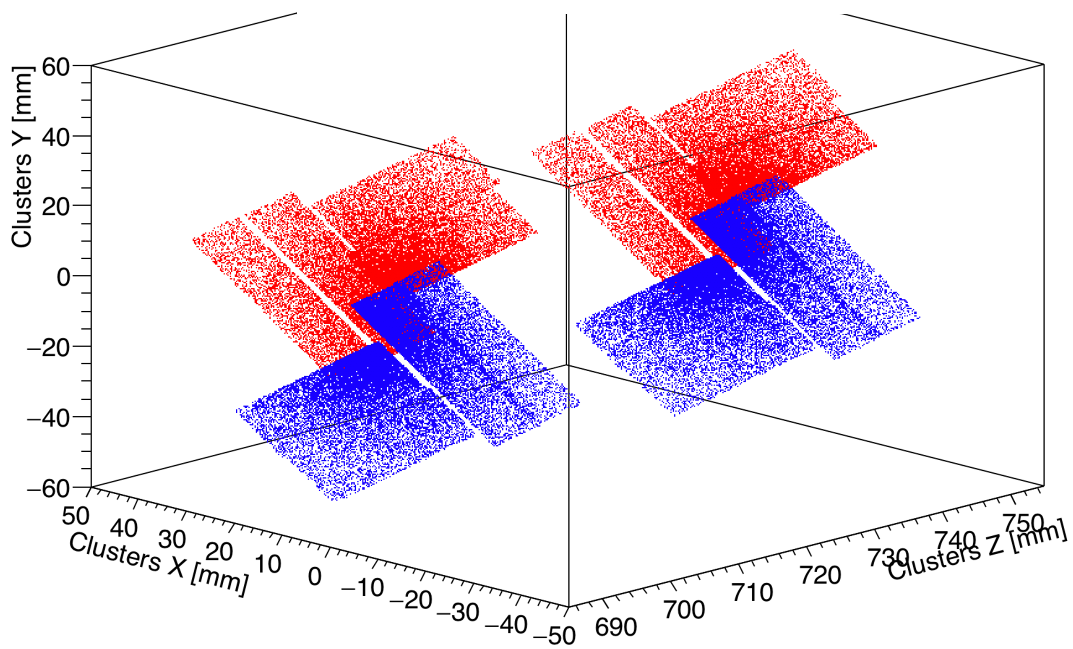
Velo reconstruction

Velo track reconstruction is one of the most time consuming algorithm of the sequence.

Factor 3 speed-up w.r.t. the version used in [LHCb-PUB-2017-005] with tiny loss for HLT1 physics capabilities



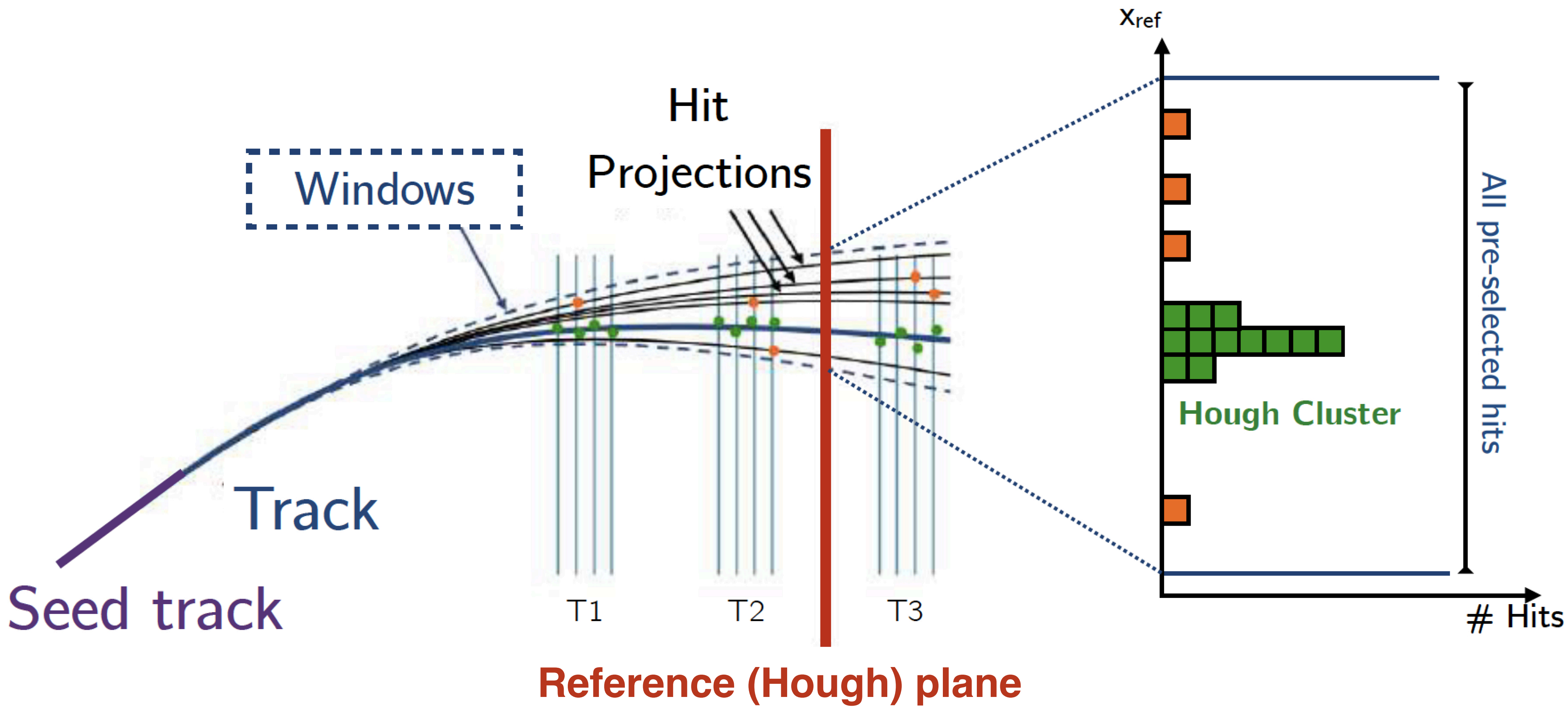
Last 4 modules of the Velo



Selected doublets from "same-side" modules are extrapolated to the other modules.

- Sorting hits by ϕ in each module and using ϕ-search windows instead of x, y.
- Early checks and kill of combinatoric.
- Lighter (in memory) hit objects used for tracking.
- Redesign of the tracking logic [first forward tracks, then backward ones].

Forward Tracking



- Hits are pre-selected according to the min p_T requirement
- The lower the min p_T requirement, the wider the window, the larger the amount of hits used for each seed track, the slower the algorithm.
- For each seed track, all p_T compatible hits are projected to a reference place and a cluster search is performed to find the matching SciFi hits.

Outline

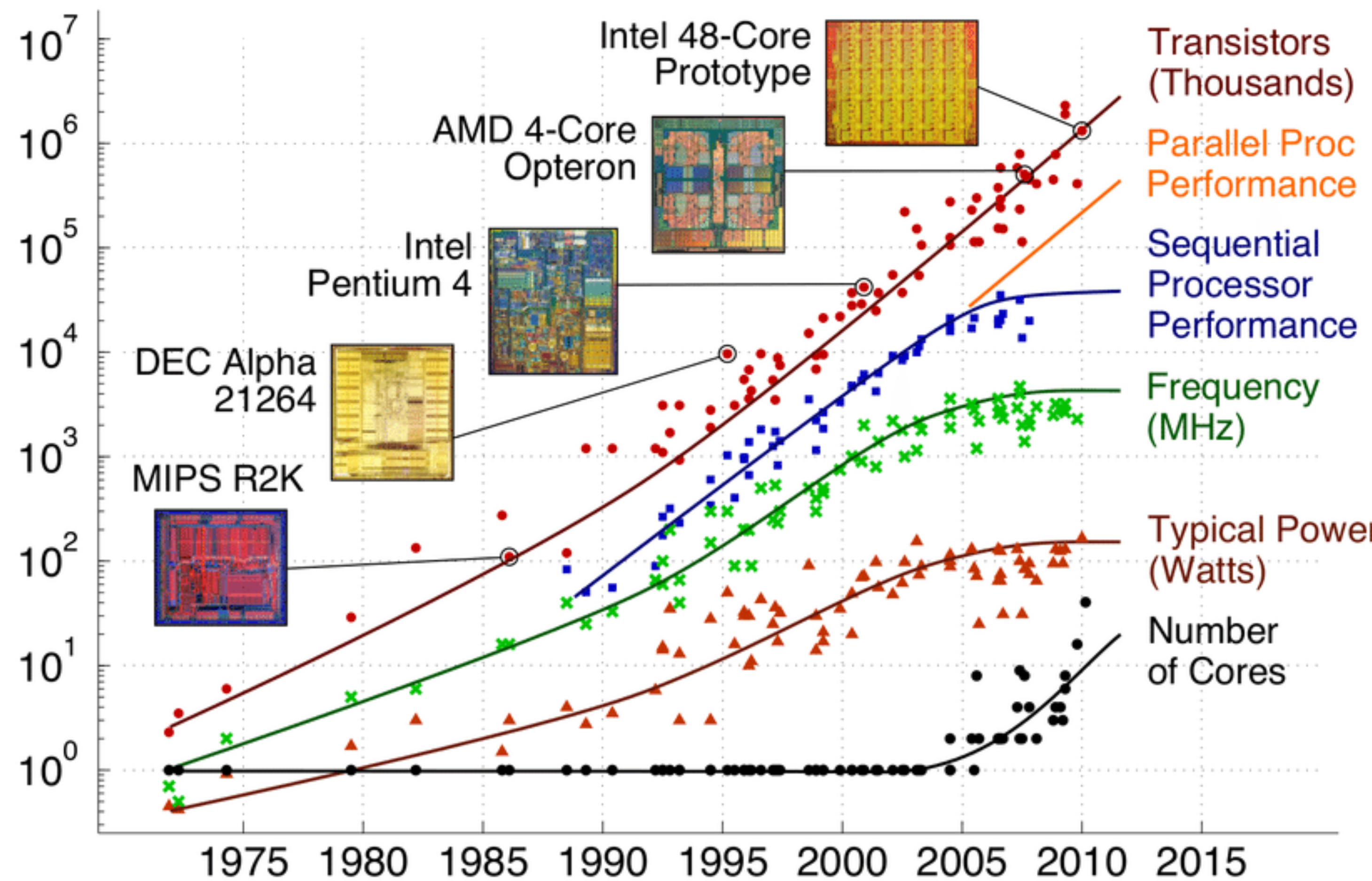
1. LHCb upgrade and the trigger strategy
2. Track reconstruction for the LHCb upgrade at collision rate
3. **High-Throughput computing for the LHCb upgrade and current status**
4. Conclusions and prospects

High-throughput software for the LHCb upgrade

- **Evolution trend of faster single-threaded CPU performance broken 10 years ago.**
- **Increase of CPU cores** and more execution units.

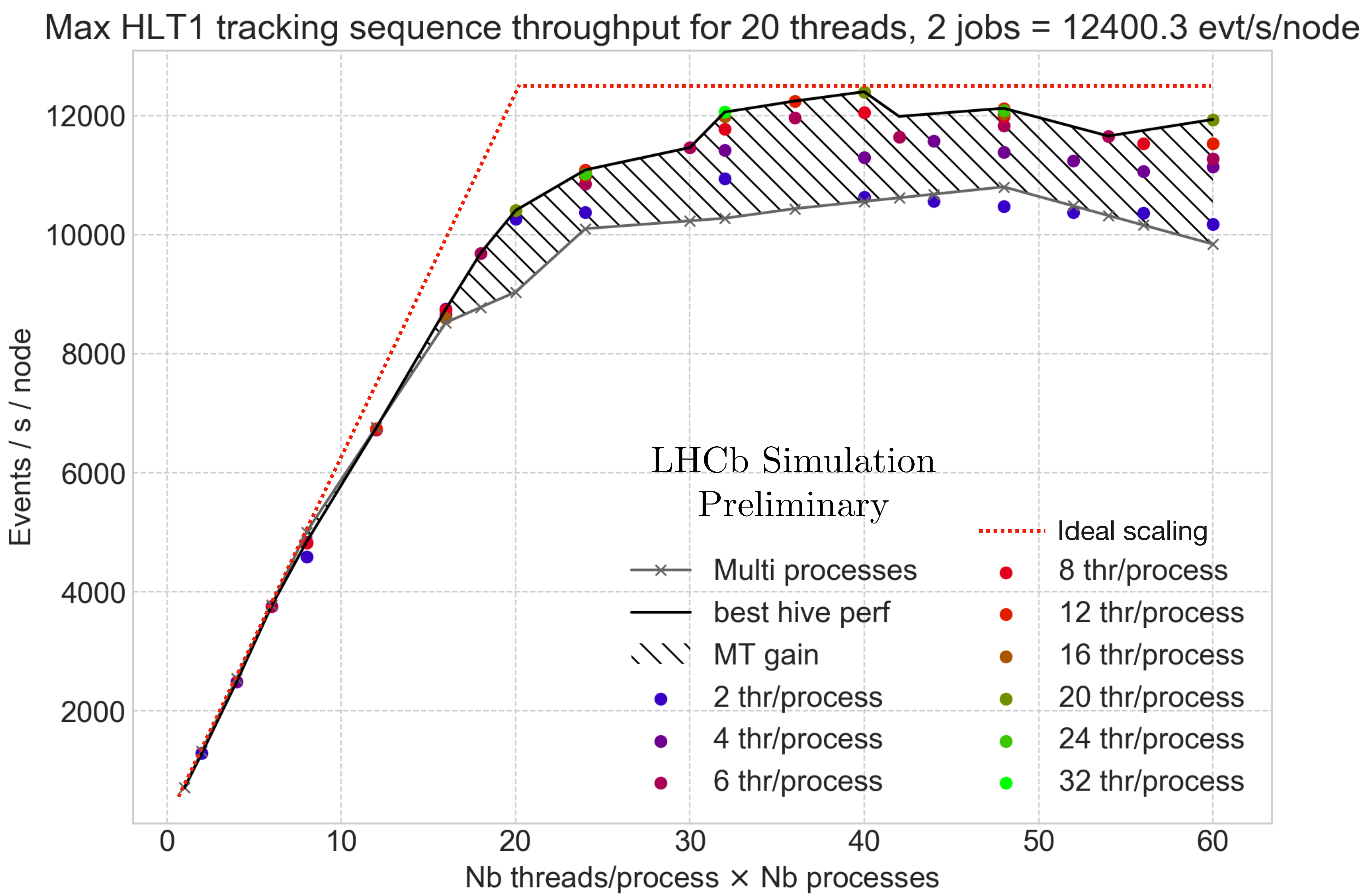
Old framework: sequential event data processing model.

- Weak scalability in RAM usage
- Inefficient disk/network I/O



New concurrent, task-based model developed for the LHCb upgrade: multi-thread framework.

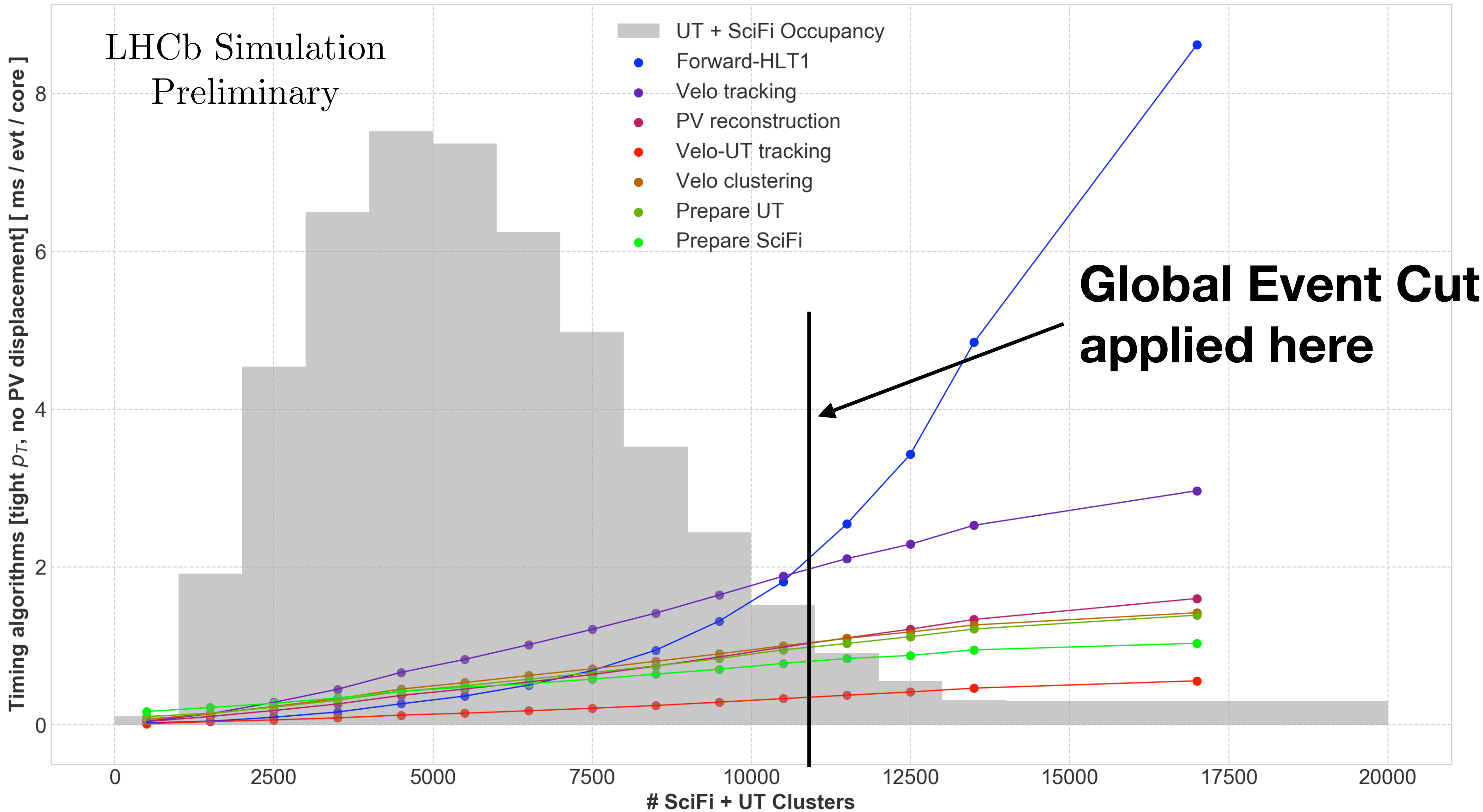
High-throughput software for the LHCb upgrade



Throughput of the HLT1 upgrade reconstruction sequence on a 20 physical core machine with 40 hyper threaded cores.

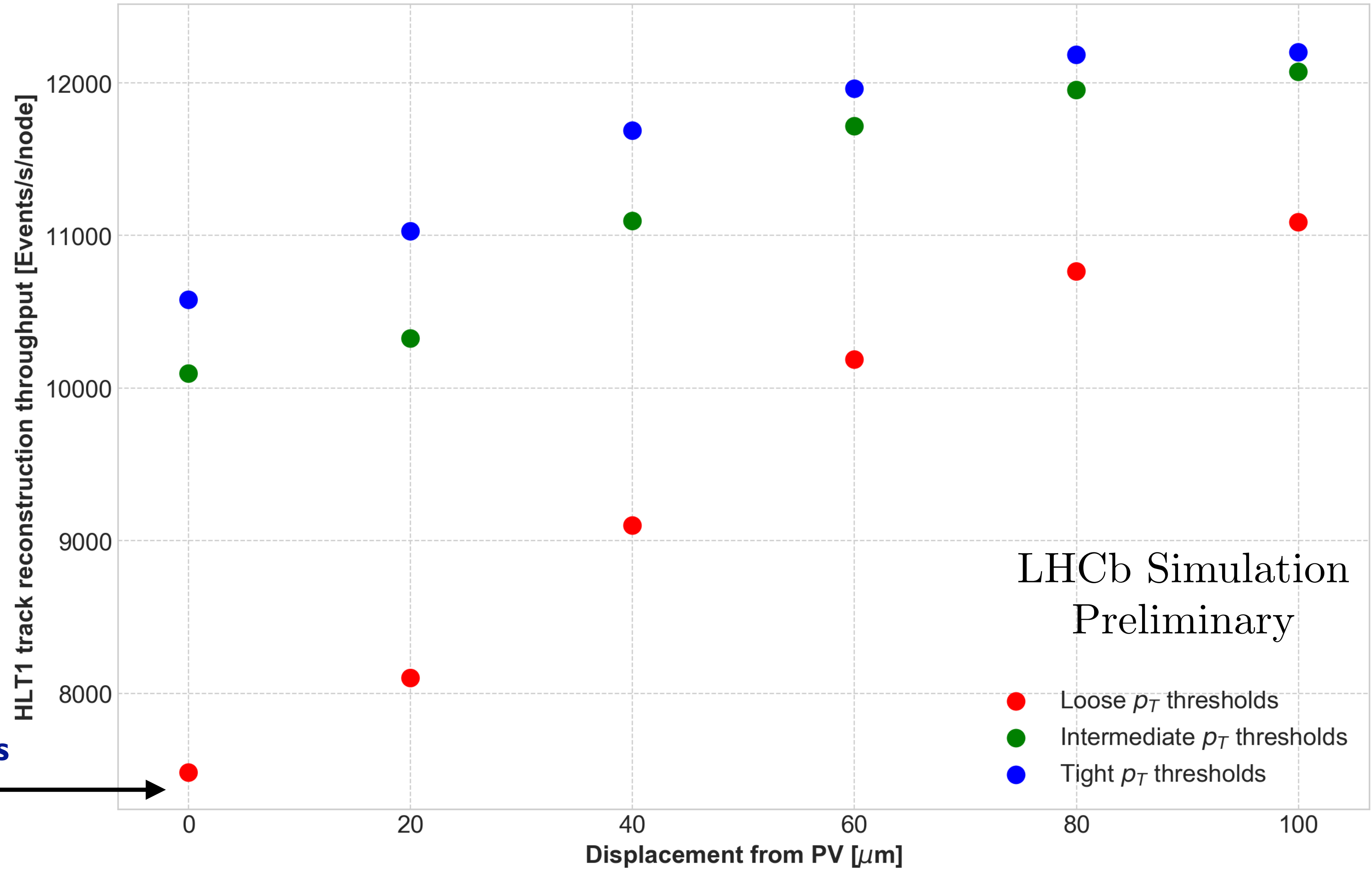
The multi-thread framework brings a 20% extra speed-up in the reconstruction sequence.

Throughput dependence on detector occupancy



- Almost all algorithms scales linearly with the occupancy.
- Exception done for the forward tracking, which is currently implementing a tracking strategy oriented for HLT2 purposes.

Throughput dependence on tracking scenario



Same working point as
[LHCb-PUB-2017-005]
(used to be 3500 evt/s/node)

- The displaced-tight p_T track reconstruction scenario is almost fitting in the 30 MHz computing budget and it fulfills the B physics scientific program.
- Prompt physics and multi-hadron final state B physics will suffer in tightest scenario.
- Factor 2 in throughput gained since biannual upgrade review document thanks to the optimization of the various algorithms.

Conclusions and prospects

- **LHCb will perform track reconstruction at collision rate in Run III**
  - Taking benefit of the real-time alignment and calibration developed for Run II.
  - Selectively persisting subset of event to save disk space.
- **The trigger strategy for the LHCb upgrade is challenging**
  - Multithread framework is in place and works well.
  - Work is on-going for algorithm optimization, a factor 2 speed-up in less than 6 months has been achieved.
  - In depth profiling of the algorithms shows that there are large margins to achieve additional speed-ups.
- **Performance studies on decay modes for the HLT1 selections well under way.**
- We are also currently exploring the usage of architectures different from x86 and accelerators (GPUs).
- **Lots to come in the next couple of years.**

Backup

The fast reconstruction sequence : data preparation

Velo Raw data
↓
Velo Clustering
↓
Velo Clusters

UT raw Clusters
↓
UT decoding
↓
UT Clusters
↓
StoreUTHit
↓
UT Hit

FT Raw Clusters
↓
FT decoding
↓
FT Clusters
↓
StoreFTHit
↓
FT Hit

Preparing the data is already a challenge!

Data preparation: convert binary format to geometrical position and sort hits as expected by pattern recognition algorithms.

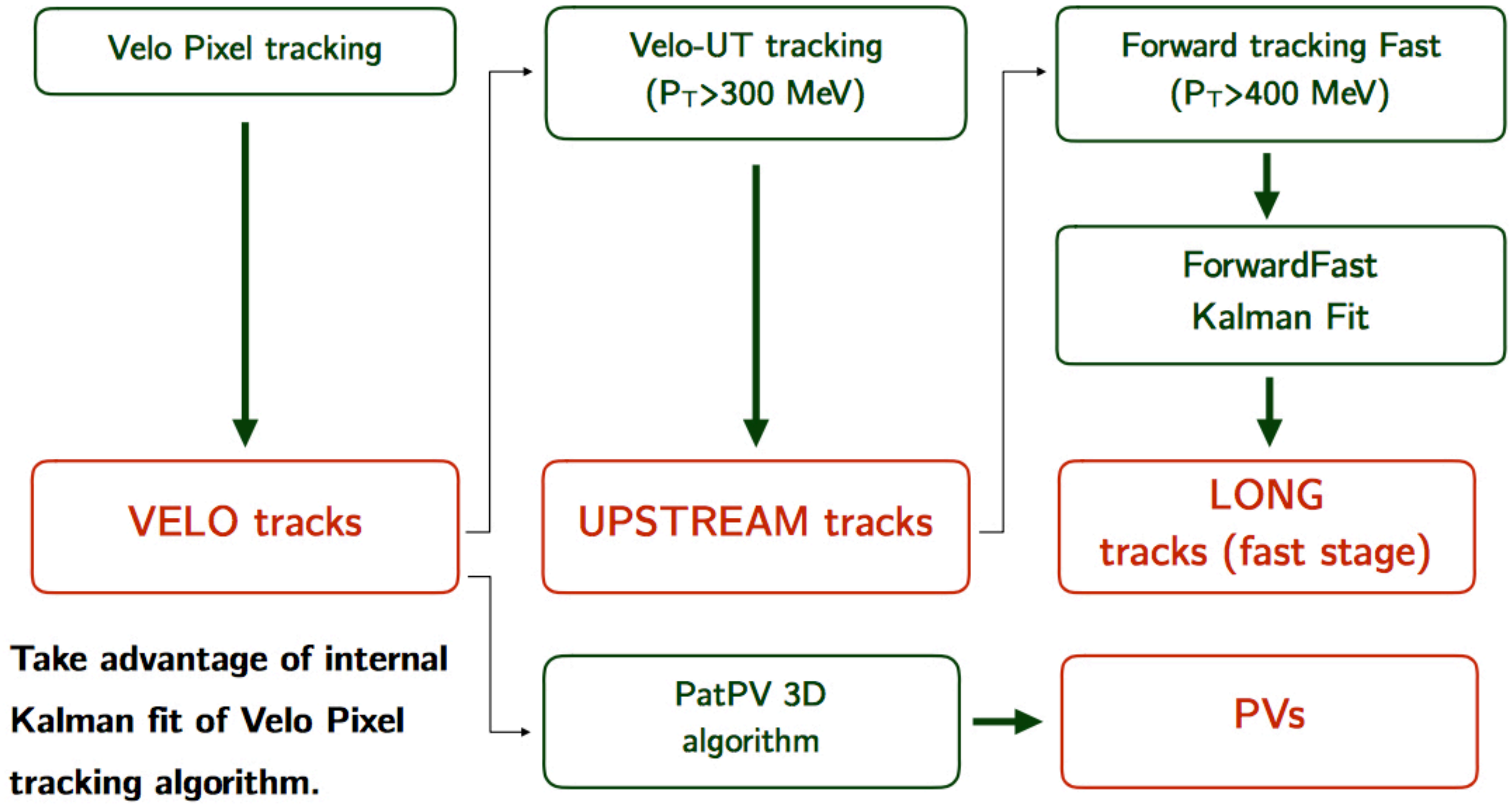
Velo: some work is already done in read-out, but need to do clustering on software

UT: clustering will be done offline

FT: clustering and sorting by x value already done in FPGA.

Tracking for the LHCb upgrade (fast stage)

The HLT-I like sequence for the LHCb upgrade



Fast stage aims at reconstructing PVs, VELO and long (high p_T) tracks to allow trigger selection and input for online alignment and calibration tasks.

We can tight the p-kick method threholds in velo-UT and forward as well as process only those Velo tracks being displaced from the reconstructed PVs

Extra speed-up from physics handles

Different tracking scenarios for different throughput outcome

Loose tracking

- ☞ **Velo-UT $p_T > 300$ MeV**
- ☞ **Forward tracking $p_T > 400$ MeV**

Intermediate tracking

- ☞ **Velo-UT $p_T > 600$ MeV**
- ☞ **Forward tracking $p_T > 800$ MeV**

Tight tracking

- ☞ **Velo-UT $p_T > 800$ MeV**
- ☞ **Forward tracking $p_T > 1000$ MeV**

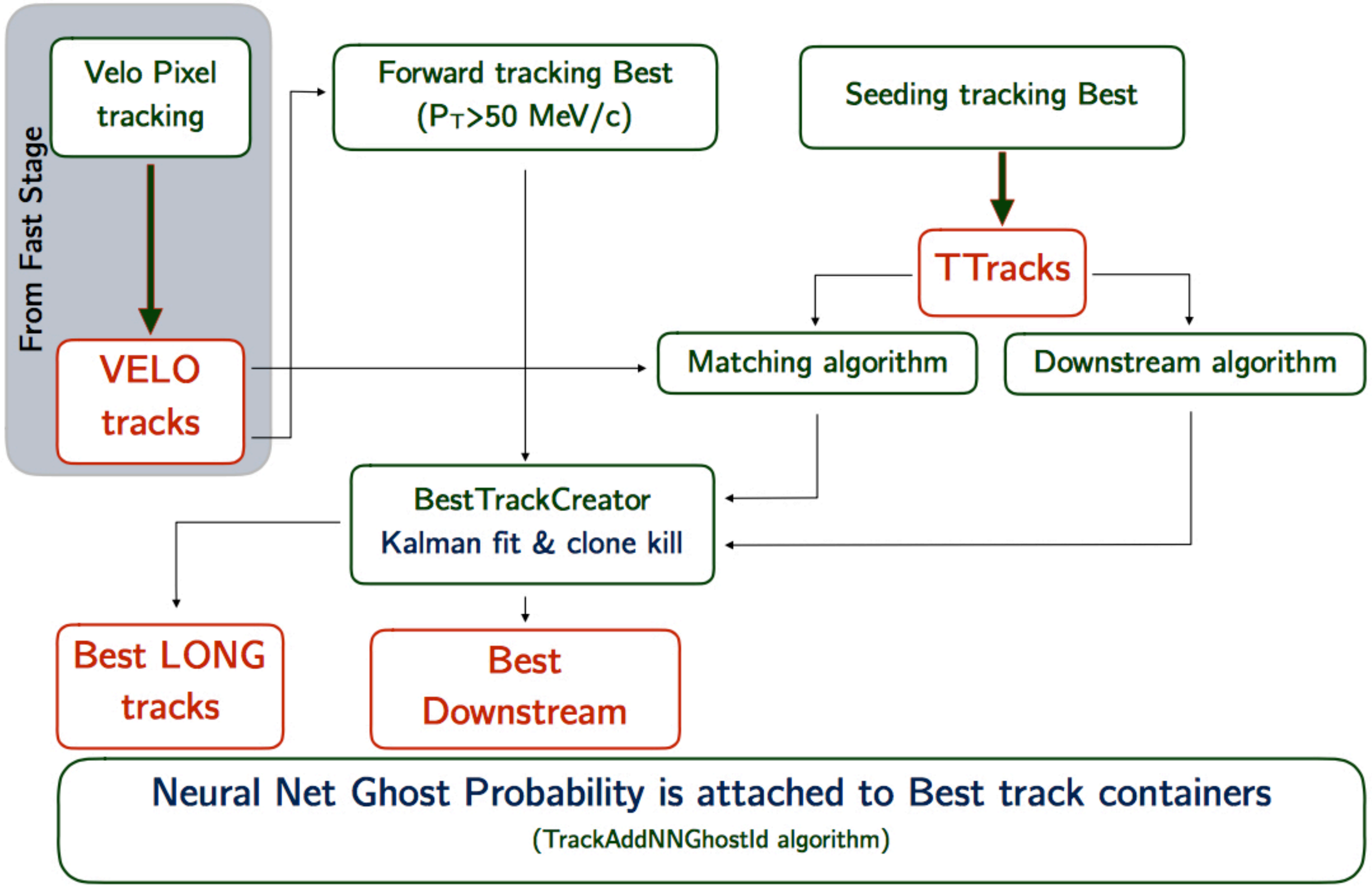
IP Cut to filter Velo tracks

- ☞ **No IPCut**
- ☞ **IP > 20 μm**
- ☞ **IP > 40 μm**
- ☞ **IP > 60 μm**
- ☞ **IP > 80 μm**
- ☞ **IP > 100 μm**

Each configuration has its own drawback in terms of physics.

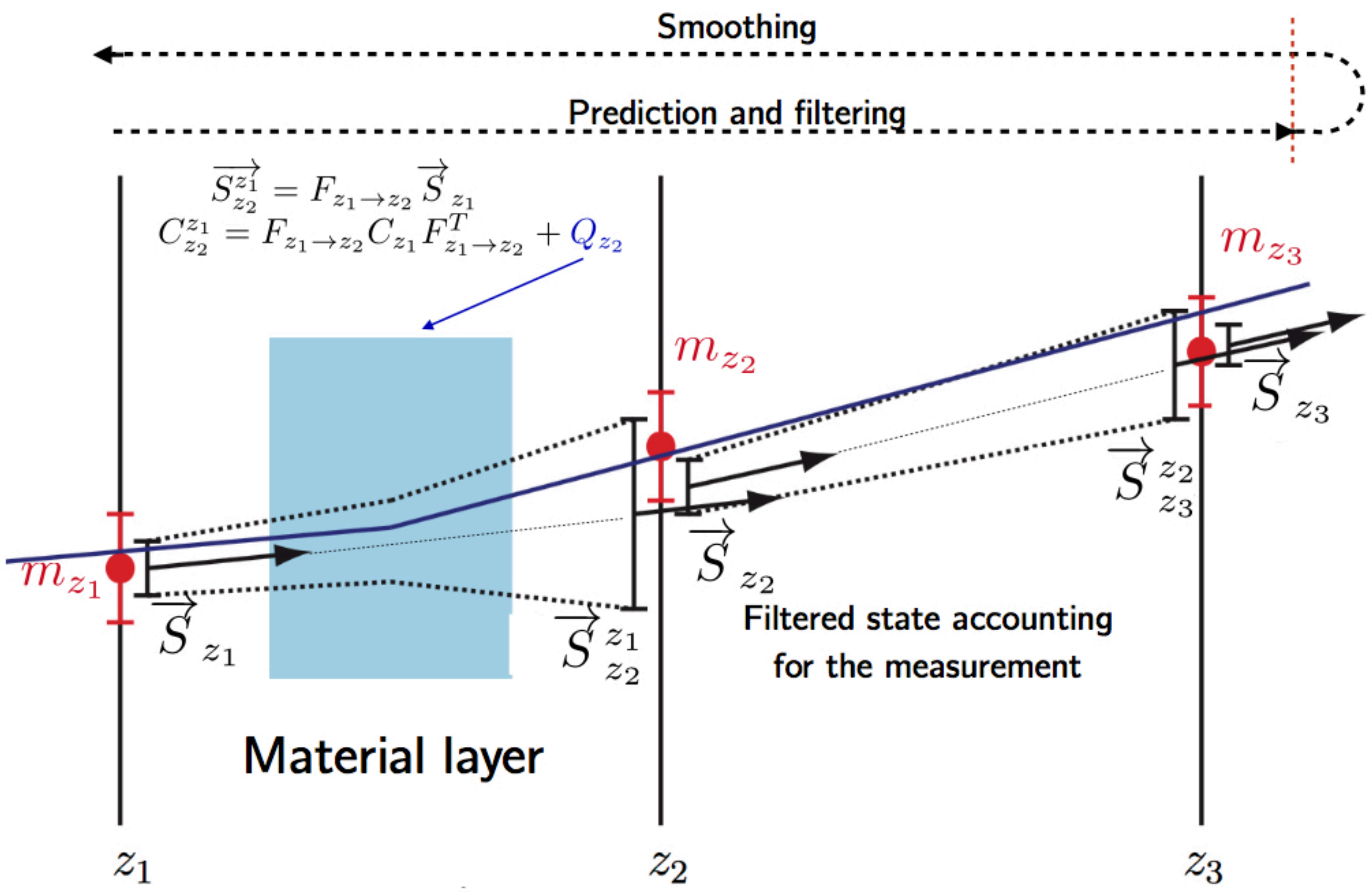
Tracking for the LHCb upgrade (*best* stage)

The HLT-II like sequence (*best*) for the LHCb upgrade



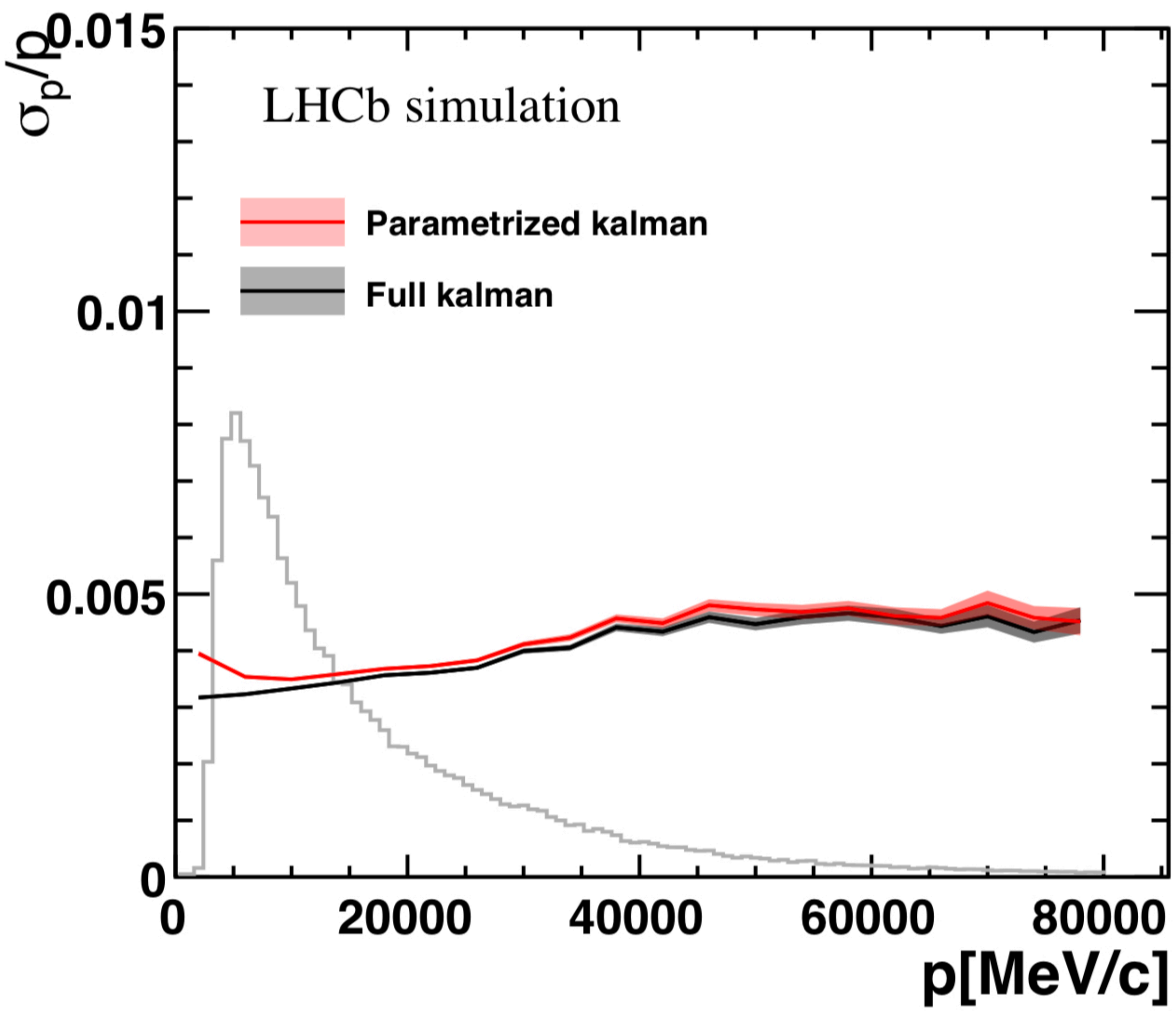
- *Best* stage aims at reconstructing as efficient and fast as possible all tracks in the event.
- CPU resources are limited by available budget.
- Make all the workflow as fast as possible.
- Extra tasks which can be useful for physics can be executed only if extra timing budget is available.

Kalman Filter



- **Prediction**: propagate a track state to another track state
- **Filtering**: compare propagated track state to actual measurement. Minimisation of χ^2 based on residual.
- Evaluate best estimate of new track state (also reject/accept measurement)
- Iterate over all measurements
- **Smoothing**: perform previous steps in the opposite direction using the prediction and filtering
- **Material interactions accounted for enlarging the errors when propagating states**

Track fit : Parameterized Kalman Fit



- **Much faster than the default Kalman Filter** using simple parameterization to propagate through magnetic field and material.
- Moderate loss in precision.

Limits and upgrade framework

Extrapolation at TDR time (2007 to 2010)

The trigger TDR performance extrapolation determined the time per event for a single process to be $t = 13$ ms total. The extrapolation used was:

$$t = \frac{g^y \times n \times N}{R} \tag{3}$$

Where g is the performance growth per year at equal cost, $y = 9$ is the number of years before purchase, $n \times N$ is the total number of processes per node times the number of nodes and R is the bunch crossing rate of 30MHz . For the extrapolation n was assumed to be equal to the number of virtual cores, 24 in the case of the X5650 candidate node, and $N = 1000$ was the number of X5650 nodes purchasable with the 2.8MCHF budget.

The performance growth per year at equal cost was estimated in the trigger TDR to be $g = 1.365$.

Revisited extrapolation using machines available in 2010 to 2016 g = 1.1

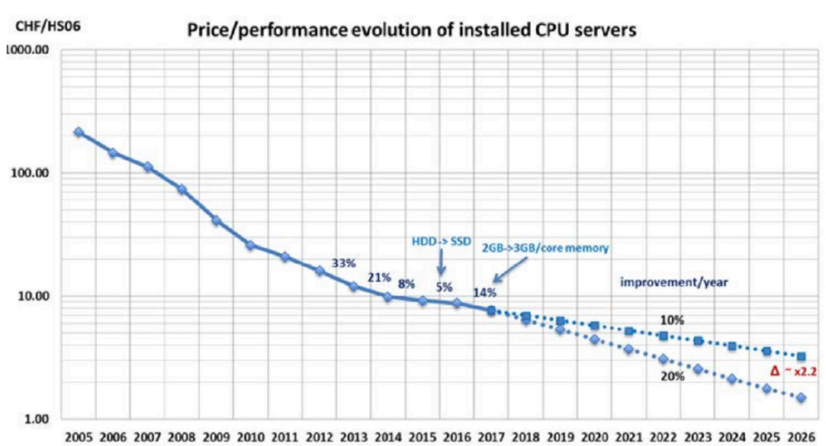


Figure 5: The evolution of the cost of one HEPSPEC06 unit in CHF as a function of the year of procurement in purchasing procedures conducted by the CERN IT department. Two extrapolations are shown assuming a 10% and 20% growth rate. Source: Bernd Panzer CERN/IT

Also different costs w.r.t. TDR extrapolation

Current throughput in fast sequence needs to be speed up by at least a factor 10!!!

Recent years evolution of upgrade track reconstruction

From [LHCb-PUB-2017-005] :

☞We realized we had to gain at least a factor 10 in tracking to be able to fit in the 30 MHz.

☞The computing resources available budget is 3 M CHF dedicated for the EFF farm.
(test node used for the presented results is ~3000 CHF)

☞We managed to speed-up the reconstruction of a factor 2, a factor 3.5 in the mitigation scenario. (displaced tracks and tight p_T)

☞We need now another factor 3:

- ☞We can get a factor 3 if we re-use the full LHCb Run II farm and we will get a hardware at same cost in 2020 with a speed-up of a factor 1.5.

We have a long road ahead us, but we are moving big steps forward

Trigger at collision rate starts to become feasible.

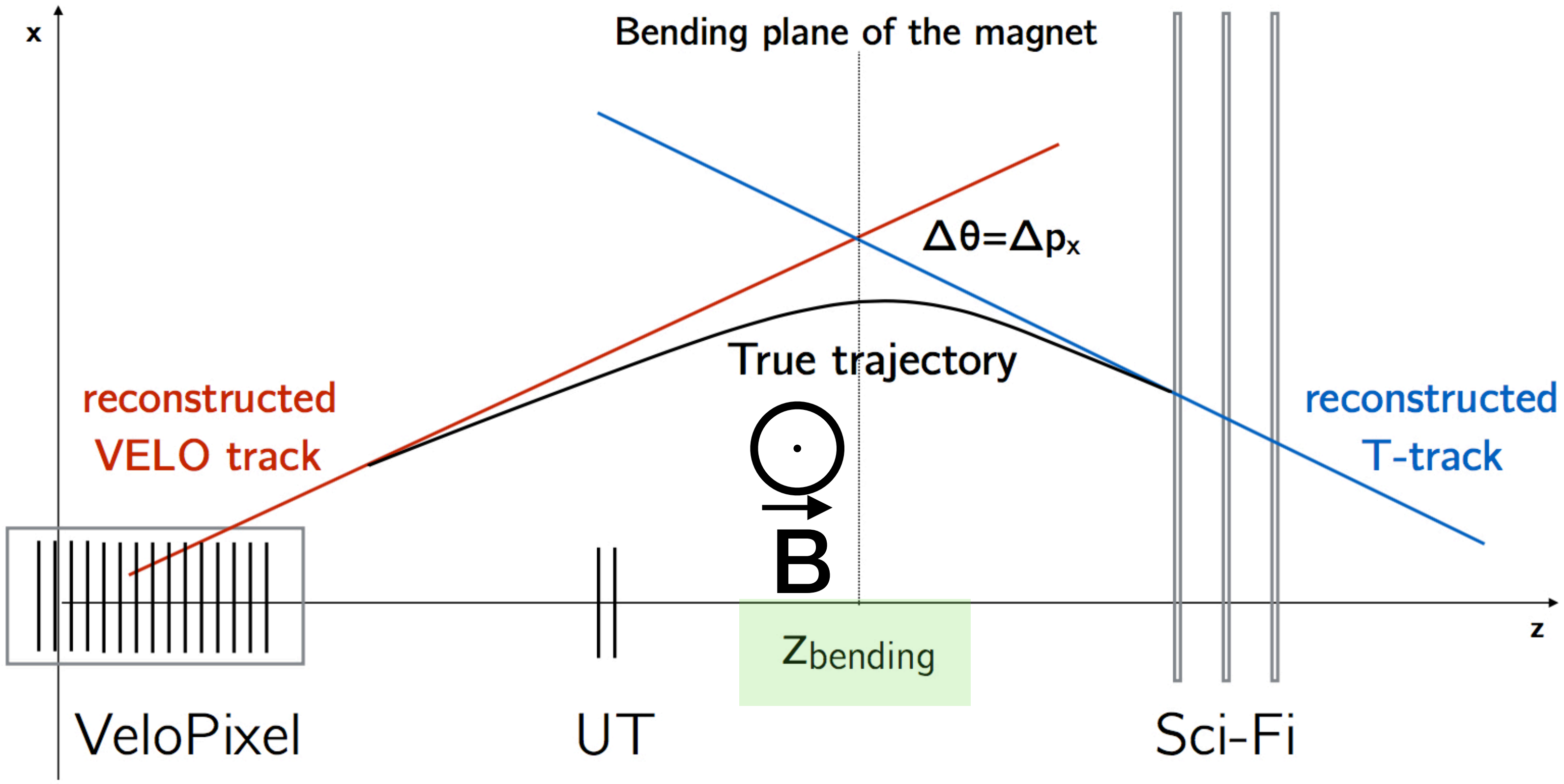
Recent years evolution of upgrade track reconstruction

HLT1 algorithm	[LHCb-PUB-2017-005]	Computing TDR [preliminary]	Speed-up
Velo (Clustering + Tracking)	10 kHz	30 kHz	$\times 3$
Prepare UT data	43 kHz	88 kHz	$\times 2$
Prepare SciFi data	22 kHz	88 kHz	$\times 4$
Velo-UT tracking	66 kHz	146 kHz	$\times$ 2.2
PV finding	32 kHz	91 kHz	$\times 2.8$
Forward tracking	15 kHz	19 kHz	$\times 1.2$
Throughput	3500 kHz	7500 kHz	$\times 2.14$

Tracks behaviour in LHCb and pattern recognition

p_T-kick method is heavily used in LHCb for patter recognition (PR).

(typically B_x and B_z assumed to be negligible at any z in pattern recognition)



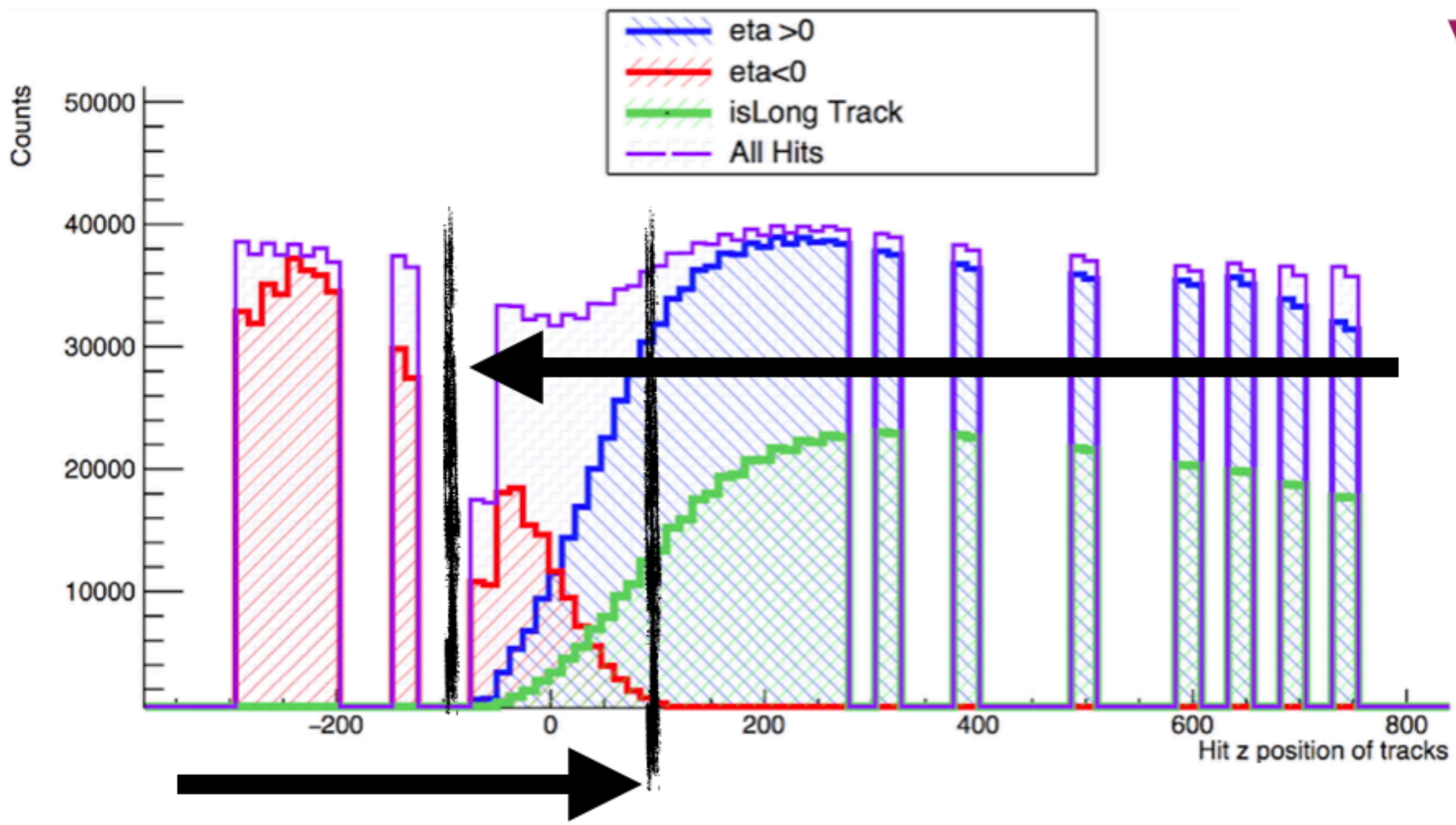
Example of the p_T-kick used in the matching algorithm

$$\frac{q}{p} = \frac{1}{\int \left| d\vec{l} \times \vec{B} \right|_x} \cdot \left(\frac{t_{x,f}}{\sqrt{1 + t_{x,f}^2 + t_{y,f}^2}} - \frac{t_{x,i}}{\sqrt{1 + t_{x,i}^2 + t_{y,i}^2}} \right)$$

- ☞Used to estimate q/p.
- ☞Used to open search windows according to momentum requirements.
- ☞Used to quantify matching quality of reconstructed segments.

Pixel Tracking : search pair and remaining modules dr/dz

Indeed, looking to the z of the hits for negative **η>0** and **η<0**



We introduce here the

1) "Forward search":

Requiring also dr/dz > 0 and breaking hit search at ***minZForwardSearch [-100 mm]***

2) "Backward search":

Requiring dr/dz < 0 and breaking at ***maxZBackwardSearch [100 mm]***

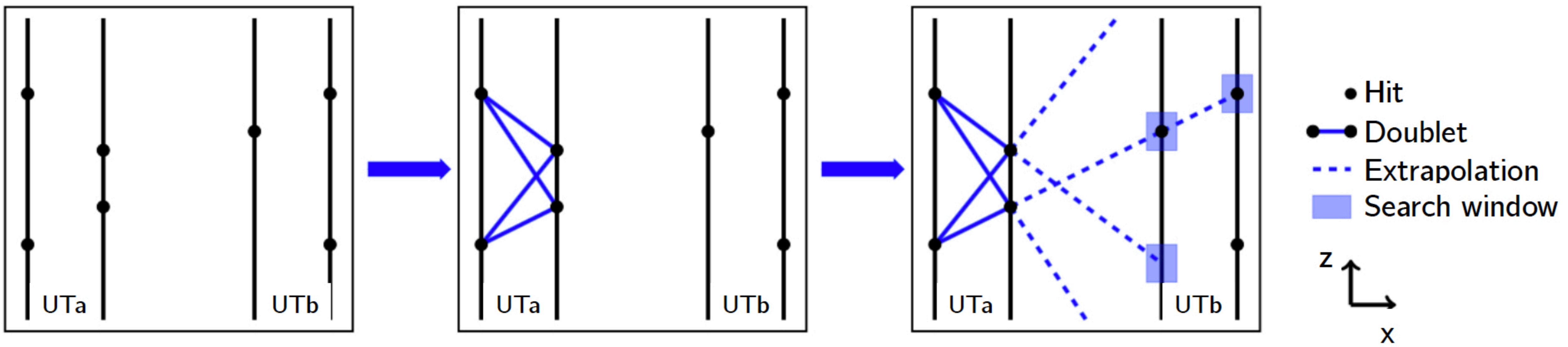
Algorithm supports different configurations

PrPixelTracking.AlgoConfig

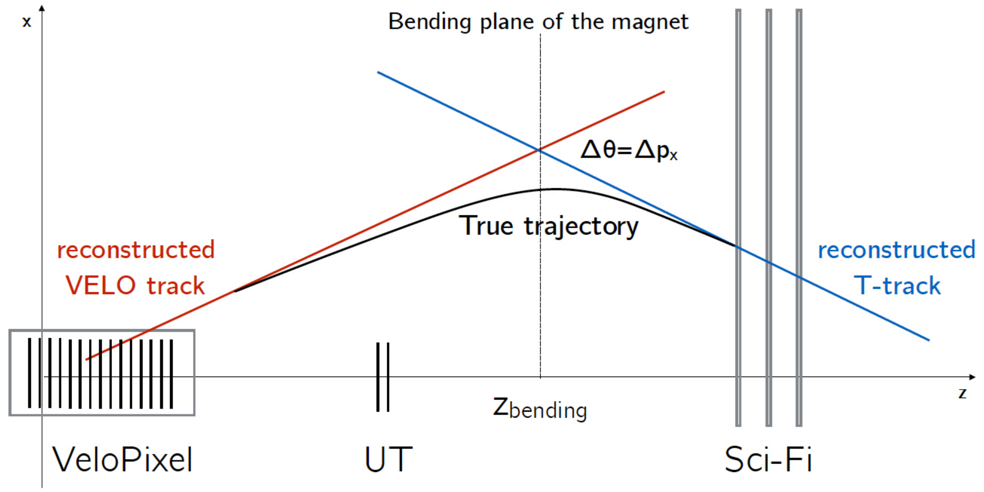
"Default", "OnlyForward", "OnlyBackward", "ForwardThenBackward", "BackwardThenForward"

1) Create less combinatorics due to ***dr/dz*** requirement along the way.
2) In the shared region [-100,100] *mm* , searching first in forward and then in backward implies you will touch (and skip) flagged hits.

Velo-UT tracking



- **Propagate VELO track to UT correcting for integrated B field**
- Search for doublets and propagate
- Find 3-4 hits matching predictions using also y-z plane information
- Estimation of q/p



$$\frac{q}{p} = \frac{1}{\int \left| d\vec{l} \times \vec{B} \right|_x} \cdot \left(\frac{t_{x,f}}{\sqrt{1 + t_{x,f}^2 + t_{y,f}^2}} - \frac{t_{x,i}}{\sqrt{1 + t_{x,i}^2 + t_{y,i}^2}} \right)$$

Forward Tracking algorithm

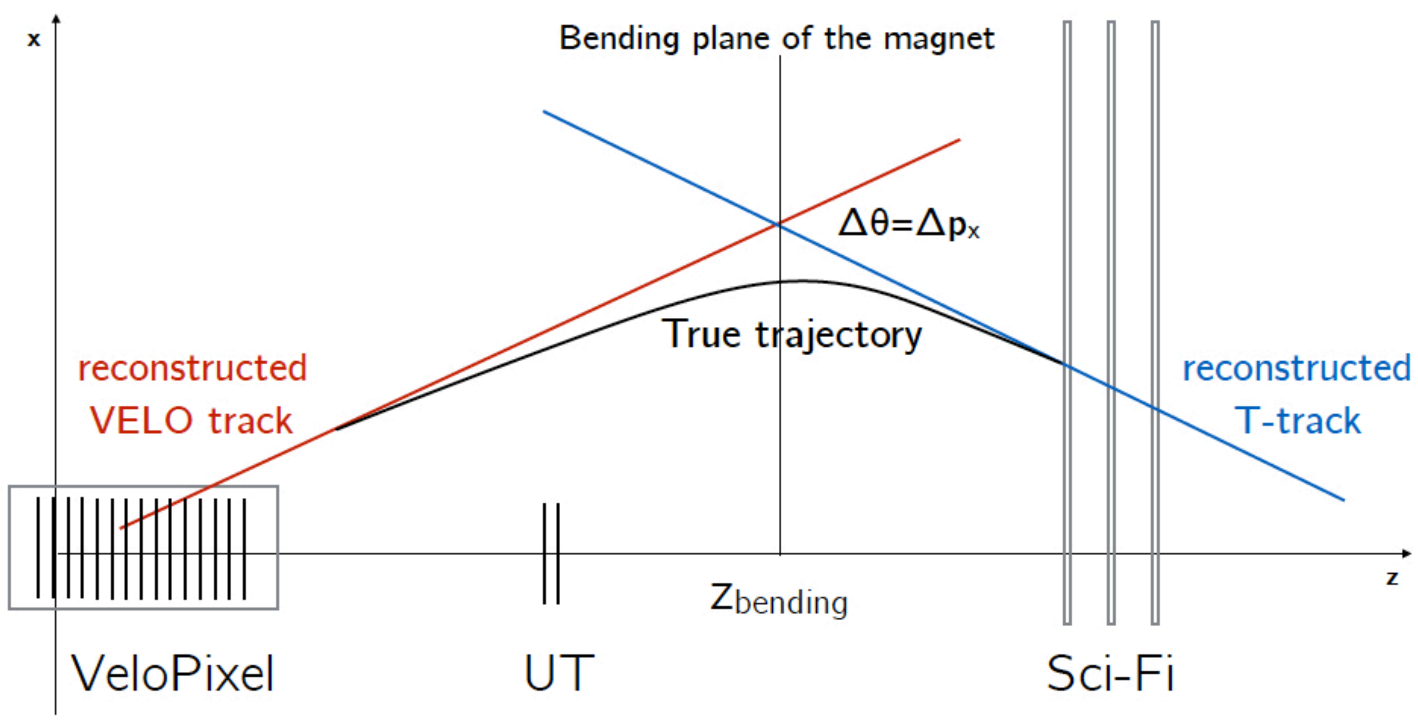
Seed track: VELO or VELO-UT depending if fast (HLT 1) or best (HLT2) sequence

Search windows based on p-kick method

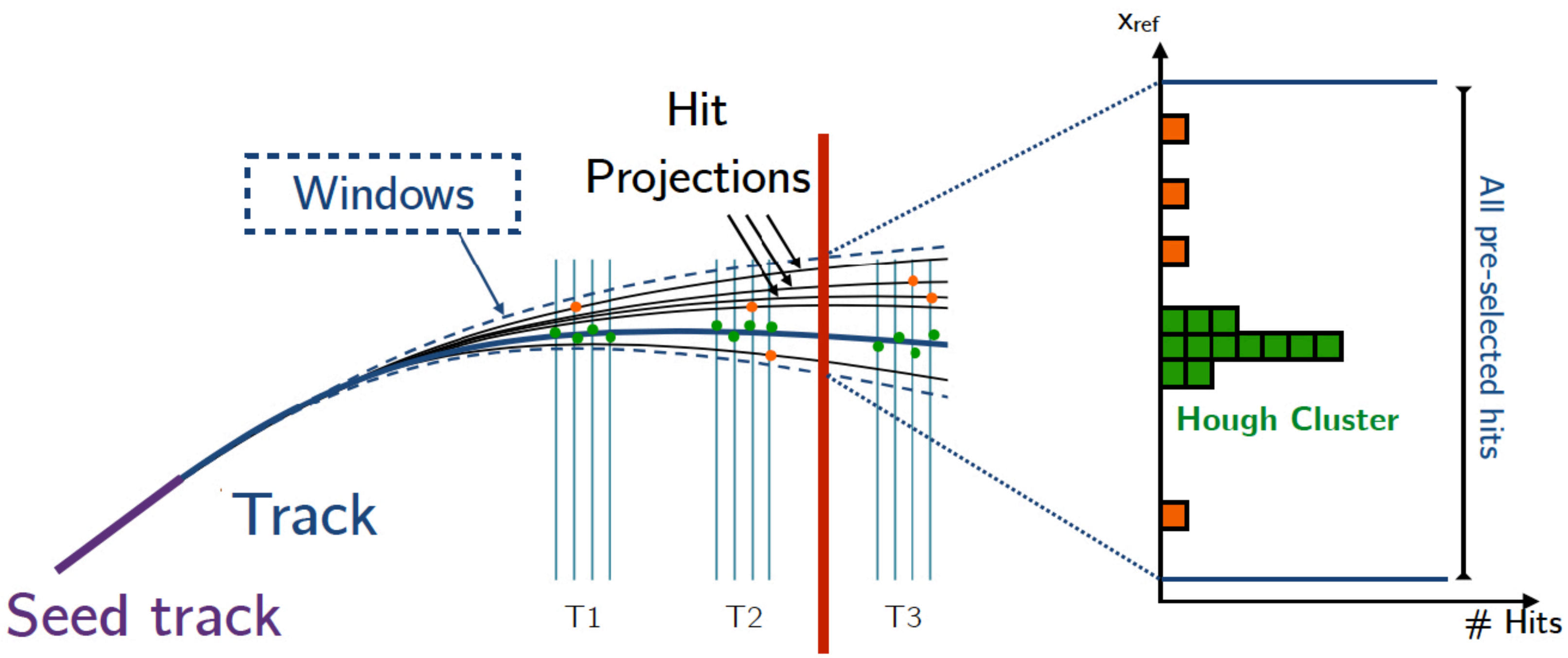
1 hit + Seed track: allows to know expected track motion in bending plane in all T-stations.

Hough-Clusters: project all measurements to reference plane and find accumulation of them

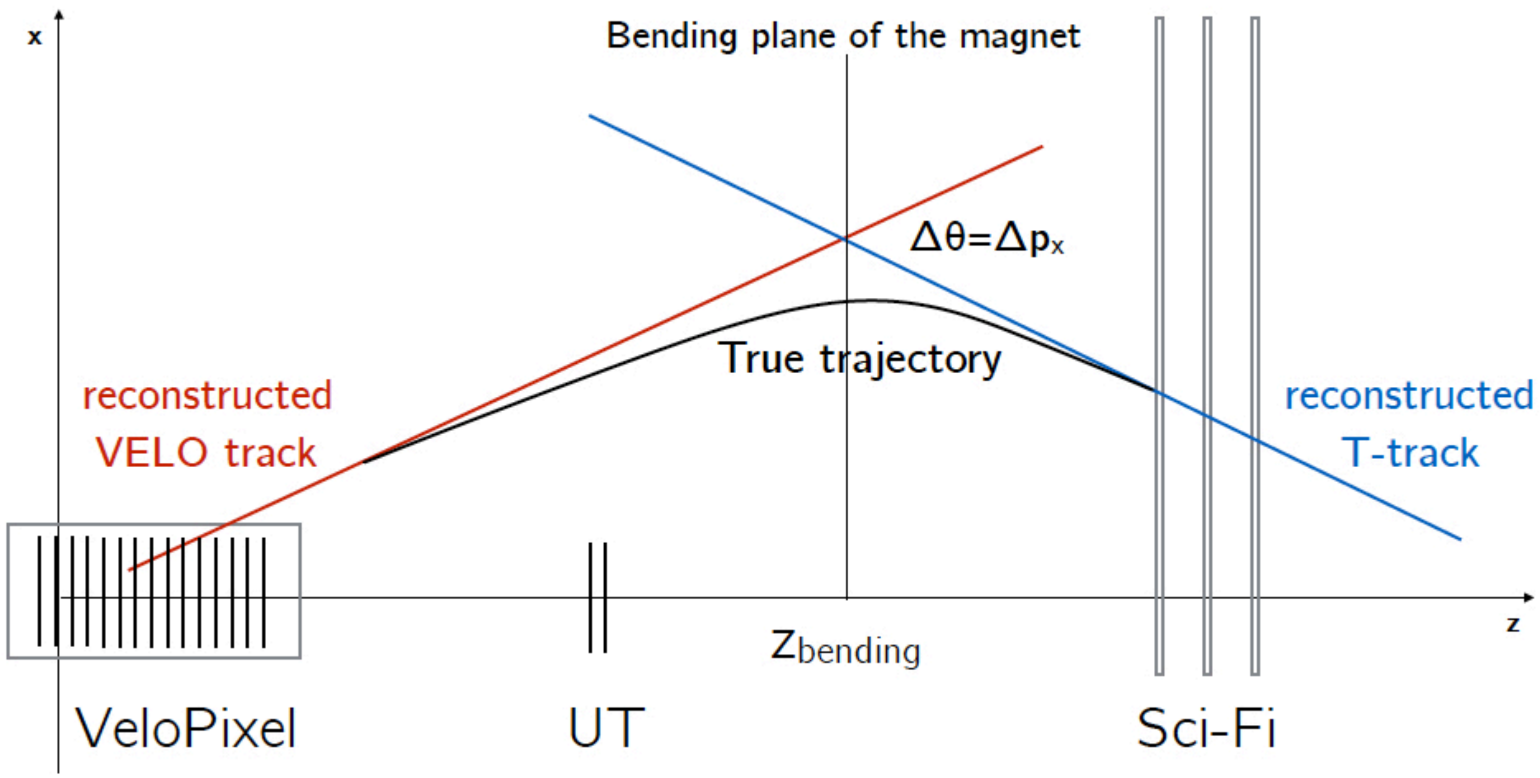
p_T-kick method



Forward tracking and Hough Clustering (1D)



Matching algorithm

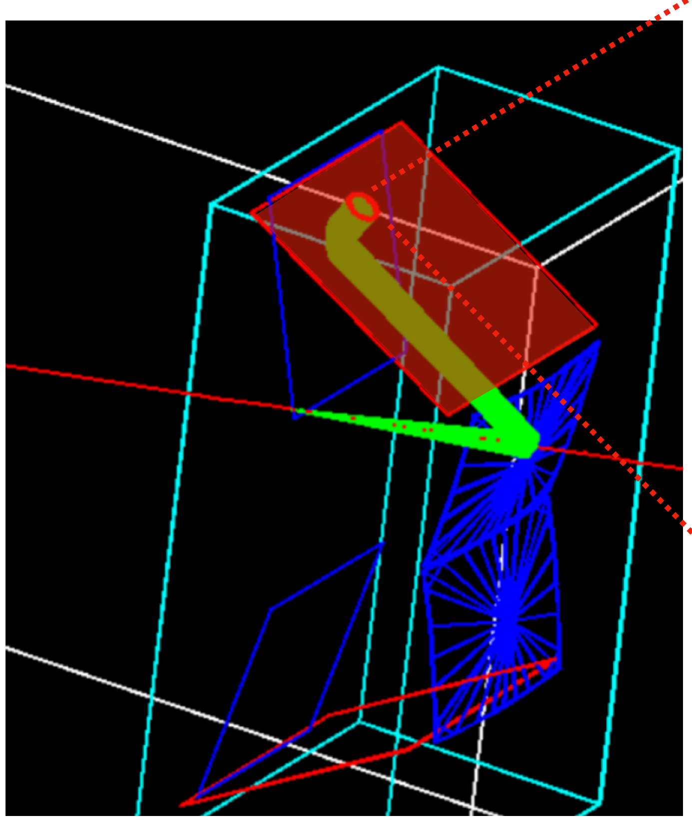


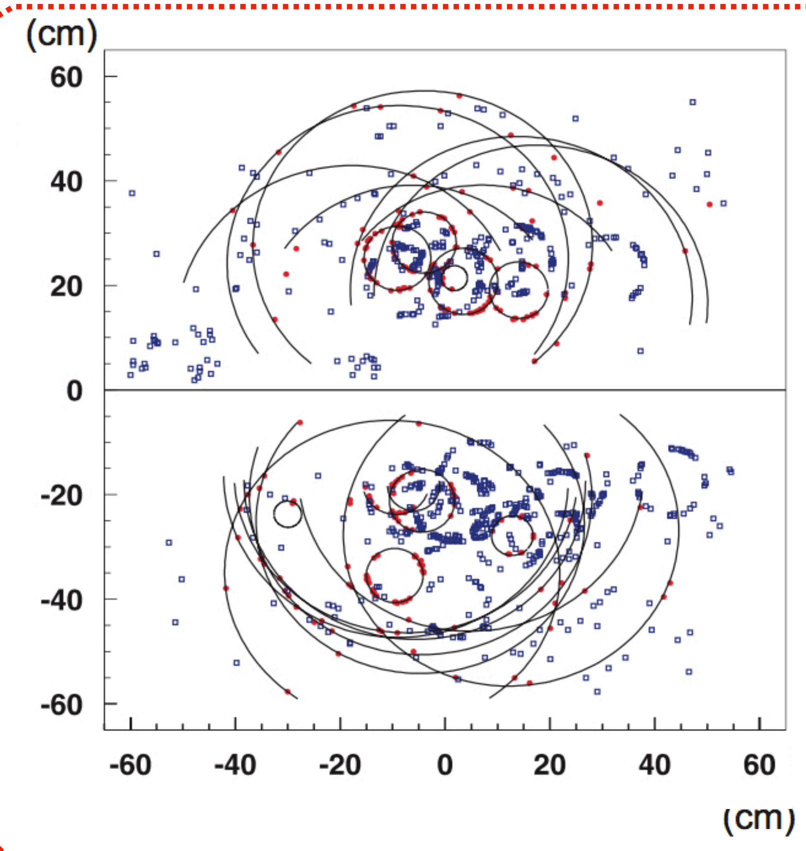
- **Seed tracks: T-Track state (S_T) from Seeding and Velo Track state from Velo Tracking (S_V)**
- **Matching of them building a χ^2 matching at $z_{bending}$ plane and Neural Net based selection**

$$\chi^2_{match'} = \left(\overrightarrow{S}^K_V(z_{mag}) - \overrightarrow{S}^K_T(z_{mag})\right)^T \cdot \left(C^K_{VELO} + C^K_T\right)^{-1} \cdot \left(\overrightarrow{S}^K_V(z_{mag}) - \overrightarrow{S}^K_T(z_{mag})\right)$$

PID using RICH detectors

Add *PID* information to tracks using RICH detectors





$$m = \frac{p}{c\beta\gamma}$$

from tracking

$$\cos\theta_C = \frac{1}{n\beta}$$

ring radius

- Rings used to assign PID likelihood.
- (offline) Info combined with other sub-detectors into a Neural Net PID probability

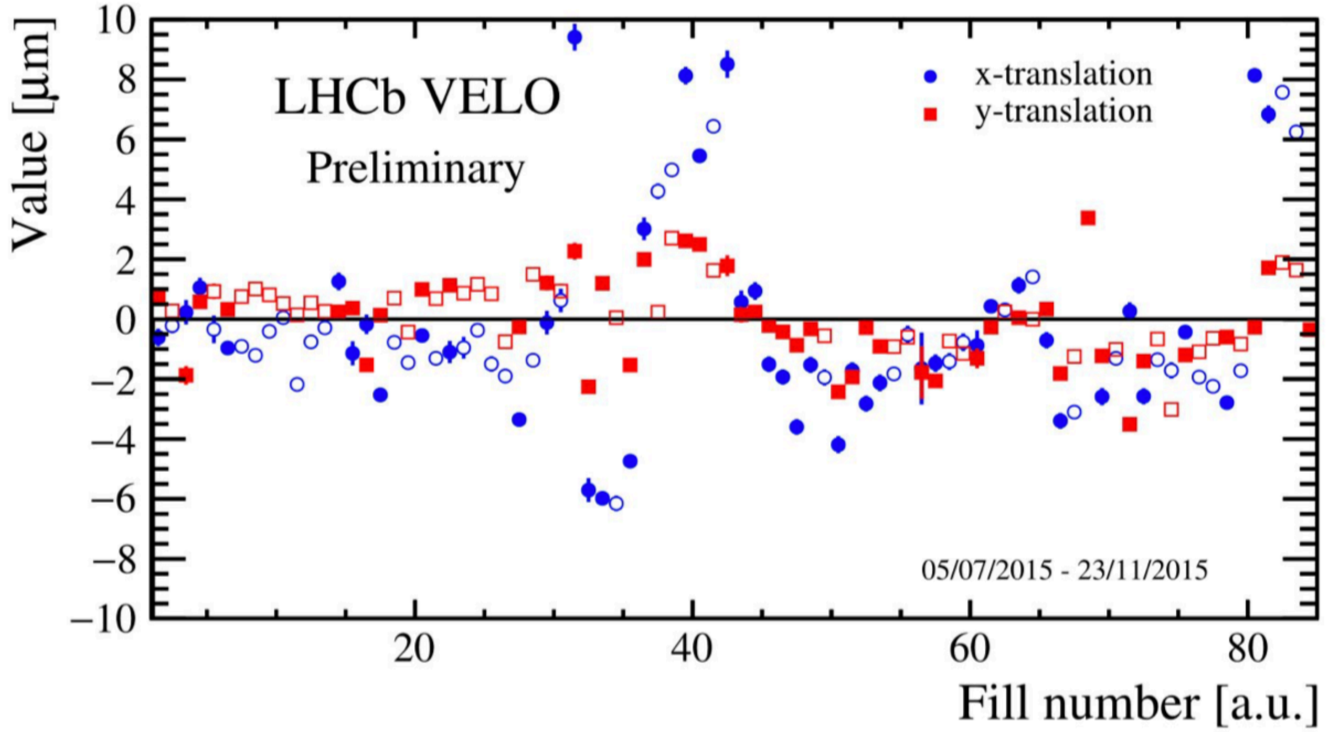
		SSE4		AVX2	
		time (s)	Speedup	time (s)	Speedup
double	scalar	233.462		228.752	
	vectorized	122.259	1.90	58.243	3.93
float	scalar	214.451		209.756	
	vectorized	55.707	3.85	26.539	7.90

Table 3.2: Performance of vectorized Rich's Ray Tracing

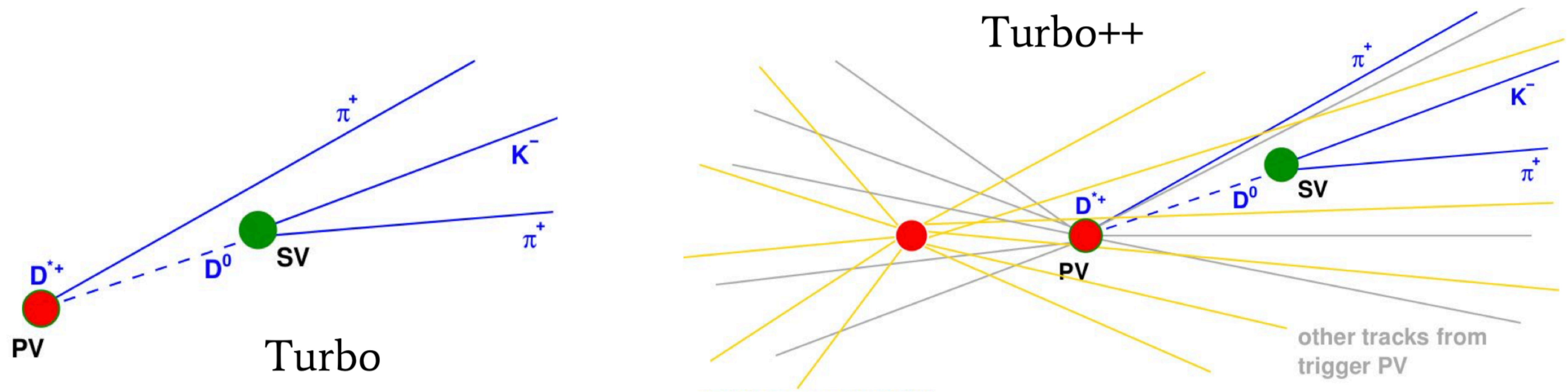
- Vectorization leads to around a factor 8 speed-up.

Real time alignment and calibration

- **Alignment and calibration crucial for optimal physics performance**
- **Alignment per fill:**
  - Collect suitable data with dedicated HLT1 selections
  - Run alignment workers on the HLT farm (1 per node)
  - Controller iterate until converged (~5 minutes)
  - Apply updates of VELO and/or Tracker alignment if needed
  - RICH mirror alignment and muon alignment for monitoring
  - ECAL gain calibration
- **Calibration per 1 h run:**
  - RICH and Outer Tracker t_0
  - Available ~1 min after collection of data



TURBO and TURBO ++



Store on disk only high level object for the decay mode of interest.

Turbo worked quite well in Run II : few analysis published after 1 week of data taking.

☞ **Towards the upgrade: Turbo++ introduced in 2016**

Turbo++ able to persist arbitrary variables on any reconstructed object

Can save HLT candidate + any reconstructed objects

Custom binary serialization and compression per event

Event size of 50 kB including minimal subset of raw data

Allow to do new things on HLT output

Run II trigger

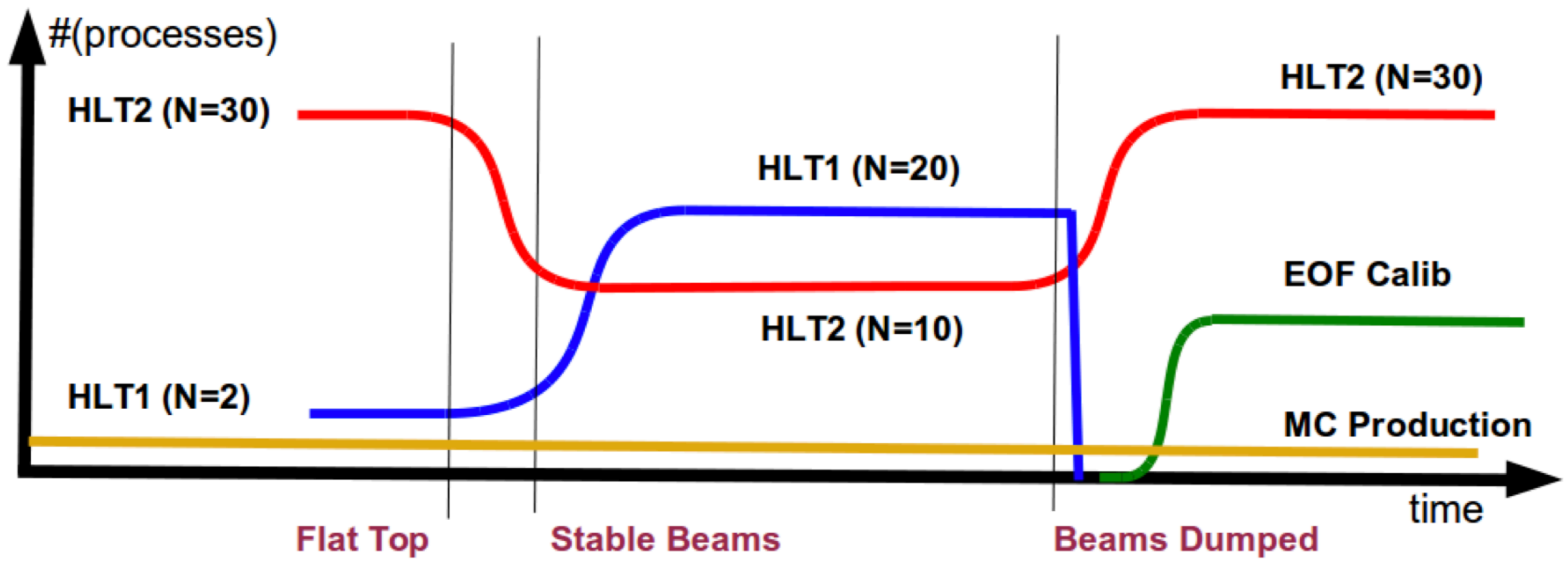


Figure 2.1: A schematic diagram showing the amount of CPU dedicated to the HLT1 and HLT2 activity during various states of the LHC collider.

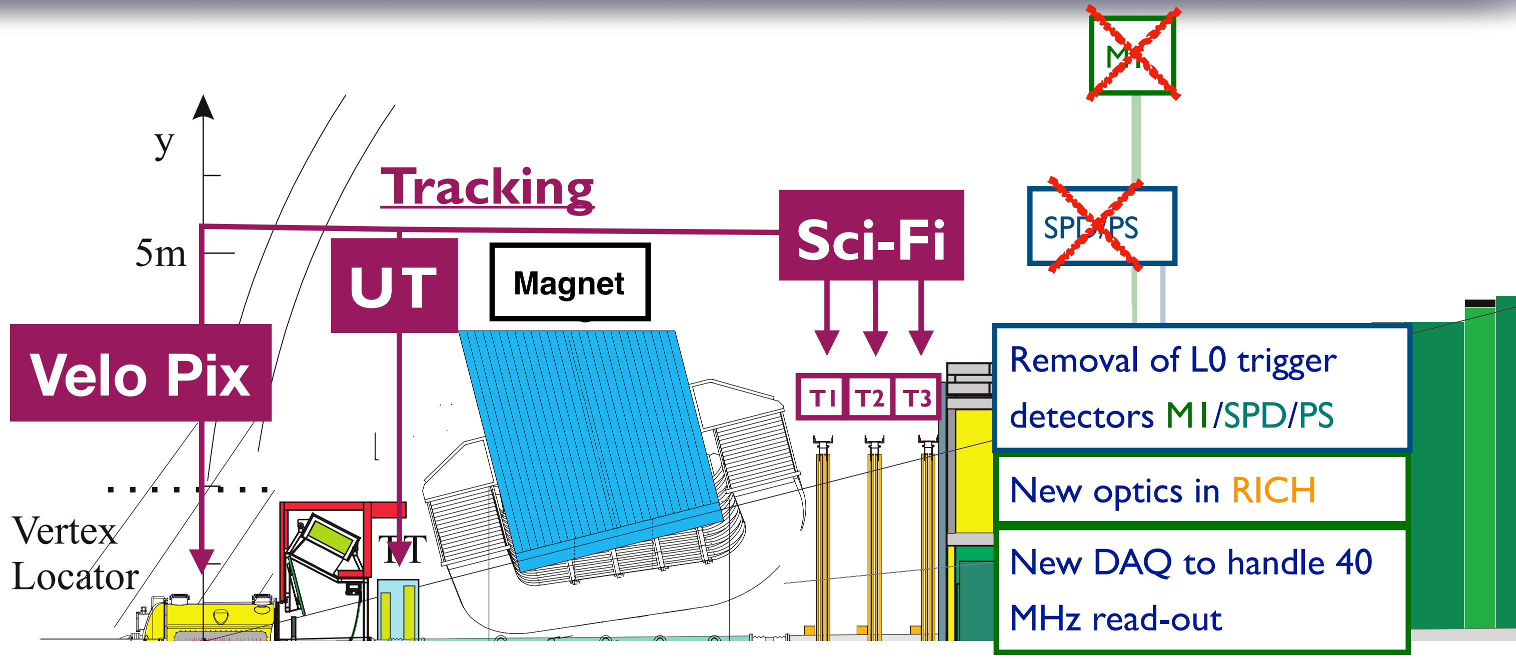
- Stable beams ~50% of the time
- Buffer events to disk and process between fills

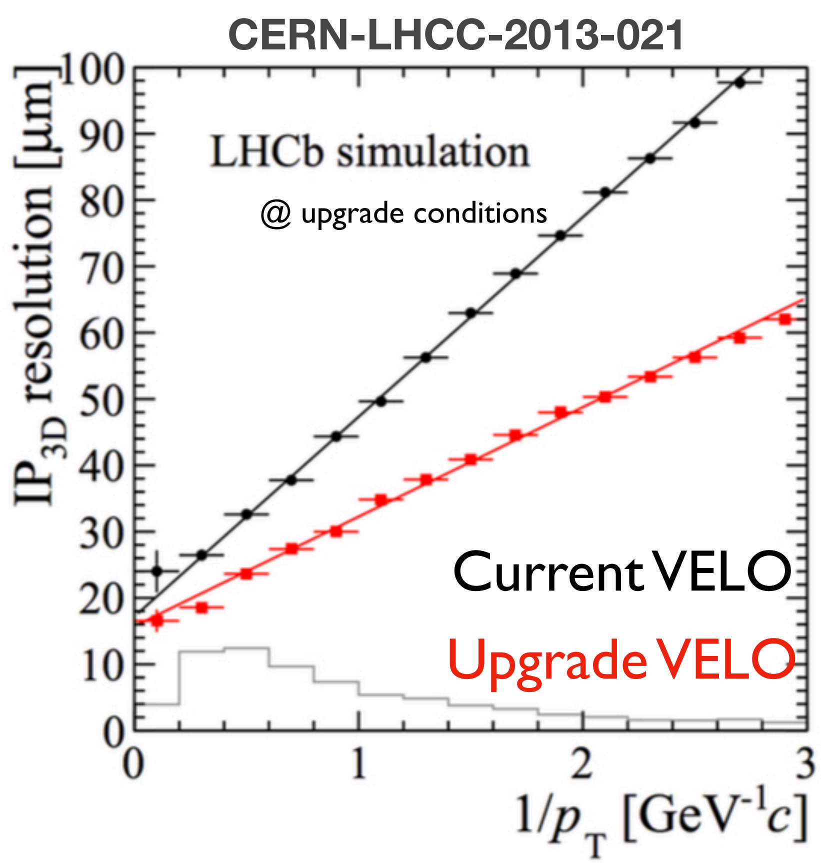
- Run II
  - Defer 100% of HLT 1 accepted events
  - More efficient use of buffers due to larger real-time reduction
  - Save 100% of events at 150 kHz instead of 20% at 1 MHz
  - Use HLT 1 output for calibration and alignment
  - 10 PiB in farm (half in 2015)

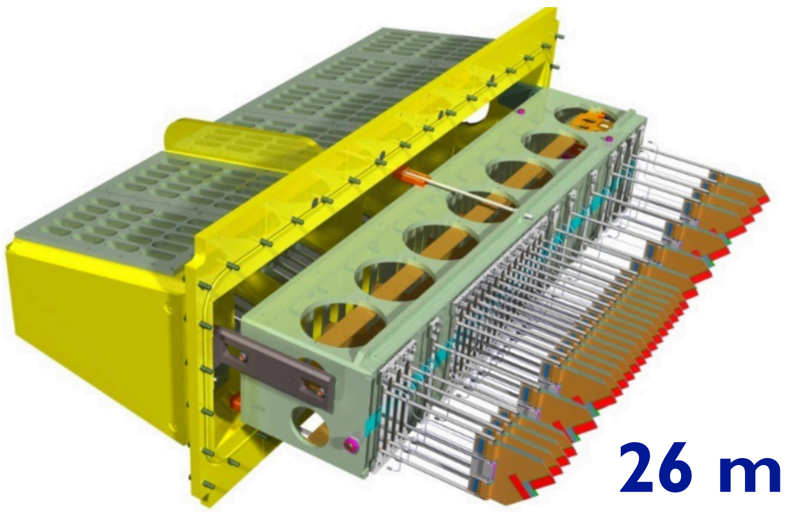
LHCb detector upgrade : Velo Pix detector

Velo Pix

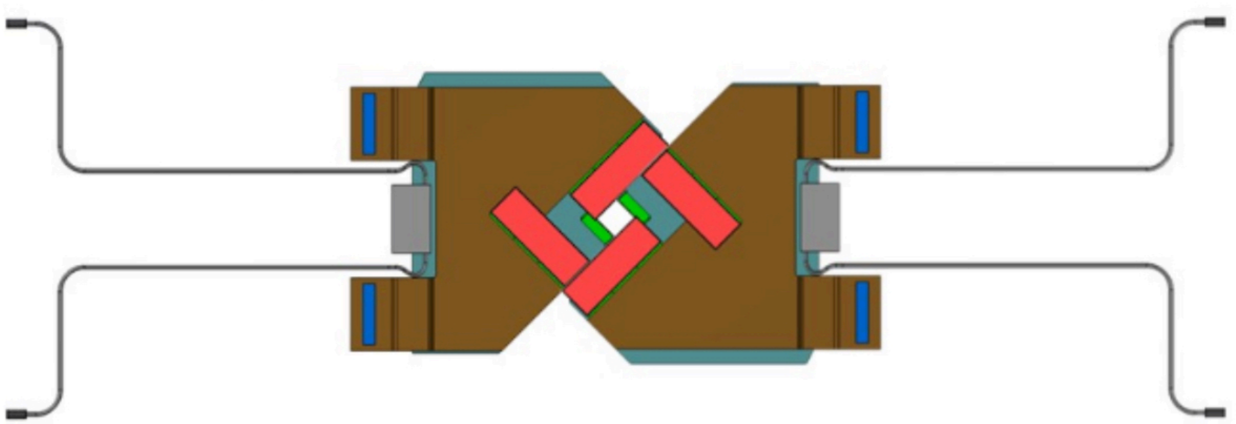
- **2-D pixel sensor** @-20°C: direct **x-y** measurement.
- **Fast tracking on raw data.**
- **5.1 mm distance** from interaction point (8.2 mm current VELO)





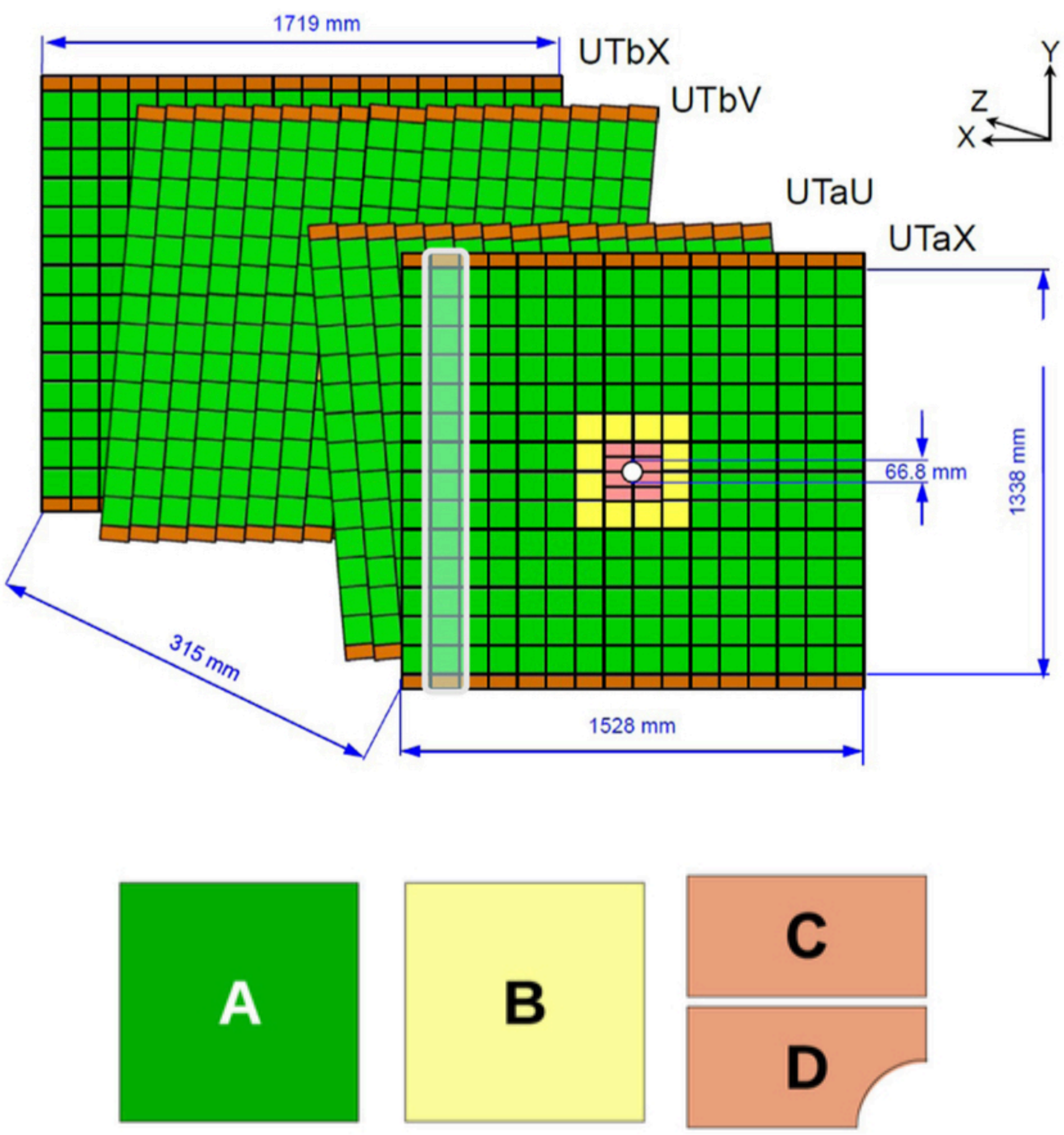


26 modules : optimized acceptance

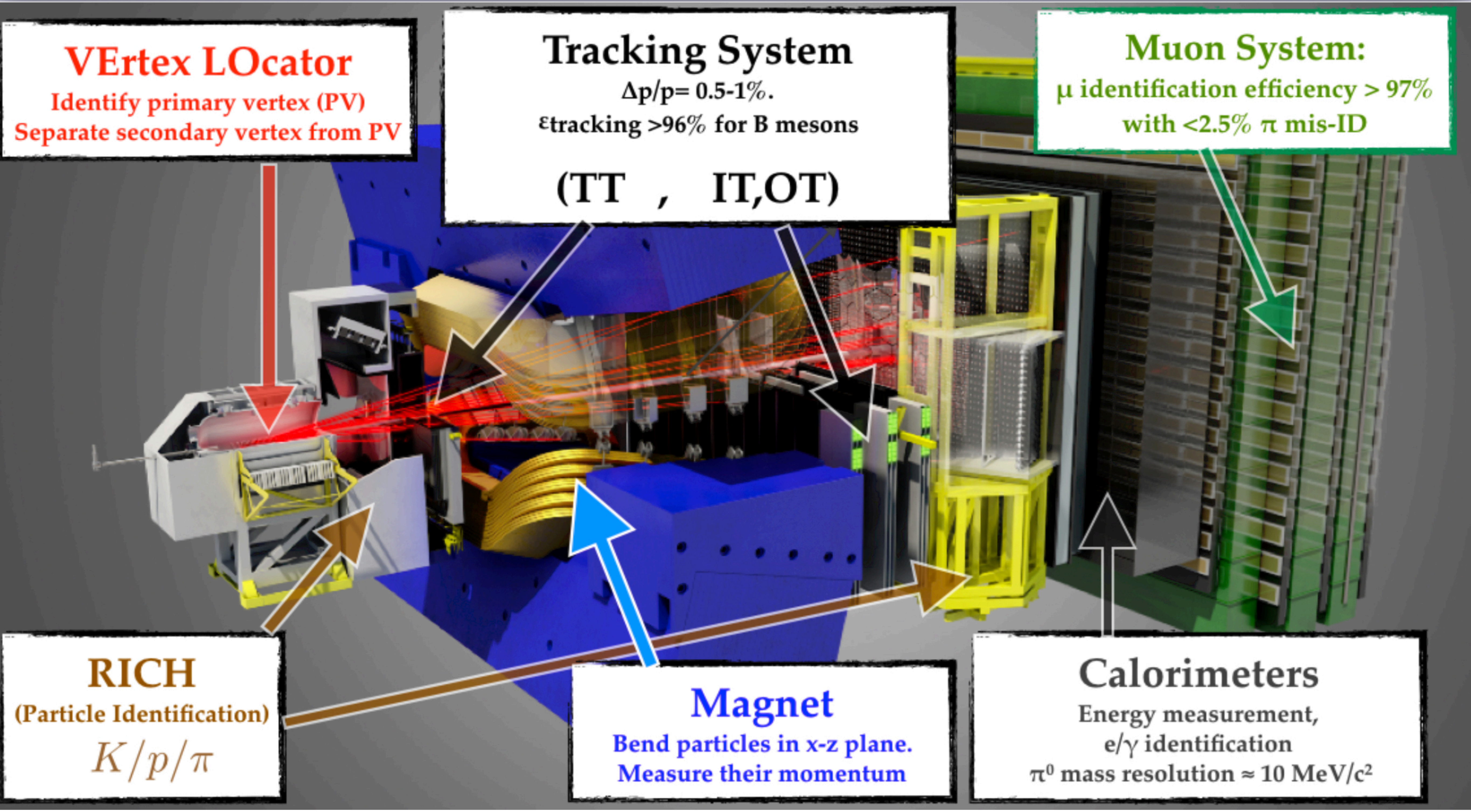


Upstream tracker (UT)

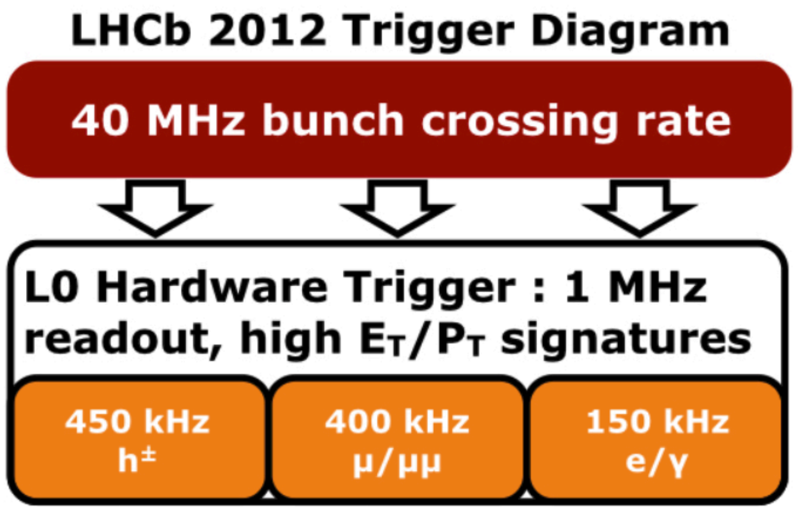
- **Larger acceptance** in central region
- **Reduced thickness**
- **Improved $\sigma_{x\text{-}z} \sim 50\ \mu m$**
- **Fast VELO tracks validation**
- **Velo-UT tracking: $\Delta p/p \sim 15\text{-}30\%$**
  allows to reduce search window in forward tracking.



Current LHCb detector



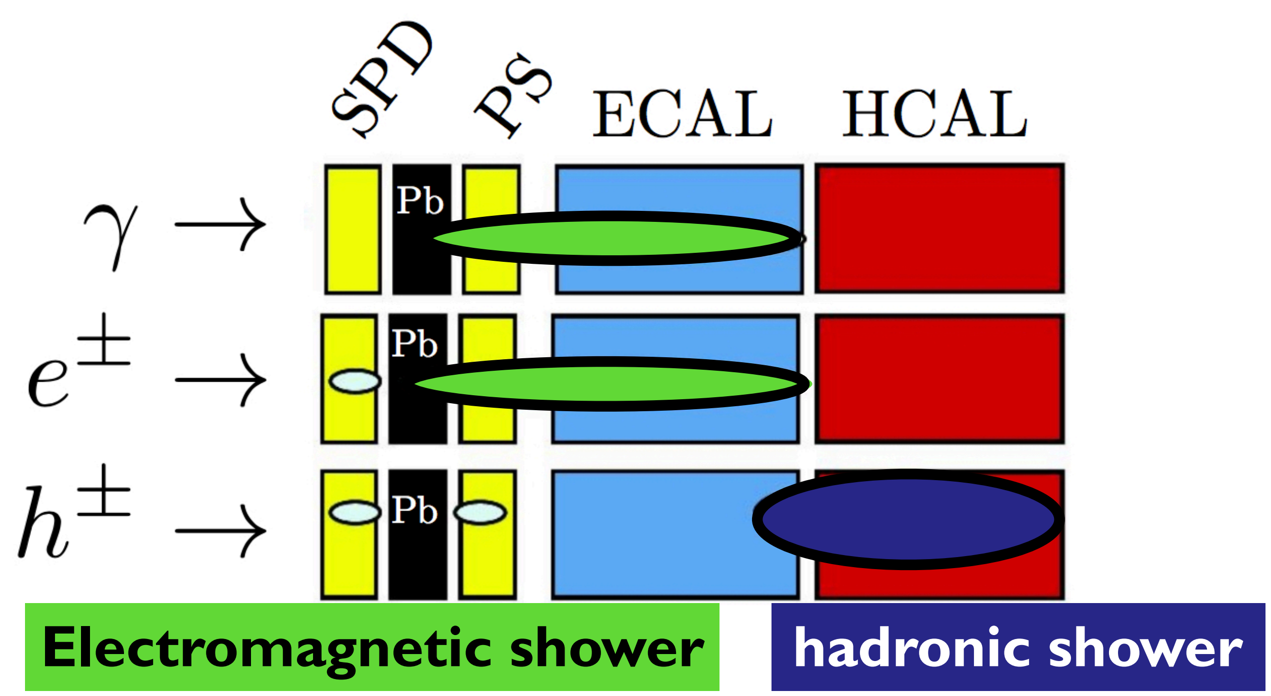
L0 hardware trigger



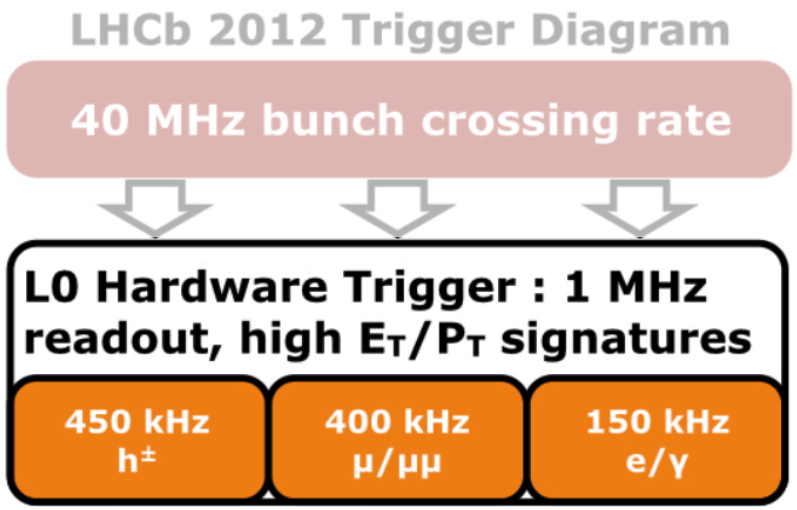
Luminosity levelling ➜ L0 hardware trigger

- ☞**Reduce 40 MHz input rate to the front end electronics read-out limitation of sub-detectors (1 MHz).**
- ☞Muon Stations and Calorimeters are able to identify the presence of **high p_T muons, high E_T hadrons, photons and electrons**.
- ☞Events with very high track multiplicity are rejected (n_{SPD} cut).

Calorimeters search for high E_T deposits



LHCb L0 trigger

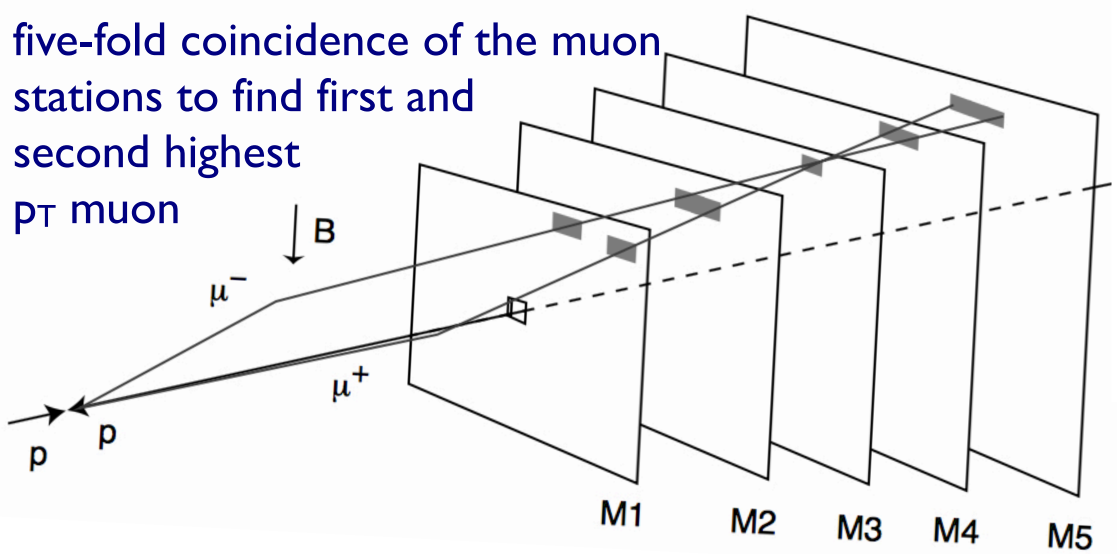


L0 Hardware trigger

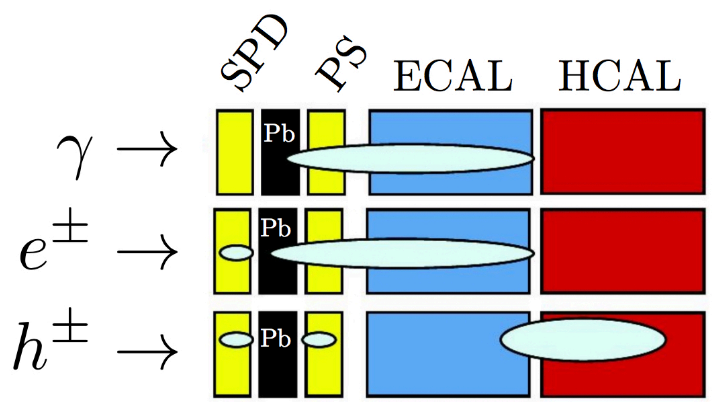
- ☞Reduce 40 MHz input rate to 1 MHz.
- ☞Selection performed at Hardware level using information from Muon Stations (p_T muons), hadronic & electromagnetic Calorimeters (HCAL and ECAL) + Scintillating Pad Detector (SPD) and pre-shower (PS) measure the E_T and event multiplicity.

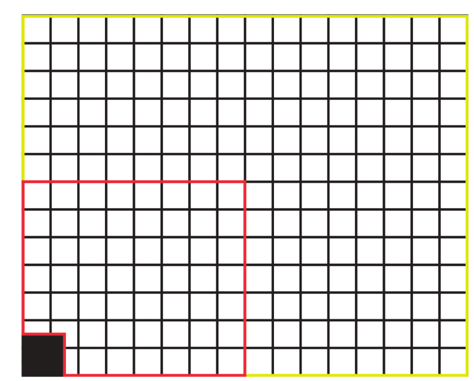
Muon Stations

five-fold coincidence of the muon stations to find first and second highest p_T muon

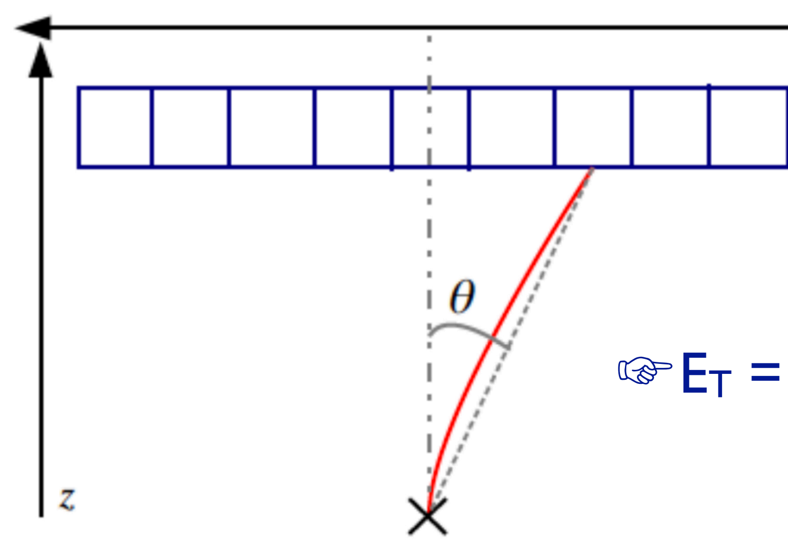


Calorimeters





highest 2x2 ECAL/HCAL cluster to build candidate and perform L0 trigger

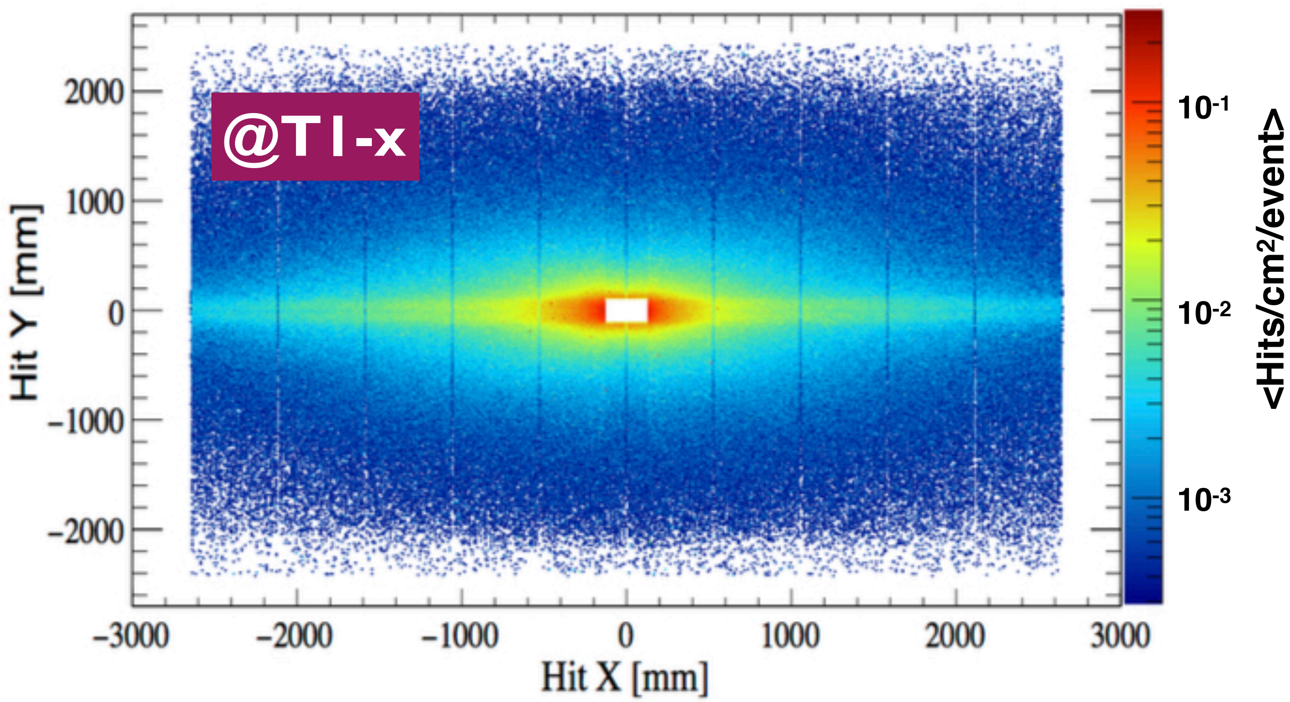


☞$E_T = E \times \sin(\theta)$

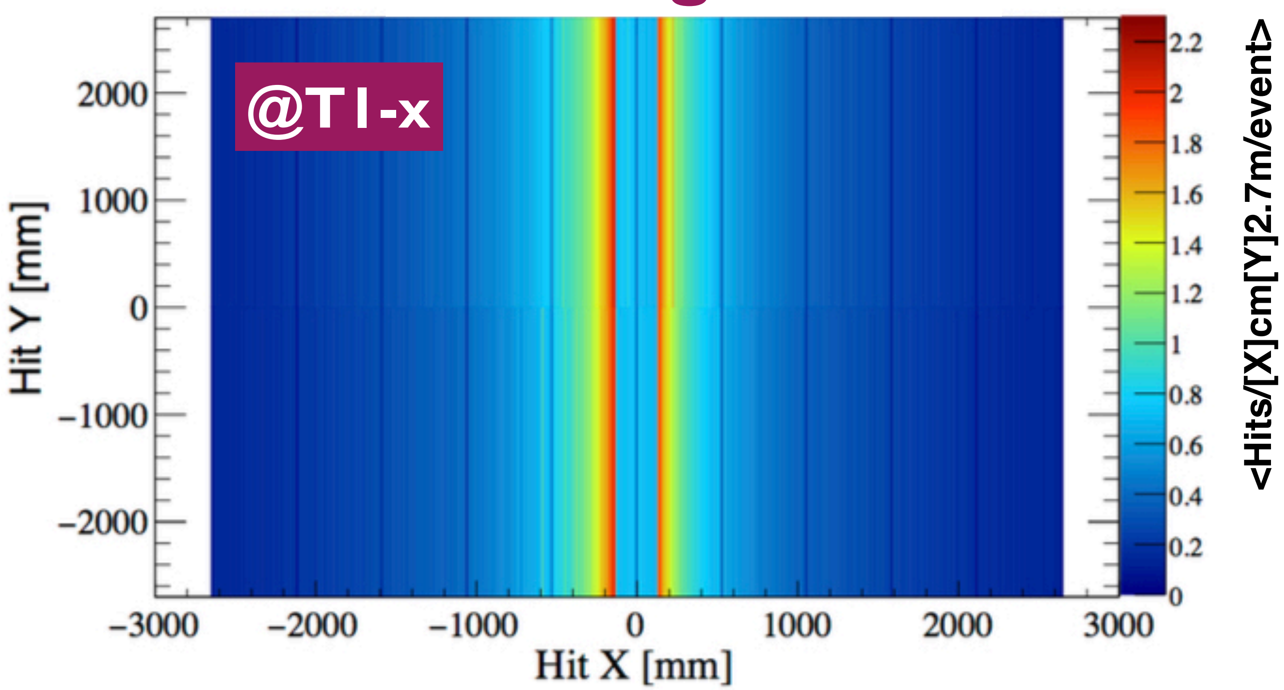
Multiplicity

L0 Decision	2011 thresholds	2012 thresholds	SPD multiplicity
L0Muon	$p_T > 1.48$ GeV/c	$p_T > 1.76$ GeV/c	< 600
L0DiMuon	$p_T^{12} > 1.296$ GeV/c	$p_T^{12} > 1.6$ GeV/c	< 900
L0Hadron	$E_T > 3.6$ GeV	$E_T > 3.5 - 3.74$ GeV/c	< 600
L0Electron	$E_T > 2.5$ GeV	$E_T > 2.5 - 2.86$ GeV/c	< 600
L0Photon	$E_T > 2.5$ GeV	$E_T > 2.5 - 2.96$ GeV/c	< 600

Before digitisation



After digitisation



Integrated inefficiencies effects. →

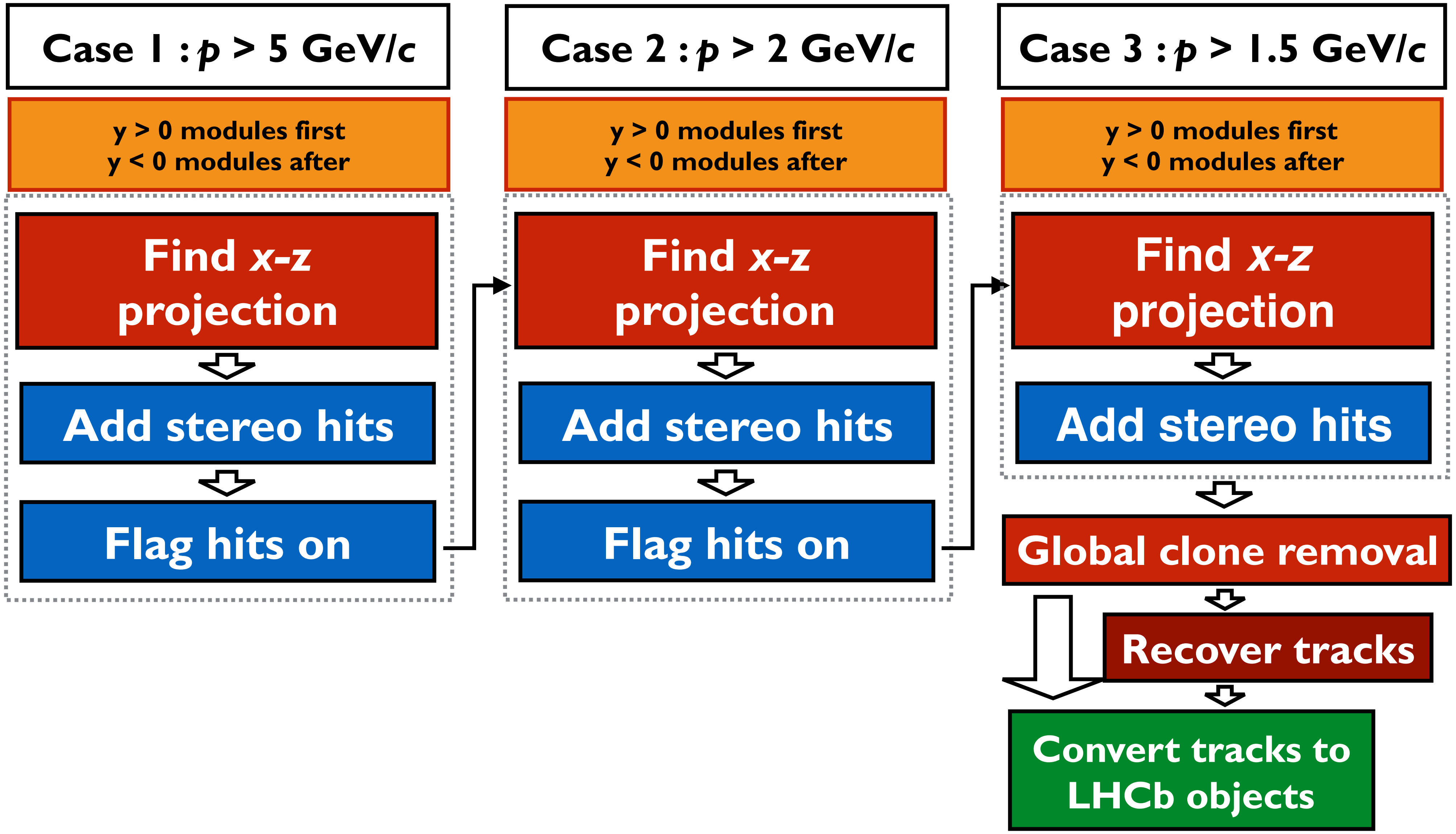
- **Light attenuation.**
- **Irradiation damage.**
- **Clustering inefficiencies.**
- **Dead-regions.**
- **Noise and spill-over.**
- **....**

Track reconstruction algorithm for the Sci-Fi must be efficient

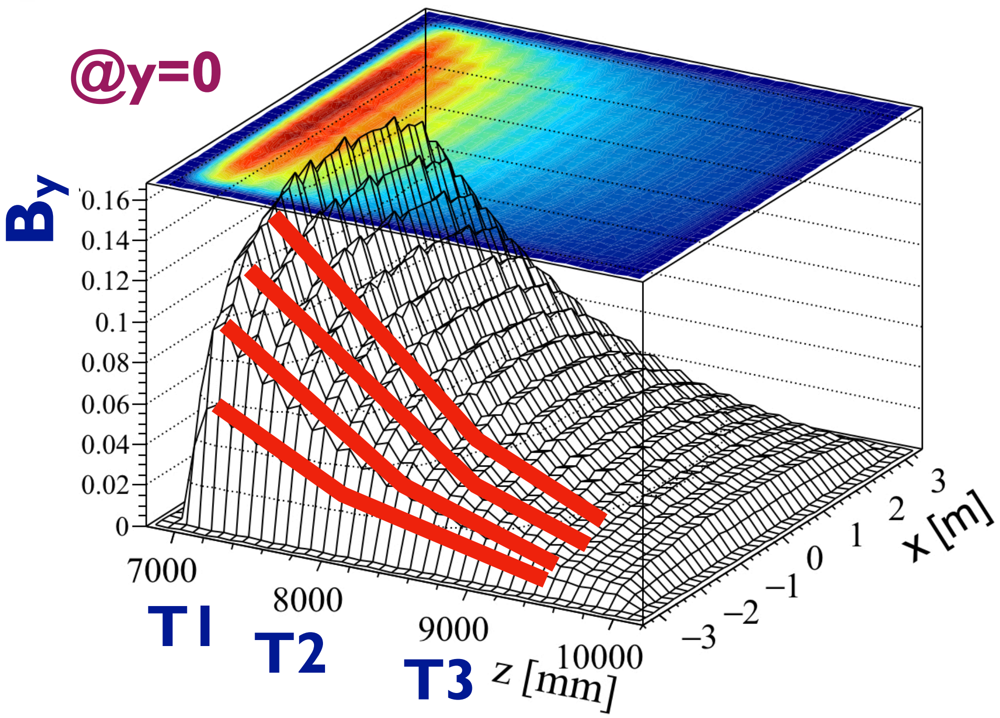
- Recover hit inefficiencies (<12 fired layers) — $\epsilon_{tracking}$ ↑ fake rate ↑ timing ↑
- Reduce fake rate (large slopes) — $\epsilon_{tracking}$ ↓ fake rate ↓
- Find low momentum tracks (large slopes) — $\epsilon_{tracking}$ ↑ fake rate ↑ timing ↑

Hybrid Seeding algorithm : algorithm layout

- Designed as a *tracking in projection* track reconstruction algorithm.
- Able to cope with hit inefficiencies and low momenta (< 5 GeV/c) tracks at a low *ghost rate*.
- Hit flagging leads to cleaner environment for low momentum tracks search.
- Improved track fit model accounts for local (Sci-Fi region) magnetic field.
- Track recovery introduced, new parameterisation of search windows, fast stereo hit strategy.



Fringe magnetic field parameterisation in Sci-Fi region



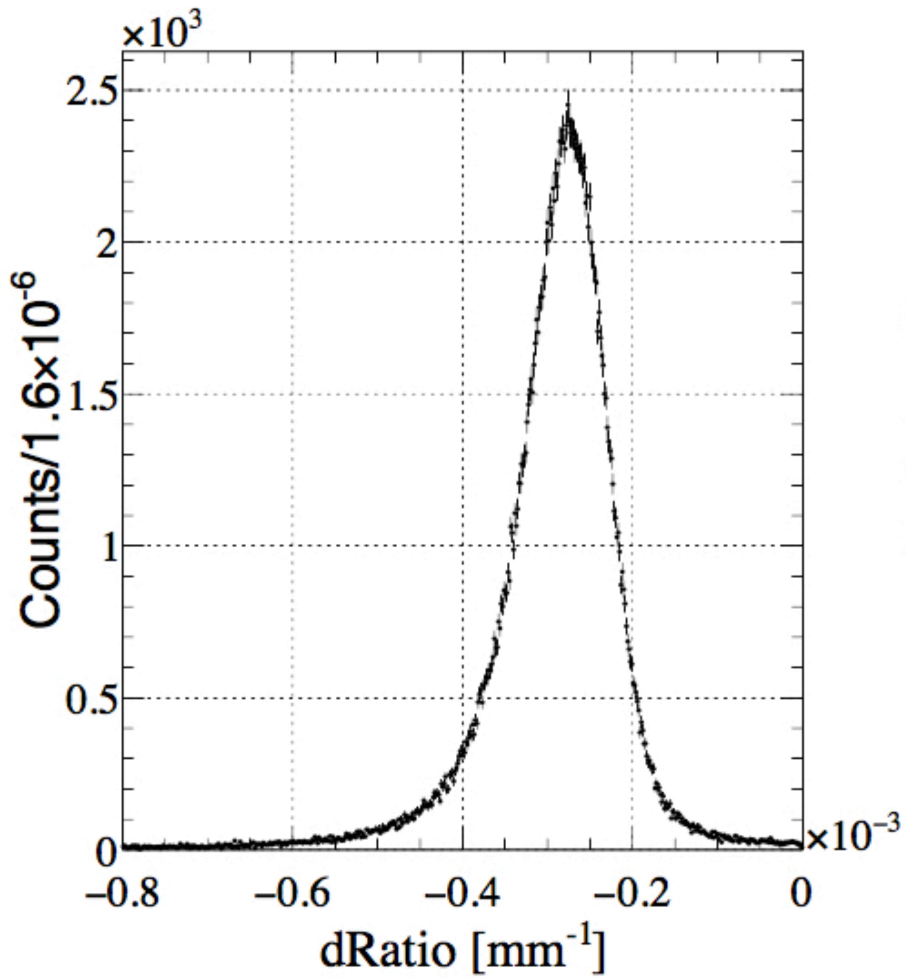
☞ **Sci-Fi region equation of motion**

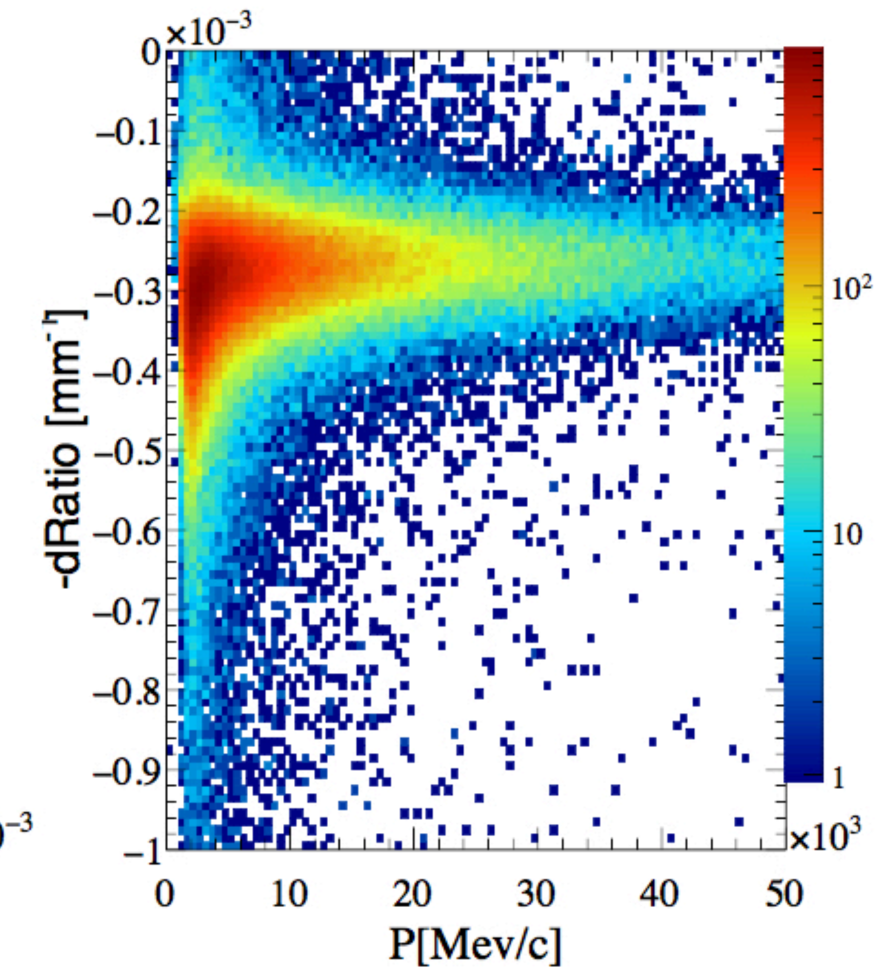
$$\vec{B} \sim \begin{pmatrix} 0 \\ B_0 + B_1 \cdot z \\ 0 \end{pmatrix} \Rightarrow \begin{cases} x(z) = x_0 + t_x \cdot z + \frac{q}{2p} B_0 \cdot z^2 \left(1 + \frac{B_1}{3B_0} \cdot z\right) \\ \hookrightarrow \quad x(z) = a_x + b_x \cdot z + c_x \cdot z^2 \left(1 + \mathbf{dRatio} \cdot z\right) \\ y(z) = y_0 + b_y \cdot z \end{cases}$$

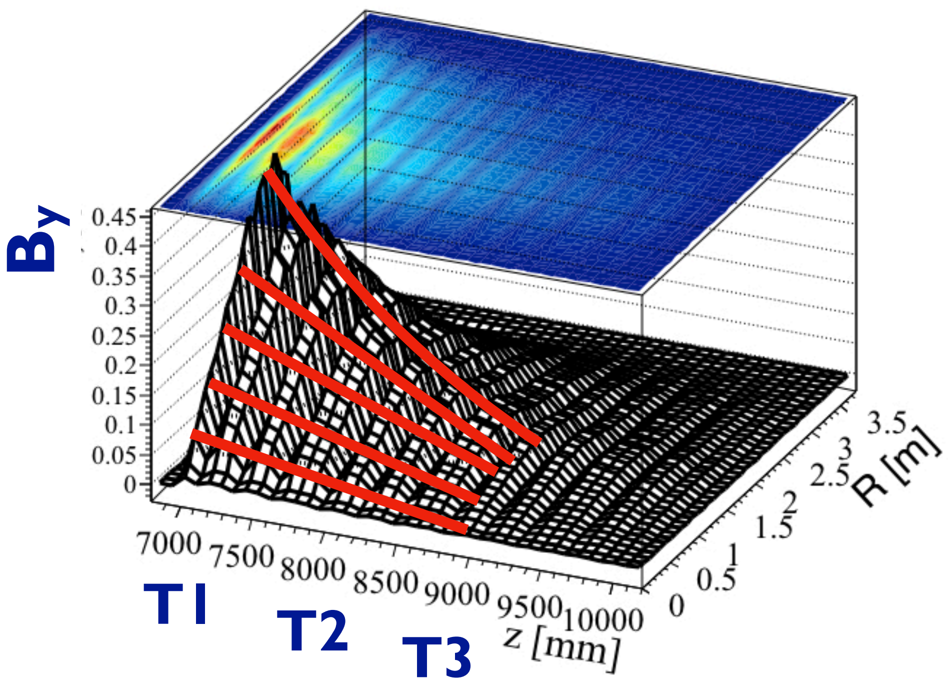
☞ **“Cubic correction” (dRatio)** accounts for B_y field effects on tracks in the Sci-Fi region.

☞ **TDR seeding** was using a simple **parabolic** model.

☞ **On true tracks:** fit for cubic model and look at ratio cubic and quadratic term







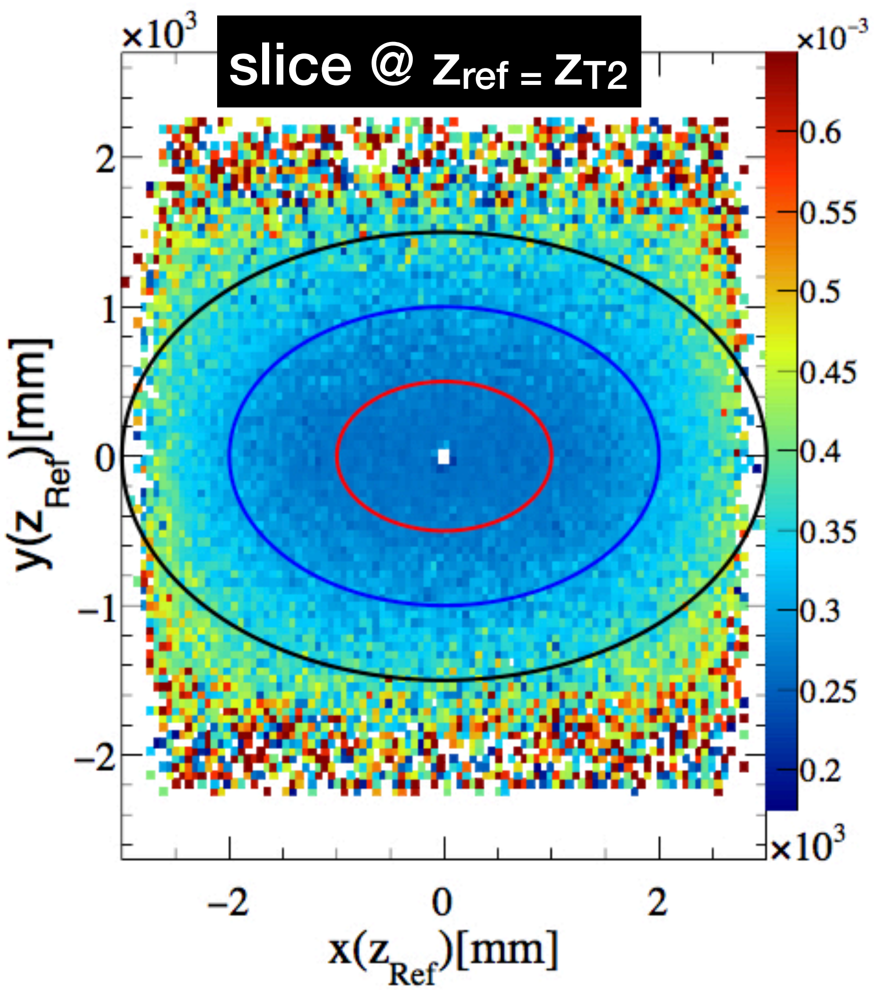
☞ **Sci-Fi region equation of motion**

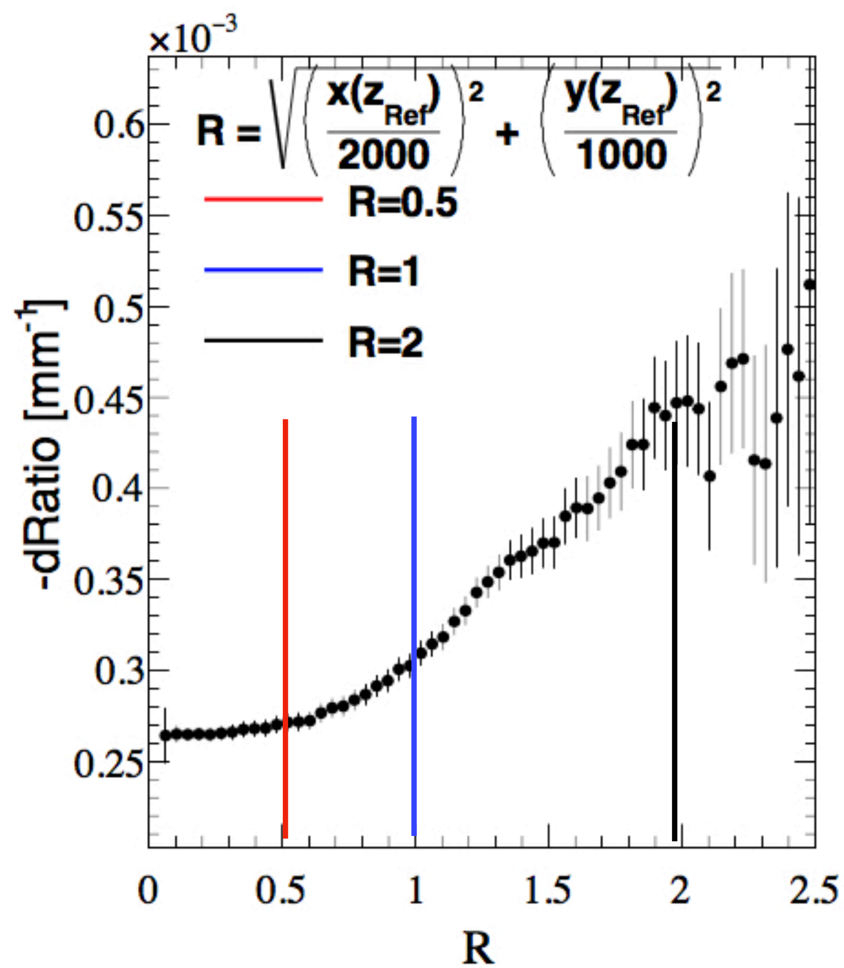
$$\vec{B} \sim \begin{pmatrix} 0 \\ B_0 + B_1 \cdot z \\ 0 \end{pmatrix} \Rightarrow \begin{cases} x(z) = x_0 + t_x \cdot z + \frac{q}{2p} B_0 \cdot z^2 \left(1 + \frac{B_1}{3B_0} \cdot z\right) \\ \hookrightarrow \quad x(z) = a_x + b_x \cdot z + c_x \cdot z^2 \left(1 + \mathbf{dRatio} \cdot z\right) \\ y(z) = y_0 + b_y \cdot z \end{cases}$$

☞ **Fringe field effects accounted for depending where the track is going through at z ~ z_{T2}**

$$\begin{aligned} B_1 &= B_1(x,y,z) \simeq F(x,y)g(z) \\ B_0 &= B_0(x,y,z) \simeq F'(x,y)g(z) \end{aligned}$$

$$\texttt{dRatio} = \frac{B_1}{3B_0} = \frac{1}{3}\frac{F(x,y)}{F'(x,y)}$$





☞ **High resolution homogeneous** detector **requires** a **precise track fit model**

☞ **No additional degrees of freedom** to a parabolic model, but **requires x-z** and **y-z** track motion information.

Hybrid Seeding algorithm : performance

Tracking efficiency ($\epsilon_{tracking}$) ↑ fake rate ↓ Timing ↓

	TDR algorithm $\epsilon_{tracking}$ (clones)	My Algorithm $\epsilon_{tracking}$ (clones)
long$_{\text{from B}}$	$(84.4 \pm 0.1)(2.0)$ %	$(93.4 \pm 0.1)(0.0)$ %
long$_{\text{from B}}$ $p > 5$ GeV/c	$(90.0 \pm 0.1)(1.6)$ %	$(95.4 \pm 0.1)(0.0)$ %
downstream	$(73.6 \pm 0.1)(3.0)$ %	$(89.7 \pm 0.1)(0.0)$ %
downstream $p > 5$ GeV/c	$(88.7 \pm 0.2)(1.8)$ %	$(95.2 \pm 0.1)(0.0)$ %
Fake tracks rate	(21.3 ± 0.1) %	(7.9 ± 0.1) %
Avg. Timing [ms/evt]	79.44	22.30