Real-time analysis in Run 3 with the LHCb experiment

Mika Vesterinen

University of Warwick

On behalf of the LHCb RTA project

CPAD Instrumentation Frontier Workshop 2021

Stony Brook University, NY

18-22 March 2021

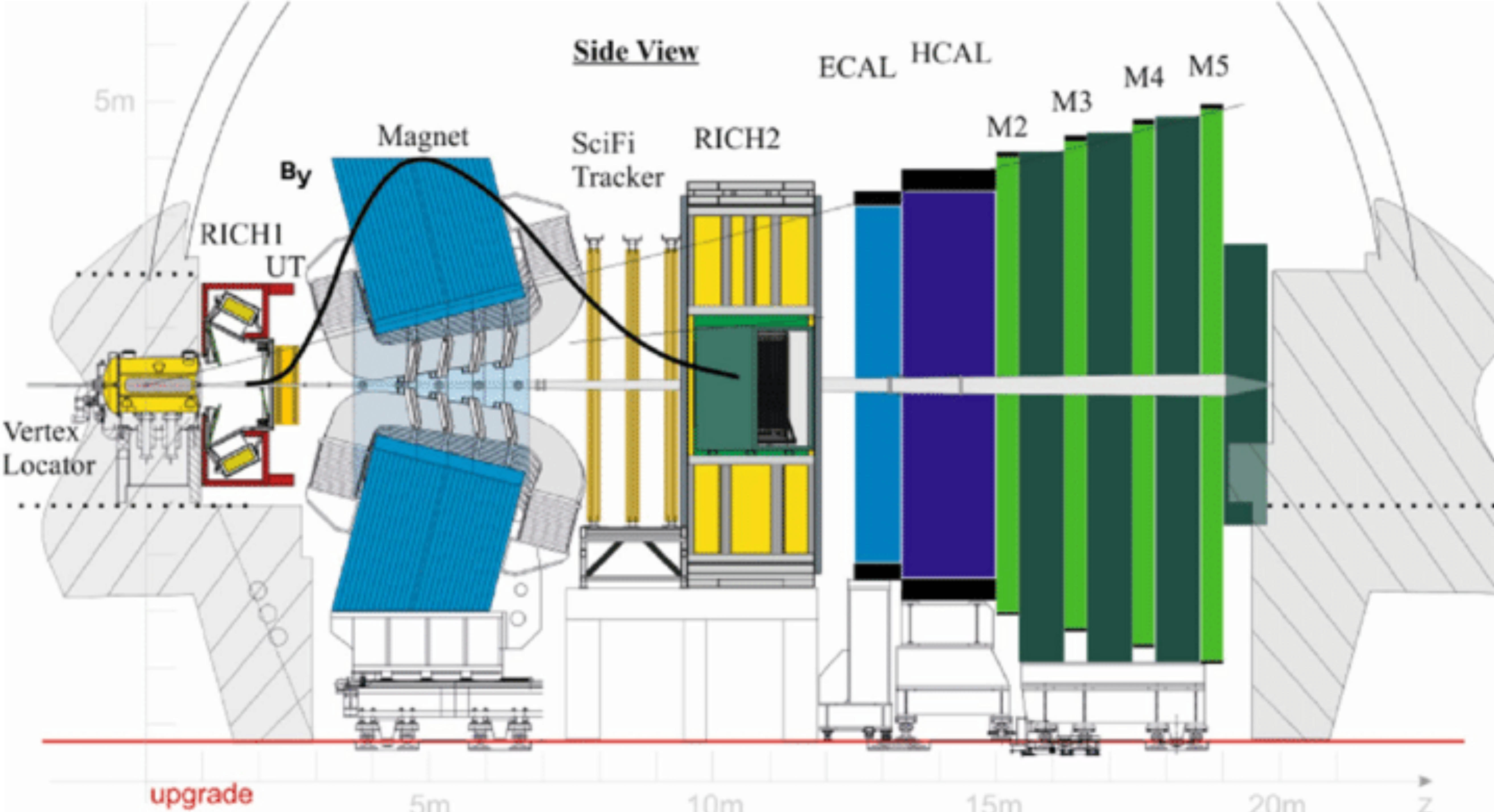
WARWICK

erc

UKRI

Science and Technology Facilities Council

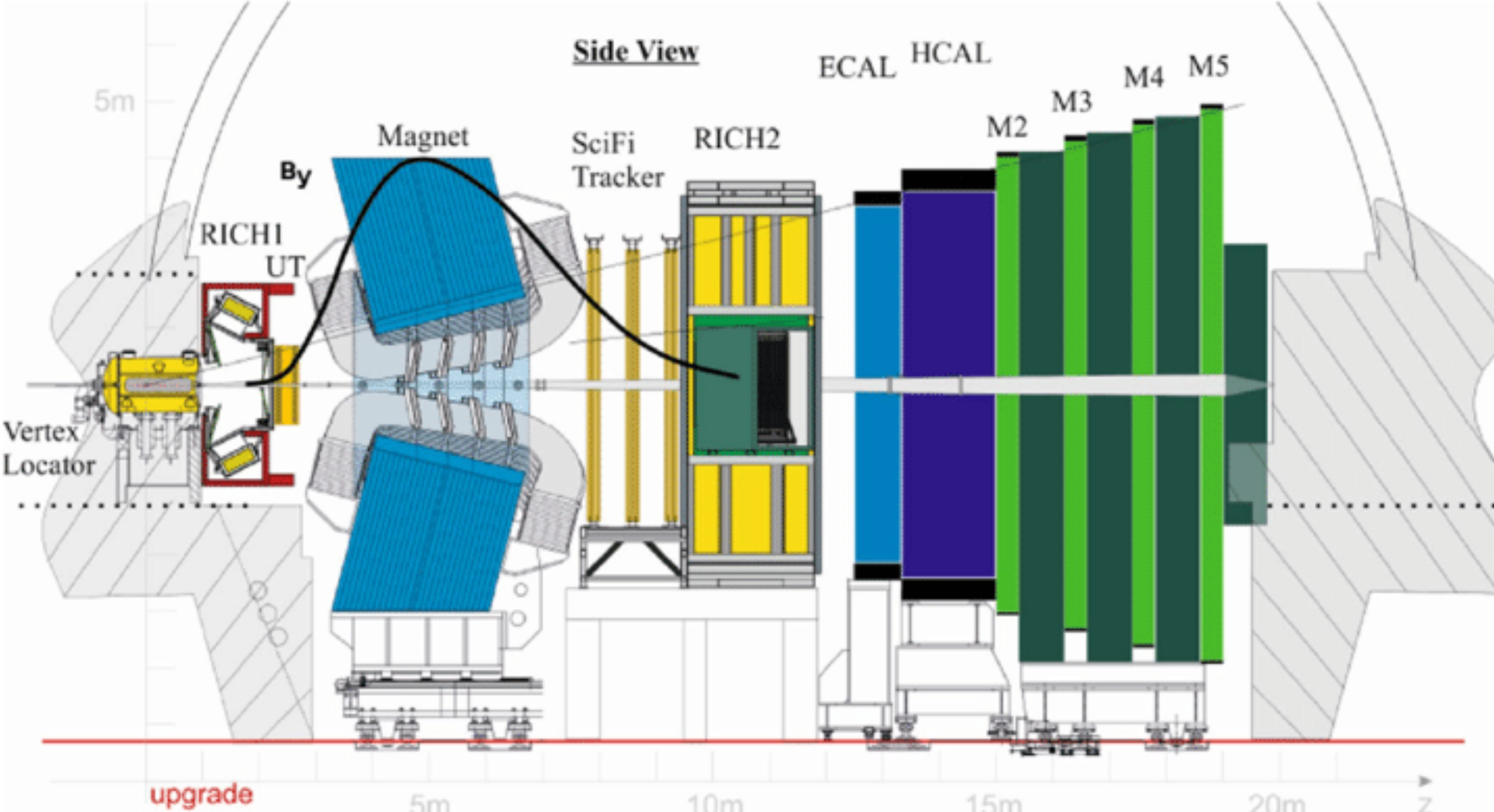
LHCb detector (Upgrade I configuration)



Compared to original experiment

1. Higher granularity detectors
2. Triggerless readout + full software trigger

LHCb detector (Upgrade I configuration)



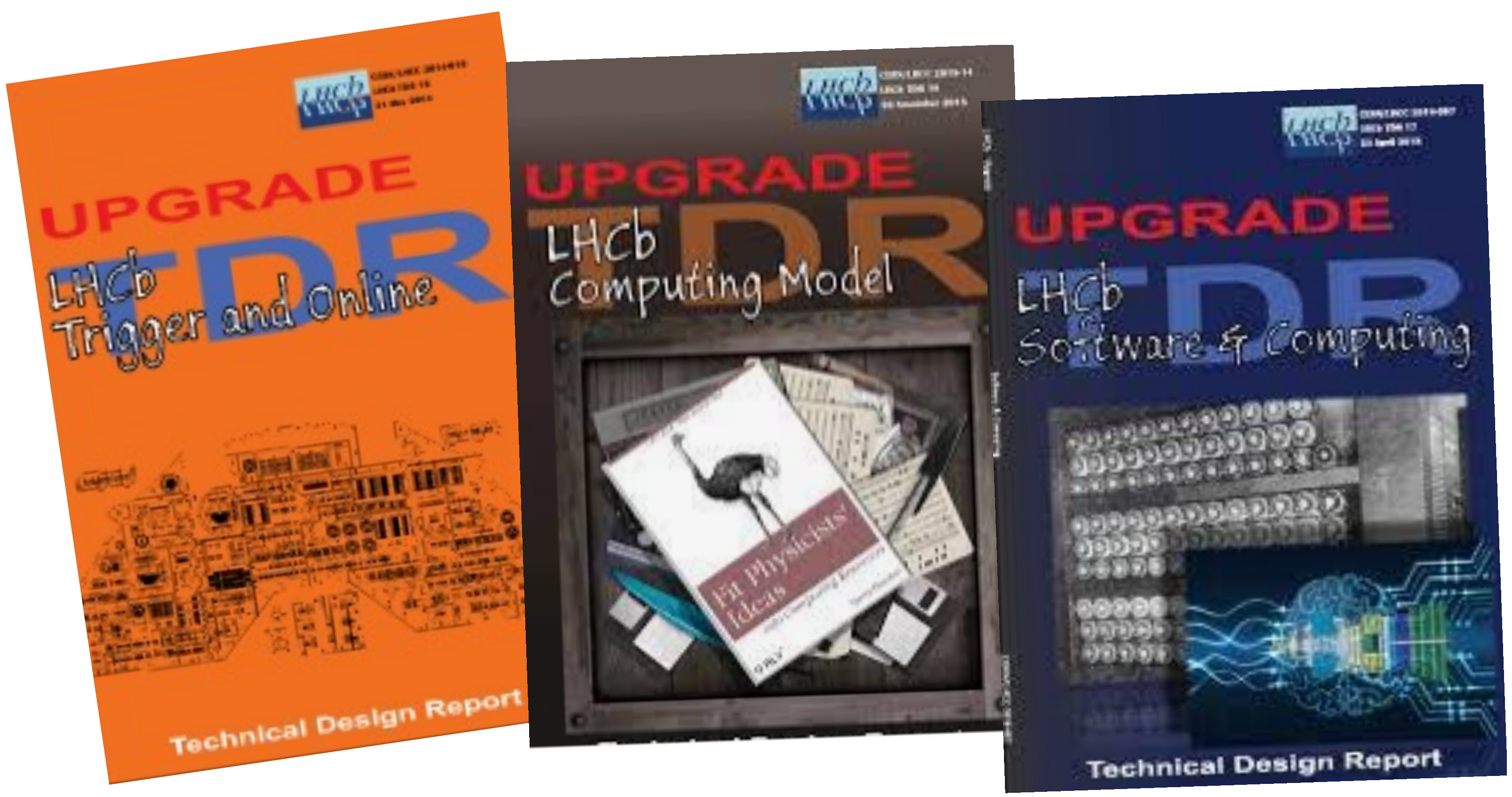
Compared to original experiment
1. Higher granularity detectors
2. Triggerless readout + full software trigger

Same/better physics performance @5x higher luminosity

Opportunities and challenges

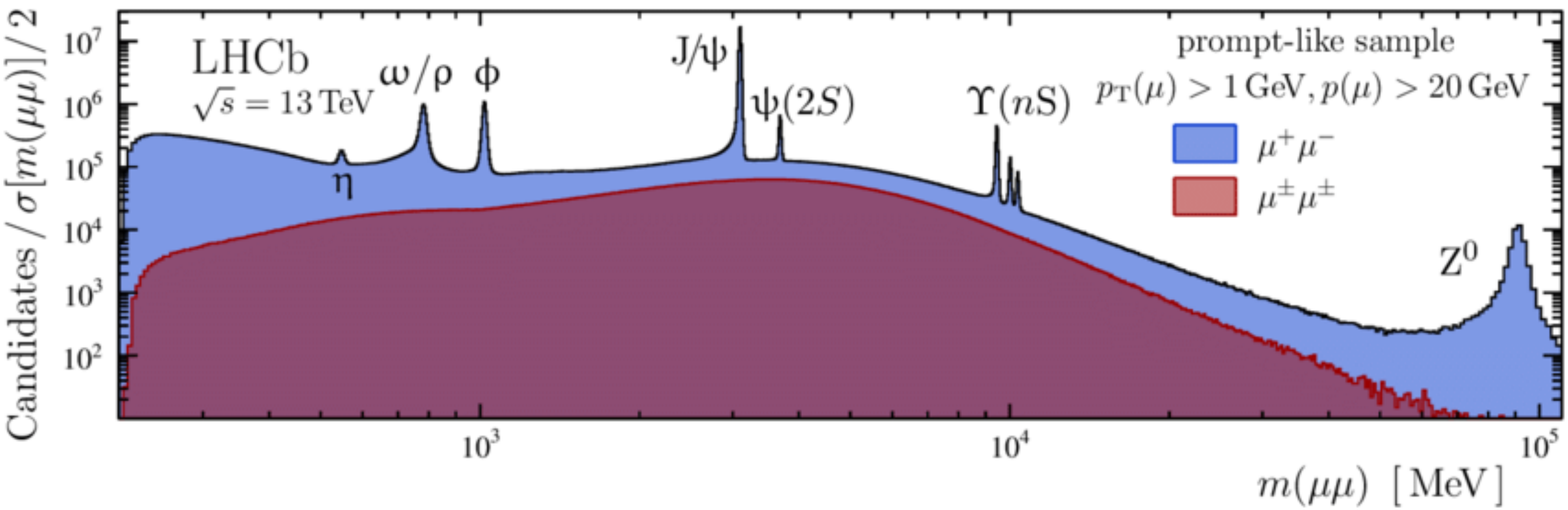
🤞lots of physics

😬lots of data to process



For example, muons

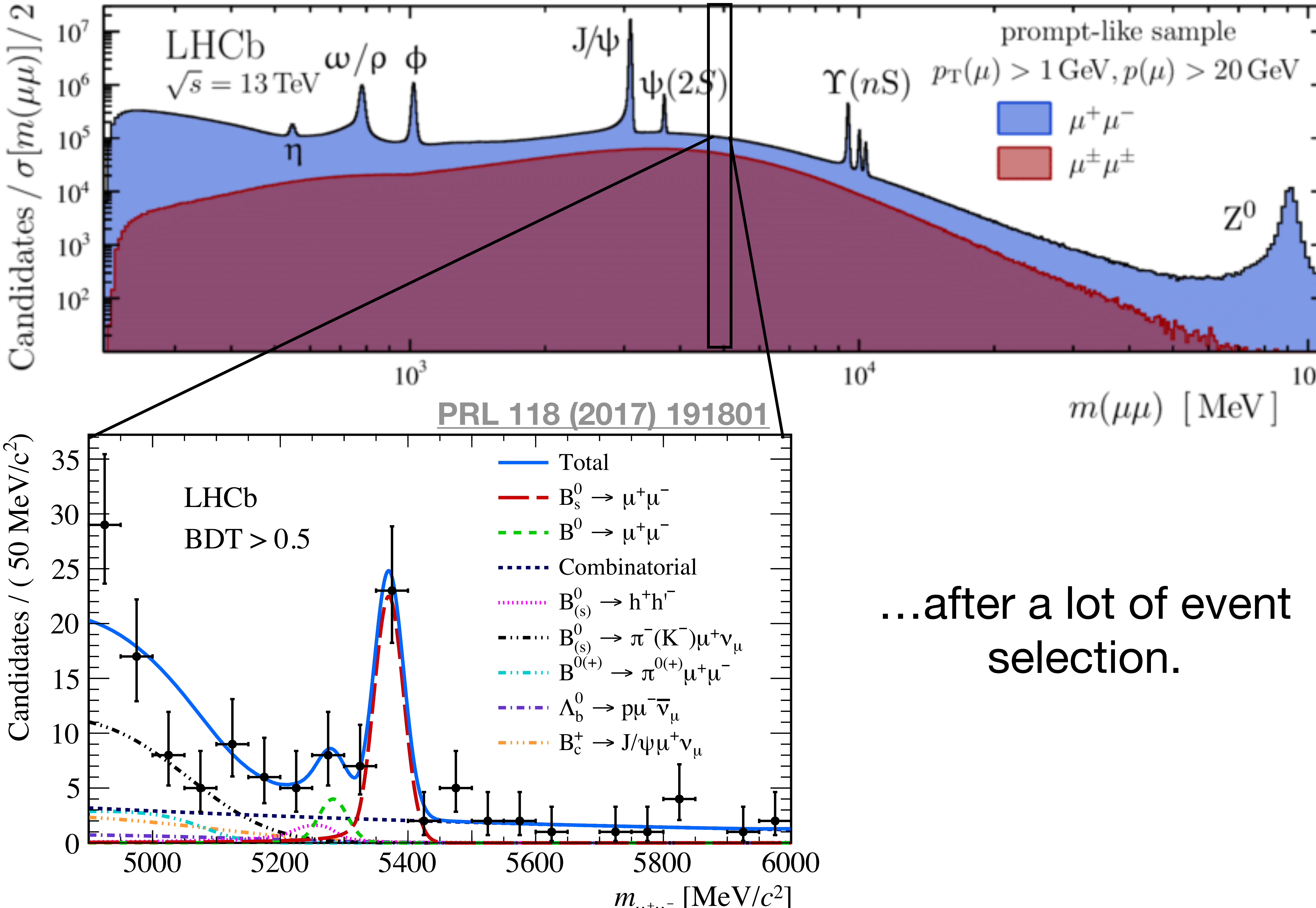
PRL 120 (2018) 061801



For example, muons

PRL 120 (2018) 061801

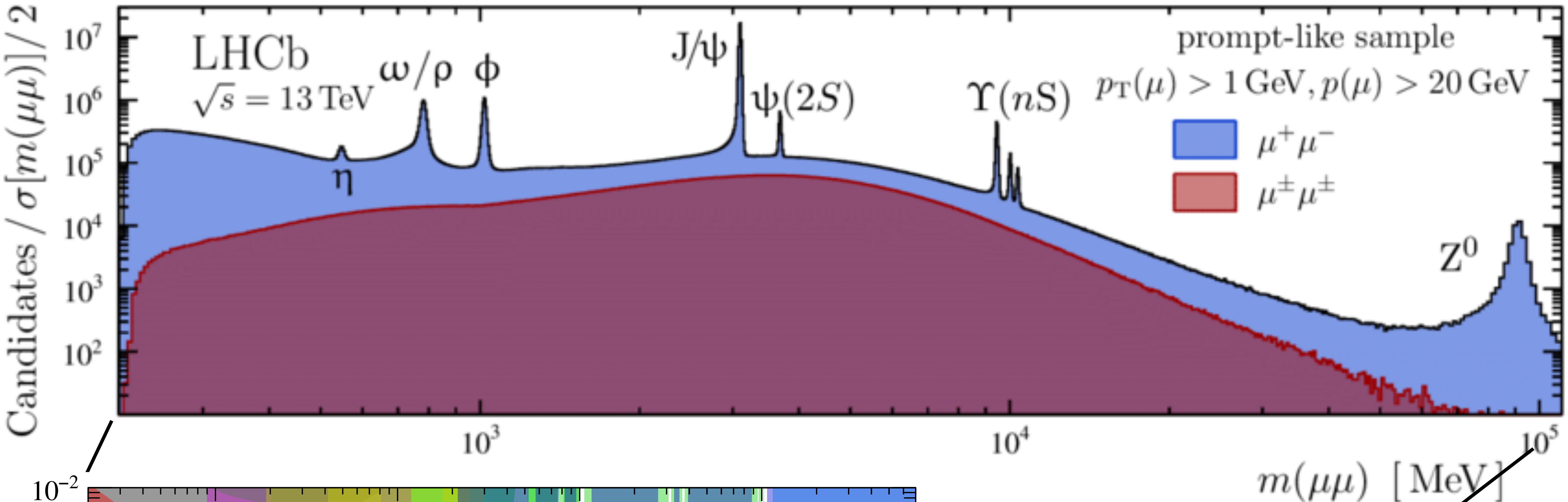
PRL 118 (2017) 191801

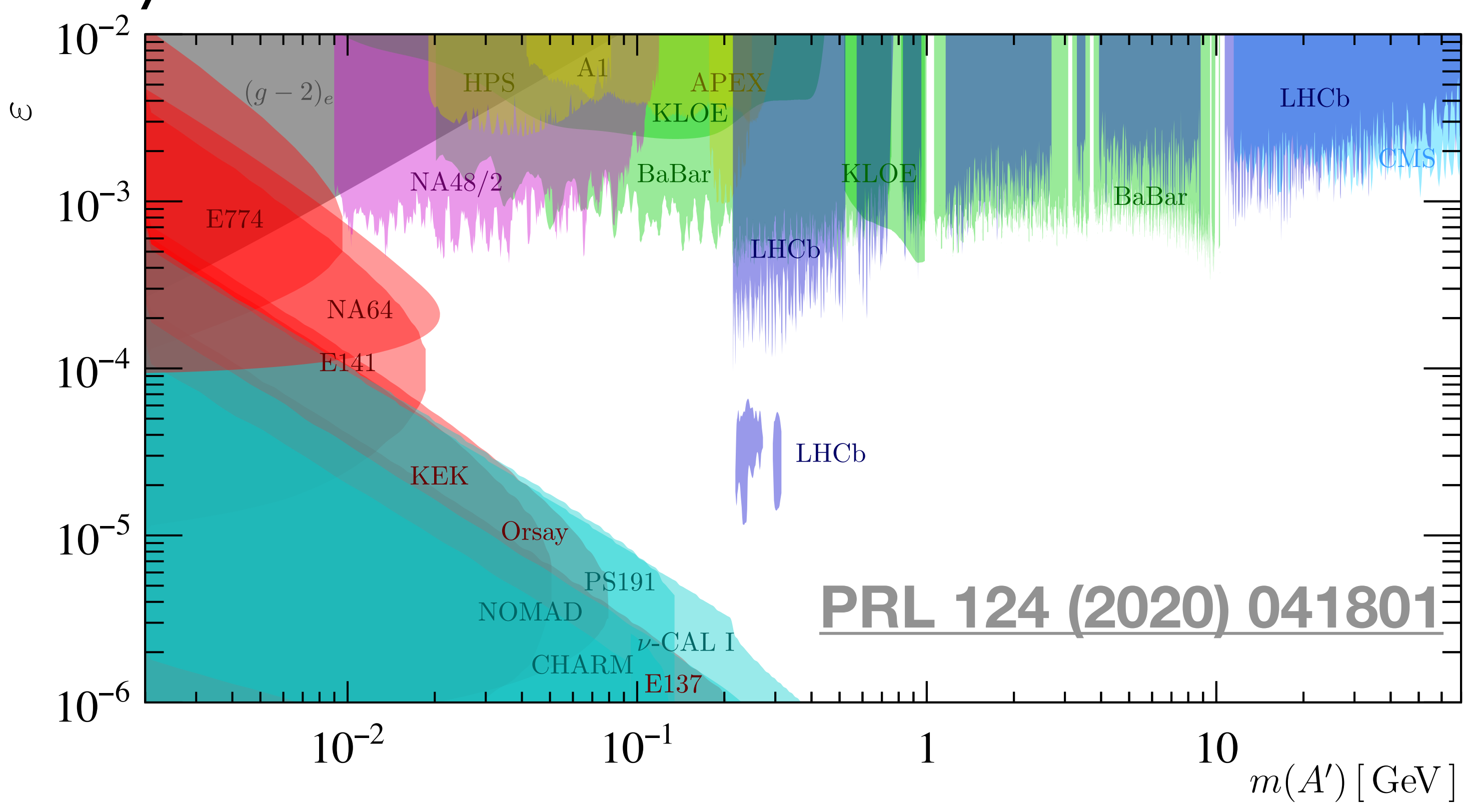


…after a lot of event selection.

For example, muons

PRL 120 (2018) 061801



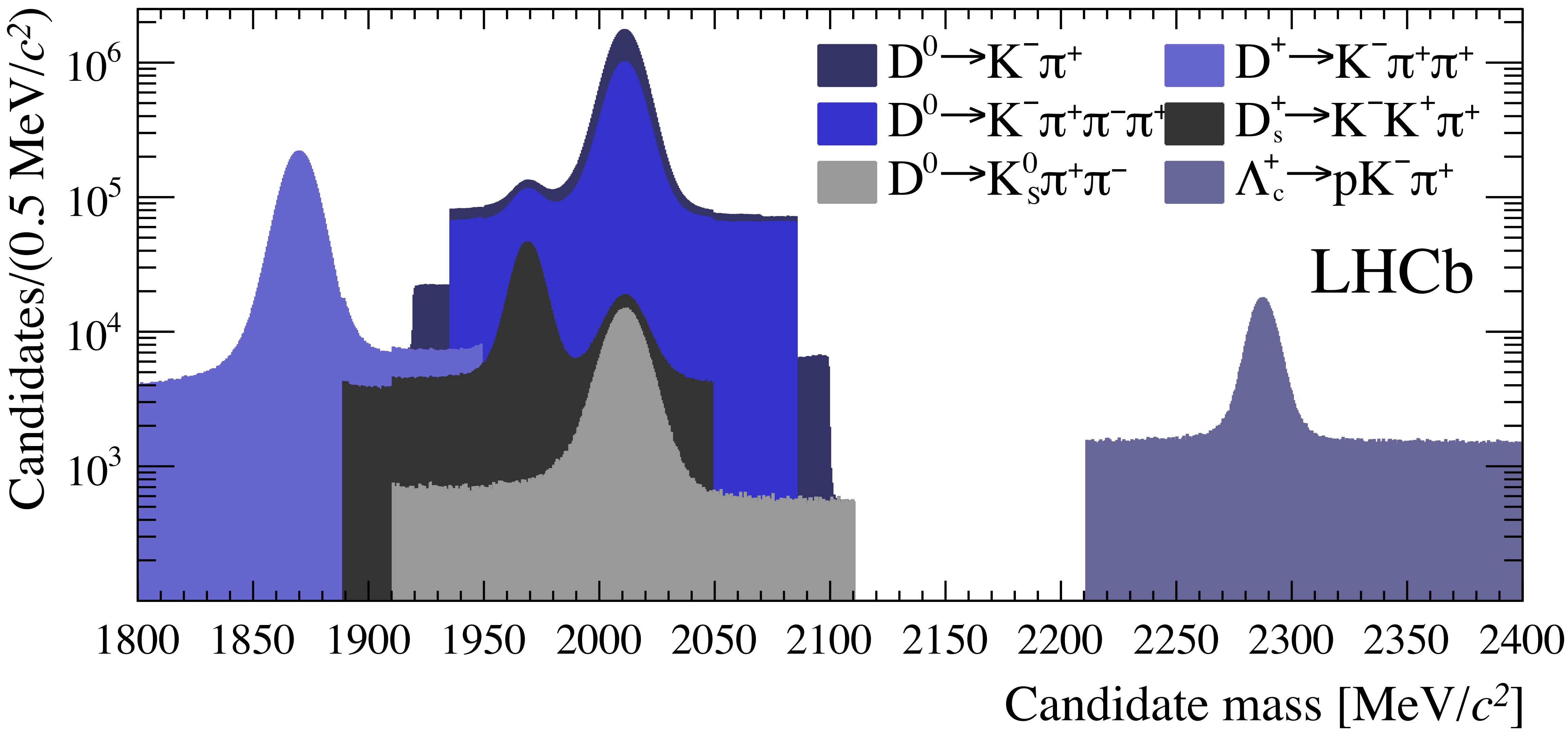


PRL 124 (2020) 041801

…after analysing a small amount of information from ~all events!

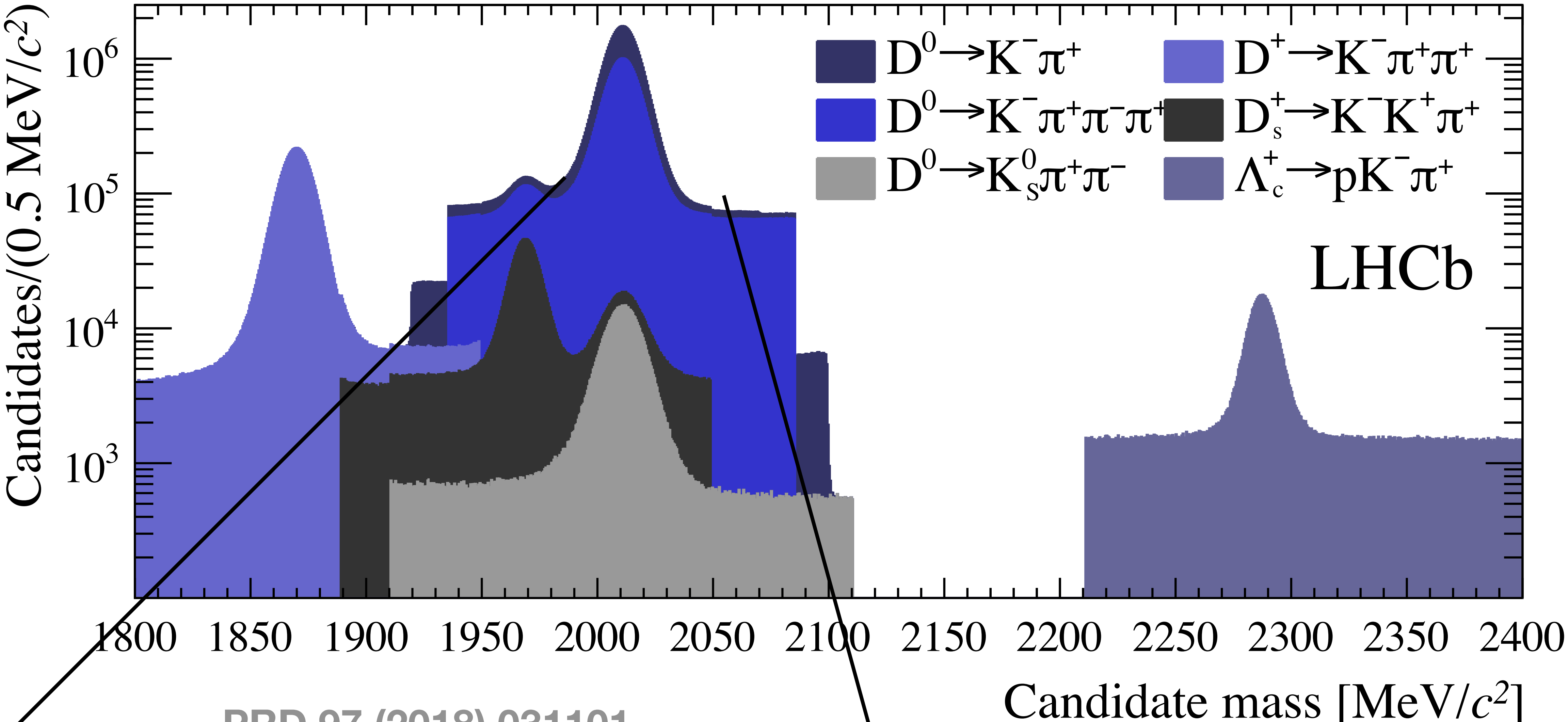
Another example, charm

JINST 14 (2019) P04013

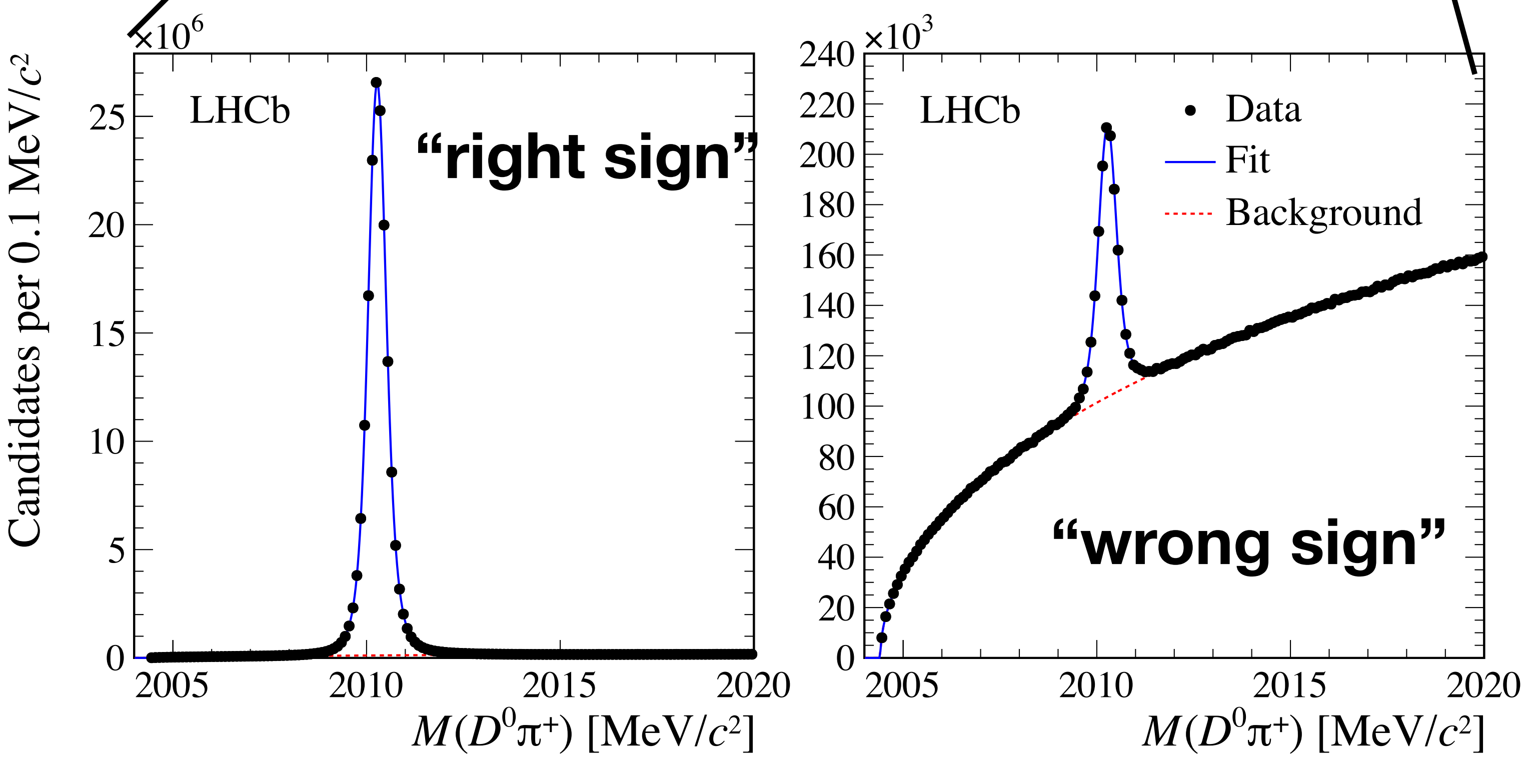


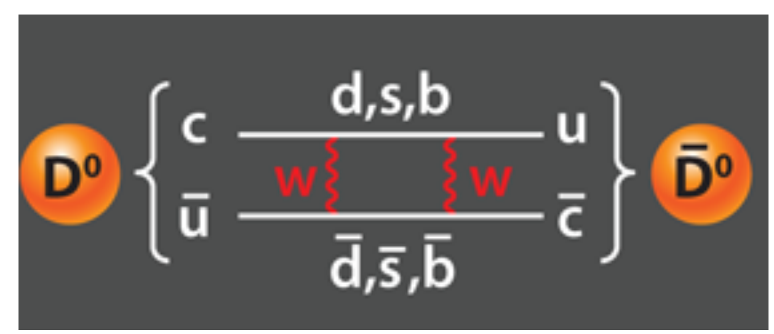
Another example, charm

JINST 14 (2019) P04013



PRD 97 (2018) 031101





…after reconstructing very low momentum "slow pions"

Real time analysis

$$\boxed{\text{Data rate} \sim \text{event size} * \text{event rate}}$$

1. Aggressive reduction required quickly and early

   Requires “offline quality” alignment, calibration and reconstruction in HLT2, including RICH PID etc…

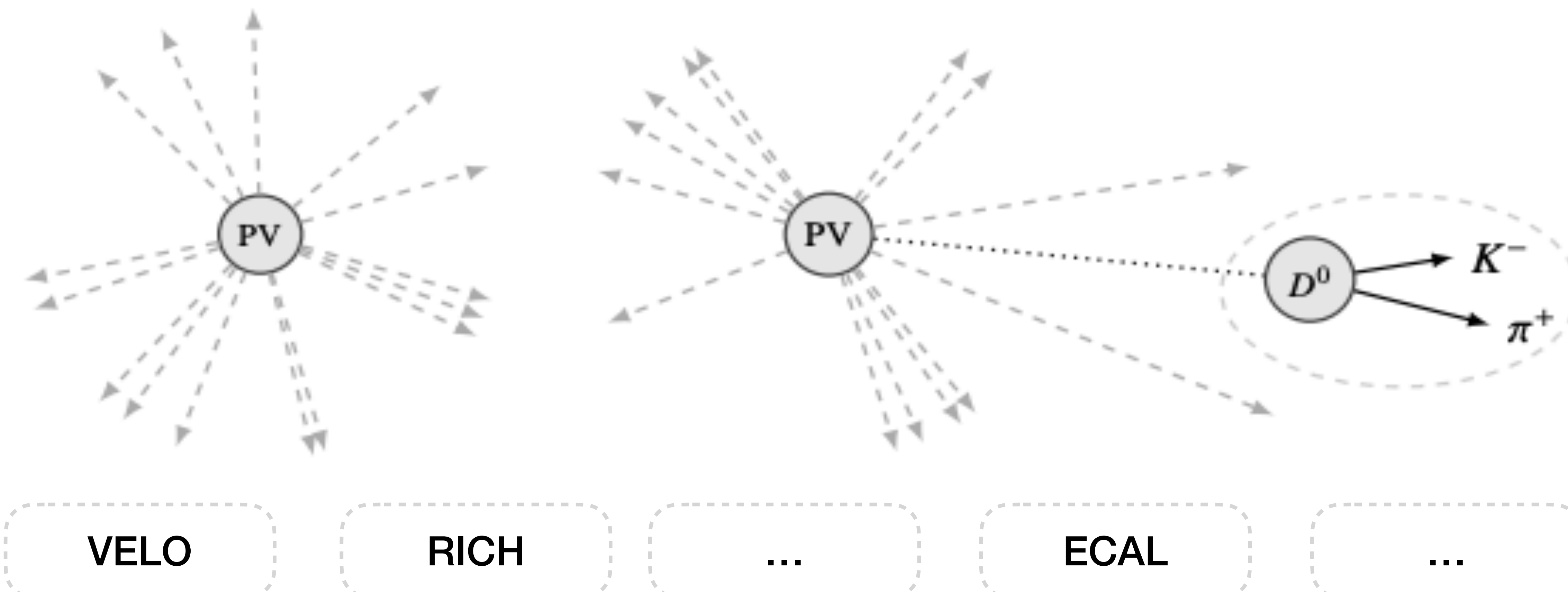
2. Flexibility w.r.t. *how* to reduce the data

   Evolution of LHCb’s “Turbo” stream

JINST 14 (2019) P04006
***JINST* 14 (2019) P04013**
Comp. Phys. Coms. 208 (2016) 35-42,

Turbo stream evolution in Run-II

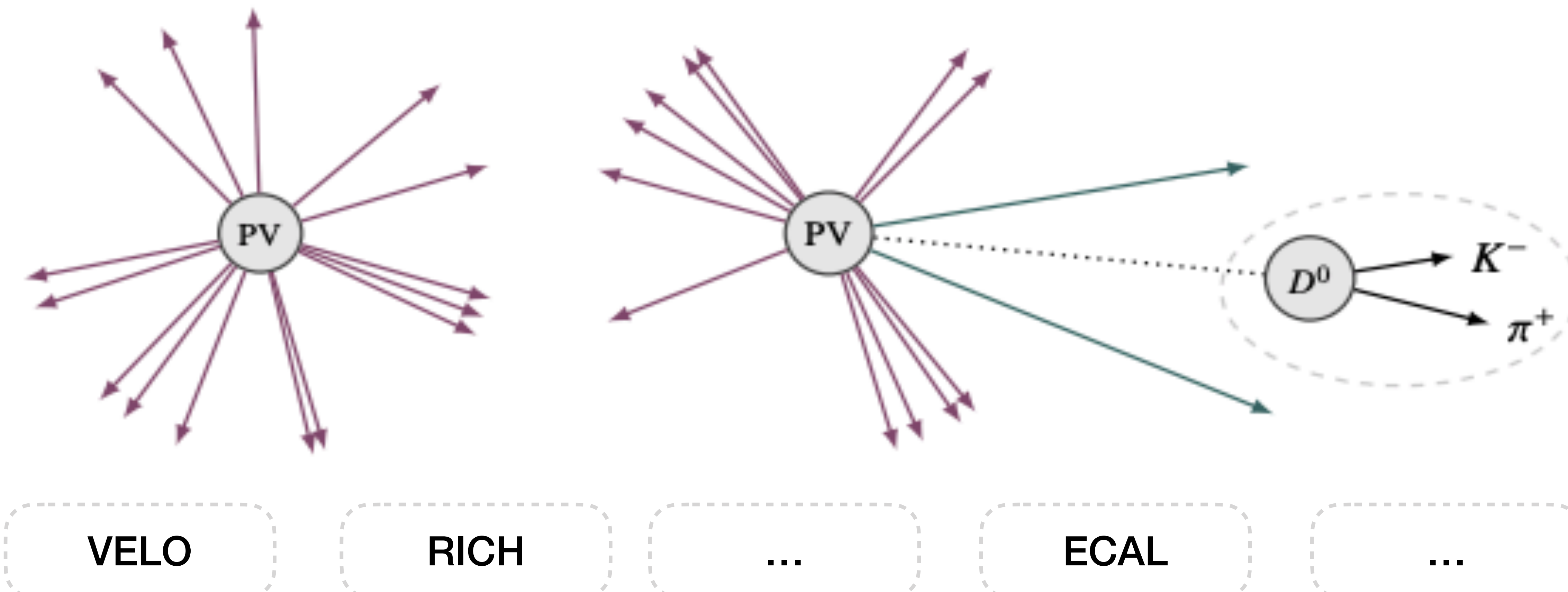
JINST 14 (2019) P04006



2015 - Orignal Turbo scheme where we persist, in this example, the D→Kπ candidate

Turbo stream evolution in Run-II

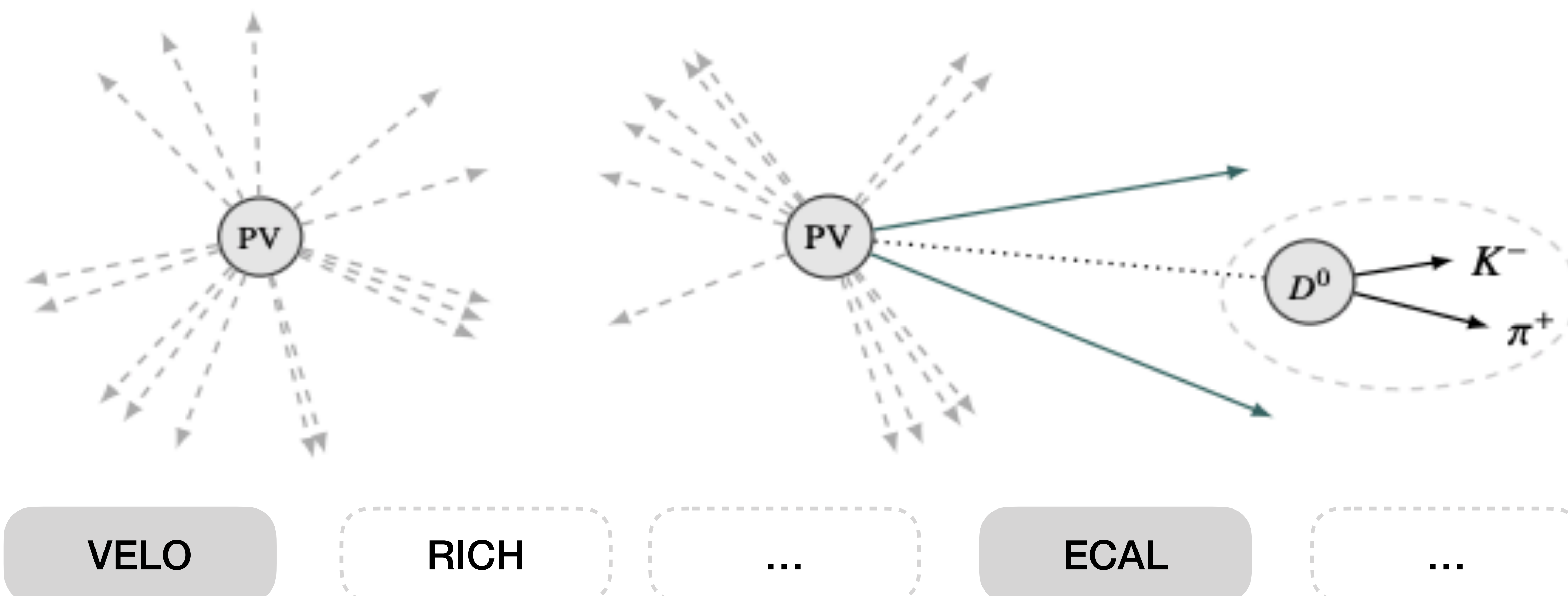
JINST 14 (2019) P04006



2016 - Full persistence scheme - all of the reconstructed event.

Turbo stream evolution in Run-II

JINST 14 (2019) P04006



2017 - Selective persistence scheme - pick and choose raw and reconstructed data.

Turbo stream evolution in Run-II

Persistence	Since	Average event size
Turbo [1]	2015	7 kB
Selective persistence [2]	2017	15 kB
Complete persistence	2016	50 kB
Raw event	2010	70 kB

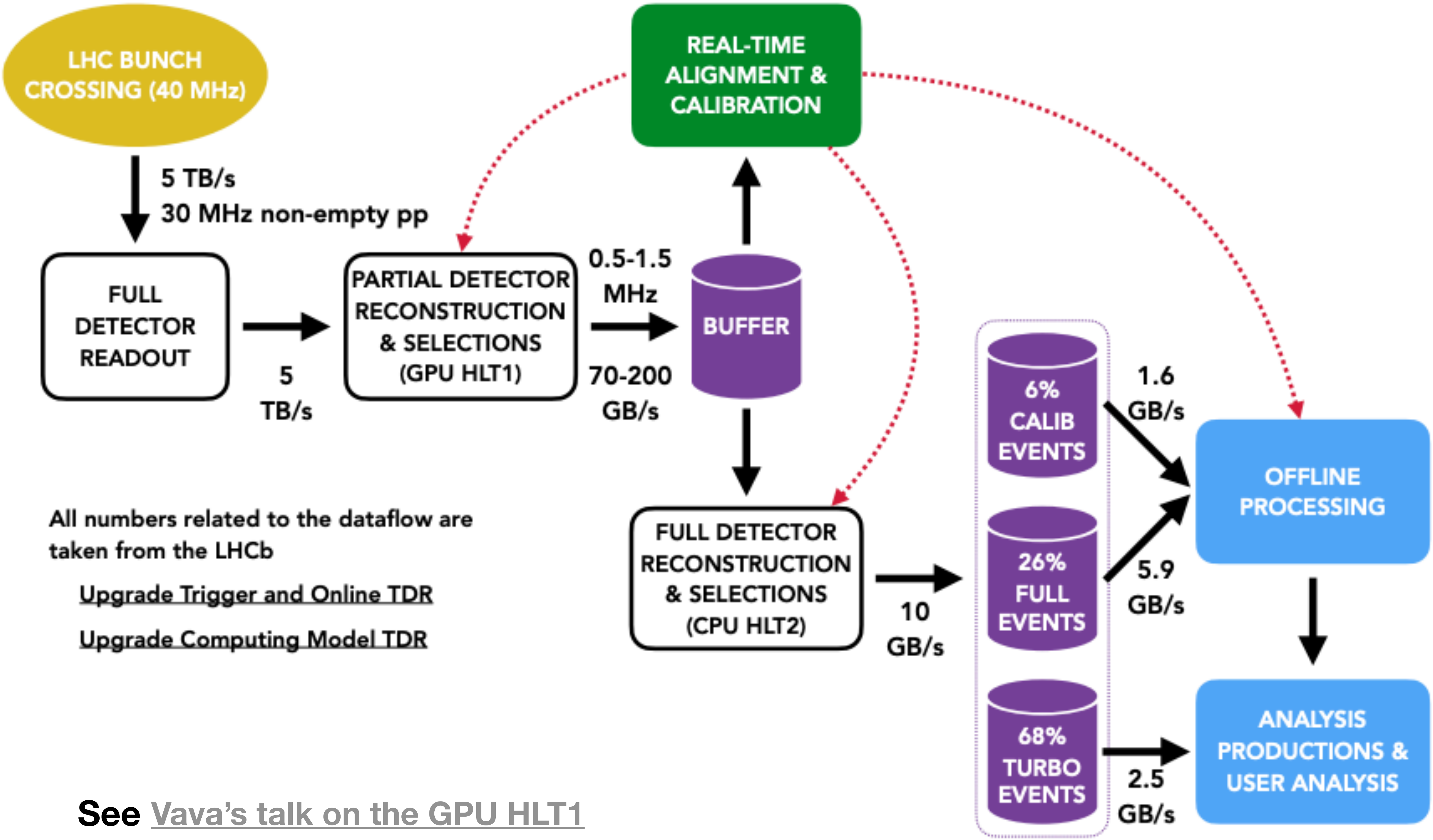
By the end of Run-II:

- 528 HLT2 lines,
- ≈50% of which were Turbo,
- Taking 25% of the rate for 10% of the bandwidth

1. Comp. Phys. Coms. 208 (2016) 35-42
2. JINST 14 (2019) P04006

LHCb upgrade data flow diagram

LHCB-FIGURE-2020-016

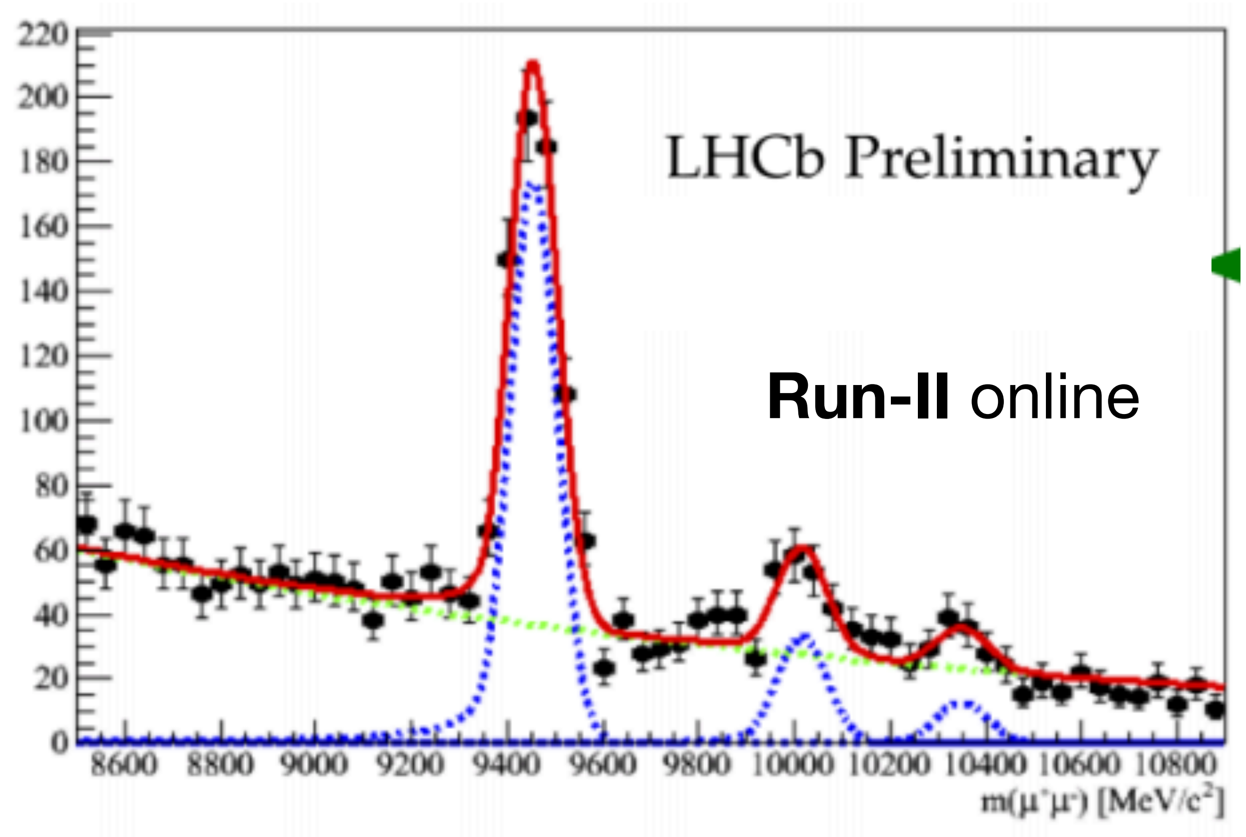


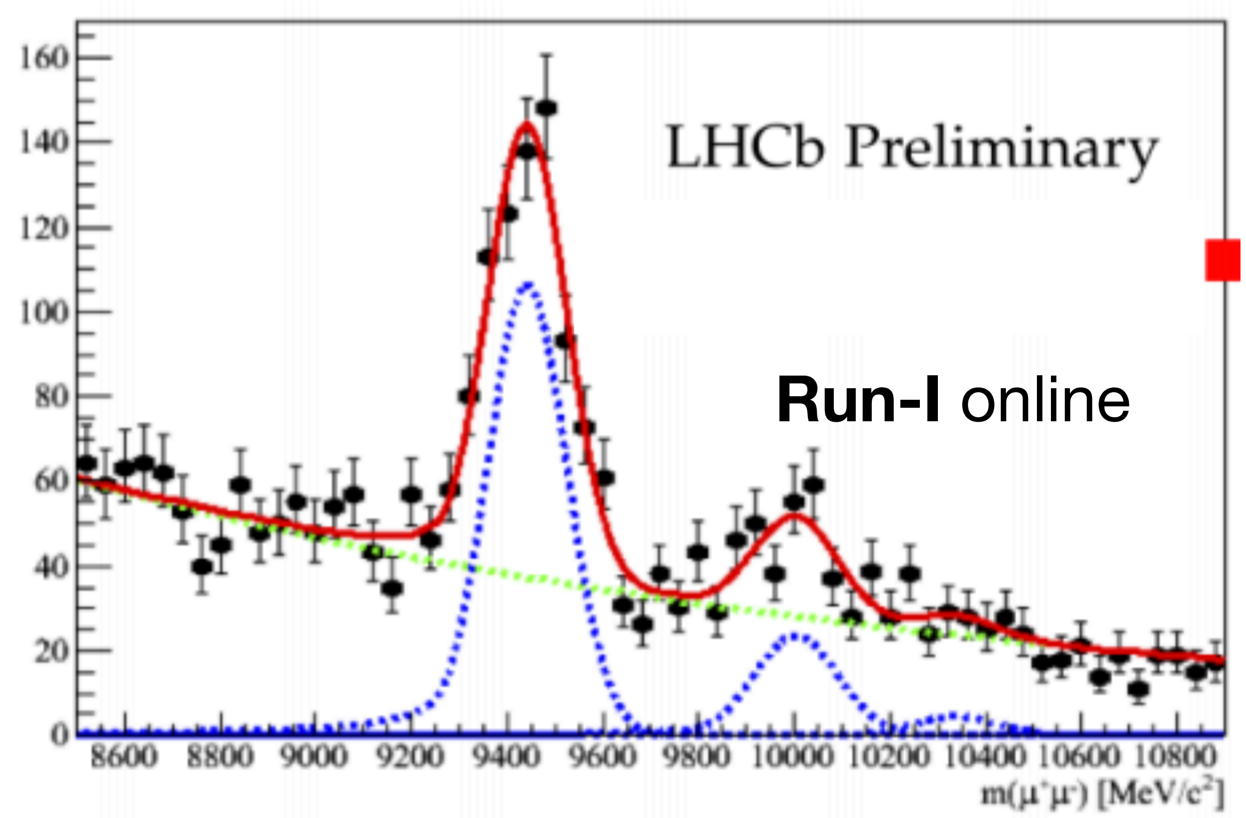
All numbers related to the dataflow are taken from the LHCb

- Upgrade Trigger and Online TDR
- Upgrade Computing Model TDR

See Vava's talk on the GPU HLT1

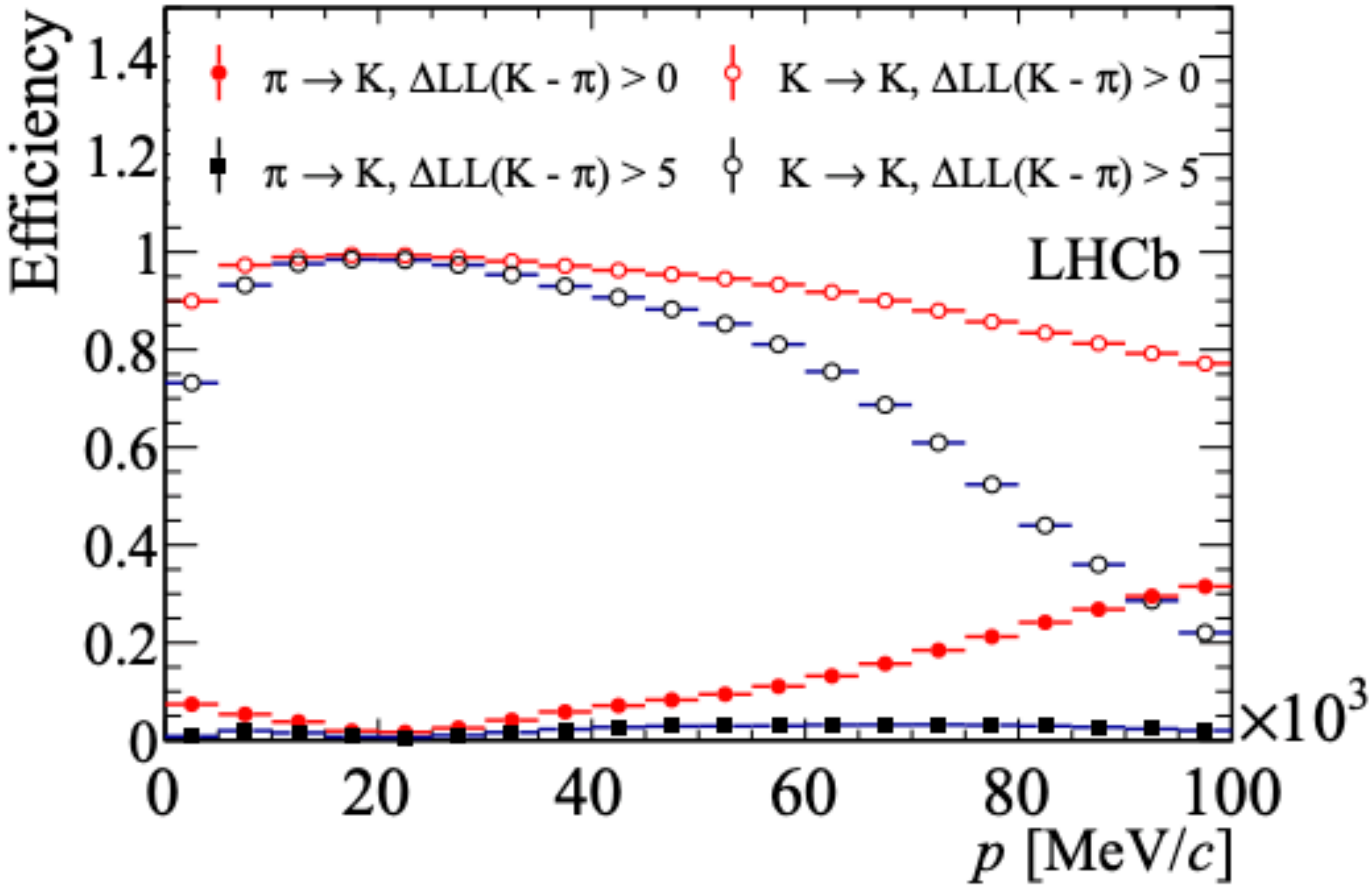
The need for offline quality alignment and calibration





E.g., offline quality RICH PID for HLT2

JINST 14 (2019) P04013

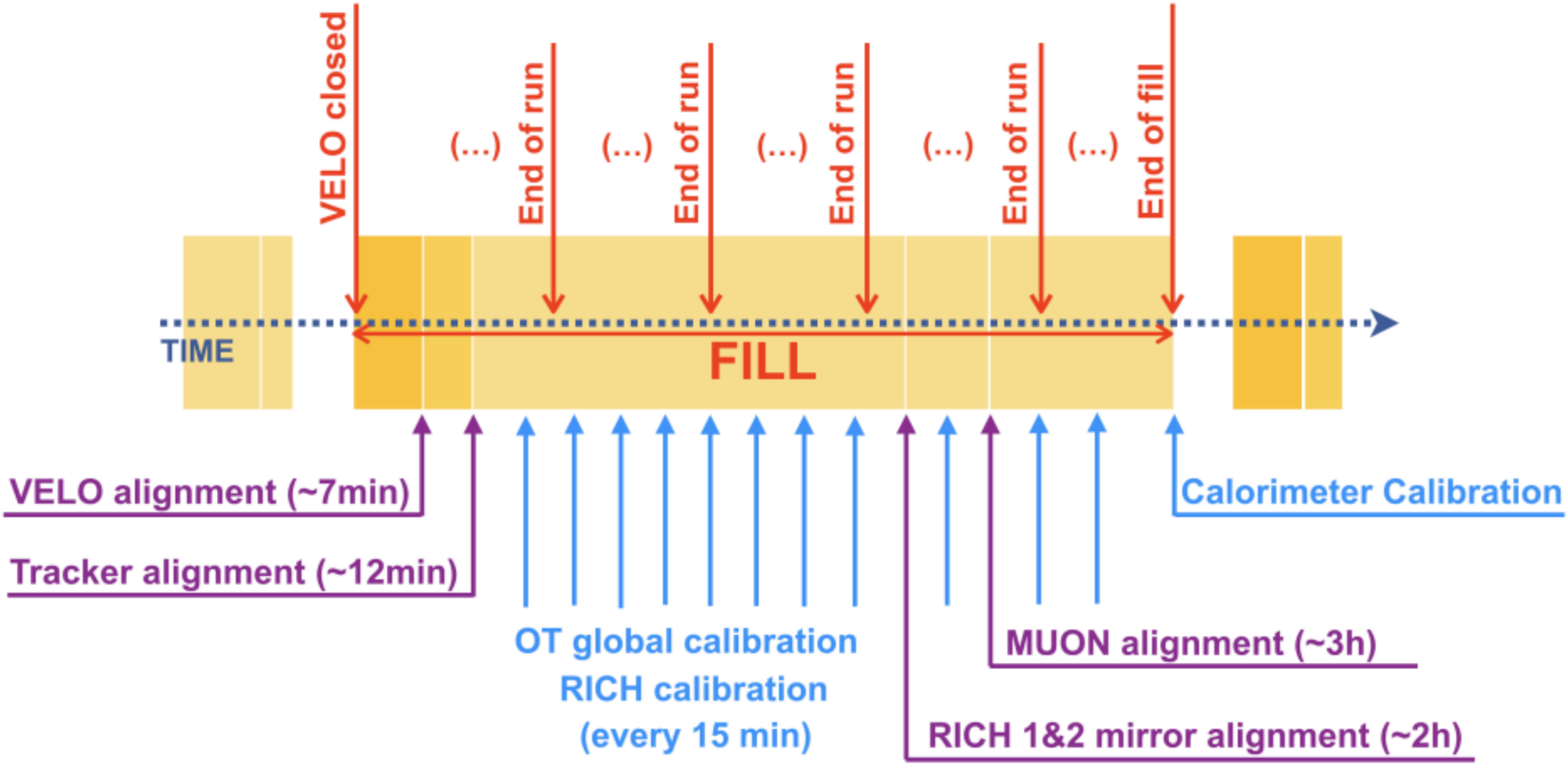


Performance slightly *better* than the offline version from Run-I.

RICH PID is a crucial requirement for the Turbo stream.

Real time alignment and calibration

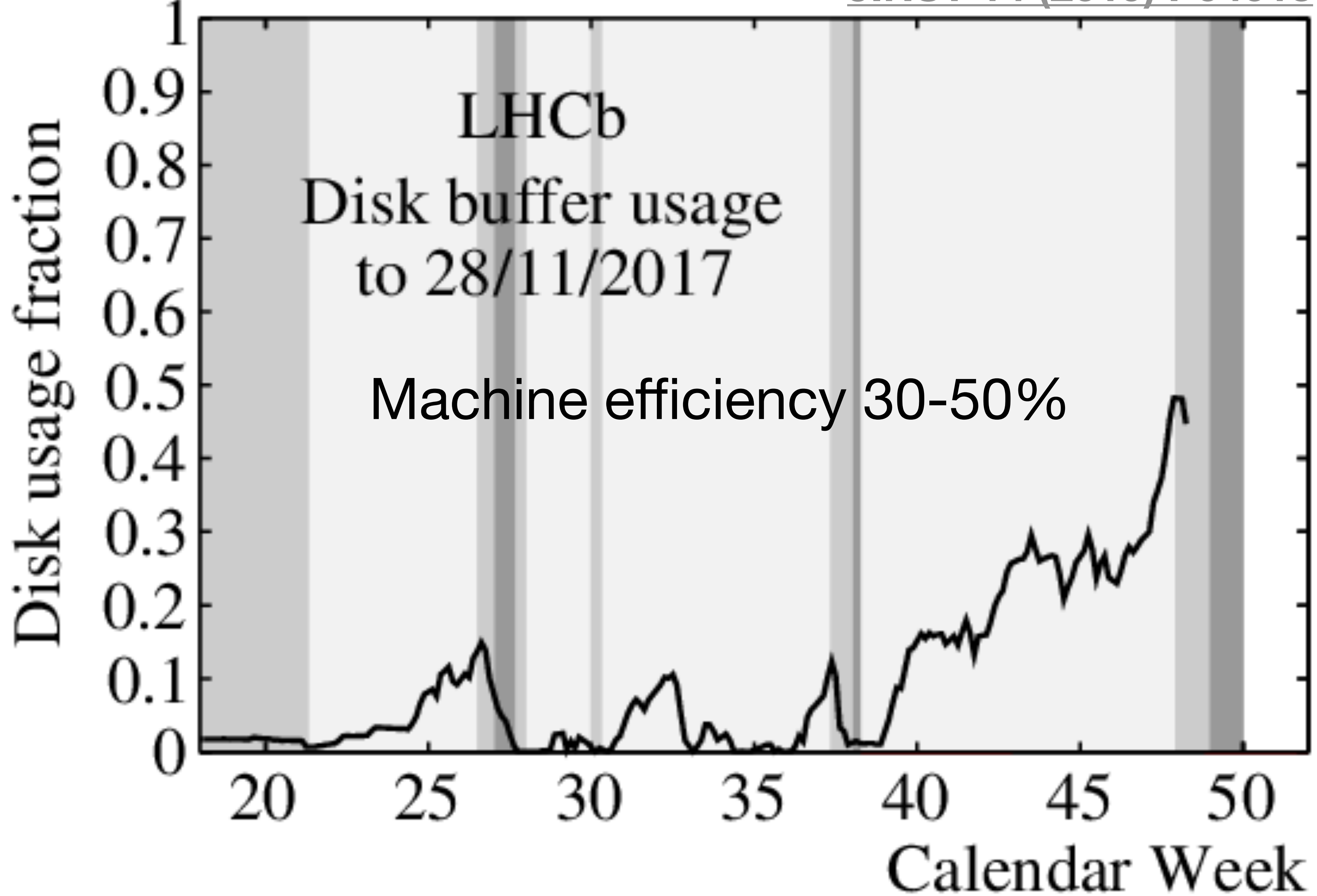
JINST 14 (2019) P04013



((~7min),(~12min),(~3h),(~2h)) - time needed for both data accumulation and running the task

Buffering in Run-II

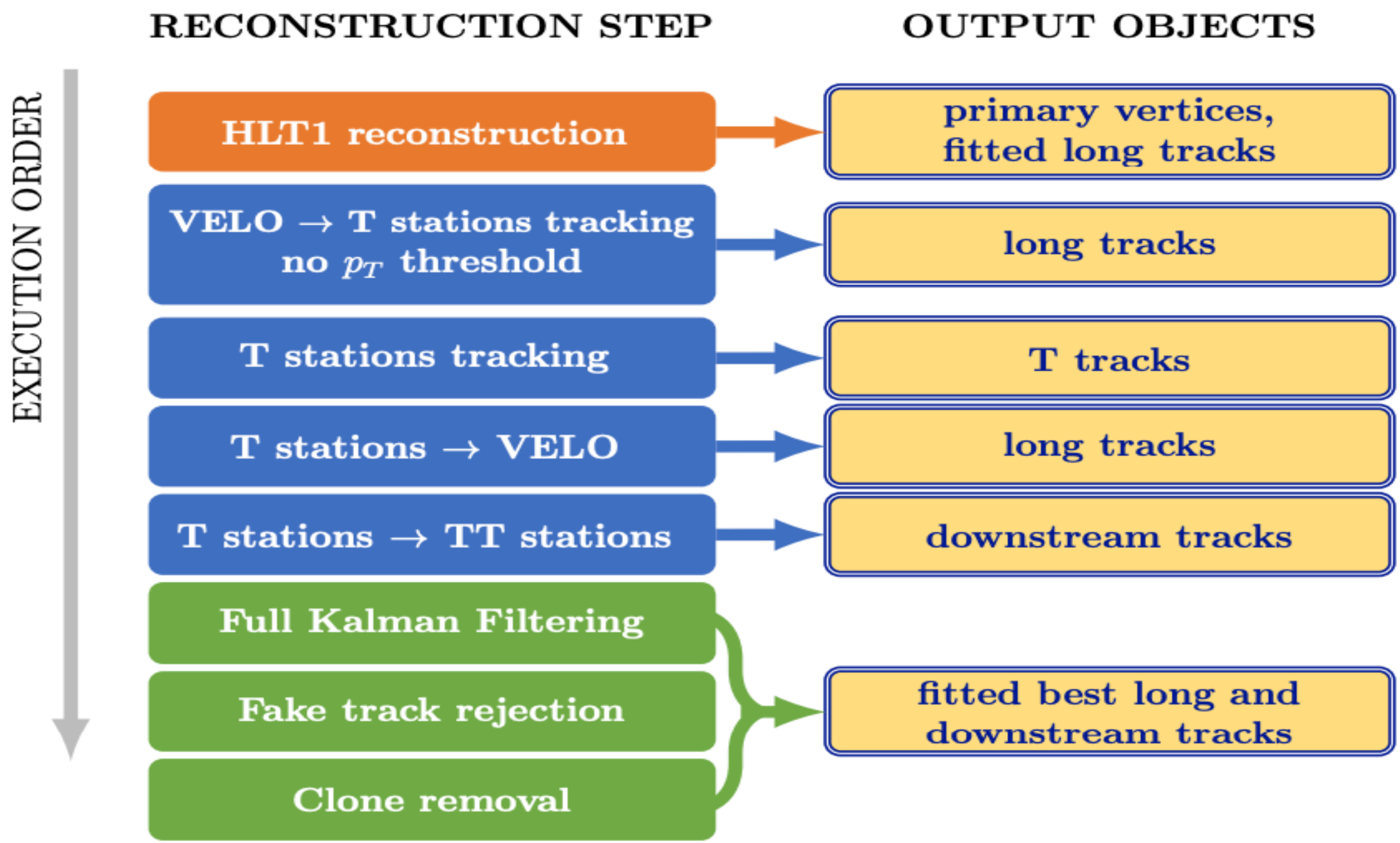
JINST 14 (2019) P04013



- Run-II HLT farm: ~50k x86 cores
- HLT2 throughput ~80 kHz out of fill
- HLT1 output ~ 150 kHz

Full offline quality reconstruction in HLT2

JINST 14 (2019) P04013



Plus full upfront RICH and CALO reconstruction

Aim for same/better in Run-3
How fast does it need to be?

Getting to O(500 kHz) HLT2 throughput

1. Multithreading

   Task based scheduler in Gaudi, and major effort to port algorithms for thread safety.

2. Vectorisation

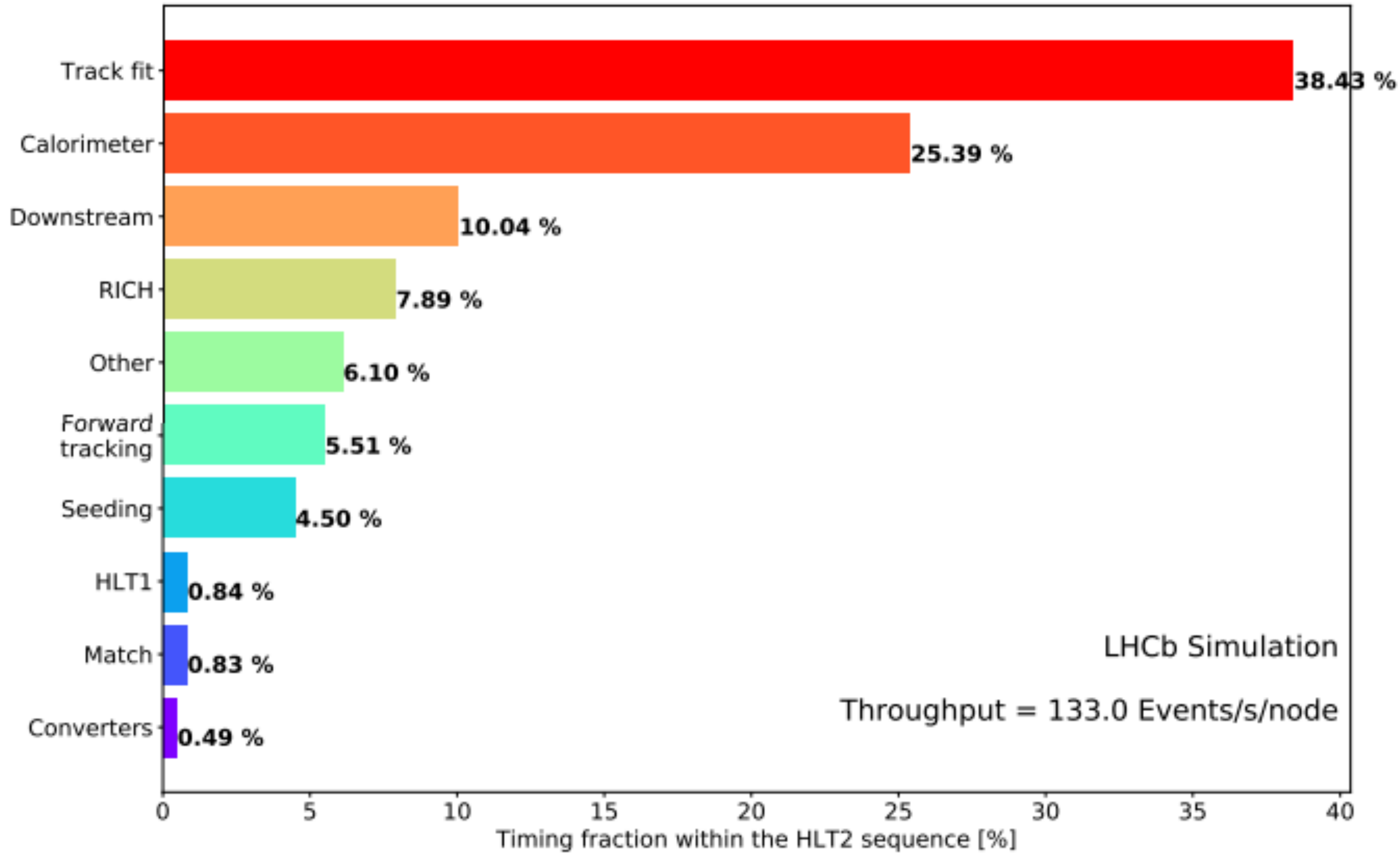
   E.g. VELO JINST 15 (2020) 06, P06018

3. Simplifications/approximations where the impact on physics performance is negligible or tolerable.

   E.g. parameterised Kalman fit 2101.12040

State of HLT2 (~April 2020)

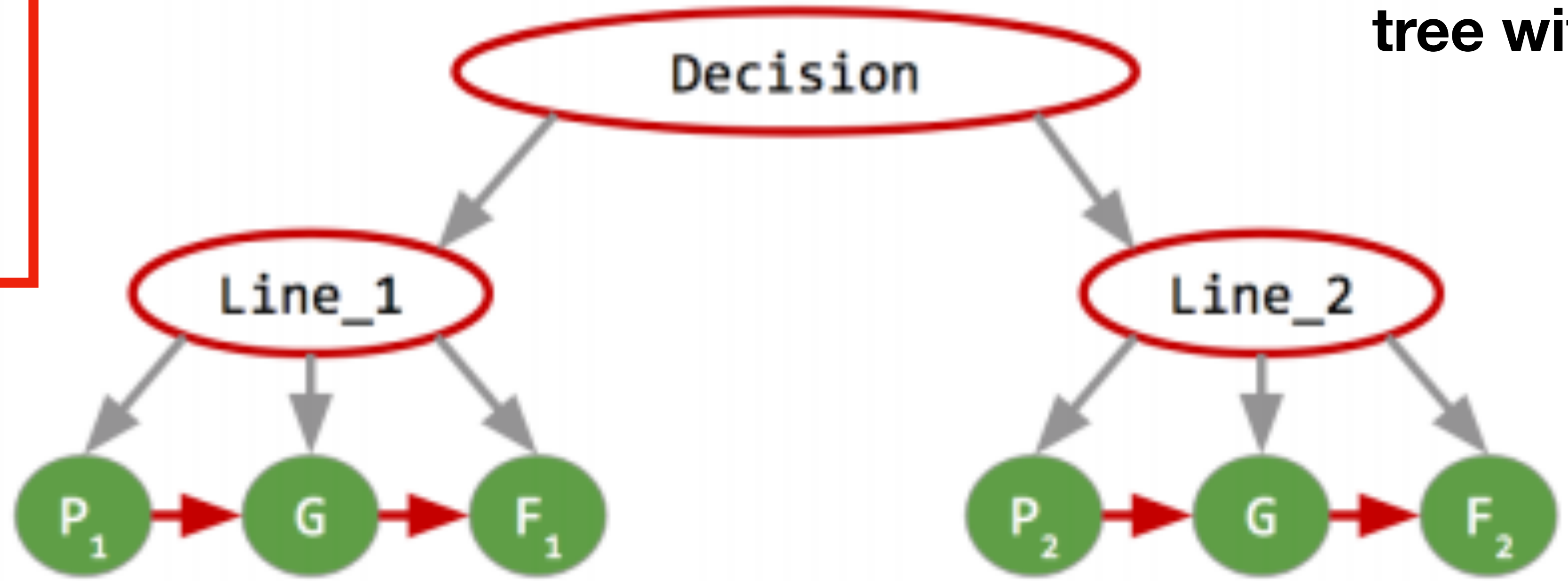
LHCb-FIGURE-2020-007



Managing O(1k) HLT2 selections

LHCb-PROC-2020-003

```
LAZY_AND
NON_LAZY_AND
LAZY_OR
NON_LAZY_OR
NOT
```



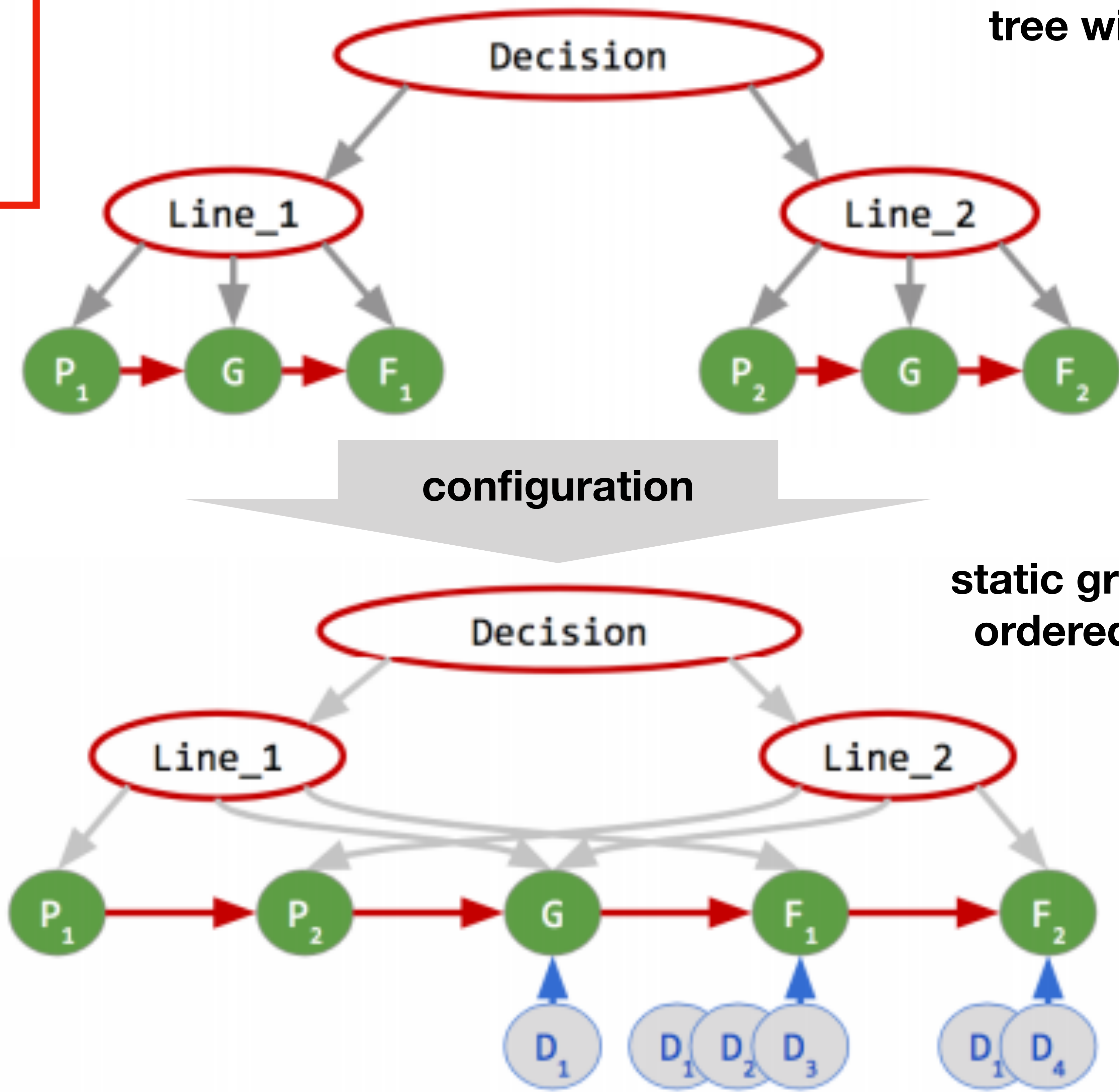
Example dependency tree with two lines

Managing O(1k) HLT2 selections

LHCb-PROC-2020-003

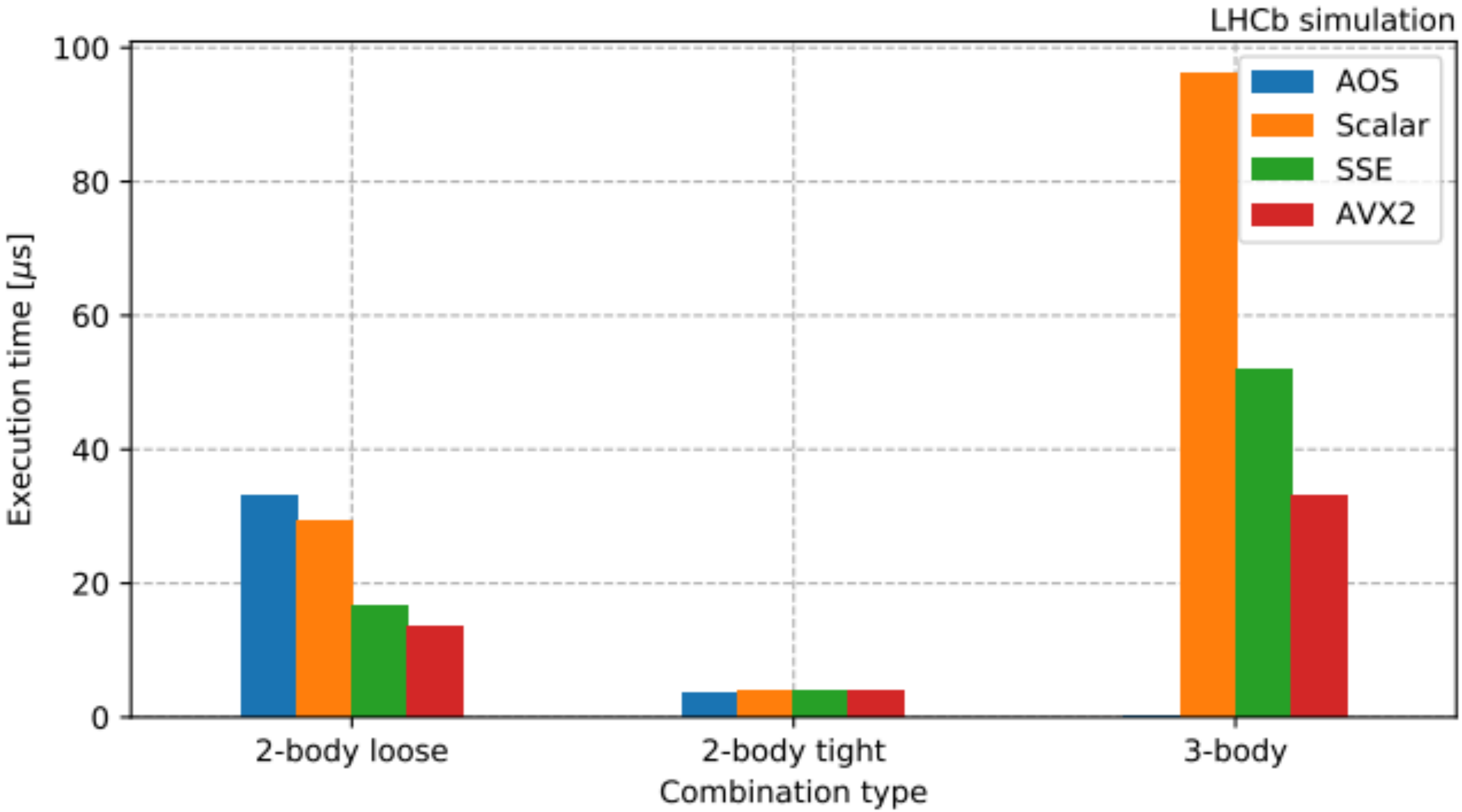
```
LAZY_AND
NON_LAZY_AND
LAZY_OR
NON_LAZY_OR
NOT
```

Example dependency tree with two lines

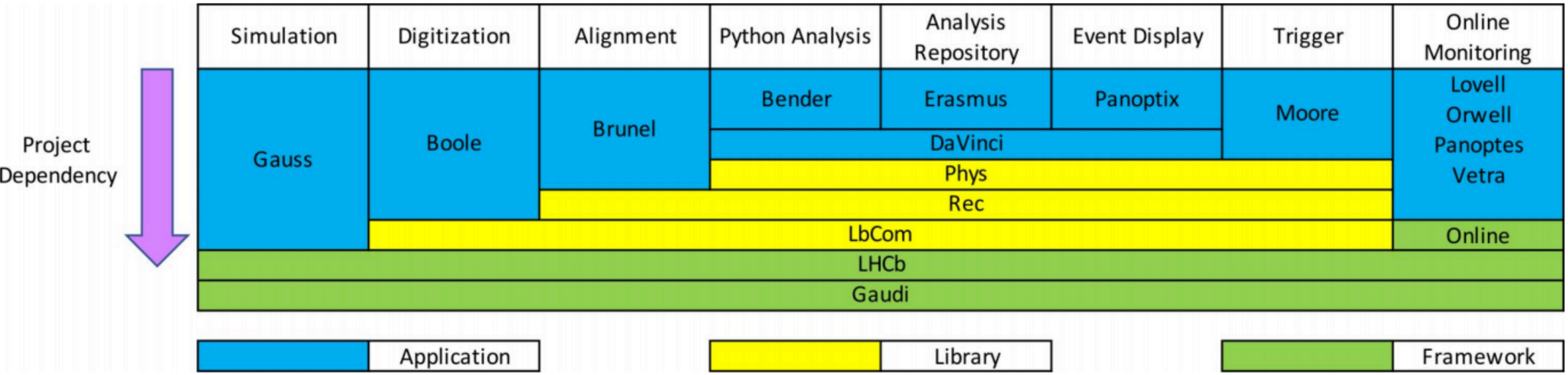


static graph with ordered nodes

Vectorised selections LHCB-FIGURE-2020-018

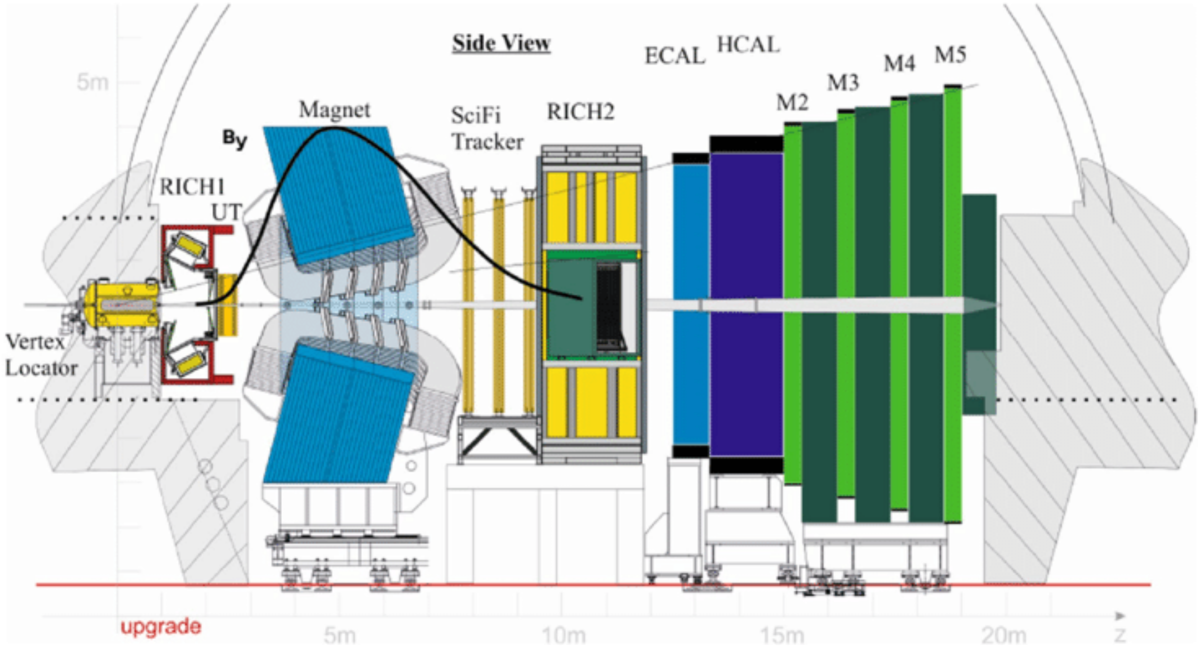


Software development and testing



- Migration to gitlab in early 2016, but usage has evolved for Run-3.
- Formal code review coordinated by “RTA shifters” with support from [RTA project] “maintainers”.
- Larger suite of regression tests, quantifying physics performance and throughput.

Conclusions



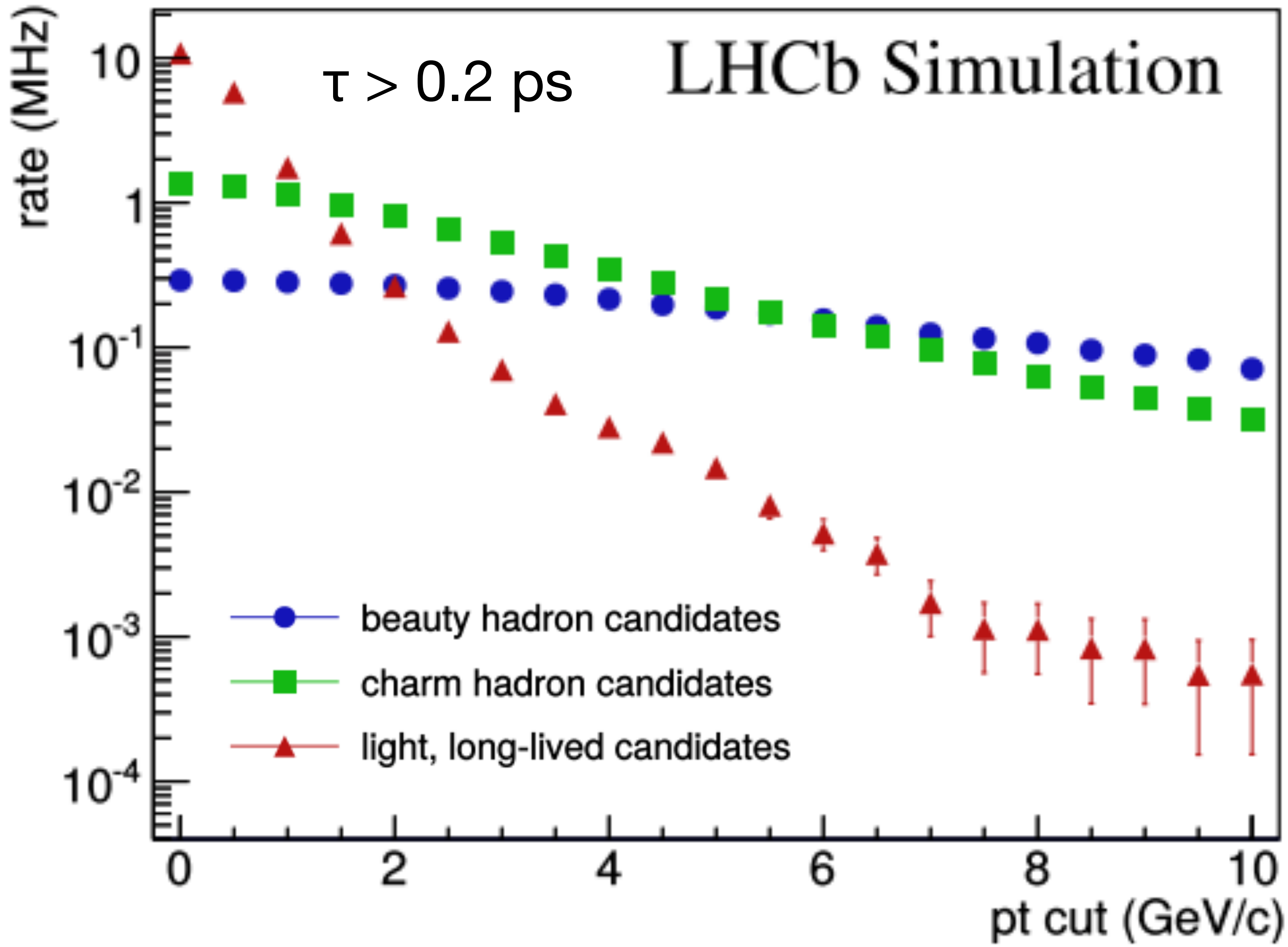
🤞lots of physics

😬lots of data to process

Ambitious real time analysis scheme is on track thanks to huge effort and innovation with the design of software and algorithms.

Backup slides start here

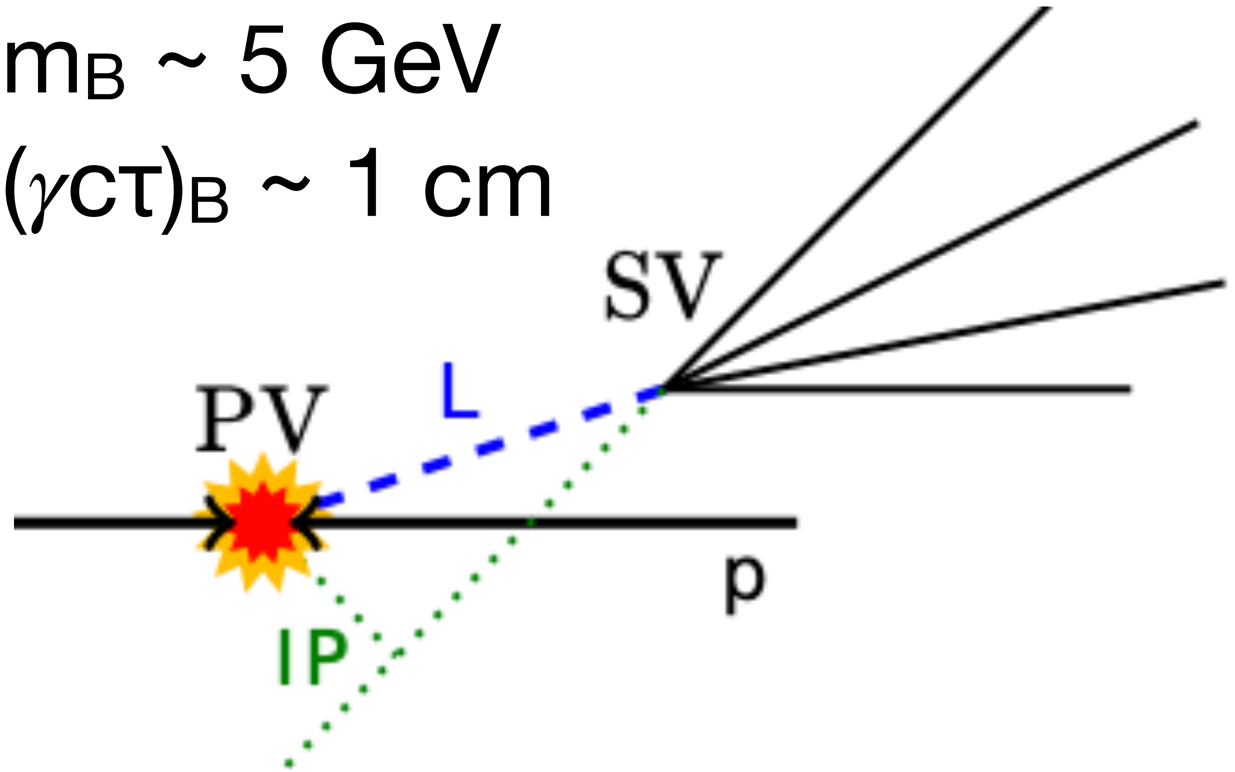
Upgrade I environment



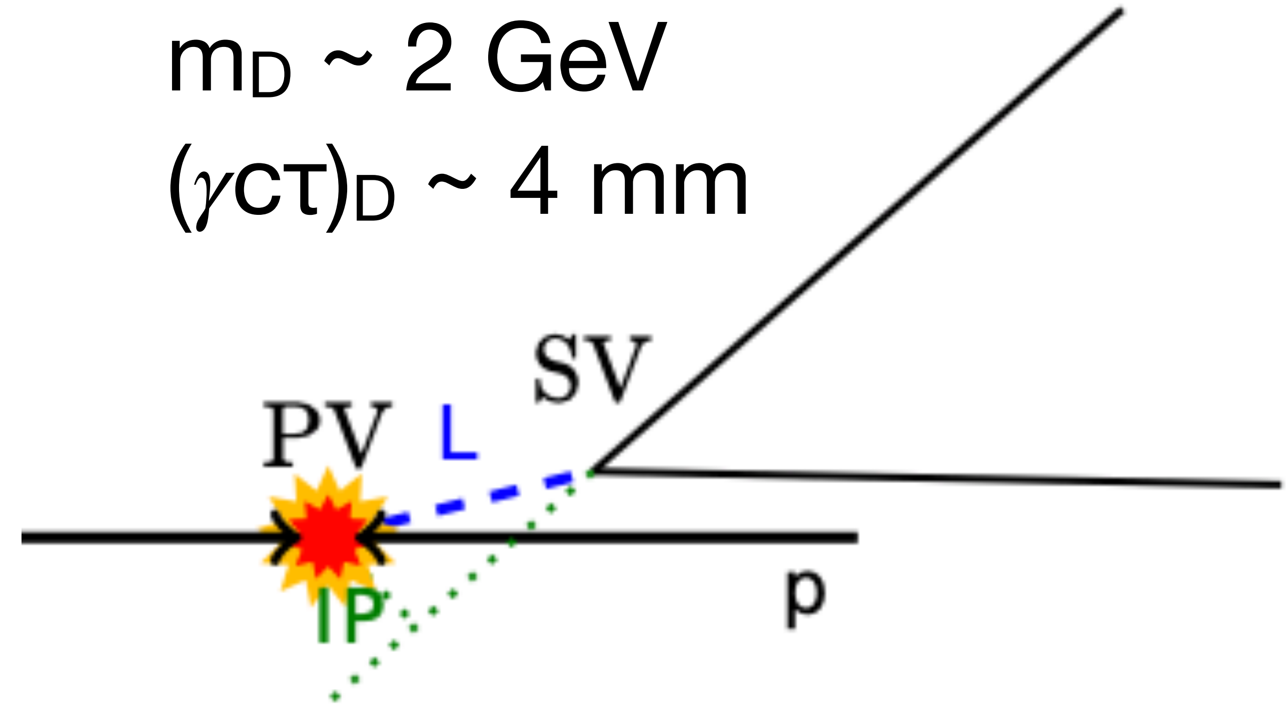
Fitzpatrick and Gligorov, Anatomy of an upgrade event in the upgrade era, and implications for the LHCb trigger, LHCb-PUB-2014-027

Physics goals → requirements for trigger

$m_B \sim 5$ GeV
$(\gamma c\tau)_B \sim 1$ cm



$m_D \sim 2$ GeV
$(\gamma c\tau)_D \sim 4$ mm



(within a broad program including W, Z, Onia, etc… production, light-exotics, ions and fixed target physics…)

Beauty and charm signatures aren't well suited to typical low-level triggers.

Run at levelled luminosity of 4×10^{32} $cm^{-2}s^{-1}$ with 1 MHz triggered readout.

HLT2 selections in Run-II

JINST 14 (2019) P04013

LHCB-TDR-018

stream	event size (kB)	event rate (kHz)	rate fraction	throughput (GB/s)	bandwidth fraction
FULL	70	7.0	65%	0.49	75%
Turbo	35	3.1	29%	0.11	17%
TurCal	85	0.6	6%	0.05	8%
total	61	10.8	100%	0.65	100%