Analysis at LHCb
Environment and general matters

Eduardo Rodrigues
University of Cincinnati

HSF Data Analysis WG – Analysis Requirements Jamboree, CERN, 23rd Jan. 2019

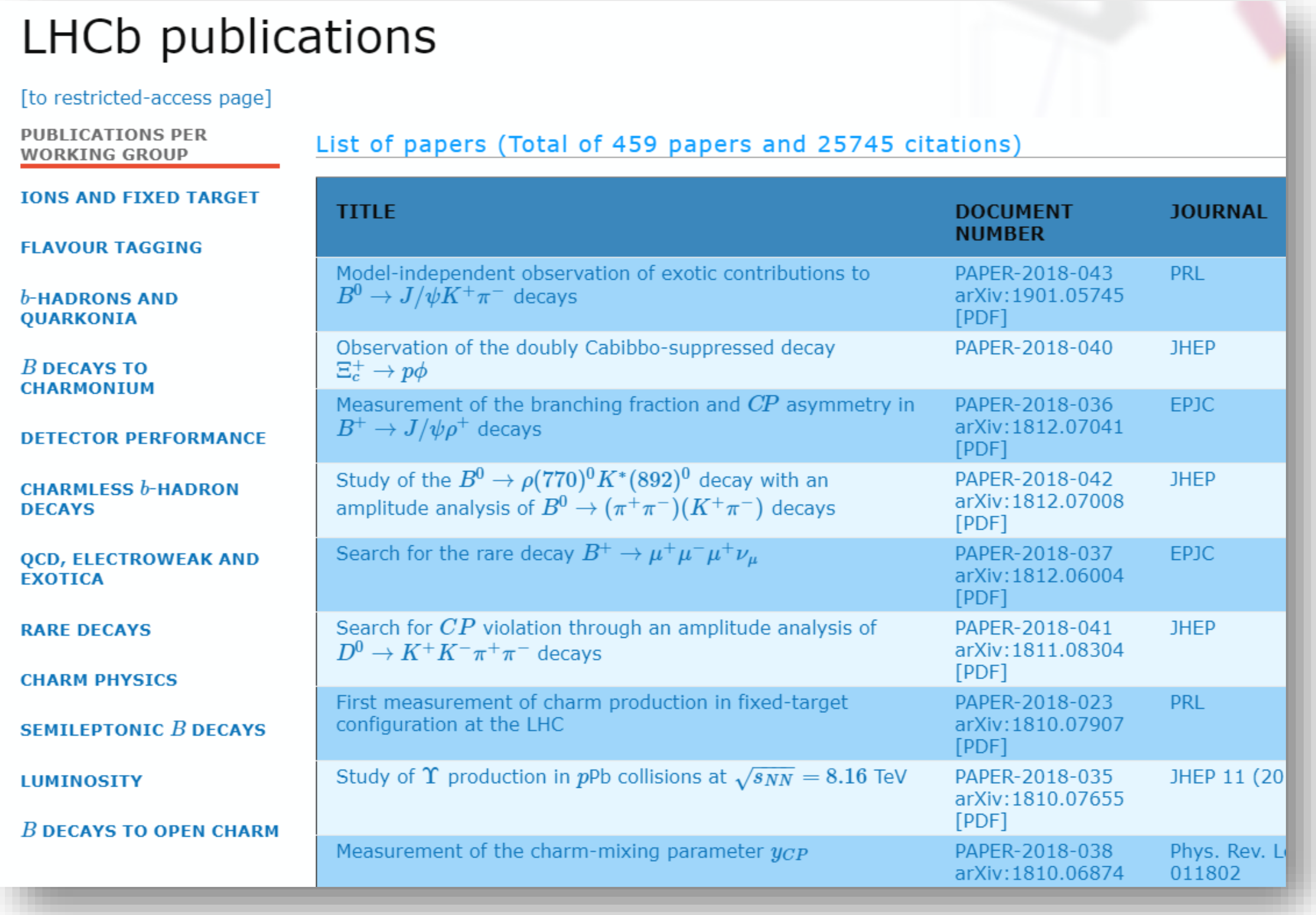
Thank you to the LHCb colleagues who provided feedback !

Setting the scene – set of LHCb physics (& phys. performance) WGs

❑ We have a large number of physics Working Groups

❑ Analysis “environments” are not the same: Some WGs deal with very rare decays, others analyse millions or even billions of signal candidates !

❑ Requirements are hence rather diverse



LHCb publications

[to restricted-access page]

PUBLICATIONS PER WORKING GROUP

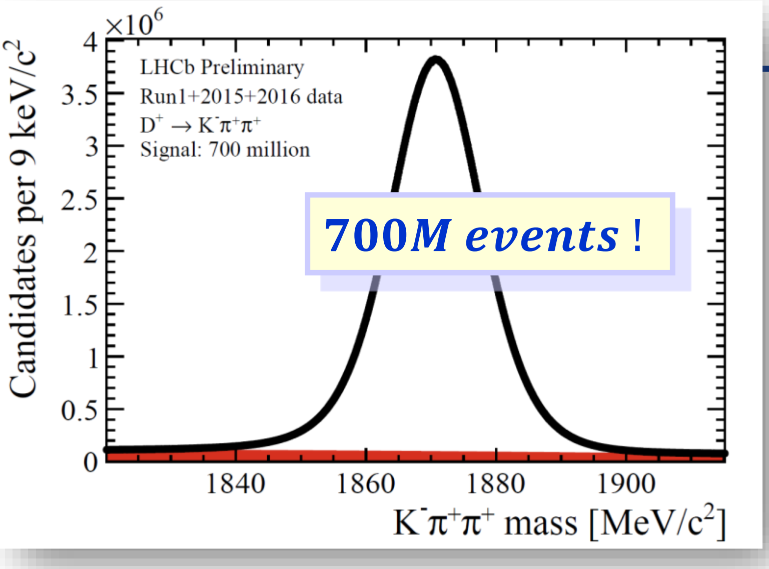
- IONS AND FIXED TARGET
- FLAVOUR TAGGING
- b-HADRONS AND QUARKONIA
- B DECAYS TO CHARMONIUM
- DETECTOR PERFORMANCE
- CHARMLESS b-HADRON DECAYS
- QCD, ELECTROWEAK AND EXOTICA
- RARE DECAYS
- CHARM PHYSICS
- SEMILEPTONIC B DECAYS
- LUMINOSITY
- B DECAYS TO OPEN CHARM

List of papers (Total of 459 papers and 25745 citations)

TITLE	DOCUMENT NUMBER	JOURNAL
Model-independent observation of exotic contributions to $B^0 \to J/\psi K^+\pi^-$ decays	PAPER-2018-043 arXiv:1901.05745 [PDF]	PRL
Observation of the doubly Cabibbo-suppressed decay $\Xi_c^+ \to p\phi$	PAPER-2018-040	JHEP
Measurement of the branching fraction and CP asymmetry in $B^+ \to J/\psi\rho^+$ decays	PAPER-2018-036 arXiv:1812.07041 [PDF]	EPJC
Study of the $B^0 \to \rho(770)^0 K^*(892)^0$ decay with an amplitude analysis of $B^0 \to (\pi^+\pi^-)(K^+\pi^-)$ decays	PAPER-2018-042 arXiv:1812.07008 [PDF]	JHEP
Search for the rare decay $B^+ \to \mu^+\mu^-\mu^+\nu_\mu$	PAPER-2018-037 arXiv:1812.06004 [PDF]	EPJC
Search for CP violation through an amplitude analysis of $D^0 \to K^+K^-\pi^+\pi^-$ decays	PAPER-2018-041 arXiv:1811.08304 [PDF]	JHEP
First measurement of charm production in fixed-target configuration at the LHC	PAPER-2018-023 arXiv:1810.07907 [PDF]	PRL
Study of Υ production in pPb collisions at $\sqrt{s_{NN}} = 8.16$ TeV	PAPER-2018-035 arXiv:1810.07655 [PDF]	JHEP 11 (20
Measurement of the charm-mixing parameter y_{CP}	PAPER-2018-038 arXiv:1810.06874	Phys. Rev. L 011802

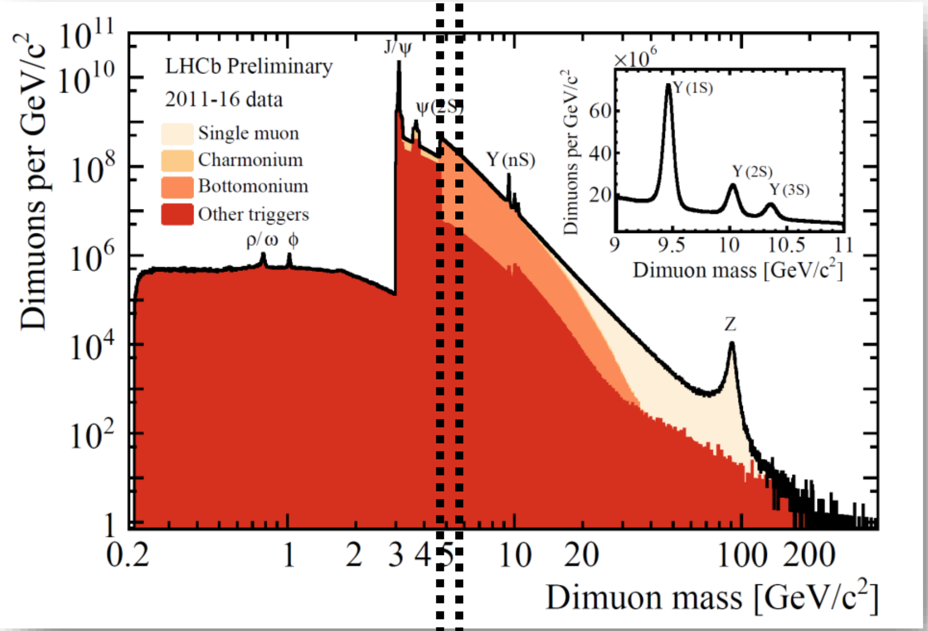
Setting the scene – challenges

Wildly different Challenges !

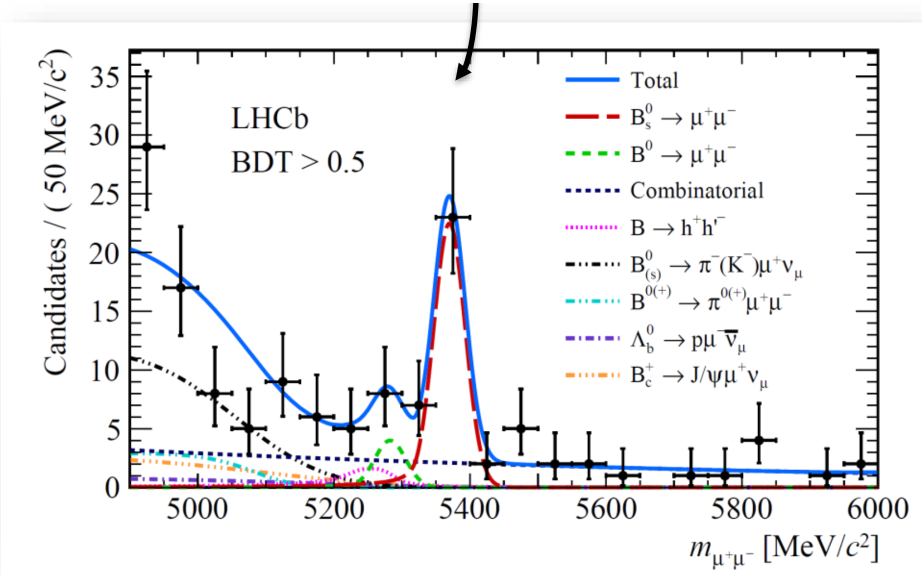


700M events !

- ❑ Example of $\mu^+\mu^-$ data sample in 2011-16
  - Spans 2 orders of magnitude in mass and several in yields !
- ❑ Amazing that mass peaks are seen at all scales, out of the trigger !

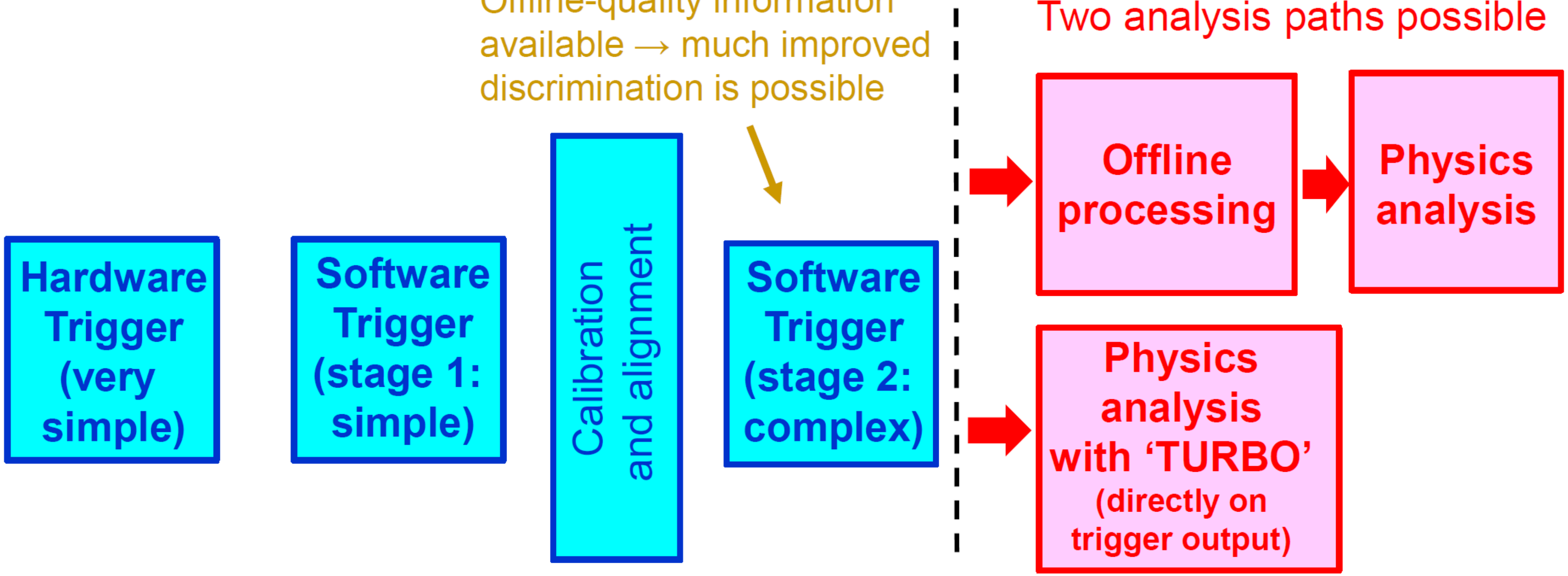


$$\mathcal{B}(B_s^0 \to \mu^+\mu^-) = (3.0 \pm 0.6\,^{+0.3}_{-0.2}) \times 10^{-9}$$



Note : structure given the numerous "trigger lines" with different requirements

Setting the scene – 2 different data flows



Offline-quality information available → much improved discrimination is possible

Two analysis paths possible

Hardware Trigger (very simple)

Software Trigger (stage 1: simple)

Calibration and alignment

Software Trigger (stage 2: complex)

Offline processing

Physics analysis

Physics analysis with 'TURBO' (directly on trigger output)

Online Offline

This brings many advantages !

Much quicker and more lightweight offline procedure required; TURBO route allows instant analysis !

Intermezzo – LHCb Analysis Survey in Autumn 2018

❑ Last Autumn I prepared and ran a (the first) survey around “all” aspects of data analysis in LHCb

❑ I got great participation, with 153 responses !

- We have ~840 LHCb authors, so ~18% author participation

❑ Topics were:

- LHCb data & analysis flows and models
- LHCb analysis software stack
- Analysis software ecosystem
- Analysis preservation
- C++, Python, what else?
- Early Career, Gender and Diversity (ECGD) matters

❑ Around 50 questions overall

❑ Summary of all survey words, questions and answers:

❑ *I will disclose a few bits of non-LHCb-specific information …*

Analysis steps and evolution:

- ***Which kind of main tasks and operations do you perform on data?***
- ***How does this change from day zero of preliminary analysis studies to last day before publication?***
- ***How do you deal with systematics?***

Analysis steps & evolution

Which kind of main tasks and operations do you perform on data?

- ❑ **The workflow is in most cases the same, broadly speaking:**
  data collection (detector) ⇒ trigger ⇒ selection ⇒ data analysis
- ❑ **Selection: “stripping” (=skimming&slimming), so ntuple production, and further skimming/slimming**
  - May require iterations upon discovery of some missing info in ntuples (requires re-production, on the Grid)
- ❑ **Analysis involves e.g. data-MC corrections, fine-tuned selections, data and calibration data fits, extraction of signal yields/angular analysis/limits setting**
- ❑ **Fit to extract physical parameters: mass fit, amplitude analyses, time-dependent fit, etc.**

- ❑ **Usage of simulation comes at various places - standard**
- ❑ **Corrections, re-weighting of distributions to match data**
  - Tend to become more and more sophisticated and common/necessary

- ❑ **Case of charm studies: at the forefront with massive yields for some favoured channels, reaching billions of events**
  - Yields will increase by a factor 10 in the upgrade … this means 2021++ ;-
- ❑ **Alternative/novel ideas are needed (beyond Turbo, see slide 4)**

Analysis steps & evolution

How does this change from day zero of preliminary analysis studies to last day before publication?

- ❑ To be honest, analysis flows are reasonably well set from the start (the big picture)
- ❑Many analyses already done in Run I ⇒ strategy is usually more or less clear and thus it doesn't change *drastically* from day 1
- ❑ This being said, analysis improvements/changes happen almost always

- ❑ Slightly different story for surprises – think the observation of pentaquarks
- ❑ Such analyses take a different course or focus

Analysis steps & evolution

How do you deal with systematics?

❑ The systematics usually come at the end, when the core of the analysis is developed
- Also fairly true for blind analyses

❑ No unique recipe for systematics studies – it largely varies on a case-by-case basis

❑ Typically:
- Consider all possible assumptions during analyses procedure and apply systematics by varying conditions
- Compare results, take differences as systematics

❑ Some complex analyses, such as amplitude analyses, do often require significant compute time to perform parameter scans and alike. Large clusters or GPUs employed by some LHCb groups

❑ Special cases such as heavy-ion studies: data samples for PID and tracking calibrations can be too small

Analysis steps & evolution – ~~stripping~~ skimming&slimming

❑ Stripping = run, in a centralised way, selections on reconstructed events to alleviate the size of datasets and to improve data handling by users

❑ Can be ran synchronously during the data-taking run (after data is reconstructed on-the-fly) or a full dataset is processed after the runs are over

❑ Takes around 3-4 months to complete a 'stripping' campaign on a complete dataset (code preparation, validations and productions)

❑ Different types of 'stripped' data (DSTs, μDSTs):
- Save only reconstructed candidates passing a certain 'stripping' selection,
- Save other parts of the event as well for offline re-processing
- Save also raw information from certain sub-detectors for special purposes (i.e. calibrations)

❑ Reconstructed information goes to tape, while 'stripped' data goes to disk - easily accessible by analysts:
- Around 2000 different selections ("stripping lines") are processed per data-taking year worth dataset
- Between 1% and 3% are requiring other parts of the event, while only a 0.05% ask for raw information

❑ Some of the lines are now being replaced with Turbo selections

❑ This framework will still be needed for the Upgrade, optimised even more to meet a higher reduction rate
- In which “format” is still to be seen …

Sketch of analysis workflow

- ***An overall sketch of the complete analysis flow, even via a cartoon.***
- ***On what datasets does it start (group reduced ntuples, central datasets)?***
- ***Where and what does run on them (experiment framework, own program, on a university cluster, on the Grid)?***
- ***What is the output (histograms, reduced ntuples) and how is it processed (ROOT macro/program, PyROOT script, own analysis framework)?***

Sketch of analysis workflow

- ❑ **Typical workflow described in a previous slide. Some analyses have special needs**

- ❑ **E.g. for charm studies, very large datasets imply that the laptop is reserved for the analysis of the data at the last stages of the analysis, and most of the skimming is performed in batch mode**
- ❑ **Usage of WG productions to make tuples is becoming standard**

- ❑ **The number of "analysis frameworks" is huge …**
- ❑ **"A common framework for amplitude analysis (like RooFit) would be appreciated, considering that at the time there are many efforts in various directions, sometimes reinventing the wheel."**

Sketch of analysis workflow

Where and what does run on them (experiment framework, own program, on a university cluster, on the Grid)?

- ❑ **A bit of all**

- ❑ **Outcome from survey on “what kind of resources do you use for the majority of your analysis work?”**

- ❑ **lxplus still very popular !**
- ❑ **Own and institute resources mandatory**
- ❑ **Usage of GPUs not at all insignificant**

Package	Usage
lxplus	74%
Laptop or desktop	63%
At institute or lab	52%
GPUs	7%
SWAN	3%
Openstack VMs	1%

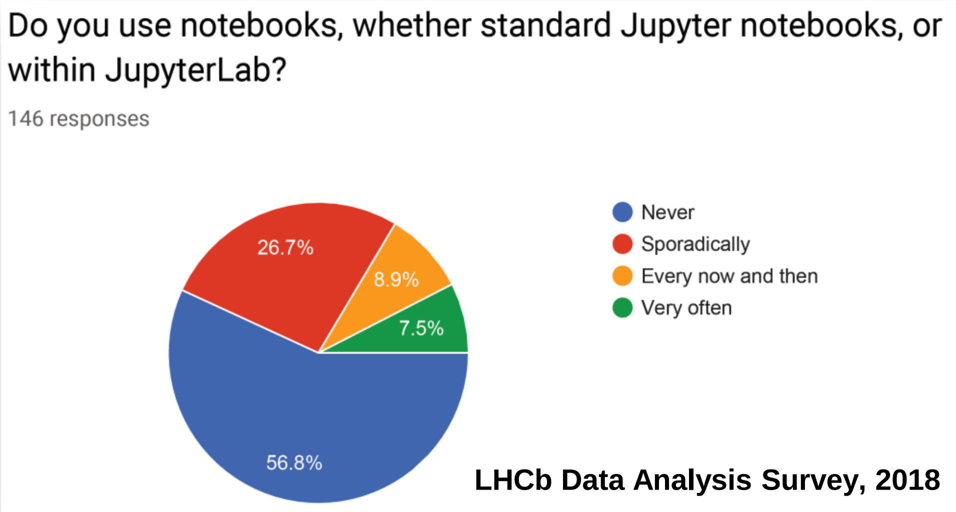
Analysis Interface

- ***The method through which you actually execute the analysis, i.e. the analysis interface of your choice. Multiple options are of course possible and example interfaces are scripts, compiled programs dynamically compiled, jupyter notebooks, graphical user interfaces…***
- ***The interface can of course depend on the step of the analysis being considered.***

Analysis interface

The method through which you actually execute the analysis, i.e. the analysis interface of your choice. Multiple options are of course possible and example interfaces are scripts, compiled programs dynamically compiled, jupyter notebooks, graphical user interfaces…

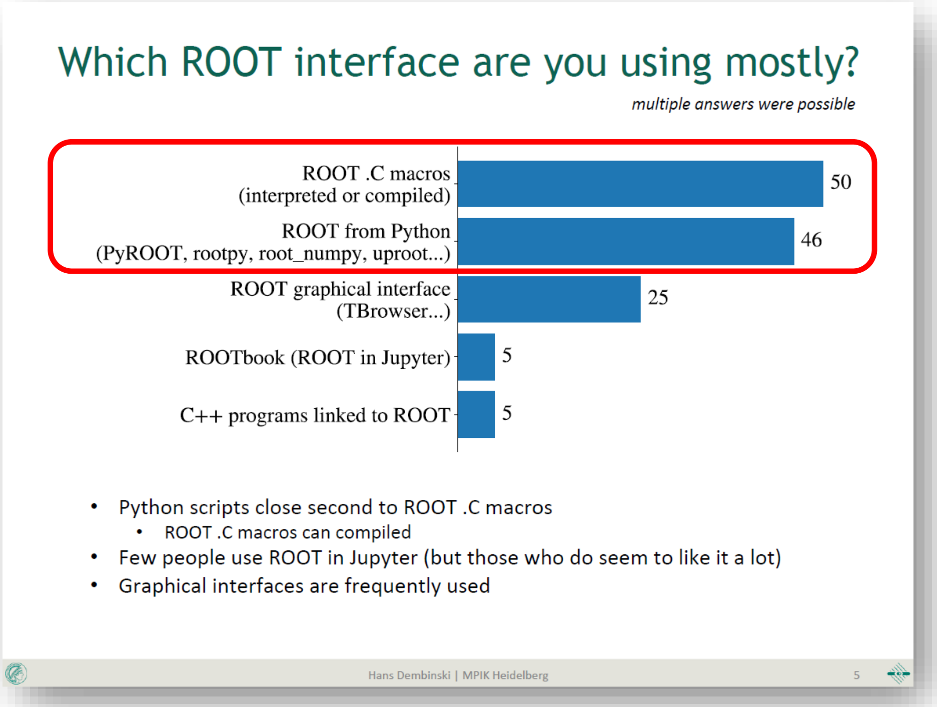
- ❑ **All "interfaces" are very much used in LHCb, except notebooks that are starting to be exploited**
  - **- Scripts probably most common**
- ❑ **Depends largely on what each user finds convenient (except for what is imposed, e.g. interface to grid submission)**
- ❑ **ROOT, PyROOT, etc.**



Do you use notebooks, whether standard Jupyter notebooks, or within JupyterLab?

146 responses

LHCb Data Analysis Survey, 2018



Which ROOT interface are you using mostly?

multiple answers were possible

- Python scripts close second to ROOT .C macros
  - ROOT .C macros can compiled
- Few people use ROOT in Jupyter (but those who do seem to like it a lot)
- Graphical interfaces are frequently used

Taken from Hans Dembinski, *User Feedback from LHCb*, ROOT Users' workshop, Sarajevo, Sep. 2018

The interface can of course depend on the step of the analysis being considered

- ❑ **This is probably true for us all ;-)**

Scaling

- ***The way in which you achieve a competitive turn-around time, i.e. how you make the analysis procedure scale well with the input data size increase. For example, do you exploit mass processing resources such as batch clusters or the Grid and, if yes, do you see any shortcoming with this approach?***
- ***Do you feel that more interactive approaches could boost your productivity? The answer can of course depend on the step of the analysis being considered.***

Scaling you said? I say, see it big !

❑ $D^0 \rightarrow K_S^0\,\pi^+\pi^-$ in Run 1-2 has ~50M signal candidates after all cuts. Expect in Runs 3-4 ~550M signal candidates

❑ Charm signal yields in Run-3 and beyond will be huge:

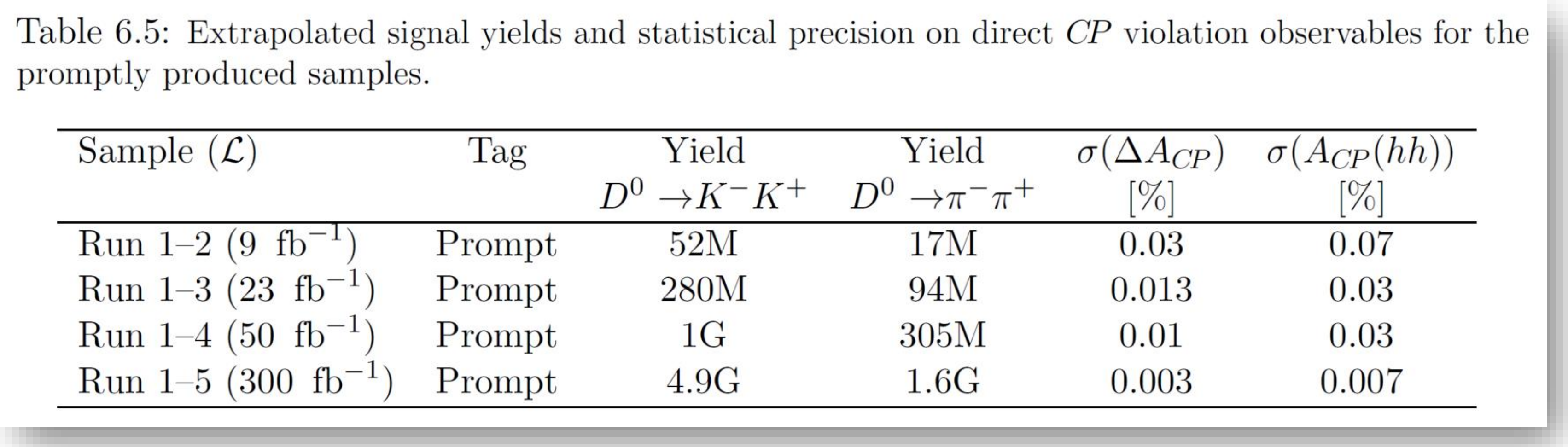


Table 6.5: Extrapolated signal yields and statistical precision on direct *CP* violation observables for the promptly produced samples.

Sample ($\mathcal{L}$)	Tag	Yield $D^0 \to K^-K^+$	Yield $D^0 \to \pi^-\pi^+$	$\sigma(\Delta A_{CP})$ [%]	$\sigma(A_{CP}(hh))$ [%]
Run 1–2 (9 fb^{-1})	Prompt	52M	17M	0.03	0.07
Run 1–3 (23 fb^{-1})	Prompt	280M	94M	0.013	0.03
Run 1–4 (50 fb^{-1})	Prompt	1G	305M	0.01	0.03
Run 1–5 (300 fb^{-1})	Prompt	4.9G	1.6G	0.003	0.007

Physics case for an LHCb Upgrade II, https://arxiv.org/abs/1808.08865

❑ Charm WG productions: we have produced approximately 15 TB of analysis ntuples on 2016 data!

Scaling

The way in which you achieve a competitive turn-around time, i.e. how you make the analysis procedure scale well with the input data size increase. For example, do you exploit mass processing resources such as batch clusters or the Grid and, if yes, do you see any shortcoming with this approach?

❑ **For sure: Grid basically always used by each analysis, e.g for repeated fitting, tuple processing, pseudo-experiment generation/fitting (see previous slides)**
- Batch clusters also rather popular

❑ **Waiting time will become significant with the data samples expected in Run 3 …**

❑ **Certain analyses use GPUs and there are no Grid resources available. Local resources exploited / set up**

❑ **BTW, multi-core machines are a must for many**

❑ **"In general, having a few fast cores interactively with fast access to a moderate amount of storage is much more important than a large batch/grid system.**
During analysis development, disk I/O from ROOT trees is often more limiting than CPU."

Do you feel that more interactive approaches could boost your productivity? The answer can of course depend on the step of the analysis being considered.

❑ **Yes, more interactive approaches will boost the analyses (Notebooks being used more these days)**

Reusability

- ***Specific software developed explicitly for some analysis or group of analyses.***
- ***If any analysis specific software has been developed in your case, do you think that the effort which was spent in developing such "software setup" was sizeable?***
- ***If yes, do you think there could be opportunities to share pieces of it with others, or at least knowledge about it, which could make the creation of such setups less onerous in the future?***

Reusability

Specific software developed explicitly for some analysis or group of analyses.

❑ **Typical and by-far dominant case: fitting code**

- Many groups have tailored fitting code and/or programs. Some depend on ROOT, others don't

- Some analyses (amplitude analyses) exploit frameworks for GPUs

❑ **Also calibration and reweighting tools start often with analysts before being made mode widely usable**

- We do have standard calibration tools across LHCb for tracking and PID calibrations, for example

If any analysis specific software has been developed in your case, do you think that the effort which was spent in developing such "software setup" was sizeable?

❑ **It's often the case, yes!**

❑ **Fortunately, there's ever more interest in using / looking at other's packages …**

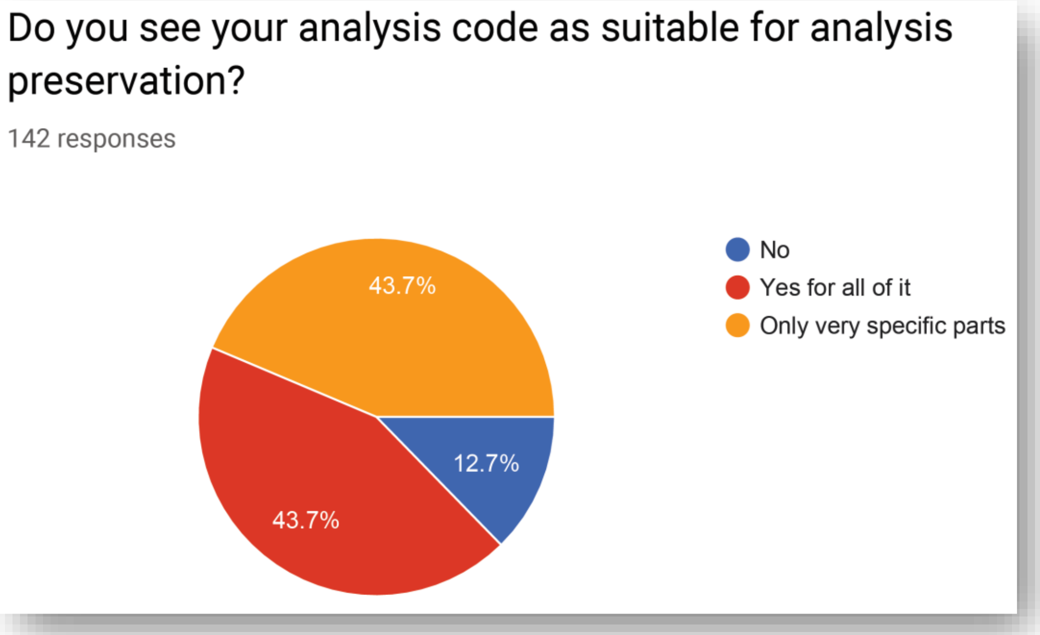
Reusability

If yes, do you think there could be opportunities to share pieces of it with others, or at least knowledge about it, which could make the creation of such setups less onerous in the future?

- ❑ **For sure!**
- ❑ **To be fair, there are already many analyses & groups in LHCb sharing code between each other**
  - Also, "update analyses" with newer data samples largely re-use code developed in previous analyses
- ❑ **For some WGs tools are even being developed with theorists, as necessary to address very specific issues (e.g. tool for semileptonic matrix element reweighing, currently being developed by theorists)**
- ❑ **GitLab is the main sharing hub. Also GitHub for less LHCb-specific packages (largely Python)**

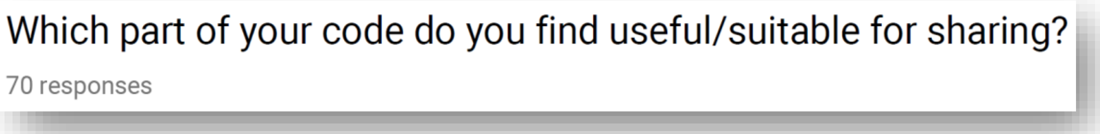
- ❑ **BTW, making tools shareable and usable by others comes with its own challenges:**
  - Public code demands higher standards than private code
  - Time to develop, which may be significant

Reusability – some results from the LHCb Analysis Survey



Do you see your analysis code as suitable for analysis preservation?

142 responses



Which part of your code do you find useful/suitable for sharing?

70 responses

Just about anything was cited. Recurrent replies:

❑ All of it

❑ Core code – ntuple making code, scripts, selections, fitting code

❑ Analysis workflow scripts

❑ Plotting scripts

❑ Miscellaneous utilities

❑ Models

❑ Systematics seen as a tricky thing to share

Missing functionality

- ***Among the operations you needed to carry out, some might have been more difficult than others with the current set of software tools.***
- ***What set of tools did you feel could need improvement? Can you also describe how?***

Missing functionality

Among the operations you needed to carry out, some might have been more difficult than others with the current set of software tools

- ❑ **Not trivial to extract such feeling given that some "analysis operations" are *inherently* far more complicated and intricate than others**

- ❑ **We see a tendency for certain WGs to collaborate on the development of tools for the benefit of all members**
  - Often tools for very specific needs of analyses typical to that WG

Missing functionality

What set of tools did you feel could need improvement? Can you also describe how?

❑ **It’s not just about the tools but *also* about the tools documentation!**

❑ **In general, HEP tools tend not to be well documented, with useful “getting started” and tutorials**

❑ **Benefit from better interfaces to popular machine learning libraries (TensorFlow/Keras/scikit-learn/...) that can be dropped into the experiment software.**
“Some generic C++ interface would probably be useful for all of the LHC experiments in the future.”

❑ **Example usages:**
- Advanced ML techniques in production in the trigger such as a flatness boosting technique to reduce phase space and lifetime biases

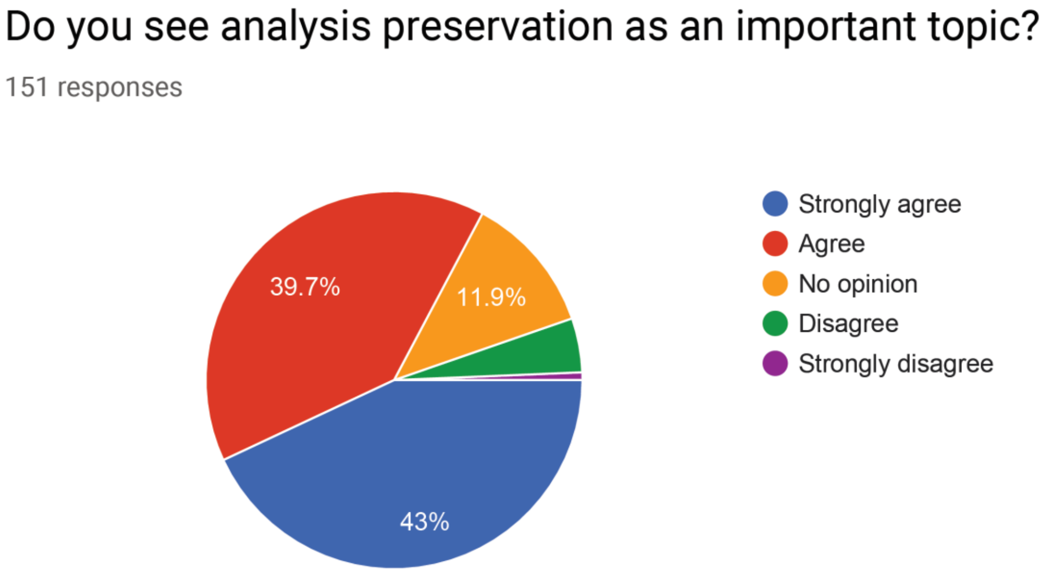
❑ **Existing frameworks for binned fits (RooHistFactory, RooFit) aren't fully suited to some WG needs**
- “We are in the process of deciding on and implementing a common solution for the group …
We are really thankful for the stability and prompt bug fixing of ROOT, and we wish this will be kept going.
We would profit from improvements in the fit frameworks: having a framework widely used/maintained and continuously developed/improved to cover several different possible use cases.”

Preservation and sharing

- *The issue of long term preservation of analyses as well as its very short term incarnation, the sharing, is a concern.*
- *What steps did you take to make sure your analysis procedure was shareable among your colleagues?*
- *And to ensure long term reproducibility?*

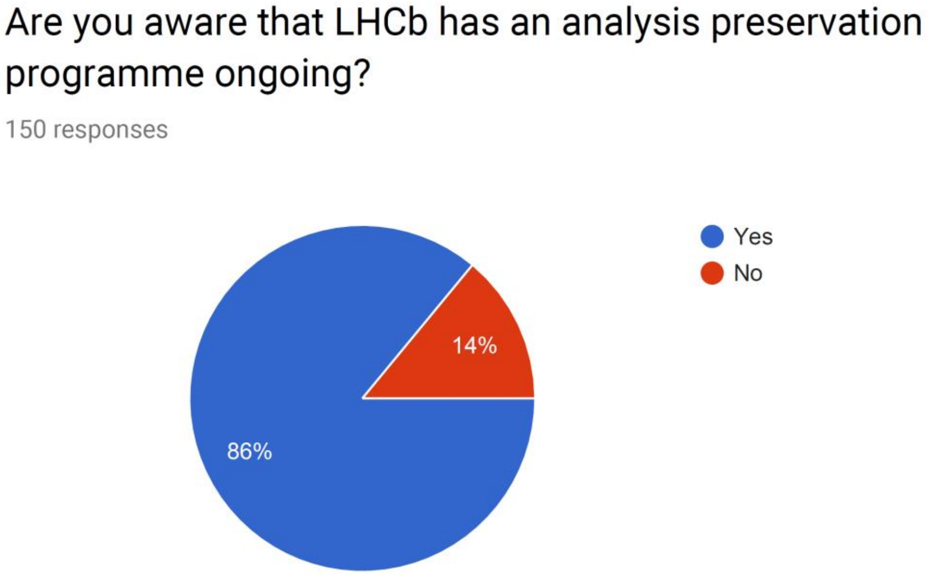
Preservation & sharing – some results from the LHCb Analysis Survey

- **Close to 85% of LHCb says AP is important**
  - **The same people that are aware of an LHCb AP programme?**



Do you see analysis preservation as an important topic?

151 responses



Are you aware that LHCb has an analysis preservation programme ongoing?

150 responses

Preservation & sharing

The issue of long term preservation of analyses as well as its very short term incarnation, the sharing, is a concern.

❑ Agreed 100% & LHCb is taking the matter seriously

❑ Twiki https://twiki.cern.ch/twiki/bin/view/LHCb/AnalysisPreservationReproducibility has been set-up to provide relevant information to analysts

❑ 4 domains identified on the route towards analysis reproducibility:
- Analysis code repositories
- Ntuple storage
- Analysis automation
- Runtime environment preservation

Preservation & sharing

What steps did you take to make sure your analysis procedure was shareable among your colleagues?

- ❑ **In short, LHCb now requires that all analyses post code on GitLab & store final-version ntuples on EOS**
  - Details such as what exactly to store, yet to be fine-tuned as experience is gathered
- ❑ **This includes basic documentation on what is stored in these long-term supported repositories, and documentation of the software experiment version and option files used**
- ❑ **All WGs and analysts required to follow LHCb guidelines/rules**

- ❑ **To be fair, many analysts and groups were already keeping and/or sharing analysis code on GitLab**
  - Some went as far as having Continuous Integration set up!

Preservation & sharing

And to ensure long term reproducibility?

❑ There is still some way to go, to be honest !

❑ Making AP a reality requires community effort and solutions

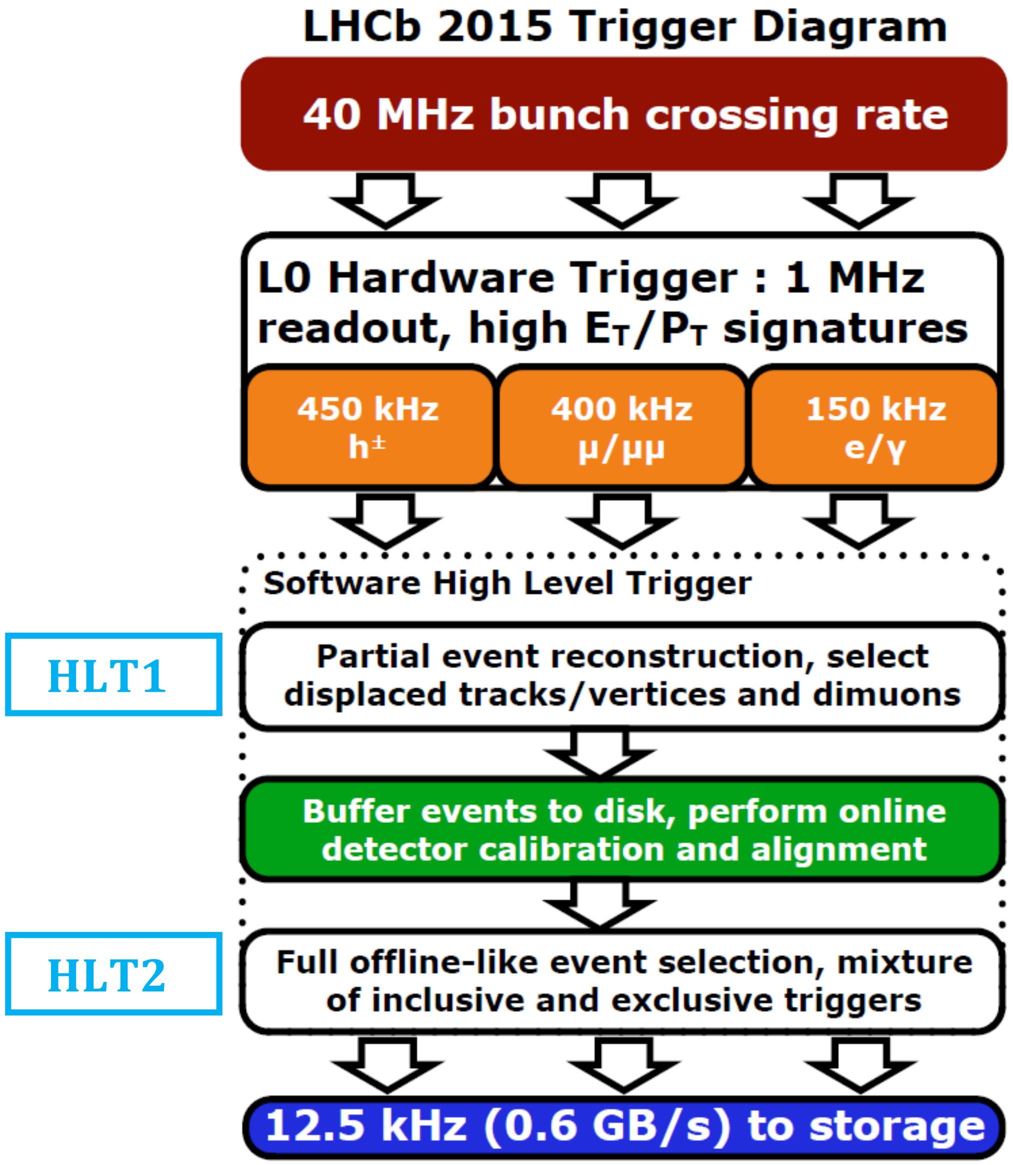
❑ Present guidelines and rules in LHCb will expand ⇒ more rules and less guidelines

❑ The Graal: “It would be really nice to have some framework so that people just need a few clicks to reproduce the published results. At present it is just too time-consuming to check that we can actually reproduce the preserved analysis.”
- Understood that frameworks & tools are under development, e.g. CERN’s Analysis Preservation Portal (CAP)

❑ Kind of worries:
- AP non-trivial for complicated analyses (e.g. very time-consuming fits, specific environments such as GPUs)

Back-up

LHCb trigger – run II compared to the upgrade model for 2021++



LHCb 2015 Trigger Diagram

40 MHz bunch crossing rate

L0 Hardware Trigger : 1 MHz readout, high E_T/P_T signatures

450 kHz $h^\pm$

400 kHz μ/μμ

150 kHz e/γ

Software High Level Trigger

HLT1

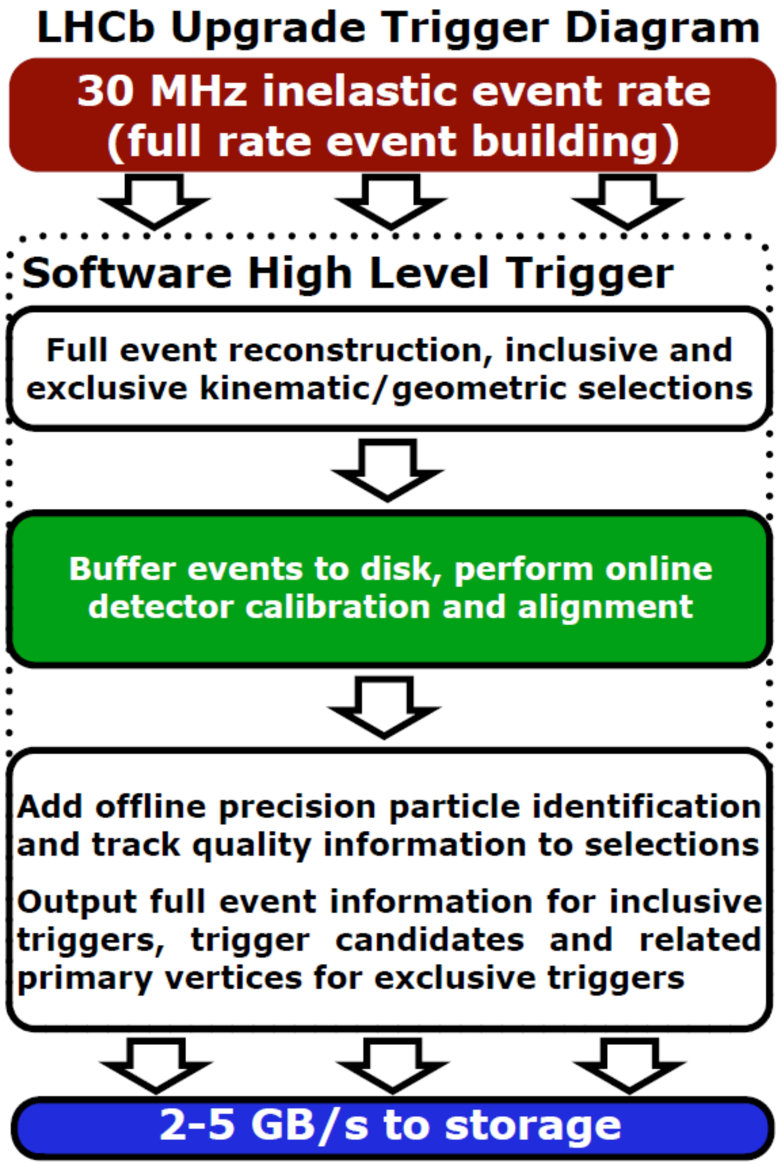
Partial event reconstruction, select displaced tracks/vertices and dimuons

Buffer events to disk, perform online detector calibration and alignment

HLT2

Full offline-like event selection, mixture of inclusive and exclusive triggers

12.5 kHz (0.6 GB/s) to storage



LHCb Upgrade Trigger Diagram

30 MHz inelastic event rate (full rate event building)

Software High Level Trigger

Full event reconstruction, inclusive and exclusive kinematic/geometric selections

Buffer events to disk, perform online detector calibration and alignment

Add offline precision particle identification and track quality information to selections

Output full event information for inclusive triggers, trigger candidates and related primary vertices for exclusive triggers

2-5 GB/s to storage