The particle identification (PID) in LHCb

Marianna Fontana[1],
with the precious help of Lucio Anderlini[2] and Donal Hill[3]

[1]INFN (Cagliari) and CERN, [2] INFN (Firenze), [3] Oxford University

UK students meeting - 8 February 2017

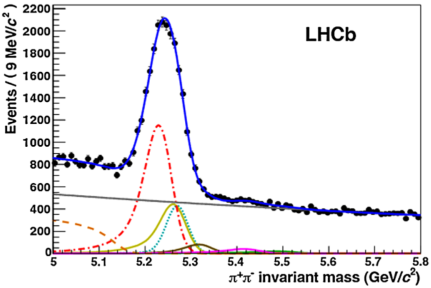
Overview

- Importance of PID
- PID subdetectors in LHCb
- The Global PID
- The new strategy for Run 2
    - WGP for PID samples
    - PIDCalib

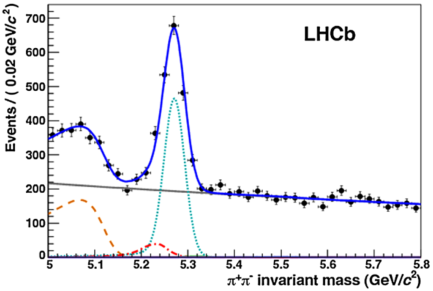
Neutral PID and relevant tools not included in this talk

PID matters

- One of the major requirements in a flavour physics experiment is the **reduction of combinatorial background** and the possibility to **distinguish final states** of otherwise identical topology
- Moreover particle identification plays a crucial role in the determination of the **flavour tagging** of the neutral *B* mesons

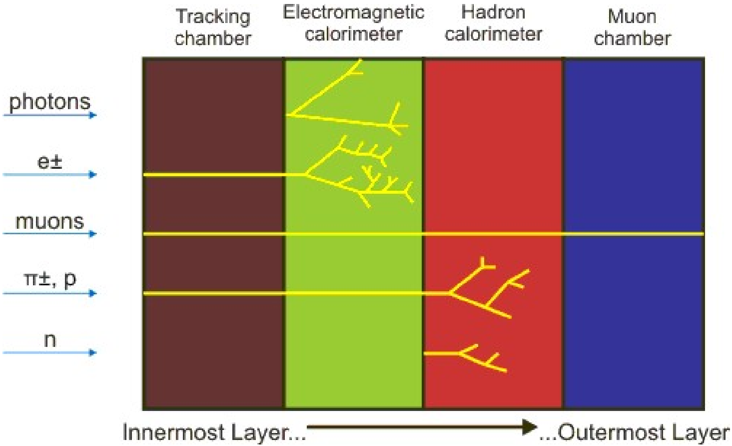


Before any PID requirements

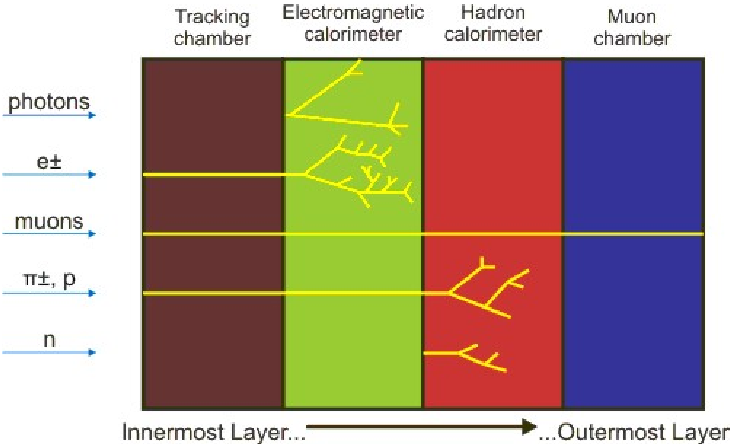


After PID requirements

Material interaction

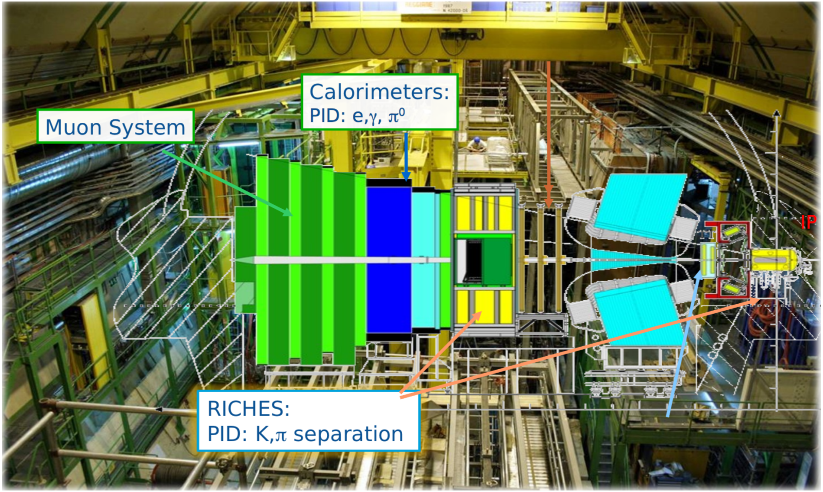


Material interaction



This is not enough in LHCb to distinguish different hadrons

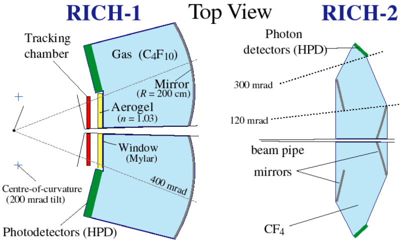
How we identify particles



The PID detectors: RICH

- Need to identify heavy flavour decays from huge hadronic background
- Good $\pi/K/p$ separation over a wide momentum range

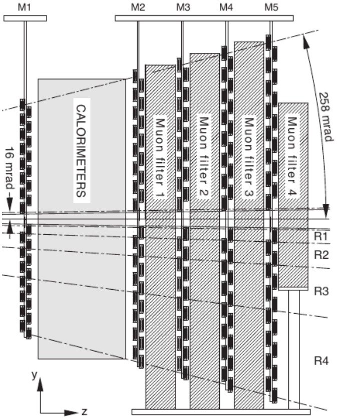
- Use 2 separate sub-detectors with different radiators gases:
  - RICH1 covering low p (2-60 GeV/c) region, using C_4F_{10} radiator
  - During LS1 the aerogel has been removed from RICH1
  - RICH2 covers higher momenta (15-100 GeV/c) with CF_4 radiator



- The light rings are produced on an array of HPD located outside the LHCb acceptance (usage of spherical and flat mirrors)
- Combine photon rings and track momentum information
- Log likelihood recomputed for the mass hypothesis of all charged particles

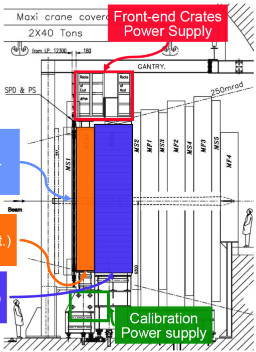
The PID detectors: MUON

- 5 tracking stations interspersed with hadron absorbers ($\sim 23\lambda$)
  - M1 before the calorimeter
- Technology
  - MWPC
  - 3-GEM in M1 (inner region)
- Identification based on
  - Track extrapolation to the μ-system
  - Look for hits in the μ stations around the extrapolated track
  - Calculate probability from hit distribution in μ-stations



The PID detectors: CALO

- Calorimeter system identifies electrons/photons/π^0 and hadrons
- It combines information from:
  - SPD
  - Preshower
  - ECAL
  - HCAL



- Photon PID based on 2D PDF $\rightarrow \Delta LL$ method
  - Energy: total cluster energy in the ECAL and reconstructed energy deposit in the PS
  - Direction: from the interaction point and the energy-weighted position of the photon candidate

Global PID - DLL

- The PID information obtained from the MUON, RICH, and CALO systems is combined to provide a single set of more powerful variables.
- **DLL**: the likelihood information produced by each sub-system is simply multiplied, to form a set of combined likelihoods

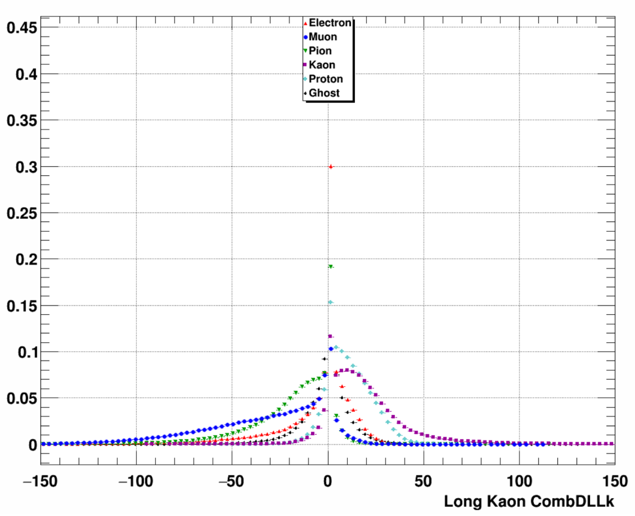
$$\mathcal{L} = \mathcal{L}_{RICH} \cdot \mathcal{L}_{CALO} \cdot \mathcal{L}_{MUON}$$

- These variables give a measure of how likely the mass hypothesis under consideration is, for any given track, relative to the pion hypothesis

$$\Delta ln\mathcal{L}_{K\pi} = ln\mathcal{L}(K) - ln\mathcal{L}(\pi)$$

- You can get these information in your ntuple using `TupleToolPID`

Example of DLL distribution

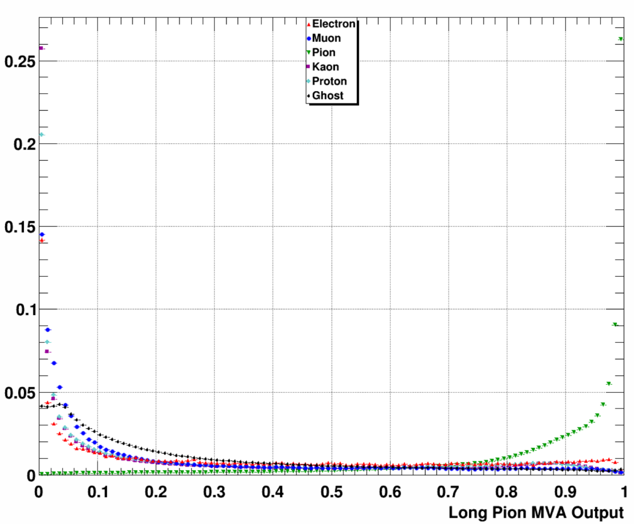


Courtesy of C. Jones

Global PID - ProbNN

- **ProbNN**: these variables are built using multivariate techniques
- Training on **MC events** using:
  - PID information from each sub-system
  - Tracking information
- All tracks in the events considered, including ghosts
- Separate electron, muon, pion, kaon, proton and ghost networks
- Separate networks trained for Long, Downstream and Upstream tracks
- The output is a probability distribution [0,1]: ProbNN(K, π, p, μ, e)
- Different tunings depending on the training and data sample:
  - MC12TuneV1: first developed, too old and not recommended
  - MC12TuneV2: default for Run 1
  - MC12TuneV3: gives better performances for electrons
  - MC15TuneV1: default for Run 2
  - MC12TuneV4: same training variables as MC15TuneV1
- The ntuples can be filled with this information using `TupleToolANNPID`

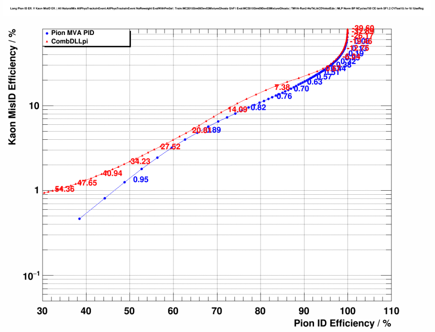
Example of ProbNN distribution



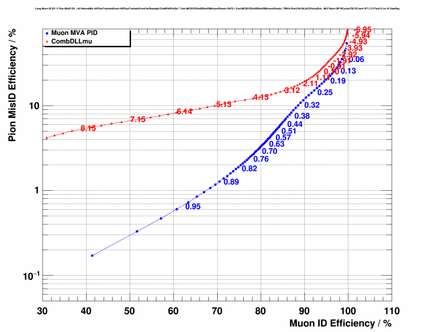
Courtesy of C. Jones

DLL vs ProbNN

- The choice of the variable depends strongly on the particular analysis
- The performance changes as a function of the kinematics
- As a very general statement ProbNN improves a lot in case of muons
- The discriminating power could be enhanced using a combination of ProbNNs or ProbNN + DLL



Courtesy of C. Jones



Courtesy of C. Jones

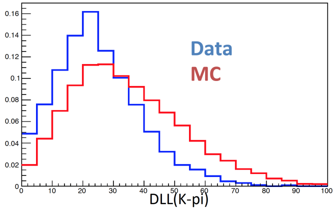
How we use PID in our analysis

- The PID information is used to select the signal particles in your decay and reject particle types of backgrounds
- What we measure after the selection (N_{raw}) needs to be corrected for an efficiency loss

$$N_{true} = \varepsilon \cdot N_{raw}$$

- Often we determine ε from MC

- The Data/MC agreement for PID is poor
- Suboptimal selection using MC
- The ε_{PID} from MC will be wrong
- The best way is to use calibration samples from data



Courtesy of S. Malde

Data driven calibration

- PID calibration samples contains decay candidates with a topology which allows unambiguous identification of one daughter without relying on its PID-related variables

Species	Soft	Hard
$e^{\pm}$	—	$J/\psi \to e^+e^-$
$\mu^{\pm}$	$D_s^+ \to \mu^+\mu^-\pi^+$	$J/\psi \to \mu^+\mu^-$
$\pi^{\pm}$	$K_S^0 \to \pi^+\pi^-$	$D^* \to D^0\pi^+$, $D^0 \to K^-\pi^+$
$K^{\pm}$	$D_s^+ \to K^+K^-\pi^+$	$D^* \to D^0\pi^+$, $D^0 \to K^-\pi^+$
$p^{\pm}$	$\Lambda^0 \to p\pi^-$	$\Lambda^0 \to p\pi^-$, $\Lambda_c^+ \to pK^-\pi^+$

- low-multiplicity modes with large BRs
- tag-and-probe method (e.g. $J/\psi \to \mu\mu$)

How to select the samples: Run 1 vs Run 2

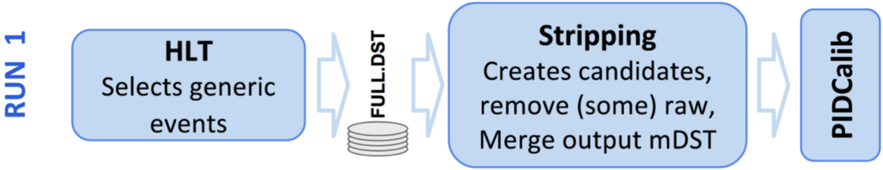
Run 1

- Selected with specific stripping lines
- Despite huge samples, some lack of statistics and/or kinematic coverage

Run 2

- Selected directly at HLT2 level in a new stream called TURCAL (TurboCalib)
- Some analysis are done at HLT2 level in the Turbo stream, while some others are done at stripping level
- Need to have PID samples for both type of analysis
- **Solution**: write the calibration samples to a dedicated stream, containing the online information together with the full raw event, which is then reconstructed and processed offline
- TurboCalib events are stored completely on disk and processed through Brunel

How to select the samples: Run 1 vs Run 2



In order to

- *Persist Turbo PID information as computed in the trigger*
- *Make samples available to test PID algorithms for the upgrade*
- *Select low-pT samples otherwise unavailable*

In Run 2 we select PID Calibration samples in HLT using dedicated lines.

How to produce the samples: Run 1 vs Run 2

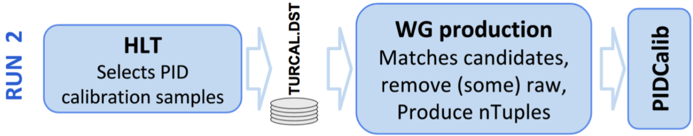
Run 1

- Done by Oxford, PIDCalib maintainer, via Ganga jobs

Run 2

Introduce a Stripping-like step to be run centrally (WG production): optimisation in the use of resources and big control on the dataset used and jobs running

1. Match online and offline candidates and store them
2. The residual background is subtracted with the sPlot technique
3. Compute the sWeight associated to each candidate
4. Produce nTuples and sWeighted MicroDSTs
5. Store nTuples and MicroDSTs in official LHCb Bookkeeping



The packages involved

- Quite intense validation during 2016

Project	Package	Maintenance	What manages
Moore	Hlt/Hlt2Lines	HLT	Selection strategies in Hlt2: TurboCalib
DaVinci	Phys/TeslaTools	HLT	Online-Offline matches, Tesla Histograms
DBASE	ProdConf	Core	Mixed output-file types: MicroDST & tuples
DBASE	AppConfig	-	Definitions of monitoring histograms
Erasmus	PIDCalib/PidCalibProduction	PID	TupleTools, Application sWeights, ...
-	PIDCalib/PidCalibProductionScripts	PID	Fit, HTML report, sTables
DBASE	WG/PIDCalib	PID	Histograms, calib. samples, nTuple branches
Urania	PIDCalib/PidPerfScripts	PIDCalib	User interface to PIDCalib

Matching online-offline candidates

- Tracks are reconstructed independently online and offline
- Online tracks are used to create decay candidates in the Turbo lines
- Online tracks, candidates for PID calib sample decay daughters are matched to offline tracks
- Matching is based on the number of shared LHCbIDs of the online and offline versions of the same track
- Both versions of all tracks involved in the decay are stored in MicroDST and nTuples
- Offline CombinedParticles are never created. Combinations only happen in the HLT and only basic objects are matched to offline versions

Compute the sWeights of each candidate

- The sPlot technique is used to subtract the background from the calibration samples
- The method is based on a **binned maximum likelihood fit** which is performed on histograms of the most discriminating variable: **mother particle mass distribution**
- Different histograms are defined for different decay modes
- 2D histograms are used for decays including an intermediate state such as:
  1. $D^{*+} \to D^0\pi^+$ with $D^0 \to K^-\pi^+$, using $m(D^{*+} - D^0)$ and $m(D^0)$
  2. $B^+ \to J/\psi K^+$ with $J/\psi \to \mu\mu$, using $m(B^+)$ and $m(J/\psi)$
- Using 1D histograms otherwise
- Note that the sWeight of the offline particles is never computed and the online version is not copied to the offline track candidates
- Today the online sWeight would be identical to the offline sWeight, but in future this may change, therefore the two worlds are separate
- The online sWeight is used to subtract the background from the offline candidates

Fit quality monitor

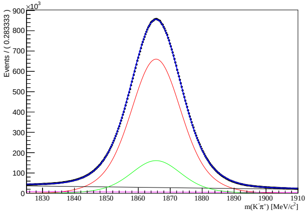
- Official website created as repository for the PID plots and fit results, transparent and available for all the users:

  https://lhcb-pid-wgp-plots.web.cern.ch/lhcb-pid-wgp-plots/Run2/

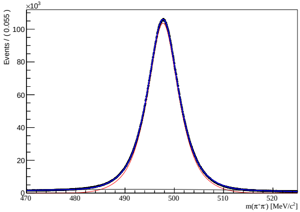
RSDStCalib_Pos	Fit model: Dstar_4comp	Fit status: SUCCESS	χ^2: ['4.66', '16.29']	Candidates: 16.672×10^6	More...
Lam0Calib_Neg	Fit model: Lambda0_2CB	Fit status: SUCCESS	χ^2: ['1.31']	Candidates: 4.173×10^6	More...
DsPhiCalib_Pos	Fit model: DsPhi	Fit status: SUCCESS	χ^2: ['3.07', '806.43']	Candidates: 4.868×10^6	More...
LbLcMuCalib	Fit model: Lc	Fit status: SUCCESS	χ^2: ['1.27']	Candidates: 0.464×10^6	More...
Lam0Calib_HPT_Neg	Fit model: Lambda0_2CB	Fit status: SUCCESS	χ^2: ['7.81']	Candidates: 6.517×10^6	More...
PhiCalib_Neg	Fit model: Phi_RBW	Fit status: SUCCESS	χ^2: ['0.93']	Candidates: 2.880×10^6	More...
Lam0Calib_Pos	Fit model: Lambda0_2CB	Fit status: SUCCESS	χ^2: ['8.23']	Candidates: 4.267×10^6	More...
Lam0Calib_HPT_Pos	Fit model: Lambda0_2CB	Fit status: SUCCESS	χ^2: ['4.60']	Candidates: 6.566×10^6	More...
RSDStCalib_Neg	Fit model: Dstar_4comp	Fit status: SUCCESS	χ^2: ['6.91', '16.44']	Candidates: 16.600×10^6	More...
DSt4bCalib_Pos	Fit model: Dstar4body	Fit status: SUCCESS	χ^2: ['2.77', '10.38']	Candidates: 4.703×10^6	More...
BJpsiEECalib_Pos	Fit model: BJpsiEE_delta	Fit status: SUCCESS	χ^2: ['1.16', '1.00']	Candidates: 0.013×10^6	More...
JpsiCalib_Pos	Fit model: Jpsi_1CB	Fit status: SUCCESS	χ^2: ['4.64']	Candidates: 4.237×10^6	More...
Lam0Calib_VHPT_Pos	Fit model: Lambda0_CB	Fit status: SUCCESS	χ^2: ['3.72']	Candidates: 2.268×10^6	More...
DSt4bCalib_Neg	Fit model: Dstar4body	Fit status: SUCCESS	χ^2: ['12.86', '22.42']	Candidates: 4.620×10^6	More...
PhiCalib_Pos	Fit model: Phi_RBW	Fit status: SUCCESS	χ^2: ['0.96']	Candidates: 2.711×10^6	More...
DsPhiCalib_Neg	Fit model: DsPhi	Fit status: SUCCESS	χ^2: ['3.06', '3.79']	Candidates: 4.892×10^6	More...
BJpsiEECalib_Neg	Fit model: BJpsiEE_delta	Fit status: SUCCESS	χ^2: ['1.43', '1.17']	Candidates: 0.012×10^6	More...
BJpsiCalib_Pos	Fit model: BJpsi	Fit status: SUCCESS	χ^2: ['1.99', '2.26']	Candidates: 0.060×10^6	More...
Lam0Calib_VHPT_Neg	Fit model: Lambda0_CB	Fit status: SUCCESS	χ^2: ['3.88']	Candidates: 2.108×10^6	More...
DsPhiCalib_notag	Fit model: DsPhiNoTag	Fit status: SUCCESS	χ^2: ['4.49', '11.65']	Candidates: 12.869×10^6	More...
BJpsiCalib_Neg	Fit model: BJpsi	Fit status: SUCCESS	χ^2: ['2.08', '2.10']	Candidates: 0.059×10^6	More...
JpsiCalib_Neg	Fit model: Jpsi_1CB	Fit status: SUCCESS	χ^2: ['2.16']	Candidates: 4.128×10^6	More...
KS0Calib	Fit model: Ks0	Fit status: SUCCESS	χ^2: ['1.24']	Candidates: 6.712×10^6	More...

Fit quality monitor: pions and kaons (2016)

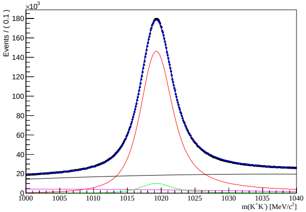
$D^{*+} \to D^0(K^-\pi^+)\pi^+$



$K_S^0 \to \pi^+\pi^-$

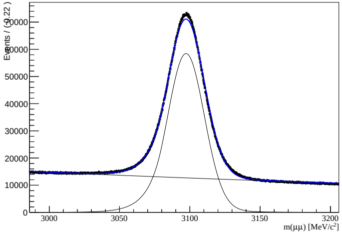


$D_s^+ \to \phi(K^+K^-)\pi^-$

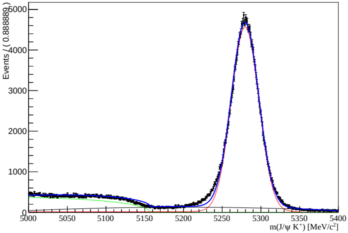


Fit quality monitor: muons (2016)

$J/\psi \to \mu^+\mu^-$



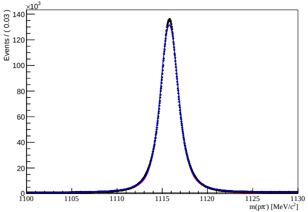
$B^+ \to J/\psi K^+$



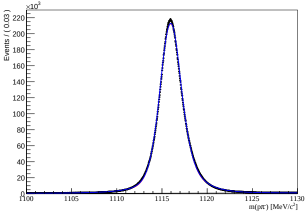
- Additional sample to access lower (p,pT) muons: $D_s^+ \to \phi(\mu^+\mu^-)\pi^- \to$ fit model in progress

Fit quality monitor: protons (2016)

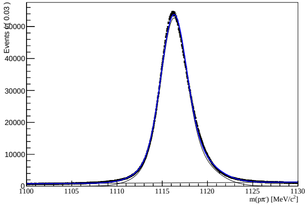
$\Lambda^0 \to p\pi^-$



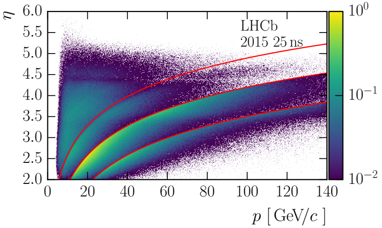
$\Lambda^0 \to p\pi^-$ $(pT > 3000$ MeV$)$



$\Lambda^0 \to p\pi^-$ $(pT > 6000$ MeV$)$

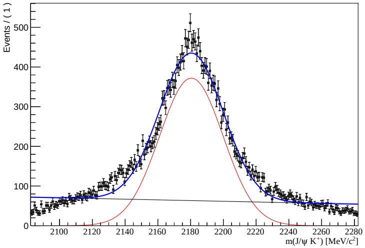


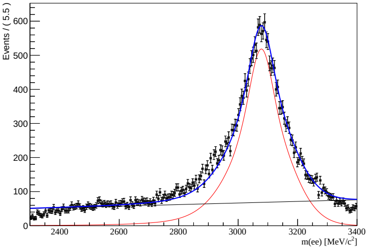
Improved coverage



The electron sample (2016)

- Inclusive $J/\psi \rightarrow e^+e^-$ studied in 2015: background too high
- $B^+ \rightarrow J/\psi(e^+e^-)K^+$ quite pure but low statistics

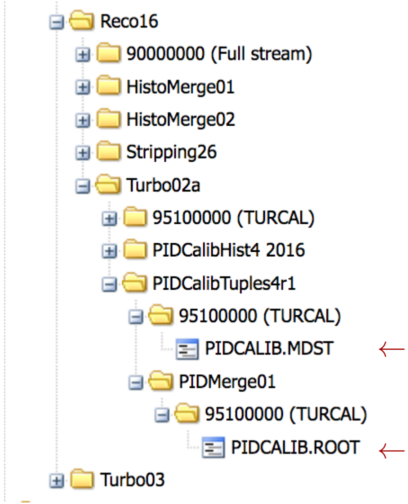




- We are currently revisiting the fit model (many thanks Thibaud!).

Where to find the samples

- The ntuples and microDSTs are available in the LHCb bkk



Where to find the samples

- Example path

`/LHCb/Collision15/Beam6500GeV-VeloClosed-MagDown/Real Data/Reco15a/Turbo02/PIDCalibTuples2r1/95100000/PIDCALIB.ROOT`

Year of data taking

LHC and Detector conditions

OFFLINE Reco version

ONLINE Tesla version

Tuples and MicroDST

ROOT: nTuples
MDST: MicroDST

- The version number has a specific meaning

`PIDCalibTuples2r1`

Version of the histograms/fits/sWeights

Version of MicroDST and nTuples

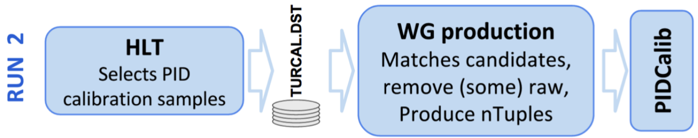
Possible reasons for version-change

Major
- ➢ Updated selection strategy
- ➢ Improved fit models
- ➢ New calibration modes

Minor
- ➢ New variables in the nTuples (*tunings*?)
- ➢ Modified `RelatedInfo` in μDSTs
- ➢ Changing RAW info available in μDSTs

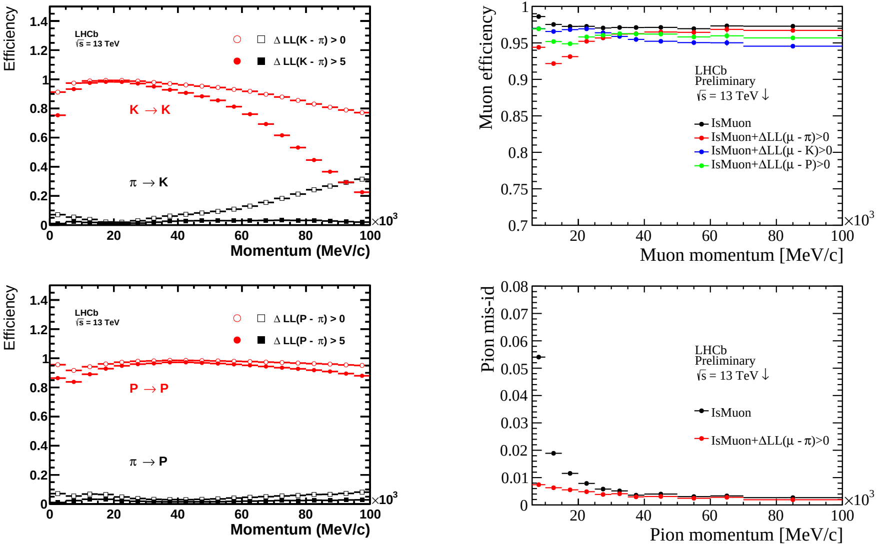
How we measure PID efficiencies: PIDCalib

- PIDCalib is the main interface between PID samples and analysts



- It constructs PID tables for each track type for given PID cut and then use signal MC to determine a event-by-event weight
- Code has been updated to read Run 2 WG production ntuples:
  - Read ntuples directly from bkk
  - Online and offline available for analysis using Turbo and Stripping
  - Access to all the available variables
- Twiki updated with instructions for using Run 2 samples
  `https://twiki.cern.ch/twiki/bin/view/LHCb/PIDCalibPackage`
- Code now lives in git
  `https://gitlab.cern.ch/lhcb/Urania/tree/master/PIDCalib`

Examples of performance histograms



Summary

- PID is an essential requirement at LHCb in order to achieve its physics goals
- Lots of improvements to PID strategies for Run 2: new way of producing the calibration samples, more samples with more information available
- PIDCalib infrastructure updated to cope with the new samples, and is continually being improved for users

- Prospects:
  - Charged and neutral PID live now under the same roof
  - Main plan for this year is to parallelise the production of the samples and to integrate the tools to determine neutral PID efficiency in PIDCalib


thank you!
pptbackgrounds.org

Backup

Documentation

Notes

- LHCb-INT-2015-016, LHCb-PUB-2016-020: Computing and selection strategy for PID calibration samples for LHCb Run-II
- LHCb-PUB-2016-005: Calibration samples for particle identification at LHCb in Run 2
- LHCb-INT-2016-028, LHCb-PUB-2016-021: The PIDCalib package
- LHCb-INT-2016-029: Working group production for calibration samples [in preparation]

Twiki page

- Global PID: `https://twiki.cern.ch/twiki/bin/view/LHCb/GlobalParticleID`