Università degli Studi di Ferrara

DOCTORAL COURSE IN

PHYSICS

CYCLE XXXVI

COORDINATOR
Prof. Luppi Eleonora

Lepton Flavour Universality and Analysis Frameworks

Scientific-Disciplinary Sector (SDS): FIS/01

Ph.D. Candidate:
Couturier Benjamin

Supervisor:
Dr Bozzi Concezio

Co-Supervisor:
Dr. Vecchi Stefania

Years 2020/2023

Contents

Introduction

The Large Hadron Collider (LHC) located at CERN, at the Swiss-French border near Geneva, allows physicists around the world to investigate the properties of matter at extremely high energies, found in no other facility in the world. It is a gigantic particle accelerator that took decades to design and build [1], with the collaboration of more than 20 countries. The particle detectors installed at the four collision points of the LHC beams are truly global efforts that started taking data in 2010 and are updated in order to continue well into the 2030s.

By 2012, experiments had already provided experimental evidence for the existence of the long sought-after Higgs Boson, a key element of the Standard Model (SM) of Particle Physics predicted in the 1960s by Brout, Englert and Higgs. Data recorded at the LHC already provided a host of results, with more than 3000 scientific papers published by the four largest experiments by 2023. The Standard Model has been very successful at describing the fundamental particles and their interactions, but many questions remain open: it features free parameters that have to be measured by experiments, it does not explain the observed asymmetry between matter and antimatter in the universe, nor explain a non-zero mass of the neutrinos. Cosmological evidence for dark matter and dark energy in the universe cannot be explained by the current theory either.

To investigate those problems, the direct search approach consists in trying to produce and detect new particles in high energy collisions as was done for the Higgs boson. Another possibility is to try and perform precise measurements of known phenomena and compare with the SM predictions, highlighting the shortcomings of the theory. This is the approach also chosen by this thesis which aims at investigating the assumption, dubbed "Lepton Flavour Universality" or LFU, that the electroweak force couples to all charged leptons, the electron the muon and the tau, in the same way. The Large Hadron Collider Beauty experiment (LHCb) was not initially designed to perform LFU measurement with τ leptons. However its precision in event reconstruction and the performance of its particle identification systems allowed physicists to devise methods to perform them.

Such endeavours require a colossal detector, 21 m long and weighing 5600 T, and a global collaboration to build and operate it: the LHCb collaboration gathers around 1400 scientists, engineers and technicians representing 86 different universities and laboratories from 18 countries [2]. Massive computing resources are also required to process and analyse the recorded data.

This data and all research outcomes are available to all participants, and should be usable by future physicists. This principle is even stated in the convention for the establishment of

CERN[3] in 1953:

> *"The Organisation shall have no concern with work for military requirements and the results of its experimental and theoretical work shall be published or otherwise made generally available."*

Preserving the research and associated data is therefore crucial to the organisation, as formulated in its Open data Policy[4] in 2020, which:

> *"aims to empower the LHC experiments to adopt a consistent approach towards the openness and preservation of experimental data. Making data available responsibly (applying FAIR standards). [...] The foreseen use cases of the Open Data include reinterpretation and reanalysis of physics results, education and outreach, data analysis for technical and algorithmic developments and physics research."*

Ensuring that scientific results be available to all is also a priority for many governments and funding agencies: Open Science is key in the European Union's strategy for science and innovation [5] and in the US, the White House's office for Science and Technology Policy declared 2023 the *Year of Open Science* [6] with actions across federal administrations to advance open and equitable research. At the core of Open Science, the FAIR guiding principles for scientific data management and stewardship [7] imply that all results be Findable, Accessible, Interoperable and Reusable and details the meaning of those statements. Applying them has many consequences on the way research is carried out.

This thesis investigates Lepton Flavour Universality by studying B_s^0 decays to τ leptons in the final states, while checking the adequacy of the tools available within LHCb software to ensure tracking of the data provenance and reproducibility of the analysis, which are key parts of the FAIR principles. To set the context, it starts by presenting CERN, the LHCb experiment and its data processing chain in Chapters 1 and 2. It then reviews the current landscape preservation tools in High Energy Physics in Chapter 3 before exposing improvements implemented in the context of this work in Chapter 4. Chapter 5 is an overview of the Lepton Flavour Universality concept and of the tests that have been carried through until now. The current state of the LFU measurement is detailed in Chapter 6. Finally, the use of preservation tools is reviewed in the context of this analysis in Chapter 7.

Chapter 1

The LHCb experiment at CERN

1.1 CERN

The CERN convention signed in June 1953 by 12 different states established the basis for the European Organisation for Nuclear Research (CERN), with the goal to advance knowledge in particle physics by creating a European research laboratory, as well as to promote international cooperation and help training new generations of physicists, engineers and technicians. Located at the border between France and Switzerland near Geneva, CERN now counts 23 member states and several associate members as shown by Figure 1.1.

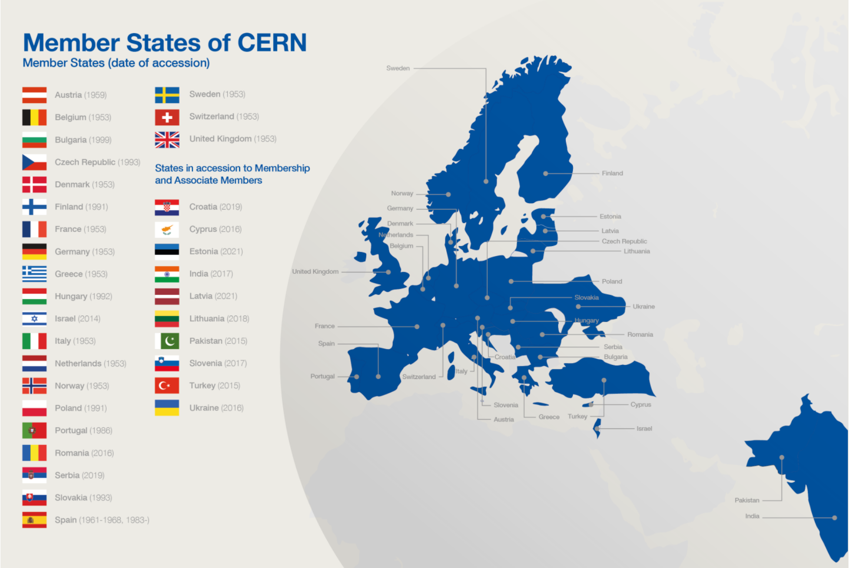


Figure 1.1: CERN members in 2021 [8]

CERN's first accelerator, the Synchrocyclotron started operating in 1957, followed by the Proton Synchrotron in 1959. Multiple accelerators were built over the years, providing the conditions for experiments to probe fundamental physics further and further leading to many discoveries such as weak neutral currents at the Gargamelle bubble chamber [9] in 1973, the discovery of W and Z bosons in the UA1 [10] and UA2 [11] experiments in 1983 and the Higgs boson at ATLAS and CMS [12, 13].

As a host laboratory, CERN builds and operates accelerators to provide the particle beams required for research: Figure 1.2 shows the accelerator complex as of 2022. The large Hadron Collider (LHC) and the four experiments located on its ring are key in the laboratory's research program but many other experiments take advantage of the beams. For example, the protons accelerated in the Super Proton Synchrotron (SPS) are used by COMPASS, NA62, NA64 to study hadrons, rare kaons decays and τ neutrinos, respectively. AEgIS, ALPHA, ASACUSA use the Antiproton Decelerator to study the properties of antimatter.

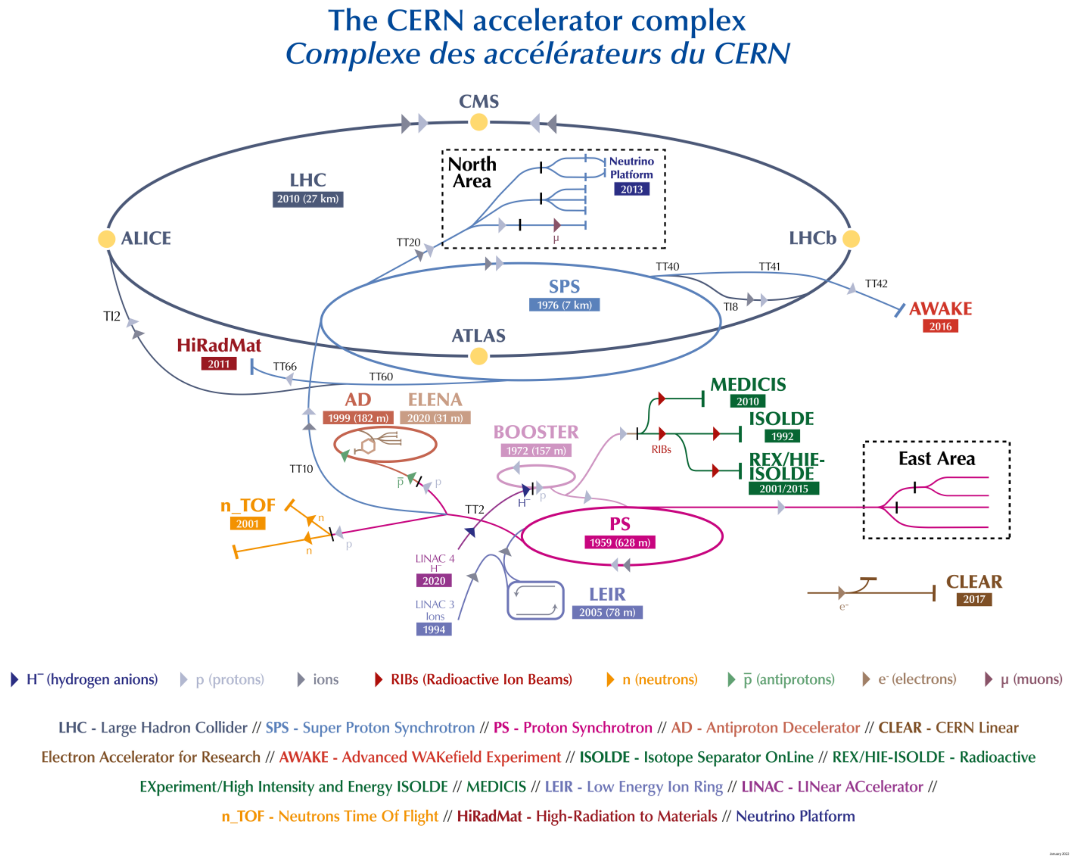

The CERN accelerator complex
Complexe des accélérateurs du CERN
CMS
LHC
2010 (27 km)
North Area
Neutrino Platform
2013
μ
ALICE
LHCb
TT20
TT40
TT41
TT42
SPS
1976 (7 km)
TI8
AWAKE
2016
TI2
ATLAS
HiRadMat
2011
TT66
TT60
AD
1999 (182 m)
ELENA
2020 (31 m)
MEDICIS
2010
ISOLDE
1992
BOOSTER
1972 (157 m)
RIBs
REX/HIE-ISOLDE
2001/2015
East Area
TT10
p̄
p
n_TOF
2001
n
p
TT2
H⁻
PS
1959 (628 m)
LINAC 4
H⁻
2020
LINAC 3
Ions
1994
LEIR
2005 (78 m)
CLEAR
2017
e⁻
H⁻ (hydrogen anions)
p (protons)
ions
RIBs (Radioactive Ion Beams)
n (neutrons)
p̄ (antiprotons)
e⁻ (electrons)
μ (muons)
LHC - Large Hadron Collider // SPS - Super Proton Synchrotron // PS - Proton Synchrotron // AD - Antiproton Decelerator // CLEAR - CERN Linear Electron Accelerator for Research // AWAKE - Advanced WAKefield Experiment // ISOLDE - Isotope Separator OnLine // REX/HIE-ISOLDE - Radioactive EXperiment/High Intensity and Energy ISOLDE // MEDICIS // LEIR - Low Energy Ion Ring // LINAC - LINear ACcelerator // n_TOF - Neutrons Time Of Flight // HiRadMat - High-Radiation to Materials // Neutrino Platform
January 2022

Figure 1.2: CERN Accelerator complex in 2022 [14]

CERN also provides the infrastructure needed to develop and maintain the detectors as well as to record and preserve the research produced by the experiments. The meaning of this has changed alongside technology: with bubble chambers such as Gargamelle recording was done as photographs of the chamber, while modern detectors use electronic readout systems to record millions of data channels to characterise the outcome of particle collisions (called an *event*). Figure 1.3 illustrates this evolution by contrasting a picture taken at the Gargamelle with a 3D representation of the particles detected in a proton-proton collision recorded in 2022 by the LHCb detector. Technology has greatly improved and now allows for electronic readout of the detectors at very high frequency: collisions of particles occur every 25 ns in the detectors at the LHC which produce terabytes of data per second which have to be filtered and stored. The Data Acquisition technologies, the development and maintenance of computer software and the management of the recorded data have therefore become crucial for the research. In June 2022 the CERN Data Centre was running more than 10 000 servers counting more than 450 000 processor cores to process the data. It had a total storage capacity of 634 PB on disk and kept more than 400 PB of data on tape.

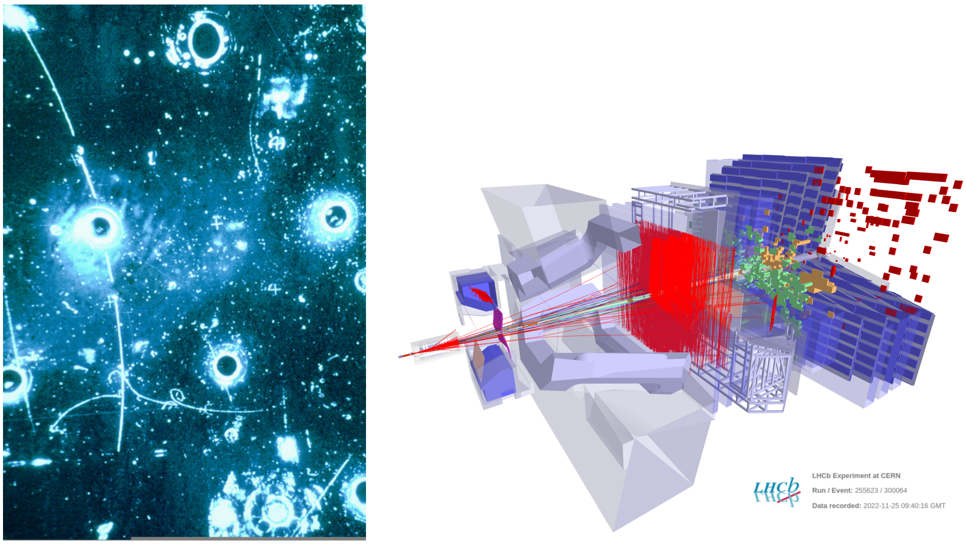

LHCb
LHCb Experiment at CERN
Run / Event: 255623 / 300064
Data recorded: 2022-11-25 09:40:16 GMT

Figure 1.3: Left: Picture of an event recorded at the Gargamelle heavy-water bubble chamber in 1972 [15]. Right: a reconstructed LHCb event from 2022, with the recorded particle hits and a cut-out view of the detector.

1.2 The Large Hadron Collider

Discussions to build a large electron-positron collider (LEP) at CERN started in 1976. A dedicated 27 km underground tunnel (see Figure 1.4) was built at CERN, to host the circular electron/positron accelerator that operated between 1989 and 2000. The Large hadron Collider (LHC) replaced it, with a 27-kilometre ring of superconducting magnets and RF cavities that accelerate protons in order to collide them with a centre of mass energy of up to 14 TeV, with a very high luminosity[1]$(10^{34}\text{cm}^{-2}\text{s}^{-1})$, making it the highest energy and brightest particle collider ever built.

Several accelerators from CERN's accelerator complex (Figure 1.2) are required to inject particles into the LHC. H^- ions are accelerated up to 160 MeV in the LINAC4 and stripped of their electrons at injection in the Proton Synchrotron booster (PSB). The PSB accelerates the protons up to 1.4 GeV before injection into the Proton Synchrotron (PS) from which they exit with an energy of 26 GeV before entering the SPS from which they exit with an energy of

[1] instantaneous luminosity quantifies the density of particles in the colliding beams and can be used to derive the number of particle collisions per unit of time. A higher luminosity means a greater likelihood particles will collide and result in a desired interaction.

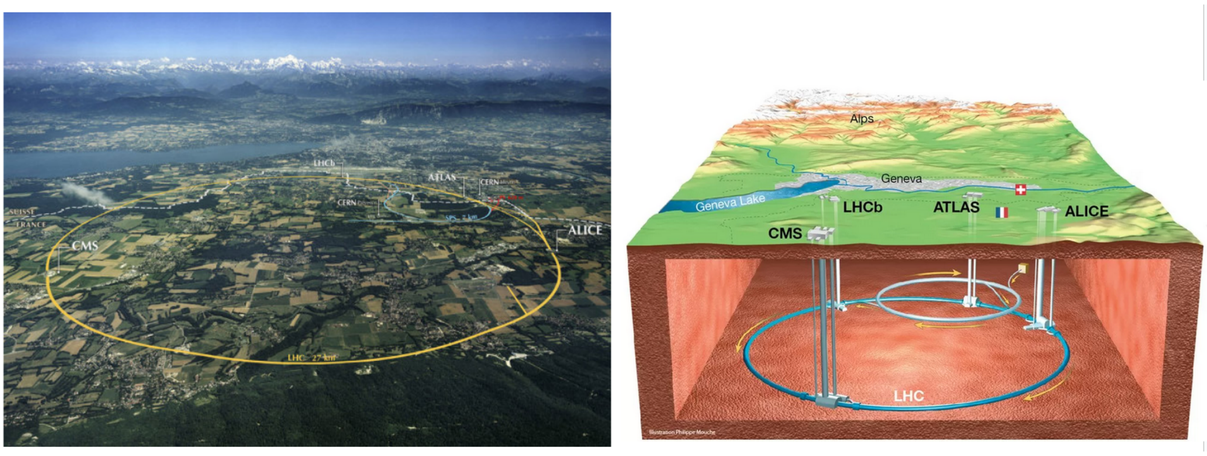


Figure 1.4: The LHC in situation at the Swiss-French border near Geneva.

450 GeV to be injected as two counter-rotating beams in the LHC, where they can be accelerated up to 7 TeV with the beam quality required by the experiments. Protons in the LHC beams are grouped in up to 2808 *bunches* containing around 1.15×10^{11} protons each which cross at each of the four collision points at 25 ns intervals. In order to facilitate the injection of the beams in the accelerator, the potential bunches are not all filled, leading to an effective collision rate of 30 MHz at LHCb instead of 40 MHz. After injection of the bunches and acceleration to nominal energy, experiments record collisions until too many particles have been lost. At that point, the beams are dumped and the process can restart. This phase which typically lasts 15 to 20 hours is called a *fill* as shown in Figure 1.5.

Four experiments are located at the collision points: two general-purpose detectors, ATLAS and CMS that discovered the Higgs boson and make precision tests of its decays and direct searches of new particles, ALICE which studies quark-gluon plasma and LHCb specialising in heavy-flavour physics and as general-purpose detector in the region closer to the beam axis (*forward region*).

Figure 1.5 shows the evolution of the instantaneous luminosity during a typical LHC fill, which in this case lasted nearly 20 hours. The luminosity at LHCb is limited compared to the one available at ATLAS and CMS to avoid saturating the detector and operate it at its optimal point. For this purpose, a specific system is in place called *luminosity levelling* which keeps the luminosity as constant as possible during the fill by adjusting the separation between the two beams.

The LHC is a vital tool for the High Energy Physics field (HEP). It started recording data with a centre of mass energy of 7 TeV in 2010, and has been operated since, increasing the centre of mass energy to 13 TeV. An upgrade of the accelerator to increase the luminosity, i.e. the number of collisions produced is planned to start in 2026, increasing it to 5 to 7.5 times the nominal design value as show by Figure 1.6. Further improvements are planned well into the 2030s. The uniqueness of this machine and associated experiments, as well as their long lifetime, highlight the needs to preserve the recorded data and its full provenance, one of the main topics

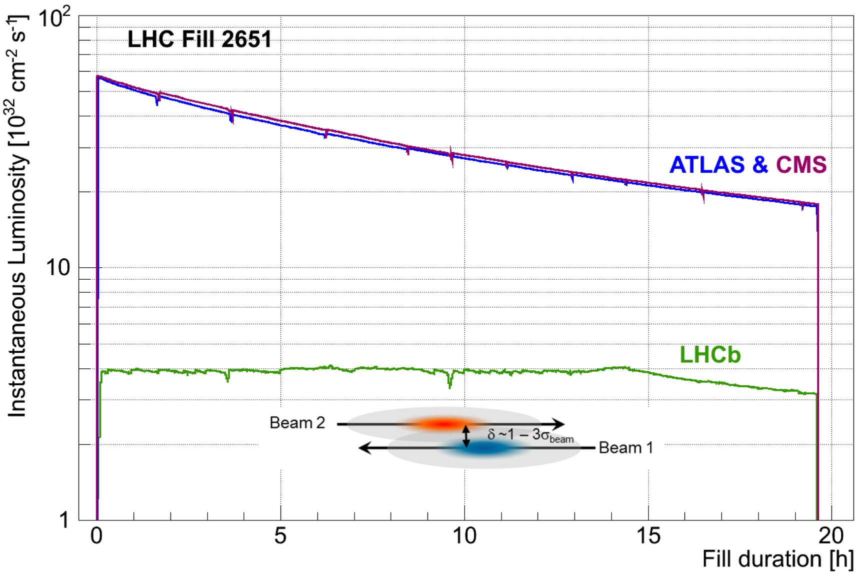


Figure 1.5: Evolution of the instantaneous luminosity during a typical fill of the LHC (number 2651, in 2012).

of this thesis.

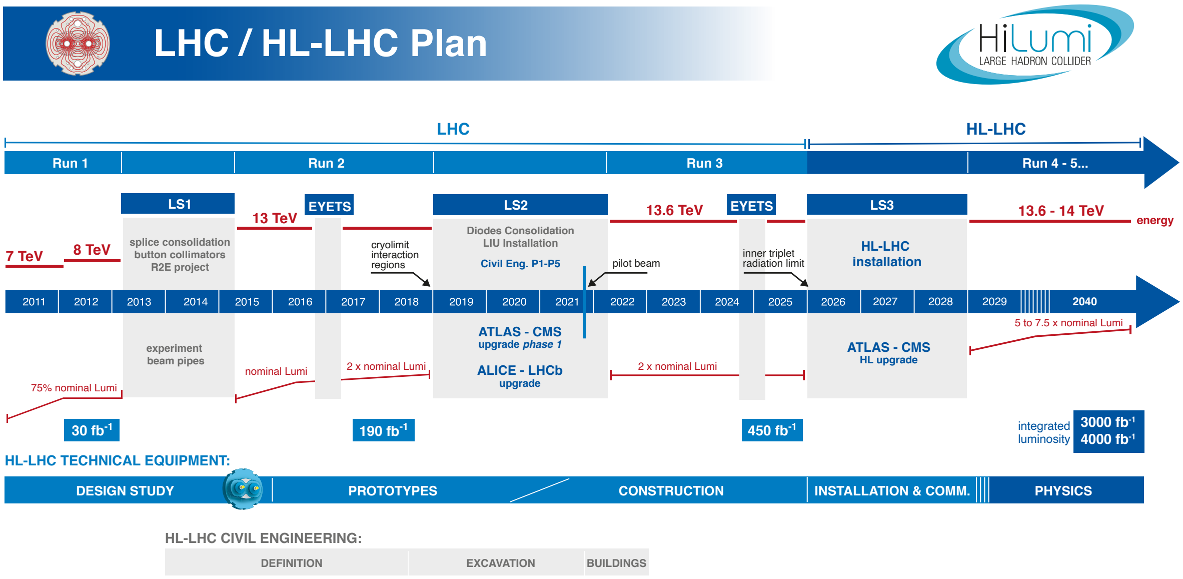


Figure 1.6: LHC and Hi-Lumi LHC project planned schedule.

As all experiments follow the LHC schedule, this work uses the naming convention shown in Figure 1.6:

- **Run 1** corresponds to the LHC operation period 2010-2012.
- **Run 2** corresponds to 2015-2018.
- **Run 3** corresponds to 2022-2025.

In between those running periods, the LHC and experiments were stopped for upgrades during the **Long Shutdown 1 (LS1)** in 2013 until early 2015 and **LS2** from 2019 until the start of 2022. **LS3** is planned for 2026-2028.

1.3 The LHCb Experiment

In proton-proton collisions at the LHC, beauty quarks and their anti-particles (as well as charm quarks and their anti-particles, see Chapter 5) are produced within a small angle with regards to the beam line as shown by Figure 1.7, with a relative high cross section[2] with respect to the total inelastic process. The resulting hadrons can therefore be detected in a small solid angle around the beam. However, due to the large event multiplicity, a very granular detector is needed, with as little material as possible in the particles paths to limit secondary (inelastic or multiple elastic scattering) interactions. During the design of LHC detectors, it became clear that there was an opportunity to study the physics of beauty quarks at that accelerator, with the possibility to reduce costs by reusing the infrastructure already built for the LEP experiments.

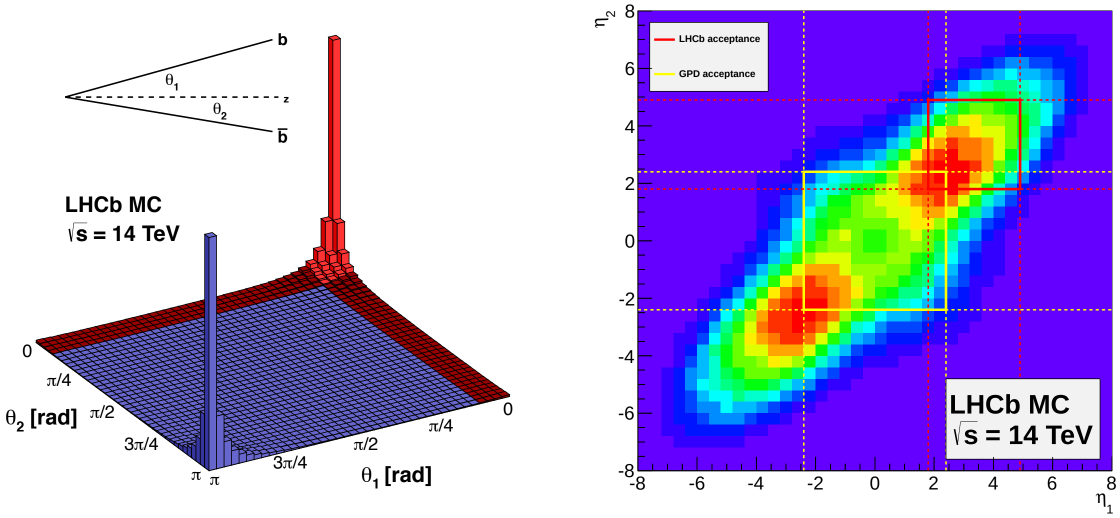


Figure 1.7: (Left) Production cross-section of $b\bar{b}$ pairs as a function of their polar angle with respect to the beam axis, for pp collisions simulated with `PYTHIA8` [16] at a centre-of-mass energy of 14 TeV. The LHCb acceptance is in red [17]. (Right) Two-dimensional pseudorapidity plot of $b\bar{b}$ production phase space in simulated pp collisions at $\sqrt{s} = 14$ TeV with the LHCb acceptance is highlighted in the red square while the General Purpose Detectors (ATLAS and CMS) acceptance is in yellow.

[2] the cross-section of a process is a measure of the probability of it happening in the particle collisions.

The LHCb detector, approved by the LHC Committee (LHCC) in 1998, exploits this opportunity: it is a single arm spectrometer located at the LHC intersection point 8 (see Figure 1.8) of the LHC, which was previously hosting the DELPHI experiment at LEP. The collision point was shifted by 11 m to one side of the cavern: the LHCb detector sacrificed one side of the solid angle to allow for a longer detector thus allowing for better precision in the measurements [18, 19]. The LHCb vertex locator can be moved close to the beam after injection, in order to allow the detector to cover the region where most of the $b\bar{b}$ pairs are produced, corresponding to a pseudorapidity[3] between 2 and 5 that is not covered by general purpose detectors as illustrated by Figure 1.7.

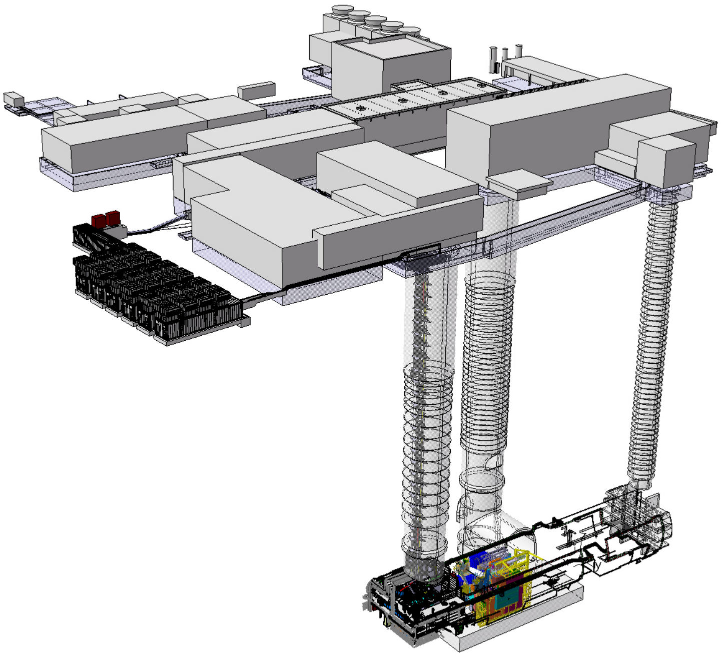


Figure 1.8: Visualisation of the LHC Point8 in Ferney-Voltaire, France. The LHCb detector is visible in the cavern 100 m underground, as well as the three access shafts. The computing facilities used to record and filter the data are located in the containers on the left of the picture.

Figure 1.9 shows the LHCb detector in the context of the cavern initially built for the DELPHI experiment. It occupies the whole of the cavern width (20 m, see also Figure 2.1 for a schematic view of the detector) and its angular acceptance is 10 mrad(0.57°) to 300 mrad(17.2°) in the horizontal plane and 10 mrad to 250 mrad(14.3°) in the vertical plane. In order to measure the momentum of the particles, a dipole magnet produces a vertical magnetic field. It is a warm magnet providing an integrated field of 4 Tm, with sloping poles that match the required detector acceptance. This design allows the majority of the detector to be made in flat planes perpendicular to the beam pipe, and the electronics to be located outside of the detector acceptance.

[3] Pseudorapidity describes the angle between a particle's trajectory and the beam axis and is defined by $\eta = \frac{1}{2}log(\frac{|\mathbf{p}|+p_z}{|\mathbf{p}|-p_z})$, with $\mathbf{p}$ the total momentum and p_z the longitudinal momentum of the particle.

Figure 1.9: The LHCb Run1/Run2 detector side view in the cavern.

A first version of the detector was successfully operated from 2010 to 2018 during the LHC Run1 (2010–2012) and Run2 (2015–2018) data-taking periods with excellent performance [20]. It collected 9 fb^{-1} of proton-proton (pp) collision data, about 30 nb^{-1} of Pb-Pb and p-Pb collisions and about 200 nb^{-1} of fixed target data. In spite of these excellent results, the LHCb collaboration planned and built an upgrade of the detector physics during the LHC long shutdown 2 (2019 until 2022) in order to accumulate larger statistics for decays of interest.

1.4 LHCb results so far

During the LHC Run 1 which started in 2010 with pp collisions at a centre of mass energy of 7 TeV, before being upgraded to 8 TeV, LHCb recorded around 3 fb^{-1} of data. After a two-year shutdown, the centre of mass energy was upgraded to 13 TeV and LHCb recorded an integrated luminosity[4] of around 6 fb^{-1}, as shown by Figure 1.10. Increasing the energy in the beam also increases the cross-section for the b-quark production as measured in [21].

The experiment collected and analysed an unprecedented number of $b-$decays, allowing to measure the CKM quark mixing matrix elements and CP violation parameters to world-leading precision, to observe CP violation in charm decays [23] as well as charm mixing [24] and discover many new resonances (shown in Figure 1.11) including exotic four- and five-quarks states. LHCb also provided significant measurements regarding Lepton Flavour Universality (LFU), which are detailed in Chapter 5.

[4] The integrated luminosity is measures the total number of collisions over a certain period of time.

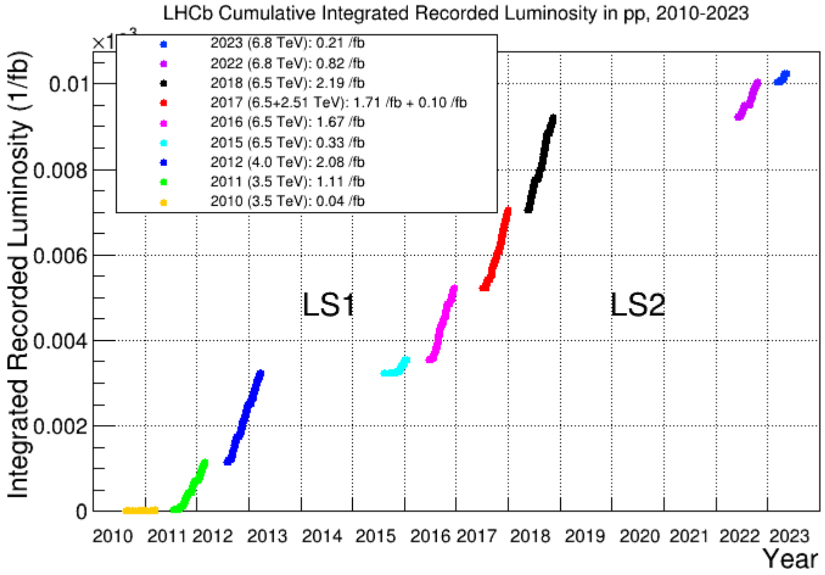


Figure 1.10: Integrated luminosity recorded by LHCb as a function of time

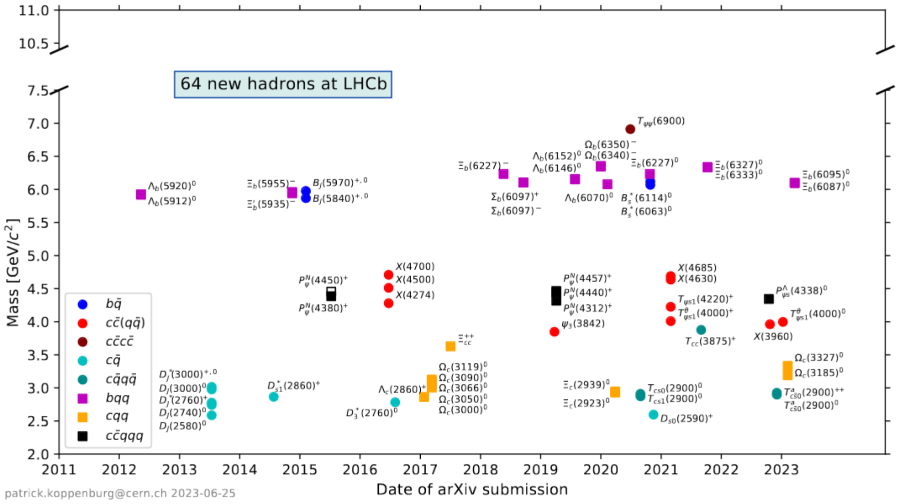


Figure 1.11: Hadrons discovered at LHCb, according to [22]

1.5 Conclusion

The LHCb detector, one of the four main experiments located on the LHC accelerator, has been recording data since 2010. The potential of the recorded data has not been fully exploited yet, and physicist are using it to validate and challenge various aspects of the standard model. The next Chapter details the detectors and the data processing required to perform measurements.

Chapter 2

The LHCb detector and data processing chain

This chapter presents the design of the LHCb detector, how information on the particles generated in collisions is gathered, namely how the raw information provided by the detectors is processed in order to derive quantities that can be used to analyse the physics processes at play. The detector initially built and operated during Run 1 and Run 2 is first described. The improvements made for Run 3 are summarised in Section 2.1.5.

2.1 LHCb Detector design

Figure 2.1 is a cut of the LHCb detector in the cavern hosting it, alongside the beam axis (the LHC beam pipe is not shown). The particles collide within the Vertex Locator, on the left of the picture and are detected by a set of detectors placed in the cavern. The compromise made to limit the detection to one side of the collision (the *forward region* and to a zone close to the beam-line is clearly visible on this diagram. The coordinate system used in LHCb has its origin at the nominal pp interaction point, the z axis along the beam points towards the muon system, the y axis points vertically upward and the x axis defines a right-handed system. Most of the subdetectors (except for the Cherenkov ones) are split into two mechanically independent halves (the access side or Side A at $x > 0$ and the cryogenic side or Side C at $x < 0$), which can be opened for maintenance. In order to measure the momenta of the charged particles crossing the detector and to distinguish between different species (π^+, K^+, p, etc), LHCb uses a magnet to deflect the particles, a tracking system and a particle identification system which are described in the following sections.

2.1.1 Magnet

As a spectrometer, LHCb features a warm dipole magnet, described in Refs [25, 26, 27]. It provides a vertical magnetic field with a bending power of 4 Tm, that deflects charged particles in the horizontal plane. Two saddle-shaped coils are mounted symmetrically inside the yoke. The gap between the poles increases towards the downstream tracking stations to match the

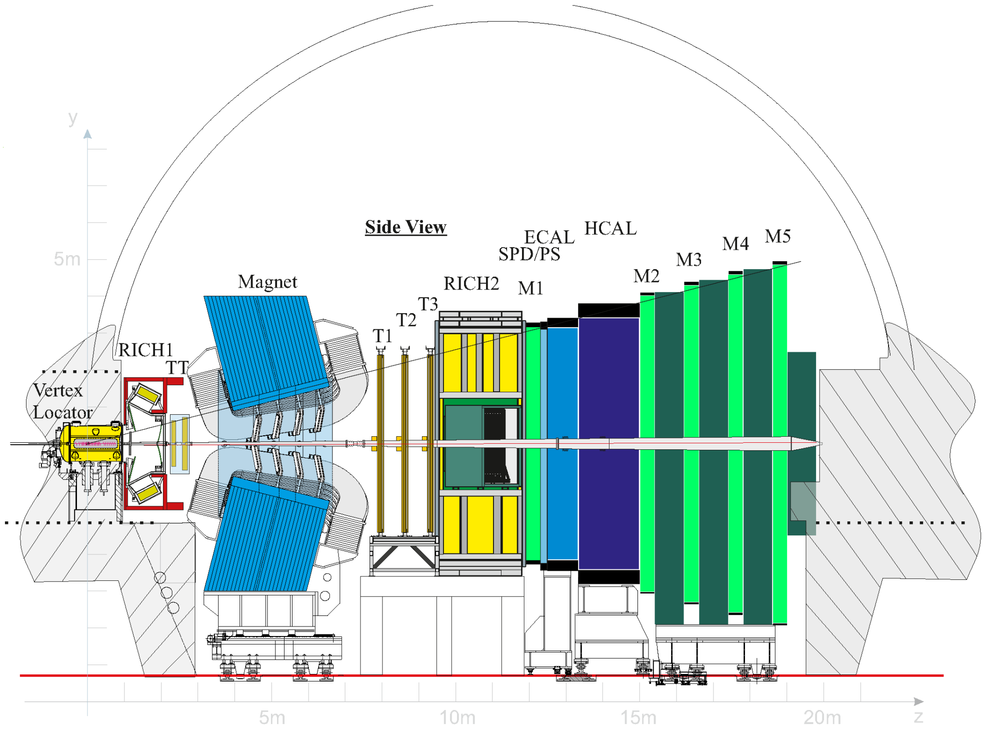


Figure 2.1: Schematic representation of the LHCb detector (side view).

detector acceptance. During data taking, the magnet polarity is reversed every few weeks to collect data sets of roughly equal integrated luminosities with the two field configurations (MagDown, MagUp) in order to reduce experimental biases. This is particularly important for measurements of CP violation or asymmetries in general.

2.1.2 Tracking

Particle tracking in LHCb is done using the Vertex Locator (VELO), the Trigger Tracker (TT) placed upstream of the magnet, and the Inner and Outer trackers located downstream. The VELO is crucial for the physics program of LHCb as it is fundamental in the identification of displaced, secondary vertices from long-lived b- or c-hadron decays. An excellent spatial resolution is required also to measure the fast oscillations of the $B_s^0 - \bar{B}_s^0$ system and related CP violation observables. A precise tracking is needed to achieve high-precision momentum and invariant mass resolutions, which play an important role in suppressing background and/or identifying new resonances. A limiting factor to high performing tracking is the amount of material on the path of the detected particles, which is minimised as much as possible to reduce multiple scattering while maintaining the highest tracking efficiency.

Vertex Locator

The LHCb VELO is unique at the LHC, as its movable sensors can be brought very close to the colliding proton bunches, recording the tracks of particles with trajectories close to the beam axis. The VELO sensors are organised in two halves, kept in a secondary vacuum and separated from the LHC beam by aluminium RF-boxes which are 250 µm thick on the side facing the beam, as shown by Figure 2.2. The RF-boxes protect the VELO sensors from the fields induced by the charged particles in the LHC beam. The VELO halves can be moved from a distance of 35 mm from the LHC beam during the injection of the protons into the LHC, to 8 mm from the beam when it is properly focused and the protons have reached their final energy.

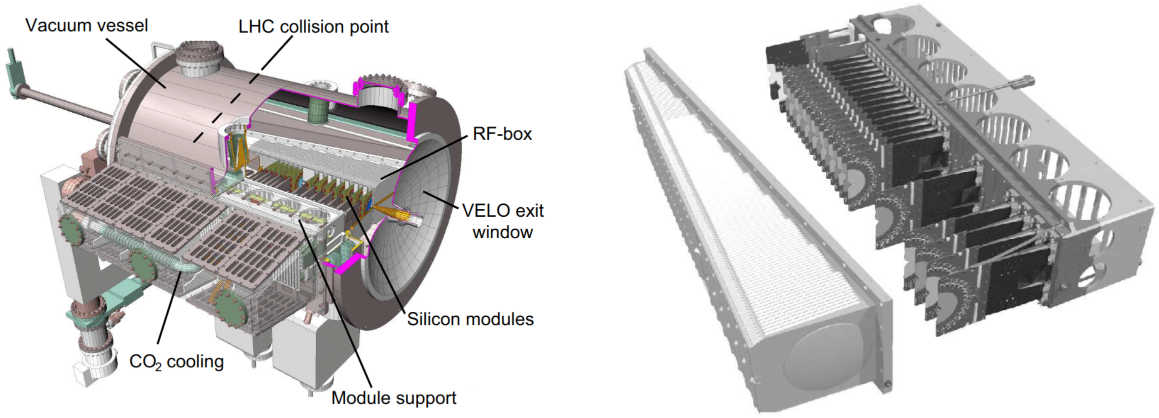


Figure 2.2: Left: view of the vacuum tank containing both sides of the VELO detector. Right: view of the modules on a module support and the RF-box that surrounds them

Each side holds 23 modules of 2 sensors which are silicon strip detectors with a pitch of 38 µm to 102 µm, one providing of the radial distance to the beam and the other providing a measurement of the angle ϕ around the beam. They are arranged in the z direction as per Figure 2.3.

This proximity to the LHC beam and the minimal material present between the origin and the first measurement of the tracks allows for a resolution in the position the pp collision point (primary vertex) of 13 µm in the transverse plane and 71 µm in the z direction [28].

Upstream tracking

The Trigger Tracker (TT) is placed upstream of the magnet, after the VELO and the RICH1 Cherenkov detector. This Silicon tracker covers the full LHCb acceptance and is composed of four vertical silicon microstrip detectors with a pitch of 200 µm in a $x-u-v-x$ layout where the u and v layers are rotated by $-5°$ and $+5°$ with respect the y direction respectively. This configuration is a compromise which offers precise hit reconstruction in the horizontal plane, which is crucial for quantifying the momentum of tracked particles, at the cost of less accuracy in the vertical plane.

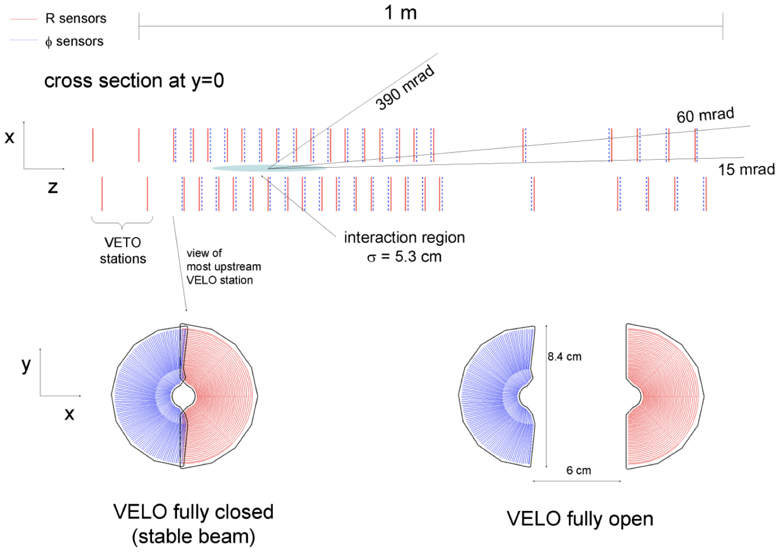


Figure 2.3: Top: position of the VELO sensors in the $x - y$ plane. Bottom: two modules in the closed and open positions.

Downstream tracking

Tracking downstream of the LHCb magnet is performed by two sub-detectors:

- the Inner Tracker (IT) covers the innermost region of the acceptance where the occupancy is highest. This silicon detector is cross-shaped, 120 cm wide and 40 cm high and provide detailed tracking in the centre of three large planar tracking stations downstream of the magnet. Each of the four Silicon Tracker stations consists of four detection layers, with a layout similar to that of the TT.

- The Outer tracker (OT) [29] is a gaseous straw tube detector which covers an area of approximately 5 x 6 m^2 with 12 double layers of straw tubes (for a total of around 55000). They are grouped in three stations, T1, T2, T3 made of two halves in order to facilitate maintenance and give access to the beam pipe as shown by Figure 2.4. Each station is made of four layers, in a $x - u - v - x$ layout where the u and v are rotated by $-5°$ and $+5°$ with respect the y direction, in the same fashion as the Silicon Tracker microstrips.

When charged particles enter a tube, the gas mixture is ionised and the electrons are attracted towards the anode at the centre of the straw. As the electron drifts, a phenomenon known as Townsend discharge occurs amplifying the signal to a level that allows it to be detected by electronics connected to the wire. The straw tubes are 2.4 m long with 4.9 mm inner diameter, and are filled with a gas mixture which guarantees a fast drift-time (below 50 ns).

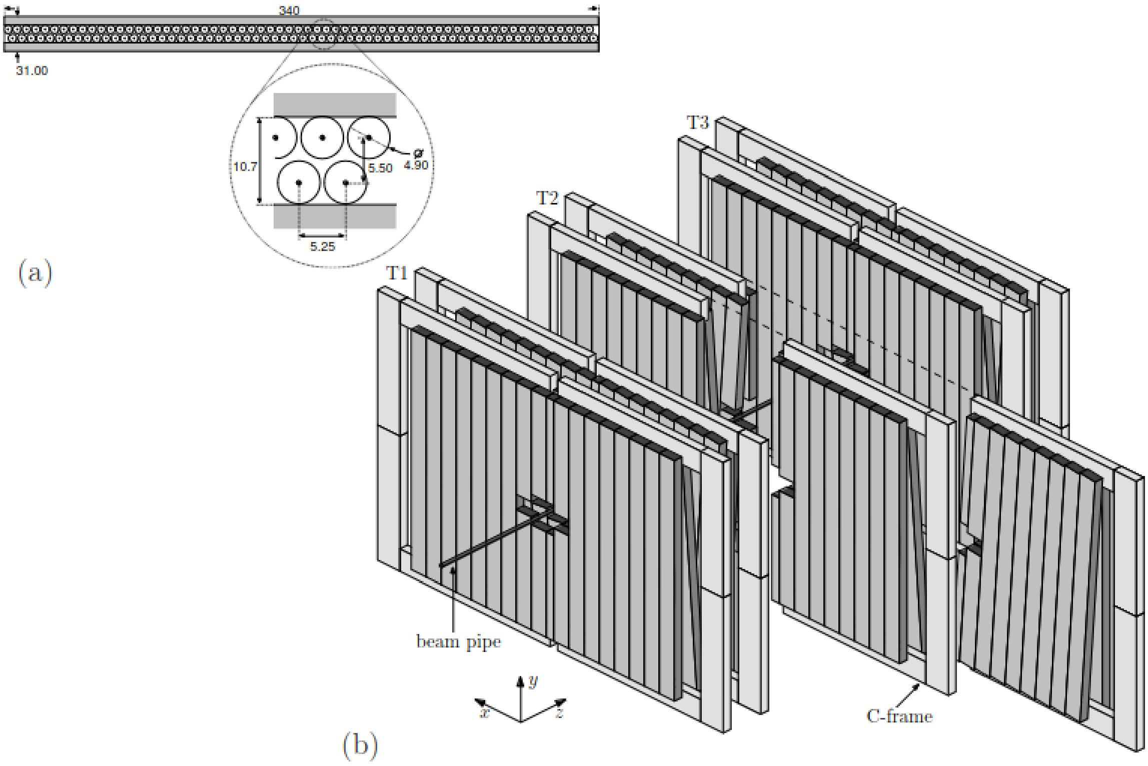


Figure 2.4: (a): Module cross section. (b) OT module layout. Figure taken from Ref. [29].

2.1.3 Particle Identification

Particle identification (PID) at LHCb is performed by several detectors, arranged as shown in Figure 2.1. Two Ring-Imaging Cherenkov detectors are placed upstream and downstream of the magnet. A calorimeter system is used, consisting of: a Scintillating Pad Detector (SPD), a PreShower detector (PS), a shashlik-type electromagnetic calorimeter (ECAL), an iron-scintillator tile sampling hadronic calorimeter (HCAL). Five muon stations are placed further downstream. The success of the LHCb experiment largely depends on the excellent performances of its PID detectors, which are crucial for signal identification and, even more important, background rejection. Their information is also used at trigger level for an efficient selection of interesting decays.

RICH detectors

Ring Imaging Cherenkov detectors (RICH) use the radiation emitted by charged particles travelling through a medium with a velocity larger than that of the light in the media. Cherenkov radiation is emitted in a cone around the particle direction with an aperture angle depending on the particle speed. These photons are reflected and focused by mirrors on the focal plane outside of the detector acceptance, where they are detected by Hybrid Photon Detectors. Figure 2.5 shows the Cherenkov angle as a function of particle momentum for the different particle species as measured by RICH1 and illustrates how they can be separated.

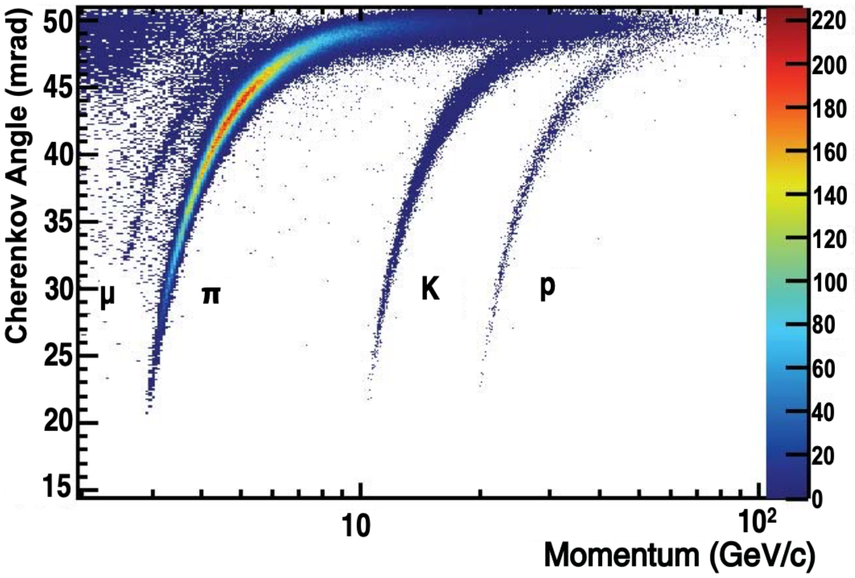


Figure 2.5: Reconstructed Cherenkov angle for isolated tracks from LHCb data recorded in 2011, as a function of track momentum in the C4F10 radiator of LHCb RICH1 [20]

RICH1 allows identifying low momentum particles while RICH2 located downstream of the magnet identifies high momentum particles, albeit with a more limited acceptance.

Calorimeters

Calorimeters are used to measure the energy and position of electrons, photons and hadrons crossing the detector. Unlike trackers, they are destructive and aim to stop the particles, measuring their energy by characterising the avalanches of secondary particles generated in the interaction with the absorbers. Most calorimeters use scintillators to detect photons generated by the avalanche of charged particles crossing them. At LHCb wavelength shifting fibres are used to collect the scintillator light and transport it efficiently to the Photo Multiplier Tubes (PMTs). The calorimeter system at LHCb consist of:

- The **Scintillating Pad Detector (SPD)** used to distinguish charged and neutral particles as they enter the calorimeter.
- The **PreShower (PS)** which helps distinguishing between electrons, photons and pions,
- The **Electromagnetic Calorimeter (ECAL)** which measures the position and transverse energy of electrons, photons and neutral pions.
- The **Hadron Calorimeter** is intended to provide the Level-0 trigger on presence of high transverse momentum hadrons, a signature of $b-$hadron decays.

The calorimeters are split in regions with higher granularity when close to the beam pipes. This ensures a reasonable occupancy in the calorimeter cells while keeping the cost of the covering the whole acceptance of the detector reasonable.

Muon system

The LHCb Muon system [30] counts 5 muon stations (M1 to M5) featuring a total of 1380 chambers, mostly multi-wire proportional chambers (MWPC), except for the inner part of M1 which is equipped with 12 triple Gas-Electron-Multipliers (GEM). Each station is split in two sides that can be moved horizontally to access the beam pipe and the detector chambers for installation and maintenance. Station M1 is located upstream of calorimeters while stations M2 to M5 are placed downstream of the calorimeters and are interleaved with iron absorbers 0.8 m thick to suppress the remaining hadron contamination. This detector traces all the charged particles that did not interact with the calorimeters or muon filters.

2.1.4 Trigger

The LHCb detectors records collision happening at a frequency of 30 MHz and produces terabits of data every second. Recording all this data would be very challenging and costly. A way to filter the data as it is being produced is therefore required. This is the role of the LHCb trigger [31, 32] which during Run 1 and Run 2 counted three stages:

- The **Level 0 trigger (L0)** is implemented with electronics directly in the detector and it reduces the event rate to 1 MHz. It uses information from the muon chambers to identify muons with high transverse momentum, information from the calorimeters to find charged or neutral hadrons with significant transverse momentum likely to be the product of decays of particles containing b quarks.

- The **High Level Trigger level 1 (HLT1)** is implemented in software. It performs a partial track reconstruction and keeps interesting events, reducing the event rate by a factor 100.

- The **High Level Trigger level 2 (HLT2)** also implemented in software. It performs a full reconstruction of the events selected by the HLT1 and performs analysis-dependent selections. The selected events are stored offline and made available for analysis by the members of the collaboration.

This scheme evolved during the life of the experiment; methods were found to provide better ways to filtering, notably by the introduction of a disk buffer between HLT1 and HLT2 in 2014, the development of output files that are ready for analysis without further offline processing (Turbo stream), and a full software trigger for the LHCb upgrade. These improvements will be detailed in Sec. 2.3.

2.1.5 The LHCb Upgrade I

From 2010 to 2018 LHCb collected a considerable data set of proton-proton, lead-lead, lead-proton and fixed target collision data. The measurement of many key observables for flavour physics are limited by statistical uncertainty, and would benefit from more data being recorded. The LHCb Run 1-2 system design however does not allow a significant increase in statistics, especially for fully hadronic final state decays. The main limitation is due to the maximum rate of the L0 trigger rate, implemented in hardware.

In order to increase the collected samples by an order of magnitude, the LHCb upgrade increases the luminosity of the detector by a factor 5, and replaces the existing trigger by a new system implemented in software. With the increase of luminosity, every event in the LHCb acceptance contains on average two long-lived hadrons not containing heavy quarks which means that simple cuts based on displaced vertices or on transverse momentum are not effective at rejecting background. Removing the L0 trigger stage and introducing more discriminating selections based on the full event reconstruction is therefore essential for this strategy.

To adapt to the increased luminosity the detector readout system has been upgraded to record events at up to 40 MHz, and the detectors have been upgraded during the LHC Long Shutdown 2 (2019-2022), some replaced altogether. The new detector is described in Ref. [33]:

- the VELO has been replaced by a new detectors using 52 modules featuring 12 hybrid pixel detector sensors each. Each sensor is a matrix of 256x256 square pixels with a pitch of 55 µm.

- The Upstream Tracker (UT) using silicon strip detectors replaces the TT to provide tracking information between the VELO and the magnet.

- The downstream tracking systems (IT and OT) have been replaced by a Scintillating Fibre tracker (SciFi) consisting of 3 tracking stations. Each of the tracking stations is made of four layers with the same configuration as in the OT.

- The Hybrid Photo Detectors in the RICH detectors have been replaced by Multi Anode Photon Multiplier Tubes with a higher resolution. The RICH1 structure and the mirrors used to focus Cherenkov light have been replaced.

- SPD, PS and Muon station M1 have been removed because they would not provide useful information due to the larger occupancy, and to make place for the SciFi and for a neutron shielding system made of polyethylene with Boron to protect the Silicon Photo Multipliers (SiPMs) used by the SciFi from the flux of neutrons generated by the particle showers in the calorimeters.

- The L0 trigger has been removed and all the electronics used for detector readout has been replaced by new systems allowing for detector readout at up to 40 MHz. This makes possible the realisation of a trigger fully implemented in software described in Sec. 2.3.

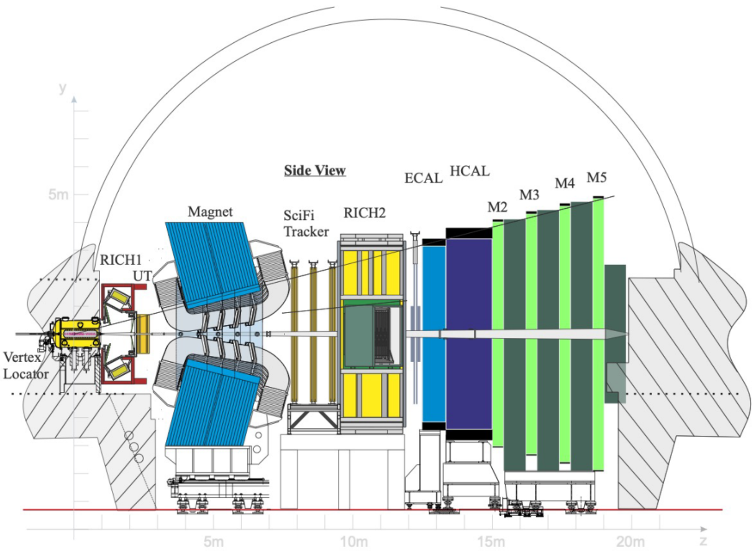


Figure 2.6: The LHCb Upgrade I detector used in LHC Run 3.

2.1.6 Conclusion

The LHCb detector exploits the unique capabilities of the LHC and its design was optimised to perform very precise measurements of the characteristics of heavy-flavour particles produced in *pp* collisions. Its tracking system offers great precision on the measurements of displaced vertices, which is a great asset to correctly identify the topology of long-lived particle decays. Its particle identification system gives excellent performance in the identification of particle decay products and allows for signal decay reconstruction with low background. For this reason LHCb can perform measurements that make it a general-purpose detector in the forward region. Its sub-detectors have been upgraded between 2019 and 2022 to improve performance. A matching data acquisition and processing chain has been put in place to record the large amounts of data needed to perform measurements. It is described in the next section.

2.2 Data acquisition

This part explains how the signals from all subdetectors are gathered and processed to identify the physical processes involved in the collisions produced in LHCb. The set of computing systems installed at the LHC point 8 alongside the LHCb detector (globally called the *online system*) are crucial to control the experiment and gather the signals from all the sub-detectors in a timely fashion. The online system is described in Ref. [18] for the Run 1/Run 2 system and in Ref.[33] for the one used in Run 3. Figure 2.7 shows the architecture of the data acquisition part of the system for Run 3, which uses 11000 fibre links to read data for the detector and to assemble them on the so-called *event-builder* farm, where they are processed by the LHCb software trigger. The events recorded at this stage are called RAW events and contain only the output of the subdetectors composing LHCb.

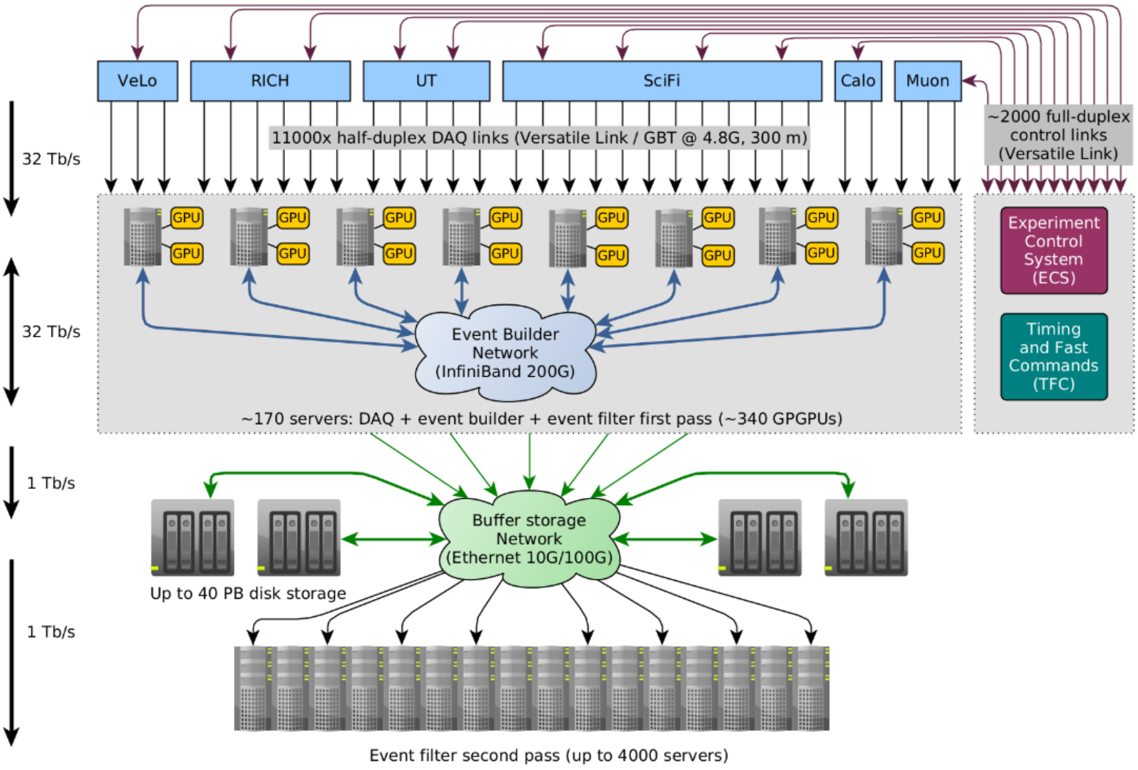

VeLo
RICH
UT
SciFi
Calo
Muon
32 Tb/s
11000x half-duplex DAQ links (Versatile Link / GBT @ 4.8G, 300 m)
~2000 full-duplex control links (Versatile Link)
GPU
32 Tb/s
Event Builder Network (InfiniBand 200G)
Experiment Control System (ECS)
Timing and Fast Commands (TFC)
~170 servers: DAQ + event builder + event filter first pass (~340 GPGPUs)
1 Tb/s
Buffer storage Network (Ethernet 10G/100G)
Up to 40 PB disk storage
1 Tb/s
Event filter second pass (up to 4000 servers)

Figure 2.7: Data Acquisition system design for LHCb Upgrade I (Run 3).

As from this stage, software is used to process the recorded data. Tracing the processing is crucial for the analysis and the preservation of the data as it is necessary to measure the biases that could be introduced by the event processing and selection steps. We now explore the data flow and the tools involved.

2.3 Data processing

Figure 2.8 gives an overview of the data processing in Run 1 and Run 2, from the trigger to analysis. The processing is split in two distinct parts:

- An *online* part performed in the computer centre at LHC point 8, with the goal to reduce the data volumes to a size reasonable enough to be stored offline in a sustainable way.
- An *offline* part, performed on the Worldwide LHC Computing Grid (WLCG) [34] which prepares the data for analysis by the members of the collaboration.

The offline part of the processing is further split into two different phases:

- a central processing phase which includes steps common to all analyses (see Section 2.8). This includes the reconstruction of the events in Run 1, but also deals with filtering and indexing the events, separating them in different files depending of the physics processes.

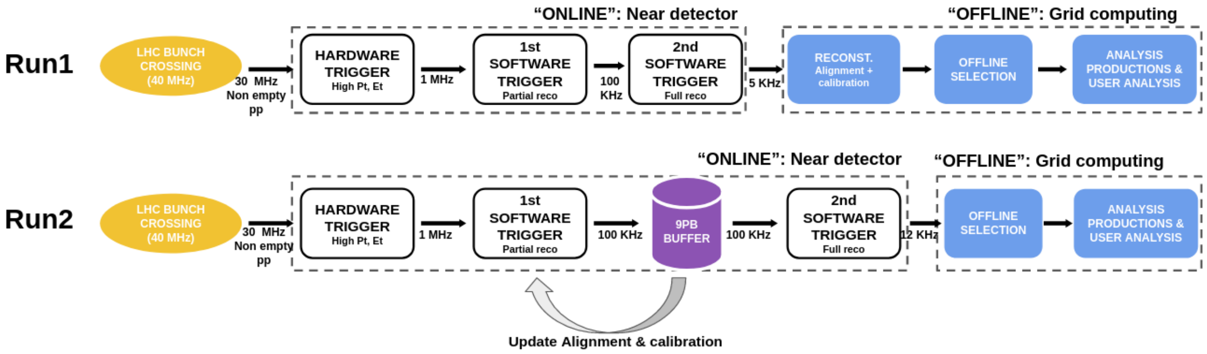

Run1
LHC BUNCH CROSSING (40 MHz)
30 MHz Non empty pp
"ONLINE": Near detector
HARDWARE TRIGGER High Pt, Et
1 MHz
1st SOFTWARE TRIGGER Partial reco
100 KHz
2nd SOFTWARE TRIGGER Full reco
5 KHz
"OFFLINE": Grid computing
RECONST. Alignment + calibration
OFFLINE SELECTION
ANALYSIS PRODUCTIONS & USER ANALYSIS
Run2
LHC BUNCH CROSSING (40 MHz)
30 MHz Non empty pp
"ONLINE": Near detector
HARDWARE TRIGGER High Pt, Et
1 MHz
1st SOFTWARE TRIGGER Partial reco
100 KHz
9PB BUFFER
100 KHz
2nd SOFTWARE TRIGGER Full reco
12 KHz
"OFFLINE": Grid computing
OFFLINE SELECTION
ANALYSIS PRODUCTIONS & USER ANALYSIS
Update Alignment & calibration

Figure 2.8: LHCb data processing in Run 1 and Run 2.

This is described in detail in Ref. [35]. Once this is done, the data is registered as official LHCb data in the bookkeeping system (see Section 3.2.1),

- the last step, *Analysis productions and user analysis*, consists in the extraction of the relevant data by analysts and all the processing needed to perform the physics measurement based on this data. This in itself is a considerable amount of work that is difficult to characterise in a generic way as it is very dependent on the measurement performed.

As explained in Section 2.1.4, LHCb needs to filter the data to keep the collisions that are interesting from a physics point of view. This filtering is indispensable to keep the data stored within the limits of the resources available[1], but it is very difficult to perform it with a 30 MHz collision rate while retaining the events with interesting physics.

In Run 1, the L0 hardware trigger was used to identify events with muons or hadrons with large transverse momentum, reducing the event flow to 1 MHz which could then be processed by the two-level software trigger. However the reconstruction and filtering performed were done with limited precision due to the time constraint imposed by the need to keep up with the flow of data from the detector. For this reason, a second reconstruction and filtering was performed offline without this time limitation. The output of this second phase was used for physics analyses.

In order to improve the quality of the online reconstruction, and to minimise its differences with respect to offline, this scheme was changed for Run 2 to the so-called *deferred triggering* scheme. Within this scheme, shown in Figure 2.8, the events passing HLT1 were stored on a large disk buffer of 9 PB. Alignment and calibration software then processed the recorded data to allow for a more precise reconstruction of the events with a quality equivalent to what could be achieved offline [36]. At this stage, it became possible to fully reconstruct the events online and to keep only the quantities required for physics, discarding the raw information from the detectors. This was implemented at LHCb in the Tesla [37] application and the events stored in this way known as the *Turbo Stream*, which ran as a complement to the classic way of saving LHCb data to disk. Events persisted this way cannot be reconstructed further but as their size

[1] During Run 3, it is planned that LHCb will record 10 GB s^{-1} when LHC is running, leading to around 60 PB of data per year.

is typically one order of magnitude smaller that those where the raw subdetectors information is kept, they allow to store more physics events within the available resources. This scheme was enhanced further with the introduction of *Selective Persistence* [38] where trigger developers can choose which parts of the reconstructed event should be saved in the Turbo Stream. This became the basis for the *Real-Time Analysis* paradigm put in place for the LHC Run 3.

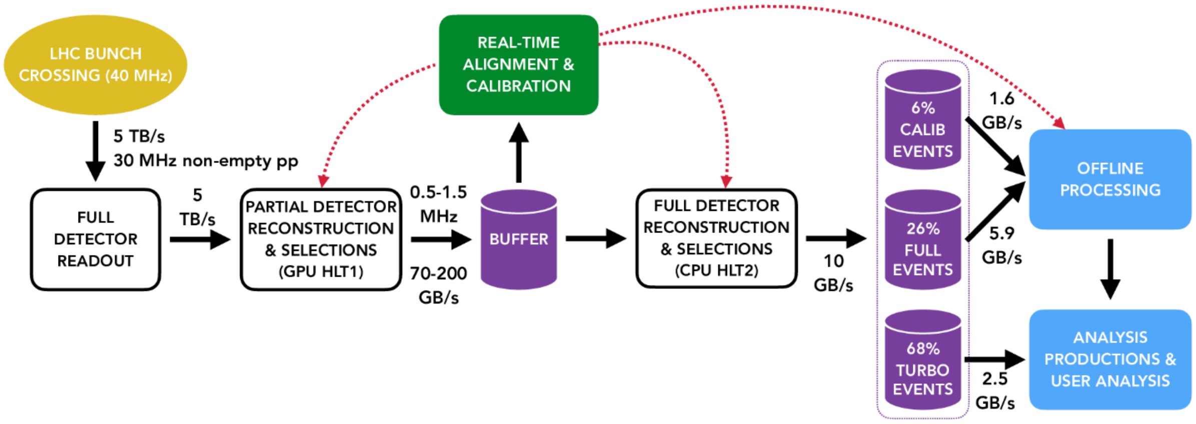

LHC BUNCH CROSSING (40 MHz)
5 TB/s
30 MHz non-empty pp
FULL DETECTOR READOUT
5 TB/s
PARTIAL DETECTOR RECONSTRUCTION & SELECTIONS (GPU HLT1)
0.5-1.5 MHz
70-200 GB/s
REAL-TIME ALIGNMENT & CALIBRATION
BUFFER
FULL DETECTOR RECONSTRUCTION & SELECTIONS (CPU HLT2)
10 GB/s
6% CALIB EVENTS
26% FULL EVENTS
68% TURBO EVENTS
1.6 GB/s
5.9 GB/s
2.5 GB/s
OFFLINE PROCESSING
ANALYSIS PRODUCTIONS & USER ANALYSIS

Figure 2.9: LHCb data processing in Run 3.

As already mentioned, in order to increase further the physics reach of the experiment, the LHCb detector was upgraded during LS2 and the trigger was fully implemented in software, removing the existing L0 trigger. Figure 2.9 illustrates the effect of this change on the data flow:

- The HLT1 has to process the full flow of data coming from the subdetectors (5 $\mathrm{TB\,s^{-1}}$). To do so GPGPU (General Purpose Graphics Processing Units) accelerators were placed directly on the event builders in the online system (Figure 2.7). The Allen application [39] is used to perform the filtering.

- The HLT2 is run on CPUs in a server farm as was done previously.

- Most of the events are kept in the Turbo Stream. A small fraction of events have to be kept with the full detector information in order to accommodate analyses requiring large amounts of processing that cannot be performed on the online farm, and to check the quality of the data recorded (but this ends up using a large fraction of the bandwidth available).

- An offline processing is still needed and shown in Figure 2.10. The Stripping/Sprucing application [40] is used to filter the events with full detector information and acts as a pass-through for the Turbo Stream.

Figure 2.7 gives an idea of the resources required to for the online data processing: around 170 servers are needed for the event builder which assembles the data from all subdetectors. Two GPGPU per event builder server (so 340 in total) are needed for the HLT1. The HLT2 runs on the event filter farms which counts up to 4 000 servers.

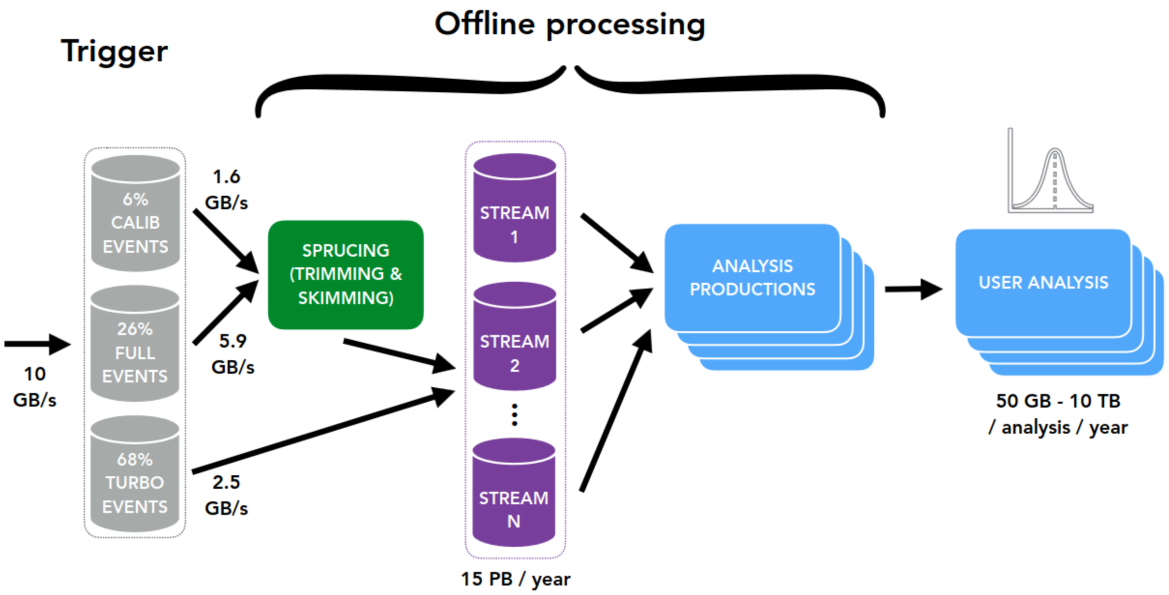


Figure 2.10: LHCb offline data processing in Run 3.

The evolution of the data processing model has undoubtedly increased the importance of software in the processing chain as a way to optimise the use of resources. As the experiment only allows to reconstruct events from scratch on a limited fraction of the dataset, the quality of the software used to process the data has become crucial. The applications used and the development process is described in the next section.

2.4 LHCb Software

LHCb uses the C++ [41] based Gaudi [42] event processing framework for most of its applications, with the exception of the HLT1 which uses the Allen framework [39] to run on GPGPU accelerators, as well as on traditional computer architectures. While the core of the algorithms are written in the C++ language, the configuration of the applications is done using Python [43]. Many external software packages are used; they are compiled and released for use by High Energy Physics (HEP) experiments as part of the common LCG Releases [44] also used by ATLAS and other experiments. Some packages are specific to HEP such as ROOT [45], used for data processing and persistency, DD4Hep [46], used to build the 3D model of the detector needed for reconstruction and simulation, Geant4 [47], used to simulate the interactions of the particles with the detector, and a number of event generators such as Pythia [16], tuned to match the events occurring in the collisions.

The LHCb code base is organised in projects with well identified roles. Table 2.1 lists the major projects that count around 1.5 million lines of C++ (and a lot of Python configuration), while Figure 2.11 illustrates the dependencies. To develop and maintain such a software infrastructure, the work of many contributors is required. As an example, in the three months between the 10th of April and the 10th of July 2023, the LHCb software projects in GitLab

Project	**Description**
Gaudi	Base framework (in common with the ATLAS experiment)
LHCb	Common tools and interface used by all applications
Detector	Representation of the detector
Lbcom	Common tools that are not used by the simulation
Rec	Algorithms and tools used in event reconstruction
Analysis	Tools used only for offline analysis
Allen	HLT1 Trigger framework and application
Moore	HLT2 Trigger application
DaVinci	Analysis application
Run2Support	Functionality needed to simulate the Run 1/Run 2 detector
Gauss	Simulation application
Online	Tools needed by the DAQ to record the data
MooreOnline	Trigger software linked with Online to read data from the Online system
Alignment	Algorithms to measure the detector alignment constants from recorded data
Panoptes	RICH Online Monitoring and Calibration Application

Table 2.1: Main LHCb software applications tested in the continuous integration system in July 2023 (non exhaustive list). Together they counted then around 1.5 million lines of C++ code (measured with cloc [48]), excluding dependencies such as ROOT or Geant4.

counted 262 contributors. Modern software development infrastructure is needed: the code is versioned using the Git source code versioning system [49], and the instance of the GitLab [50] development and operations platforms hosted at CERN is used. A custom continuous integration system using the Jenkins [51] open source automation server was also developed to fulfil the needs of the experiment.

2.4.1 Data processing steps

Interpreting the raw data recorded by the LHCb detector requires several steps: reconstructing the trajectories of the particles, identifying them and then filtering for interesting physics. This part gives a brief overview of the processing involved.

Reconstruction

Reconstruction of the charged particles tracks is the first task when processing the data recorded by LHCb. Charged particles crossing the sensors in the tracking detectors create so-called *hits*. Individual hits from all sensors must be matched to reconstruct the original tracks.

Figure 2.12 presents the sequence of steps needed to identify and select tracks in the HLT1 which has to be able to process events at a frequency of 30 MHz. Data from the detectors involved must be converted from their internal representation (the decoding step), translating it to hits in three dimensions in the detector. Then it is necessary to find the patterns of hits characteristics to tracks crossing each sub-detector. For example in the VELO, which is outside of the LHCb magnet field, those tracks are straight lines and Figure 2.13 illustrates an expected

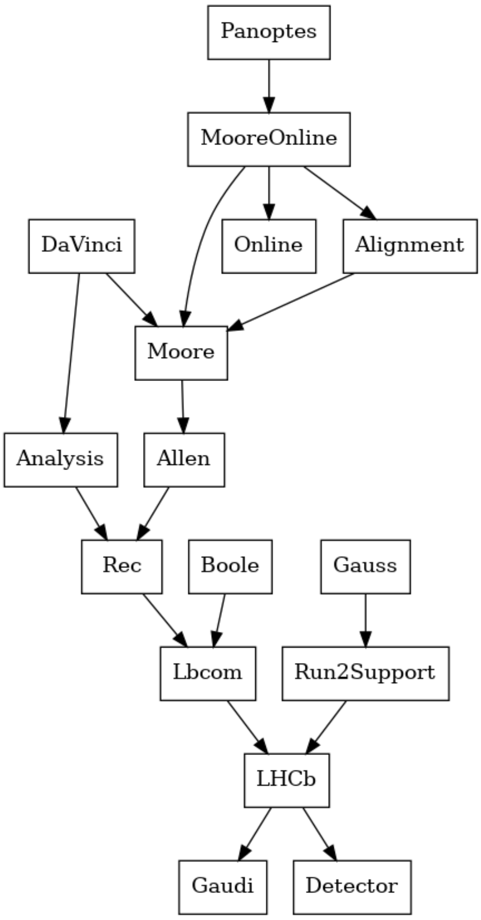


Figure 2.11: Dependencies between LHCb software applications in July 2023. Applications described in Table 2.1.

hit layout.

Performing this operation with the speed required in the trigger is complex, and different methods can be used. They can be separated in local and global methods. Local methods typically look for track *seeds* and complete them with other hits to create track candidates. This is the approach used in the *search-by-triplet* algorithm used in the LHCb HLT1 [52]. Global methods group together the hits based on their properties or projections. The Hough transform [53], which is used in LHCb to match upstream tracks (c.f. Figure 2.14) with SciFi hits, is one of such methods.

Of course, it is necessary to match the hits from all the sub-detectors, taking into account the bending of the trajectories by the LHCb magnet. Figure 2.14 presents the categorisation of tracks identified in the LHCb tracking system. Long tracks are the ones for which information is known most precisely as they are built from hits before and after the magnet and their momentum can be evaluated based on their deflection.

Once the tracks have been identified, a fitting stage is necessary to evaluate their parameters (x, y position at the point closest to beam, slope and momentum) and the uncertainty on those parameters. The classic Kalman algorithm [54] is used for this purpose.

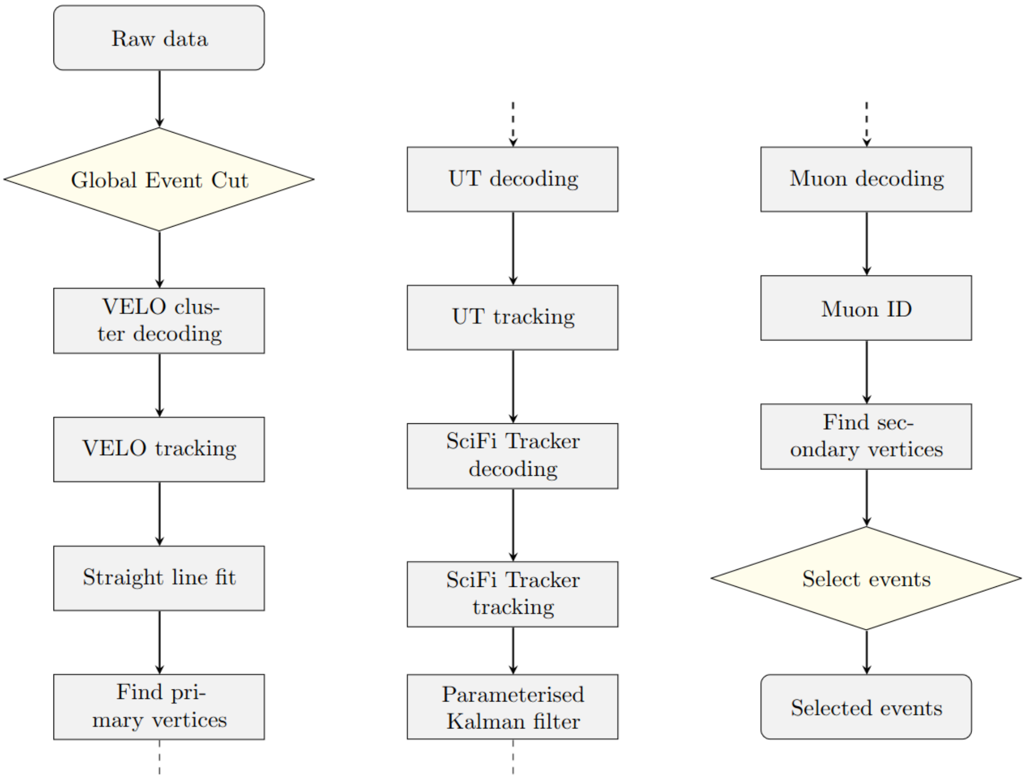


Figure 2.12: HLT1 Selection sequence for Run 3.

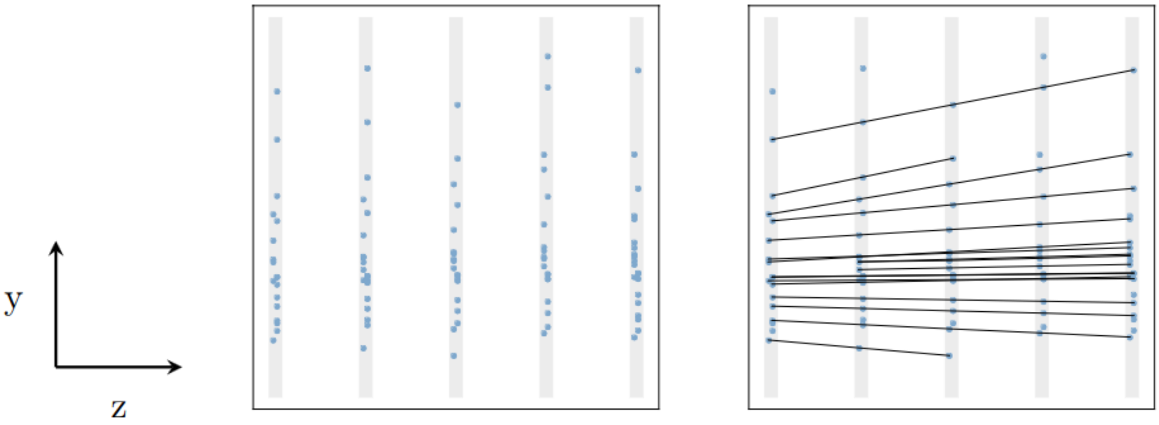


Figure 2.13: (Left) A minimal track reconstruction instance projected in 2D, consisting in a set of hits with position information. (Right) The actual particle trajectories sought when doing track reconstruction, from Ref. [52].

It is also necessary to identify the precise location of the primary interaction point, known as the primary vertex, which is done using the VELO tracks. Once the full tracking has been performed, it is possible to identify displaced vertices, i.e. the decay location of particles whose daughter particles were actually detected. This is crucial in LHCb as particles with b and c quarks decaying through the weak force live long enough and have enough momentum such that their decay points appear as displaced vertices in the detector.

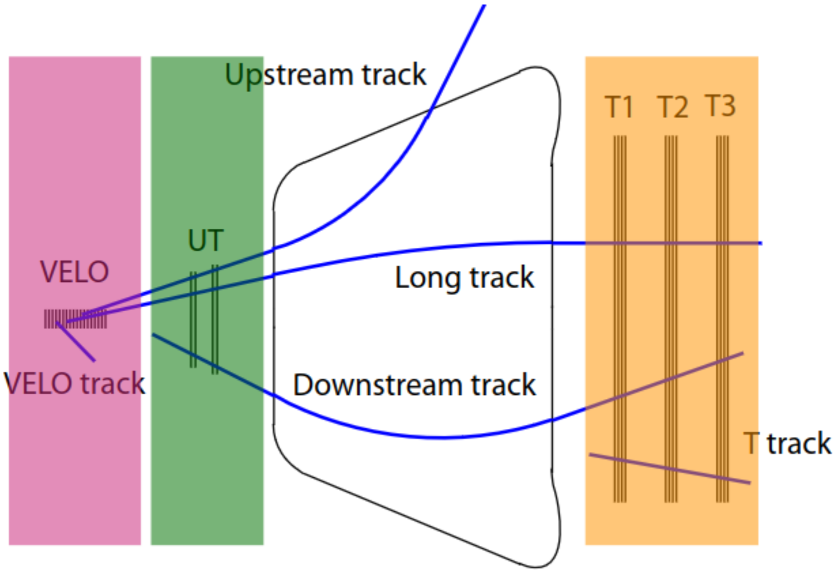


Figure 2.14: Type of tracks in the Run 3 LHCb detector (similar classification holds for Run 1 and Run 2).

Many different steps are therefore needed to reconstruct LHCb events, and the complexity of this sequence increases the difficulty in optimising the software.

Particle identification information

Different particles types leave different signatures in the sub-detectors mentioned in previous sections, as shown in Figure 2.15.

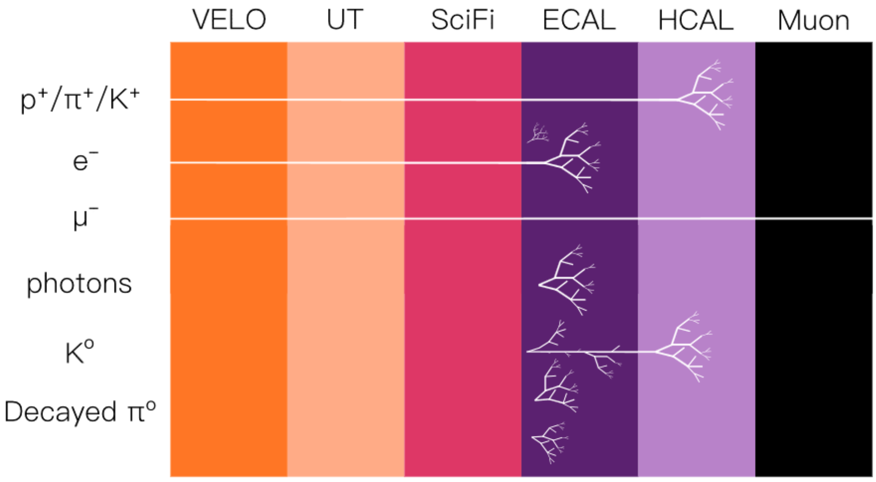


Figure 2.15: Signature left in LHCb for various species of particle, figure from [55]

Combining information from all subdetectors it is possible to distinguish between the various particles that are produced within LHCb. For neutral particles, such as photons and π^0, the identification is conceptually simple, as the energy deposit in the ECAL have no associated tracks or signal in the HCAL. For charged particles instead the PID is inferred from the combination of the momentum measurement and the information from the RICH, ECAL and the muon detectors. Two methods are used in LHCb (and the associated data used in this analysis): the first one uses the *combined differential log likelihood* (DLL) i.e. the combined likelihood difference between the kaon, proton, muon and electron hypotheses and the pion one, as the charged pions are the particles most commonly produced at LHCb. The other approach uses the output of neural networks trained to identify each particle species based on a wide variety of tracking and PID detector information. The performance of these tools is measured by means of data-driven calibration techniques as detailed in Ref. [56].

Filtering and Indexing

Once the data is reconstructed and the particle identification information is available, events are first filtered online according to a number of criteria defined a priori based on simulation studies. HLT1 performs a fast reconstruction and pre-selection using a small number of so-called *HLT1 lines* developed by specialists with the goal to keep the interesting physics while being fast enough to be used in the first level trigger and reduce the background a level that satisfies the output rate requirement (1 MHz).

Once the study of the calibration and alignment on the data collected by HLT1 has been performed and updated alignment constants are available, the HLT2 runs on the data, performing a detailed reconstruction and filtering the data according to a large number (in the order of 1000) of *HLT2 lines*, specific of each analysis, developed by the analysts from the various LHCb physics working groups. These lines are developed using a simplified configuration language that allows analysts to declare which decays they want to identify, and to place cuts on the various parameters of the particles in the decay. While HLT1 lines are developed in C++ and tuned by a small number of developers and physicists with software experience (in the order of 10 or 20), HLT2 lines are configured by a large number of members of the physics working groups using a selection framework prepared for this purpose, as described in Ref. [57].

The output of HLT2 is transferred to persistent storage and then filtered, skimmed and split into different files grouped into different *Streams* according to the type of physics involved, a step called *Stripping* in Run 1 and Run 2 and *Sprucing* in Run 3 [40]. In Run 3, the *Sprucing lines* are the offline equivalent of the HLT2 lines, as Sprucing shares the same software framework and application as the HLT2 trigger, namely the Moore application. Furthermore, the same algorithms and tools are shared between the trigger, Sprucing and the offline analysis software project DaVinci.

Simulating the detector

Simulation is crucial for many aspects of the experiment: at design stage, it is fundamental to optimise the sub-detectors and study their performance, as well as to develop the reconstruction and particle identifications tools, to develop the trigger and Stripping/Sprucing lines, and

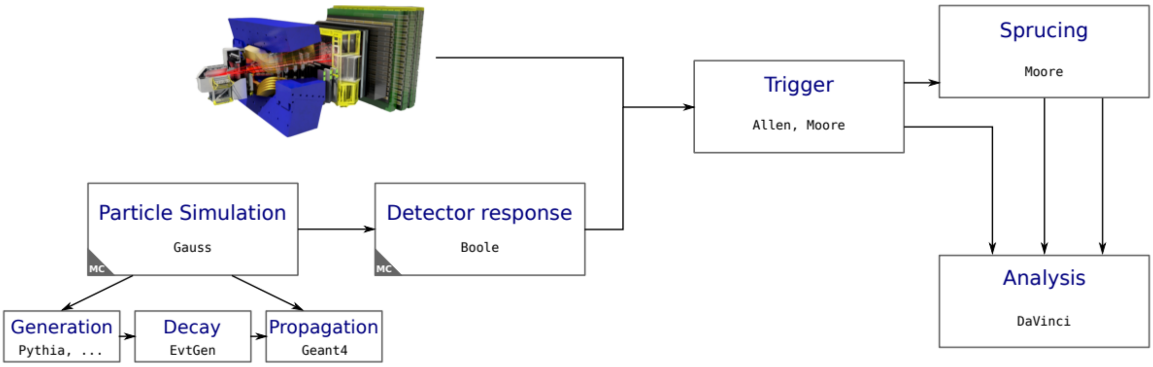


Figure 2.16: LHCb data processing.

finally to assess the physics reach of the experiment. During commissioning, simulation is useful to compare with data and to spot possible malfunctions or biases. For most of the analyses simulation is precious to define the analysis strategy, understand the background sources, model the data and finalise the measurements. Simulation in HEP is split in two phases, the event generation which reproduces the primary collisions, the elementary particles that are produced, their hadronisation and subsequent decays, and the transport phase which deals with simulating the propagation of the particles though the detector, as well as their interactions with the materials they encounter. This is a generic need across the HEP field, and common applications are available. LHCb primarily uses Pythia [16] and EvtGen [58] for event generation and decay, and the simulation framework Geant4 [47] for the propagation through the detector. These two steps are embedded into the Gauss application. Figure 2.16 shows the processing done for simulated events: after generation and propagation, the detector response for the simulated particles has to be emulated before using the same reconstruction chain as for real data. This is done by the Boole application.

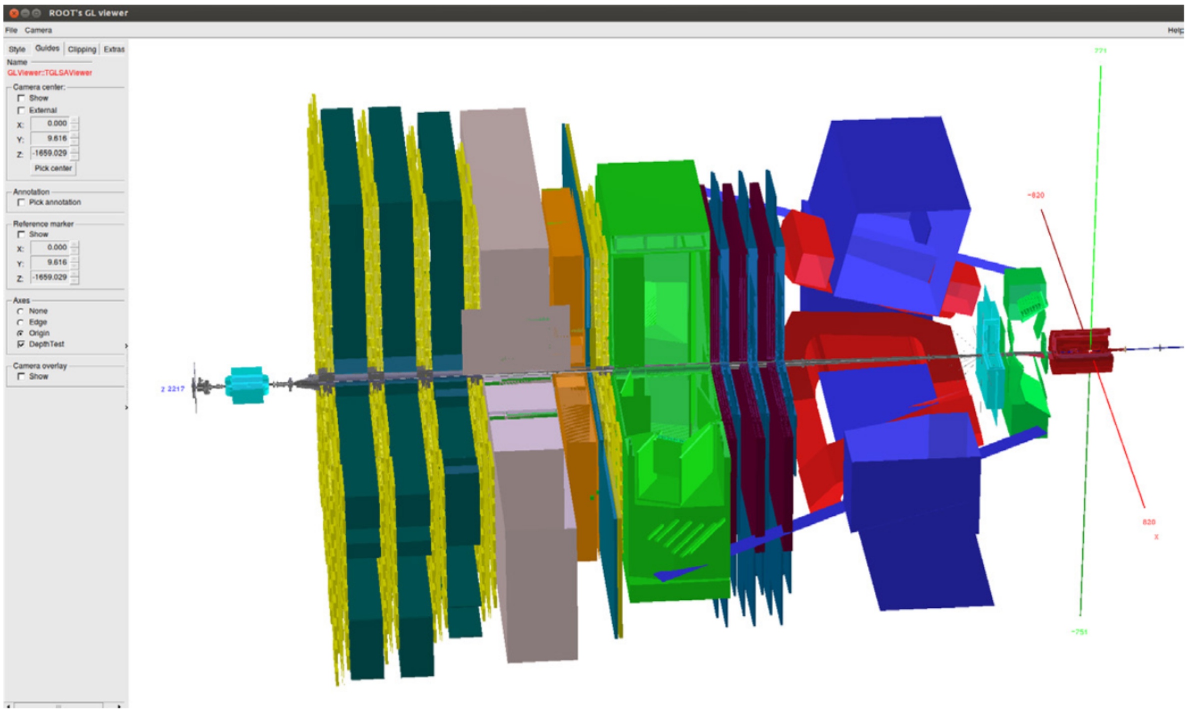
In order to simulate events, it is necessary to have a full and detailed model of the detector, and to be able to set the conditions matching the real-world detector. This is explained in the next section.

2.4.2 Detector description and conditions

Having an accurate description of the detector is mandatory to be able to simulate it. It is also needed when evaluating the quantity of material on the trajectory of particles, and therefore the influence of multiple scattering. Until Run 2, this was done using the LHCb specific *DetDesc* tool packaged within the Gaudi framework, for which the geometry and event visualisation tools are no longer supported. In Run 3, this was therefore replaced by the *DD4hep* framework [46], a tool developed for HEP experiments in the context of the AIDA 2020 project [59]. It provides functionality roughly equivalent to DetDesc but also natively integrates with tools in wide use such as ROOT and Geant4. The migration was described in Ref. [60], where the *Detector* project was setup as a way to access the geometry of the detector (see Figure 2.17). The proponent of this thesis was a key developer in this transition and presented it in Ref. [60].

In order to reconstruct and process raw events recorded at the LHCb experiment, it is necessary to have external context information about the detector at the time: the exact value of the current in the magnet, the exact value of the high voltage or gains used in a specific detector, a list of channels that are not operational and therefore have to be ignored, and so on. In High Energy Physics, such non-event specific data is called *condition data* and needs to be available for any processing to be done. Each event is associated by the online system to a specific *run*, and it is possible to look up the conditions based on the run number from a file repository versioned using the Git source code management system and the *GitCondDB* library [61]. DD4hep is also used by LHCb software to access condition data.

The proponent of this thesis integrated the *DD4hep* geometry within the LHCb software stack, and structured the project to support the LHCb description for Run 3, 4 and 5.



Name
GLViewer::TGLSAViewer

Camera center:
□ Show
□ External
X: 0.000
Y: 9.616
Z: -1659.029

Annotation
□ Pick annotation

Reference marker
□ Show
X: 0.000
Y: 9.616
Z: -1659.029

Axes
○ None
○ Edge
● Origin
☑ DepthTest

Camera overlay
□ Show

Figure 2.17: Representation of the LHCb detector with the DD4hep framework.

One important aspect in the Detector description is that in order to study recorded data from any data period, it must be possible to simulate LHCb as it was then. When using the Gaudi DetDesc description, the description consisted of files which could be versioned using the condition database. The issue is more complex with the *DD4hep* description which consist of text files and C++ code, and where versioning is done by introducing a directory structure for the text files, and simple versioning of the C++ by method name. To ensure that no undesired changes affect the geometry, and as the geometry is defined as a tree of volumes, the proponent of this thesis introduced a checksum that summarised the placement of all volumes in the tree. A similar functionality was later integrated in *DD4hep* itself.

2.4.3 Software optimisation efforts

LHCb has long sought to optimise its software to reduce its resource usage, both from the point of view of CPU and storage use. A major drive behind this effort is the need to filter the data in the trigger to make sure the appropriate events are selected, leading to many efforts to specialise the code to the hardware available, as shown by the number of related papers [62, 63, 64, 65]. Using the hardware most adapted to the task is crucial hence efforts to port the LHCb software to the ARM architecture [66, 67], or the Allen project [39] that implements the trigger on GPGPU, and has been chosen for Run 3 (see Figure 2.7). Of course, optimising resource usage is also crucial during offline processing, and significant efforts are made to adapt to the resources available on the WLCG. These efforts tend to make critical portions of the code (e.g. the data structures and algorithms used by HLT1) more complex and harder to maintain, putting the emphasis on quality assurance systems that can ensure that the code continues to fulfil its role with adequate performance. They have also shown that having one version of the code that is efficient on all architectures is extremely difficult to do, leading to some algorithms having several implementations depending on the platform used, increasing the maintenance complexity. With fast evolving technologies and experiments with a lifetime of tens of years, this has of course an impact on the long-term preservation of the code base and workflows, that will be explored in Chapter 3.

2.5 LHCb Data organisation

LHCb stores its RAW data in the LHCb-specific MDF file format [68]. All files processed offline and simulation output use the ROOT file format [45], which is very generic and allows saving C++ objects to file. There are several types of files created offline, the ones containing the full reconstructed events are called DST files (for *Data Summary Tape*, a historical name used in HEP). Micro DST files contain only the physics objects matched by the stripping or sprucing lines and are therefore more efficient from a disk space point of view.

Data analysts access the output of the stripping (sprucing) processing, which organises the data in a number of streams, depending on the type on physics events retained, and how much of the event is retained. For example Table 2.2 shows streams available for Run 1 and Run 2 data.

Each stream contains the events selected by a number of *stripping lines* that are specified by the stripping configuration used. The stripping project [69] gathers all configurations, listing the streams and the stripping lines associated. For example, the stripping *stripping34r0p2* used to process 2018 data counted a total of 659 different lines split in 7 streams (Bhadron, Semileptonic, BhadronCompleteEvent, Leptonic, Charm, Dimuon, CharmCompleteEvent). Stripping lines specify all the criteria required to decide whether an event matches the desired physics selection (for example which decay decay should be matched, the energy threshold on the particles and so on).

Stream name	Physics content
BHADRON.MDST	Micro-DST with B decays into hadronic final states
BHADRONCOMPLETEEVENT.DST	B decays into hadronic final states, full events
CALIBRATION.DST	B decays into hadronic final states used for calibration purposes
CHARM.MDST	Micro-DST with charm decays into hadronic final states
CHARMCOMPLETEEVENT.DST	Charm decays into hadronic final states
DIMUON.DST	Decays to two muons final states
EW.DST	Selections aimed at electroweak physics studies
LEPTONIC.MDST	B decays into leptonic final states
MINIBIAS.DST	Minimum bias events
PID.MDST	Events to calibrate the particle identification tools
RADIATIVE.DST	B and C decays with photons in the final states
SEMILEPTONIC.DST	B decays into semi-leptonic final states

Table 2.2: Available streams across the Run 1 and Run 2 data.

2.6 Worldwide LHC Computing Grid (WLCG) and offline resource use

The large amount of computational resources required for LHC data processing and storage caused the institutes and agencies funding the experiments to pool their computing resources in order to optimise their use. The resources are therefore available as part of the Worldwide LHC Computing Grid (WLCG), a global collaboration of around 170 computing centres in more than 40 countries, linking up national and international grid infrastructures. Common interfaces are used, for example to authenticate and authorise users, thereby allowing sites to provide computing infrastructure to one or more of the LHC experiments. The WLCG is organised in three tiers:

- The **Tier 0** at CERN where LHC data is recorded.
- The **Tier 1** consists of large national computing centres providing computing and storage resources (both disk and tape). Seven such sites currently provide resources to LHCb.
- The **Tier 2** consists of smaller computing centres at universities or scientific institutions, providing computing resources and sometimes storage (but in more limited quantity than Tier 1s).

LHCb on the WLCG uses the DIRAC middleware [70] to distribute the LHCb computing workload across the grid, as well as to manage the data stored at the different sites. Across the WLCG LHCb customarily uses an average of 100,000 computer cores to process its data, while the total reaches 1,400,000 cores for all four LHC experiments in 2022, as shown by Figure 2.18. As most of the LHCb recorded data processing is done online (by HLT1 and HLT2), simulations of the detector make up the vast majority of the CPU work on the grid.

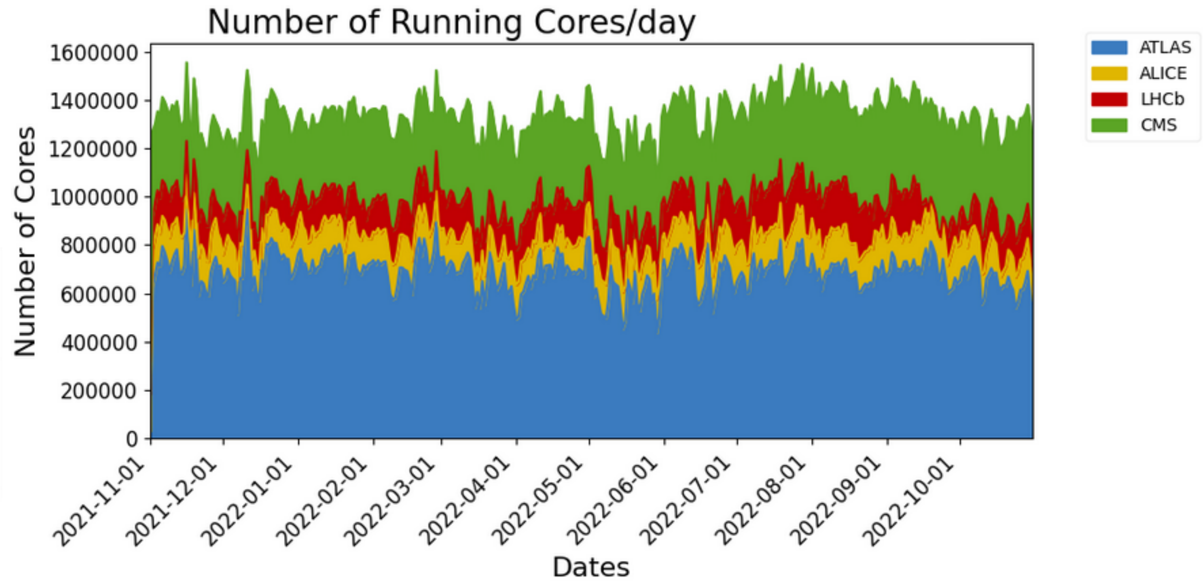


Figure 2.18: Number of computer cores used in 2022 on the WLCG.

From its inception LHCb stored a total of around 80 PB of data on tape, either at CERN (Tier 0) or in the Tier 1 sites.

2.7 Conclusion

LHCb is a unique detector dedicated to the study of the outcome of particle collisions in the forward region at the LHC. The excellence of its tracking and particle identification systems have been key to its physics program in LHC Run 1 and Run 2, and further improvements were made for Run 3. A matching data acquisition and processing capability is crucial, and software is an important part of it: a number of programs counting millions of lines of code are required to prepare the data for analysis. Running this software also requires massive resources, both at LHC point 8 where LHCb is located, but also on the WLCG where LHCb customarily uses an average of 100 000 computer cores to process its data. The developed software is needed not just at data taking time, but also to exploit the recorded and simulated events at a later stage. It is therefore crucial to make sure it can be preserved for later use and Chapter 3 describes what this entails.

Chapter 3

Analysis Preservation and Reproducibility at LHCb

3.1 Data Analysis at LHCb

As mentioned in Chapter 2, the LHCb data processing illustrated by Figure 2.16 can be split in:

- a centralised recording and processing phase. Its output is dubbed *production data.*
- an analysis phase which performs measurements based on production data.

The distinction is not related to the complexity of the processing, but to the generality of the output data and to the tools and processes in place to produce and validate it. Centralised productions perform all the common processing required before the data can be analysed: the reconstruction of the events, their filtering and indexing. They require the use of the LHCb specific software, and are managed, organised and prioritised by the software and computing projects. They require large amounts of CPU power and storage and are therefore carefully planned.

LHCb data analyses start with the extraction of relevant data from the production files (stored in LHCb specific DST format) by individual analysts. This data is commonly stored as a set of *ntuples*, i.e. files usually in ROOT[45] format containing a table of values for each event or candidate selected. An ntuple contains only base types, and therefore does not require experiment specific tools for processing.

This step was traditionally done in LHCb using Ganga [71], a tool capable of launching user jobs on the Worldwide LHC Computing Grid. While this method is practical as it gives a large freedom to users, it comes with many disadvantages:

- users have to monitor the jobs themselves, and make sure that the processing on the WLCG occurred correctly.
- As there is no central quality control, it is up to the users to check that the code is correct. As this is not ensured, massive job failures can lead to an inefficient use of the WLCG

(between October 2022 and October 2023 the failure rate of Monte Carlo simulation and reconstruction jobs is below 2%, whereas it was around 11% for user jobs).

- The file produced are private, with no central registration nor provenance tracking. This often leads to multiple copies being kept, and the provenance information being lost. The data produced within one working group is not *Findable* by members of another working group, thus preventing the sharing.
- Jobs are started independently by different users and it is therefore impossible to coalesce them to improve processing efficiency.

LHCb investigated other ways to perform this task, leading to the introduction of Analysis Productions a centralised and fully tracked way for users to extract data from the LHCb dataset, further explained in Section 3.3.1.

Whether physicists use Ganga or Analysis Productions to extract their ntuples, the next step is to measure physics quantities based on the extracted data. The quantity of data extracted, the complexity of the processing and the amount of CPU cycles needed are highly dependent on the analysis being performed. They involve different software tools, sometimes require dedicated clusters, sometimes the processing can be done on the WLCG. This phase is inherently complex, and requires many different investigations to evaluate quantities from data, and their associated uncertainties. It is therefore critical for analysts to have freedom to use the most appropriate tools at disposal for the best results to be attained: there is a lot of interest in machine learning tools for example. For this reason, it would be ill-advised to enforce a strict software development procedure that limits the possibilities, but good practices have nevertheless to be encouraged.

3.1.1 Publication procedure

Within High Energy Physics collaborations, a large amount of effort is dedicated to ensuring the quality of the publications signed on behalf of the collaboration. Analyses undertaken at LHCb are split in different working groups (WG) depending on the physics subjects (e.g. spectroscopy, CP violation, semileptonic decays, charm decays, rare decays...) or the type of measurements. Within each working group, analyses are discussed throughout their development. For each analysis, an internal *Analysis Note* (ANA note) is written, much longer than the final paper but with the goal of containing all the details that would allow repeating the measurement. It is carefully reviewed by the working group and a review committee.

Once deemed ready for publication, the paper (and the accompanying ANA note) follow a formal review procedure involving referees internal to the collaboration, multiple institutes from the collaboration and a general presentation before approval. The approved paper can then be submitted for publication and eventually published in relevant journals, following peer review. Traditionally, no emphasis was put on preserving the code used, and the ntuples produced, the reason being that keeping track of all the steps in an analysis is indeed complex.

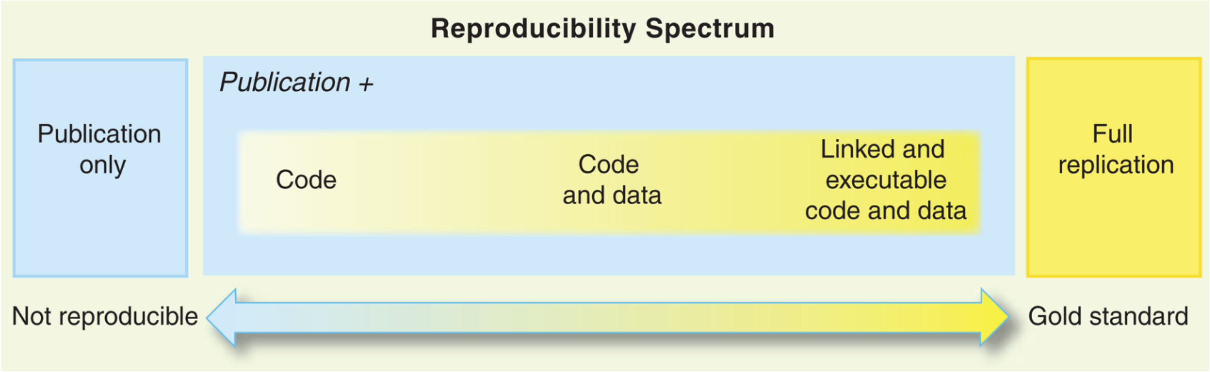


Figure 3.1: Reproducibility spectrum according to R. Peng [74]

3.1.2 Preservation and reproducibility aspects

Being able to reproduce an experimental result is critical to the scientific method; it ensures that generally applicable knowledge is derived. Other teams throughout the world should be able to reach the same conclusion from similar experiments. Across the various fields of science, guaranteeing reproducibility is however difficult as shown by researchers trying to reproduce results based on the published information: investigations published in Ref. [72] or [73] led to the realisation that research was facing a *reproducibility crisis* and that improvements to methodologies had to be done.

The first pre-requisite is the repeatability of the experiment itself, meaning that researchers should be able to re-run the same exact experiment and find consistent results. This is routinely done on LHC experiments by comparing measurements of physical quantities for different data taking periods.

Reproducing also means that other teams should be able to assemble a machine that produces consistent results. For this purpose, enough information should be available on the design of the experiment, and the description of its environment that allows interpreting the results. As HEP experiments use bespoke detectors, this means giving full access to the data and description of the hardware, but also the software used to process the data. Indeed, with the increase of reliance on software, *reproducibility* also took a narrower meaning which we address here: with a set of input data, can we re-process and get to identical figures as those published? The gold standard, as defined by Peng [74] and presented in Figure 3.1, would be to have all the code and access to all the data used for the paper.

With HEP experiments producing petabytes of data per year, and requiring significant CPU power to process them, reaching this goal is not simple and compromises have to be made: for example reduced ntuples that allow deriving the quoted numbers are recorded with published analyses, alongside the recipe used to produce the ntuples themselves (c.f. Section 3.1.3).

Full computational reproducibility is possible when all resources used for the papers are available and all workflows and tools are available with the associated environments. This may however not even be enough: a change of computer architecture, for example moving from 32-

bit to 64-bit CPUs, or moving from Intel to ARM CPUs may produce equivalent results but not bit-by-bit identical (c.f. [75], [76] and for details [77]). In that case, we can have at most statistical reproducibility, which in itself is not a problem if computational errors are checked and the results compared with reasonable precision.

The need for FAIR research

The need for reproducibility of the results has implications on the way the research is conducted and on the material released alongside the research papers. The Nature journal requires that [78]:

> *"authors are required to make materials, data, code, and associated protocols promptly available to readers without undue qualifications."*

This need for openness in research is also driven by governments and funding agencies, as a way to ensure good use of the public investment in science. Enabling knowledge sharing and efficient reuse of research data is a complex task, so various stakeholders met in 2015 to gather best practices and provide recommendations on good data management and stewardship, a prerequisite for high-quality publications. The outcome of this effort was published in 2016 and laid out *The FAIR Guiding Principles for scientific data management and stewardship* [7], in the journal Scientific Data. FAIR stands for Findable, Accessible, Interoperable and Reusable and the publications details the meaning and implications of those statements on scientific data management. The FAIR principles were quickly adopted by the community. For example the European Union Horizon 2020 programmes started providing guidelines on FAIR Data Management[79] as early as 2016. The European Commission requested a cost-benefit analysis for FAIR research data [80] in 2018 that concluded that the case for implementing them was "overwhelming". FAIR principles are also at the core of the European Open Science Cloud [81], a virtual environment federating research data and services to support EU science, and are even part of their mandate:

> *"EOSC ultimately aims to develop a Web of FAIR Data and services for science in Europe upon which a wide range of value-added services can be built"*

The principles state that the data should be:

- **Findable**: metadata and data should be easy to find for both humans and computers. Machine-readable metadata are critical for automatic discovery of datasets.
- **Accessible**: once the data found, the user should be able to find out how it can be accessed (even if authentication and authorisation are needed).
- **Interoperable**: it should be possible to integrate the data with applications or workflows.
- **Reusable**: the data and its context should be well-described enough so that they can be replicated and/or combined in different settings.

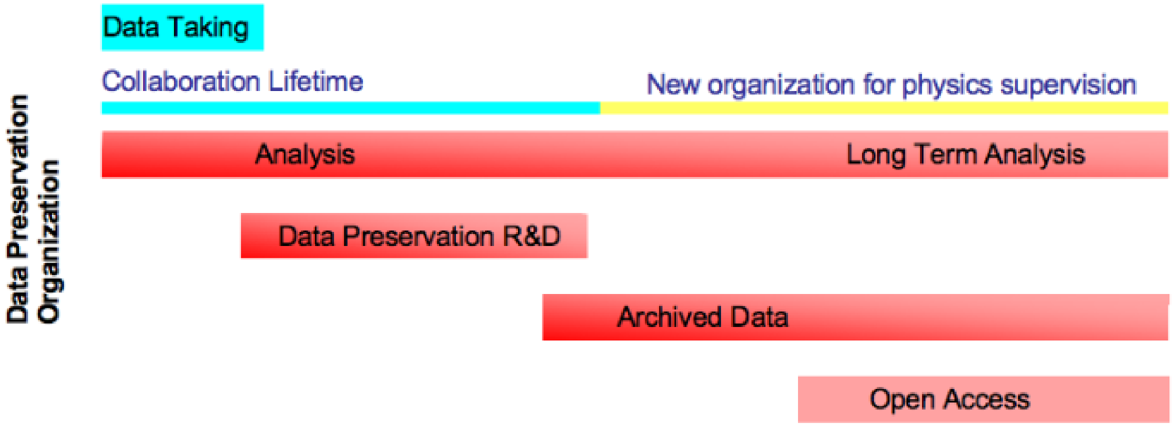


Figure 3.2: Data preservation organisation according to DPHEP [87]

The *Reusable* aspect has deep implications, notably that it implies keeping track of the Provenance of all the data items, i.e. the source data and all processing steps that it went through. This is an area where this thesis proposes tools that can help throughout the analysis of LHCb data.

Scientific Data Preservation

All scientific domains are affected by reproducibility and preservation issues: while some of the authors of the FAIR principles came from life sciences or bio-technologies, it is clear that the issues underlined by *The FAIR guiding principles* [7] are general. In fact the paper also mentions existing astronomical databases such as SIMBAD [82] or NASA's Space Physics Data Facility [83]. Digital data preservation always was a concern in space sciences: for example, the Consultative Committee for Space Data Systems, a forum for national space agencies with the mission to develop common data handling standards, started working the Open Archival Information System (OAIS) [84] in 1990. It was standardised in 2002 and updated in 2012 [85]. It is a conceptual model dedicated to the management, the archival and long term preservation of digital documents, which defines the tasks and roles to be performed to put in place a digital archive. Furthermore, Findability and Interoperability are crucial for multi-messenger astronomy [86] which aims to correlate information from different detectors sensitive to different probes (photons, cosmic rays, gravitational waves).

Data collected at facilities such as the Large Hadron Colliders is unique, and therefore has to be curated appropriately to be sure it is used to its full potential. This was acknowledged at the start of LHC data taking and a working group was established, to be able to find sustainable ways to preserve the produced artefacts. The Data Preservation In High Energy Physics (DPHEP)[87] study group was created in 2009 and stressed the importance of planning for preservation from the start, describing the experimental data lifecycle as in Figure 3.2.

Considering that HEP collaborations last for decades, this highlights the fact that it is necessary to record every aspect of the data, and to preserve it in self-describing formats as early as possible, as experts in the various aspects of the experiment will come and go during such periods.

In short, the preservation of the results has to be planned from the start of the experiment.

The study group also defined a classification of research data, which was adopted by the CERN Open Data Policy [4]:

- Level 1 data corresponds to additional information, related to the paper published. This can be for example metadata related to the running conditions, ntuples used to provide the plots. They should be made available in Open Access through suitable community platforms.
- Level 2 data is experimental data preserved in a simplified format. This is useful for outreach and training exercises but an analysis cannot be realistically be reproduced this way. For example, all LHC experiments have been releasing level 2 data for the International Particle Physics masterclasses, in the case of LHCb with the D^0 lifetime measurement exercise [88].
- Level 3 data covers reconstructed data and simulation data as well as the analysis software needed to allow a full scientific analysis. The whole or part of the dataset is made available after a certain period.
- Level 4 data includes raw data and provides access to the full potential of the experimental data. Access to those data is restricted even within collaborations as processing them requires major resources and operational overhead. They are stored for long-term preservation.

This classification offered a common framework to define the efforts to be undertaken. The report also insisted on the common requirements between all experiments and advocated the development of common tools to support the preservation policies. A further report in 2012 [89] proposed to formalise the data preservation efforts in HEP. This organisation continues to report regularly on the data and analysis preservation efforts within the HEP community, for each experiment, as well as on the common tools.

The last report was published in 2022 [90] and shows the advancement since 2012: many tools are now available to enact the CERN Open data Policy [4], and major efforts have been undertaken by all experiments. Furthermore, the democratisation of continuous integration tools in the software development world, close to data analysis in term of tools, if not in spirit, provided new ways to improve the reproducibility of analyses. This is shown by several theses in LHCb such as [91] or [55], but is also a concern for other High Energy Physics experiments, such as ATLAS [92], or CMS [93].

3.1.3 LHCb Analysis Preservation roadmap

As part of the DPHEP efforts, LHCb reviewed its tools and procedures to improve them. While full reproducibility is difficult to attain, a number of simpler requirements can greatly improve the situation. Therefore as of December 2017, LHCb adopted its Analysis Preservation Roadmap[94], some recommendations and best practices, as well as a minimal set of mandatory analysis preservation practices added to the analysis review process. This involved steps necessary for the integration with the CERN Analysis Preservation portal (CAP, c.f. section 3.3.2):

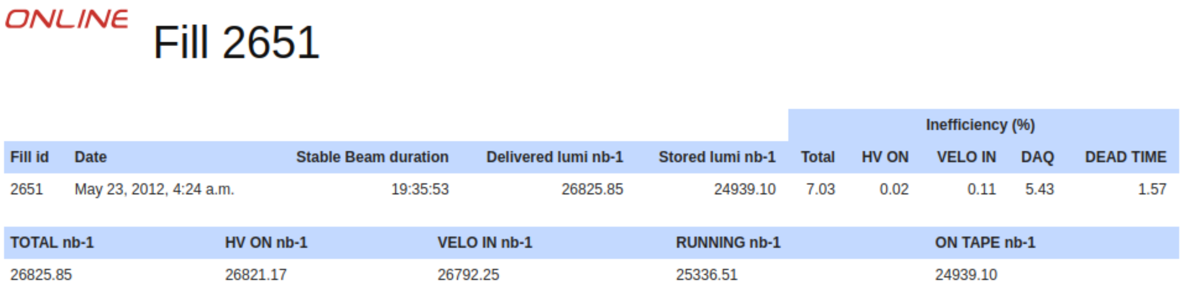
- Analysis code repositories: keeping track of the repositories is important, even if further requirements are not imposed, in order to leave freedom to analysts to organise their work.
- Ntuple storage: for each analysis, the final ntuple from which the paper plots and diagrams are produced should be kept.
- Analysis automation: The use of workflows should be encouraged. The use of the Snakemake workflow engine is taught during the LHCb StarterKit[95, 96], as recommended by Stodden [97].
- Run-time environment preservation: analysts are encouraged to document the environment they use.

3.2 Reproducibility at the LHCb experiment

As described in Chapter 2, two distinct phases are present in the data processing: 1) a common processing phase, and 2), an analysis phase. They are performed by potentially different teams with different processes and tools. The reproducibility features of the result of each processing phase have to be investigated separately.

3.2.1 Preservation of production data and software

LHCb data recording history and online quality assurance



ONLINE

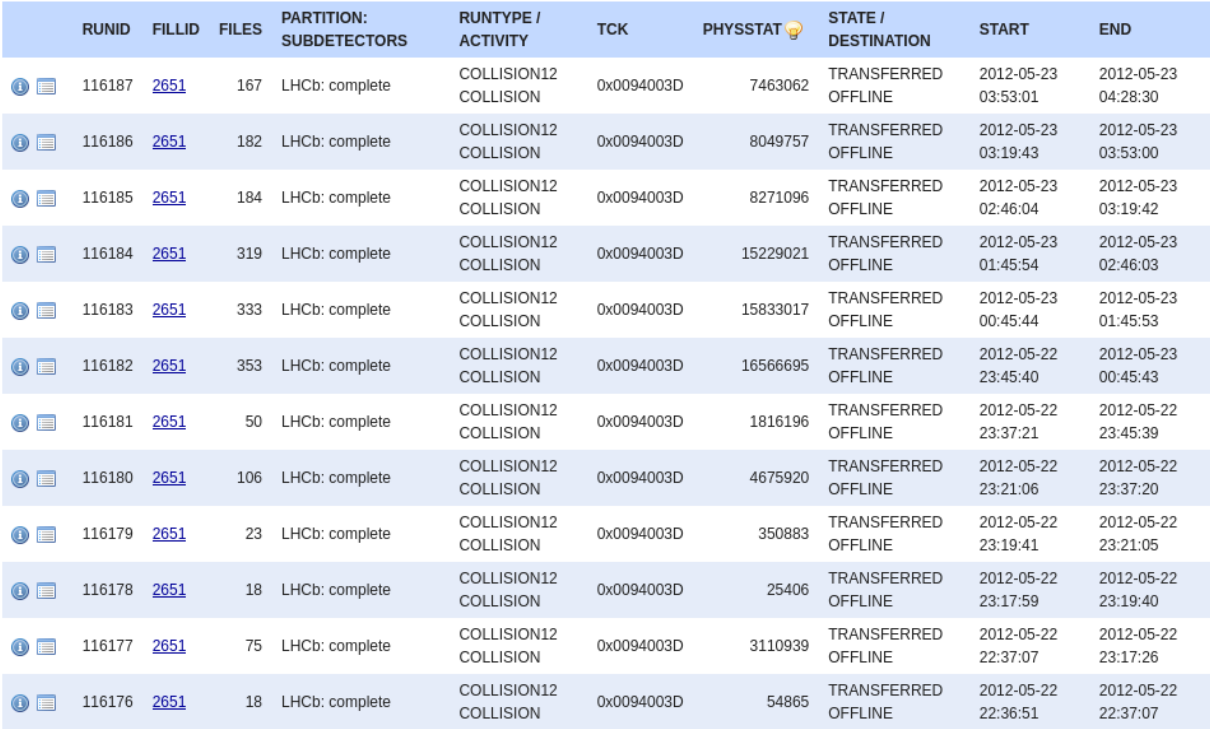
Fill 2651

					Inefficiency (%)				
Fill id	Date	Stable Beam duration	Delivered lumi nb-1	Stored lumi nb-1	Total	HV ON	VELO IN	DAQ	DEAD TIME
2651	May 23, 2012, 4:24 a.m.	19:35:53	26825.85	24939.10	7.03	0.02	0.11	5.43	1.57

TOTAL nb-1	HV ON nb-1	VELO IN nb-1	RUNNING nb-1	ON TAPE nb-1
26825.85	26821.17	26792.25	25336.51	24939.10

Figure 3.3: View of the information of fill 2651 in the Run Database.

The Run Database application[98] stores information about the data recorded by the LHCb detector, and its configuration at the time. It organises the data according to the LHC fills (see Section 1.2) during which the data was recorded, as shown by Figure 3.3. Each fill is further split in in a number of *runs* (of one hour of data taking maximum, see Figure 3.4) corresponding to one specific configuration of the online system.



	RUNID	FILLID	FILES	PARTITION: SUBDETECTORS	RUNTYPE / ACTIVITY	TCK	PHYSSTAT	STATE / DESTINATION	START	END
ⓘ ▤	116187	2651	167	LHCb: complete	COLLISION12 COLLISION	0x0094003D	7463062	TRANSFERRED OFFLINE	2012-05-23 03:53:01	2012-05-23 04:28:30
ⓘ ▤	116186	2651	182	LHCb: complete	COLLISION12 COLLISION	0x0094003D	8049757	TRANSFERRED OFFLINE	2012-05-23 03:19:43	2012-05-23 03:53:00
ⓘ ▤	116185	2651	184	LHCb: complete	COLLISION12 COLLISION	0x0094003D	8271096	TRANSFERRED OFFLINE	2012-05-23 02:46:04	2012-05-23 03:19:42
ⓘ ▤	116184	2651	319	LHCb: complete	COLLISION12 COLLISION	0x0094003D	15229021	TRANSFERRED OFFLINE	2012-05-23 01:45:54	2012-05-23 02:46:03
ⓘ ▤	116183	2651	333	LHCb: complete	COLLISION12 COLLISION	0x0094003D	15833017	TRANSFERRED OFFLINE	2012-05-23 00:45:44	2012-05-23 01:45:53
ⓘ ▤	116182	2651	353	LHCb: complete	COLLISION12 COLLISION	0x0094003D	16566695	TRANSFERRED OFFLINE	2012-05-22 23:45:40	2012-05-23 00:45:43
ⓘ ▤	116181	2651	50	LHCb: complete	COLLISION12 COLLISION	0x0094003D	1816196	TRANSFERRED OFFLINE	2012-05-22 23:37:21	2012-05-22 23:45:39
ⓘ ▤	116180	2651	106	LHCb: complete	COLLISION12 COLLISION	0x0094003D	4675920	TRANSFERRED OFFLINE	2012-05-22 23:21:06	2012-05-22 23:37:20
ⓘ ▤	116179	2651	23	LHCb: complete	COLLISION12 COLLISION	0x0094003D	350883	TRANSFERRED OFFLINE	2012-05-22 23:19:41	2012-05-22 23:21:05
ⓘ ▤	116178	2651	18	LHCb: complete	COLLISION12 COLLISION	0x0094003D	25406	TRANSFERRED OFFLINE	2012-05-22 23:17:59	2012-05-22 23:19:40
ⓘ ▤	116177	2651	75	LHCb: complete	COLLISION12 COLLISION	0x0094003D	3110939	TRANSFERRED OFFLINE	2012-05-22 22:37:07	2012-05-22 23:17:26
ⓘ ▤	116176	2651	18	LHCb: complete	COLLISION12 COLLISION	0x0094003D	54865	TRANSFERRED OFFLINE	2012-05-22 22:36:51	2012-05-22 22:37:07

Figure 3.4: Partial list of runs for fill 2651 in the Run Database.

Each data file recorded by the LHCb online system corresponds to a specific run, and therefore a specific configuration of the detector: the sub-detectors status and configuration, for example whether the VELO was open or closed. The trigger configuration, which is crucial to analyse the data including the software version, and all other parameters of the system needed to reproduce the results are included.

The LHCb run identifier is also the base granularity for the computation of alignment parameters and calibration constants, and is therefore in Run 3 the key to query the conditions database for all parameters (this was done by event time for Run 1 and Run 2, but was changed for Run 3 in order to simplify the management of the condition information).

The Run DB database information is also available in the bookkeeping database once the data files have been recorded and processed. Preserving the Run DB can however be useful as it allows querying the data by fill and by run.

In order to ensure the quality of the recorded data, the Monet [99] online data quality system is in place. It stores quality measures histograms for each run, and allows checking for issues and mis-configuration of the detector as soon as the data is recorded.

In summary, the Run database and the associated data quality measures stored in Monet are crucial to interpret and process the huge amounts of data recorded by the detector. These applications are deployed on infrastructure supported by CERN IT and are backed-up and preserved according to the standard procedures.

Physical Data preservation

As of 2023, LHCb kept around 80 PB of data on tape, and 45 PB on disk according to the report on resource usage [100] [1]. Managing such quantities of data is a major task and the policy to organise the storage was spelled in the LHCb Upgrade Computing TDR [35].

Among all the data:

- some is recorded directly by the detector and is therefore inestimable. This is traditionally called *raw data* or *custodial data* and is the output of the LHCb trigger system in the LHCb specific MDF format[68]. It contains the raw information recorded by the data acquisition system plus information from the trigger system to understand why the event was selected. In the case of the LHCb Turbo stream, the event is reconstructed online and the reconstructed information about the event is kept while the raw detector information is discarded, in order to save storage space on disk.
- some is the result of Monte Carlo simulation and can be regenerated (albeit with significant CPU use, as is takes in the order of a minute to simulate a LHCb event on one core of standard server in 2023). This is stored in ROOT format [45].
- some is derived from either raw or simulated data. This is the case of files produced by the filtering and indexing stages (stripping or sprucing).

Long-term data storage is ensured by the WLCG, at the Tier 0 (CERN) or at the Tier 1s. Curating such amounts of data (more than 500 PB for the CERN tape archive) is a major endeavour and the sites have dedicated teams to fulfil this task. CERN has a dedicated system called the CERN Tape Archive (CTA) [101], which uses a TFinity library from Spectra Logic that can hold up to 15'000 LTO Ultrium magnetic tapes (with 18 TB per cartridge for LTO-9 standard tapes).

According to recent studies [102], LTO tapes can last up to 30 years but random failures happen nonetheless. Therefore, for safety reasons, LHCb keeps two tape copies of all data that cannot be regenerated.

Furthermore, single sites can suffer catastrophic failures, such as the flood that occurred in the WLCG Bologna Tier 1 site in 2017 [103]. While all sites try to take all measures possible to avoid such problem and limit their impact when they occur, LHCb keeps one copy of the custodial data at CERN, and a copy distributed between all the LHCb Tier 1 sites.

It is worth noting that maintaining a tape system means reading the tape cartridges on a regular basis to ensure they are still readable. Furthermore, the capacity of tape cartridges increases at each generation of the tape technology, as show by Figure 3.6. It is therefore worth

[1] This is for Run 1 and Run 2 data; in Run 3 will add 120 PB on tape and 40 PB of disk per year of data taking.

Figure 3.5: Spectra Logic TFinity library at CERN

copying data from old tapes to new generation tape cartridges on a regular basis. This has the advantage of increasing the density of the existing tape libraries and also checks that the data is accessible, but it creates an operational burden. Dedicated tools are however developed to ease that process, such as repack [105] for CTA at CERN.

Preserving the data is therefore complex and onerous, and laboratories and institutes that are part of the WLCG invest significant resources in this task. They have been doing so for many years and have a significant experience developed over the decades.

LHCb Data Formats preservation

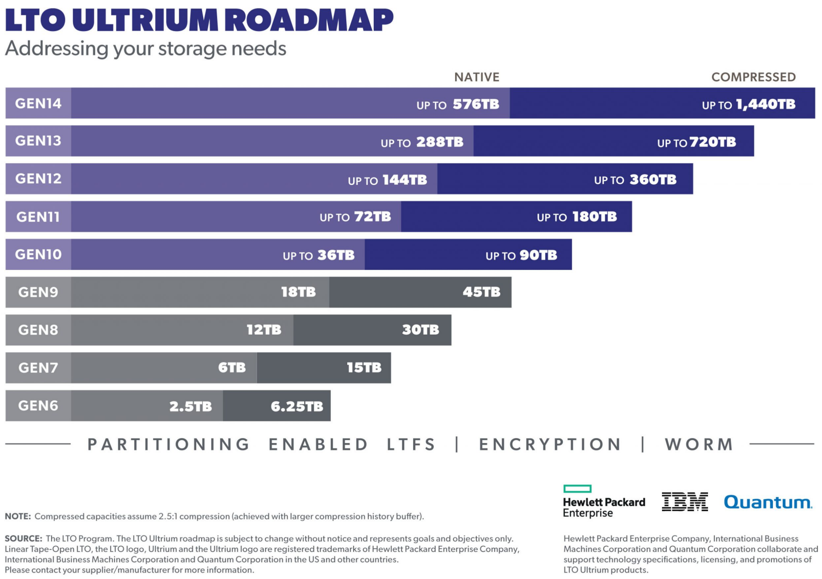
LHCb uses two different file formats for its data:

- The MDF file format [68] for the data recorded by the trigger,
- The ROOT file format [45] for all offline and simulation artefacts.

The choice of the MDF format for online use is due to simplicity: it consist of a series of raw *banks* with specified length where the output of the detectors, or any object can be written. The simplicity of the format [68] gives reassurances of the long term readability of the data files.

However, such format does not provide tools to write C++ complex objects and there is no way to systematically save them. The code to write them to file, and subsequently read them, must therefore be hand-crafted for each class, a long and error-prone operation; it must also of course be preserved in order to be able to read the data from the files in the future.

Offline and simulation tools use the ROOT file format, which is more complex but allows storing C++ objects easily. Making sure that ROOT files can be read in the future is a concern for all experiments, and the ROOT team guarantees backwards and forward compatibility for the ROOT Input/Output layer[106]: this means that new versions of ROOT should be able to read any files produced in the past.



NOTE: Compressed capacities assume 2.5:1 compression (achieved with larger compression history buffer).

SOURCE: The LTO Program. The LTO Ultrium roadmap is subject to change without notice and represents goals and objectives only. Linear Tape-Open LTO, the LTO logo, Ultrium and the Ultrium logo are registered trademarks of Hewlett Packard Enterprise Company, International Business Machines Corporation and Quantum Corporation in the US and other countries. Please contact your supplier/manufacturer for more information.

Hewlett Packard Enterprise Company, International Business Machines Corporation and Quantum Corporation collaborate and support technology specifications, licensing, and promotions of LTO Ultrium products.

Figure 3.6: LTO tape capacity roadmap according to the LTO program[104]

This does not alleviate all long term preservation worries however, as ROOT is a very generic container format, and LHCb needs to ensure the consistency of the objects it stores in the ROOT files. Furthermore, there are different types of files depending on the content: DST files contain the full reconstructed event with all related objects. As the storage is limited, many efforts have been made to reduce the event size and therefore allow storing many more events within the same storage budget. The LHCb computing TDR for Run 2 [107] introduces the notion of reduced DST to be used for data filtering. Micro-DST [108] files were invented during Run 2, inorder to contain only the part of the event of interest to a specific working group. Turbo Events, for which the raw data from the sub-detectors is not kept, contain just the interesting part of the events that are useful for analysis; analysts can even specify which particles they would like saved in the final file, with the notion of *selective persistency* (as described in Section 2.3 and in the LHCb Computing TDR for the Upgrade [35]).

A common aspect to all those files formats is that in order to read them, it is necessary to use LHCb applications which are therefore key to read LHCb data in the future. A sustainable way to keep running the LHCb software is therefore critical.

Versioning the processing chain

While preserving the data is of course the necessary base, the tools that are used to decode and process the data also have to be versioned for the base data formats to be usable in the long term. The following items are used for this purpose:

- **a source control system** for all developments: the source code of LHCb tools is managed using the git control system as part of the CERN instance of the GitLab code repository. This allows for collaborative software development, coordinating the introduction of new features and keeping track of developers interactions and comments.

- **A clear definition of the dependencies**: LHCb software is built with a strict definition of the software dependencies: each tool version depends on specific versions of its dependencies. In the case of internal dependencies, i.e. developed within LHCb, it is possible to find the related sources in GitLab. For external dependencies, LHCb software depends on a specific version of the LCG Software stack[109] which explicitly specifies the external tools used. The dependencies are Open source (with a few exceptions) so it is possible to find their own repositories and find the related code.

- **A definition of the binary platforms and compilation flags**: this is specified by a string, defined in a common way for all experiments using LCG releases [110], which specifies the target computer architecture for the build, the base operating system, the compiler used and the compilation flags. For example x86_64_v3-el9-gcc12-opt+g means:
  - the architecture of the processor involved is x86_64_v3. The code was compiled for the x86_64 architecture, with AVX2 extensions [111],
  - the base operating system is a RedHat Enterprise Linux version 9,
  - the compiler used is GNU GCC, version 12
  - the code was compiled in optimised mode, keeping the debug information (+g).

This allows identifying the platform and all the configuration that went into building the software, thus allowing a rebuild from scratch if necessary.

Versioning software configuration

The configurations used to run the LHCb software to process data are strictly defined to ensure reproducibility of the processing:

- the configuration of the LHCb trigger is defined by a *Trigger Configuration Key* (TCK) [112] [113] which defined exactly the processing done,

- offline filtering and indexing steps are defined and documented, in order for analysts to understand what events to find in which file [35, 18],

- LHCb uses the Gauss application[114] for Monte Carlo simulations. Great care is taken to preserve the configuration of the simulation, and the seeds for pseudo random numbers are set carefully to be able to reproduce and debug the results.

The LHCb distributed computing uses the DIRAC [70] middleware, for which each data production stage consists of interdependent steps where the outputs of initial steps serve as input to subsequent ones. Each step has to refer to a specific version of a LHCb application, it is therefore easy to track which software was used.

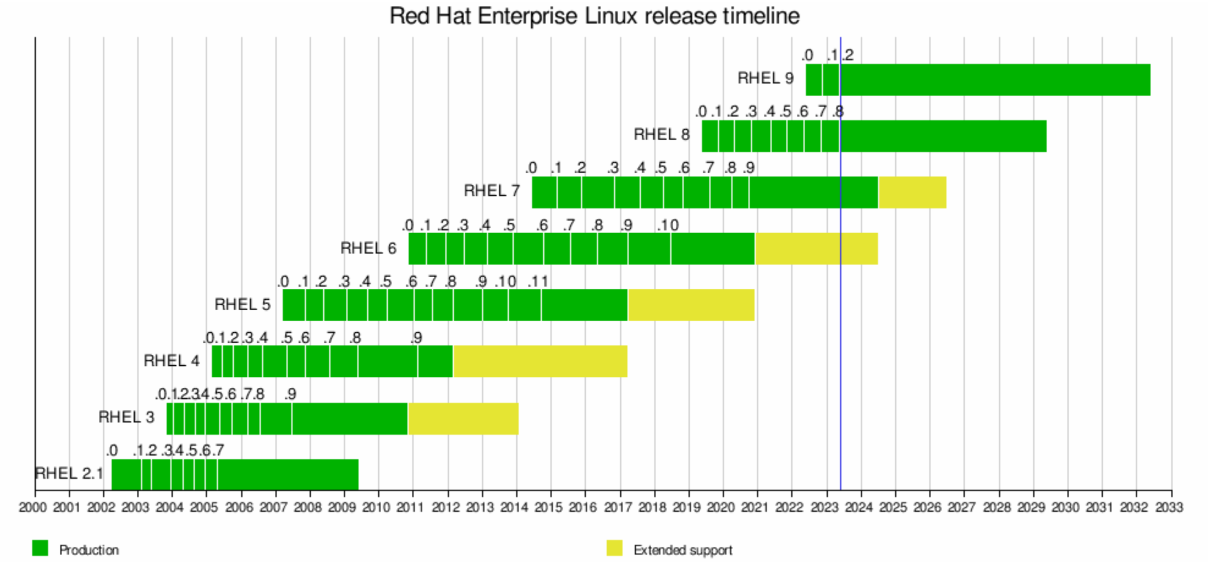


Figure 3.7: RedHat Linux version history according to Wikipedia[116].

Operating system evolution

Operating systems installed on personal computers or servers evolve quickly, to solve existing problems and provide new features to the users. This does not lead to a stable environment in which it is possible to run the same program forever. Figure 3.7 shows the history of the versions of the RedHat Enterprise Linux system, which is used as a base for the systems installed at LHCb and on the WLCG. Considering that the software development in LHCb started in the early 2000s, and that the experiment is still running now, it is clear that it has been necessary to update the base system used for running LHCb software: LHCb started processing data in 2009/2010 with a RHEL5 derivative, for which support finished in 2017. Then RHEL6 and 7 derivatives were used and the move to RHEL9 (or equivalent distributions) started in 2023. Furthermore, upgrading allows using newer versions of the Linux *kernel* [115], the core of the system that runs the processes, providing benefits in terms of functionality and execution speed.

One more reason behind this change is that the underlying hardware needs to be recognised and handled by the system. As the technology advances, the operating system needs to be upgraded in lockstep: a system from 2006 will not support the devices present in a modern computer.

Virtual machines and containers The fact that it is difficult to completely isolate users from the operating system and other users has long been acknowledged by the computing industry. The idea of splitting the resources of a large computing system, dedicating parts to groups of users is as old as operating system design[117]: this was already done on the 1960s mainframes. It consists in running a so-called hypervisor which can run several systems at the same time. This approach is very powerful, as there is good isolation between the virtual machines installed, and it can even allow emulation of old hardware architecture but this comes at a cost:

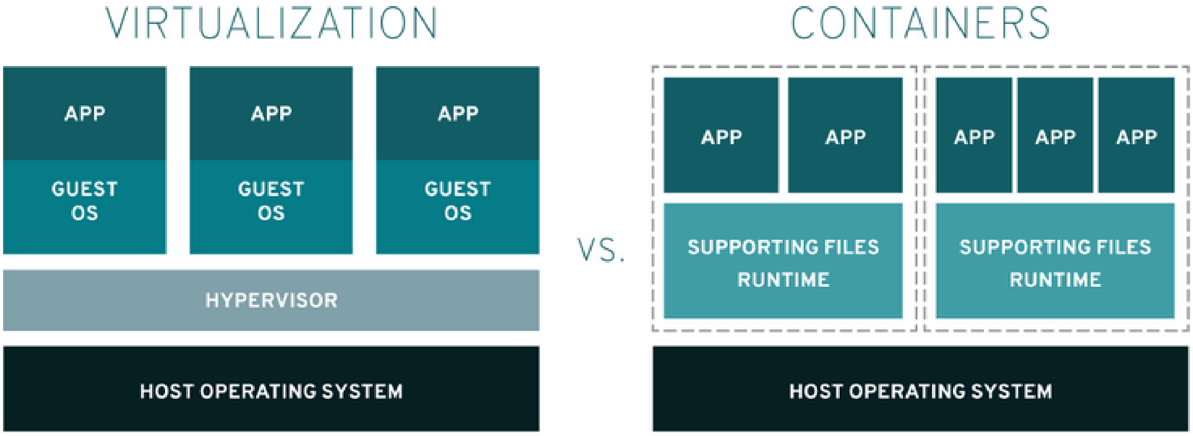


Figure 3.8: Comparison of virtual machines and containers (source RedHat[118]).

each system has to be installed separately, resulting in duplication of system files. Furthermore, the use of an hypervisor has a performance impact that can however be alleviated by the correct use of modern hardware.

Containers are a lightweight way to isolate processes, or group of processes on a Linux system. While they do not offer the same level of isolation between environments as virtual machines, they are much more lightweight to run. Figure 3.8 illustrates the differences.

It is possible to preserve the content of both virtual machines and containers, in order to restart them later. Containers have dependencies to the Linux kernel, whereas virtual machines only depend on the underlying hardware primitives and therefore might be more suited to long term preservation. Managing container images was eased by Docker [119], which was a precursor in this domain. One downside of preserving images is that this includes both the system files and the user installed files. This is also also inefficient in term of storage space, as each application potentially have their own image. Using Docker containers can alleviate this problem by preserving a *Docker file*, which is a recipe to prepare a container image, assembling pre-existing images and adding programs and files on top. Docker is not the only tool available in this domain: Podman [120] is compatible with docker and very popular. Singularity [121], now renamed Apptainer [122], was simpler to integrate into the WLCG infrastructure and was chosen to encapsulate LHCb jobs on the WLCG.

Preserving LHCb software

The LHCb software stack described in Section 2.4 consists of millions of lines of C++ and Python code. Preserving it is crucial for reproducibility and preservation. The experiment uses the GitLab development operation and management platform hosted by CERN[123] to keep track of the whole software history. Preserving the GitLab content is of course crucial, as it allows re-creating all LHCb tools provided the appropriate operating system is available and preserved as defined in Section 3.2.1. GitLab also contains extra information such as the issues related to the code, and their solutions, the code review comments which, while not strictly necessary to rebuild and rerun the software, give a lot of insight in its development and can help understanding problems.

All the applications and libraries used by LHCb to process data on the WLCG are installed on the CERN VM File system (CERNVM Filesystem (CVMFS)) [124]. It is planned to keep the integrality of its content, as this makes re-running the programs easier: this file system can be used directly from the processing jobs, or it is possible to extract the files needed to do so as in Ref. [125].

Detector geometry and conditions

Section 2.4.2 shows how the LHCb detector is described, and how the external conditions required for processing are preserved. To ensure reproducibility of the LHCb data processing, the Detector software project can describe each version of the LHCb detector ever used. This is already preserved as part of the software stack (see 3.2.1). The condition data is stored in a git repository on GitLab, which is also copied to CVMFS, as is done for the software.

Keeping track of data provenance

For every file registered in the LHCb bookkeeping system, it is possible to know which files it was derived from: listing 1 shows the files from which a file from an Analysis Production was derived.

```
$ dirac-bookkeeping-get-file-ancestors -l /lhcb/LHCb/Collision12/DATA_BS_WS.ROOT/00173017/0000/00173017_00000210_1.data_bs_ws.root
↪  --Depth=10
Getting ancestors for 1 files (depth 10) : completed in 1.3 seconds
Successful :
    /lhcb/LHCb/Collision12/DATA_BS_WS.ROOT/00173017/0000/00173017_00000210_1.data_bs_ws.root :
        /lhcb/LHCb/Collision12/BHADRONCOMPLETEEVENT.DST/00041834/0006/00041834_00069235_1.bhadroncompleteevent.dst : Replica-Yes
        /lhcb/LHCb/Collision12/BHADRONCOMPLETEEVENT.DST/00041834/0008/00041834_00081994_1.bhadroncompleteevent.dst : Replica-Yes
        /lhcb/LHCb/Collision12/BHADRONCOMPLETEEVENT.DST/00041834/0009/00041834_00090717_1.bhadroncompleteevent.dst : Replica-Yes
        /lhcb/LHCb/Collision12/BHADRONCOMPLETEEVENT.DST/00041834/0009/00041834_00090719_1.bhadroncompleteevent.dst : Replica-Yes
        /lhcb/LHCb/Collision12/BHADRONCOMPLETEEVENT.DST/00041834/0009/00041834_00093010_1.bhadroncompleteevent.dst : Replica-Yes
                                        /lhcb/LHCb/Collision12/FULL.DST/00020391/0004/00020391_00040776_1.full.dst : Replica-Yes
                                        /lhcb/LHCb/Collision12/FULL.DST/00020391/0004/00020391_00040778_1.full.dst : Replica-Yes
                                        /lhcb/LHCb/Collision12/FULL.DST/00020391/0004/00020391_00040783_1.full.dst : Replica-Yes
                                        [...]
                                        /lhcb/LHCb/Collision12/FULL.DST/00020846/0005/00020846_00054453_1.full.dst : Replica-Yes
                                        /lhcb/LHCb/Collision12/FULL.DST/00020846/0005/00020846_00054476_1.full.dst : Replica-Yes
                                        /lhcb/LHCb/Collision12/FULL.DST/00020846/0006/00020846_00061905_1.full.dst : Replica-Yes
                                        /lhcb/LHCb/Collision12/FULL.DST/00020846/0006/00020846_00061933_1.full.dst : Replica-Yes
                                        /lhcb/LHCb/Collision12/FULL.DST/00025046/0000/00025046_00000082_1.full.dst : Replica-Yes
                                            /lhcb/data/2012/RAW/FULL/LHCb/COLLISION12/123790/123790_0000000006.raw : Replica-Yes
                                            /lhcb/data/2012/RAW/FULL/LHCb/COLLISION12/123790/123790_0000000027.raw : Replica-Yes
                                            /lhcb/data/2012/RAW/FULL/LHCb/COLLISION12/123790/123790_0000000039.raw : Replica-Yes
                                            [...]
                                            /lhcb/data/2012/RAW/FULL/LHCb/COLLISION12/123790/123790_0000000092.raw : Replica-Yes
                                            /lhcb/data/2012/RAW/FULL/LHCb/COLLISION12/123790/123790_0000000097.raw : Replica-Yes
                                            /lhcb/data/2012/RAW/FULL/LHCb/COLLISION12/123790/123790_0000000106.raw : Replica-Yes
                                            /lhcb/data/2012/RAW/FULL/LHCb/COLLISION12/123790/123790_0000000118.raw : Replica-Yes
```

Listing 1: Data provenance among LHCb files

The indentation of the files show the ancestry level. The Analysis Production file contains the filtered data from 5 different files from the BHADRONCOMPLETEEVENT filtered stream of 2012 data. Those were obtained while filtering FULL.DST files, i.e. reconstructed events, from 2012 RAW data.

For each of the files, it is possible (listing 2) to get information about the DIRAC *job* that produced the file, including the site and the worker node where the job was run, and the application name and version that was used to produce it.

```
$ dirac-bookkeeping-job-info /lhcb/LHCb/Collision12/BHADRONCOMPLETEEVENT.DST/00041834/0006/00⌋
↪  041834_00069235_1.bhadroncompleteevent.dst
Failed : []
Successful :
    /lhcb/LHCb/Collision12/BHADRONCOMPLETEEVENT.DST/00041834/0006/00041834_00069235_1.bhadron⌋
    ↪  completeevent.dst
    ↪  :
        Job 95540455 :
               DIRACVersion : v6r11p23
                   Location : LCG.IN2P3.fr
                 Production : 41834
            TotalLumonosity : 0
                    WNCACHE : 2600.169
                  WNCPUHS06 : 7.5
                 WNCPUPOWER : 1808
                   WNMEMORY : 768896.0
                  WNMJFHS06 : None
                    WNMODEL : Intel(R)Xeon(R)CPUE5-26700@2.60GHz
                 WORKERNODE : ccwsge0231
        Step 00041834_00069235_1 :
                            LFN : /lhcb/LHCb/Collision12/BHADRONCOMPLETEEVENT.DST/00041834/00⌋
                            ↪  06/00041834_00069235_1.bhadroncompleteevent.dst
                ApplicationName : DaVinci
             ApplicationVersion : v36r1
                        CPUTIME : 10275.85
                 EventInputStat : 2579628
                       ExecTime : 10357.4413559
               FirstEventNumber : 1
                          JobId : 272283069
                 NumberOfEvents : 45648
            StatisticsRequested : -1
```

Listing 2: Detailed information about a specific file provenance

All information related to the files is stored in an Oracle relational database, described in Figure 3.9. This schema structures the information kept about all LHCb files: it allows different fields for real data vs Monte Carlo simulation, allows setting data quality flags needed for operational purposes, and fulfils the LHCb file processing needs. The links between files and jobs, as shown when extracting file provenance are clear, as well as the various attributes available on the data processing steps. While this schema provides the needed information to perform the LHCb data processing, it is however inflexible, and some more information is needed when dealing when files needed for physics analysis, as will be shown in chapter 4.

Finding the data

All LHCb production data are registered in the LHCb bookkeeping database. This system, described in [126], features a hierarchical view of the data shown in Figure 3.10, derived from the database schema shown in Figure 3.9. To each such *bookkeeping path* corresponds a number

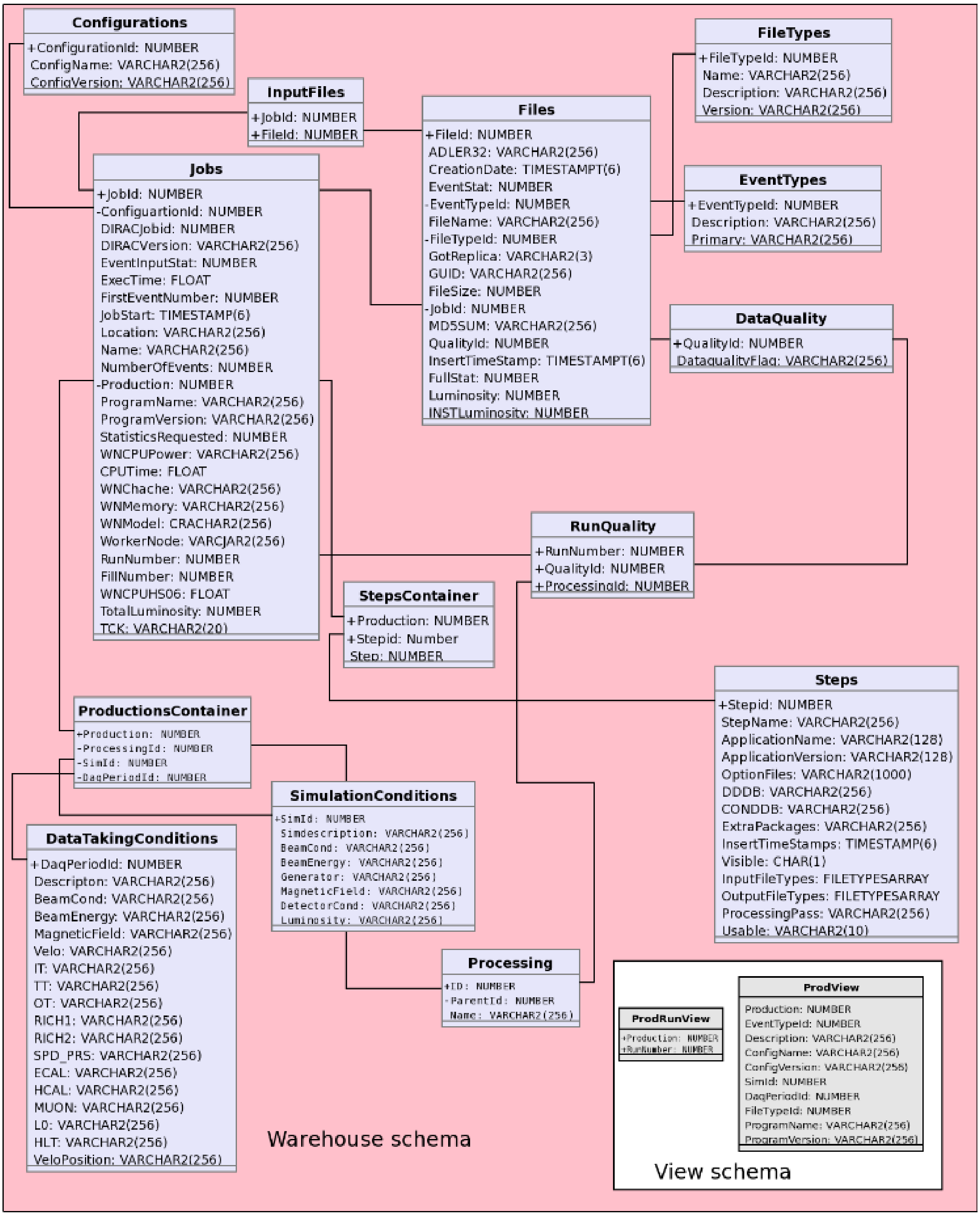


Figure 3.9: LHCb bookkeeping metadata structure according to[126].

of files with similar attributes.

The LHCb bookkeeping interface allows LHCb users to browse the production files according to paths such as in Listing 3:

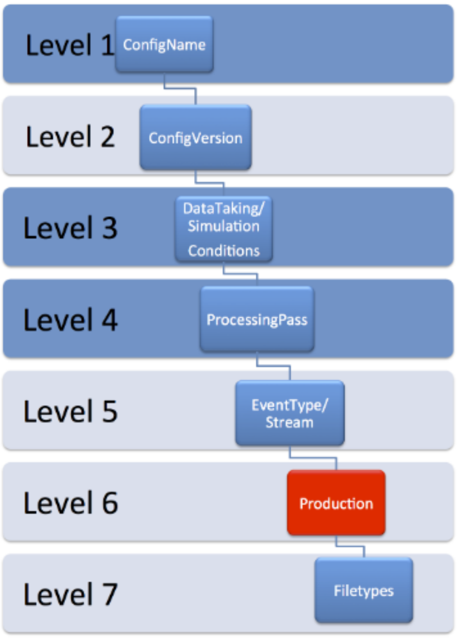


Figure 3.10: LHCb bookkeeping path structure according to [126].

```
/LHCb/Collision12/Beam4000GeV-VeloClosed-MagDown/Real
↪  Data/Reco14/Stripping21/90000000/SEMILEPTONIC.DST
```

Listing 3: Example of a dataset path in the bookkeeping system.

with:

- At level 1 from Figure 3.10, the *configName* **/LHCb** meaning that we deal with production data, Monte Carlo simulations are under MC.
- At level 2 the *configVersion* or data type **Collision12** means collision data from 2012, there is one such type for each data taking year plus others for calibration data.
- At level 3 the *conditions* **Beam4000GeV-VeloClosed-MagDown** corresponds to the data taking conditions: beam energy, magnet polarity...
- At level 4 the *processingPass* can be found. **Reco14** corresponds to the reconstruction software that was used to reconstruct the data, **Stripping21** corresponds to the filtering and indexing step.
- At level 5 follow the *EventType* and *stream*, respectively **90000000** and **SEMILEPTONIC.DST**.

As a part of the DIRAC middleware system, it is possible to retrieve the configuration of the computing jobs that created the files: each *processing step* is carefully recorded with all the versions of the software tools to be used, the condition configuration and the configuration needed to run the software.

While DIRAC contains the details of all processing steps, it only records the name of the *processing pass* they are related to. Details of the various offline processing passes of the LHCb data (for example which is the latest reconstruction/filtering for a given year of data taking) can be found in the processing passes twiki[127]. This website is readable by users, but it does not allow software applications to query the data. This should be improved and the data made available in machine readable format, to make the data more *Findable* in the FAIR sense.

It is also interesting to note that the files are not self-describing: while offline files are stored in the root format, the physics objects location in the file may depend on the version of the stripping software used (e.g. the name of the stripping line is included in the object path). The stripping configuration is a python module module that can be queried from any python program, provided the version of the Stripping software used to process the file was loaded. Understanding the content of a LHCb file therefore requires to know how to gather information from the processing passes twiki, the stripping/sprucing project pages. The developers of the Ntuple wizard (see Section 3.3.1) started gathering this information in their tool in order to make the search for specific decays in LHCb data files possible.

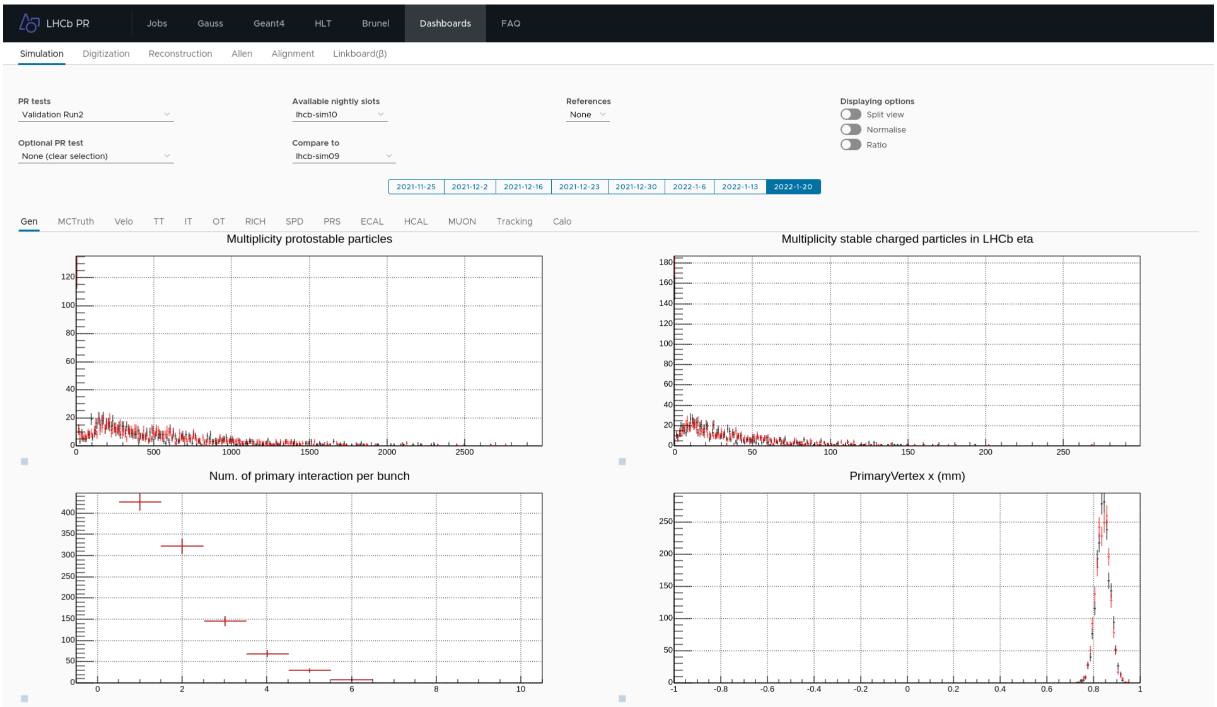
Quality assurance and preservation

Significant efforts are spent within the LHCb collaboration to ensure the quality of the results produced, and therefore on the quality of the software to derive them. In 2012, LHCb started developing its Performance and Regression testing application (LHCbPR) as presented at the Computing for High Energy Physics conference in 2014 [128]. This application is continuously upgraded [129] [130]. It allows checking for example:

- the quality and performance of Monte Carlo simulations [131],
- the LHCb trigger [132] output and speed.

Figure 3.11 shows for example how easy it is to compare the result of the simulation of the LHCb detector for two different versions of the software stack, in this instance the characteristics of the collisions generated.

Thousands of measures are collected in each job related to the various aspects covered by LHCb software: the generation of events, their simulation in the detector, the tracking of the particles, the behaviour of the Alignment software etc. LHCbPR is therefore crucial in the short term, to ensure the quality of the data processing tools, but it is also key for preservation as it allows verifying that the results haven't been affected by the changes in underlying infrastructure. To ensure the LHCbPR system can be preserved, the results are stored in a MySQL relational database, hosted by the CERN IT department, and the histogram are kept in ROOT format, the same format as the LHCb data files.



PR tests: Validation Run2

Optional PR test: None (clear selection)

Available nightly slots: lhcb-sim10

Compare to: lhcb-sim09

References: None

Displaying options
○ Split view
○ Normalise
○ Ratio

Figure 3.11: Comparison of the LHCb simulation output with two different versions of the software stack.

Conclusion on the preservation of production data

Preserving experimental data at the LHC is a gigantic task, especially with the data volumes involved, and it is worth noting that:

- this is an issue faced by all experiments and the institutes participating in the WLCG have made significant investments to preserve data at the physical level (tape libraries).
- the bookkeeping application is essential to trace the data derived from LHCb RAW data, and therefore must be preserved in the long term. It currently includes a large Oracle database supported by the CERN IT department. Long-term plans for the support of this application should be established in view of the growing amount of data recorded by the experiment.
- the metadata associated with LHCb data taking should also be preserved: the Run database records the data taking conditions. While some of its information is redundant with the bookkeeping system, it is nonetheless useful to easily query information about recorded runs. Information from the Monet online data quality system is also important to understand the properties of the recorded data.
- files are recorded in MDF (LHCb-specific) or ROOT (HEP-specific) formats. In all cases, LHCb software is crucial to exploit the data as the decoding of the files. Furthermore some of the information needed to interpret the files is scattered across several locations

(processing passes twiki, DIRAC job, stripping project configuration...) which does not make the task easy (hence the creation of the Ntuple wizard for Open Data which gathers that information and makes it accessible).

The software stack needed to process LHCb data is complex but its development is managed in a way that allows preservation. The key aspects of the preservation are that:

- the source code repositories for the applications and libraries should all be preserved (and more generally the whole GitLab server). Detector conditions are also stored in GitLab and should also be preserved, thus reinforcing this need.
- Environments for which the software was prepared must be preserved, and it should be possible to build and run applications within them. Virtual machines and containers are already available, but it is necessary to test that the technology, as it evolves, fulfils the preservation needs.
- Preserving the CVMFS repositories used for production is also useful, as it contains the compiled versions of the applications used to process LHCb data. Its size is limited (a few TB) compared to the amount of data preserved in any case.
- The LHCbPR quality assurance system is key in guaranteeing the physics performance of the system. Preserving its data is also important to continue using the software in the long term. This involves preserving its relational database (MySQL) as well as all the related files.

3.2.2 Preservation of Analysis repositories

As of 2018, new analyses being published were required to provide a repository containing the analysis code, as a GitLab project in the instance hosted by CERN[123]. The figures 3.13 and 3.12 show that indeed, nearly all analyses published by the LHCb experiment since 2018 feature a GitLab project with the script for preservation. The repositories reflect the diversity of tasks that have to be performed during an analysis. A variety of tools are used to perform the analysis tasks, developed mostly as ROOT macros (interpreted C++), compiled C++ programs or python scripts.

Figure 3.14 shows that ROOT macros, the *classic* way to analyze HEP data is prevalent. The python language and its interface with ROOT are also used in nearly all analyses. It is however interesting to see the growing popularity of newer analysis tools, such as ROOT::RDataFrame[133] or the Snakemake workflow engine [134].

Investigating the content of the repositories, it is only possible to link the analysis with the data used by reading the analysis note, or various documents in among the other files. There is no convention for finding which input data was used from the whole LHCb dataset. Furthermore, the only way to find out the analysis workflow is to read the note and match the various steps with the tools in the analysis repository and the correct input data. Analyses using workflow engines are simpler to re-run, provided the input data is still in place. If it was produced using

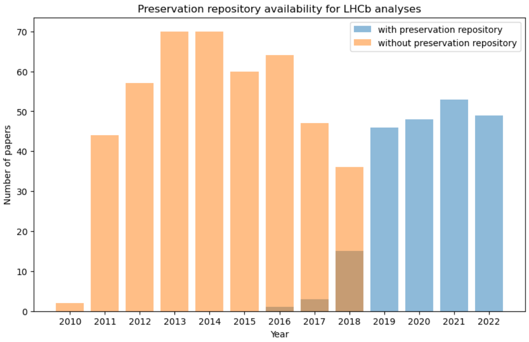


Figure 3.12: Number of published LHCb analyses with or without git repositories.

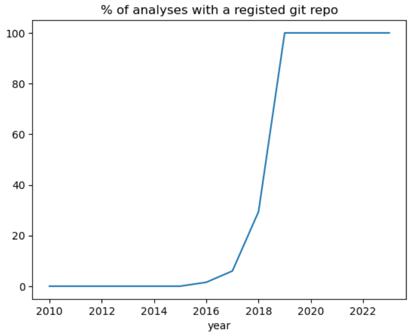


Figure 3.13: % of published LHCb analyses with git repositories.

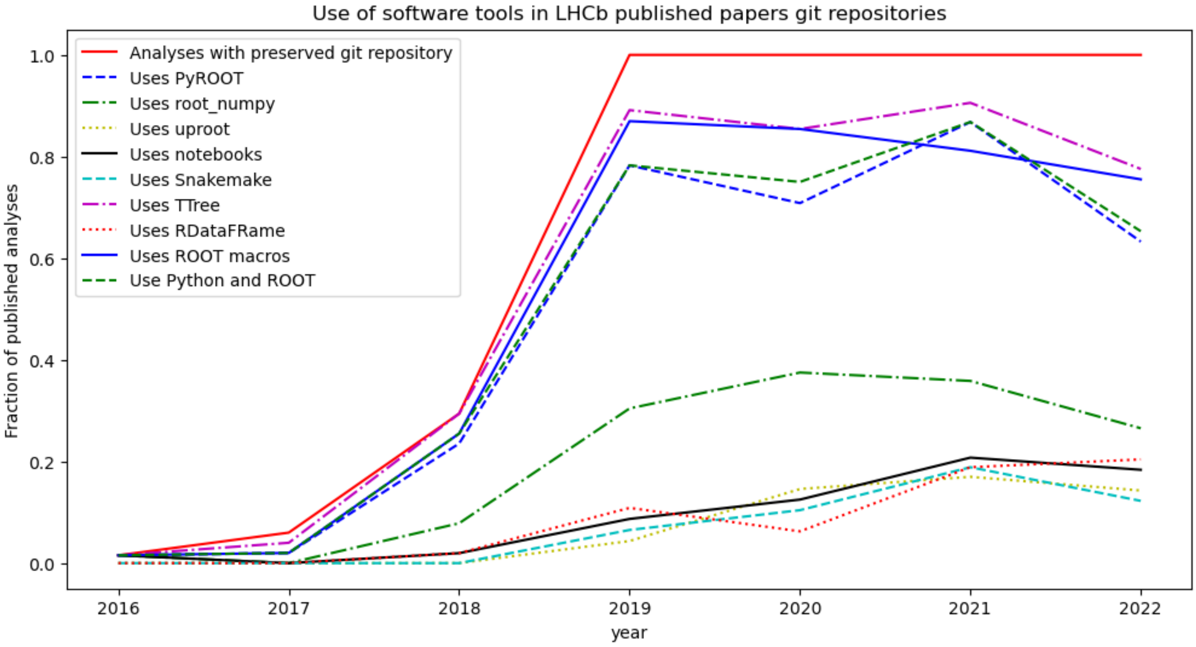


Figure 3.14: Evolution of software tools use in analysis repositories. Information extracted by parsing the code committed in the repositories containing the analysis code.

the Ganga tool, one has to hope that the Ganga scripts have not been modified since the jobs were run.

The critical issue to be addressed is data provenance, or how to link the ntuples analysed to LHCb data. Analysis Productions (see Section 3.3.1) introduced for LHC Run 3 improve the situation.

3.3 Analysis preservation tools

A number of tools help with analysis preservation and open data, both developed within the LHCb collaboration or across experiments. This section presents them and how they are used or planning to be used within the LHCb collaboration. The notion of *workflow*, which refers here to *computational workflows* has been mentioned several time in this chapter, and it is necessary to explain first the amplitude of challenge at hand, and why workflows are are regarded as appropriate tools for preservation.

Data analysis at LHCb

The LHCb collaboration gathers more than 1400 scientists, engineers and technicians representing 86 different universities and laboratories from 18 countries. Data analysis is performed by the members of the collaboration and organised in a numbers of working groups focusing on different physics processes, currently they are:

- QCD, Electroweak and Exotica
- B hadrons and Quarkonia
- Charm physics
- Rare decays
- B decays to Charmonia
- Charmless b-hadron decays
- B decays to Open Charm
- Semileptonic decays
- Ions and Fixed Target

Several other working groups exist to organise the work around the monitoring of physics performance, statistics and so on. Tens of analyses are ongoing at any time, involving many analysts and requiring a bit less that 0.5 PB of data for their ntuples. At the time of this writing, 79 analyses were being reviewed, i.e. in the last stages of the process, involving 142 different analysts. This figure of course excludes all analyses not at that stage. It is therefore critical to find appropriate tools in order to save time and resources for all these efforts.

The need for computational workflows

Computational workflows describe processes to be performed in term of tasks and their dependencies. At execution time a *workflow engine* determines the tasks to be executed and their order, given the artefacts to be produced. This approach decouples the definition of how to perform the steps from the data items to which they should be applied and is very useful to be able to extend the processing to new data items.

Furthermore, one inherent advantage is that the dependency graph between the various task is one of the artefacts derived by the *workflow engine* and can very often be exported to extract metadata or for further visualisation. This is a key ingredient to enable the application of the

FAIR principles [135]. This is particularly interesting in the case of complex analyses that can include hundreds of steps.

Another major advantage is that workflows allow decoupling the tasks from where they should be run, and helps identifying places where parallelisation of the work is possible. It is then possible to develop a workflow on a personal machine and, once ready, extend its use to many data items on a large cluster.

This approach has a long history and workflows are used in many domains: for business process modelling this lead to the Business Process Model Notation (BPMN) [136]; some workflows are dedicated to specific computing tasks (c.f. the Make tool to build computer programs). There are many different workflow tools with their strong and weak points. Even in the Scientific Computing domain, many different workflow engines are available and Ref.[55] reviewed some of them. *Snakemake* [134] is a very interesting tool due to its simplicity and its integration with the Python programming language, which is very popular for data analysis. Its application to LHCb data processing is reviewed in Chapter 4.

Preservation tools landscape

Table 3.1 lists the main tools available to preserve LHCb analyses and their purpose. They are presented in details in the next sections.

Tool	**LHCb internal**	**Lifecyle phase**	**Purpose**
Analysis database	yes	Throughout	Track analyses performed at LHCb
Analysis Productions	yes	Data Extraction	Extract ntuples for analysis
Ntuple wizard	yes	Open data	Easy extraction of ntuple from LHCb open data
HEP Data	no	After publication	Gather measurement results from HEP experiments
CERN Open Data portal	no	Open data	Portal to provide access to CERN experiments Open Data
CERN Analysis preservation	no	After publication	Preserve information of analyses published by CERN experiments
REANA	no	After publication	Re-run analysis workflows on demand

Table 3.1: Summary of data and analysis preservation tools available to LHCb analysts.

3.3.1 LHCb Internal tools

Analysis Database

To ensure that the correct procedure for validating the results produced by the experiment, and described in section 3.1.1, LHCb uses a mix of tools: spreadsheets, twikis as well as a publication database[137]. While all the information related to the ongoing and published analyses is recorded, it is difficult to search for specific information due to the different data sources and formats. Furthermore, this does not fulfil the FAIR requirement that all information should be available in machine readable format for easier search and integration with other tools.

To improve this situation, a project is currently under way to include more information in the Glance system [138, 139] which is already used to manage collaboration-wide information, such as the memberships. The Analysis Lifecycle Management (ALCM) part of Glance is therefore being developed; for the moment, only workflows to manage LHCb public figures are being considered.

Analysis Productions

Chapter 2 demonstrated the complexity of the chain of operations required to process LHCb data. The triggering and first filtering are common stages and the preservation of the software tools, environment and configuration are part of the release process. Using Ganga to extract data from production files gave a lot of freedom to the analysts, but also loaded them with the burden of checking the jobs on the WLCG and making sure that the provenance of the ntuples was carefully recorded.

This was not so easy and Analysis Productions (AP) [140] were introduced as a system with a low entry bar that also improves the provenance tracking of the ntuple produced. Instead of embedding the provenance data in the ntuples themselves, as proposed by [91, 141], AP run user jobs within the LHCb DIRAC production system, thus making sure that the processing is carried out successfully and provenance is automatically tracked.

Another key aspect of the Analysis Productions is the declarative aspect of the configuration, which is defined in a file from which rich metadata can easily be extracted. It is kept in a GitLab repository and is fully versioned. It is therefore always possible to re-run AP, as their code is fully tracked, and as they depend on production application versions which are preserved alongside the LHCb software stack.

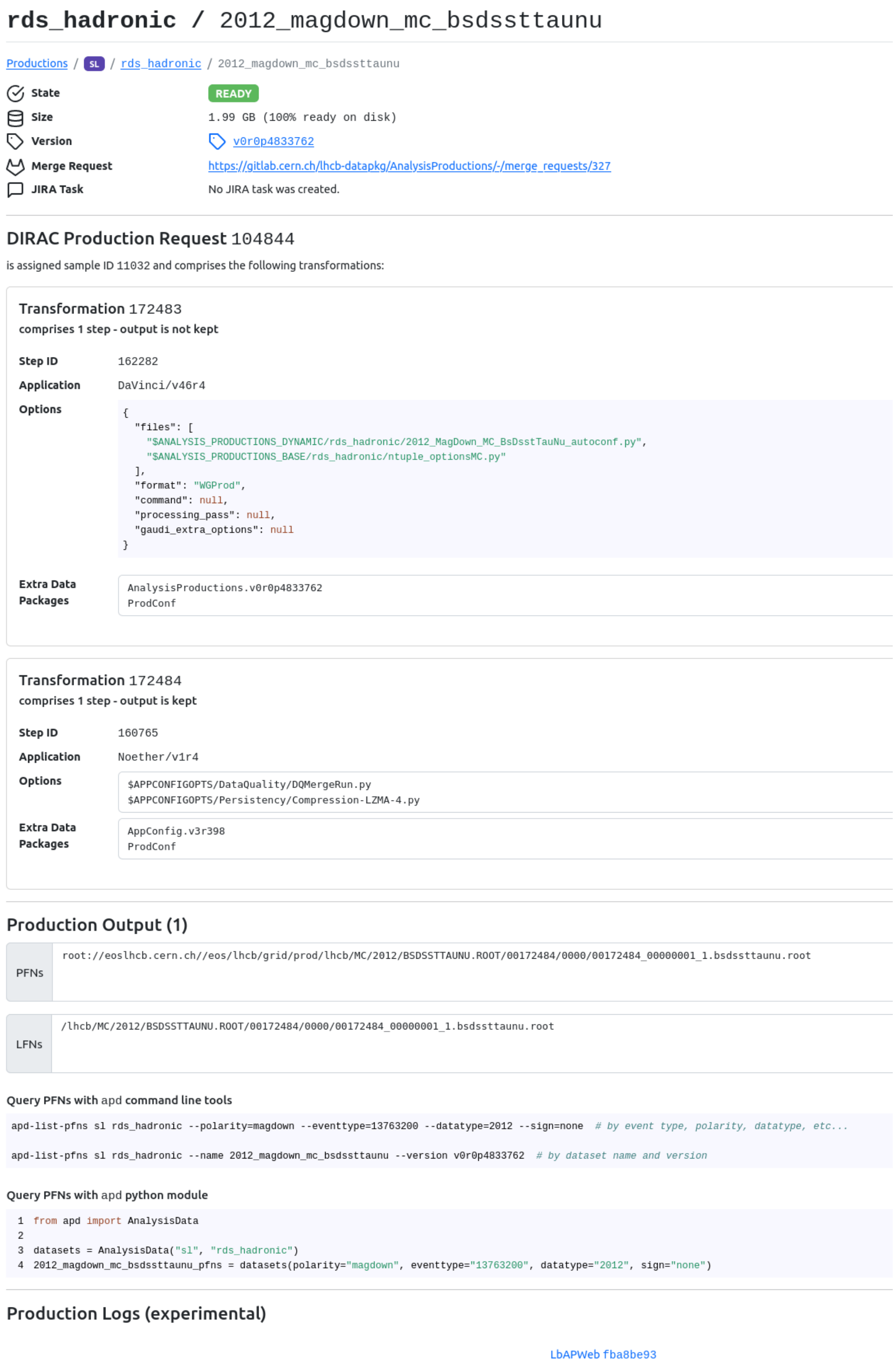
Analysis Productions are very attractive to LHCb physicists as they do not need to follow up on the processing as this is done by the DIRAC system. Furthermore, a system of test of the requests was put in place to avoid saturating the system with problematic code: the GitLab Continuous Integration system is used to check the output of the submitted Analysis Production on one data file. This allows checking for potential bugs and provides an evaluation of the extracted data size. Only after validation can the production proceed on the full dataset.

The user can follow the progress of her or his request with a dedicated user interface (Figure 3.15) which is simpler that that of the LHCb Dirac portal as it hides the complexity of that system, and presents only the Analysis Production related information.

The results are referenced in the LHCb bookkeeping which keeps track of all files produced by LHCb[126] [142]. This is a great improvement over individuals extracting data from the production files:

- the files are stored in the LHCb grid area, and cared for in the same way as production files: they are stored on hosts with redundant storage and managed by the LHCb grid production team.

- they are registered on the LHCb bookkeeping, and therefore can be found by any LHCb analyst, using the dedicated browser. In practice this is however difficult as the Analysis



rds_hadronic / 2012_magdown_mc_bsdssttaunu

Productions / SL / rds_hadronic / 2012_magdown_mc_bsdssttaunu

State	READY
Size	1.99 GB (100% ready on disk)
Version	v0r0p4833762
Merge Request	https://gitlab.cern.ch/lhcb-datapkg/AnalysisProductions/-/merge_requests/327
JIRA Task	No JIRA task was created.

DIRAC Production Request 104844

is assigned sample ID 11032 and comprises the following transformations:

Transformation 172483

comprises 1 step - output is not kept

Step ID 162282

Application DaVinci/v46r4

Options

```
{
  "files": [
    "$ANALYSIS_PRODUCTIONS_DYNAMIC/rds_hadronic/2012_MagDown_MC_BsDsstTauNu_autoconf.py",
    "$ANALYSIS_PRODUCTIONS_BASE/rds_hadronic/ntuple_optionsMC.py"
  ],
  "format": "WGProd",
  "command": null,
  "processing_pass": null,
  "gaudi_extra_options": null
}
```

Extra Data Packages

```
AnalysisProductions.v0r0p4833762
ProdConf
```

Transformation 172484

comprises 1 step - output is kept

Step ID 160765

Application Noether/v1r4

Options

```
$APPCONFIGOPTS/DataQuality/DQMergeRun.py
$APPCONFIGOPTS/Persistency/Compression-LZMA-4.py
```

Extra Data Packages

```
AppConfig.v3r398
ProdConf
```

Production Output (1)

PFNs

```
root://eoslhcb.cern.ch//eos/lhcb/grid/prod/lhcb/MC/2012/BSDSSTTAUNU.ROOT/00172484/0000/00172484_00000001_1.bsdssttaunu.root
```

LFNs

```
/lhcb/MC/2012/BSDSSTTAUNU.ROOT/00172484/0000/00172484_00000001_1.bsdssttaunu.root
```

Query PFNs with apd **command line tools**

```
apd-list-pfns sl rds_hadronic --polarity=magdown --eventtype=13763200 --datatype=2012 --sign=none  # by event type, polarity, datatype, etc...

apd-list-pfns sl rds_hadronic --name 2012_magdown_mc_bsdssttaunu --version v0r0p4833762  # by dataset name and version
```

Query PFNs with apd **python module**

```
from apd import AnalysisData

datasets = AnalysisData("sl", "rds_hadronic")
2012_magdown_mc_bsdssttaunu_pfns = datasets(polarity="magdown", eventtype="13763200", datatype="2012", sign="none")
```

Production Logs (experimental)

LbAPWeb fba8be93

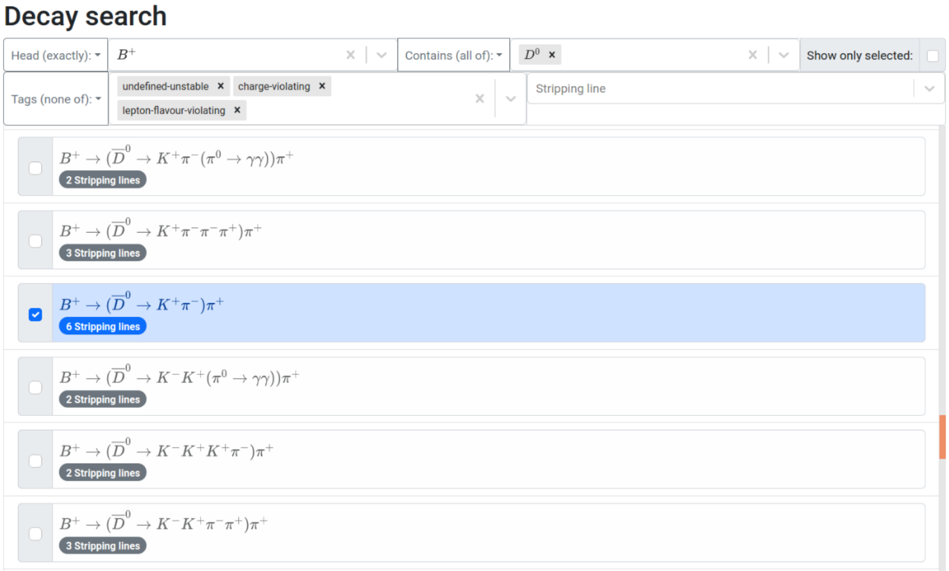
Figure 3.15: Detailed information on a data sample as viewed in the Analysis Productions interface.

Productions are referenced by name and version which are not explicit. This point will be detailed and remedied to in chapter 4.

Analysis Productions can replace Ganga to extract ntuples from LHCb production data, but they however do not cover the use case for parametrised Monte Carlo jobs (also known as toy Monte Carlo), simulation jobs run by analysts for example to evaluate the sensitivity of a measurement to the variations of various parameters.

Ntuple wizard

In order to provide access to the data released by LHCb in the CERN Open Data Portal, LHCb developed the Ntuple wizard [143, 144]. This web applications allows third-party users to request derived data samples in the format used in LHCb physics analysis (Ntuples). It has been designed to limit the LHCb-specific knowledge needed to extract data. Users can chose from a list of pre-defined decays and selections present in the data, as can be seen in Figure 3.16.



Decay search

Head (exactly): B^+

Contains (all of): D^0

Show only selected: □

Tags (none of): undefined-unstable, charge-violating, lepton-flavour-violating

Stripping line

- □ $B^+ \to (\overline{D}^0 \to K^+\pi^-(\pi^0 \to \gamma\gamma))\pi^+$ 2 Stripping lines
- □ $B^+ \to (\overline{D}^0 \to K^+\pi^-\pi^-\pi^+)\pi^+$ 3 Stripping lines
- ☑ $B^+ \to (\overline{D}^0 \to K^+\pi^-)\pi^+$ 6 Stripping lines
- □ $B^+ \to (\overline{D}^0 \to K^-K^+(\pi^0 \to \gamma\gamma))\pi^+$ 2 Stripping lines
- □ $B^+ \to (\overline{D}^0 \to K^-K^+K^+\pi^-)\pi^+$ 2 Stripping lines
- □ $B^+ \to (\overline{D}^0 \to K^-K^+\pi^-\pi^+)\pi^+$ 3 Stripping lines

Figure 3.16: Example of decay candidate search in the LHCb Ntuple Wizard [144].

This interface is interesting as, in line with the CERN Open Data Policy, it is an investment that opens the data for further use. It explicits the content of the files and inherently makes the data *Findable*, as defined by the FAIR principles. In order to do so, the Ntuple Wizard exploits information from the LHCb bookkeeping and data collated from the LHCb software tools documentation as shown by Figure 3.17.

The Ntuple Wizard produces a configuration that can be used to run LHCb Analysis Productions to extract information from the reconstructed and filtered data.

3.3.2 External tools

HEPdata

The Durham High Energy Physics Database (HEPData) [145, 146] is a repository for gathering data from experimental particle physics papers. Its inception dates from the 1970s and

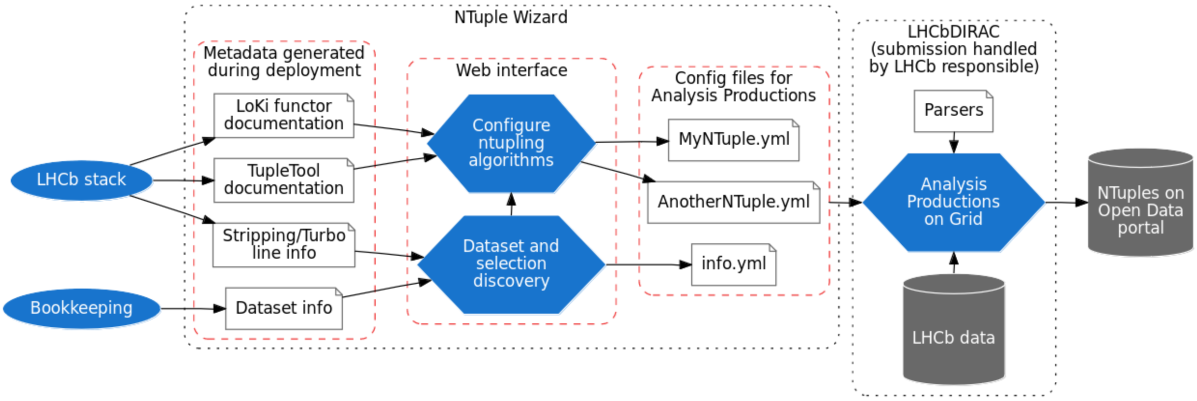

NTuple Wizard
Metadata generated during deployment
LoKi functor documentation
TupleTool documentation
Stripping/Turbo line info
Dataset info
LHCb stack
Bookkeeping
Web interface
Configure ntupling algorithms
Dataset and selection discovery
Config files for Analysis Productions
MyNTuple.yml
AnotherNTuple.yml
info.yml
LHCbDIRAC (submission handled by LHCb responsible)
Parsers
Analysis Productions on Grid
LHCb data
NTuples on Open Data portal

Figure 3.17: LHCb Ntuple Wizard architecture [144].

it currently includes data from several thousand publications, including those from the LHC experiments, and corresponds to Level 1 data according to the DPHEP classification. It is therefore complementary to the CERN Open Data Portal, which focuses on the release of data from Levels 2 and 3.

HEPData focuses on measurements such as production cross sections, but in the last few years its scope broadened to data from particle decays and neutrino experiments, and data used for tuning physics event generators, for example using the Rivet Framework [147].

CERN Open data portal

Following the creation of DPHEP, it was clear that a common framework was needed to help preserve HEP analysis data. In 2014, the CERN Open Data Portal, a common effort between the CERN Scientific Information Service and CERN IT was created[148] [149]. Using key pre-existing software already available at CERN, it provides an easy way to preserve and make available analysis data:

- The Invenio digital library software [150] which provides a front-end application.

- The EOS storage system [151] which provides the required backend to store the data.

The Open Data Portal is designed to host Level 2 data for outreach, as well as the data shared by HEP experiments: all LHC experiments have added some data in open access, and more than 4 petabytes are available as of 2023.

CERN Analysis Preservation

As all experiments face the need to preserve their scientific results, CERN decided to put in place a common CERN analysis preservation portal (CAP)[148]. It aims to gather and structure information about all published data analyses at CERN. It allows physicists to

- describe all artefacts produced,

- track provenance,
- collect documentation,
- gather context information.

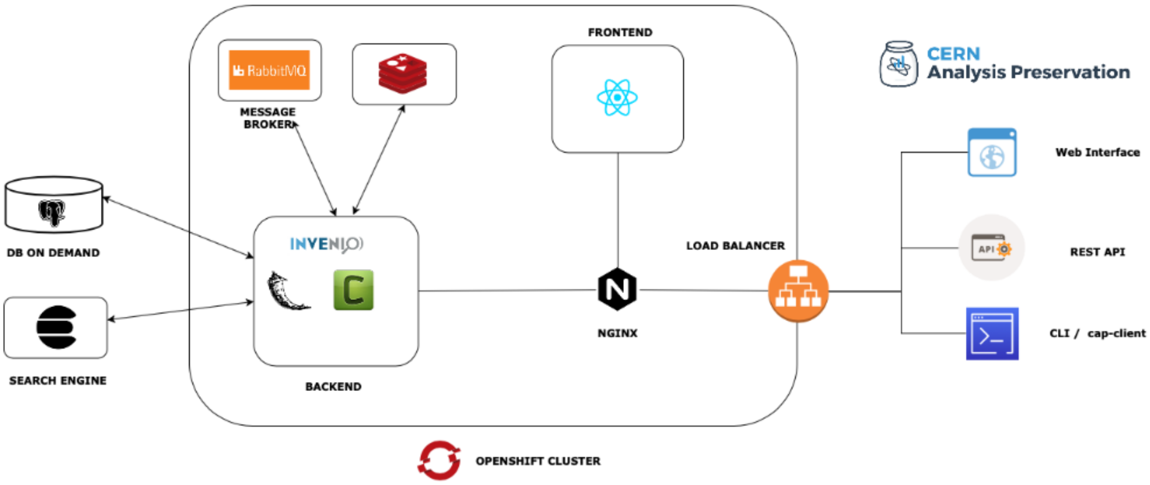


Figure 3.18: General architecture of the CERN Analysis Preservation Portal [148] (non exhaustive list of services).

For this purpose, it aims to collect the data (e.g. datasets, code, results) and the metadata (e.g. analysis name, provenance information, related publications, etc.). This is done using a user-friendly website and a command line tool. Workflows can be submitted as well, and containerised payloads can be re-run using REANA detailed below. It should be possible to extract metadata directly from the LHCb systems (such as the Analysis LifeCycle Management System being developed). The architecture of CAP is shown in Figure 3.18. Like the CERN Open Data portal it uses the Invenio digital repository to manage the collected information and CERN services to run the services and databases as well as store data files.

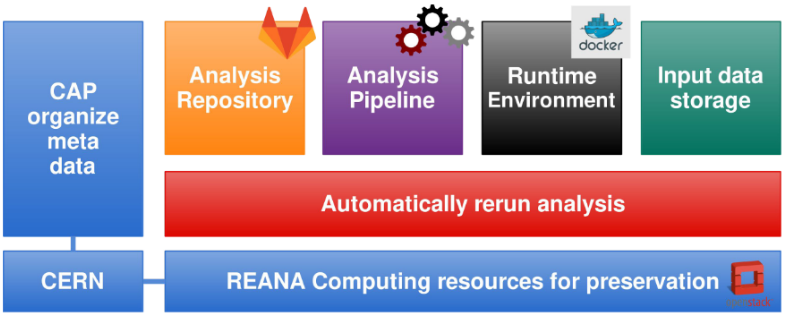


Figure 3.19: Systems involved in the LHCb analysis preservation roadmap.

CAP is part of LHCb's analysis preservation strategy, as indicated by the Figure 3.19 taken from the LHCb analysis preservation roadmap, which shows the role of CAP and of REANA in the LHCb preservation strategy.

REANA

The REusable ANAlysis system allows physicists to run containerised data analysis pipelines on computing clouds. Its goal is to provide *workflow-as-a-service* functionality on top of existing CERN infrastructure.

Figure 3.20 illustrates its design: a REANA cluster is used as a gateway to CERN computing resources. It can be accessed either using *reana-client*, a Python package that can be installed on local nodes using standard Python installation tools, or via a browser connecting to a notebook on the gateway.

REANA supports a number of workflow engines, including *Snakemake* which is in use at LHCb. It executes the workflows on the computing infrastructure available at CERN. To simplify the deployment of workflows, REANA is integrated with GitLab for the source code management, CVMFS to be able to use binaries and environments deployed there. Docker containers are used to have a well-defined environment for the analysis. Artefacts can be copied to the EOS mass storage system for persistency.

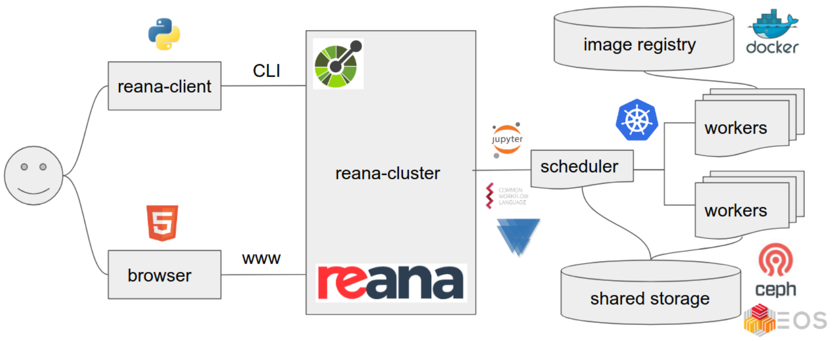


Figure 3.20: REproducible ANAlysis platform design.

This platform provides a convenient way to re-run analyses that have been defined as one of the supported workflows.

Its separation from the LHCb infrastructure is potentially a good isolation test to ensure that no hidden dependency was forgotten during preservation.

3.3.3 Conclusion

Preserving High Energy Physics analysis results is a complex task because of the complexity of the analyses, the number of analyst involved, the volumes of data involved and the amount of processing necessary. This is however crucial both to ensure the quality results produced, and to make sure we can continue exploring the datasets in the future.

While this perspective is daunting, this is not a new realisation and common efforts started even before LHC experiments took data. The Data Preservation in High Energy Physics working group proposed solutions that were developed in the following years: all LHC experiments have plans for sustainable preservation of their production data, and there are common tools and services available to preserve the analysis artefacts (CAP), as well as to re-run the workflows (REANA).

Preservation of the LHCb centrally processed data and of the associated software always was a key preoccupation of the LHCb Computing team and is part of the system design. As Software Librarian for the experiment, the proponent of this thesis was personally involved in the design of the system [152], the continuous integration tools [153, 154, 155, 156] and the implementation of some of the tools used (e.g. the quality assurance system LHCbPR system [128, 129, 130, 157]).

Concerning analysis data, the repositories provided for preservation of LHCb analyses since the LHCb Analysis Preservation roadmap (in which this thesis proponent was involved [94]) was implemented show large differences in what is preserved, depending on the analysis: this is visible just by counting the number of files per repository, which goes from a few tens to to tens of thousands. In some cases, there are scripts to perform the final fits on the preserved ntuples published in the paper whereas in some others the scripts necessary to extract data using Ganga. This is a consequence of the differences between the analyses, and of the fact that no strict guidelines about what should be preserved were specified. To understand what data is used, it is necessary to read through the analysis descriptions and this cannot be automatised, so there is no way to ensure the quality of the provenance of the ntuples published with the analysis. Gathering metadata to publish to the CERN Analysis Preservation portal has to be done manually at the end of the analysis. Considering that analyses take several years to complete, guaranteeing the completeness and accuracy is difficult.

The best way to preserve LHCb analyses therefore seems to provide tools and methods that can help recording the data provenance during the analysis. The approach taken by this work introduces lightweight tools to help analysts in that direction: keeping track of provenance has to be done from the start of an analysis, and that to be successfully adopted, tools should have a minimal adoption barrier and help analysts instead of introducing extra burden. The Analysis Production framework allows for better provenance tracking of analysis ntuples but was successful because it alleviates the burden of following up on the data extraction jobs on the grid. Building on this success, we can provide a way to enrich the metadata describing the analysis, making them more *Findable* and *Reusable* according the FAIR principles. The key point is that it can also make it easier for physicists to perform their measurements. The next chapter describes how this was implemented in the LHCb environment.

Chapter 4

Improving the reproducibility of LHCb analyses

Tools are already available to preserve analyses and to re-run them, and Chapter 3 concluded that the best approach was to help LHCb analysts gather all the data needed for preservation. The ntuple provenance is a crucial part of this information, as also explained in that same chapter. Within LHCb, analysts traditionally create ntuples selecting their data from the production files and keep copies on personal or shared space. Tracking provenance with this setup is difficult, as the extraction scripts have to be preserved as well. This is very error prone if multiple files extractions have to be done. Analysis Productions introduced in Section 3.3.1 improved the situation, as the data selection configuration is preserved, but they did not solve all problems: physicists either copy (and merge) the ntuples to personal or working group space which is wasteful of disk space, or they keep a list of hard-coded locations within their analysis code. While this is easy to do in a first approach, it is not not durable as changes in the mass storage systems may force a move of the files, thus breaking the analysis code reading them. This chapter delves into the details on the LHCb bookkeeping and Analysis Productions system needed to understand how this problem can be remedied, as well as into the details of the developments undertaken during this thesis to do so.

4.1 Data provenance

4.1.1 Analysis Production metadata

As stated in the FAIR principles, to be easily findable, data should be described by metadata that qualifies it with enough details (*rich*). The LHCb bookkeeping system organises the data as a tree, and the datasets as a leaf. Datasets are therefore associated to a path in the tree which is organised in a way that characterises the data taking periods and conditions. For example:

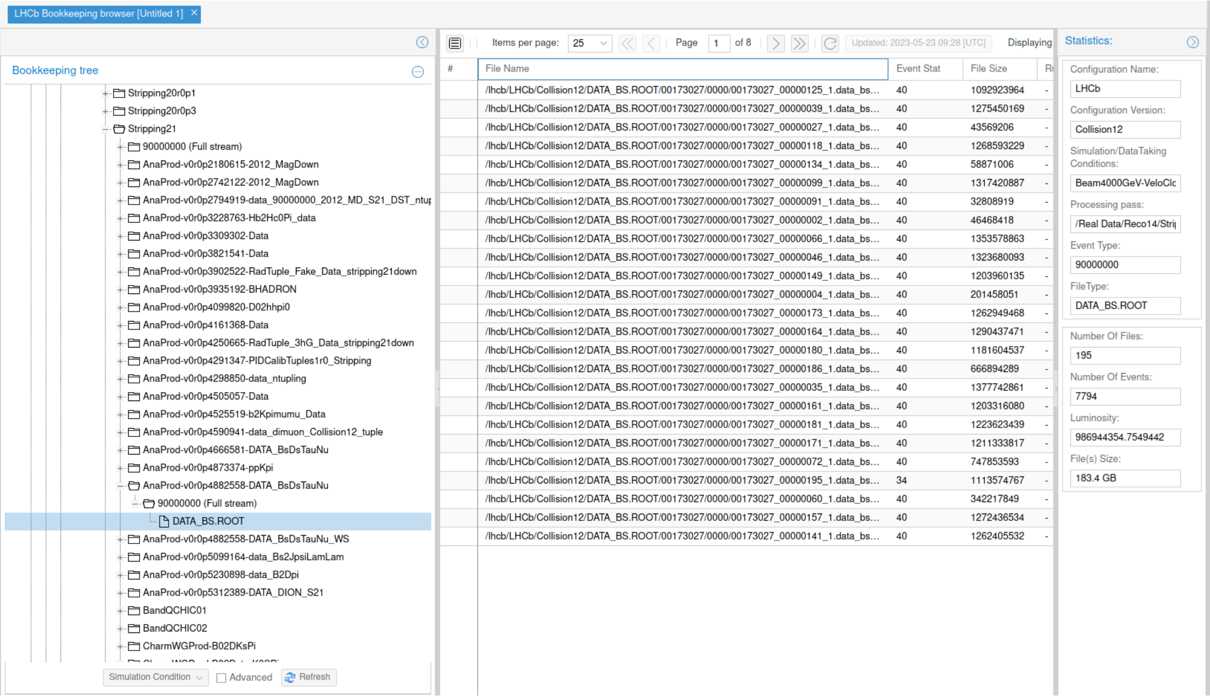
> /LHCb/Collision12/Beam4000GeV-VeloClosed-MagDown/Real Data/Reco14/Stripping21/
> 90000000/BHADRON.MDST

corresponds to physics data taken by the LHCb detector in 2012, with 4000 GeV energy per beam, with the VELO closed and the dipole magnet set so that the fields points downwards. It was then reconstructed using the Reco14 configuration and filtered using Stripping21.

Analysis Productions derived from data would be referenced by a bookkeeping path such as:

> /LHCb/Collision12/Beam4000GeV-VeloClosed-MagDown/Real Data/Reco14/Stripping21/
> AnaProd-v0r0p4882558-DATA_BsDsTauNu_WS/90000000/DATA_BS_WS.ROOT

From this path the same information is used as for the data this Analysis Production derives from: LHCb means that this is derived from data, Collision12 means physics data from 2012, Beam4000GeV-VeloClosed-MagDown corresponds to the data-taking conditions, Reco14/Stripping21 correspond to the reconstruction and filtering conditions and then follow the Analysis Production identifier (*AnaProd* plus the version which is internal to the system, followed by a string matching the name passed by the analyst). Figure 4.1 shows how this is presented in the LHCb bookkeeping system.



- Stripping20r0p1
- Stripping20r0p3
- Stripping21
  - 90000000 (Full stream)
  - AnaProd-v0r0p2180615-2012_MagDown
  - AnaProd-v0r0p2742122-2012_MagDown
  - AnaProd-v0r0p2794919-data_90000000_2012_MD_S21_DST_ntup
  - AnaProd-v0r0p3228763-Hb2Hc0Pi_data
  - AnaProd-v0r0p3309302-Data
  - AnaProd-v0r0p3821541-Data
  - AnaProd-v0r0p3902522-RadTuple_Fake_Data_stripping21down
  - AnaProd-v0r0p3935192-BHADRON
  - AnaProd-v0r0p4099820-D02hhpi0
  - AnaProd-v0r0p4161368-Data
  - AnaProd-v0r0p4250665-RadTuple_3hG_Data_stripping21down
  - AnaProd-v0r0p4291347-PIDCalibTuples1r0_Stripping
  - AnaProd-v0r0p4298850-data_ntupling
  - AnaProd-v0r0p4505057-Data
  - AnaProd-v0r0p4525519-b2Kpimumu_Data
  - AnaProd-v0r0p4590941-data_dimuon_Collision12_tuple
  - AnaProd-v0r0p4666581-DATA_BsDsTauNu
  - AnaProd-v0r0p4873374-ppKpi
  - AnaProd-v0r0p4882558-DATA_BsDsTauNu
    - 90000000 (Full stream)
      - DATA_BS.ROOT
  - AnaProd-v0r0p4882558-DATA_BsDsTauNu_WS
  - AnaProd-v0r0p5099164-data_Bs2JpsiLamLam
  - AnaProd-v0r0p5230898-data_B2Dpi
  - AnaProd-v0r0p5312389-DATA_DION_S21
  - BandQCHIC01
  - BandQCHIC02
  - CharmWGProd-B02DKsPi

Items per page: 25 | Page 1 of 8 | Updated: 2023-05-23 09:28 [UTC] | Displaying

#	File Name	Event Stat	File Size	R
	/lhcb/LHCb/Collision12/DATA_BS.ROOT/00173027/0000/00173027_00000125_1.data_bs...	40	1092923964	-
	/lhcb/LHCb/Collision12/DATA_BS.ROOT/00173027/0000/00173027_00000039_1.data_bs...	40	1275450169	-
	/lhcb/LHCb/Collision12/DATA_BS.ROOT/00173027/0000/00173027_00000027_1.data_bs...	40	43569206	-
	/lhcb/LHCb/Collision12/DATA_BS.ROOT/00173027/0000/00173027_00000118_1.data_bs...	40	1268593229	-
	/lhcb/LHCb/Collision12/DATA_BS.ROOT/00173027/0000/00173027_00000134_1.data_bs...	40	58871006	-
	/lhcb/LHCb/Collision12/DATA_BS.ROOT/00173027/0000/00173027_00000099_1.data_bs...	40	1317420887	-
	/lhcb/LHCb/Collision12/DATA_BS.ROOT/00173027/0000/00173027_00000091_1.data_bs...	40	32808919	-
	/lhcb/LHCb/Collision12/DATA_BS.ROOT/00173027/0000/00173027_00000002_1.data_bs...	40	46468418	-
	/lhcb/LHCb/Collision12/DATA_BS.ROOT/00173027/0000/00173027_00000066_1.data_bs...	40	1353578863	-
	/lhcb/LHCb/Collision12/DATA_BS.ROOT/00173027/0000/00173027_00000046_1.data_bs...	40	1323680093	-
	/lhcb/LHCb/Collision12/DATA_BS.ROOT/00173027/0000/00173027_00000149_1.data_bs...	40	1203960135	-
	/lhcb/LHCb/Collision12/DATA_BS.ROOT/00173027/0000/00173027_00000004_1.data_bs...	40	201458051	-
	/lhcb/LHCb/Collision12/DATA_BS.ROOT/00173027/0000/00173027_00000173_1.data_bs...	40	1262949468	-
	/lhcb/LHCb/Collision12/DATA_BS.ROOT/00173027/0000/00173027_00000164_1.data_bs...	40	1290437471	-
	/lhcb/LHCb/Collision12/DATA_BS.ROOT/00173027/0000/00173027_00000180_1.data_bs...	40	1181604537	-
	/lhcb/LHCb/Collision12/DATA_BS.ROOT/00173027/0000/00173027_00000186_1.data_bs...	40	666894289	-
	/lhcb/LHCb/Collision12/DATA_BS.ROOT/00173027/0000/00173027_00000035_1.data_bs...	40	1377742861	-
	/lhcb/LHCb/Collision12/DATA_BS.ROOT/00173027/0000/00173027_00000161_1.data_bs...	40	1203316080	-
	/lhcb/LHCb/Collision12/DATA_BS.ROOT/00173027/0000/00173027_00000181_1.data_bs...	40	1223623439	-
	/lhcb/LHCb/Collision12/DATA_BS.ROOT/00173027/0000/00173027_00000171_1.data_bs...	40	1211333817	-
	/lhcb/LHCb/Collision12/DATA_BS.ROOT/00173027/0000/00173027_00000072_1.data_bs...	40	747853593	-
	/lhcb/LHCb/Collision12/DATA_BS.ROOT/00173027/0000/00173027_00000195_1.data_bs...	34	1113574767	-
	/lhcb/LHCb/Collision12/DATA_BS.ROOT/00173027/0000/00173027_00000060_1.data_bs...	40	342217849	-
	/lhcb/LHCb/Collision12/DATA_BS.ROOT/00173027/0000/00173027_00000157_1.data_bs...	40	1272436534	-
	/lhcb/LHCb/Collision12/DATA_BS.ROOT/00173027/0000/00173027_00000141_1.data_bs...	40	1262405532	-

Statistics:

Configuration Name: LHCb
Configuration Version: Collision12
Simulation/DataTaking Conditions: Beam4000GeV-VeloClc
Processing pass: /Real Data/Reco14/Stri
Event Type: 90000000
FileType: DATA_BS.ROOT

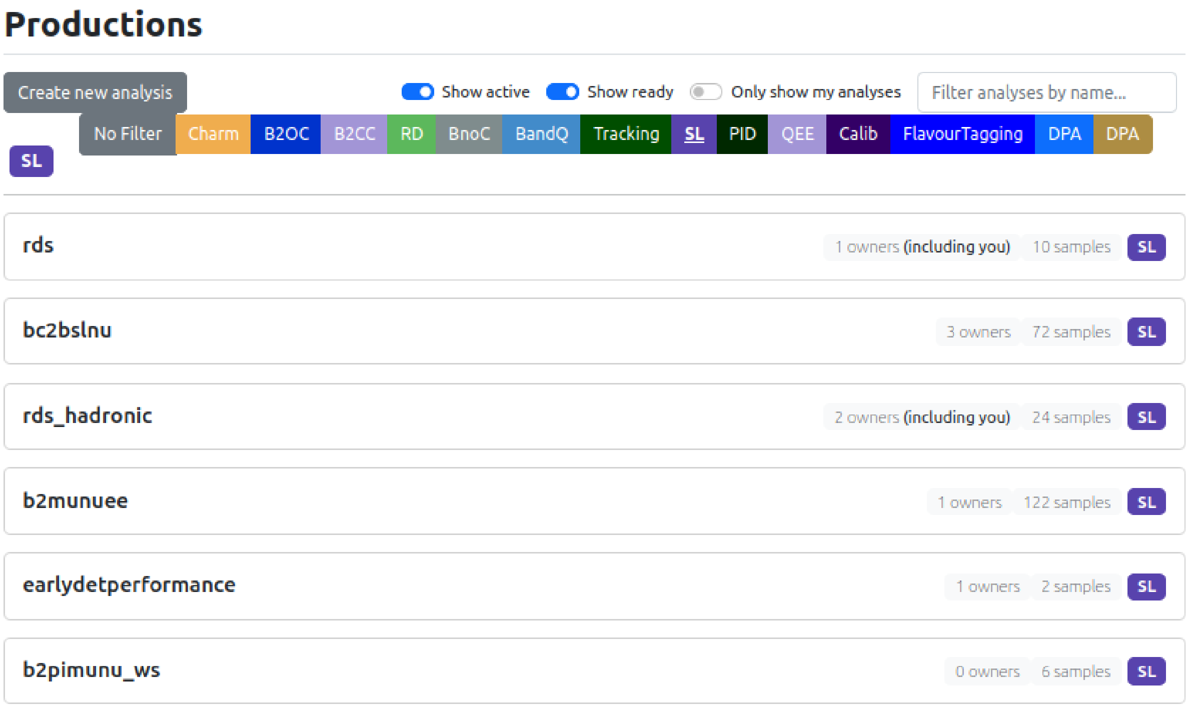
Number Of Files: 195
Number Of Events: 7794
Luminosity: 986944354.7549442
File(s) Size: 183.4 GB

Figure 4.1: Analysis Production data in the LHCb bookkeeping.

This view of the data is not very useful to understand what was actually run as it only references the Analysis Production as:

> AnaProd-v0r0p4882558-DATA_BsDsTauNu_WS

To understand more about the data, the dedicated Analysis Productions interface provides more information, as shown by Figure 4.2 which lists a number of Analysis Productions. This interface allows searching for productions by working group or by name, and provides a list of datasets, corresponding to a set of files recorded with the same conditions and processed in the same way as shown by Figure 4.3.



rds 1 owners (including you) 10 samples SL

bc2bslnu 3 owners 72 samples SL

rds_hadronic 2 owners (including you) 24 samples SL

b2munuee 1 owners 122 samples SL

earlydetperformance 1 owners 2 samples SL

b2pimunu_ws 0 owners 6 samples SL

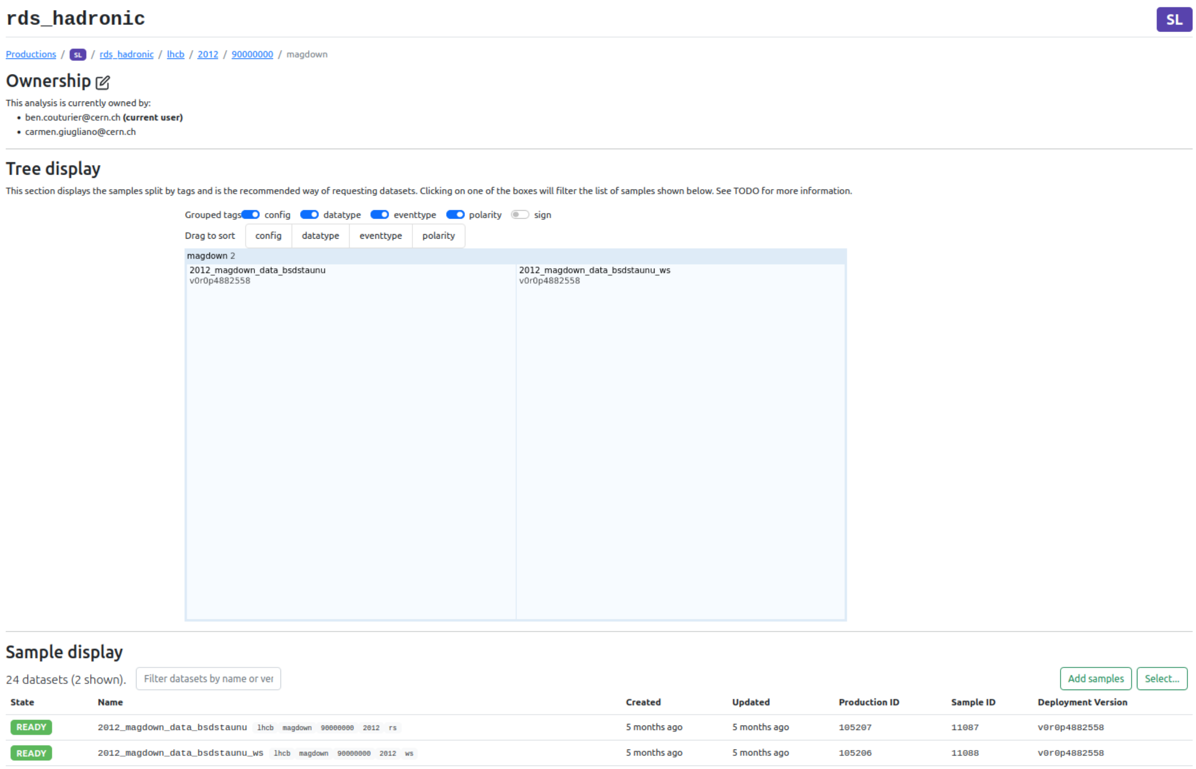
Figure 4.2: Analysis Production web application, list view.

For each sample, it is possible to know (see Figure 3.15):

- the size of the sample on disk
- the version of the analysis package, which allows finding the commit in git containing all the configuration for running the Analysis Production.
- The GitLab merge request with all information about this processing (like the application used, and its version, the configuration files, the list of steps needed to process the data and their identifier in the DIRAC system etc.) and the results of the continuous integration tests (such as the output file size an the expected output data size for the whole run, the memory used etc) run to validate this request.
- the names of the files produced: the Logical File Name (LFN) which is a generic identifier for the file, independent of its location on disk, as well as the Physical File Name (PFN) on EOS at CERN.

4.1.2 Adding metadata

The above information is enough to have the full provenance of the files listed, but also to reproduce them if necessary, as it references LHCb released applications available on CVMFS, and for which the environment is preserved. However, it is difficult to understand what type of events were selected and what type of data was extracted, as one would need to read the program configuration to understand this. It would therefore be useful to allow analysts to add metadata



rds_hadronic

Productions / SL / rds_hadronic / lhcb / 2012 / 90000000 / magdown

Ownership

This analysis is currently owned by:

- ben.couturier@cern.ch **(current user)**
- carmen.giugliano@cern.ch

Tree display

This section displays the samples split by tags and is the recommended way of requesting datasets. Clicking on one of the boxes will filter the list of samples shown below. See TODO for more information.

Grouped tags ● config ● datatype ● eventtype ● polarity ○ sign

Drag to sort: config | datatype | eventtype | polarity

magdown 2

2012_magdown_data_bsdstaunu
v0r0p4882558

2012_magdown_data_bsdstaunu_ws
v0r0p4882558

Sample display

24 datasets (2 shown). Filter datasets by name or ver

State	Name	Created	Updated	Production ID	Sample ID	Deployment Version
READY	2012_magdown_data_bsdstaunu lhcb magdown 90000000 2012 rs	5 months ago	5 months ago	105207	11087	v0r0p4882558
READY	2012_magdown_data_bsdstaunu_ws lhcb magdown 90000000 2012 ws	5 months ago	5 months ago	105206	11088	v0r0p4882558

Figure 4.3: View of the samples for *2012* data with magnet polarity *down* for the Analysis Production *rds_hadronic*.

as tags, i.e. key/value pairs, to the datasets created by the Analysis Productions system. This would make the data more Findable and Reusable, according to the FAIR principles.

Some of the metadata can be derived directly from the Analysis Production jobs because, as shown previously, the bookkeeping path of the produced files already contains information about the data taking conditions (or whether this is simulation) and about the processing. For instance it is easy to know whether the ntuples are derived from real or simulated data, to which data taking year they correspond, the centre of mass energy of the beam, whether the VELO was open or closed, the magnet polarity.

It is also possible to add extra tags which indicate what the data is used for in the analysis (e.g. evaluating errors, double checking resolution....). These tags can be set in the LHCb Analysis Productions web application shown in Figure 4.4.

The treemap view presented in Figure 4.4 allows visualising the datasets, grouping them by tags. This is a useful tool to check what data is linked to a specific analysis, how many datasets have which tags, etc. Furthermore, it is important to add safeguard to make sure analysts don't process the same input files twice (e.g. when two versions of the processing for the same events are available). In this case tags have to be added to differentiate the two otherwise the python processing tools will return an error, as will be discussed in Section 4.2.

Once this is done, it is possible to identify the files by specifying a pair working group/analysis name plus a list of tags names/values instead of specifying a list of files. The `apd` python package

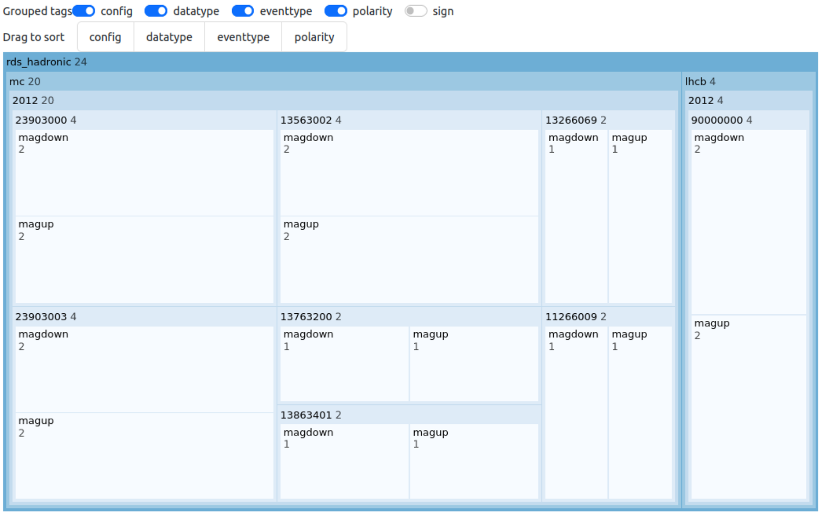


Figure 4.4: Visualisation of the data for an analysis.

allows selecting the data in this fashion, as described in Section 4.2. This has major advantages:

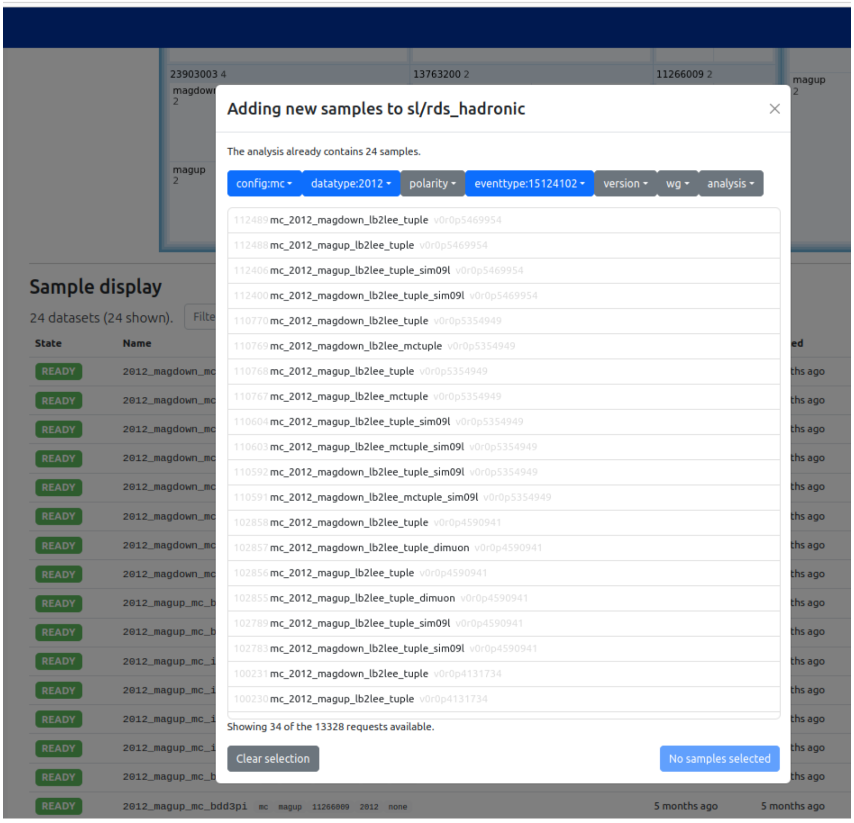
- analysts can locate their data using tags that represent their content instead of hard-coding file lists. This is less error prone, but also allows for easier sharing of data between analysts (provided the naming of the tags is clear enough and there is enough commonality in the naming throughout the experiment).
- the correctness of the tags is better than for metadata added a posteriori: in this case the tags are used to select data and analysts will notice quickly if there are issues with the characterisation.
- when the analysis is published, the tags of the Analysis Production can be used directly to create the metadata in the CERN analysis preservation portal.

4.1.3 Automating the metadata extraction

Extracting metadata from Analysis Production jobs is possible, given that their configuration is committed to a Git repository and the environment to re-run them is available. This is however not a very simple task as the configuration of the DaVinci application used to process event data is not very easy to parse. It is however possible to find which decay was matched, and the path of the input data loaded. In conjunction with the stripping/sprucing project configuration, this allows retrieving metadata that can be added as tags on the AP samples.

4.1.4 Sharing datasets between analyses

Sharing data between analyses is necessary but also very difficult to track: it was commonly done by sharing file locations in storage system and using them as is or performing copies. This is error prone: for example, in case the team sharing the data realises there was problem in the data and recreates the files but forgets to inform the other team using the old files. Analysis Productions can also help in this case: with a specific analysis it is possible to reference files produced for another analysis, using the web interface and selecting the samples as per figure 4.5.



Adding new samples to sl/rds_hadronic

The analysis already contains 24 samples.

112489 mc_2012_magdown_lb2lee_tuple v0r0p5469954
112488 mc_2012_magup_lb2lee_tuple v0r0p5469954
112406 mc_2012_magup_lb2lee_tuple_sim09l v0r0p5469954
112400 mc_2012_magdown_lb2lee_tuple_sim09l v0r0p5469954
110770 mc_2012_magdown_lb2lee_tuple v0r0p5354949
110769 mc_2012_magdown_lb2lee_mctuple v0r0p5354949
110768 mc_2012_magup_lb2lee_tuple v0r0p5354949
110767 mc_2012_magup_lb2lee_mctuple v0r0p5354949
110604 mc_2012_magup_lb2lee_tuple_sim09l v0r0p5354949
110603 mc_2012_magup_lb2lee_mctuple_sim09l v0r0p5354949
110592 mc_2012_magdown_lb2lee_tuple_sim09l v0r0p5354949
110591 mc_2012_magdown_lb2lee_mctuple_sim09l v0r0p5354949
102858 mc_2012_magdown_lb2lee_tuple v0r0p4590941
102857 mc_2012_magdown_lb2lee_tuple_dimuon v0r0p4590941
102856 mc_2012_magup_lb2lee_tuple v0r0p4590941
102855 mc_2012_magup_lb2lee_tuple_dimuon v0r0p4590941
102789 mc_2012_magup_lb2lee_tuple_sim09l v0r0p4590941
102783 mc_2012_magdown_lb2lee_tuple_sim09l v0r0p4590941
100231 mc_2012_magdown_lb2lee_tuple v0r0p4131734
100230 mc_2012_magup_lb2lee_tuple v0r0p4131734

Showing 34 of the 13328 requests available.

Figure 4.5: Adding samples already produced by others to an existing analysis in the Analysis Production web application.

This keeps a record in the bookkeeping database of which analysis uses which datasets. This reference count can be used to prevent archiving datasets when used for at least one analysis.

4.1.5 Conclusion

The Analysis Production application allows physicists to add metadata to the available LHCb datasets. To be useful this metadata must be easily accessible from the tools used to analyse the data. We therefore show in chapter 4.2 how a python interface to the Analysis Productions database was implemented for this purpose.

4.2 The `apd` python package

To be useful in the context of data analysis, a programming interface has to be defined to query the files and specify which ones to process. As well as a graphical user interface, the Analysis Productions application proposes a RESTful [158] interface. This architecture can be queried easily from any programming language, as it simply uses the HTTP protocol and the URLs to identify uniquely the objects they serve (and therefore always return the same content). They therefore can be replaced by static websites for long term preservation.

The Python language [43] is very popular amongst LHCb analysts as by its frequency of use in the preserved repositories in Figure 3.14, and it is therefore a good candidate to provide a client interface: this is the goal of the Analysis Production Data (apd) python package.

`apd` provides a simple interface to the Analysis Productions server: it is a simple python package published to standard repositories (pypi.org) and conda-forge [159]. It is installed in the default LHCb analysis environment, has few dependencies and can be installed on any machine independently of the other LHCb software applications. Using `apd` from python scripts requires minimal effort from the analysts.

```
from apd import get_analysis_data
dataset = get_analysis_data("sl", "rds_hadronic")
files = dataset(config="lhcb", datatype="2012", polarity="magdown",
↪  eventtype="90000000", sign="rs")
```

Listing 4: Retrieving a list of files for the SL/RDs_hadronic analysis for data for polarity magdown and the sample with “right-sign” decay.

In order to avoid processing samples that are derived from the same input data with different configurations, which would be problematic, some safeguards have been put in place: `apd` checks that this is not the case and throws an exception if it is. Listing 5 illustrates this case.

Application Programming Interfaces (API) have not yet been written for other languages, such as C++ for ROOT macros. REST interfaces are however easy to develop, and it would be easy to do if needed. In the meantime, it is easy to invoke the python API and save the list of files to CSV or JSON format for consumption from other languages.

In practice, if using Snakemake as recommended, dealing with the list of files to process should be done by the workflow: workflow steps in C++ should be coded to take one or several files as input, and write their output where specified by Snakemake. This simplifies the integration with `apd` without limiting the functionality. Furthermore, experience with the analysis presented in Chapter 6 show that it is very easy to either use root to either run the scripts directly, or to compile them using the compiler available in the environment. Both cases have been tested successfully, leaving the analysts the choice on how to organise their code.

In conclusion, the `apd` python package is an abstraction layer that allows analysis code to specify the data to process by a series of meaningful tags, instead of hard-coding file lists. This is necessary but not sufficient to allow tracking the provenance of the results: the integration with a workflow tool can help further in that direction.

4.3 Dataset location abstraction and data management

The abstraction layer provided by the `apd` package as described in Section 4.2 is also useful to decouple the analysis scripts from the storage system containing the data: `apd` allows

```
from apd import get_analysis_data
dataset = get_analysis_data("sl", "rds_hadronic")
dataset(config="lhcb", polarity="magdown", eventtype="90000000",
↪  datatype="2012")
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
  File "/cvmfs/lhcbdev.cern.ch/conda/envs/default/2023-04-26_20-20/linux-64/"
       "lib/python3.10/site-packages/apd/analysis_data.py", line 391, in
        ↪  __call__
    raise ValueError("Error loading data: " + error_txt)
ValueError: Error loading data: 1 problem(s) found
{'config': 'lhcb', 'polarity': 'magdown', 'eventtype': '90000000', 'datatype':
↪  '2012'}:
2 samples for the same configuration found, this is ambiguous:
     {'config': 'lhcb', 'polarity': 'magdown', 'eventtype': '90000000',
     ↪  'datatype': '2012', 'sign': 'rs',
     'version': 'v0r0p4882558', 'name': '2012_magdown_data_bsdstaunu',
     ↪  'state': 'ready'}
     {'config': 'lhcb', 'polarity': 'magdown', 'eventtype': '90000000',
     ↪  'datatype': '2012', 'sign': 'ws',
     'version': 'v0r0p4882558', 'name': '2012_magdown_data_bsdstaunu_ws',
     ↪  'state': 'ready'}
```

Listing 5: Retrieving data with a list of tags selecting multiple datasets.

downloading datasets locally and subsequently using the local version, without modifying the analysis programs. This is useful to run on a personal machine with datasets of limited size.

This mechanism could also be extended to allow configuring the storage system used to load the data without affecting the analysis workflows, for example to run on High Performance Computing systems or commercial clouds.

4.4 Snakemake interface

We choose to provide an interface to the Snakemake workflow engine [134], as it is adapted to data analysis and already popular among LHCb physicists, as shown by the study of preserved analysis repositories (see Figure 3.14). Furthermore, this is a Python based tool and therefore allows using the code described previously. We propose an integration that eases the use of ntuples created by LHCb Analysis Productions within the workflows, following several axes:

- an API that integrates easily with the Snakemake *wildcards*, a key feature of this workflow that allows for generalisation of the rules: while wildcards apply to patterns in the naming of inputs or outputs of rules, they can be used to query LHCb data, as will be explained in listing 6.
- Snakemake needs to track changes in input and output files to only re-run the necessary

rules. By default, it works with local files but also has the notion of *remote files*, than can be access with different protocols. A Snakemake remote for the `xrootd` protocol[160] was implemented and used to access the data for analysis in LHCb. Utility methods in apd wrap the filenames with `xrootd` remote in order for the workflow to work transparently.

- LHCb data is not open for public access, and credentials are necessary to interact with the data storage system. For interactive session, normal user credentials can be used (grid proxies or kerberos tokens). In the case of continuous integration systems such as GitLab installed at CERN, `apd` allows using storage tokens for the EOS[161] system at CERN where data are stored.

4.4.1 Generalising data processing with wildcards

The example below shows how easy it is to extend data processing of LHCb datasets using wildcards. The Listing 6 demonstrates a rule that takes a LHCb dataset as input, instead of a local file. Snakemake rules are defined by:

- their output: here a file called "bmass_config_datatype_eventtype_polarity.root" where config, datatype, eventtype, polarity are called *wildcards*. When requested for a file whose name is matching this pattern e.g. bmass_mc_2012_13266069_magdown.root, Snakemake will deduce the value of each wildcard and use it to select the input data and pass it as argument to the code run within the rule.
- their input: here a function that invokes an `apd` dataset object that will query the Analysis Production database for a list of files.
- the actual code to produce the output based on the input

Once the rule *create_histo* has been defined, listing 7 shows a rule that triggers the build of histograms for Analysis Production data for Monte-Carlo simulation for the year 2012, with a separate file for each LHCb magnet polarity. It is only required to ask Snakemake to produce the files with those combinations of wildcards, and the processing follows.

4.4.2 Workflows and provenance tracking

Of course, introducing an extra tool such as the Snakemake workflow engine, is an extra burden for data analysts. This is alleviated by the available tutorial as part of the LHCb Starterkit, and examples already available in the collaboration. This extra burden is however quickly compensated by the benefits of easily tracking the processing done, and of being able to re-run the whole chain when necessary (new input data etc).

In the case of the Hadronic $R(D_s)$ measurement performed in this thesis, having workflows to define the processing was very useful. As several steps in the processing are needed (correction of the PID information, running datasets corresponding to several decision trees on the existing events and using their output to filter the data), a Snakemake workflow was developed to enrich and filter the Analysis Production output. After the first data extraction from LHCb files, new

```
from apd.snakemake import get_analysis_data
dataset = get_analysis_data("sl", "rds_hadronic")

rule create_histo:
   input:
       data=lambda w: dataset(config=w.config, datatype=w.datatype,
                              eventtype=w.eventtype, polarity=w.polarity)
   output: f"bmass_{{config}}_{{datatype}}_{{eventtype}}_{{polarity}}.root"
   run:
       import ROOT
       inputfiles =  [ f for f in input ]
       f = ROOT.TFile.Open(output[0], "RECREATE")
       rdf = ROOT.RDataFrame("SignalTuple/DecayTree", input)
       h = rdf.Histo1D(("BM_Hist","BM_Hist", 200, 0., 25e3), "B_M")
       h.Write()
       f.Close()
```

Listing 6: Snakemake rule that creates histograms for a specific tag combination.

```
rule needed_histograms:
   input:
       "bmass_mc_2012_13266069_magdown.root",
       "bmass_mc_2012_13266069_magup.root"
```

Listing 7: Snakemake rule triggering the creation of histograms for both magnet combinations.

files were generated for a new event type, and performing this processing on the new data just meant re-running the workflow that used the apd output to identify some work needed to be done for the new datasets. This tool will also make the processing of samples for other years than 2012 very easy as well.

Using Snakemake and wildcards enforces the use of naming conventions for the output files (as the wildcards should be present in the name). While this may appear as a constraint, it makes understanding the content and provenance of the files easier. In any case workflow tools inherently track the dependency graph between rules, and normally allow exporting them to machine readable format and viewing them. The Snakemake workflow engine allows exporting the workflow to the GraphViz *dot* format[162] for visualisation[163]. Figure 4.6 shows the diagram for the production of one file containing the histograms produced by the rule mentioned above, for both polarities of the LHCb magnet and two different types of simulated events.

Snakemake also allows exporting the workflow to the Common Workflow Language (CWL) [164]. This opens the door to the visualisation of the dependencies between files, the processing, or re-processing of the data using other tools or workflow engines. For example, the REANA system [165] allows processing CWL workflows, and ensures the long term preservability of the workflows that have been created.

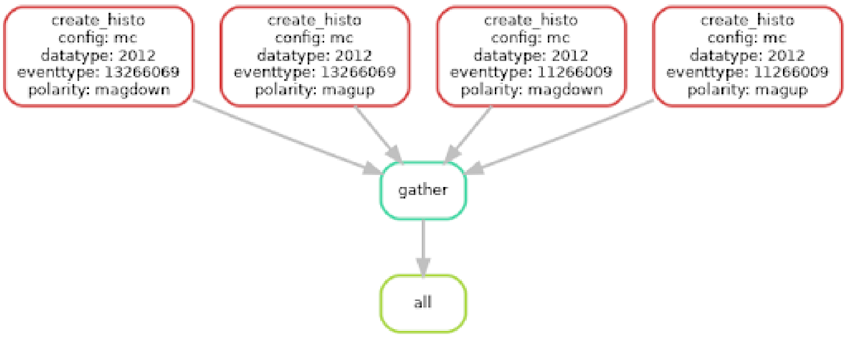


Figure 4.6: Dependency graph to apply the create_histo rule to 2 event types for both LHCb magnet polarities.

4.4.3 Continuous integration and security considerations

It has long been a known fact in Software Engineering that automating software tests, and running them as soon as changes are done to the code base (a practice known as Continuous Integration [166]), is the best way to ensure the quality of the software being delivered. Data analyses use a lot of custom of software tools and typical Software Engineering tools can therefore be used to ensure the quality of the software chain used to produce the results.

In this approach:

- all the files related to the analysis, except the data files, are versioned using a Source Control System. In our case, Git[49] is used.
- A common Git repository for each analysis is stored using the GitLab platform. This allows for coordination between the analysts.
- GitLab features a continuous integration system (CI): for each change, or requested change, to the codebase, GitLab can run a series of tests defined in the repository (in a file named `.gitlab-ci.yml`)

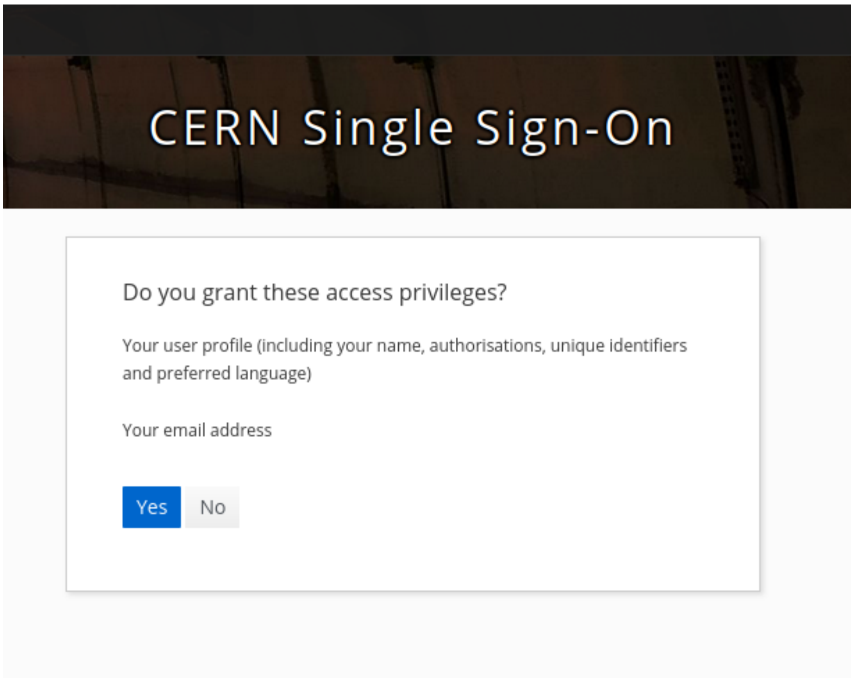
The various steps of the analysis can therefore be run as part of the CI system, but for this to happen, GitLab requires:

- an environment with the software needed to run the jobs: in our case at least python, `apd` and Snakemake plus software to access and analyse the data: ROOT, Scipy, Scikit-learn... This is very analysis-dependent and a wide range of tools should be provided.
- credentials to access the data on behalf of the user.

Providing environments for LHCb analysis has already been implemented and is described in section 4.5. Here we focus on the credentials part which was not easy to solve. The GitLab CI job should be able to read and write data on behalf of the user requesting the changes. When running interactively, the user needs credentials to access the Analysis Productions (AP) database, and to access the data.

Interactive access

Access to the AP database is provided by the **apd-login** command. This command uses the CERN Single-Sign-On system[167], and a standard OAuth2[168] workflow to validate the user's CERN account (available to each member of the LHCb collaboration). After invoking the command, users have to connect to a specific application hosted at CERN and enter their credentials, and have to agree to share their email as shown in Figure 4.7.



CERN Single Sign-On

Do you grant these access privileges?

Your user profile (including your name, authorisations, unique identifiers and preferred language)

Your email address

Figure 4.7: CERN Single Sign On application after credentials are entered.

Once that is done, the CERN SSO grants credentials to the apd-login command which stores them in a local directory for further use. Listing 8 shows a successful log-in session.

```
$ apd-login
CERN SINGLE SIGN-ON

On your tablet, phone or computer, go to:
https://auth.cern.ch/auth/realms/cern/device
and enter the following code:
ABCD-EFGH

You may also open the following link directly and follow the instructions:
https://auth.cern.ch/auth/realms/cern/device?user_code=ABCD-EFGH

Waiting for login...
Login successful as ben
```

Listing 8: Logging using apd-login

Access to the data is done using the classic credential systems used by the WLCG grid or by EOS:

- Grid credentials, i.e. X509 proxies in the case of WLCG credentials,
- Kerberos credentials from the CERN Kerberos server are also supported by EOS at CERN.

Access from the continuous integration system

Access from the continuous integration system cannot follow the same flow: manually intervention from the users is not possible in that workflow. However, when starting a CI job, the server provides it with a JSON Web token [169] which can be used to authenticate the job itself[1]. The AP application then validates the token against the GitLab server and provides the correct credentials to access the AP database.

Giving access to the files is a trickier issue. Delegating users kerberos credentials is possible using so-called keytab files. This is however not desirable for several reasons:

- The GitLab CI job has in this case the same access right as the users: it can potentially access all their private data stored in the CERN shared files systems.
- keytab files are valid until the user changes password, as a new key has to be redone then: if a third party gets hold of this file, they have the same access as the users until a password change.
- the keytab file has to be added to the GitLab configuration, and re-updated every time the users change password

This is definitely not an ideal way to delegate credentials to the CI jobs. One method to mitigate the security implications of providing the keytab file for a specific user is to create a dedicated account (called service account in the CERN infrastructure), with limited access to the LHCb data. If the credentials are stolen from the CI job, the security impact is much more limited. This is however still not an ideal solution as the accounts have to be managed, and the credentials still do not provide fine grained access control.

To remedy this situation, the use of EOS tokens [161, 151] was implemented alongside apd. EOS tokens allow for fine-grained access to the EOS filesystem in which AP files are stored: it is possible to give access to a single file, or a whole directory, in read-only or read-write mode, for a given duration. To be used by EOS, the tokens have to be appended to the file URL as per:

root://eoslhcb.cern.ch//eos/lhcb/FILE.root?xrd.wantprot=unix&authz=⟨eostoken⟩

As each token has limited capabilities (read-only/read-write, for a specific path), each CI job needs to be provided with several tokens and to use them appropriately. `apd` therefore provides the logic to do this:

[1] The JSON Web Token provided by GitLab to its job is stored in the environment as the variable CI_JOB_JWT_V2. If this variable is defined apd-login forwards to the AP application for authentication.

- `apd-login` provides EOS tokens to the CI job, for directories specified in a GitLab project per GitLab project configuration.
- the `apd` package manages the list of tokens and provides two methods to query the token list and wrap the file names being accessed with the appropriate token (or raise an exception if none is found):
  - `auth()` appends a read-only token to the file URL and returns it
  - `authw()` appends a read-write token to the file URL and returns it

Of course this is not transparent as the analysis code has to be modified to wrap all URLs with the auth() or authw() methods, but this is normally not a big issue. In practice `apd` can wrap directly the files read from storage (as they are read-only anyway).

Tracking the metadata evolution

Throughout an analysis, it may be needed to update the metadata associated with some files. This can of course lead to confusion, as the result of the same query for files with given tags depends on the time. For this reason, the database schema storing the data is versioned and allow retrieving the metadata for any date in the past.

Allowing to specify a date when querying the `apd` python package is not yet possible, but this functionality will be added when needed by analyses.

4.5 LHCb Analysis environments

4.5.1 Containers

The LHCb Analysis Preservation roadmap[94] stresses that as part of the best practices, one should be:

> "Capturing the run-time environment, for example through a Docker container image."

In order to make this easier, a container template is available for physicists, as a GitLab project[170].

As discussed in Section 3.2.1, there are several ways to provide virtual environments with containers. Docker containers have many advantages but also drawbacks. For example, it is not possible to run docker containers on the cluster available at CERN for interactive analysis (lxplus). Other containerisation systems can be used (Apptainer[122] for example) but preparing and distributing the system images remains an effort. To help with this issue, a CVMFS repository (*unpacked.cern.ch* is in place. It allows users to deploy their container images to the CVMFS repository */cvmfs/lhcb.cern.ch*. On machines with CVMFS installed, it allows using those images with Apptainer.

Apptainer has become the main container system to run production jobs on the WLCG grid while Docker is used in some contexts (e.g. for GitLab continuous integration jobs). It is therefore necessary to provide images for several container system, depending on the use cases. In practice, as most analysts require the same set of tools, it is possible to provide environments distributed via the CernVM File System (CVMFS) that satisfy most users as explained in the next Section.

4.5.2 Distributing software on CVMFS

Preparing environments for running analyses jobs, and deploying adequately is critical to running them efficiently in batch clusters, or on the WLCG. Analyses require many software packages and require more flexibility than centralised processing of data: they require packages specific to High Energy Physics such as ROOT[45], but also standard tools from the python ecosystem (numpy, pandas, scikit-learn...) or dedicated to high energy physics (ScikitHEP[171]). In practice, most analyses make use of the same packages and a default environment is sufficient for nearly all users provided there is flexibility to add packages relatively quickly.

We therefore decided to prepare environments using the Conda tool (from the Anaconda distribution [172]) installing packages from Conda-forge [159], and deploying them using the CernVM file system (CVMFS) [173] which is extensively used by the LHCb distributed computing system, so that they can easily be used during the analysis, and preserved afterwards. This approach is presented in Ref. [174] and the `lb-conda` allows using those environments. In this system, each environment is defined by a list of packages, and their versions, making it easy to recreate it, if needed, based on this definition. This requires of course the package repository (in our case conda-forge) to be available and accessible. Preparing a system for preservation is easy as the base operating system is known. While the long term durability of the current infrastructure is not guaranteed, enough information has however been recorded to reproduce the environment for preservation.

4.5.3 Containers or metadata preservation?

Containers and virtual images are a popular way to preserve complete system environments. However, they include all the files needed to run the system and they can therefore be quite large (in the order of gigabytes). Furthermore, within those images, it is not always possible to separate the base system from what is really needed to run the analysis software.

To avoid these issues, LHCb therefore proposes a lightweight way to track software requirements, which integrates with the current distributed systems available: data analysts are encouraged to use software environments prepared with Conda (see Section 4.5.2), instead of building containers for preservation. This approach keeps track of the list of tools via the *environment.yaml* file used by Conda, and it is easy to recreate an environment from scratch provided access to the conda-forge repository is available.

This does not exclude preparing and storing containers or virtual machines based on the environment definition if necessary.

4.6 Conclusion

Data and analysis preservation are crucial but are not a new challenge to researchers. Many tools are available but the issue is to find the best practices that make the preservation easy: preservation is an active process that starts at the beginning of the analysis. This thesis introduces tools that should make it easier for physicists to analyse the data while helping with keeping track of data provenance: the `apd` python package, its Snakemake integration and the integration with EOS tokens for authentication was developed in the course of this thesis, and were presented at the CHEP 2023 conference. They were prototyped and tested in the course of the test of Lepton Flavour Universality presented in the next chapters. The first version of `apd`, released with minimum functionality, was nonetheless embraced by LHCb analysts and is experiencing a rapid uptake.

Chapter 5

Lepton flavour universality

5.1 Standard Model

For centuries, physicists have been trying to understand the forces that rule our Universe. Historically, gravitation was the first to be described. Electric and magnetic phenomena, unified by J. C. Maxwell [175], followed suit. This led the way for the probing of matter itself during the 20^{th} century: radioactivity, cosmic ray studies and particle accelerator experiments allowed studying the atomic nucleus themselves and discovered many new particles. Quantum mechanics, devised to explain the behaviour of particles, was very successful but could not be applied to fields. Quantum Field Theory originated in the 1920s, with the goal to resolve this issue and over the years was compared to experimental results. Quantum ElectroDynamics (QED) describes all electromagnetic interactions. The weak interaction, at the origin of nuclear β decay, was unified with with QED to produce the electroweak theory [176, 177, 178]. Together with Quantum ChromoDynamics (QCD) which explains the Strong force, binding quarks together in the nucleons, and binding nucleons in the nucleus, they form the Standard Model of Particle Physics (SM), the most accurate quantum field theory describing the fundamental particles and their interactions. Figure 5.1 lists the fundamental particles in the model:

- Spin 1 bosons are force carriers and follow the Bose-Einstein statistics. There are 12 of them, the photon for the electromagnetic force, the W^+, W^-, Z^0 for the weak force and 8 gluons for the strong force.

- Spin $\frac{1}{2}$ fermions constitute matter. They count leptons interacting through the gravitational, weak and electromagnetic forces (when charged) and quarks, subjected to the strong force in addition. They can be grouped in three families of increasing mass.

- The Higgs boson has spin 0. It was discovered at CERN in 2012 [12, 13]. It is produced by an excitation of the Brout-Engler-Higgs field. The interaction of the other particles with this field gives rise to their mass.

The Standard Model (SM) is a quantum field theory [179] invariant under transformations of three gauge groups. Each group is associated with spin 1 vector bosons: $U(1)_Y$ is associated

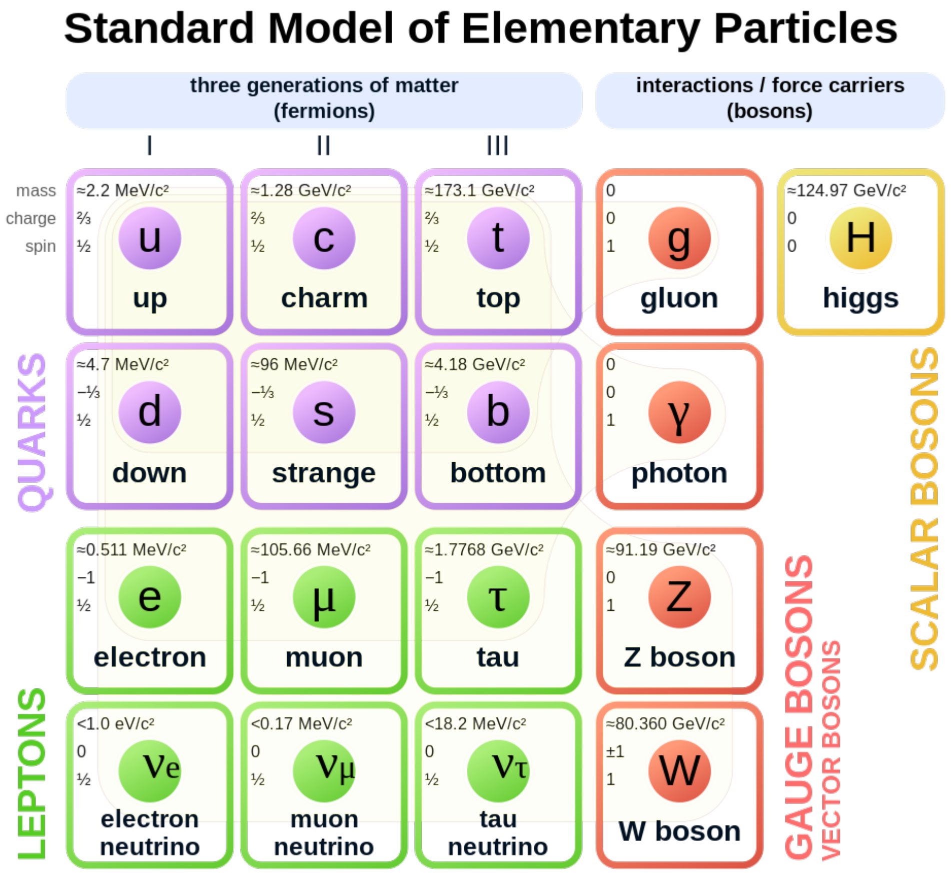


Figure 5.1: Standard Model particles

to B_μ which interacts with particles with weak hypercharge Y, $SU(2)_I$ to $W_\mu^{1,2,3}$ which interacts with fermions carrying weak isospin I. $SU(3)_C$ gives rise to the eight $G_\mu^{1,\ldots,8}$ vector fields for the strong force. In total there are 12 vector fields associated with three gauge symmetries, all of which can be summarised by the tensor product of groups that defines the gauge symmetry of the Standard Model:

$$SU(3)_C \otimes SU(2)_I \otimes U(1)_Y \, . \tag{5.1}$$

The $SU(2)_I \otimes U(1)_Y$ term from Eq. 5.1 represents the electroweak force, the unification of the electromagnetic and weak interactions proposed by S. Glashow, A. Salam and S. Weinberg [176, 177, 178]. The third component of I, called I_3 is always conserved and is linked to the weak hypercharge by the Gell-Mann-Nishima relation:

$$Y = 2(Q - I_3) \, , \tag{5.2}$$

Q being the electric charge. At low energy scale three of these massless fields acquire mass with the Higgs mechanism and the observable electroweak force mediators are a combination of the four gauge bosons:

$$W^{\pm} = \frac{1}{\sqrt{2}}(W^1 \pm W^2) \tag{5.3}$$

$$\begin{pmatrix} \gamma \\ Z \end{pmatrix} = \begin{pmatrix} \cos\,\theta_w & \sin\,\theta_w \\ -\sin\,\theta_w & \cos\,\theta_w \end{pmatrix} \begin{pmatrix} B \\ W^3 \end{pmatrix}, \tag{5.4}$$

where θ_w is the electroweak mixing angle. Strong and Electromagnetic interactions do not change the flavour of the fermions, however the weak interaction does not couple to the mass eigenstates, but to a linear superposition of these as described by the Cabibbo–Kobayashi–Maskawa (CKM) matrix [180]:

$$\begin{pmatrix} d' \\ s' \\ b' \end{pmatrix} = \begin{pmatrix} V_{ud} & V_{us} & V_{ub} \\ V_{cd} & V_{cs} & V_{cb} \\ V_{td} & V_{ts} & V_{tb} \end{pmatrix} \begin{pmatrix} d \\ s \\ b \end{pmatrix} \tag{5.5}$$

where q' (d', s' or b') is the weak eigenstate of mass eigenstate q (d, s or b), and the V_{ij} coefficients are complex numbers, which have to be determined experimentally and represent the relative strength of the interaction for the transition i to j. The CKM matrix is unitary and can be parametrised by three angles and a phase, which is a source of CP-violation in the standard model. According to the Particle Data Group (PDG), which summarises the most up-to-date knowledge in the domain of Particle Physics [181], the amplitude of the CKM matrix elements (mixing parameters) are:

$$|V_{\mathrm{CKM}}| = \begin{pmatrix} 0.97435 \pm .00016 & 0.22500 \pm 0.00067 & 0.00369 \pm 0.00011 \\ 0.22486 \pm 0.00067 & 0.97349 \pm 0.00016 & 0.04182^{+0.00085}_{-0.00074} \\ 0.00857^{+0.00020}_{-0.00018} & 0.04110^{+0.00083}_{-0.00072} & 0.999118^{+0.000031}_{-0.000036} \end{pmatrix} \tag{5.6}$$

It is nearly diagonal and the measured phase $\delta = 1.144 \pm 0.027$ [181] is not enough to explain the observed asymmetry in the universe.

For neutral leptons, an analogous mixing matrix is defined to explain the experimental evidence for neutrino mixing: the PMNS matrix from Pontecorvo, Maki, Nakagawa and Sakata [182, 183, 184]. This is however different from the CKM matrix, as the relative amplitude of the elements is much more homogeneous as shown by Figure 5.2.

The Standard Model has been extremely effective at describing the measurements performed at particle physics experiments since its inception. However, it does not account for the observed matter-antimatter asymmetry in the Universe, nor provides a Dark Matter candidate. Furthermore, it does not explain its own gauge group structure, the charge assignment of the fermions or the mass hierarchy between the different families. There are clues that a more global theory extending the SM at higher energies and shorter distances could resolve those issues. Investigating the limits of the SM and searching for New Physics (NP) is currently a prioritary objective of particle physics experiments.

There are two methods to perform this search: the first one, dubbed ”direct search”, tries to directly produce the new particles and identify their decay products. This was the method used

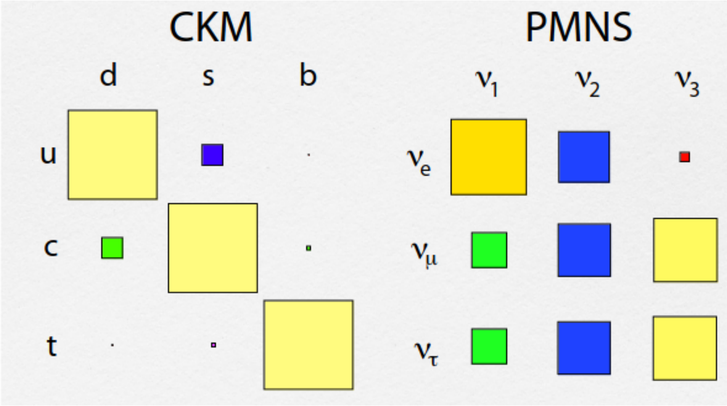


Figure 5.2: Relative amplitude of CKM and PMNS matrix elements from Ref. [185].

to produce and detect the Higgs boson at the LHC. Of course, the centre of mass energy of the colliding beams limits the mass of the particles produced, and one needs to be able to isolate the decay products of the sought-after particle. The second method, called ”indirect search”, exploits the presence of virtual states in the decays of SM particles, and the fact that this affects the observables of specific decay modes, such as the branching ratio, the angular distribution of the decay products or whether they violate symmetries. This method was used to prove the existence of weak neutral currents in the Gargamelle neutrino experiment at CERN in 1973 [9] before the Z^0 boson was produced and identified in the UA1 and UA2 experiments.

According to the Standard Model, the carriers of the electroweak force (the γ, W^+, W^- and Z^0) couple in the same manner with the three lepton generations (e, μ and τ). This prediction, called Lepton Flavour Universality (LFU), is at the core of the SM and any deviation would be a sign that virtual particles not yet included in the model are involved in the decays.

5.2 Lepton Flavour Universality

In the Standard Model, the three generations of fermions have the same weak isospin and electric charge, leading to the same coupling, hence the notion of universality. Furthermore, the coupling with the electroweak fields is not affected by the Higgs mechanism for the breakdown of the electroweak gauge symmetry. Therefore, the only difference between the three generations is the mass (the Yukawa interactions between the Higgs field and the fermion field).

The best way to test the lepton flavour universality is to measure ratios of decay rates or branching fractions to final states with leptons of different generations but with the same quark transition. From a theoretical point of view, this has the advantage to keep equal all other factors (due to phase-space, form factors, etc.) and provide precise predictions. From an experimental point of view, this also has the advantage to reduce the uncertainties linked to the final state reconstruction.

5.2.1 Electroweak sector

Tests using Z boson decays

Z^0 boson decays provide a way to test LFU very precisely, from a theoretical point of view as well as experimentally. Measurement have been performed both at e^+e^- (LEP and SLC see ref. [186]), $p\bar{p}$ (Tevatron [187]) and pp (LHC) colliders, and averaged by the PDG [181]:

$$\frac{\Gamma_{Z\to\mu^+\mu^-}}{\Gamma_{Z\to e^+e^-}} = 1.0009 \pm 0.0028\,,$$

$$\frac{\Gamma_{Z\to\tau^+\tau^-}}{\Gamma_{Z\to e^+e^-}} = 1.0019 \pm 0.0032\,,$$

in perfect agreement with the SM predictions that the decay widths are the same assuming negligible lepton masses.

Tests using W boson decays

$W^- \to \ell^-\bar{\nu}_\ell$ decays depend on the coupling g_ℓ, which is identical in the SM for all the three lepton families. Like in the previous case, the ratio of the decay rates to final states with different leptons tests LFU, though the presence of undetectable neutrinos makes the measurement more difficult to perform. The ratio of the decay rates $W \to e^-\bar{\nu}_e$ and $W^- \to \mu^-\bar{\nu}_\mu$ measured by several experiments at Tevatron [188], LEP [189] and LHC [190, 191] (see figure 5.3) constrain the ratio $(g_e/g_\mu)^2$ to the value of 1.005±0.008 which is in excellent agreement with the value of 1 predicted by the SM. Averaging the various measurements, the PDG finds $\frac{\Gamma_{W\to\mu^+\nu_\mu}}{\Gamma_{W\to e^+\nu_e}} = 1.002 \pm 0.006$ [181].

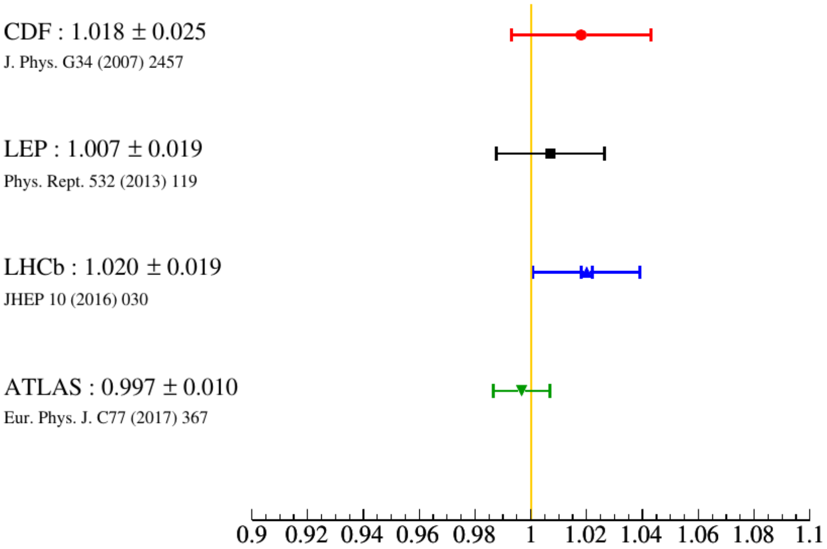


Figure 5.3: Ratio between $\mathcal{B}(W^- \to e^-\bar{\nu}_e)$ and $\mathcal{B}(W^- \to \mu^-\bar{\nu}_\mu)$ measured by Tevatron, LEP and LHC experiments and comparison with the SM model expectation (yellow vertical line).

Measurements involving decays to the τ lepton family are less precise than those involving the first two lepton families, due to the challenging reconstruction of the τ decays. Assuming that LFU holds between the first and the second families, the test performed by measuring the ratio of the partial decay widths[1],

$$\frac{2\Gamma_{W^-\to\tau^-\bar{\nu}_\tau}}{\Gamma_{W^-\to e^-\bar{\nu}_e}+\Gamma_{W^-\to\mu^-\bar{\nu}_\mu}} = 1.066 \pm 0.025\,,$$

is consistent with the SM expectation at the level of 2.6σ [189].

5.2.2 Meson decays

Leptonic decays of pseudoscalar mesons with total spin 0 and odd partity, such as pions or kaons are good tests of LFU: they are helicity-suppressed in the SM and therefore more sensitive to new physics. The most precise measurement of the ratio for kaons, considering e and μ has been performed by the NA62 experiment [192]:

$$\frac{\Gamma_{K^-\to e^-\bar{\nu}_e}}{\Gamma_{K^-\to\mu^-\bar{\nu}_\mu}} = (2.488 \pm 0.009) \times 10^{-5}.$$

This measurement agrees with the SM forecast of $(2.477 \pm 0.001) \times 10^{-5}$, known with high precision taking into account the meson and leptons masses, and the QED contribution from Ref.[193].

The same measurement using pions was done by the PIENU experiment [194]:

$$\frac{\Gamma_{\pi^-\to e^-\bar{\nu}_e}}{\Gamma_{\pi^-\to\mu^-\bar{\nu}_\mu}} = (1.230 \pm 0.004) \times 10^{-4}\,,$$

which is consistent with the corresponding SM prediction $(1.2352 \pm 0.0001) \times 10^{-4}$ [195], though with poorer precision. Further tests can be performed considering charmed-meson and quarkonia resonances. The measurement obtained using the charmed-meson [196] is:

$$\frac{\Gamma_{D_s^-\to\tau^-\bar{\nu}_\tau}}{\Gamma_{D_s^-\to\mu^-\bar{\nu}_\mu}} = 9.95 \pm 0.61,$$

in agreement with the SM prediction 9.76±0.10 at level of 6% [2].

Lastly, the most precise test of LFU is given by the measurement of quarkonia partial widths ratio [197]:

$$\frac{\Gamma_{J/\psi\to e^+e^-}}{\Gamma_{J/\psi\to\mu^+\mu^-}} = 1.0016 \pm 0.0031\,,$$

[1] as the branching ratio for a particular decay mode $\mathcal{B}(decay)$ is $\frac{\Gamma_{decay}}{\Gamma_{total}}$

[2] This ratio is given by $\left(\frac{M_\tau}{M_\mu}\right)^2\left(\frac{M_{D_s}^2-M_\tau^2}{M_{D_s}^2-M_\mu^2}\right)^2$ according to Eq. 307 in [196]. The difference in order of magnitudes compared to $\left(\frac{\Gamma_{K^-\to e^-\bar{\nu}_e}}{\Gamma_{K^-\to\mu^-\bar{\nu}_\mu}}\right) = \left(\frac{M_e}{M_\mu}\right)^2\left(\frac{M_K^2-M_e^2}{M_K^2-M_\mu^2}\right)$ [193] is due to the light mass of the electron which is therefore much more suppressed.

that reaches a precision of 0.31%.

5.2.3 Tests using $b-$hadron decays

b-hadron decays proceeding via $b \to c\ell^-\bar{\nu}_\ell$ flavour-changing-charged current (FCCC) or $b \to s\ell^+\ell^-$ flavour-changing-neutral current (FCNC) also provide means to test LFU.

The difference in the mixing parameters between the three generations of fermions as explained in Section 5.1, has an impact on the decay rates. For example, in the case of Flavour-Changing Charged-Current (FCCC) with transitions like $b \to c\ell^-\bar{\nu}_\ell$ one CKM matrix element is involved (V_{cb}), as in figure 5.4.

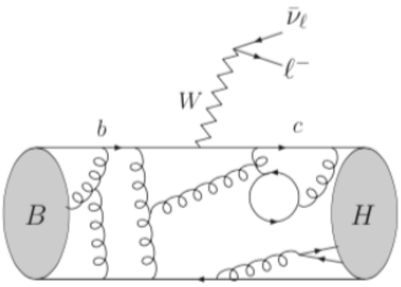


Figure 5.4: Illustration of a FCCC $b \to c\ell^-\bar{\nu}_\ell$ transition in the SM, through the semi-leptonic decay of a B meson in a final state containing a charm hadron H

In the case of Flavour-Changing Neutral-Current (FCNC) with transitions such as $b \to s\ell^+\ell^-$,the matrix elements involved depend on the quarks in the the loop and have the form ($V_{ib}V_{is}^*$) where $i = u, c, t$, as illustrated by Figure 5.5.

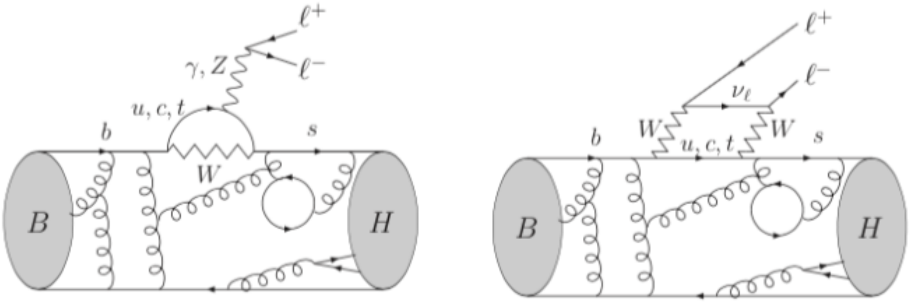


Figure 5.5: Feynman diagram for a FCNC $b \to s\ell^+\ell^-$ transition due to (left) penguin and (right) box diagrams responsible for the decay of a B meson to a final state containing a strange hadron H and two oppositely charged leptons.

In the following we are reviewing the most relevant experimental results, starting with the latter and finishing with the semileptonic decays.

$b \to s\ell^+\ell^-$ transitions

These transitions are highly suppressed in the SM [198], they involve non-diagonal elements of the CKM matrix, and are challenging both experimentally and theoretically. The Feynman diagrams for the $B^0 \to K^*\ell^+\ell^-$ decay are shown in figure 5.5 and in the SM proceed through so-called penguin or box diagrams, depending on the exchange of Z/γ or W^+W^-. Additional New Physics contributions could also play a role, affecting the following ratios:

$$R(H_s) = \frac{\int_{q^2_{min}}^{q^2_{max}} \frac{d\Gamma(H_b \to H_s\mu^+\mu^-)}{dq^2} dq^2}{\int_{q^2_{min}}^{q^2_{max}} \frac{d\Gamma(H_b \to H_s e^+e^-)}{dq^2} dq^2}, \tag{5.7}$$

where H_b is a b-hadron (*e.g.* B^+, B^0, ...), H_s is an s-hadron (*e.g.* K, K^*..), q^2 is the invariant mass squared of the two leptons and q^2_{min} and q^2_{max} the integration limits.

As the same theoretical uncertainties affect both the numerator and the denominator, these ratios are predicted with high precision in the SM [199, 200, 201] and therefore represent excellent check of its validity. New Physics contributions could affect either or both decays rates and angular distributions of the final-state particles.

Two ratios are particularly interesting: $R(K)$ and $R\left(K^{*0}\right)$ obtained considering the decays $B^+ \to K^+\mu^+\mu^-$ and $B^+ \to K^+e^+e^-$, for the former, and $B^0 \to K^{*0}\mu^+\mu^-$ and $B^0 \to K^{*0}e^+e^-$ for the latter. They have been measured using B-meson decays by BaBar [202], Belle [203, 204, 205] and LHCb [206, 207, 208, 209], as listed in Table 5.1. Figure 5.6 shows how they compare to the SM expectations. Measurements performed by LHCb in 2019 [208] are consistent with the SM at the level of 2.5 standard deviations. This was updated in December 2022 [209], where a measure in bins of q^2 with improved background corrections showed a very good agreement with the SM as shown in Figure 5.7.

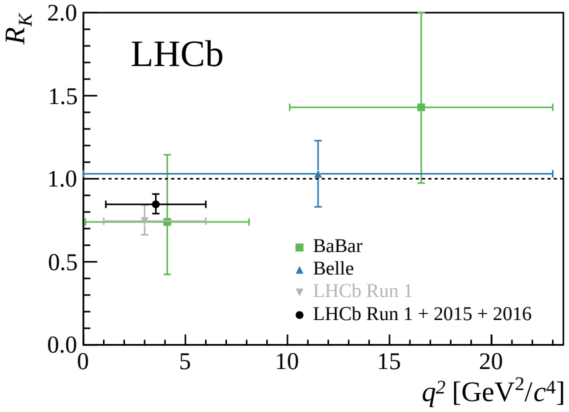


Figure 5.6: $R(K)$ measurements performed by BaBar [202], Belle [203] and LHCb [206, 207, 208] compared to the SM expectation. The LHCb results are superseded by [209].

Tests of LFU with b-baryons have been performed by LHCb using Λ_b decays [210, 211]. The measurement of the R^{-1}_{pK} ratio is compatible with the SM prediction within one standard deviation.

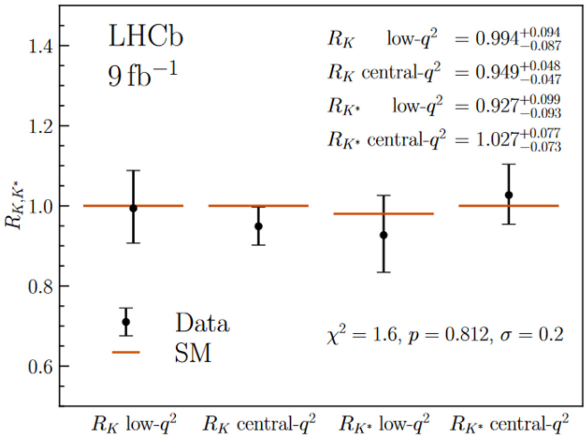


Figure 5.7: Measurement of $R(K)$ and $R\left(K^{*0}\right)$ by LHCb as of 2022 [209]

Table 5.1: Summary of the measurements related to the $b \to s\ell^+\ell^-$ transitions performed by BaBar, Belle, and LHCb experiments.

Experiment	Parameter	q^2 range	Value	SM consistency	Ref.
Belle	$R(K)$	[0.1,4.0]	$1.01^{+0.28}_{-0.25} \pm 0.02$	$< 1\sigma$	[204]
Belle	$R(K)$	[4.8,12]	$0.85^{+0.30}_{-0.24} \pm 0.01$	$< 1\sigma$	[204]
Belle	$R(K)$	[1,6]	$1.03^{+0.28}_{-0.24} \pm 0.01$	$< 1\sigma$	[204]
Belle	$R(K)$	[10.2,12.8]	$1.97^{+1.03}_{-0.89} \pm 0.02$	$< 1.1\sigma$	[204]
Belle	$R(K)$	>14.18	$1.16^{+0.30}_{-0.27} \pm 0.01$	$< 1\sigma$	[204]
Belle	$R(K)$	whole range	$1.10^{+0.16}_{-0.15} \pm 0.02$	$< 1\sigma$	[204]
BaBar	$R(K)$	[0.10,8.12]	$0.74^{+0.40}_{-0.31} \pm 0.06$	$< 1\sigma$	[202]
BaBar	$R(K)$	>10.11	$1.43^{+0.65}_{-0.44} \pm 0.12$	$< 1\sigma$	[202]
LHCb	$R(K)$	[1.1,6.0]	$0.846^{+0.042}_{-0.013}{}^{+0.039}_{-0.012}$	3.1σ	[207]
LHCb	$R(K)$	[0.1,1.0]	$0.994^{+0.090}_{-0.082}{}^{+0.029}_{-0.027}$	$< 1\sigma$	[209]
LHCb	$R(K)$	[1.1,6.0]	$0.949^{+0.042}_{-0.041}{}^{+0.022}_{-0.022}$	$< 1\sigma$	[209]
Belle	$R\left(K^{*0}\right)$	[0.045,1.1]	$0.52^{+0.36}_{-0.26} \pm 0.06$	$\approx 1\sigma$	[205]
Belle	$R\left(K^{*0}\right)$	[1.1,6]	$0.96^{+0.45}_{-0.29} \pm 0.11$	$< 1\sigma$	[205]
Belle	$R\left(K^{*0}\right)$	[0.1,8]	$0.90^{+0.27}_{-0.21} \pm 0.10$	$< 1\sigma$	[205]
Belle	$R\left(K^{*0}\right)$	[15,19]	$1.18^{+0.52}_{-0.32} \pm 0.11$	$< 1\sigma$	[205]
Belle	$R\left(K^{*0}\right)$	> 0.045	$0.94^{+0.17}_{-0.14} \pm 0.08$	$< 1\sigma$	[205]
BaBar	$R\left(K^{*0}\right)$	[0.10,8.12]	$1.06^{+0.48}_{-0.33} \pm 0.09$	$< 1\sigma$	[202]
BaBar	$R\left(K^{*0}\right)$	> 10.11	$1.18^{+0.55}_{-0.37} \pm 0.11$	$< 1\sigma$	[202]
LHCb	$R\left(K^{*0}\right)$	[0.045,1.1]	$0.66^{+0.11}_{-0.07} \pm 0.03$	2.2σ	[210]
LHCb	$R\left(K^{*0}\right)$	[1.1,6.0]	$0.69^{+0.11}_{-0.07} \pm 0.05$	2.4σ	[210]
LHCb	$R\left(K^{*0}\right)$	[0.1,1.1]	$0.927^{+0.093}_{-0.087}{}^{+0.036}_{-0.035}$	$< 1\sigma$	[209]
LHCb	$R\left(K^{*0}\right)$	[1.1,6.0]	$1.027^{+0.072}_{-0.068}{}^{+0.027}_{-0.026}$	$< 1\sigma$	[209]
LHCb	R_{pK}^{-1}	[0.1,6.0]	$1.17^{+0.18}_{-0.16} \pm 0.07$	1σ	[211]

$b \to c\ell^-\bar{\nu}_\ell$ transitions

These transitions are called semileptonic decays, and occur in the SM through tree-level diagrams as shown in Figure 5.4. The only difference in the decay rate to final states with different lepton families arises from the different mass of the leptons involved that affects the kinematics.

The semileptonic decays involving the first two lepton generations (*i.e.* e^- and μ^-) are consistent to each other within experimental uncertainties and in agreement with LFU. The large τ mass makes this kind of transitions more sensitive to the presence of potential new physics effects.

The useful observable to probe NP contributions are ratios of branching fractions defined as:

$$R(H_c) = \frac{\mathcal{B}(H_b \to H_c\tau^-\bar{\nu}_\tau)}{\mathcal{B}(H_b \to H_c\ell^-\bar{\nu}_\ell)}\,, \tag{5.8}$$

where H_b and H_c are hadrons containing a b and a c quark, respectively, and ℓ represents an electron or a muon. The ratio helps to cancel large part of uncertainties due to e.g. $|V_{cb}|$ and form factors, as well as the experimental uncertainties given by the measurement of branching fractions, and systematic uncertainties related to reconstruction efficiencies. The SM predicts a smaller decay rate for the τ due to its larger mass, but that also makes it more sensitive to the presence of new physics contribution in case the coupling to the leptons depends on their mass.

The Babar, Belle and Belle II experiments measured the $R\,(D)$ and $R(D^*)$ ratios by using different τ decay modes and different techniques to fully reconstruct the signal candidate. BaBar in Ref. [212, 213], Belle in Ref. [214, 215, 216, 217] and Belle II in Ref. [218].

The LHCb collaboration, instead, performed the measurement of the $R\,(D)$, $R(D^*)$ and $R(J/\psi)$ ratios exploiting the leptonic τ decays $\tau^- \to \mu^-\nu_\tau\overline{\nu_\mu}$ [219, 220, 221], and of the $R(D^*)$ and $R(\Lambda_c)$ ratios using the 3-prong hadronic τ decays $\tau^- \to \pi^-\pi^+\pi^-\nu_\tau$ or $\tau^- \to \pi^-\pi^+\pi^-\pi^0\nu_\tau$ [222, 223, 224, 225]

Table 5.2: Measurements performed by BaBar, Belle and LHCb collaborations to test the LFU in semitauonic H_b decays.

Experiment (year)	H_b-tag	τ decay	$R\,(D)$	$R(D^*)$	Ref.
BaBar (2012)	Had.	$\tau^- \to \ell^-\bar{\nu}_\ell\nu_\tau$	0.440±0.058 ± 0.042	0.332±0.024±0.018	[212, 213]
Belle (2015)	Had.	$\tau^- \to \ell^-\bar{\nu}_\ell\nu_\tau$	0.375±0.064 ± 0.026	0.293±0.038±0.015	[214]
Belle (2016)	SL	$\tau^- \to \ell^-\bar{\nu}_\ell\nu_\tau$	/	0.302±0.030±0.011	[215]
Belle (2017)	Had.	$\tau^- \to \pi^-(\pi^0)\nu_\tau$	/	$0.270\pm0.035^{+0.21}_{-0.16}$	[216]
Belle (2019)	SL	$\tau^- \to \ell^-\bar{\nu}_\ell\nu_\tau$	0.307±0.037 ± 0.016	0.283±0.018±0.014	[217]
LHCb (2015)	/	$\tau^- \to \mu^-\bar{\nu}_\mu\nu_\tau$	/	0.336±0.027±0.030	[220]
LHCb (2017)	/	$\tau^- \to \pi^-\pi^+\pi^-(\pi^0)\nu_\tau$	/	0.291±0.019±0.029	[222, 223]
LHCb (2022)	/	$\tau^- \to \mu^-\bar{\nu}_\mu\nu_\tau$	0.441±0.040±0.066	0.281±0.018±0.024	[219]
LHCb (2023)	/	$\tau^- \to \pi^-\pi^+\pi^-(\pi^0)\nu_\tau$	/	0.247±0.015±0.019	[224]
Belle II (2023)	Had.	$\tau^- \to \ell^-\bar{\nu}_\ell\nu_\tau$	/	$0.267^{+0.041+0.028}_{-0.039-0.033}$	[218]
Experiment (year)	H_b-tag	τ decay	$R(\Lambda_c)$	$R(J/\psi)$	Ref.
LHCb (2017)	/	$\tau^- \to \mu^-\bar{\nu}_\mu\nu_\tau$	/	0.71±0.17±0.18	[221]
LHCb (2022)	/	$\tau^- \to \pi^-\pi^+\pi^-(\pi^0)\nu_\tau$	0.242±0.026±0.071	/	[225]

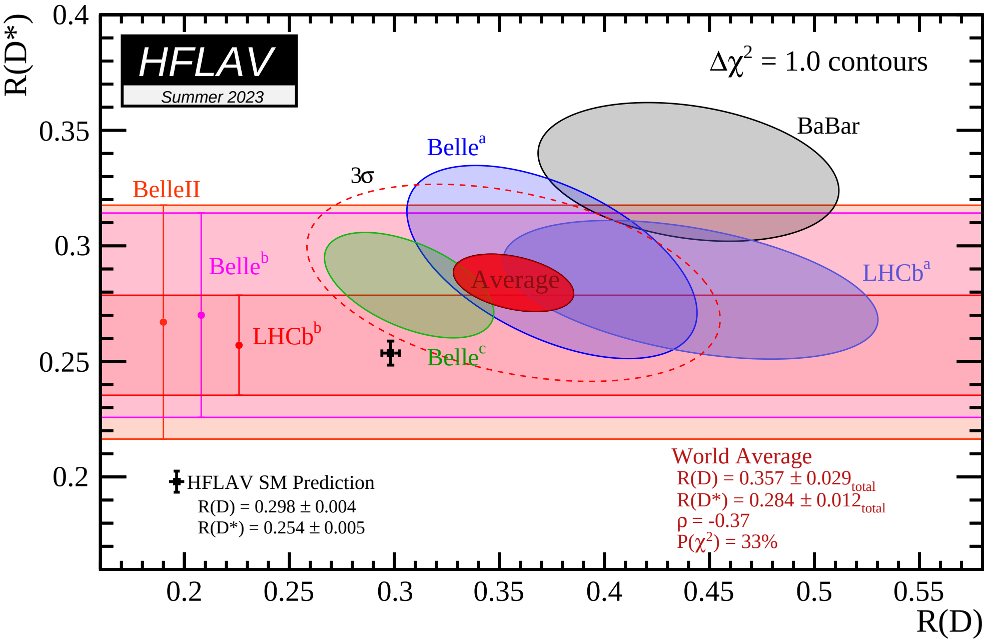


Figure 5.8: Current status of the combination of both $R\,(D)$ and $R(D^*)$ measurements and their comparison with the SM prediction [196].

A summary of the measurements of $R\,(D)$ and $R(D^*)$, as well as their combination and comparison with SM predictions, can be seen in Figure 5.8 while Table 5.2 lists all the LFU measurements performed on the semitauonic B decays by the different collaborations. The SM predictions of $R\,(D)$ and $R(D^*)$ are estimated to be [196]

$$R(D) = 0.298 \pm 0.004$$

$$R(D^*) = 0.254 \pm 0.005.$$

From the experimental side both the averages of $R\,(D)$ and $R(D^*)$ measurements exceed the SM prediction at 1.4σ and 2.9 σ, respectively. Considering a $R\,(D)$-$R(D^*)$ correlation of -0.37, the new combination is consistent with the SM within 3.2 σ. The deviation from unity of both the ratios $R\,(D)$ and $R(D^*)$ is related to the different lepton masses and to the ratios of form factors when considering all the terms in Eq. 5.8. The measurements of $R(\Lambda_c)$ and $R(J/\psi)$, instead, agree with the SM prediction within the current limited precision.

Improving the precision of the current measurements and measuring additional observables is crucial to establish whether the observed tension is confirmed. In such a case it would be a clear indication of NP contributions at the tree level process. With this purpose in mind the measurement of $R\,(D_s)$ is proposed and presented in this thesis.

5.3 Conclusion

Lepton Flavour Universality is implicit in the Standard Model of Particles Physics and its validation a good test of the theory's validity. Direct tests using Z of W bosons decays have shown agreement between the measurements and the theory, as well as leptonic decays of pseudoscalar mesons. b-hadron decays can also be used to probe this assumption, and while $b \to s\ell^+\ell^-$ decays are in agreement with the SM prediction, some tension is found in measurements involving heavy quarks such as $b \to c\ell^-\bar{\nu}_\ell$ decays, e.g. with $\bar{B} \to D^{(*)}\tau^-\bar{\nu_\tau}$ decays.

To contribute to widen the experimental tests of LFU, this thesis presents a study of $B_s^0 \to D_s^-\tau^+\bar{\nu_\tau}$ decays to measure the $R\left(D_s\right)$ ratio. Like $R(J/\psi)$ and $R(\Lambda_c)$, this measurement is possible only at LHC, where copious amounts of B_s^0, B_c and Λ_b hadrons are produced in pp collisions.

Chapter 6

Hadronic R(Ds) measurement

6.1 Analysis principles

Exploiting the wealth of b-hadrons produced in proton-proton collisions within the LHCb acceptance we aim to verify Lepton Flavour Universality by analyzing B_s^0 decays. More specifically, we want to measure:

$$R(D_s) = \frac{\mathcal{B}(B_s^0 \to D_s^- \tau^+ \nu_\tau)}{\mathcal{B}(B_s^0 \to D_s^- \mu^+ \nu_\mu)}\,, \tag{6.1}$$

which is predicted with high precision in the SM, $R(D_s)^{SM} = 0.2971 \pm 0.0034$ [226], thanks to the cancellation of several theoretical uncertainties that contribute in the calculation of both the branching fractions at the numerator and the denominator.

Measuring the ratio implies counting the number of events of the two decays, referred to *signal* and *reference* in the following, rescaled by their reconstruction efficiency and by the involved branching fractions of the decay processes leading to the chosen final states. If the candidates are reconstructed from the same data sample and selected using the same criteria, many experimental systematic uncertainties can cancel out and a precise measurement can be achieved, provided the statistics in the two channels is large enough.

We identify the signal decay $B_s^0 \to D_s^- \tau^+ \nu_\tau$ by looking for the D_s^- decays to $K^+K^-\pi^-$ and, similarly to the $R(D^*)$ analysis [223, 224], looking for the τ^+ decays to $\pi^+\pi^-\pi^+\nu_\tau$ and potentially a non reconstructed π^0. τ leptons decays to three charged particles are not as common as single-prong decays, but they still occur in 15.2% of the cases, as shown in Table 6.1. Such reconstruction implicitly includes $B_s^0 \to D_s^{*-} \tau^+ \nu_\tau$ decays with D_s^{*-} decaying to $D_s^- \gamma$ or $D_s^- \pi^0$. A dedicated selection is also developed to identify such decays.

A schematic representation of the decay is shown in Figure 6.1. Since the reference decay has a different final state topology, containing a muon instead of three pions, the event selection is necessarily different from the signal one and requires a dedicated analysis. The branching fraction of the reference channel has already been measured by LHCb to be $\mathcal{B}(B_s^0 \to D_s^- \mu^+ \nu_\mu)$ = (2.49 ± 0.24)% [227].

To reach the optimal experimental precision on $R\,(D_s)$ the best approach is to determine

Decay	Branching ratio
1-prong decays	(85.25± 0.06)%
3-prong decays	(15.20± 0.06)%
$\tau \to \pi^-\pi^+\pi^-\nu_\tau$	(9.31 ± 0.05%)
$\tau \to \pi^-\pi^+\pi^-\pi^0\nu_\tau$	(4.62 ± 0.05%)

Table 6.1: τ lepton decays branching ratios according to the PDG [181]

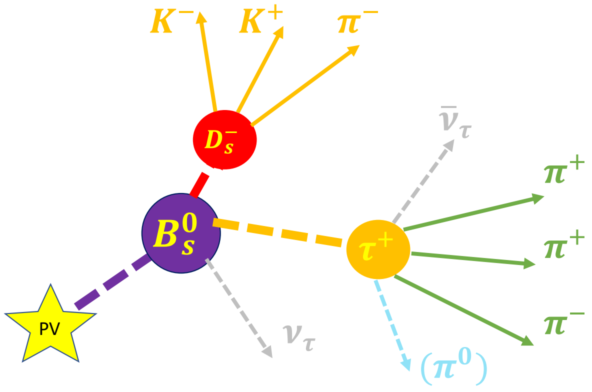


Figure 6.1: Schematic view of the signal decay.

the signal branching fraction relative to a decay with a final state similar to the signal (*norm*) and a known branching fraction and therefore determine $R(D_s)$ as:

$$R(D_s) = \frac{\mathcal{B}(B_s^0 \to D_s^-\tau^+\nu_\tau)}{\mathcal{B}(norm)} \times \frac{\mathcal{B}(norm)}{\mathcal{B}(B_s^0 \to D_s^-\mu^+\nu_\mu)} = \mathcal{K} \times \alpha\,, \tag{6.2}$$

where the α term is computed using the values of known branching fractions, while $\mathcal{K}$ is determined from data as:

$$\mathcal{K} = \frac{N_{sig}}{\epsilon_{sig}}\frac{\epsilon_{norm}}{N_{norm}}\frac{1}{\mathcal{B}(\tau^+ \to \pi^+\pi^-\pi^+(\pi^0)\nu_\tau) \times \mathcal{B}(D_s^- \to K^+K^-\pi^-)}\,. \tag{6.3}$$

In this way, when measuring $\mathcal{K}$, the reconstruction and selection biases implicit in any data analysis should then be similar and mostly cancel out in the ratio as they are correlated.

A detailed study on different suitable decays to be used as *normalisation* channels, listed in Table A.1, has been performed in Ref. [228]. Among all the considered decays, the $B_s^0 \to D_s^-(\to K^+K^-\pi^-)\pi^+\pi^-\pi^+$ and $B_d^0 \to D^-(\to K^+\pi^-\pi^-)\pi^+\pi^-\pi^+$ decays (Figure 6.2) are considered the best choices based on the determination on each channel of the contribution to the $R(D_s)$ uncertainty due to statistics, systematics and external inputs (branching fractions, fragmentation fractions, etc). In the first case the final state is exactly the same as the signal and thus the reconstruction and signal selection are identical. Minor differences in the efficiencies arise from the different kinematics of the final state particles and have a small impact to the systematic uncertainty. The contribution to $R(D_s)$ uncertainty is estimated to be about 19% and is dominated by the poor precision on the $B_s^0 \to D_s^-(\to K^+K^-\pi^-)\pi^+\pi^-\pi^+$ branching fraction, which amounts to about 16% as detailed in Ref. [228].

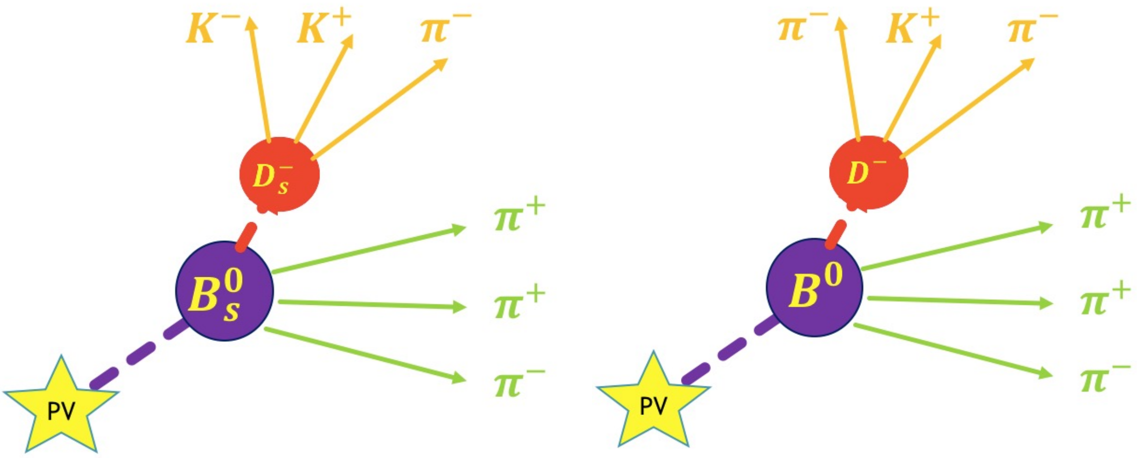


Figure 6.2: (Left) Schematic representation of the $B_s^0 \to D_s^- \pi^+\pi^-\pi^+$ and (Right) $B_d^0 \to D^-\pi^+\pi^-\pi^+$ decay.

In the case of $B_d^0 \to D^-(\to K^+\pi^-\pi^-)\pi^+\pi^-\pi^+$ decays the final state differs from the signal one for the presence of one pion instead of one kaon from the D^- decay instead of the D_s^-. This implies differences in the reconstruction and the selection of the *norm* and the *signal* channels both in the particle identification requirements and in the kinematics that contribute to a non perfect cancellation of the systematic uncertainties. Moreover, since the decay involves a B_d^0 instead of a B_s^0, an additional source of uncertainty related to the corresponding $b-$quark fragmentation fractions, f_s/f_d, needs to be accounted for. Nevertheless the overall contribution to $R(D_s)$ uncertainty is estimated to be smaller than the previous case, about 15%, because of the better precision of the $B_d^0 \to D^-(\to K^+\pi^-\pi^-)\pi^+\pi^-\pi^+$ branching fraction, which amounts to about 10%.

6.1.1 Analysis outline

As discussed in the previous section, the determination of $R(D_s)$ is performed by measuring $\mathcal{K}$ on a suitable *normalisation* channel, evaluating the yields of *signal* and *normalisation* decays in data and correcting for their reconstruction efficiencies. As for the majority of the LHCb branching fraction measurements, the analysis implies several aspects:

- the **event reconstruction** based on the data recorded by the detector. At LHCb this is done both at data filtering/recording time (online) and after it has been stored (offline). Online it is performed on a sub-sample of particles useful for the selection of the interesting events at trigger level whereas offline it is done using the complete information. Reconstructed final state particles consistent with the signal decay topology (vertices and kinematics) are identified and retained for further processing. The reconstruction of the semileptonic *signal* decay presents some challenges due to the fact that the final state contains two undetectable neutrinos. As a consequence, usual kinematics constraints that are typically used to identify a decay cannot be applied and this limits the capability to suppress the background (see next item). Since the B_s^0 kinematics is a key ingredient to determine the yield of the semileptonic signal decay (see Sec.6.6), its determination is improved by means of a few approximations (discussed in Sec. 6.2.1).

- Candidate **selection**, to suppress the contamination of background originated from different sources (combinatorial, different b-hadron decays that are confused as the signal because they have similar final state topology or are mis-reconstructed). As mentioned in the previous item, this step is particularly important for the measurement of $R\left(D_s\right)$ due to the limitations of the semileptonic decays reconstruction. At LHCb the selection is performed at different levels: run time (at trigger level) and offline, after reconstruction. The offline selection consists of several steps: the *stripping* (see Sec.2.4.1), where minimal requirements on the reconstructed decay are applied and selected events are stored, together with other selections of the same *stream*, in different files for a more practical processing. The stripped candidates are then further selected using several detailed information on the candidate reconstruction (kinematics: momentum, transverse momentum or invariant mass of the final state particles; geometrical: vertices separation, impact parameter; quality: track and vertex χ^2; and particle identification), and on the event properties by means of simple cuts or, more often, developing a multivariate analysis. For the $R\left(D_s\right)$ measurement discussed here, different *boosted decision tree* (BDT) classifiers have been developed to identify at best the *signal* and the *normalisation* candidates in the most similar way while suppressing the background contamination (see Sec.6.2). The BDT selections have been studied using simulated events of signal, of the main background contributions, and also pure combinatorial background from data. The work on the analysis preservation and reproducibility discussed in Chapter 3 is particularly important for this part of the analysis as it improves efficiency, reproducibility and also facilitates the development of the BDT classifiers and their integration in the dataset.

- An **intensive study of Monte Carlo simulation data** is needed, especially in the case of semileptonic *signal* decays. Firstly, to study the kinematic corrections to apply to mitigate the effect of the undetected neutrinos and improve the signal reconstruction. Secondly, to optimise the selection (see Sec. 6.4) and classify all the background sources (see Sec. 6.3). Finally, to determine the distributions (templates) needed to fit the data and extract the yields (see Sec. 6.6). For this purpose, large Monte Carlo samples of inclusive decays of b-hadrons decaying to a D_s^- and 3 pions at least, or to a D_s^- and a c-hadron decaying to 3 pions at least were generated (see Table 6.2). Of course the simulation implements the current knowledge of all possible b-hadrons decays; in some cases the branching fractions are extrapolated from measurements of related processes or from theoretical predictions. Needless to say that simulation is extremely useful but cannot be fully trusted, so any possible data/simulation comparison and validation or studies on possible systematic variations mentioned before are mandatory.

- *Signal* (and *norm*) **yield determination**. This is commonly performed by fitting the distribution(s) of a/more meaningful observable of selected candidates. For the *signal* a

3D-fit to the q^2 (the square of the momentum transferred to the τ-ν_τ system), reconstructed τ decay time and the output of the final BDT is used. For the *norm* yield determination a fit to the reconstructed invariant mass $m(B)$ is performed taking advantage of the precise kinematic reconstruction of the decay. In both cases the distributions of the *signal* (*norm)* and of the main different background contributions are determined from simulation (see Sec. 6.3) and validated using data, when possible, to be confident that simulation represents data reliably. The integration of this analysis step in the analysis workflow of Chapter 3 is also precious to be able to reproduce the fits, perform studies on systematic uncertainties, etc.

- *Signal* (and *norm*) **efficiency determination**. It is usually computed using simulation with eventual correction factors derived from data/simulation comparisons (for example, the Monte Carlo simulation data may need to be re-weighted to match real data).

- The determination of the **systematic uncertainties** is an important step of all the analyses. For each of the main sources of systematic uncertainty one usually needs to repeat the yield and efficiency determination with modified settings and compare the result with the nominal one. Possible sources of systematic uncertainties are related to the selection, to the uncertainty on the particle identification efficiency, to data/simulation differences that can be recovered by weighting the selected candidates by appropriate weights, and to variations of the fit due to the uncertainties on the input distributions. In the case of the $R(D_s)$ measurement, for example, one should consider the uncertainties on the distributions used in the fit related to the form-factors used to simulate the decay, or the uncertainties on the background composition, etc. Having the possibility to perform systematic uncertainty studies within the analysis workflow of Sec. 3 is extremely powerful.

6.2 Signal candidate reconstruction and selection

This study uses data collected in 2012 by the LHCb collaboration, corresponding to a luminosity of 2 fb^{-1} of *pp* collisions at $\sqrt{s} = 8$ TeV. The selection has been studied on several samples of simulated *signal*, *normalisation* and inclusive *b*-hadron decays (incl_$H_b \to D_s 3\pi X$ and incl_$H_b \to D_s H_c(\to 3\pi)X$), listed in Table 6.2. Upon successful completion, it will be extended to all samples available, taking into account the particularities of each sample that will have to be studied in detail.

The events are filtered using the experiment's *stripping* framework described in 2.4.1, using stripping lines `Bs2DsTauNuForB2XTauNu` (for the B_s^0 decays) and `B0d2DTauNuForB2XTauNu` (for the B_d^0 decays). The selection criteria are in Table B.1 and are detailed in Ref. [229] as the same data is used to continue the study.

The τ candidate is formed by the combination of three charged particles from a common vertex with a mass hypothesis and particle identification information (PID) compatible with a pion. To suppress the contamination of background due to the combination of a random pion

Decay type	Event type	saved events	eq. fb^{-1}
$B_s^0 \to D_s^-(\to K^+K^-\pi^-)\tau^+(\to \pi^+\pi^-\pi^+(\pi^0))\nu_\tau$	13563002	40k/40k	107
$B_s^0 \to D_s^{*-}(\to D_s^-\gamma/\pi^0)\tau^+(\pi^+\pi^-\pi^+(\pi^0))\nu_\tau$	13763200	80k/80k	108
$B_s^0 \to D_s^-(\to K^+K^-\pi^-)\pi^+\pi^-\pi^+$	13266069	61k/61k	12
$B_d^0 \to D^-(\to K^+\pi^-\pi^-)\pi^+\pi^-\pi^+$	11266009	425k/425k	14.5
incl_$H_b \to D_sH_c(\to 3\pi)X$	23903003	2.5M/2.5M	–
incl_$H_b \to D_s3\pi X$	23903000	5.7M/5.7M	25

Table 6.2: Simulated samples corresponding to 2012 data taking conditions produced for this analysis, listed with Event type that is a eight digit number following the convention defined by LHCb collaboration, the number of events saved in the bookkeeping (after generation and filtering selection) correspond to the Up/Down polarity of the magnetic field. When possible, an estimate of the equivalent luminosity of the samples is given. The last two lines represent the inclusive sample of $b-$hadron decaying into D_s^- and a charm hadron decaying to three pions and into D_s^- and three prompt pions.

with $D^0 \to K^-\pi^+$ decays where the kaon is misidentified as a pion a cut is also applied [1].

The D_s^- candidate is formed by combining three charged particles originating from a common vertex with mass hypothesis compatible with two kaons and one pion. To improve the purity of the signal, a selection is done as described in Ref. [229]. It uses the reconstructed invariant mass of the particles (which should be within ±20 MeV from the known D_s^- mass value) and the particle identification information from the LHCb detector, to provide the *Xc_selection* criteria which is used to filter the data. Candidates likely to be decays from c-hadrons such as D^+ or Λ_c^+ are also excluded using alternative mass hypotheses on the final state particles ($K\pi\pi$ and $Kp\pi$, respectively) and mass and PID requirements. Finally, the B_s^0 candidate is formed by combining the D_s^- and the τ candidates and requiring they are originated from a common vertex, displaced with respect to the PV and with minimal requirements on the kinematics of the final state particles.

Specific selections based on *boosted decision trees* (BDT) implemented with ROOT/TMVA [230] have been developed to enhance the signal-to-background ratio, S/B, of D_s^-, τ and B_s^0 candidates. The BDTs combine information about the kinematics of final state particles and of the decay; they are trained on samples of "true" *signal* candidates, selected from the simulation samples by exploiting the true-ID information, and *background* candidates obtained either from simulation or from data.

Figures 6.3, 6.4 and 6.5 show the distributions of the BDTs and the cuts applied on their output to reduce the background contamination respectively for the B_s^0, D_s^- and τ candidates.

In order to reduce the background due to decays of b-hadrons to final states containing additional tracks with respect the signal (partially reconstructed b-hadron decays) a further BDT is developed. It uses isolation variables obtained by a specific LHCb tool, `TupleToolIsoGeneric`, that classifies all the tracks in the event in terms of their isolation properties with respect to the

[1] we require the delta log likelihood of the kaon vs pion hypothesis to be less than 0 for pairs with a combined mass outside of a ±40 MeV interval around the D^0 mass, and less than -5 within that interval.

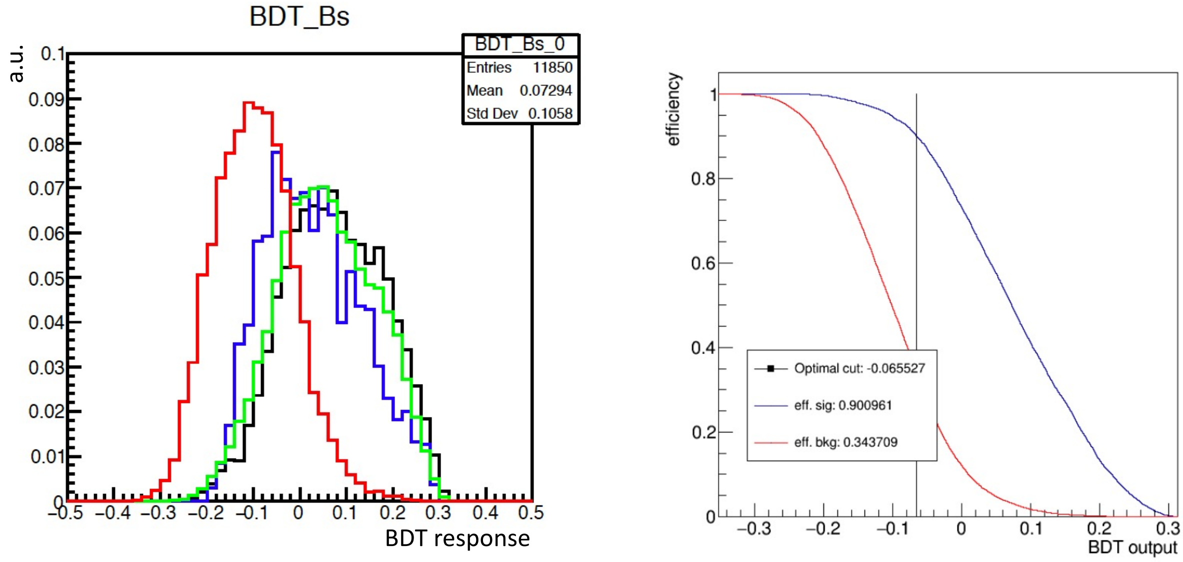


Figure 6.3: (Left) Distribution of the BDT output for the *"$B_{(s)}$ selection"* for the (red) background, (black) signal and normalisation samples: (green) $B_s^0 \to D_s^- \pi^+\pi^-\pi^+$ and (blue) $B_d^0 \to D^-\pi^+\pi^-\pi^+$. (Right) Efficiency for (blue) signal and (red) background of the BDT *"$B_{(s)}$ selection"* as a function of the cut value. The vertical line indicate the cut to which the signal efficiency is 90% and the background contamination around 35%.

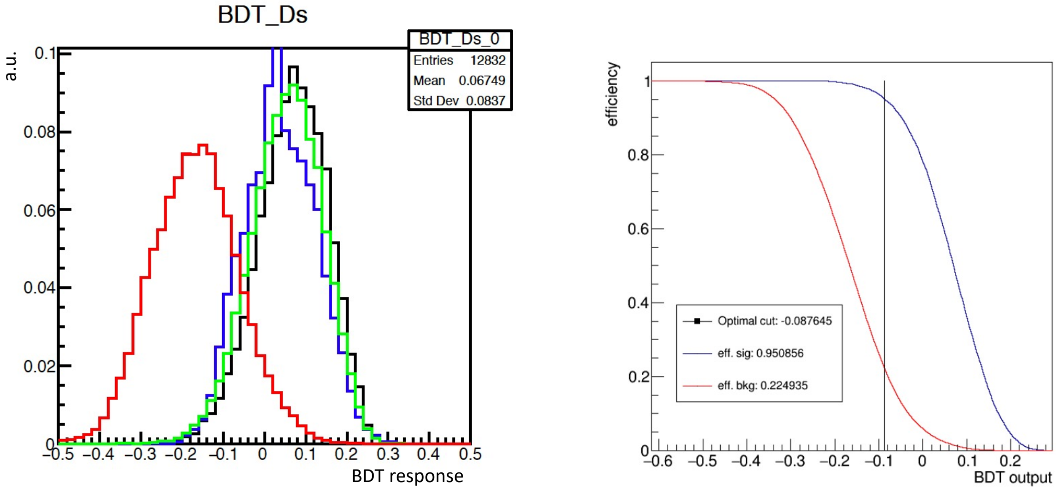


Figure 6.4: (Left) Distribution of the BDT output for the *"$D_{(s)}^+$selection"* for the (red) background and D_s^- signal in (black) $B_s^0 \to D_s^- \tau^+ \nu_\tau$ and (green) $B_s^0 \to D_s^- \pi^+\pi^-\pi^+$ and D^- signal in (blue) $B_d^0 \to D^-\pi^+\pi^-\pi^+$ simulation samples. (Right) Efficiency for (blue) signal and (red) background of the BDT *"$D_{(s)}^+$selection"* as a function of the cut value, from [229]. The vertical line indicate the cut to which the signal efficiency is 95% and the background contamination around 23%.

signal candidate and identifies non-isolated tracks. The BDT is trained on selected candidate decays from signal $B_s^0 \to D_s^-(\to K^+K^-\pi^-)\tau^+(\to \pi^+\pi^-\pi^+(\pi^0))\nu_\tau$ decays (*signal* sample) and

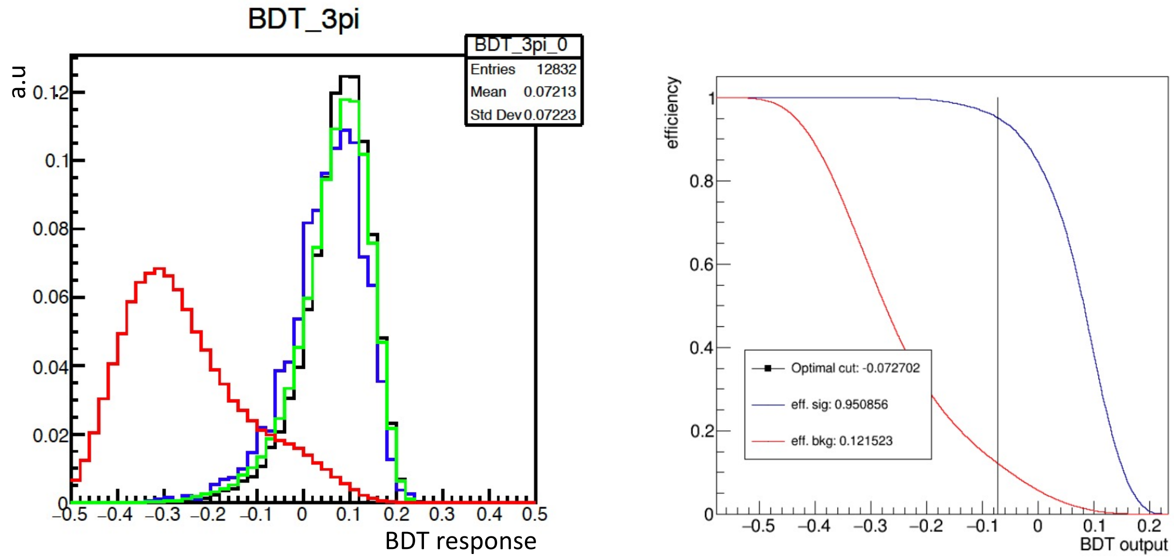


Figure 6.5: (Left) Distribution of the BDT output for the *"3π selection"* for the (red) background, (black) signal $B_s^0 \to D_s^- \tau^+ \nu_\tau$ and normalisation samples: (green) $B_s^0 \to D_s^- \pi^+ \pi^- \pi^+$ and (blue) $B_d^0 \to D^- \pi^+ \pi^- \pi^+$. (Right) Efficiency for (blue) signal and (red) background of the BDT *"3π selection"* as a function of the cut value, from [229]. The vertical line indicate the cut to which the signal efficiency is 95% and the background contamination almost 88%.

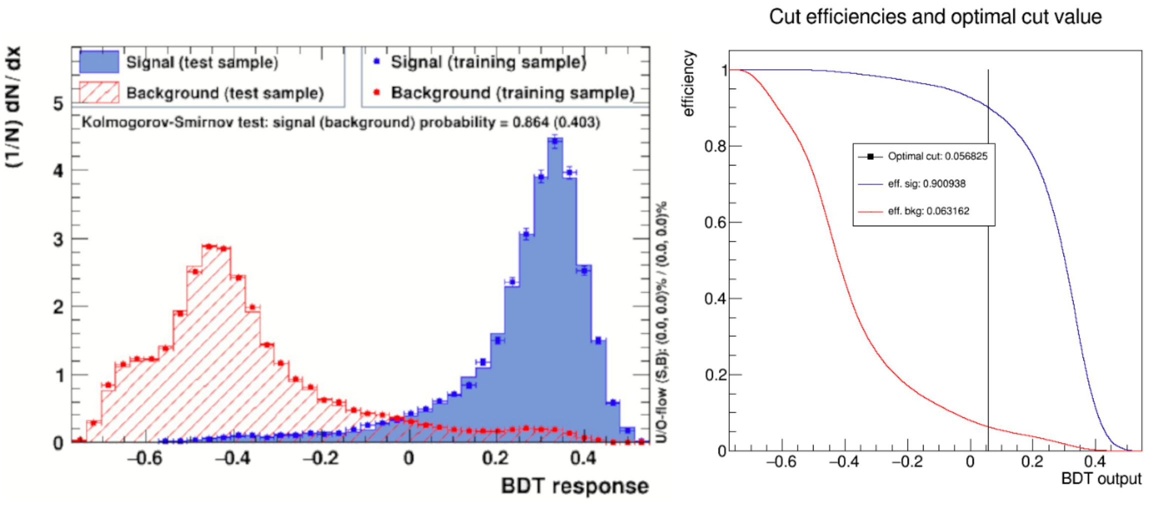


Figure 6.6: (Left) Distribution of the BDT Iso output for the background (in red) and D_s^- signal (in blue). (Right) Efficiency for (blue) signal and (red) background of the BDT Iso as a function of the cut value, from [229]. The vertical line indicate the cut to which the signal efficiency is 90%.

selected candidate decays from the incl_$H_b \to D_s 3\pi X$ containing a "true" non-isolated track (*background* sample) . Figure 6.6 shows the distribution of the BDT output for the two categories of events. A cut that maximises the figure of merit $\frac{S}{\sqrt{S+B}}$, corresponding to 90% of efficiency on signal and 7.7% on background, as computed in the inclusive incl_$H_b \to D_s 3\pi X$ MC sample, is applied to the samples used further in this analysis.

The particle identification (PID) variables derived from the detectors described in Sec.2.1.3 are crucial to identify the decays and to separate background from signal. Simulating the response of the aforementioned detectors is very complex, and it is difficult to make it match the distributions seen on recorded data in all cases. Data-driven methods to calibrate or regenerate the PID variables depending on the conditions have therefore been developed, using the PIDCalib[231] and PIDGen tools. Based on recorded calibration samples, the information used in the selection (PIDK, PIDπ and PIDp) is sampled from the corresponding distributions of true K, π and p in bins of particle's momenta, p_{T} and nTracks, the number of reconstructed tracks in the event. This implies a processing of the output of the ntuples that is integrated in the Snakemake workflows used to process the data.

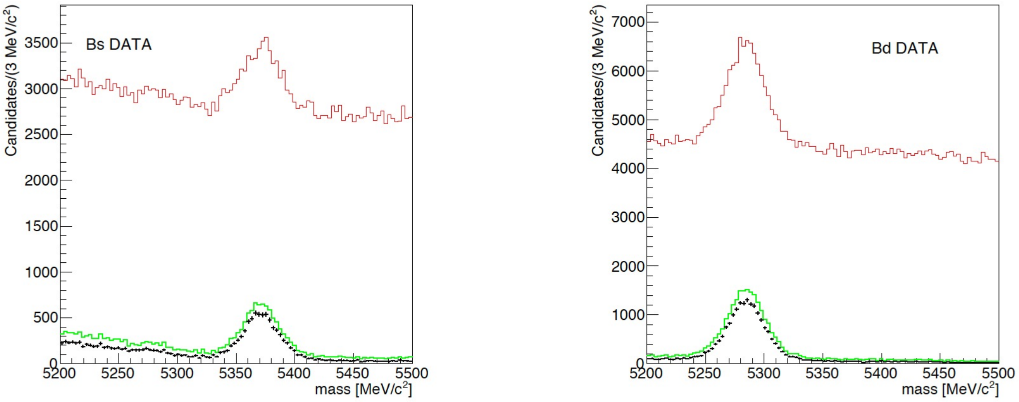


Figure 6.7: Invariant mass distribution for the (Left) B_s^0 and (Right)B_d^0 candidates after the *Xc_Selection* (red), BDT selections(green), and after a preliminary cut to select the normalisation channels based on the distance between the $b-$meson and 3π vertices(black), from [229].

The selection described so far was developed in common between signal and normalisation channels following the criteria mentioned at the beginning of the chapter to minimise the differences of the selection efficiencies and possible sources of systematic uncertainties. Such selection is enough to clearly identify candidates of the *normalisation* channels as shown by Figure 6.7, which feature a clear peak around the nominal B_s^0 (B_d^0) mass, but it is still by far too loose to identify semileptonic signal decays.

Section 6.3 will detail the background composition and the additional selection developed to further reduce the background to signal ratio.

6.2.1 Reconstruction of the signal decay kinematics

Due to the presence of two undetectable neutrinos in the semileptonic signal decay, it is not possible to correctly reconstruct the B_s and τ momenta using the visible particles. However,the position of the τ and B_s^0 decay vertices and of the primary vertex can be used to infer the τ and B_s^0 flight directions, and using the known masses of the τ and the B_s it is possible to reconstruct the full decay kinematics, with the caveat that the equations yield two (plus two) possible solutions.

The τ momentum amplitude in the laboratory frame, $|\vec{p}_\tau|$, is

$$|\vec{p}_\tau| = \frac{(m^2_{3\pi} + m^2_\tau)|\vec{p}_{3\pi}|\cos\theta_{\tau,3\pi} \pm E_{3\pi}\sqrt{(m^2_\tau - m^2_{3\pi})^2 - 4m^2_\tau|\vec{p}_{3\pi}|^2\sin^2\theta_{\tau,3\pi}}}{2(E^2_{3\pi} - |\vec{p}_{3\pi}|^2\cos^2\theta_{\tau,3\pi})}, \tag{6.4}$$

with:

- $\theta_{\tau,3\pi}$ the angle between τ line of flight, which is determined from the measured vertices positions, and the three-momentum of the 3π system;

- $m_{3\pi}$, $|\vec{p}_{3\pi}|$, and $\mathrm{E}_{3\pi}$ the invariant mass, the three momentum and the energy of the 3π system;

- m_τ the known τ mass.

By choosing $\theta_{\tau,3\pi}$ to its maximum allowed value defined as

$$\theta^{max}_{\tau,3\pi} = \arcsin\left(\frac{m^2_\tau - m^2_{3\pi}}{2m_\tau|\vec{p}_{3\pi}|}\right),$$

the two solutions degenerate to a single value for the τ momentum, with a small bias on the signal reconstruction due to this choice. Similarly, naming ξ the system consisting of the D^-_s and the τ, we have

$$|\vec{p}_{B^0_s}| = \frac{(m^2_\xi + m^2_{B^0_s})|\vec{p}_\xi|\cos\theta_{B^0_s,\xi} \pm E_\xi\sqrt{(m^2_{B^0_s} - m^2_\xi)^2 - 4m^2_{B^0_s}|\vec{p}_\xi|^2\sin^2\theta_{B^0_s,\xi}}}{2(E^2_\xi - |\vec{p}_\xi|^2\cos^2\theta_{B^0_s,\xi})}, \tag{6.5}$$

and by imposing

$$\theta^{max}_{B^0_s,\xi} = \arcsin\left(\frac{m^2_{B^0_s} - m^2_\xi}{2m_{B^0_s}|\vec{p}_\xi|}\right)$$

with $\vec{p}_\xi$ and E_ξ being the three-momentum and the energy of the D^-_s τ, where

$$\vec{p}_\xi \equiv \vec{p}_{D_s} + \vec{p}_\tau$$

$$E_\xi \equiv E_{D_s} + E_\tau$$

the two solutions for $\vec{p}_{B^0_s}$ degenerate. With these assumptions, the τ and the B^0_s momenta are recomputed. In turn this allows to compute the mass of the virtual W boson (M_W) and the square of the momentum transferred to the τ-ν_τ system, called q^2 and defined as: $q^2 \equiv (p_{B_s} - p_{D_s})^2 = (p_\tau + p_{\nu_\tau})^2$.

In addition, the τ proper decay time, t_τ, can be obtained by retrieving the Lorentz factors β and γ:

$$\beta = \frac{|\vec{p}_\tau|c}{E_\tau} \tag{6.6}$$

$$\gamma = \frac{1}{\sqrt{1-\beta^2}} \tag{6.7}$$

and the distance of flight of the τ from the

$$L = \sqrt{(\tau_x - B_{s_x})^2 + (\tau_y - B_{s_y})^2 + (\tau_z - B_{s_z})^2} \tag{6.8}$$

with $(B_{s_x}, B_{s_y}, B_{s_z})$ and (τ_x, τ_y, τ_z) respectively the B_s and τ decay vertex positions, as:

$$t_\tau = \frac{L}{\beta\gamma c} \tag{6.9}$$

being c the speed of light. From the τ distance of flight from the B_s^0, L given by Eq. 6.8, and the reconstructed momenta, the τ proper decay time can be expressed as

$$t_\tau = \frac{m_\tau L}{p_\tau} \tag{6.10}$$

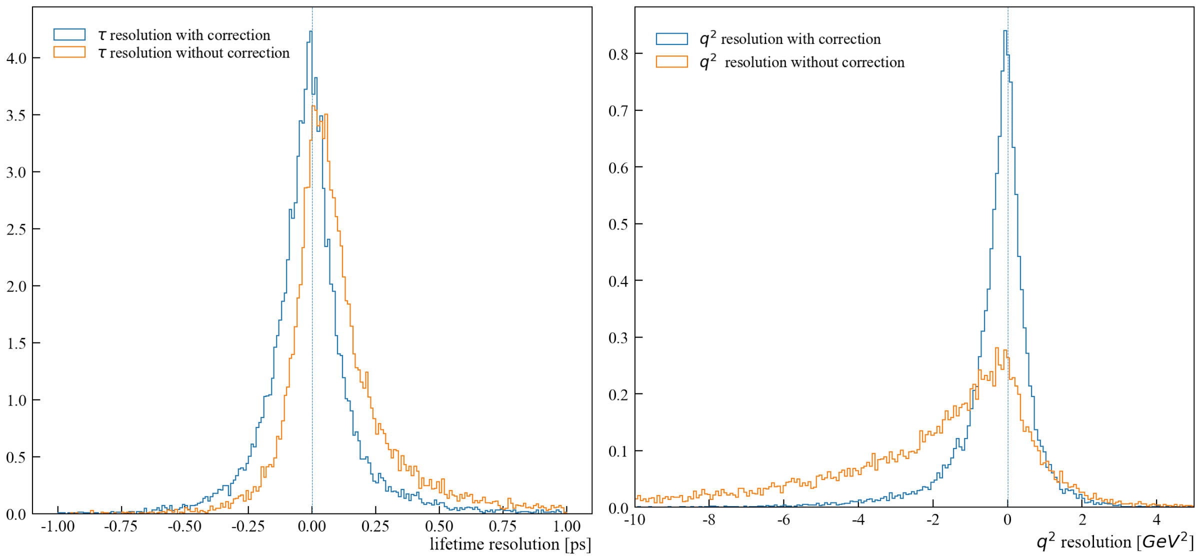


Figure 6.8: Distribution of the difference between (left) the measured and the true τ decay time and (right) the measured and true q^2 value using (blue) uncorrected and (red) corrected kinematics with the approximation described in the text.

The resolution can be derived as the width of a Gaussian model used to fit the distribution of the residuals, defined as the difference of reconstructed and true values on the simulated signal sample. Figure 6.8 shows the comparison of the resolution for the τ decay time (left) and q^2 (right) using the value reconstructed by the LHCb software with the correction mentioned in this section, or without correction (i.e. using only the information from the three pion candidates). The kinematic correction improves the resolution and reduces the bias between the measured and the true value.

Double charm decays of the form $B_s \to D_s(\to KK\pi)H_c(\to 3\pi X)$ are a major source of background as will be discussed in Sec. 6.3. In order to identify them, we compute additional kinematics variables under the hypothesis that the decay is $B_s^0 \to D_s^- H_c(\to 3\pi + X)$ and exploiting the measured kinematics and the position of the reconstructed vertices. As was done in LHCb for the the $R(D^*)$ analysis, the B momentum can be reconstructed by different methods called "scalar" and "vector" approaches (see Appendix C).

Those variables, listed below, will be used by a BDT to separate the double charm background:

- `PBsn`: the B momentum reconstructed using the scalar approach, and considering the corrected B momentum defined in Appendix C (obtained by correcting the B_s^0 vertex position);
- `log(abs(PBv/B_P))`: the ratio between the reconstructed B momentum using the vector approach and the visible one;
- `log(abs(PBvn/B_P))`: the ratio between the B momentum reconstructed using the vector approach and the visible one, using the corrected B vertex;
- `log(abs((PBsn-PBvn)/PBvn))`: the normalised difference between the different estimates of the B momentum;
- `mN2v`: the squared mass of the missing neutral particle (the X in $B_s \to D_s(\to KK\pi)Hc(\to 3\pi X))$ in the H_c decay using the "vector" approach to evaluate the B momentum;
- `sqrt(abs(mDs2vn))`: the reconstructed mass of the H_c system using the vector approach to evaluate the B momentum.

6.3 Background classification

Optimising the signal selection requires a detailed understanding of the background sources. This section details the method used to analyse the Monte Carlo simulated data. It uses the naming conventions for the particles in the decay presented in Figure 6.9:

- X_c represents the candidate for the D_s particle;
- Y represents the candidate for the τ particle;
- B represents the candidate for the decaying B_s particle, reconstructed from the X_c and Y candidate decays.

All data stored in the related ntuples is prefixed with the corresponding particles name.

Each reconstructed particle is associated by means of a specific tool (called `TupleToolMCTruth`, available in the LHCb analysis software) to the corresponding generated "MCParticle" and from that to its origin and successors. A first classification is based on requirements on the particle true-ID.

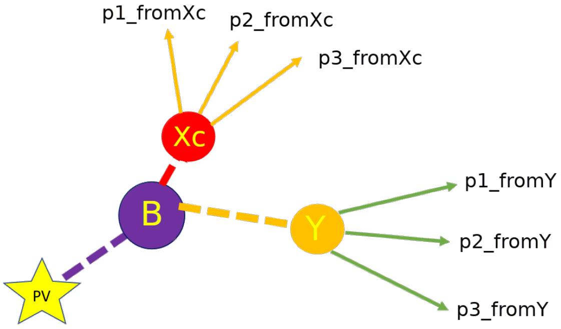


Figure 6.9: Naming conventions for the particles.

6.3.1 Tools

The LHCb collaboration has developed a background category tool [232] running on simulated samples able to "tag" the selected candidates of a given decay chain according to the "true" information available in the simulated data sample. In the case of the reconstructed X_c candidates, `BKGCAT == 0`, identifies true $D_s \to KK\pi$ decays. This tool is sub-optimal for decays involving missing particles such as neutrinos.

For the Y candidate, to have a more precise indication of the ancestry of the 3π final state particles, the `TupleToolB2XMother` tool created for the Run 1 $R(D^*)$ analysis [233] has been used.[2] This tool creates in the final ntuple, a pair of variables for the specified particles (in our case each of the pions forming the Y candidate) with the list of up to 15 ancestors keys (i.e. a unique identifier for the particle in the event) and also the particle ID as defined in the PDG [181] of the matched particle in the simulated event. Listing 9 shows an example of the information available.

```
IDs 1: [-521, -523, 5, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
IDs 2: [-521, -523, 5, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
IDs 3: [213, 20213, 421, -521, -523, 5, 0, 0, 0, 0, 0, 0, 0, 0, 0]
Keys 1: [6348, 6340, 172, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
Keys 2: [6348, 6340, 172, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
Keys 3: [6371, 6370, 6369, 6348, 6340, 172, 0, 0, 0, 0, 0, 0, 0, 0, 0]
```

Listing 9: Example output of the `TupleToolB2XMother` tool used to list the ancestry of a specific particle. In this example 1, 2 and 3 refer to the three pions forming the reconstructed Y candidate. This tool only shows the ancestors of the pions matched, showing a partial view of the decay that ignores the other particles produced.

In this example, for the case the 3 pions originate from the decay $B^{*-} \to B^- \to (D^0 \to (a_1(1260)^+ \to \rho(770)^+ \to \pi^+ X)X)\pi^-\pi^-$, the following output is available:

$$b - \text{quark}_{(5)} \to B^{*-}_{(-523)} \to B^{-}_{(-521)} \to D^0_{(421)}\pi^+_{(IDs1)}\pi^-_{(IDs2)} \tag{6.11}$$

[2] To allow its use in Analysis Production, this tool needed to be available in a released version of the LHCb software so it was improved and merged in the LHCb Analysis application DaVinci/v46r4.

with

$$D^0_{(421)} \to a_1(1260)^+_{(20213)} \to \rho(770)^+_{(213)} \to \pi^+_{(IDs3)} \tag{6.12}$$

By matching the ancestor keys, and using the particle identifiers, it is possible to reconstruct the decays even though other decay products are missing. This allows classifying the selected candidates in the simulation samples. The main criteria are:

- we separate the cases where the reconstructed X_c candidate does or does not correspond to a true D_s by using the output of the Background category tool (i.e. according to the `BKGCAT` value);
- we consider whether the reconstructed Y candidate could be matched to a particle from the underlying simulated event (i.e. whether the `Y_TRUEID` variable is different from 0);
- we consider whether the X_c and the Y candidates originate from the same common particle;
- we consider whether the pions forming the Y candidate originate from the same vertex and whether this vertex is displaced from the B vertex.

Figure 6.10 presents the main categorisation workflow:

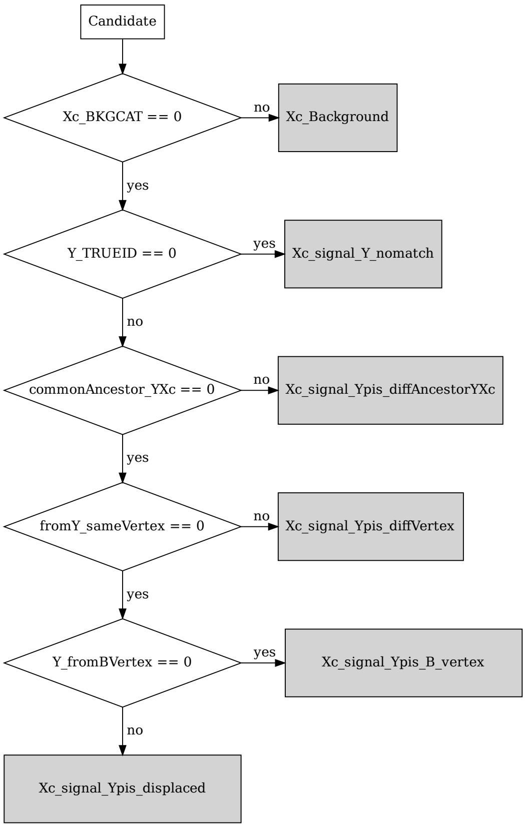


Figure 6.10: Criteria applied for the background categorisation.

Background categorisation also has to take into account whether a pion decay ($\pi^+ \to \mu^+\nu_\mu$) and a photon conversion ($\gamma \to e^+e^-$) have occurred.

Overall the following categories are produced:

- **Xc_background**, when the X_c candidate does not match a true D_s^-.
- **Xc_signal_Y_nomatch** corresponds to the cases where the reconstructed Y candidate could not be matched to a MC particle, it therefore is combinatorial background.
- **Xc_signal_Ypis_diffAncestorYXc** where the true particles matched to Y and X_c candidates do not come from the same ancestor (thus another kind of combinatorial background).
- **Xc_signal_Ypis_diffVertex** corresponds to the cases where the the three pions did not originate from the same vertex, i.e. there is a mismatch or a decay in flight.
- **Xc_signal_Ypis_B_vertex**: the three pions originate from the same vertex as the b-hadron in the event, leading to a ”normalisation-like” topology.
- **Xc_signal_Ypis_displaced**: the three pions originate from different vertex than the b-hadron in the event, leading to a ”signal-like” topology.

These categories can be further detailed, depending of the particles involved in the decay. For the normalisation-like topology (**Xc_signal_Ypis_B_vertex**) we mostly distinguish the categories reported in Table 6.3:

Category	Description
Xc_signal_Ypis_B_vertex_fromBs	Case when the b-hadron is a B_s^0
Xc_signal_Ypis_B_vertex_fromOtherB	Case when B is not a B_s^0

Table 6.3: Categories of candidates originating from the B vertex.

For the signal-like topology (**Xc_signal_Ypis_displaced**) many cases are possible depending on the origin of the reconstructed Y candidate, which is displaced with respect to the reconstructed B vertex, as shown by table 6.4.

This leads to a very fined grained categorisation which may not be very easy to use, hence the need for a coarser granularity to analyse the data. The categorisation in table 6.5 was used.

6.3.2 Results

Tables 6.6 and 6.7 show the simplified and detailed categorisation of the background and signal candidates. The evaluation is done on the inclusive simulation sample incl_$H_b \to D_s 3\pi X$ (event type 23903000 simulated with LHCb 2012 conditions as listed in Table 6.2) where the common selection described in the previous sections has been applied.

Category	Description: Y origin	Simplified category
Xc_signal_Ypis_displaced_fromBs_fromDs	$B_s^0 \to D_s^-$ decay	Double Charm
Xc_signal_Ypis_displaced_fromB0_fromDp	$B^0 \to D^+$ decay	Double Charm
Xc_signal_Ypis_displaced_fromBp_fromD0	$B^+ \to D^0$ decay	Double Charm
Xc_signal_Ypis_displaced_fromLambdab_fromLambdac	$\Lambda_b^- \to \Lambda_c^0$ decay	Double Charm
Xc_signal_Ypis_displaced_fromBs_fromDp	$B_s^0 \to D^+$ decay	Double Charm
Xc_signal_Ypis_displaced_fromBp_fromDp	$B^+ \to D^+$ decay	Double Charm
Xc_signal_Ypis_displaced_fromBs_fromTau	**Signal** decay	**Signal**
Xc_signal_Ypis_displaced_fromB0_fromD0	$B^0 \to D^0$ decay	Double Charm
Xc_signal_Ypis_displaced_fromB0_fromDs	$B^0 \to D_s^+$ decay	Double Charm
Xc_signal_Ypis_displaced_fromBs_fromD0	$B_s^0 \to D^0$ decay	Double Charm
Xc_signal_Ypis_displaced_fromBp_fromDs	$B_s^0 \to D_s^+$ decay	Double Charm
Xc_signal_Ypis_displaced_fromBs_fromDs_fromTau	$B_s^0 \to D_s^+ \to \tau^+$ decay	Tau from Charm
Xc_signal_Ypis_displaced_fromLambdab_fromDs	$\Lambda_b \to D_s^+$ decay	Double Charm
Xc_signal_Ypis_displaced_fromLambdab_fromDp	$\Lambda_b \to D^+$ decay	Double Charm
Xc_signal_Ypis_displaced_fromXic	Ξ_c decay	Other displaced
Xc_signal_Ypis_displaced_fromBs	B_s^0 decay	Other displaced
Xc_signal_Ypis_displaced_fromB0_fromLambdac	$B^0 \to \Lambda_c$ decay	Double Charm
Xc_signal_Ypis_displaced_fromB0_fromDs_fromTau	$B^0 \to D_s^+ \to \tau^+$ decay	Tau from Charm
Xc_signal_Ypis_displaced_fromLambdab_fromD0	$\Lambda_b \to D^0$ decay	Double Charm
Xc_signal_Ypis_displaced_fromB0_fromDp_fromTau	$B^0 \to D^+ \to \tau^+$ decay	Tau from Charm
Xc_signal_Ypis_displaced_NA	Unmatched	Other displaced
Xc_signal_Ypis_displaced_fromBp_fromDs_fromTau	$B^+ \to D_s^+ \to \tau^+$ decay	Tau from Charm
Xc_signal_Ypis_displaced_fromBp	B^+ decay	
Xc_signal_Ypis_displaced_fromLambdab_fromDs_fromTau	$\Lambda_b \to D_s^+ \to \tau^+$ decay	Tau from Charm
Xc_signal_Ypis_displaced_fromBp_fromLambdac	$B^+ \to \Lambda_c$ decay	Double Charm
Xc_signal_Ypis_displaced_fromDs	D_s^+ decay	Other displaced
Xc_signal_Ypis_displaced_fromBs_fromDp_fromTau	$B_s^0 \to D^+ \to \tau^+$ decay	Tau from Charm
Xc_signal_Ypis_displaced_fromBp_fromDp_fromTau	$B^+ \to D^+ \to \tau^+$ decay	Tau from Charm
Xc_signal_Ypis_displaced_fromBs_fromLambdac	$B_s^0 \to \Lambda_c$ decay	Double Charm
Xc_signal_Ypis_displaced_fromLambdab	Λ_b decay	Other displaced
Xc_signal_Ypis_displaced_fromB0	B^0 decay	Other displaced
Xc_signal_Ypis_displaced_fromLambdac	Λ_c decay	Other displaced

Table 6.4: List of detailed categories of Xc_signal_Ypis_displaced and to which simplified category they contribute.

Simplified category	Description
Bad Xc	Badly reconstructed D_s^- (Xc_BKGCAT !=0)
Combinatorial	Various cases of combinatorial background different other than badly reconstructed D_s^-. Xc_signal_Y_nomatch and Xc_signal_Ypis_diffAncestorYXc fall in this category
Double Charm	double charm decays, consisting in Xc_signal and Y from a charmed hadron
normalisation like	Y candidate originated from the reconstructed B decay vertex
Other displaced	Physics background with displaced vertex other than double charm
Signal	$B_s^0 \to D_s^-(\to K^+K^-\pi^-)\tau^+(\to \pi^+\pi^-\pi^+(\pi^0))\nu_\tau$
Tau from Charm	Double charm case where we have a B_s^0 decaying to D_s^- that decays to τ
Uncategorised	Uncategorised cases

Table 6.5: Simplified categorisation for the Monte-Carlo data. This minimal set of categories includes all the background topologies discussed in the text.

	count	Percentage
Category		
normalisation like	541243	44.82
Double Charm	464918	38.50
Bad Xc	183031	15.16
Signal	9966	0.83
Combinatorial	6657	0.55
Tau from Charm	1441	0.12
others	441	0.04

Table 6.6: Background composition according to the simplified categorisation of inclusive MC data (incl_$H_b \to D_s 3\pi X$) for 2012 after the selection of Sec.6.2.

	count	Percentage
Category		
Xc_signal_Ypis_B_vertex_fromBs	481611	39.88
Xc_background	183031	15.16
Xc_signal_Ypis_diffVertex_doubleCharm_OneFromB	130954	10.84
Xc_signal_Ypis_displaced_fromBs_fromDs	84307	6.98
Xc_signal_Ypis_displaced_fromB0_fromDp	66975	5.55
Xc_signal_Ypis_nomatch_doubleCharm	40822	3.38
Xc_signal_Ypis_nomatch_Prompt	37559	3.11
Xc_signal_Ypis_displaced_fromBp_fromD0	30159	2.50
Xc_signal_Ypis_displaced_fromLambdab_fromLambdac	23554	1.95
Xc_signal_Ypis_diffVertex_doubleCharm_TwoFromB	23275	1.93
Xc_signal_Ypis_displaced_fromBs_fromDp	16769	1.39
Xc_signal_Ypis_diffVertex_CharmStrange	12415	1.03
Xc_signal_Ypis_B_vertex_fromOtherB	12176	1.01
Xc_signal_Ypis_diffVertex_normlike	9843	0.82
Xc_signal_Ypis_displaced_fromBs_fromTau	9772	0.81
Xc_signal_Ypis_displaced_fromBp_fromDp	8571	0.71
Xc_signal_Ypis_diffVertex_doubleCharm	8124	0.67
Xc_signal_Ypis_displaced_fromB0_fromD0	6639	0.55
Xc_signal_Ypis_displaced_fromB0_fromDs	4702	0.39
Xc_signal_Ypis_diffAncestorYXc	3436	0.28
Xc_signal_Ypis_nomatch_charmStrange	3316	0.27
Xc_signal_Ypis_diffVertex_SomeFromPV	2983	0.25
Xc_signal_Ypis_displaced_fromBs_fromD0	1910	0.16
Xc_signal_Ypis_displaced_fromBp_fromDs	1505	0.12
others	1481	0.12
Xc_signal_Ypis_displaced_fromBs_fromDs_fromTau	1266	0.10
Xc_signal_Ypis_displaced_fromLambdab_fromDs	542	0.04
total	1207697	100

Table 6.7: Detailed categorisation of inclusive MC data for 2012.

6.4 Separation of signal- and normalisation-like channels

So far the analysis selection has been developed in common between signal and normalisation channel(s) in order to minimise the possible sources of systematic uncertainties and achieve the best precision on $\mathcal{K}$. However such selection is mainly suppressing background from random combination of tracks (combinatorial), misreconstructed or misidentified D_s^- and 3-pion combinations originating from the PV, fake combinations of D_s^- and 3-pions not consistent with a B decay, and partially reconstructed b-hadron decays with additional tracks in the final state. The contamination of background due to b-hadron decays consistent with the signal final state topology is still extremely high, being the normalisation-like decay channels the largest component (∼44%), followed by the double charm category (∼38%). As a result, the signal contribution amounts to about 1% only (see table 6.6).

The $B_s^0 \to D_s^- \pi^+\pi^-\pi^+$ ($B_d^0 \to D^-\pi^+\pi^-\pi^+$) normalisation channel can be clearly identified by its reconstructed invariant mass, as shown by Figure 6.11, which peaks at the B_s^0 (B_d^0) known mass, and by the 3π vertex which coincides with the $B^0_{s(d)}$ vertex. A second peak between 5 GeV and 5.4 GeV due to partially reconstructed decays is also visible.

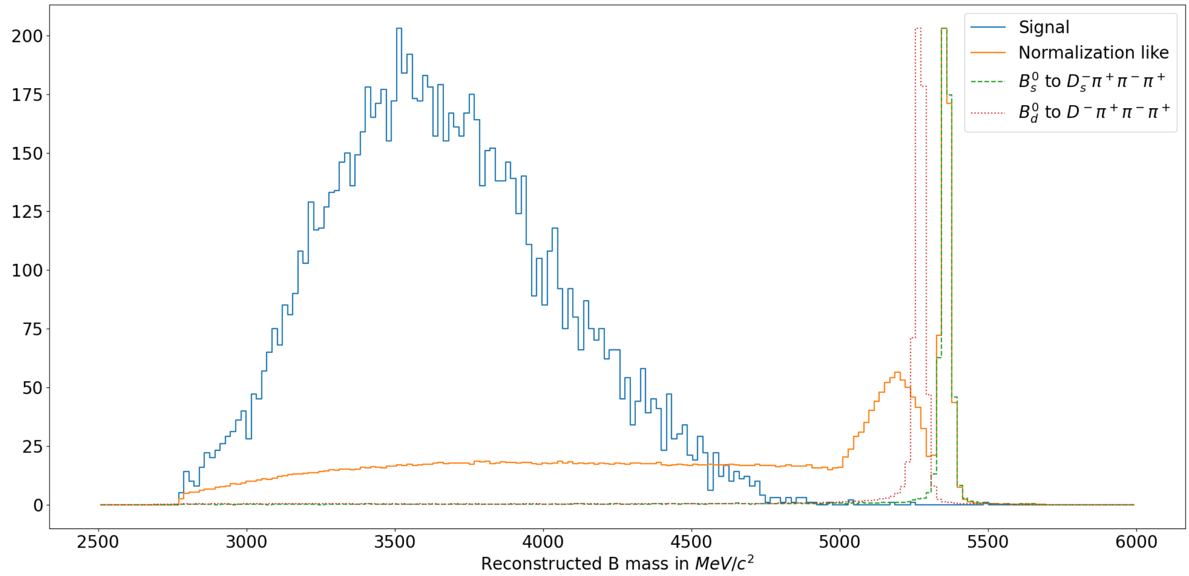


Figure 6.11: Reconstructed B_s^0 mass distribution for the signal, normalisation-like categories in the inclusive MC as well as for the $B_s^0 \to D_s^-\pi^+\pi^-\pi^+$ and $B_d^0 \to D^-\pi^+\pi^-\pi^+$ decays (arbitrary normalisation).

Conversely, signal decays are characterised by a reconstructed B_s^0 invariant mass that vanishes above 5000 MeV/c^2, and feature a displacement of the 3π vertex from τ decays with respect to the B_s^0 vertex, observed mainly along the z direction.

A cut on the B_s^0 invariant mass at 5000 MeV/c^2 is first applied, leading to the candidate categorisation shown in Table 6.8 and Table D.2 in Appendix D.

Comparing the topologies in Figures 6.1 and 6.2, it is clear that another cut on the separation

Category	count	Percentage
Double Charm	452226	47.15
normalisation like	332365	34.65
Bad Xc	157220	16.39
Signal	9961	1.04
Combinatorial	5503	0.57
Tau from charm	1438	0.15
others	415	0.04

Table 6.8: Background composition according to the simplified categorisation of inclusive MC data (incl_$H_b \to D_s 3\pi X$) for 2012 after the selection of Sec.6.2 and for $B\_M < 5000$ MeV/c^2.

of the B and 3π vertices allows separating signal and normalisation channels further. We define the separation of the vertices as follows:

$$B\_Y\_SEP = \frac{V_z(B) - V_z(Y)}{\sqrt{\sigma^2_{V_z(B)} + \sigma^2_{V_z(Y)}}}\,, \tag{6.13}$$

where the uncertainties on the vertices reconstruction along the z direction, σ_{V_z}, are taken into account.

Similarly, having in mind Figure 6.9, we also explore the separation between the D_s^- (X_c) and the 3π (Y) vertices, that could add some discrimination power between the two categories of decays:

$$Xc\_Y\_SEP = \frac{V_z(X_c) - V_z(Y)}{\sqrt{\sigma^2_{V_z(X_c)} + \sigma^2_{V_z(Y)}}}\,. \tag{6.14}$$

Checking the B_Y_SEP separation

Figure 6.12 shows the distribution of the B_Y_SEP variable for the simplified categories of candidates in the simulation sample. From this plot it is clear that by applying a cut on the B_Y_SEP variable it is possible to suppress the normalisation-like background, while it doesn't help reducing the double charm and other backgrounds.

The best cut is decided by maximizing the following figure of merit (FoM)

$$\text{FoM} = \frac{S}{\sqrt{S+B}} \tag{6.15}$$

where S is the number of signal candidates[3], and B is the number of background candidates of "normalisation-like" category. Figure 6.13 shows the FoM as a function of the upper cut on the B_Y_SEP variable, when applied on two different datasets that cuts the contribution of the $B_d^0 \to D^-\pi^+\pi^-\pi^+$ and $B_s^0 \to D_s^-\pi^+\pi^-\pi^+$ normalisation channels). The optimised B_Y_SEP cut value is -4.5 and corresponds to an efficiency on the signal decays of 36.1%.

[3] From category "Xc_signal_Ypis_displaced_fromBs_fromTau"

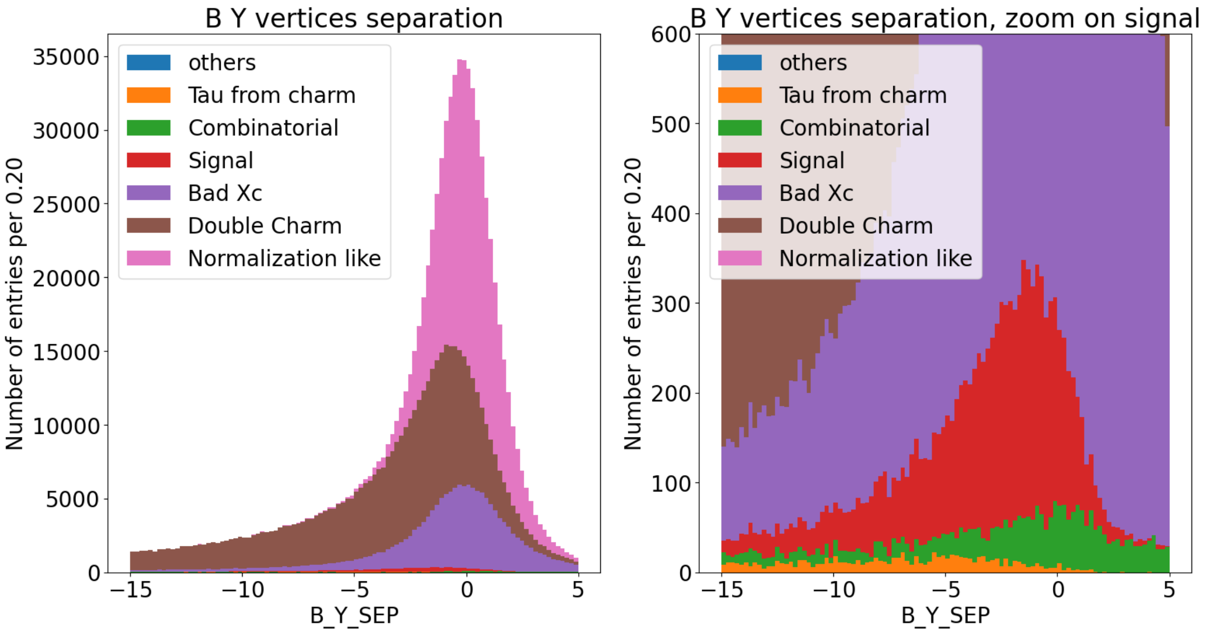


Figure 6.12: Distribution of the B_Y_SEP variable in linear scale after the cut on reconstructed B mass. Left plot shows all categories, dominated by normalisation like, Double charm and Bad Xc while the right plot shows all other categories with less entries that are barely visible in the left plot.

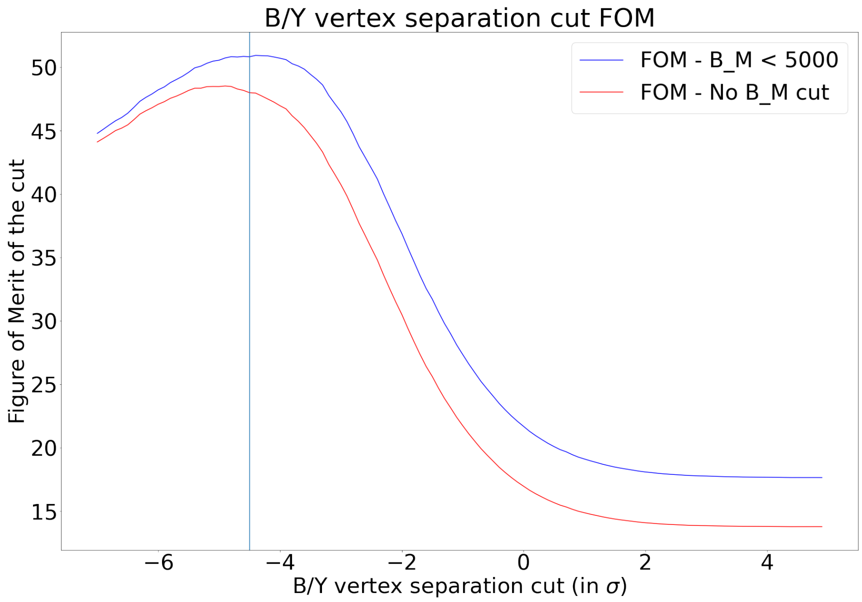


Figure 6.13: Figure of Merit (FoM) as a function of the cut on the B_Y_SEP variable.

Checking the Xc_Y_SEP separation

By using the same FoM defined above we also study if a cut on the Xc_Y_SEP variable could be useful in the separation of the two data samples.

Figure 6.14 shows the FoM as a function of the upper cut on both the B_Y_SEP and Xc_Y_SEP variables, which indicates that the Xc_Y_SEP doesn't add discrimination power with respect to the "normalisation-like" background, so it is not considered.

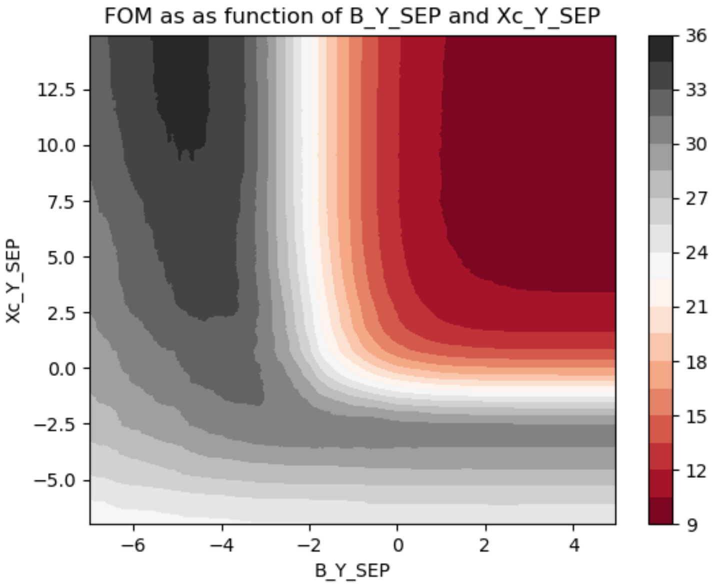


Figure 6.14: Figure of Merit (FoM) as a function of the upper cut on both the B_Y_SEP and Xc_Y_SEP variables (B_Y_SEP<cut1 & Xc_Y_SEP< cut2).

Background categorisation after cut on reconstructed B mass and B_Y_SEP < -4.5

Table 6.9 shows the candidates categorisation after the B_Y_SEP cut. While the normalisation-like background have been greatly reduced, it is clear that the issue is now to suppress the Double Charm background. In detail we draw the following conclusions:

- **normalisation-like** background has been strongly cut by a factor 0.006, which was the intended effect,
- candidates with **Bad Xc** are greatly reduced too,
- the cut has unfortunately a significant effect on **Signal**, which is reduced by a factor 0.36,
- unfortunately the effect of the cut on **Double charm** background (efficiency of 0.464) is less than on signal. This is related to the smaller proper decay time of the τ lepton compared to the charmed hadrons involved in double charm decays (see table 6.10) that have a clear impact on the B_Y_SEP, as shown in Figure 6.15.

Moreover, Table D.2 in Appendix D lists the detailed composition of the sample after application of the B_Y_SEP cut. A cut to ensure that q^2 is positive is also applied to ensure that the kinematic corrections make sense.

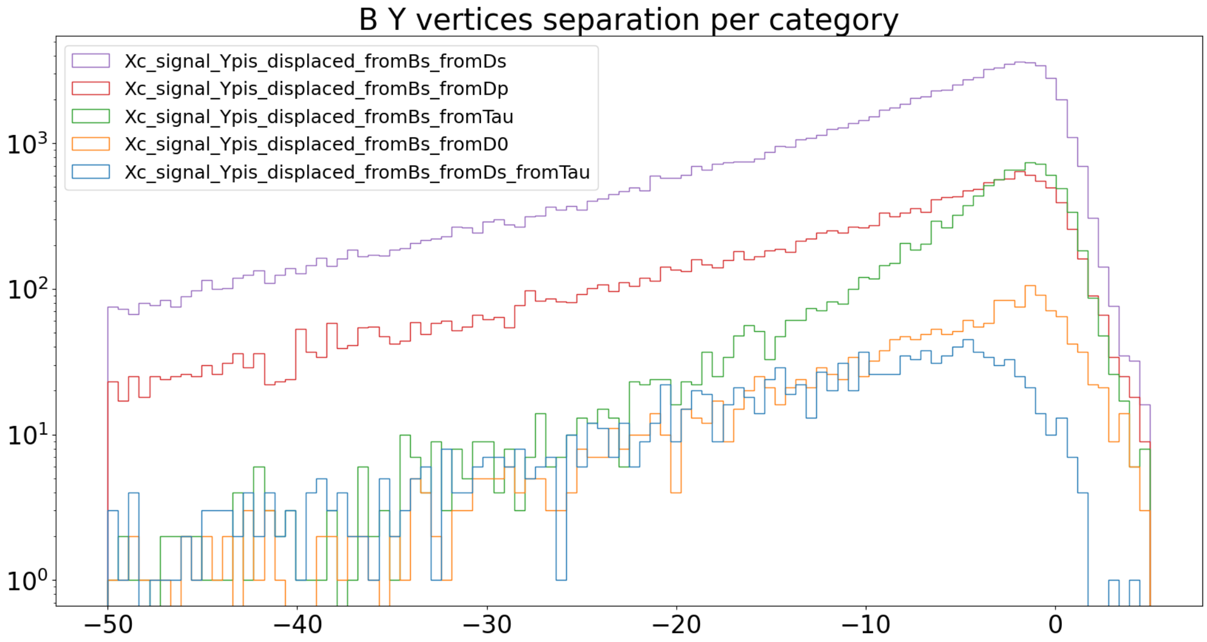


Figure 6.15: B_Y_SEP distribution for signal and different double charm background contributions from B_s^0. The different proper decay times of the charm hadrons and of the τ lepton explain the difference in efficiency introduced by the B_Y_SEP cut.

Category	count	Percentage	Cut efficiency
Double Charm	204311	87.646	0.45
Bad Xc	20449	8.772	0.13
Signal	3562	1.528	0.36
Normalisation like	2034	0.873	0.006
Combinatorial	1433	0.615	0.26
Tau from charm	1110	0.476	0.77
others	211	0.070	0.51

Table 6.9: Simplified categorisation after cut B_Y_SEP < -4.5.

	τ^+	D^+	D^0	D_s^+	Λ_c^+
lifetime (fs)	290.3 ± 0.5	1040 ± 7.0	410.3 ± 1.0	504 ± 4.0	201.5 ± 2.7

Table 6.10: Lifetimes of most common B decay products in our sample.

6.5 Suppression of the Double Charm background

This section explores the use of multivariate techniques to separate the Double Charm background from the signal.

We use *supervised learning* to train a classifier to separate double charm background from the signal. We therefore need as much data as possible to perform that task and gather as many candidates as possible using both the incl_$H_b \to D_s H_c(\to 3\pi)X$, incl_$H_b \to D_s 3\pi X$ and the signal MC samples, corresponding to about 338k, 300k and 12k candidates after the selection described so far, respectively.

We train Gradient Boosted Decision Trees (BDT) using the XGBoost library [234] to distinguish between our signal and all backgrounds. The samples are divided in three sets: 50% is used for training, the rest is used for test and validation of the BDT parameters. The comparison of the two is useful to check for overtraining. Kinematic variables obtained by assuming the B_s^0 decays to D_s^- and a charmed hadron are useful features to identify those decays and they are therefore used to train the BDT, similarly to what was developed in the case of the analysis of $R(D^*)$ decays [222, 235].

We start by normalising the inputs of the BDT, removing the mean and scaling them to unit variance, as that helps with the convergence of the training.

A number of parameters can be used to tune the XGBoost classifier, the following were investigated:

- the *evaluation metric* of the classifier is the classic log loss metric.
- the *n_estimators*, the number of boosting rounds
- the *max_depth* is the maximum depth of the decision trees (crucial to avoid overfitting).
- the *eta*, the learning rate
- the *scale_pos_weight* is set to $\frac{N_{background}}{N_{signal}}$ as is typically done with unbalanced samples with categories with very different counts.
- L1 regularisation (lasso method) with α equals to 0 is used (the default on XGBoost)
- L2 regularisation (ridge method) with λ equals to 1 is used (the default on XGBoost)

The boosted decision tree was trained and tuned, optimising the area under the ROC curve (AUC) presented in in Fig. 6.16 while avoiding overtraining, by minimising the difference between the BDT response on the training and test samples. We therefore minimised $(1 - AUC_{ROC}) \times S$ where S is the Kolmogorov-Smirnov similarity [236] between the responses. In order to find the optimal configuration of the BDT, various parameters combinations were tested within the Optuna [237] optimisation framework and validated against a part of the simulation sample dedicated to this test. This optimal was found with a value for *n_estimators* of 150, a *max_depth* of 3 and *eta* of 0.04, leading to an area of the ROC curve of 0.88. The details of this investigation are listed in Appendix E.

Figure 6.17 shows the classifier response for signal and background candidates on training and test data. There is a good separation between the two categories, and the response of the BDT on training and test data is sufficiently similar to show that we are not overtraining the training sample. This is particularly important for the signal sample where the number of candidates is limited.

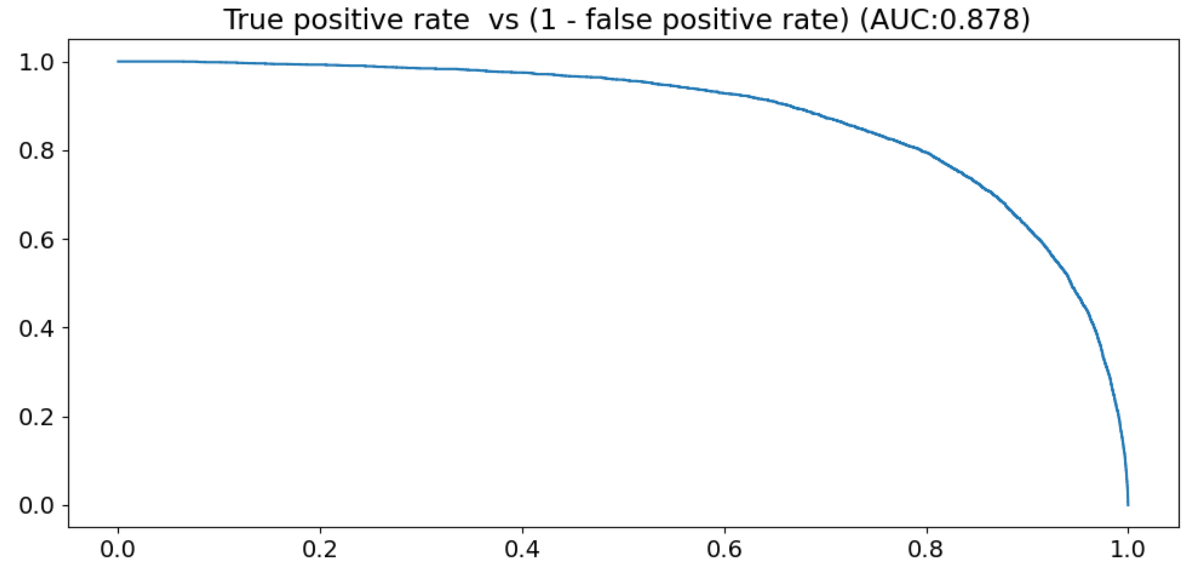


Figure 6.16: BDT ROC curve where the x and y axes are the signal efficiency and background inefficiency. The AUC value represents the integral below the curve, and measures the discriminating power of the BDT.

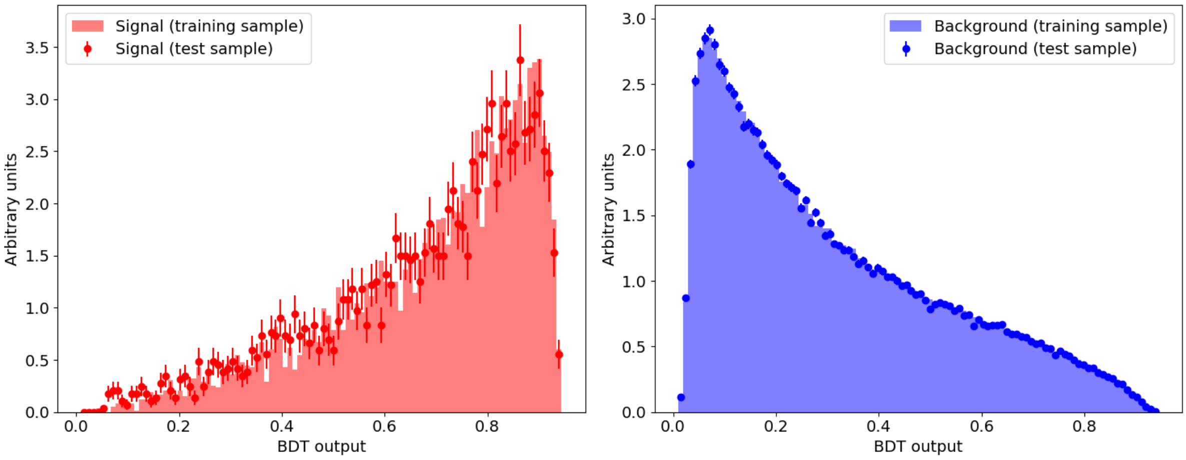


Figure 6.17: BDT response on signal and background.

Several studies were made to check whether training BDTs on specific background categories helps separating the signal, that concluded that training a BDT against all other categories was more effective. The details are explained in Sec. E.3.

The background suppression capability of the BDT is demonstrated by Figures 6.18 and 6.19 which show the signal efficiency, double charm background rejection and FoM as a function of the cut value. The FoM is defined by eq. 6.15, where the signal and double charm background yields are derived from the incl_$H_b \to D_s 3\pi X$ sample which implements the current best knowledge of the $b-$hadron branching fractions. The best cut corresponds to a value of BDT>0.75, a signal efficiency of 0.437, and a background suppression of a factor $\sim$20.

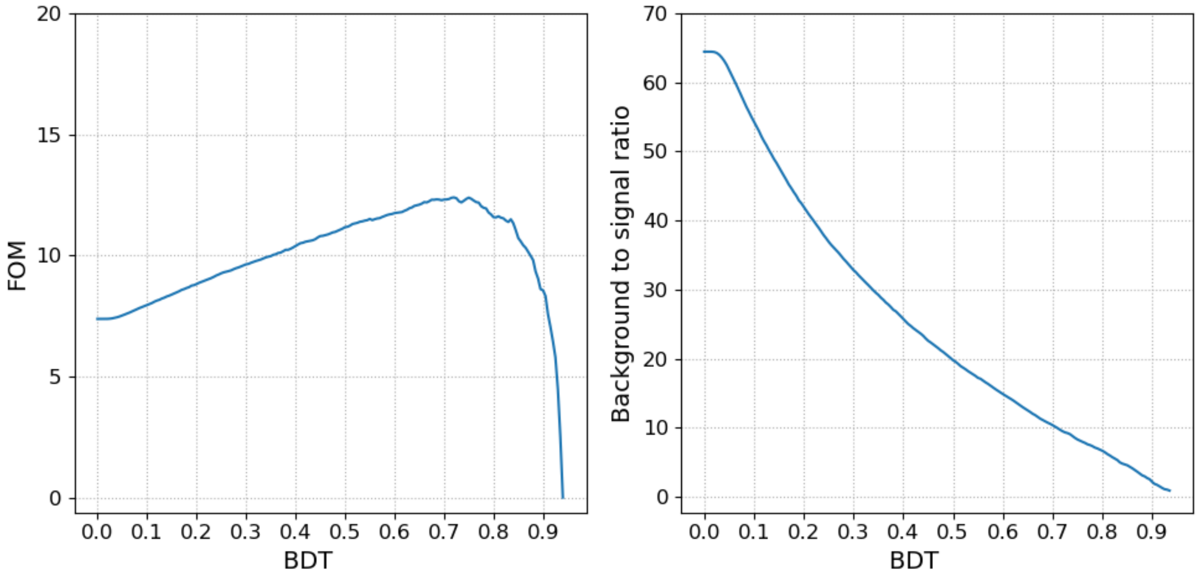


Figure 6.18: FoM (as defined in Eq. 6.15) and background to signal ratio B/S for the BDT computed on the inclusive incl_$H_b \to D_s 3\pi X$ test sample.

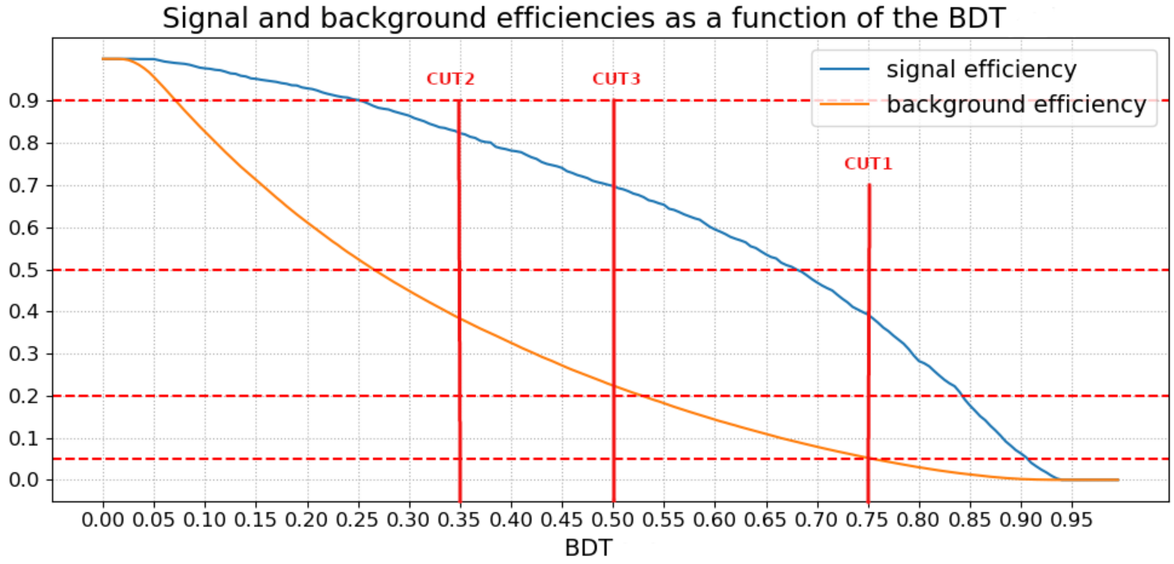


Figure 6.19: Signal and background efficiencies for the BDT output.

Given that the BDT output distribution is used in the final fit to data to extract the signal yield (see Sec. 6.6), we can apply a looser cut with respect the one that optimises the FoM to maximise the statistical sensitivity. In fact, as it will be discussed in Sec. 6.6.1, the BDT output shows different distributions for signal and background, which can be exploited in the fit to determine the yields. The best cut is defined based on sensitivity studies that will be discussed in Sec.6.6.2. Three cut values are considered:

- `CUT1`: BDT >0.75, (i.e. we keep ∼40% of signal, and only 5% of double charm background , with a total background to signal ratio B/S∼8.3)

- CUT2: BDT >0.35, (i.e. we keep ∼84% of signal and ∼38% of double charm background, B/S∼29)
- CUT3: BDT >0.50, (i.e. we keep ∼70% of signal and ∼22% of double charm background , B/S∼20)

The corresponding number of events of each category that pass the selection are reported in tables 6.11 6.12 6.13.

Category	count	Percentage	cut efficiency
Double Charm	10127	76.103	0.050
Signal	1429	10.739	0.401
Bad Xc	1337	10.047	0.065
Normalisation like	212	1.593	0.104
Tau from charm	157	1.180	0.141
others	45	0.338	0.938

Table 6.11: Categorisation of the incl_$H_b \to D_s 3\pi X$ MC sample after CUT1.

Category	count	Percentage	cut efficiency
Double Charm	77343	85.123	0.379
Bad Xc	8352	9.192	0.408
Signal	3005	3.307	0.844
Normalisation like	931	1.025	0.458
Tau from charm	794	0.874	0.715
Combinatorial	370	0.407	0.258
others	65	0.072	1.354

Table 6.12: Categorisation of the incl_$H_b \to D_s 3\pi X$ MC sample after CUT2.

Category	count	Percentage	cut efficiency
Double Charm	44742	83.062	0.219
Bad Xc	5151	9.563	0.252
Signal	2592	4.812	0.728
Normalisation like	610	1.132	0.300
Tau from charm	555	1.030	0.500
Combinatorial	176	0.327	0.123
others	40	0.074	0.833

Table 6.13: Categorisation of the incl_$H_b \to D_s 3\pi X$ MC sample after CUT3.

At this stage, we have categorised the types of background candidates in the Monte Carlo simulation data, and have developed filters that allow characterising and separating it. Figure 6.20 shows an overview of the corresponding analysis steps.

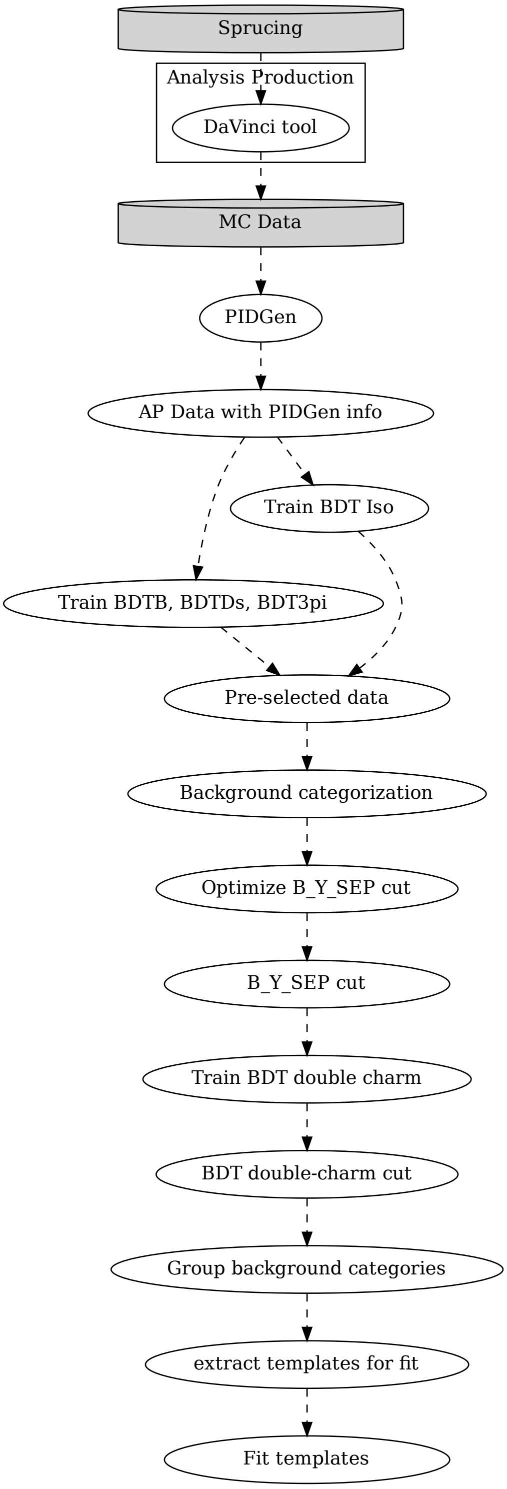


Figure 6.20: Analysis steps

This analysis was developed using the Snakemake workflow engine and the `apd` tool described in Chapter 4. The lessons learned while applying those tools are detailed in Chapter 7.

6.6 Fit

Following the $R(D^{*})$ analysis, the signal yield is obtained from an extended maximum-likelihood fit to the distributions of three observables that show a discrimination power between signal and the different background sources. The observables are physics-related quantities that characterise the signal decay, such as the squared momenta transferred to the τ-ν_τ system, q^2, and the τ lepton decay time, (both calculated with the corrections discussed in Sec.6.2.1), and the output of the final BDT discussed in Sec. 6.5.

The fit is performed by comparing the binned data distributions with 3D binned template distributions that model the Probability Density Functions (PDF) of each signal and background contribution and adjusting while varying their yields. The templates are derived from simulation and, whenever possible, in particular for the main background contribution, it will be validated on data, using control samples.

The templates are obtained from the simulation samples, by exploiting the full available statistics, given that the shapes of the training and test samples are consistent according to Kolmogorov-Smirnov test. The range of the observable, as well the binning scheme used has been chosen in order to avoid empty bins and to reflect the physics scenario. A variable-binning scheme will also be foreseen, to have equal statistics in each bin and avoid empty bins altogether. The statistical uncertainty on the templates can be accounted for by using the specific `RooHistFactory` tool, available within the `RooFit` package. Other uncertainties, *e.g.* variations due to different form factors, or data/MC differences, need to be considered separately as systematic effects.

The fit maximises the likelihood function that can be written as product of Poissonian Probability Density Functions corresponding to each bin whose number of entries is given by n_i. The expected number of entries in each bin depends on some unknown parameters: $\mu_i = \mu_i(\theta_1, ..., \theta_m)$. To avoid numerical problems, instead of maximising the extended likelihood function [238] it is more convenient to use $-2ln\mathcal{L}$. The function being minimised, in order to estimate $\theta_1, ..., \theta_m$ is the following sum:

$$-2ln\mathcal{L}(\vec{n};\vec{\theta}) = -2\sum_{i}^{n_{bins}}(n_i\, ln\mu_i(\theta_1, ..., \theta_m) - \mu_i(\theta_1, ..., \theta_m))$$

with i an index running over the bins, n_i, the number of observed events in the i^{th} bin, and $\mu_i(\vec{\theta})$ the number of expected events in the i^{th} bin. This number depends on $\vec{\theta}$, a vector of nuisance parameters and on the parameter of interest, *i.e.* the yield of signal and of each background component, which are free to vary in the fit. The expected number of entries in each bin, μ_i, is given by the superposition of the templates.

The analysis still needs to be finalised before proceeding to fit the data. Nevertheless the fit is used to perform feasibility studies of the $R\left(D_s\right)$ measurement by using pseudo-data samples generated using the templated distributions that are also used to fit.

6.6.1 Template definition grouping

As anticipated, the goal of the BDT to separate double charm is two-fold: it is used to remove some of the background from data by specifying a cut on the BDT output, but also it is used as one on the variables used to fit the final data, exploiting its discrimination power between the signal and the various types of backgrounds.

The templates for signal and different background categories are determined from simulation. Until this point, we have been using two categorisations for the decay candidates matched in Monte Carlo simulation data: one very fine-grained, described in Sec. 6.3.1 with around 40 categories depending on the hadrons in the decay chain, and a very coarse one, useful to have a synthetic view of the decays in the sample as in tables 6.11 6.12 6.13.

In order for our fit to converge, we need each of the categories to have a distinguishable distribution for the used variables, i.e. `tauY_2` (the τ proper decay time), `q2_2` (the corrected q^2) and `BDT_dc`, the output of the specific BDT trained to suppress the double charm decays. The coarse categorisation is not useful, as it is too simple to take into account the complexity of the background. The detailed categorisation is too complex, as a fit with tens of categories is unlikely converge, especially as some of the categories have similar distributions in the fit variables.

We therefore investigated how to group the different templates based on the distributions of the fit variables for each of them: for each variable, and each pair of categories, we use the two sample Kolmogorov-Smirnov test to derive the p-value for each pair of distributions as a measure of their similarity. All the background categories of Table 6.9 with less than 100 events are not considered for the moment since they shouldn't affect the results due to the low statistics and are grouped all together in a single group called "other". The Figures 6.21 and 6.22 summarise the obtained p-values.

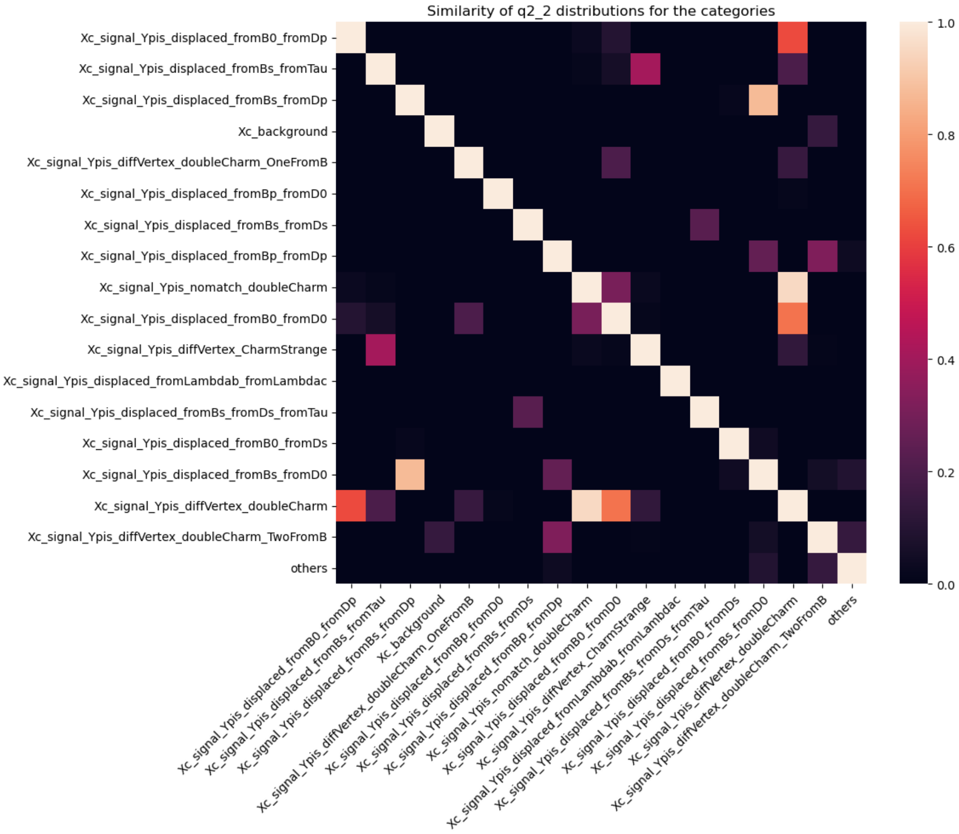


Figure 6.21: Similarity between the q2_2 distributions of the various background categories for candidates satisfying CUT1. The p-value of the Kolmogorov-Smirnov test is represented by the color code.

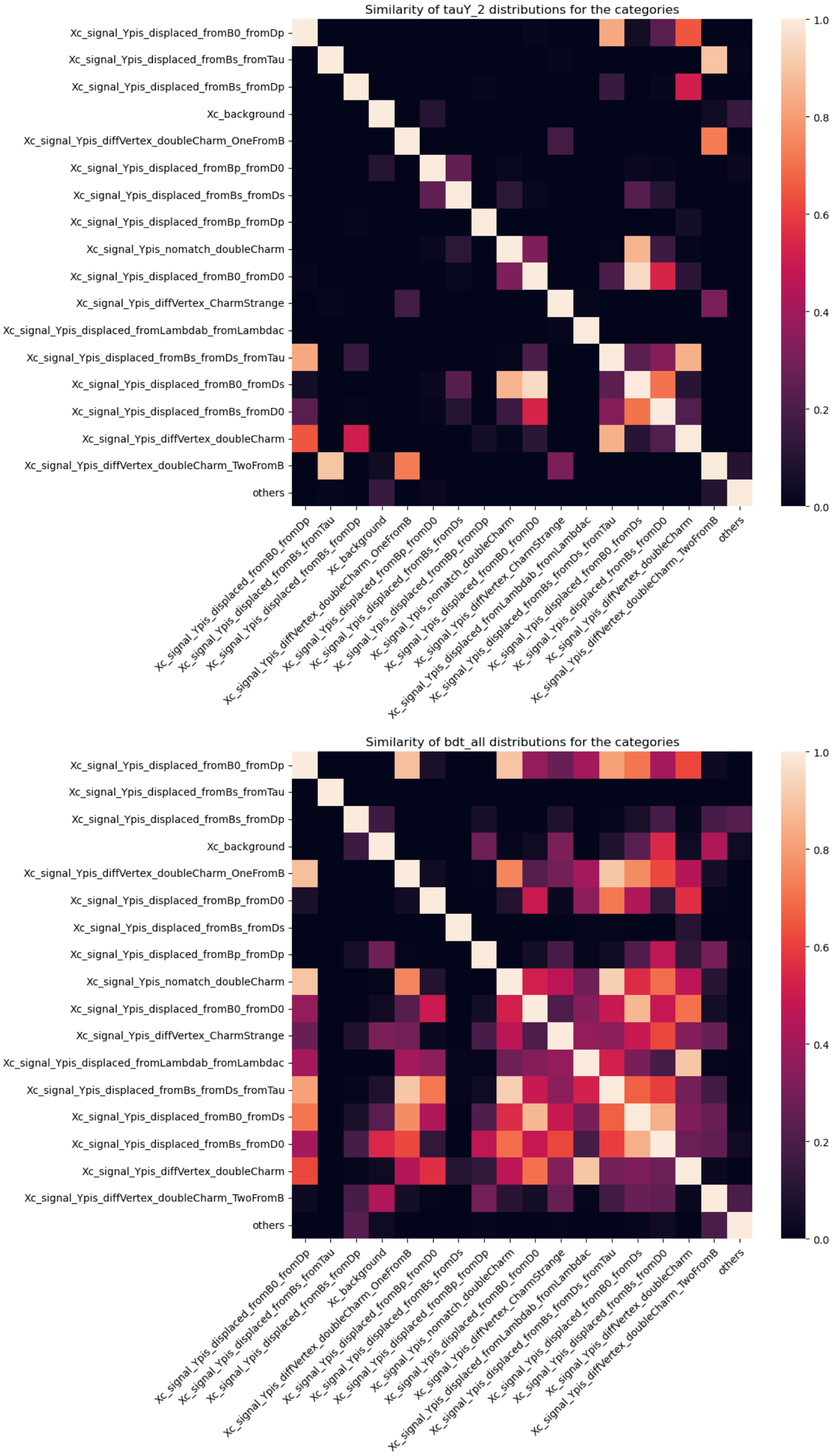


Figure 6.22: Similarity between the `tauY_2` distributions (top) and BDT double charm (bottom) of the various background categories for candidates satisfying CUT1. The p-value of the Kolmogorov-Smirnov test is represented by the colour code.

We then iteratively group together the templates for which the p-value is larger than a given threshold. Appendix E.4 shows the histograms grouped for a threshold value of 0.05 for the CUT1. Putting this together we get the categories listed in Table 6.14. The threshold is chosen to be 0.05 in order to limit the number of different templates to ten. Such choice can be eventually changed when studying the systematic uncertainties.

Template Name	Categories
DoubleCharm1	displaced_fromB0_fromD0, diffVertex_doubleCharm, displaced_fromB0_fromDp, nomatch_doubleCharm
DoubleCharm2	displaced_fromBp_fromDp, displaced_fromBs_fromDp, displaced_fromB0_fromDs, displaced_fromBs_fromD0
Signal	Signal
DoubleCharm3	diffVertex_CharmStrange, diffVertex_doubleCharm_OneFromB
BadDs	Xc_background
DoubleCharm4	displaced_fromBs_fromDs, displaced_fromBs_fromDs_fromTau
DoubleCharm5	displaced_fromBp_fromD0
Others	others
Lambdab	displaced_fromLambdab_fromLambdac
DoubleCharm6	diffVertex_doubleCharm_TwoFromB

Table 6.14: Template grouping.

The histograms for the three observables, for all the categories and for CUT1 are shown in Appendix, in Figures E.11, E.12 and E.13. Of course, these are projections of the 3D histogram which could correspond to different correlations among the observables. In order to check that the grouping is reasonable, the 2D histograms of `q2_2`, `tauY_2` and `BDT_dc` by pairs are shown in Appendix E.5.

Table 6.15 shows the grouping of the categories, and their fraction in the total as well as the expected yield in 2012 data of 2.0 fb^{-1}.

6.6.2 Pseudo-experiment studies

With the categorisation for the fit defined, and the templates extracted, preliminary pseudo-experiment studies provide useful information for the analysis:

- they allow to assess the feasibility of the signal yield determination, in particular show if there are critical instabilities of the fit that prevent to obtain a reliable result;
- they provide a preliminary estimate of the sensitivity of the $R(D_s)$ measurement, driven by the statistical uncertainty on the signal yield;
- they help finalising the selection and to choose among the three possible cuts defined in Sec. 6.4

Pseudo-experiment studies consist in generating 4000 samples of pseudo data. Each sample contains signal and background contributions with yields randomly generated according to a Poisson distribution around the expected values from simulation (see Table 6.15) corresponding

CUT	group contributions	fraction %	Yield (2 fb^{-1})
CUT1	DoubleCharm1	30.39	322
	DoubleCharm2	20.03	212
	Signal	10.74	114
	DoubleCharm3	10.43	110
	BadDs	10.05	106
	DoubleCharm4	6.65	70
	DoubleCharm5	5.49	58
	others	3.11	33
	Lambdab	2.09	22
	DoubleCharm6	1.02	10
CUT2	DoubleCharm1	33.67	2443
	DoubleCharm4	13.78	1000
	DoubleCharm3	13.2	958
	DoubleCharm2	12.71	922
	BadDs	9.19	667
	DoubleCharm5	8.26	599
	Signal	3.31	240
	Lambdab	2.67	193
	others	2.45	177
	DoubleCharm6	0.76	55
CUT3	DoubleCharm1	33.48	1440
	DoubleCharm2	14.68	631
	DoubleCharm3	12.79	550
	DoubleCharm4	11.14	479
	BadDs	9.56	411
	DoubleCharm5	7.57	325
	Signal	4.81	207
	Lambdab	2.58	111
	others	2.55	109
	DoubleCharm6	0.82	35

Table 6.15: Fraction and expected yield in 2 fb^{-1} for each category used in the fit, for the three cuts considered. The yield was computed by comparing data and the incl_$H_b \to D_s 3\pi X$ simulated sample.

to what is expected for the data recorded in 2012 by LHCb (2 fb^{-1}). The pseudo-random data is generated according to the templates for each category and fit with the PDFs corresponding to the same templates. The floating parameters are the yields of signal and of each background component, though we are only interested to the signal. Figures 6.23, 6.24 and 6.25 present the fit results to the pseudo-experiments, where the distributions of the signal yield and of its error and of the pull are shown [4].

The results are in agreement with the expected value, except for a few cases where the fit returns a signal close to zero, the fitting errors being at a level less than one per mille. The pulls follow a normal Gaussian distribution, with mean and sigma values consistent with 0 and 1, respectively. This gives confidence that the result of the fit to the data sample will be reliable both for the value and the uncertainty.

The signal yield is determined with a statistical precision of $\sim$ 37%, 43% and 39% for CUT1,

[4] The pull quantity P for a fit parameter N_{fit} is defined as the residual from the input N_i normalised for the fit error σ, $P = (N_{fit} - N_i)/\sigma$, for each bin

CUT2 and CUT3, respectively. The lowest statistical precision is obtained using the tighter cut CUT1.

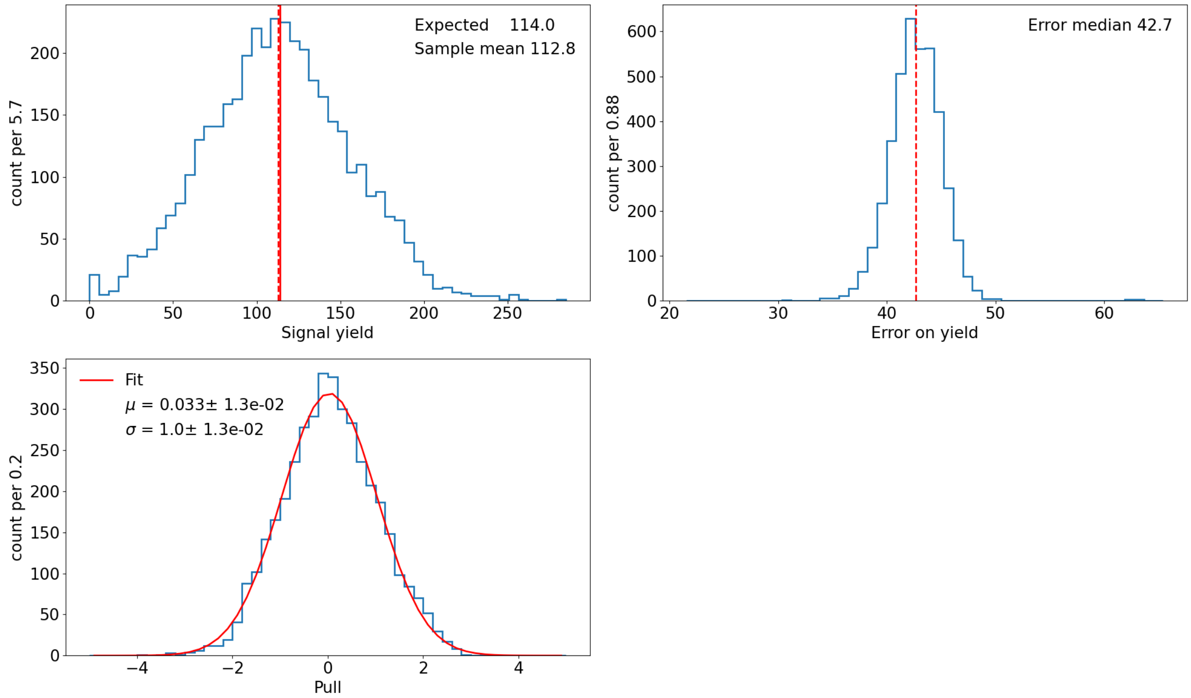


Figure 6.23: Result of 4000 pseudo-experiments run with the templates for CUT1.

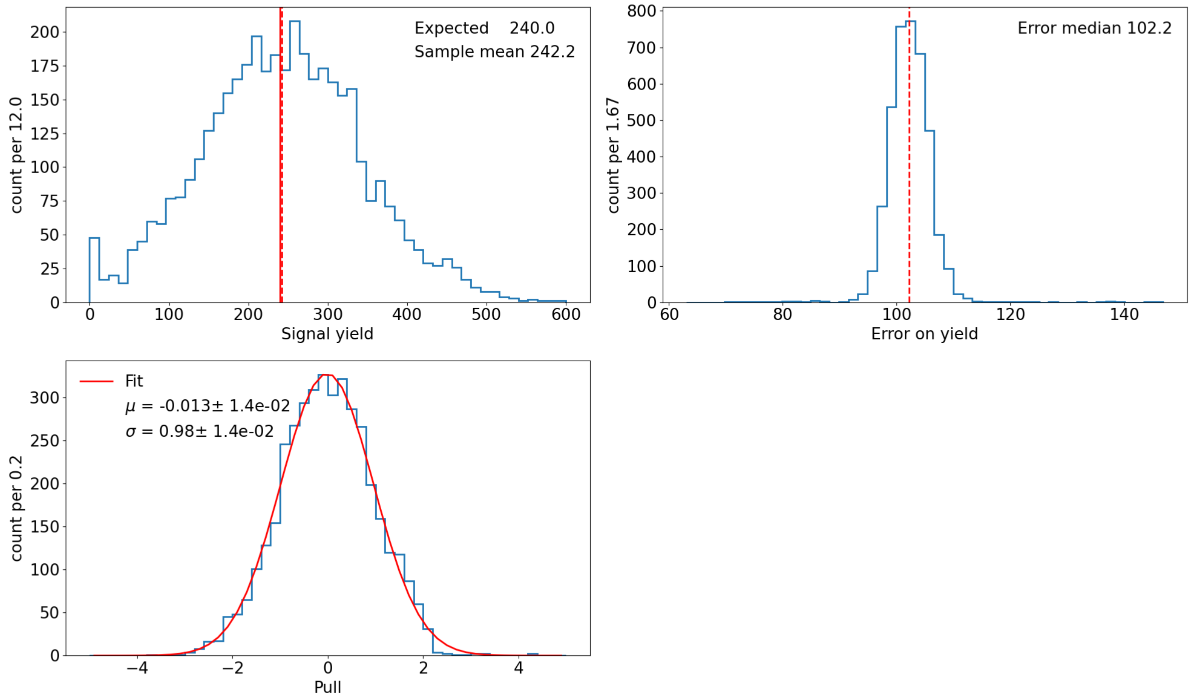


Figure 6.24: Result of 4000 pseudo-experiments run with the templates for CUT2.

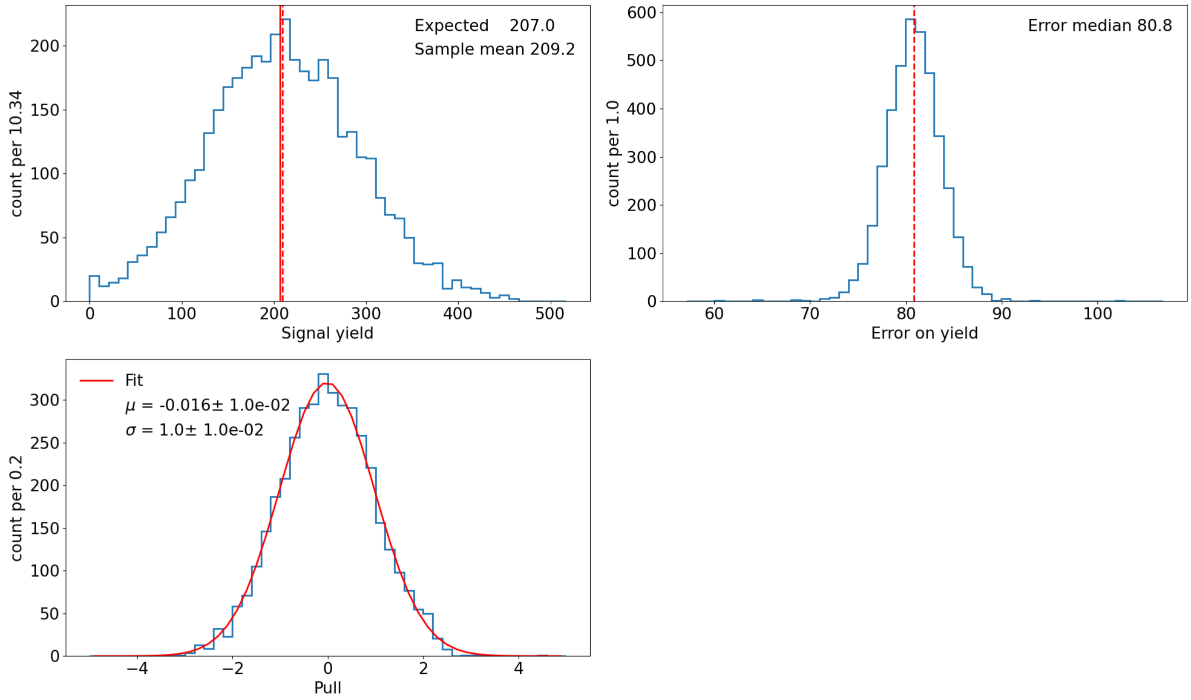


Figure 6.25: Result of 4000 pseudo-experiments run with the templates for CUT3.

Figure 6.26 shows the results of the fit for the three variables, when fitting the overall inclusive MC sample (incl_$H_b \to D_s 3\pi X$), corresponding to a statistics of 25fb^{-1}, after filtering with the three cuts. Table 6.16 shows the actual number of candidates per category, the output of the fit, including the deviation between the actual number and the fit result (in standard deviations of the fit error). The results are in accordance with the the study performed previously. Some of the backgrounds yields deviate from the fit value, but with no impact on the signal yield. Of course, the precision of the signal yield is much better in this test because the equivalent statistics is larger. However, LHCb plans to have a larger sample of data recorded by the end of LHC Run 3 (and Run 2 and Run 3 data were recorded at higher energies than Run 1, therefore their yields per fb^{-1} are higher than for Run 1).

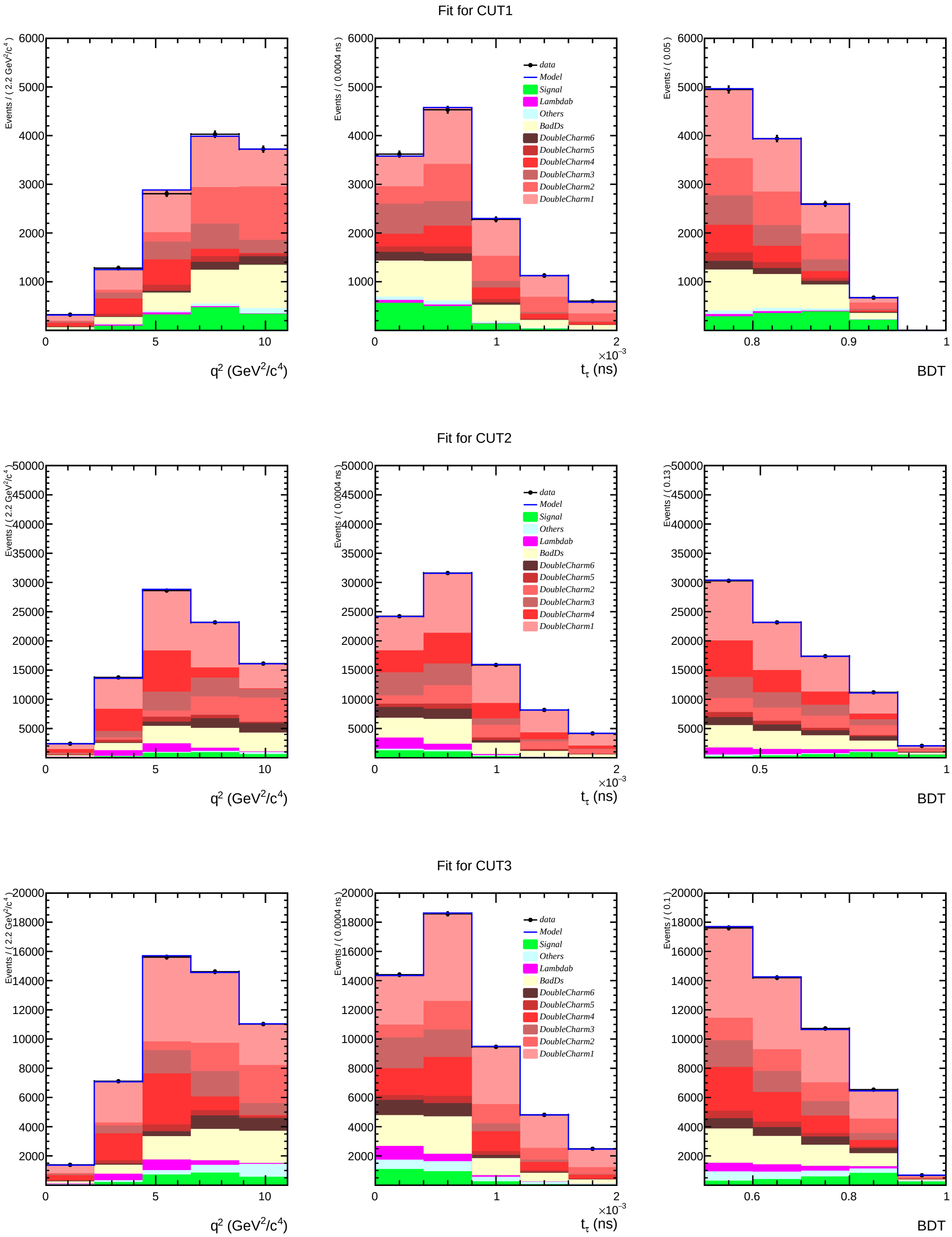


Figure 6.26: Result of the fit on the inclusive-b Monte Carlo sample with CUT1, CUT2 and CUT3, template naming according to Table 6.14.

CUT	group contributions	count	Fit result	Difference (σ)
CUT1	DoubleCharm1	4044	3210±500	-1.7
	DoubleCharm2	2665	2110±320	-1.7
	Signal	1429	1260±150	-1.1
	DoubleCharm3	1388	1300±330	-0.3
	BadDs	1337	2150±530	1.5
	DoubleCharm4	885	1070±240	0.8
	DoubleCharm5	731	360±400	-0.9
	Others	414	200±280	-0.8
	Lambdab	278	100±140	-1.3
	DoubleCharm6	136	390±210	1.2
CUT2	DoubleCharm1	30590	28490±1510	-1.4
	DoubleCharm4	12524	13350±1130	0.7
	DoubleCharm3	11996	9230±1190	-2.3
	DoubleCharm2	11547	8790±870	-3.2
	BadDs	8352	11080±2460	1.1
	DoubleCharm5	7508	2170±1900	-2.8
	Signal	3005	2780±210	-1.1
	Lambdab	2424	3160±430	1.7
	Others	2223	810±790	-1.8
	DoubleCharm6	691	4200±830	4.2
CUT3	DoubleCharm1	18037	16810±1260	-1.0
	DoubleCharm2	7910	5410±690	-3.6
	DoubleCharm3	6889	4760±870	-2.4
	DoubleCharm4	6002	6680±850	0.8
	BadDs	5151	6720±1830	0.9
	DoubleCharm5	4078	1240±1500	-1.9
	Signal	2592	2420±190	-0.9
	Lambdab	1390	1580±330	0.6
	Others	1373	1870±820	0.6
	DoubleCharm6	444	2260±630	2.9

Table 6.16: Number of candidates per category and fit result for the inclusive MC data, for CUT1, CUT2 and CUT3.

6.7 Status and outlook

In summary, a preselection of the $B_s^0 \to D_s^- \tau^+ \nu_\tau$ decays has been developed to keep the signal and normalisation candidates. It uses various cuts and multivariate selections of B_s^0, D_s^- and 3π, and charge isolation of the signal, developed on LHCb Monte Carlo simulation data for 2012.

A categorisation of the various background categories was established, with enough granularity to study a sample of inclusive b-hadron decays to D_s^- and 3π final states in details. A cut was optimised to separate the signal candidates based on the distance between the 3π vertex and the B_s^0 vertex. The sample left at that point contains a majority of candidates corresponding to double charm decays which cannot be separated from τ decays using the vertex separation cut as they have similar lifetimes. In order to separate them, a decision tree to separate signal from double decays was developed, that uses kinematic variables assuming the candidate was a true double charm decay. The output of the BDT is used to filter the data, but the remaining sample still contains a minority of signal. A model of the distributions of the signal and various background categories for the BDT output, the τ proper decay time and the q^2 was established based on simulation data. The signal yield can be extracted by fitting the distributions of selected data using the model. Toy simulations and a fit to the inclusive Monte Carlo sample showed that the result is reliable and allows an estimate of the signal yield sensitivity. It however includes a component due to D_s^* decays which cannot be disentangled from $B_s \to D_s \tau \nu$ by the current processing chain.

This approach relies on the validity of the fit model, which needs to be verified on data, using control samples, following the approach taken to measure $R(D^*)$ [224] and adapting it to this measurement. In fact, the templates of the different signal and background sources are extracted from simulation, which is only a good approximation of the true pp collision processes. Many aspects of the simulation are estimated or guessed because the measurements are not available or known with poor precision. It is therefore necessary to validate the fit model and evaluate all the related systematic uncertainties.

Chapter 7

Feedback on analysis tools

The tools presented in Chapter 4 were developed and tested during the course of the $R(D_s)$ measurement detailed in Chapter 6. This provided the opportunity to apply the computing tools in a real physics case used as a "testbed", thus rising a number of questions and issues with the current infrastructure and with the proposed improvements that are described in this chapter.

7.1 Considerations on the proposed process

7.1.1 Data analysis is not software development

While this title states the obvious, the intensive use of computing tools underlined in this thesis should not confuse analysts about the final objective: while Git, GitLab and continuous integration systems have been shown to be very useful for data analysis, the difference in goals compared to software development leads to different compromises regarding their use.

It would not be pragmatic to treat all scripts used for data analysis as software that should be reusable. The $R(D_s)$ analysis was split into a list of steps as illustrated by Figure 6.20. Each of these steps starts with a phase of data exploration and validation, before deciding on the actual processing to be performed and then moving on to the next stage. For example, before deciding on the cut to separate signal and normalisation candidates in Section 6.4, many investigations were undertaken to check the data at hand and evaluate the efficiency of the cut. It should be possible to track the provenance of the artefacts of those investigations and re-create them, but this does not imply that the tools involved are generic enough to be included in the LHCb software stack. Furthermore, during this phase, analysts should be able to choose the tools they prefer to allow for more creativity in their research. Within the LHCb collaboration, some physicists prefer the ROOT framework, some prefer tools from the Python ecosystem. In any case, all these tools are available within the analysis environment (see Section 4.5.2).

In a second stage, however, a pipeline is prepared that prepares plots, filtered or enriched versions of the data or any other artefacts needed for the analysis. This pipeline often has to be run on multiple datasets (e.g. for different years), and software tools have to be developed to perform those actions. Those tools require more care and testing and it is worth integrating them in a workflow to be able to track the data artefacts. In the $R(D_s)$ analysis, a number of jupyter notebooks [239] were used to analyse the data, and a workflow put in place to prepare ntuples with the `B_Y_SEP` cut (Section 6.4). The tools used in the workflow can be treated as software products and the adequate development methods applied. Integrating analysis scripts in Snakemake workflows takes time, but best practices can really ease this task. For example, making sure that the scripts have configurable input and output locations (in Python this is easy to implement using the standard Python module `argparse`) and that parameters can be modified via the command line.

However, it is clear that the important outcome of the process is the measurement with associated plots and datasets, and it is up to the analysts to organise the process and to decide what should be integrated in a reusable workflow.

7.1.2 Workflow structure

LHCb data analyses are complex and consist of many different steps. Unifying them in one single workflow for the whole analysis is impractical for a number of reasons. First, a large workflow file quickly becomes difficult to read and makes it harder to identify the reason for each step. Second, the full workflow is not necessarily linear, many checks are done which are not part of the "main" branch and can be viewed separately. Finally, Snakemake cleans the targets it intends to rebuild at the start: starting the workflow by mistake can result in files being deleted. They can of course be rebuilt but this causes unnecessary processing and forces the analyst to wait. For example, in the $R(D_s)$ analysis, re-running the PID regeneration for a whole sample by mistake can imply waiting for several hours.

For this reason, we took the approach to split the analysis in a number of *processing stages*, each with their own workflows. Looking at the repositories for published analysis, this approach was also taken by many analyses already using Snakemake. This forces workflows to be focused, and reflects the organisation of the analysis work.

This structure requires documentation that explains the role of each stage and its place in the overall processing required for the analysis. It however does not prevent the construction of a top level workflow that invokes rules from the smaller scope workflows for each stage (e.g. using the Snakemake *module* functionality).

The $R(D_s)$ analysis code was split into various stages:

- the post processing of the Analysis Production (adding the BDT_Ds, BDT_Bs and BDT_3pi information as well as the corrected kinematic quantities and, for MC, the PID information with PIDGen; applying several data reduction cuts),
- the checks the data after post processing (including the `B_Y_SEP` study),
- the BDT training and validation to separate the double charm background,
- the decay categorisation using the BDT and the export of the templates for the final fit,
- the final fit itself.

7.1.3 Data structure

All the workflows are linked via their dependencies, so just splitting the Snakemake workflows makes them clearer to understand but does not prevent the accidental deletion of part of the data, forcing a re-run of several workflows. For more clarity, the output of each $R\,(D_s)$ workflow was organised in its own directory structure (with the name of the workflow itself, to make provenance clearer) with two sub-directories, with the following naming convention:

- The *output* directory where the workflow writes its own output.
- The *validated* directory, containing a manual copy of the *output* directory when its data has been reviewed, presented and discussed within the analysis team. The workflow for the next stage in the processing should always depend on this validated version of the output.

With this structure, it is possible to decouple the work and for different analysts to improve on workflows separately, as each workflow's input is validated and stable. In Snakemake, configuration parameters can be loaded from JSON or YAML files, allowing for workflows where it is possible to easily switch the input directory for a rule.

7.1.4 Derived files sharing

Sharing the data within LHCb can be done using the EOS storage system at CERN, accessing the files via the XRootD protocol, as each analysis has its dedicated directory structure. Snakemake wildcards force consistency in the naming conventions for the rules concerning the input and output files, which makes provenance identification easier. It is also possible to query the rule dependency graph from Snakemake itself.

Nevertheless, the location of the files currently has to be hard-coded, or code should be developed to add some flexibility to the workflow. It is possible to switch from a remote copy to a local copy of the files by changing the workflow configuration but this is not always very practical. The structure in place with the output/validated versions of the output files is a very simple and impractical form of versioning of the data files. A more practical way to perform this versioning would be welcome.

On a side note, Snakemake enables a copy of public intermediary artefacts to be kept (a functionality documented under the name *Between workflow caching*). This can be useful within a large workflow, but if the analysis workflows are split as in Section 7.1.2 it is not necessarily the case.

7.1.5 Integration with the continuous integration system

The continuous integration system available with GitLab is very useful to run the applications and check that the scripts committed to the repository are functional.

Snakemake rules can depend on both the data they process and the code that processes it. In that case, Snakemake decides whether to re-run steps of the workflow based on the file modification times on the filesystem. This is valid as well for files stored on EOS, as the XRootD remotes can query the file metadata.

A priori, when invoked by the continuous integration system, Snakemake can re-run only what has changed, provided the timestamps of the files being tested reflect their last modification time. This is not the default behaviour when getting files from a Git repository, but it is not difficult to do (by setting a file last modified date on the filesystem to the date of the last change in Git).

In any case, re-running all the steps in the $R\left(D_s\right)$ analysis was not necessarily desirable. The steps dedicated to running a specific version of external tools such as the PID calibration, will have identical outputs. There is therefore no point in automatically re-running them in a CI system. Furthermore, computing intensive workflows, for example the optimisation of machine learning algorithms, may require more resources than what is offered by continuous integration tools, or may be difficult to setup due to the lack of adequate hardware.

Modularising the workflows allows re-running only parts of the workflows that make sense, and give the analysts the possibility to decide. For example, it can be useful to rerun workflows generating plots and reports every time the analysis tools are updated, to make sure that all the code was committed to the repository and that it is functional in a standard environment.

Delegating credentials to the continuous integration system in a sustainable and safe manner is an issue that was difficult to solve but the solution detailed in Chapter 4 has proven to be reliable and effective. From a computer security point of view, it avoids the proliferation of user accounts and keeps track of the access tokens delegated. As tokens need to be concatenated to the file names, using them require code changes. Fortunately the `apd` utility described earlier on provides an abstraction layer that is practical to use. The use of tokens for data access is not restricted to analysis code, and can be used by any LHCb GitLab project. This functionality is proving popular and new projects requiring data access are adopting it.

7.1.6 Derived Analysis Productions

Sharing the file on a common storage system as described in Section 7.1.4 however does not encourage reuse of the files outside of the analysis team that produced them. Indeed, the generated data is not *Findable* as recommended by the FAIR principles. For this purpose, it would be useful to be able to derive samples from Analysis Productions and register them in the bookkeeping database in the same manner as Analysis Productions.

The constraints on those *Derived Analysis Productions* are that:

- the input data should be the output of an identified Analysis Production;
- the script environment should be fully specified (as is the case for the LHCb conda environments);
- the code used should be fully tracked to ensure reproducibility;
- the data should be the result of a controlled production, to ensure that the files registered match the code tracked.

This could be done for example by asking users to provide:

- the dataset to process, specified as an Analysis Production, with a list of tag name/tag value to filter the data.
- a script that takes an input file and writes out the output, without accessing the external environment.

The script and the input data descriptions could be kept in a GitLab project as is done by Analysis Productions, using GitLab CI to derive the data instead of creating DIRAC productions. The CI pipeline would then be able to register the data in the bookkeeping database on behalf of the analysts, propagating tags from the source Analysis Production and adding new ones.

As the resources are limited in continuous integration systems in comparison with the WLCG, derived AP should be reserved for cases where the data is filtered so that we produce little data.

7.2 Feedback on Analysis Productions and `apd`

7.2.1 Deriving ntuples from LHCb production data

The LHCb data analysis application, DaVinci, was used to extract information about the decays relevant to this analysis. The tools available by default did not allow for detailed inspection of the information needed from the Monte Carlo data (namely the complete list of ancestors of the final state pion), so the custom tool *TupleB2XMother*, a C++ TupleTool developed for the $R(D^*)$ analysis, was implemented. While it is possible to run user's built C++ libraries in grid jobs launched using Ganga, Analysis Productions require the use of tagged and released LHCb projects only. In order to introduce Analysis Productions for this analysis, the code *TupleB2XMother* was improved, brought up to the standard required by common LHCb software and merged in the common analysis software project. While doing this seems simple enough, it however took several weeks due to various reasons; among them, the code was not optimal and could be improved. While the code was functional, it was not advisable to integrate it in the official LHCb software stack without the improvements requested by the release managers. Furthermore, each new change to the stack, has to be tested in the context of the whole stack (this is done on the same day) and releases incorporating new changes are done every few weeks (though they can be done quicker on-demand).

Integrating code in the stack therefore requires anything from one to several weeks, before it can be used in Analysis Productions. This procedure has the advantage that the code is tested and reviewed, but this comes with a delay which is not necessarily compatible with the time frames expected during data analysis. It would be beneficial for analysts to have a way to run custom tools, keeping the traceability of the software without necessarily submitting it as a common tool for all LHCb members to use. This could for example be done by submitting those tools alongside the Analysis Production configuration itself, and providing a way for the LHCb software stack to build them (this is not possible at the moment).

7.2.2 Metadata attached to Analysis Production

The Analysis Productions described in Section 3.3.1 automatically add tags to the created samples according to the common parameters specified during the processing such as the magnet polarity, the event type in case of simulations, etc. Extra information could be automatically extracted from the configuration of the jobs run by the Analysis Production. Preliminary investigations show that it would be possible to extract the particle decays matched and the name of the stripping/sprucing lines they are taken from. Matching them with the stripping/sprucing configuration, extra information can be found such as the cuts applied on the various quantities.

7.2.3 Common tools in Analysis Productions

There are a number of common tools used by LHCb analysts to calibrate the data from the detector, such as the Momentum Scaling tool which allows re-calibrating the momentum of the particles based on the reconstruction of several narrow mass peaks. It is especially used in analyses where the best momentum or mass resolutions are vital. There are also tools to correct or regenerate the Particle Identification (PID) variables according to calibration samples information. In this analysis, the PIDGen tool was used to regenerate the PID information in MC samples before selecting the events to keep.

This had to be done separately for each file produced by the Analysis Production, due to limitations in the PIDGen tool which otherwise uses too much memory. Obviously, this was done before any further processing and before the data could be used by the members of the analysis teams. This lead to the conclusion that:

- all data processing that enriches the data set (e.g. adding a column with the output of a BDT) should be done as part of an Analysis Production. It can of course be done as part of a Snakemake workflow, but this adds complication to the workflow itself: with the $R(D_s)$ analysis, the unique identifier for each file (called GUID, which is part of ROOT files) was used to track artefacts derived from it.
- it would therefore be useful to add the option to process data as a separate step within the Analysis Production. This is not currently possible and would require a new functionality in the *lbexec* tool used to invoke applications. The first candidates for integration would of course be tools developed by LHCb working groups, such as the PID correction and calibration tools (PIDGen, PIDCalib, PIDCorr), but it does not need to be limited to them.

This approach does not necessarily fulfil all common use cases: for example using machine learning algorithms requires the Analysis Production to be available first to train the code based on its output. Once this is done, the AP has to be reprocessed to add this extra column. In this case, introducing the notion of *derived Analysis Production* would be useful.

7.2.4 `apd` python package

The `apd` python package was introduced in the course of the hadronic $R(D_s)$ analysis to process the output of the Analysis Production used to extract the data. `apd` proved to be very practical and useful, even though its utility is restricted by the fact that post-processing of the data is needed before further use by all the members of the team. Derived Analysis Productions as proposed in Section 7.1.6 would alleviate this problem. A number of features are also still missing in `apd` and will be introduced when time allows. The introduction of a date at which to query the data is needed to process properly evolving datasets, this functionality is currently offered by the Analysis Production database but is not exposed in the client.

7.2.5 Use of Jupyter notebooks

Jupyter notebooks [239] are a popular way to analyse data. They allow the results of investigations on a dataset to be recorded and have proven useful during the first, exploratory session of the analysis. They can be integrated within Snakemake workflows to produce reports from continuous integration systems. They are not suitable for large amounts of code and cannot be reused. They are therefore useful for data exploration but their content often needs to be refactored before further use. In any case, their use nonetheless remains a personal choice and many analysts prefer simple scripts to produce plots and extract statistics.

7.3 Preserving analysis pipelines and outlook

Figure 7.1 illustrates the workflow of the $R(D_s)$ analysis already shown in Figure 6.20, with the addition of the improvements suggested in this chapter. The steps in grey indicate the data registered in the bookkeeping database. Allowing the PIDGen step to run within the Analysis Production would relieve a burden from analysts, and derived Analysis Production would indeed make sharing artefacts easier. It is clear that derived Analysis Productions would also introduce complications which are only desirable once it has become clear which processing should be performed (derived Analysis Productions would not be suitable for an exploratory phase).

However, it is striking to notice that while the output of Analysis Productions is itself versioned, the tools in place to process the ntuples version the code, but not the data itself. In fact, a weak form of versioning was introduced with the two folders `output` and `validated`.

Versioning the data files with Git is not technically feasible, as git repositories are not designed to hold large amounts of data. An extension of Git, called Git Large File Storage (Git-LFS [240]) allows files to be stored on a dedicated server, while keeping a local index of the files available remotely. The architecture of this solution is appealing but does not integrate with the storage infrastructure available to the LHCb experiment (using the XRootD protocol to access the data). Furthermore, Git-LFS copies files locally (using the HTTP protocol) which is something in general to be avoided as analysts potentially process very large amounts of data.

Implementing a software layer that allows one to keep track of various versions of the files, keeping track of the metadata using Git is possible. This could be integrated in `apd` which

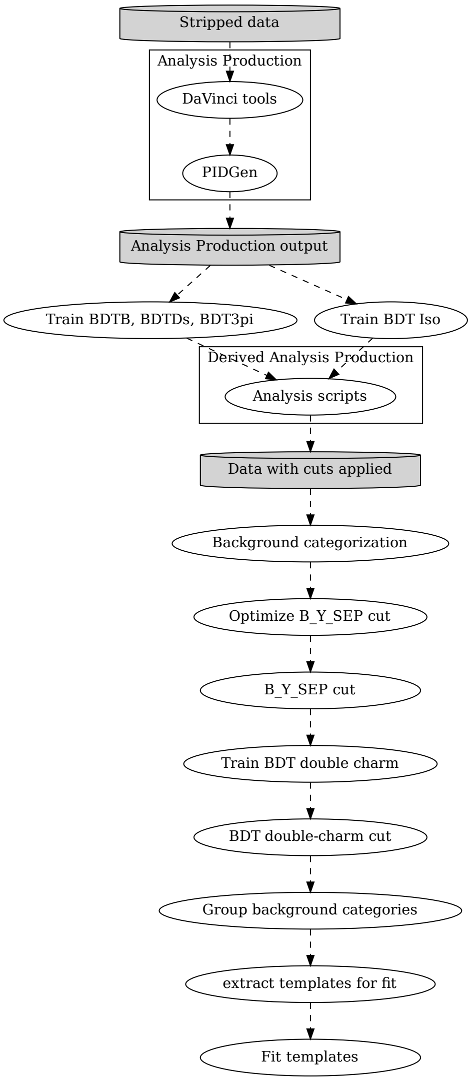


Figure 7.1: Envisaged workflow to process the Monte Carlo data for the $R\left(D_s\right)$ analysis, taking into account the improvements suggested in this chapter. The steps in grey are the data items shared on the EOS filesystem.

already provides an abstraction layer to locate files that can be used in python scripts and the Snakemake workflows.

A number of tools with these features are already available in the context of data science and machine learning, to track model development and data pipelines. They are classified as *MLOps* tools, short for *Machine Learning Operations* and aim to provide a way to build and deploy machine learning models in a reliable way. They use the same tools as *DevOps* tools, used to assist software development but with a shift towards machine learning models. They include MLflow [241], Pachyderm [242] or Data Version Control (DVC [243]). This is by no means an exhaustive list as the activity in this field is linked to the general increase in use of machine learning. They generally make use of the same underlying infrastructure as DevOps tool such as

Git for version control. While these tools are interesting, they do not all apply to data analysis at LHCb, as some are very focused on machine learning development with a number of libraries (XGBoost, TensorFlow...) or force specific infrastructure to run the models (e.g. for the storage of large files). Some are targeted to allow repeat training of fixed models on changing data, a use case very different from physics data analysis. One last point is that LHCb uses the HEP specific XRootD protocol, which is not recognised by industry tools, although some of those tools provide ways to extend the protocol they support (and this work was also done for XRootD in Snakemake). Among those tools, DVC is lightweight, provides a way to extend its remote file support to XRootD and would be a good candidate for further investigations.

As a conclusion, the tools developed for this work have proven to be useful in the context of the $R(D_s)$ analysis, and are also being adopted by other analysts in the collaboration. The developments sketched in this section would further improve the software tools preservation system. External tools from the domain of MLOps should also be investigated to see whether they can be applied effectively within LHCb and HEP experiments in general.

Conclusion

Performing a physics measurement at LHCb is a tremendously complex task that involves many different aspects, from operating the accelerator and the detector to the data acquisition, event reconstruction and lastly data analysis. Each of these steps has to be be traced and checked to guarantee the correctness and reproducibility of the result.

The goal of this thesis was to follow two axes of work: perform a measurement of Lepton Flavour Universality on B_s^0 meson decays at LHCb, while establishing an inventory of the software tools and methods available for analysis preservation, from the point of view of a software developer, and subsequently improve the tool set where needed.

This work, therefore, furthered on previous efforts to perform the measurement of the $R\,(D_s)$ ratio at LHCb: using simulated data, it refined the decay candidates categorisation in order to be able to model different types of background. A decision tree was trained to identify signal from double charm background which is used both to filter the data and as an input to a three dimension fit (alongside the q^2 and τ lepton proper decay time). This fit was run on toy simulations, as well as on the existing simulation data; this showed that it allows to extract the signal yield reliably within the available samples. Of course, this procedure relies on the validity of the fit model, dependent on the accuracy of the simulation, and therefore cannot be trusted without validation of control channels. This has to be done and systematic uncertainties have to be evaluated before being able to estimate $R\,(D_s)$ itself.

A review of the tools available within LHCb, as well as in other LHC experiments, shows that many efforts have been made to help analysts make their work reproducible, and that many tools are available, but that some gaps can nonetheless be identified. This work introduced new software which can help tracking the provenance of the measures and files derived during the analysis of LHCb data, as well as the integration with software development tools which can help with the analysis process (e.g. in terms of authentication and authorisation to access the data). The python package `apd` developed for this thesis is experiencing a rapid uptake within the experiment. In turn, its adoption brought new ideas for tools that could help analysts, thus driving future developments.

In the course of this work, significant progress has therefore been made, both in the measurement of $R\,(D_s)$ and in analysis tools compliant with the FAIR principles, but these are considerable tasks and this is not yet the last chapter. Nevertheless this effort helped identify the road ahead and further efforts along the traced path will continue in order to validate Lepton Flavour Universality at LHCb in a reproducible way.

Acknowledgements

This thesis builds on the efforts of the LHCb collaboration members, past and present. I've rarely seen a place where people are so welcoming and keen to help newcomers, in order to achieve such a unique feat of science and technology. The list of people I owe much to is too long for this page, so I would like to extend a general thanks to the LHCb collaboration and to all the members of CERN who helped me.

First and foremost, this work would not have been possible without my supervisors, C.Bozzi and S.Vecchi. I cannot thank them enough for their support and guidance. They were always available, taught me a lot and made my stays in Ferrara so enjoyable! They, as well as G.Zavattini, helped me to discover the university as well as a superb city and I enjoyed the long discussions.

Working on the analysis was extremely interesting, and many thanks to C.Giugliano, A.Scarabotto and B.Siddi as well, it was a pleasure to work with them.

I am also indebted to my CERN supervisors, M.Cattaneo, C.Gaspar and M.Clemencic who kindly supported me in this atypical thesis, and to N.Koivunen, L.Taillieu and G.Guinot from the CERN HR department who helped sort out the administrative complications.

It was a pleasure to participate in excellent lessons from the University of Ferrara ! I am very grateful to all the teachers and especially L.Pappalardo for taking the time to answer all my questions. It would have been nicer to have the lessons in presence rather than remotely but the teachers made incredible efforts during the difficult COVID period.

Special thanks go to C.Burr, for the long discussions on Analysis Preservation while cycling around the Geneva countryside (and also not begrudging my haphazard navigation skills). The chapters on preservation and the tools developed for this thesis would not have been the same without him.

I would also like to thank V.Gligorov, Y.Amhis and N.Skidmore who encouraged me and helped me by answering questions and commenting on the thesis, as well as Prof. D.Costanzo and Prof. A.Pompili who took the time to review it, and provided suggestions for improvements.

And finally, I am so grateful to Suzanne and Isabelle for the lepton flavour jokes and for bearing with me during all the evenings when I was working on this thesis.

List of Figures

Glossary

Bubble chamber Early type of detector used to track particles in high-energy particle collisions. It consists of a container filled with a liquefied gas, which can be momentarily superheated. When crossing the detector, charged particles produce tiny bubbles along their track and a camera can therefore be used to record their trajectories.

Cross section The cross section of a physical process is a measure of its quantum mechanical probability. Multiplying it by the luminosity of the beam gives the total number of interactions expected for this process.

DPHEP Data Preservation In High Energy Physics study group created in 2009 to investigate the methods and efforts to preserve the data recorded at the Large Hadron Collider at CERN.

DST Data Summary Tape, historical name given in High Energy Physics to files containing reconstructed events.

FAIR The FAIR Guiding Principles for scientific data management and stewardship, as defined by Wilkinson et al. in 2016[7]. It stands for Findable, Accessible, Interoperable and Reusable.

HLT1 High Level Trigger level 1. First stage of the event filtering system in LHCb. It performs a partial track reconstruction and keeps interesting events at a rate of 30 MHz.

HLT2 High Level Trigger level 2. Second stage in the LHCb trigger that performs a detailed reconstruction of the events selected by the HLT1 and select those with interesting physics for further analysis.

Instantaneous luminosity The instantaneous luminosity $\mathcal{L}$ is a fundamental parameter of a particle accelerator which quantifies the density of particles in the colliding beams. It is used to derive the number of particle collisions per unit of time. A higher luminosity means a greater likelihood particles will collide and result in a desired interaction. For a given process, the total number of interactions occurring in particle collisions is $N = \sigma \int \mathcal{L}(t)dt$, σ being the cross section of the process, a measure of its quantum mechanical probability.

Integrated luminosity Integral of the instantaneous luminosity over a data taking period.

Logical File Name Unique identifier for the files referenced in the LHCbDIRAC bookkeeping system. Can be used to query the system for the disk location of this file, of which there may be several, returned as a Physical File Name. A LFN is typically of the form: /lhcb/LHCb/Collision12/DATA_BS.ROOT/00173027_1.data_bs.root.

Physical File Name File name for a copy on disk of a logical file name in the LHCb Dirac data mangement system. This Unique Resource Identifier (URI) allows accessing the file with the protocol specified in the PFN. For example, for a file to be accessed with the XRootD protocol on the EOS storage at CERN, it can be of the form: root://eoslhcb.cern.ch//eos/lhcb/Collision12/DATA_BS.ROOT/00173027_1.data_bs.root.

Run 1 LHC operation period 2010-2012.

Run 2 LHC operation period 2015-2018.

Run 3 LHC operation period 2022-2025.

Acronyms

CVMFS CERNVM Filesystem.

DAQ Data Acquisition.

FCCC Flavour-Changing Charged-Current.

FCNC Flavour-Changing Neutral-Current.

GPGPU General-Purpose Graphics Processing Unit.

HEP High Energy Physics.

LFN Logical File Name.

LFU Lepton Flavour Universality.

LHC Large Hadron Collider.

PFN Physical File Name.

QCD Quantum ChromoDynamics.

QED Quantum ElectroDynamics.

SPS Super Proton Synchrotron.

WLCG Worldwide LHC Computing Grid.

Bibliography

[1] C. L. Smith. "Genesis of the Large Hadron Collider". In: *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 373.2032 (Jan. 2015), p. 20140037. DOI: 10.1098/rsta.2014.0037. URL: https://doi.org/10.1098/rsta.2014.0037.

[2] The LHCb Collaboration. *The LHCb Collaboration*. https://lhcb-outreach.web.cern.ch/collaboration/. (Accessed on 05/16/2023).

[3] CERN. *Convention for the establishment of a European organization for nuclear research: Paris, 1st July, 1953 : as amended.* Geneva: CERN, 1971. URL: https://cds.cern.ch/record/330625.

[4] CERN. *CERN Open Data Policy for the LHC Experiments.* Tech. rep. Geneva: CERN, 2020. DOI: 10.17181/CERN.QXNK.8L2G. URL: https://cds.cern.ch/record/2745133.

[5] European Commission. *Open Science.* https://research-and-innovation.ec.europa.eu/strategy/strategy-2020-2024/our-digital-future/open-science_en. [Accessed 27-Jun-2023]. 2023.

[6] OSTP. *FACT SHEET: Biden-Harris Administration Announces New Actions to Advance Open and Equitable Research.* https://www.whitehouse.gov/ostp/news-updates/2023/01/11/fact-sheet-biden-harris-administration-announces-new-actions-to-advance-open-and-equitable-research/. [Accessed 27-Jun-2023]. 23.

[7] M Wilkinson et al. "The FAIR Guiding Principles for scientific data management and stewardship". In: *Scientific Data* 3.1 (Mar. 2016). DOI: 10.1038/sdata.2016.18. URL: https://doi.org/10.1038/sdata.2016.18.

[8] E. Lopienska. *Member States of CERN.* General Photo. 2021. URL: https://cds.cern.ch/record/2790156.

[9] Gargamelle collaboration, F. J. Hasert et al. "Observation of Neutrino Like Interactions Without Muon Or Electron in the Gargamelle Neutrino Experiment". In: *Phys. Lett. B* 46 (1973), pp. 138–140. DOI: 10.1016/0370-2693(73)90499-1.

[10] UA1 collaboration, G. Arnison et al. "Experimental Observation of Isolated Large Transverse Energy Electrons with Associated Missing Energy at $\sqrt{s} = 540$ GeV". In: *Phys. Lett. B* 122 (1983), pp. 103–116. DOI: 10.1016/0370-2693(83)91177-2.

[11] UA2 collaboration, M. Banner et al. “Observation of Single Isolated Electrons of High Transverse Momentum in Events with Missing Transverse Energy at the CERN anti-p p Collider”. In: *Phys. Lett. B* 122 (1983), pp. 476–485. DOI: 10.1016/0370-2693(83)91605-2.

[12] ATLAS collaboration, Georges Aad et al. “Observation of a new particle in the search for the Standard Model Higgs boson with the ATLAS detector at the LHC.” In: *Phys. Lett. B* 716 (2012). Comments: 24 pages plus author list (38 pages total), 12 figures, 7 tables, revised author list, pp. 1–29. DOI: 10.1016/j.physletb.2012.08.020. arXiv: 1207.7214. URL: https://cds.cern.ch/record/1471031.

[13] CMS collaboration, Serguei Chatrchyan et al. “Observation of a new boson at a mass of 125 GeV with the CMS experiment at the LHC”. In: *Phys. Lett. B* 716 (2012), pp. 30–61. DOI: 10.1016/j.physletb.2012.08.021. arXiv: 1207.7235. URL: https://cds.cern.ch/record/1471016.

[14] E. Lopienska. *The CERN accelerator complex, layout in 2022. Complexe des accélérateurs du CERN en janvier 2022.* General Photo. 2022. URL: https://cds.cern.ch/record/2800984.

[15] CERN Archive. *Gargamelle: first neutral current.* 1973. URL: https://cds.cern.ch/record/39468.

[16] T. Sjostrand, S. Mrenna, and P. Z. Skands. “A Brief Introduction to PYTHIA 8.1”. In: *Comput. Phys. Commun.* 178 (2008), pp. 852–867. DOI: 10.1016/j.cpc.2008.01.036. arXiv: 0710.3820 [hep-ph].

[17] LHCb collaboration, Christian E. *$\bar{b}b$ production angle plots.* https://lhcb.web.cern.ch/lhcb/speakersbureau/html/bb_ProductionAngles.html.

[18] LHCb Collaboration, Jr. Alves et al. “The LHCb Detector at the LHC”. In: *JINST* 3 (2008), S08005. DOI: 10.1088/1748-0221/3/08/S08005.

[19] I. Belyaev et al. “The history of LHCb”. In: *The European Physical Journal H* 46.1 (Mar. 2021). DOI: 10.1140/epjh/s13129-021-00002-z.

[20] LHCb Collaboration, R. Aaij et al. “LHCb Detector Performance”. In: *Int. J. Mod. Phys. A* 30.07 (2015), p. 1530022. DOI: 10.1142/S0217751X15300227. arXiv: 1412.6352 [hep-ex].

[21] LHCb Collaboration, R. Aaij et al. “Measurement of the b-quark production cross-section in 7 and 13 TeV pp collisions”. In: *Phys. Rev. Lett.* 118.5 (2017). [Erratum: Phys.Rev.Lett. 119, 169901 (2017)], p. 052002. DOI: 10.1103/PhysRevLett.118.052002. arXiv: 1612.05140 [hep-ex].

[22] LHCb collaboration, P. Koppenburg. *List of hadrons observed at the LHC.* LHCb-FIGURE-2021-001. See 2022 update online. Mar. 2021. URL: https://cds.cern.ch/record/2693187.

[23] LHCb Collaboration, Roel Aaij et al. "Observation of CP Violation in Charm Decays". In: *Phys. Rev. Lett.* 122.21 (2019), p. 211803. DOI: 10.1103/PhysRevLett.122.211803. arXiv: 1903.08726 [hep-ex].

[24] LHCb Collaboration, R Aaij et al. "Observation of $D^0 - \overline{D}^0$ oscillations". In: *Phys. Rev. Lett.* 110.10 (2013), p. 101802. DOI: 10.1103/PhysRevLett.110.101802. arXiv: 1211.1230 [hep-ex].

[25] LHCb Collaboration, The LHCb collaboration. *LHCb magnet: Technical design report.* Tech. rep. CERN, 2000.

[26] J. Andre et al. "Status of the LHCb magnet system". In: *IEEE Trans. Appl. Supercond.* 12.1 (2002), pp. 366–371. DOI: 10.1109/TASC.2002.1018421.

[27] J. André et al. "Status of the LHCb dipole magnet". In: *IEEE Trans. Appl. Supercond.* 14.2 (2004), pp. 509–513. DOI: 10.1109/TASC.2004.829705.

[28] LHCb Collaboration, R. Aaij et al. "Performance of the LHCb Vertex Locator". In: *JINST* 9 (2014), P09007. DOI: 10.1088/1748-0221/9/09/P09007. arXiv: 1405.7808 [physics.ins-det].

[29] The LHCb Outer Tracker group. "Performance of the LHCb Outer Tracker". In: *JINST* 9.01 (2014), P01002. DOI: 10.1088/1748-0221/9/01/P01002. arXiv: 1311.3893 [physics.ins-det].

[30] The LHCb Muon group, A. A. Alves et al. "Performance of the LHCb muon system". In: *JINST* 8 (2013), P02022. DOI: 10.1088/1748-0221/8/02/P02022. arXiv: 1211.1346 [physics.ins-det].

[31] T. Head. "The LHCb trigger system". In: *JINST* 9 (2014), p. C09015. DOI: 10.1088/1748-0221/9/09/C09015.

[32] LHCb Collaboration, R Aaij et al. "The LHCb Trigger and its Performance in 2011". In: *JINST* 8 (2013), P04022. DOI: 10.1088/1748-0221/8/04/P04022. arXiv: 1211.3055 [hep-ex].

[33] LHCb Collaboration, R. Aaij et al. "The LHCb upgrade I". In: *JINST* (May 2023). arXiv: 2305.10515 [hep-ex].

[34] I. Bird et al. *Update of the Computing Models of the WLCG and the LHC Experiments.* Apr. 2014.

[35] LHCb Collaboration. *Computing Model of the Upgrade LHCb experiment.* Tech. rep. Geneva: CERN, 2018. DOI: 10.17181/CERN.Q0P4.570N. URL: https://cds.cern.ch/record/2319756.

[36] LHCb Collaboration, R. Aaij et al. "Design and performance of the LHCb trigger and full real-time reconstruction in Run 2 of the LHC". In: *JINST* 14.04 (2019), P04013. DOI: 10.1088/1748-0221/14/04/P04013. arXiv: 1812.10790 [hep-ex].

[37] R. Aaij et al. "Tesla : an application for real-time data analysis in High Energy Physics". In: *Comput. Phys. Commun.* 208 (2016), pp. 35–42. DOI: 10.1016/j.cpc.2016.07.022. arXiv: 1604.05596 [physics.ins-det].

[38] R. Aaij et al. "A comprehensive real-time analysis model at the LHCb experiment". In: *JINST* 14.04 (2019), P04006. DOI: 10.1088/1748-0221/14/04/P04006. arXiv: 1903.01360 [hep-ex].

[39] R. Aaij et al. "Allen: A high level trigger on GPUs for LHCb". In: *Comput. Softw. Big Sci.* 4.1 (2020), p. 7. DOI: 10.1007/s41781-020-00039-7. arXiv: 1912.09161 [physics.ins-det].

[40] N. Skidmore, E. Rodrigues, and P. Koppenburg. "Run-3 offline data processing and analysis at LHCb". In: *PoS* EPS-HEP2021 (2022), p. 792. DOI: 10.22323/1.398.0792.

[41] ISO Central Secretary. *Programming languages - C++*. en. Standard ISO/IEC 14882:2020. Geneva, CH: International Organization for Standardization, 2020. URL: https://www.iso.org/standard/79358.html.

[42] M. Cattaneo et al. "Status of the GAUDI event-processing framework". In: *12th International Conference on Computing in High-Energy and Nuclear Physics*. 2001.

[43] G. Van Rossum and F. L. Drake. *Python 3 Reference Manual*. Scotts Valley, CA: CreateSpace, 2009. ISBN: 1441412697.

[44] V. J. Cervantes et al. "Building, testing and distributing common software for the LHC experiments". In: *EPJ Web Conf.* 214 (2019), p. 05020. DOI: 10.1051/epjconf/201921405020.

[45] Rene Brun and Fons Rademakers. "ROOT - An object oriented data analysis framework". In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 389.1 (1997). New Computing Techniques in Physics Research V, pp. 81–86. ISSN: 0168-9002. DOI: https://doi.org/10.1016/S0168-9002(97)00048-X. URL: https://www.sciencedirect.com/science/article/pii/S016890029700048X.

[46] M. Frank et al. "DD4hep: A Detector Description Toolkit for High Energy Physics Experiments". In: *J. Phys. Conf. Ser.* 513 (2014), p. 022010. DOI: 10.1088/1742-6596/513/2/022010.

[47] GEANT4 collaboration, S. Agostinelli et al. "GEANT4–a simulation toolkit". In: *Nucl. Instrum. Meth. A* 506 (2003), pp. 250–303. DOI: 10.1016/S0168-9002(03)01368-8.

[48] AlDanial. *GitHub - AlDanial/cloc: cloc counts blank lines, comment lines, and physical lines of source code in many programming languages. — github.com*. https://github.com/AlDanial/cloc. [Accessed 07-Jul-2023]. 2023.

[49] Linux development community. *Git — git-scm.com*. https://git-scm.com/. [Accessed 26-May-2023]. 2005.

[50] GitLab. *GitLab — DevSecOps Platform*. https://about.gitlab.com/. [Accessed 26-May-2023]. 2014.

[51] Jenkins. *Jenkins - open source automation server*. https://www.jenkins.io/. [Accessed 10-Jul-2023]. 2023.

[52] D. H. Cámpora Pérez, N. Neufeld, and A. Riscos Núñez. "Search by triplet: An efficient local track reconstruction algorithm for parallel architectures". In: *J. Comput. Sci.* 54 (2021), p. 101422. DOI: 10.1016/j.jocs.2021.101422. arXiv: 2207.03936 [hep-ex].

[53] Richard O. Duda and Peter E. Hart. "Use of the Hough Transformation to Detect Lines and Curves in Pictures". In: *Commun. ACM* 15.1 (Jan. 1972), pp. 11–15. DOI: 10.1145/361237.361242. URL: https://doi.org/10.1145/361237.361242.

[54] R. E. Kalman. "A New Approach to Linear Filtering and Prediction Problems". In: *Journal of Basic Engineering* 82.1 (Mar. 1960), pp. 35–45. DOI: 10.1115/1.3662552. URL: https://doi.org/10.1115/1.3662552.

[55] C. Burr. "Searching for rare charm decays, performing alignment studies and improving the analysis ecosystem in HEP". Presented 31 Dec 2019. PhD thesis. Manchester University, 2019. URL: http://cds.cern.ch/record/2759988.

[56] R. Aaij et al. "Selection and processing of calibration samples to measure the particle identification performance of the LHCb experiment in Run 2". In: *EPJ Tech. Instrum.* 6.1 (2019), p. 1. DOI: 10.1140/epjti/s40485-019-0050-z. arXiv: 1803.00824 [hep-ex].

[57] N. Nolte. "A Selection Framework for LHCb's Upgrade Trigger". Presented 22 Feb 2021. PhD thesis. TU Dortmund, 2020. URL: https://cds.cern.ch/record/2765896.

[58] A. Ryd et al. *EvtGen: A Monte Carlo Generator for B-Physics*. May 2005.

[59] AIDA 2020. *Advanced European Infrastructures for Detectors at Accelerators*. https://aida2020.web.cern.ch/aida2020/. [Accessed 25-07-2023]. 2020.

[60] S. Borghi et al. "Perspectives for the migration of the LHCb geometry to the DD4hep toolkit". In: *EPJ Web Conf.* 214 (2019), p. 02022. DOI: {10.1051/epjconf/201921402022}.

[61] LHCb Collaboration, M. Clemencic. "A Git-based Conditions Database backend for LHCb". In: *EPJ Web Conf.* 214 (2019), p. 04037. DOI: 10.1051/epjconf/201921404037. URL: https://cds.cern.ch/record/2701404.

[62] D. H. C. Pérez and B. Couturier. "SIMD studies in the LHCb reconstruction software". In: *J. Phys. Conf. Ser.* 664.9 (2015), p. 092004. DOI: 10.1088/1742-6596/664/9/092004.

[63] A. Hennequin et al. "A fast and efficient SIMD track reconstruction algorithm for the LHCb Upgrade 1 VELO-PIX detector". In: *JINST* 15.06 (2020), P06018. DOI: 10.1088/1748-0221/15/06/P06018. arXiv: 1912.09901 [physics.ins-det].

[64] LHCb Collaboration, J. Albrecht et al. "New approaches for track reconstruction in LHCb's Vertex Locator". In: *EPJ Web Conf.* 214 (2019), p. 01042. DOI: 10.1051/epjconf/201921401042.

[65] LHCb Collaboration, F. Lemaitre, B. Couturier, and L. Lacassagne. “Cholesky factorization on SIMD multi-core architectures”. In: *J. Syst. Architecture* 79 (2017), pp. 1–15. DOI: 10.1016/j.sysarc.2017.06.005.

[66] LHCb Collaboration, L. Promberger et al. “Porting the LHCb Stack from x86 (Intel) to aarch64 (ARM) and ppc64le (PowerPC)”. In: *EPJ Web Conf.* 214 (2019), p. 05016. DOI: 10.1051/epjconf/201921405016.

[67] S. V. Kartik et al. “Measurements of the LHCb software stack on the ARM architecture”. In: *J. Phys. Conf. Ser.* 513 (2014), p. 052014. DOI: 10.1088/1742-6596/513/5/052014.

[68] LHCb Online. *Online Raw Data Format.* https://edms.cern.ch/ui/file/784588/2/Online_Raw_Data_Format.pdf. [Accessed 01-June-2023]. 2005.

[69] LHCb Collaboration. *LHCb Stripping Project.* http://lhcbdoc.web.cern.ch/lhcbdoc/stripping/. [Accessed 03-August-2023]. 2023.

[70] A. Tsaregorodtsev et al. “DIRAC: A community grid solution”. In: *J. Phys. Conf. Ser.* 119 (2008), p. 062048. DOI: 10.1088/1742-6596/119/6/062048.

[71] K. Harrison et al. “GANGA: a user-Grid interface for Atlas and LHCb. A TOROIDAL LHC APPARATUS”. In: *Unknown* (2003). URL: https://cds.cern.ch/record/622197.

[72] M. Baker. “1, 500 scientists lift the lid on reproducibility”. In: *Nature* 533.7604 (May 2016), pp. 452–454. DOI: 10.1038/533452a. URL: https://doi.org/10.1038/533452a.

[73] John P. A. Ioannidis. “Why Most Published Research Findings Are False”. In: *PLoS Medicine* 2.8 (Aug. 2005), e124. DOI: 10.1371/journal.pmed.0020124. URL: https://doi.org/10.1371/journal.pmed.0020124.

[74] R. D. Peng. “Reproducible research in computational science”. In: *Science* 334.6060 (Dec. 2011), pp. 1226–1227.

[75] Joshua Wyatt Smith et al. “ATLAS software stack on ARM64”. In: *Journal of Physics: Conference Series* 898.7 (Oct. 2017), p. 072001. DOI: 10.1088/1742-6596/898/7/072001. URL: https://dx.doi.org/10.1088/1742-6596/898/7/072001.

[76] LHCb Collaboration, Laura Promberger et al. “Porting the LHCb Stack from x86 (Intel) to aarch64 (ARM) and ppc64le (PowerPC)”. In: *EPJ Web Conf.* 214 (2019), p. 05016. DOI: 10.1051/epjconf/201921405016. URL: https://cds.cern.ch/record/2700240.

[77] David Goldberg. “What Every Computer Scientist Should Know About Floating-Point Arithmetic.” In: *ACM Comput. Surv.* 23.1 (1991). corrigendum: ACM Computing Surveys 23(3): 413 (1991), comments: ACM Computing Surveys 24(2): 319 (1992), pp. 5–48. URL: http://dblp.uni-trier.de/db/journals/csur/csur23.html#Goldberg91.

[78] Nature. *Reporting standards and availability of data, materials, code and protocols — Nature Portfolio — nature.com.* https://www.nature.com/nature-portfolio/editorial-policies/reporting-standards. [Accessed 22-May-2023].

[79] Directorate-General for Research European Commission and Innovation. *H2020 Programme: Guidelines on FAIR Data Management in Horizon 2020 Version 3.0.* 2016. DOI: 10.25607/OBP-774. URL: https://www.oceanbestpractices.net/handle/11329/1259.

[80] European Commission, Directorate-General for Research, and Innovation. *Cost-benefit analysis for FAIR research data : cost of not having FAIR research data.* Publications Office, 2019. DOI: doi/10.2777/02999.

[81] EOSC. *EOSC Portal — eosc-portal.eu.* https://eosc-portal.eu/. [Accessed 30-May-2023]. 2020.

[82] Genova, F. et al. "The CDS information hub - On-line services and links at the Centre de Données astronomiques de Strasbourg". In: *Astron. Astrophys. Suppl. Ser.* 143.1 (2000), pp. 1–7. DOI: 10.1051/aas:2000333. URL: https://doi.org/10.1051/aas:2000333.

[83] NASA. *NASA Space Physics Data Facility (SPDF) — spdf.gsfc.nasa.gov.* https://spdf.gsfc.nasa.gov/. [Accessed 05-Jun-2023]. 2023.

[84] Brian Lavoie. *The Open Archival Information System (OAIS) Reference Model: Introductory Guide (2nd Edition).* Tech. rep. Digital Preservation Coalition, Oct. 2014. DOI: 10.7207/twr14-02. URL: https://doi.org/10.7207/twr14-02.

[85] ISO Central Secretary. *Space data and information transfer systems — Open archival information system (OAIS) — Reference model.* en. Standard ISO 14721:2012. Geneva, CH: International Organization for Standardization, 2012. URL: https://www.iso.org/standard/57284.html.

[86] Andrii Neronov. "Introduction to multi-messenger astronomy". In: *Journal of Physics: Conference Series* 1263.1 (June 2019), p. 012001. DOI: 10.1088/1742-6596/1263/1/012001. URL: https://dx.doi.org/10.1088/1742-6596/1263/1/012001.

[87] DPHEP Study Group. *Data Preservation in High Energy Physics.* 2009. arXiv: 0912.0255 [hep-ex].

[88] LHCb Collaboration, Ben Couturier. "LHCb in the International Particle Physics Masterclasses. LHCb outreach activities". In: *PoS* FFP14 (2016), p. 228. DOI: 10.22323/1.224.0228. URL: https://cds.cern.ch/record/2264485.

[89] DPHEP Study Group, Zaven Akopov et al. "Status Report of the DPHEP Study Group: Towards a Global Effort for Sustainable Data Preservation in High Energy Physics". In: *arXiv* (May 2012). arXiv: 1205.4667 [hep-ex].

[90] DPHEP Study Group, T. Basaglia et al. *Data Preservation in High Energy Physics – DPHEP Global Report 2022.* Feb. 2023. arXiv: 2302.03583 [hep-ex].

[91] A. Trisovic. "Data preservation and reproducibility at the LHCb experiment at CERN". Presented 16 Aug 2018. PhD thesis. Cambridge University, 2018. URL: http://cds.cern.ch/record/2640461.

[92] Kyle Cranmer et al. *Analysis Preservation in ATLAS*. Tech. rep. 3. Geneva: CERN, 2015. DOI: 10.1088/1742-6596/664/3/032013. URL: https://cds.cern.ch/record/2016930.

[93] CMS collaboration, Lara Lloret Iglesias. "The CMS approach to Analysis Preservation". In: *EPJ Web Conf.* 245 (2020), p. 06020. DOI: 10.1051/epjconf/202024506020. URL: https://cds.cern.ch/record/2752304.

[94] Sebastian Neubert et al. *LHCb Analysis Preservation Roadmap*. Tech. rep. Geneva: CERN, 2017. URL: https://cds.cern.ch/record/2280615.

[95] LHCbStarterkit Team, Albert Puig. "The LHCb Starterkit". In: *J. Phys.: Conf. Ser.* 898.8 (2017), p. 082054. DOI: 10.1088/1742-6596/898/8/082054. URL: https://cds.cern.ch/record/2296662.

[96] Dario Berzano et al. "Software training for the next generation of physicists: joint experience of LHCb and ALICE". In: *EPJ Web Conf.* 214 (2019), p. 05044. DOI: 10.1051/epjconf/201921405044. URL: https://cds.cern.ch/record/2699855.

[97] V Stodden et al. "Enhancing reproducibility for computational methods". In: *Science* 354.6317 (Dec. 2016), pp. 1240–1241. DOI: 10.1126/science.aah6168. URL: https://doi.org/10.1126/science.aah6168.

[98] LHCb Collaboration. *LHCb Run Database*. https://lbrundb.cern.ch/. [Accessed 29-Aug-2023]. 2008.

[99] M. Adinolfi et al. "LHCb data quality monitoring". In: *J. Phys. Conf. Ser.* 898.9 (2017), p. 092027. DOI: 10.1088/1742-6596/898/9/092027.

[100] C. Bozzi. *LHCb Computing Resource usage in 2022*. Tech. rep. Geneva: CERN, 2023. URL: https://cds.cern.ch/record/2850602.

[101] E. Cano et al. "CERN Tape Archive: a distributed, reliable and scalable scheduling system". In: *EPJ Web Conf.* 251 (2021), p. 02037. DOI: 10.1051/epjconf/202125102037. URL: https://doi.org/10.1051/epjconf/202125102037.

[102] Suayb S. et al. *LTO-9 technology and user data reliability analysis*. https://www.lto.org/wp-content/uploads/2022/08/LTO-UBER-Technical-Paper-August-2022.pdf. [Accessed 31-May-2023]. 2022.

[103] Luca dell'Agnello. "Disaster recovery of the INFN Tier–1 data center: lesson learned". In: *EPJ Web of Conferences* 214 (2019), p. 09008. DOI: 10.1051/epjconf/201921409008. URL: https://doi.org/10.1051/epjconf/201921409008.

[104] Linear Tape open Program. *Linear Tape Open*. https://www.lto.org/. [Accessed 05-June-2023]. 2023.

[105] CTA Team. *ATRESYS — Automated Tape REpacking System, a tool for managing CTA repacks and tape lifecycle*. https://indico.cern.ch/event/1227241/contributions/5366313/. [Accessed 31-May-2023]. 2023.

[106] ROOT. *ROOT Versioning and compatibility.* https://root.cern/about/versioning/. [Accessed 01-June-2023]. 2023.

[107] LHCb Collaboration. *LHCb TDR computing technical design report.* June 2005.

[108] M. Cattaneo. *MicroDST.* https://twiki.cern.ch/twiki/bin/view/LHCb/MicroDST. [Accessed 01-08-2023]. 2013.

[109] Cervantes Villanueva, J. et al. “Building, testing and distributing common software for the LHC experiments”. In: *EPJ Web Conf.* 214 (2019), p. 05020. DOI: 10.1051/epjconf/201921405020. URL: https://doi.org/10.1051/epjconf/201921405020.

[110] B. Hegner. *HSF Platform Naming Conventions.* https://hepsoftwarefoundation.org/notes/HSF-TN-2018-01.pdf. [Accessed 31-May-2023]. 2018.

[111] F. Weimer. *Building Red Hat Enterprise Linux 9 for the x86-64-v2 microarchitecture level.* https://developers.redhat.com/blog/2021/01/05/building-red-hat-enterprise-linux-9-for-the-x86-64-v2-microarchitecture-level. [Accessed 31-May-2023]. 2021.

[112] Matev, R., Nolte, N., and Pearce, A. “Configuration and scheduling of the LHCb trigger application”. In: *EPJ Web Conf.* 245 (2020), p. 05004. DOI: 10.1051/epjconf/202024505004. URL: https://doi.org/10.1051/epjconf/202024505004.

[113] LHCb Trigger. *The LHCb Trigger configuration key.* https://twiki.cern.ch/twiki/bin/view/LHCb/TCK. [Accessed 31-May-2023]. 2021.

[114] LHCb Simulation. *The Gauss project.* http://lhcbdoc.web.cern.ch/lhcbdoc/gauss/. [Accessed 31-May-2023]. 2021.

[115] D. P. Bovet and M. Cesati. *Understanding the Linux Kernel; 3rd ed.* Sebastopol, CA: O’Reilly, 2006. URL: https://cds.cern.ch/record/920385.

[116] Wikipedia. *Red Hat Enterprise Linux - Wikipedia — en.wikipedia.org.* https://en.wikipedia.org/wiki/Red_Hat_Enterprise_Linux. [Accessed 27-May-2023]. 2023.

[117] A. Randal. “The Ideal Versus the Real: Revisiting the History of Virtual Machines and Containers”. In: *ACM Comput. Surv.* (Feb. 2020). DOI: 10.1145/3365199. URL: https://doi.org/10.1145/3365199.

[118] RedHat. *What’s a Linux container? — redhat.com.* https://www.redhat.com/en/topics/containers/whats-a-linux-container. [Accessed 27-May-2023]. 2022.

[119] D. Merkel. “Docker: lightweight linux containers for consistent development and deployment”. In: *Linux journal* 2014.239 (2014), p. 2.

[120] Matt et al Heon. *Podman - : A tool for managing OCI containers and pods.* https://doi.org/10.5281/zenodo.4735634. Version v1.0 and beyond. Currently at v3.0.1. Jan. 2018. DOI: 10.5281/zenodo.4735634.

[121] G. Kurtzer et al. *hpcng/singularity: Singularity 3.7.3.* https://doi.org/10.5281/zenodo.4667718. Version v3.7.3. Apr. 2021. DOI: 10.5281/zenodo.4667718.

[122] Apptainer. *Apptainer*. https://apptainer.org/. [Accessed 01-Oct-2023]. 2023.

[123] GitLab CERN. *CERN GitLab instance*. https://gitlab.cern.ch. [Accessed 01-June-2023]. 2023.

[124] J. Blomer et al. "Distributing LHC application software and conditions databases using the CernVM file system". In: *J. Phys. Conf. Ser.* 331 (2011), p. 042003. DOI: 10.1088/1742-6596/331/4/042003.

[125] A. Boyer et al. "A Subset of the CERN Virtual Machine File System: Fast Delivering of Complex Software Stacks for Supercomputing Resources". In: *Lecture Notes in Computer Science*. Springer International Publishing, 2022, pp. 354–371. DOI: 10.1007/978-3-031-07312-0_18. URL: https://doi.org/10.1007%2F978-3-031-07312-0_18.

[126] Zoltan Mathe. "Feicim: A browser and analysis tool for distributed data in particle physics". Presented 01 Jun 2012. PhD thesis. University College Dublin, 2012. URL: http://cds.cern.ch/record/1491175.

[127] LHCb Collaboration. *LHCb Processing Passes*. https://twiki.cern.ch/twiki/bin/view/Main/ProcessingPasses. [Accessed 01-June-2023]. 2023.

[128] B. Couturier, E. Kiagias, and Stefan B. Lohn. "Systematic profiling to monitor and specify the software refactoring process of the LHCb experiment". In: *J. Phys.: Conf. Ser.* 513 (2014), p. 052020. DOI: 10.1088/1742-6596/513/5/052020. URL: https://cds.cern.ch/record/2055722.

[129] LHCb Collaboration, M. Szymański and B. Couturier. "Improvements to the LHCb software performance testing infrastructure using message queues and big data technologies". In: *EPJ Web Conf.* 214 (2019), p. 05014. DOI: 10.1051/epjconf/201921405014. URL: https://cds.cern.ch/record/2700239.

[130] A. Mazurov et al. "Microservices for systematic profiling and monitoring of the refactoring process at the LHCb experiment". In: *J. Phys.: Conf. Ser.* 898.7 (2017), p. 072037. DOI: 10.1088/1742-6596/898/7/072037. URL: https://cds.cern.ch/record/2296799.

[131] Dmitry Popov. "LHCb - Quality assurance of the LHCb Simulation". Poster. 2022. URL: https://cds.cern.ch/record/2841050.

[132] LHCb Collaboration, R. Currie and C. Fitzpatrick. "Monitoring LHCb Trigger developments using nightly integration tests and a new interactive web UI". In: *EPJ Web Conf.* 214 (2019), p. 05042. DOI: 10.1051/epjconf/201921405042. URL: https://cds.cern.ch/record/2728394.

[133] D. Piparo et al. "RDataFrame: Easy parallel ROOT analysis at 100 threads". In: *EPJ Web Conf.* 214 (2019), p. 06029. DOI: 10.1051/epjconf/201921406029. URL: http://cds.cern.ch/record/2699587.

[134] Letcher B et al. Mölder F Jablonski KP. "Sustainable data analysis with Snakemake [version 2; peer review: 2 approved]". In: *F1000Research* (2021). DOI: https://doi.org/10.12688/f1000research.29032.2.

[135] Carole Goble et al. "FAIR Computational Workflows". In: *Data Intelligence* 2.1-2 (Jan. 2020), pp. 108–121. DOI: 10.1162/dint_a_00033. URL: https://doi.org/10.1162/dint_a_00033.

[136] OMG. *Business Process Model and Notation (BPMN), Version 2.0.* Object Management Group, 2011. URL: http://www.omg.org/spec/BPMN/2.0.

[137] LHCb Collaboration. *LHCb Publication database.* https://lhcbproject.web.cern.ch/Publications/LHCbProjectPublic/Summary_all.html. [Accessed 01-June-2023]. 2023.

[138] Glance team. *Glance Project.* https://readthedocs.web.cern.ch/display/FPS/Homepage. [Accessed 01-June-2023]. 2023.

[139] Carlos Brito. "CHEP 2023 - The migration to a standardized architecture for developing systems on the Glance project". [Accessed 01-June-2023]. 2023. URL: http://cds.cern.ch/record/2860038.

[140] Chris Burr. "Analysis Productions: A declarative approach to ntupling". In: *CHEP 2023* (2023). URL: https://cds.cern.ch/record/2860345.

[141] A. Trisovic et al. "Provenance Tracking in the LHCb Software". In: *Comput. Sci. Eng.* 22.2 (2020), pp. 88–94. DOI: 10.1109/MCSE.2020.2970625. arXiv: 1910.02863 [cs.DC].

[142] E Lanciotti and Z Mathe. "LHCb: The LHCb data bookkeeping system ". Unpublished. 2009. URL: http://cds.cern.ch/record/1170456.

[143] LHCb Collaboration, R. O'Neil. "The NTuple Wizard An NTuple production service for accessing LHCb Open Data". Ntuple wizard poster. 2014. URL: https://cds.cern.ch/record/2815814.

[144] C. A. Aidala et al. "Ntuple Wizard: An Application to Access Large-Scale Open Data from LHCb". In: *Comput. Softw. Big Sci.* 7.1 (2023), p. 6. DOI: 10.1007/s41781-023-00099-5. arXiv: 2302.14235 [hep-ex].

[145] Andy Buckley and Mike Whalley. *HepData reloaded: reinventing the HEP data archive.* 2010. arXiv: 1006.0517 [hep-ex].

[146] Eamonn Maguire, Lukas Heinrich, and Graeme Watt. "HEPData: a repository for high energy physics data". In: *J. Phys. Conf. Ser.* 898.10 (2017), p. 102006. DOI: 10.1088/1742-6596/898/10/102006. arXiv: 1704.05473 [hep-ex].

[147] C. Bierlich et al. "Robust Independent Validation of Experiment and Theory: Rivet version 3". In: *SciPost Phys.* 8 (2020), p. 026. DOI: 10.21468/SciPostPhys.8.2.026. arXiv: 1912.05451 [hep-ph].

[148] Pamfilos Fokianos et al. "CERN Analysis Preservation and Reuse Framework: FAIR research data services for LHC experiments". In: *EPJ Web Conf.* 245 (2020), p. 06011. DOI: 10.1051/epjconf/202024506011. URL: https://cds.cern.ch/record/2752852.

[149] J Cowton et al. "Open Data and Data Analysis Preservation Services for LHC Experiments". In: *J. Phys.: Conf. Ser.* 664.3 (2015), p. 032030. DOI: 10.1088/1742-6596/664/3/032030. URL: https://cds.cern.ch/record/2134548.

[150] Invenio. *Invenio*. https://inveniosoftware.org/. [Accessed 29-Aug-2023]. 2023.

[151] G. Bitzes et al. “EOS architectural evolution and strategic development directions”. In: *EPJ Web of Conferences* 245 (2020), p. 04009. DOI: 10.1051/epjconf/202024504009. URL: https://doi.org/10.1051/epjconf/202024504009.

[152] M. Clemencic and B. Couturier. “LHCb Build and Deployment Infrastructure for run 2”. In: *J. Phys. Conf. Ser.* 664.6 (2015), p. 062008. DOI: 10.1088/1742-6596/664/6/062008.

[153] LHCb Collaboration, S. Chitic et al. “LHCb continuous integration and deployment system: a message based approach”. In: *EPJ Web Conf.* 214 (2019), p. 05001. DOI: 10.1051/epjconf/201921405001.

[154] S. Binet and B. Couturier. “docker & HEP: Containerization of applications for development, distribution and preservation”. In: *J. Phys. Conf. Ser.* 664.2 (2015), p. 022007. DOI: 10.1088/1742-6596/664/2/022007.

[155] M. Clemencic and B. Couturier. “Implementing a Domain Specific Language to configure and run LHCb Continuous Integration builds”. In: *J. Phys. Conf. Ser.* 664.6 (2015), p. 062007. DOI: 10.1088/1742-6596/664/6/062007.

[156] LHCb Collaboration, C. Burr, M. Clemencic, and B. Couturier. “Software packaging and distribution for LHCb using Nix”. In: *EPJ Web Conf.* 214 (2019), p. 05005. DOI: 10.1051/epjconf/201921405005.

[157] Alexander Mazurov et al. “Microservices for systematic profiling and monitoring of the refactoring process at the LHCb experiment”. In: *J. Phys. Conf. Ser.* 898.7 (2017), p. 072037. DOI: 10.1088/1742-6596/898/7/072037.

[158] R. T. Fielding. “REST: Architectural Styles and the Design of Network-based Software Architectures”. Doctoral dissertation. 2000. URL: http://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm.

[159] conda-forge community. *The conda-forge Project: Community-based Software Distribution Built on the conda Package Format and Ecosystem.* July 2015. DOI: 10.5281/zenodo.4774216. URL: https://doi.org/10.5281/zenodo.4774216.

[160] XRootD Collaboration. *Home Page — XRootD — xrootd.slac.stanford.edu.* https://xrootd.slac.stanford.edu/. [Accessed 26-May-2023]. 2023.

[161] AJ Peters, EA Sindrilaru, and G Adde. “EOS as the present and future solution for data storage at CERN”. In: *Journal of Physics: Conference Series* 664.4 (Dec. 2015), p. 042042. URL: https://dx.doi.org/10.1088/1742-6596/664/4/042042.

[162] AT&T Research. *Graphviz - Graph Visualization Software.* 2008. URL: http://www.graphviz.org/.

[163] Snakemake. *Command line interface; Snakemake 7.26.0 documentation.* https://snakemake.readthedocs.io/en/stable/executing/cli.html. [Accessed 26-May-2023]. 2023.

[164] The CWL Community, M.R Crusoe et al. “Methods included: standardizing computational reuse and portability with the Common Workflow Language”. In: *Communications of the ACM* 65.6 (May 2022), pp. 54–63. DOI: 10.1145/3486897. URL: https://doi.org/10.1145/3486897.

[165] T.r Simko et al. “REANA: A system for reusable research data analyses”. In: *EPJ Web Conf.* 214 (2019), p. 06034. DOI: 10.1051/epjconf/201921406034. URL: https://cds.cern.ch/record/2652340.

[166] Martin Fowler. *Continuous Integration — martinfowler.com*. https://martinfowler.com/articles/continuousIntegration.html. [Accessed 26-May-2023]. 2006.

[167] A. Ahmad and others. “The new (and improved!) CERN Single-Sign-On”. In: *EPJ Web Conf.* 251 (2021), p. 02015. DOI: 10.1051/epjconf/202125102015. URL: https://cds.cern.ch/record/2782836.

[168] D. Dick Hardt. *The OAuth 2.0 Authorization Framework*. RFC 6749. Oct. 2012. DOI: 10.17487/RFC6749. URL: https://www.rfc-editor.org/info/rfc6749.

[169] Michael B. Jones, John Bradley, and Nat Sakimura. *JSON Web Token (JWT)*. RFC 7519. May 2015. DOI: 10.17487/RFC7519. URL: https://www.rfc-editor.org/info/rfc7519.

[170] LHCb collaboration. *lhcb-analysis-preservation containerization-cookie*. https://gitlab.cern.ch/lhcb-analysis-preservation/containerization-cookie. [Accessed 31-May-2023]. 2021.

[171] E. Rodrigues et al. “The Scikit HEP Project – overview and prospects”. In: *EPJ Web Conf.* 245 (2020), p. 06028. DOI: 10.1051/epjconf/202024506028. arXiv: 2007.03577 [physics.comp-ph].

[172] Anaconda. *Anaconda Software Distribution*. Version Vers. 2-2.4.0. 2020. URL: https://docs.anaconda.com/.

[173] P. Buncic et al. “CernVM – a virtual software appliance for LHC applications”. In: *CernVM*. 2010.

[174] C. Burr et al. *Sustainable software packaging for end users with conda*. https://doi.org/10.5281/zenodo.3599549. Jan. 2020. DOI: 10.5281/zenodo.3599549. URL: https://doi.org/10.5281/zenodo.3599549.

[175] James Clerk Maxwell. “VIII. A dynamical theory of the electromagnetic field”. In: *Philosophical Transactions of the Royal Society of London* 155 (Dec. 1865), pp. 459–512. DOI: 10.1098/rstl.1865.0008. URL: https://doi.org/10.1098/rstl.1865.0008.

[176] S. L. Glashow. “Partial Symmetries of Weak Interactions”. In: *Nucl. Phys.* 22 (1961), pp. 579–588. DOI: 10.1016/0029-5582(61)90469-2.

[177] Steven Weinberg. “A Model of Leptons”. In: *Phys. Rev. Lett.* 19 (1967), pp. 1264–1266. DOI: 10.1103/PhysRevLett.19.1264.

[178] Abdus Salam. "Weak and Electromagnetic Interactions". In: *Conf. Proc. C* 680519 (1968), pp. 367–377. DOI: 10.1142/9789812795915_0034.

[179] A. Zee. *Quantum field theory in a nutshell.* Princeton University Press, 2003. ISBN: 978-0-691-14034-6.

[180] Makoto Kobayashi and Toshihide Maskawa. "Violation in the Renormalizable Theory of Weak Interaction". In: *Progress of Theoretical Physics* 49.2 (Feb. 1973), pp. 652–657. DOI: 10.1143/ptp.49.652. URL: https://doi.org/10.1143/ptp.49.652.

[181] Particle Data Group, R. L. Workman et al. "Review of Particle Physics". In: *PTEP* 2022 (2022), p. 083C01. DOI: 10.1093/ptep/ptac097.

[182] Super-Kamiokande collaboration, Y. Fukuda et al. "Evidence for oscillation of atmospheric neutrinos". In: *Phys. Rev. Lett.* 81 (1998), pp. 1562–1567. DOI: 10.1103/PhysRevLett.81.1562. arXiv: hep-ex/9807003.

[183] SNO collaboration, Q. R. Ahmad et al. "Measurement of the rate of $\nu_e + d \to p + p + e^-$ interactions produced by ^{8}B solar neutrinos at the Sudbury Neutrino Observatory". In: *Phys. Rev. Lett.* 87 (2001), p. 071301. DOI: 10.1103/PhysRevLett.87.071301. arXiv: nucl-ex/0106015.

[184] SNO collaboration, Q. R. Ahmad et al. "Direct evidence for neutrino flavor transformation from neutral current interactions in the Sudbury Neutrino Observatory". In: *Phys. Rev. Lett.* 89 (2002), p. 011301. DOI: 10.1103/PhysRevLett.89.011301. arXiv: nucl-ex/0204008.

[185] Stephen F. King. "Discrete Symmetries and Models of Flavour Mixing". In: *J. Phys. Conf. Ser.* 631.1 (2015), p. 012005. DOI: 10.1088/1742-6596/631/1/012005.

[186] ALEPH, DELPHI, L3, OPAL, SLD, LEP Electroweak Working Group, SLD Electroweak Group, SLD Heavy Flavour Group, S. Schael et al. "Precision electroweak measurements on the Z resonance". In: *Phys. Rept.* 427 (2006), pp. 257–454. DOI: 10.1016/j.physrep.2005.12.006. arXiv: hep-ex/0509008.

[187] CDF collaboration, A. Abulencia et al. "Measurement of $\sigma(p\bar{p} \to Z).\mathrm{Br}(Z \to 2\tau)$ in $p\bar{p}$ collisions at $\sqrt{s} = 1.96$ TeV". In: *Phys. Rev. D* 75 (2007), p. 092004. DOI: 10.1103/PhysRevD.75.092004.

[188] The CDF Collaboration. "Measurements of Inclusive W and Z Cross Sections in $p\bar{p}$ Collisions at $\sqrt{s}$ =1.96 TeV". In: *Journal of Physics G: Nuclear and Particle Physics* 34.12 (2005), p. 2457. DOI: 10.48550/ARXIV.HEP-EX/0508029. URL: https://arxiv.org/abs/hep-ex/0508029.

[189] ALEPH et al. "Electroweak measurements in electron-positron collisions at W-boson-pair energies at LEP". In: *Physics Reports* 532.4 (Nov. 2013), pp. 119–244. DOI: 10.1016/j.physrep.2013.07.004. URL: https://doi.org/10.1016/j.physrep.2013.07.004.

[190] The LHCb Collaboration. “Measurement of forward $W \to e\nu$ production in pp collisions at $\sqrt{s} = 8$ TeV”. In: *Journal of High Energy Physics* 30.10 (Oct. 2016). DOI: 10.1007/jhep10(2016)030. URL: https://doi.org/10.1007/jhep10(2016)030.

[191] The ATLAS collaboration. “Precision measurement and interpretation of inclusive W^+, W^- and Z/γ^* production cross sections with the ATLAS detector”. In: *The European Physical Journal C* 77.6 (June 2017). DOI: 10.1140/epjc/s10052-017-4911-9. URL: https://doi.org/10.1140/epjc/s10052-017-4911-9.

[192] The NA62 collaboration. *Precision measurement of the ratio of the charged kaon leptonic decay rates.* 2013. DOI: https://doi.org/10.1016/j.physletb.2013.01.037. arXiv: 1212.4012 [hep-ex]. URL: https://www.sciencedirect.com/science/article/pii/S0370269313000786.

[193] V. Cirigliano and I. Rosell. “Two-loop effective theory analysis of pi (K) —> e anti-nu/e [gamma] branching ratios”. In: *Phys. Rev. Lett.* 99 (2007), p. 231801. DOI: 10.1103/PhysRevLett.99.231801. arXiv: 0707.3439 [hep-ph].

[194] A. Aguilar-Arevalo et al. “Improved Measurement of the $\pi \to e\nu$ Branching Ratio”. In: *Physical Review Letters* 115.7 (Aug. 2015). DOI: 10.1103/physrevlett.115.071801.

[195] V. Cirigliano and I. Rosell. “Two-Loop Effective Theory Analysis of $\pi(K) \to e\bar{\nu}_e[\gamma]$ Branching Ratios”. In: *Phys. Rev. Lett.* 99 (23 Dec. 2007), p. 231801. DOI: 10.1103/PhysRevLett.99.231801. URL: https://link.aps.org/doi/10.1103/PhysRevLett.99.231801.

[196] Heavy Flavor Averaging Group, Y. S. Amhis et al. “Averages of b-hadron, c-hadron, and τ-lepton properties as of 2021”. In: *Phys. Rev. D* 107.5 (2023), p. 052008. DOI: 10.1103/PhysRevD.107.052008. arXiv: 2206.07501 [hep-ex].

[197] Particle Data Group, M. Tanabashi et al. “Review of Particle Physics”. In: *Physical Review D* 98.3 (Aug. 2018). DOI: 10.1103/physrevd.98.030001. URL: https://doi.org/10.1103/physrevd.98.030001.

[198] S. L. Glashow, J. Iliopoulos, and L. Maiani. “Weak Interactions with Lepton-Hadron Symmetry”. In: *Phys. Rev. D* 2 (7 Oct. 1970), pp. 1285–1292. DOI: 10.1103/PhysRevD.2.1285. URL: https://link.aps.org/doi/10.1103/PhysRevD.2.1285.

[199] G. Hiller and F. Kruger. “More model-independent analysis of $b \to s$ processes”. In: *Physical Review D* 69.7 (Apr. 2004). DOI: 10.1103/physrevd.69.074020. URL: https://doi.org/10.1103/physrevd.69.074020.

[200] C. Bobeth, G. Hiller, and G. Piranishvili. “Angular distributions of $\bar{B} \to \bar{K}\ell\ell$ decays”. In: *Journal of High Energy Physics* 2007.12 (Dec. 2007), pp. 040–040. DOI: 10.1088/1126-6708/2007/12/040. URL: https://doi.org/10.1088/1126-6708/2007/12/040.

[201] C. Bouchard et al. “Standard Model predictions for $B \to K\ell\ell$ with form factors from lattice QCD”. In: *Physical Review Letters* 111.16 (Oct. 2013). DOI: 10.1103/physrevlett.111.162002. URL: https://doi.org/10.1103/physrevlett.111.162002.

[202] J. P. Lees et al. "Measurement of Branching Fractions and Rate Asymmetries in the Rare Decays $B \to K^{(*)}\ell^+\ell^-$ ". In: *Physical Review D* 86.3 (Aug. 2012). DOI: 10.1103/physrevd.86.032012. URL: https://doi.org/10.1103/physrevd.86.032012.

[203] J.-T. Wei et al. "Measurement of the Differential Branching Fraction and Forward-Backward Asymmetry for $B \to K^{(*)}\ell\ell$". In: *Phys. Rev. Lett.* 103.17 (17 Oct. 2009), p. 171801. DOI: 10.1103/physrevlett.103.171801. URL: https://doi.org/10.1103/physrevlett.103.171801.

[204] S. Choudhury et al. "Test of lepton flavor universality and search for lepton flavor violation in B $\to$ K$\ell\ell$ decays". In: *Journal of High Energy Physics* 2021.3 (Mar. 2021). DOI: 10.1007/jhep03(2021)105. URL: https://doi.org/10.1007/jhep03(2021)105.

[205] S. Wehle et al. "Test of lepton flavor universality in $B \to K^*\ell^+\ell^-$ decays at Belle". In: *Physical Review Letters* 126.16 (Apr. 2021). DOI: 10.1103/physrevlett.126.161801. URL: https://doi.org/10.1103/physrevlett.126.161801.

[206] R. Aaij et al. "Test of lepton universality using $B^+ \to K^+\ell^+\ell^-$ decay". In: *Physical Review Letters* 113.15 (Oct. 2014). DOI: 10.1103/physrevlett.113.151601. URL: https://doi.org/10.1103/physrevlett.113.151601.

[207] R. Aaij et al. "Test of lepton universality in beauty-quark decays". In: *Nature Physics* 18.3 (Mar. 2022), pp. 277–282. DOI: 10.1038/s41567-021-01478-8. URL: https://doi.org/10.1038/s41567-021-01478-8.

[208] R. Aaij et al. "Search for Lepton-Universality Violation in $B^+ \to K^+\ell^+\ell^-$ Decays". In: *Physical Review Letters* 122.19 (May 2019). DOI: 10.1103/physrevlett.122.191801. URL: https://doi.org/10.1103/physrevlett.122.191801.

[209] LHCb Collaboration, R. Aaij et al. "Test of lepton universality in $b \to s\ell^+\ell^-$ decays". In: *Phys. Rev. Lett.* 131.5 (2023), p. 051803. DOI: 10.1103/PhysRevLett.131.051803. arXiv: 2212.09152 [hep-ex].

[210] R. Aaij et al. "Test of lepton universality with $\Lambda_b^0 \to pK^-\ell^+\ell^-$ decays". In: *Journal of High Energy Physics* 2017.8 (Aug. 2017). DOI: 10.1007/jhep08(2017)055. URL: https://doi.org/10.1007/jhep08(2017)055.

[211] R. Aaij et al. "Test of lepton universality with $\Lambda_b^0 \to pK^-\ell^+\ell^-$ decays". In: *Journal of High Energy Physics* 2020.5 (May 2020). DOI: 10.1007/jhep05(2020)040. URL: https://doi.org/10.1007/jhep05(2020)040.

[212] The BaBar collaboration, J. P. Lees et al. "Evidence for an excess of $\bar{B} \to D^{(*)}\tau^-\bar{\nu}_\tau$ decays". In: *Physical Review Letters* 109.10 (Sept. 2012). DOI: 10.1103/physrevlett.109.101802. URL: https://doi.org/10.1103/physrevlett.109.101802.

[213] The BaBar collaboration, J. P. Lees et al. "Measurement of an Excess of $\bar{B} \to D^{(*)}\tau^-\bar{\nu}_\tau$ Decays and Implications for Charged Higgs Bosons". In: *Physical Review D* 88.7 (Oct. 2013). DOI: 10.1103/physrevd.88.072012. URL: https://doi.org/10.1103/physrevd.88.072012.

[214] The Belle collaboration, M. Huschle et al. "Measurement of the branching ratio of $\bar{B} \to D^{(*)}\tau^-\bar{\nu}_\tau$ relative to $\bar{B} \to D^{(*)}\ell^-\ \bar{\nu}_\ell$ decays with hadronic tagging at Belle". In: *Physical Review D* 92.7 (Oct. 2015). DOI: 10.1103/physrevd.92.072014. URL: https://doi.org/10.1103/physrevd.92.072014.

[215] The Belle collaboration, Y. Sato et al. "Measurement of the branching ratio of $\bar{B}^0 \to D^{*+}\tau^-\bar{\nu}_\tau$ relative to $\bar{B}^0 \to D^{**}\ell^-\bar{\nu}_\ell$decays with a semileptonic tagging method". In: *Physical Review D* 94.7 (Oct. 2016). DOI: 10.1103/physrevd.94.072007. URL: https://doi.org/10.1103/physrevd.94.072007.

[216] The Belle collaboration, S. Hirose et al. "Measurement of the τ lepton polarization and $R(D^*)$ in the decay $\bar{B}^0 \to D^{*+}\tau^-\bar{\nu}_\tau$". In: *Physical Review Letters* 118.21 (May 2017). DOI: 10.1103/physrevlett.118.211801. URL: https://doi.org/10.1103/physrevlett.118.211801.

[217] The Belle collaboration, G. Caria et al. "Measurement of $R(D)$ and $R(D^*)$ with a Semileptonic Tagging Method". In: *Physical Review Letters* 124.16 (Apr. 2020). DOI: 10.1103/physrevlett.124.161803. URL: https://doi.org/10.1103/physrevlett.124.161803.

[218] Belle II Collaboration, K. Kojima. *Recent Belle II results on semileptonic B decays and tests of lepton-flavor universality.* https://indico.cern.ch/event/1114856/contributions/5423684/attachments/2685890/4660084/2023-07-04_LP2023_KojimaFinalVer2_main.pdf. 31st International Symposium on Lepton Photon Interactions at High Energies. 2023.

[219] LHCb Collaboration, R. Aaij. "Measurement of the ratios of branching fractions $\mathcal{R}(D^*)$ and $\mathcal{R}(D^0)$". In: *arXiv* (Feb. 2023). arXiv: 2302.02886 [hep-ex].

[220] R. Aaij et al. "Measurement of the ratio of branching fractions $\mathcal{B}(B^0 \to D^{*+}\tau^-\bar{\nu}_\tau)/\mathcal{B}(B^0 \to D^{*+}\mu^-\bar{\nu}_\mu)$". In: *Physical Review Letters* 115.11 (Sept. 2015). DOI: 10.1103/physrevlett.115.111803. URL: https://doi.org/10.1103/physrevlett.115.111803.

[221] R. Aaij et al. "Measurement of the ratio of branching fractions $\mathcal{B}(B_c^+ \to J/\psi\tau^+\nu_\tau)/\mathcal{B}(B_c^+ \to J/\psi\mu^+\nu_\mu)$". In: *Physical Review Letters* 120.12 (Mar. 2018). DOI: 10.1103/physrevlett.120.121801. URL: https://doi.org/10.1103/physrevlett.120.121801.

[222] R. Aaij. "Test of Lepton Flavor Universality by the measurement of the $B^0 \to D^{*-}\tau^+\nu_\tau$ branching fraction using three-prong τ decays". In: *Physical Review D* 97.7 (Apr. 2018). DOI: 10.1103/physrevd.97.072013. URL: https://doi.org/10.1103/physrevd.97.072013.

[223] R. Aaij et al. "Measurement of the ratio of the $B^0 \to D^{*-}\tau^+\nu_\tau$ and $B^0 \to D^{*-}\mu^+\nu_\mu$ branching fractions using three-prong τ-lepton decays". In: *Physical Review Letters* 120.17 (Apr. 2018). DOI: 10.1103/physrevlett.120.171802. URL: https://doi.org/10.1103/physrevlett.120.171802.

[224] LHCb Collaboration, R. Aaij. "Test of lepton flavor universality using $B^0 \to D^{*-}\tau^+\nu_\tau$ decays with hadronic τ channels". In: *Phys. Rev. D* 108 (1 July 2023), p. 012018. DOI: 10.1103/PhysRevD.108.012018. URL: https://link.aps.org/doi/10.1103/PhysRevD.108.012018.

[225] LHCb Collaboration, R. Aaij. "Observation of the decay $\Lambda_b^0 \to \Lambda_c^+\tau^-\overline{\nu}_\tau$". In: *Phys. Rev. Lett.* 128.19 (2022), p. 191803. DOI: 10.1103/PhysRevLett.128.191803. arXiv: 2201.03497 [hep-ex].

[226] M Bordone et al. "Heavy-Quark expansion for $\bar{B} \to D_s^{(*)}$ form factors and unitarity bounds beyond the $SU(3)_F$ limit". In: *The European Physical Journal C* 80.4 (Apr. 2020). DOI: 10.1140/epjc/s10052-020-7850-9. URL: https://doi.org/10.1140/epjc/s10052-020-7850-9.

[227] LHCb Collaboration, R. Aaij et al. "Measurement of $|V_{cb}|$ with $B_s^0 \to D_s^{(*)-}\mu^+\nu_\mu$ decays". In: *Phys. Rev. D* 101.7 (2020), p. 072004. DOI: 10.1103/PhysRevD.101.072004. arXiv: 2001.03225 [hep-ex].

[228] A. Scarabotto. *Test of Lepton Flavour Universality using B_s semileptonic decays.* Master Thesis Presented 16 Jul 2020. 2020. URL: https://cds.cern.ch/record/2724839.

[229] C. Giugliano. "Test of Lepton Flavour Universality using the $B_s^0 \to D_s^-\tau^+\nu_\tau$ with 3 prongs τ^+ decays and validation of the new opto-electronics for the RICH Upgrade at the LHCb experiment". PhD thesis. Ferrara U., 2022.

[230] R. Brun and F. Rademakers. "ROOT: An object oriented data analysis framework". In: *Nucl. Instrum. Meth. A* 389 (1997), pp. 81–86. DOI: 10.1016/S0168-9002(97)00048-X.

[231] L. Anderlini et al. *The PIDCalib package.* 2016.

[232] V. V. Gligorov. *Reconstruction of the Channel $B_d^0 \to D^+\pi^-$ and Background Classification at LHCb (revised).* Tech. rep. revised version submitted on 2008-01-24 12:46:44. Geneva: CERN, 2007. URL: http://cds.cern.ch/record/1035682.

[233] LHCb Collaboration, R. Aaij et al. "Test of Lepton Flavor Universality by the measurement of the $B^0 \to D^{*-}\tau^+\nu_\tau$ branching fraction using three-prong τ decays". In: *Phys. Rev. D* 97.7 (2018), p. 072013. DOI: 10.1103/PhysRevD.97.072013. arXiv: 1711.02505 [hep-ex].

[234] T. Chen and C. Guestrin. "XGBoost: A Scalable Tree Boosting System". In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.* KDD '16. San Francisco, California, USA: ACM, 2016, pp. 785–794. ISBN: 978-1-4503-4232-2. DOI: 10.1145/2939672.2939785. URL: http://doi.acm.org/10.1145/2939672.2939785.

[235] LHCb Collaboration, F. Betti et al. *Measurement of $\mathcal{B}(B^0 \to D^{*-}\tau^+\nu_\tau)$ and $R(D^*)$ with τ three-prong pionic decays.* 2015. URL: https://cds.cern.ch/record/2040543.

[236] F. J. Massey. “The Kolmogorov-Smirnov Test for Goodness of Fit”. In: *Journal of the American Statistical Association* 46.253 (1951), pp. 68–78. DOI: 10.1080/01621459.1951.10500769. URL: https://www.tandfonline.com/doi/abs/10.1080/01621459.1951.10500769.

[237] T Akiba et al. *Optuna: A Next-generation Hyperparameter Optimization Framework.* 2019. arXiv: 1907.10902 [cs.LG].

[238] L Lista. “Practical Statistics for Particle Physicists”. In: *2016 European School of High-Energy Physics.* 2017, pp. 213–258. DOI: 10.23730/CYRSP-2017-005.213. arXiv: 1609.04150 [physics.data-an].

[239] T Kluyver et al. “Jupyter Notebooks – a publishing format for reproducible computational workflows”. In: *Positioning and Power in Academic Publishing: Players, Agents and Agendas.* IOS Press. 2016, pp. 87–90.

[240] GitLFS. *Git Large File storage.* https://git-lfs.com/. [Accessed 29-September-2023]. 2023.

[241] MLFlow Community. *MLFlow.* http://mlflow.org. [Accessed 30-Sept-2023]. 2023.

[242] Pachyderm. *Pachyderm.* https://www.pachyderm.com/. [Accessed 30-Sept-2023]. 2023.

[243] iterative.ai. *Data Version Control.* http://dvc.org. [Accessed 30-Sept-2023]. 2023.

[244] LHCb Collaboration, R. Aaij et al. “Measurement of b-hadron production fractions in 7 TeV pp collisions”. In: *Phys. Rev. D* 85 (2012), p. 032008. DOI: 10.1103/PhysRevD.85.032008. arXiv: 1111.2357 [hep-ex].

Appendix A

Possible normalisation channels

Table A.1: Channels with topology similar to the signal, their Branching Fractions, obtained by combining the values in ref. [197], and the relative error. In case of decays with B_d^0 instead of B_s^0 the Branching Fraction total is re-weighted for the f_d/f_s ratio [227, 244].

Decay type	Effective Branching Fraction	Relative error [%]
$B_s^0 \to D_s^-(\to K^+K^-\pi^-)\pi^+\pi^-\pi^+$	$(32.8 \pm 5.4)\times10^{-5}$	16.5
$B_s^0 \to D_s^-(\to \phi(\to K^+K^-)\pi^-)\pi^+\pi^-\pi^+$	$(13.5 \pm 2.2)\times10^{-5}$	16.6
$B_s^0 \to D_s^-(\to K^+K^-\pi^-)D_s^+(\to \pi^+\pi^-\pi^+)$	$(0.26 \pm 0.03)\times10^{-5}$	12.1
$f_d/f_s{\times}B_d^0 \to D^-(\to K^+K^-\pi^-)\pi^+\pi^-\pi^+$	$(24.3 \pm 2.5)\times10^{-5}$	10.4
$f_d/f_s{\times}B_d^0 \to D^-(\to K^+\pi^-\pi^-)\pi^+\pi^-\pi^+$	$(23.6 \pm 2.4)\times10^{-4}$	10.4
$f_d/f_s \times B_d^0 \to D^-(\to \phi(\to K^+K^-)\pi^-)\pi^+\pi^-\pi^+$	$(6.76 \pm 0.7)\times10^{-5}$	10.7
$f_d/f_s \times B_d^0 \to D^-(\to K^+K^-\pi^-)D_s^+(\to \pi^+\pi^-\pi^+)$	$(0.32 \pm 0.04)\times10^{-5}$	12.1
$f_d/f_s \times B_d^0 \to D^-(\to \pi^+\pi^-\pi^-)D_s^+(\to K^+K^-\pi^+)$	$(0.53 \pm 0.06)\times10^{-5}$	11.8
signal	$(4.6 \pm 0.5)\ \times10^{-5}$	–

Appendix B

Stripping selection cuts

Table B.1 lists the cuts applied by the stripping framework to the candidates selected by the Bs2DsTauNuForB2XTauNu and B0d2DTauNuForB2XTauNu selection lines used by the $R\left(D_s\right)$ analysis. Both also apply minimal requirements on reconstructed events to select decay candidates good enough for analysis.

- DOCA is the distance of closest approach between the tracks composing the particle.
- DIRA is the cosine of the angle between the momentum of the particle and the direction vector from its creation vertex.
- IP χ^2 for a track is the difference between the χ^2 of the primary vertex reconstructed with and without the track under consideration.
- PIDK is the combined delta log likelihood for the Kaon hypothesis with reference to the pion one ($\Delta \log \mathcal{L}_{K-\pi}$).

Table B.1: Cuts implemented in the `Bs2DsTauNuForB2XTauNu` and `B0d2DTauNuForB2XTauNu` selection lines.

Cut	Value
$B_{(s)}$	
$\Delta(M)$	(-2579)-300 or 720-1721 MeV/c^2
Max. DOCA	< 0.15 mm
DIRA	> 0.995
$D^+_{(s)}$	
p_T	> 1600 MeV/c
$\|M - M_{D^+_{(s)}}\|$	< 40.0MeV/c^2
DIRA	> 0.995
Vertex distance χ^2	> 36 if D_s^+ > 50 if D^+
Vertex $\chi^2/NDOF$	< 10
IP χ^2	> 10
K from $D_{(s)}$	
p_T	> 1500MeV/c
Track $\chi^2/NDOF$	< 30
IP χ^2	> 10
Ghost Probability	< 0.4
PIDK	> 3
π from $D_{(s)}$	
p_T	> 150 MeV/c
Track $\chi^2/NDOF$	< 3
IP χ^2	> 10
Ghost Probability	< 0.4
PIDK	< 50
τ	
$m(\pi\pi\pi)$	400-3500 MeV/c^2
$m(\pi_1\pi_2)$ or $m(\pi_2\pi_3)$	< 1670 MeV/c^2
Max. DOCA	< 0.15 mm
DIRA	> 0.99
Vertex χ^2	< 25
max 1 pion with p_T	< 300 MeV/c
min 2 pions with IP χ^2	> 5
π from τ	
p_T	> 150 MeV/c
Track $\chi^2/NDOF$	< 4
IP χ^2	> 4
Ghost Probability	< 0.4
PIDK	< 8

Appendix C

Reconstruction of the Double Charm decays

As discussed in Sec. 6.3, the dominant background after the selection is due to Double Charm b-hadron decays that have a topology similar to the signal as the charm-hadrons have a lifetime similar to the τ lepton and have sizeable branching fractions to final states containing at least three pions.

To suppress such background a BDT selection is developed which exploits the kinematic features of these $H_b \to D_s H_c$ decays. In particular, knowing the flight directions of the H_b, D_s and H_c candidates from their vertex ($\hat{u}_{B_s}$, $\hat{u}_{D_s^+}$ and $\hat{u}_{H_c}$) (c.f. Figure C.1) and assuming the H_b mass is the B_s^0 one, from the momentum conservation

$$|\vec{p}_{B_s}|\hat{u}_{B_s} = |\vec{p}_{D_s^+}|\hat{u}_{D_s^+} + |\vec{p}_{H_c}|\hat{u}_{H_c} \tag{C.1}$$

one can derive the following two identities:

$$\text{(vectorial)}\ \ |\vec{p}_{B_s}|\hat{u}_{B_s} \times \hat{u}_{H_c} = |\vec{p}_{D_s^+}|\hat{u}_{D_s^+} \times \hat{u}_{H_c} + |\vec{p}_{H_c}|\hat{u}_{H_c} \times \hat{u}_{H_c} = |\vec{p}_{D_s^+}|\hat{u}_{D_s^+} \times \hat{u}_{H_c} \tag{C.2}$$

$$\text{(scalar)}\ \ |\vec{p}_{B_s}|\hat{u}_{B_s} \cdot \hat{u}_{B_s} = |\vec{p}_{D_s^+}|\hat{u}_{D_s^+} \cdot \hat{u}_{B_s} + |\vec{p}_{H_c}|\hat{u}_{H_c} \cdot \hat{u}_{B_s} = |\vec{p}_{D_s^+}|\hat{u}_{D_s^+} \cdot \hat{u}_{B_s} + |\vec{p}_{B_s} - \vec{p}_{D_s^+}|\hat{u}_{H_c} \cdot \hat{u}_{B_s} \tag{C.3}$$

which lead to, respectively:

$$\text{(vectorial)}\ \ |\vec{p}_{B_s}| = \frac{|\vec{p}_{D_s^+} \times \hat{u}_{H_c}|}{|\hat{u}_{B_s} \times \hat{u}_{H_c}|} \tag{C.4}$$

$$\text{(scalar)}\quad |\vec{p}_{B_s}| = \vec{p}_{D_s^+} \cdot \hat{u}_{B_s} + (\vec{p}_{B_s} \cdot \hat{u}_{H_c})(\hat{u}_{H_c} \cdot \hat{u}_{B_s}) - (\vec{p}_{D_s^+} \cdot \hat{u}_{H_c})(\hat{u}_{H_c} \cdot \hat{u}_{B_s})\ \ (^*) \tag{C.5}$$

$$|\vec{p}_{B_s}| - (\vec{p}_{B_s} \cdot \hat{u}_{H_c})(\hat{u}_{H_c} \cdot \hat{u}_{B_s}) = \vec{p}_{D_s^+} \cdot \hat{u}_{B_s} - (\vec{p}_{D_s^+} \cdot \hat{u}_{H_c})(\hat{u}_{H_c} \cdot \hat{u}_{B_s}) \tag{C.6}$$

$$|\vec{p}_{B_s}| - |\vec{p}_{B_s}|(\hat{u}_{B_s} \cdot \hat{u}_{H_c})(\hat{u}_{H_c} \cdot \hat{u}_{B_s}) = \vec{p}_{D_s^+} \cdot \hat{u}_{B_s} - (\vec{p}_{D_s^+} \cdot \hat{u}_{H_c})(\hat{u}_{H_c} \cdot \hat{u}_{B_s}) \tag{C.7}$$

$$|\vec{p}_{B_s}|(1 - (\hat{u}_{H_c} \cdot \hat{u}_{B_s})^2) = \vec{p}_{D_s^+} \cdot \hat{u}_{B_s} - (\vec{p}_{D_s^+} \cdot \hat{u}_{H_c})(\hat{u}_{H_c} \cdot \hat{u}_{B_s}) \tag{C.8}$$

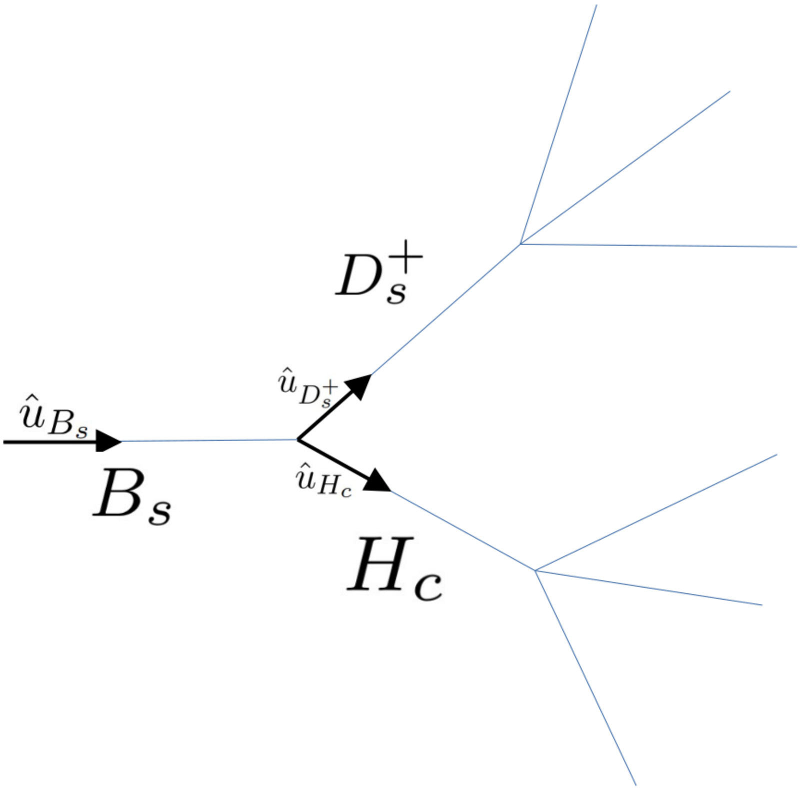


Figure C.1: Kinematics of the double charm decay.

[1]

$$\text{(scalar)}\ \ |\vec{p}_{B_s}| = \frac{\vec{p}_{D_s^+}\cdot\hat{u}_{B_s} - (\vec{p}_{D_s^+}\cdot\hat{u}_{H_c})(\hat{u}_{H_c}\cdot\hat{u}_{B_s})}{(1-(\hat{u}_{H_c}\cdot\hat{u}_{B_s})^2)} \tag{C.9}$$

Similarly to Eq. C.4 and Eq. C.9 for B_s^0, one can evaluate the vectorial and *scalar* approximated H_c momenta as

$$\text{(vectorial)}\ \ |\vec{p}_{H_c}| = \frac{|\vec{p}_{D_s^+}\times\hat{u}_{B_s}|}{|\hat{u}_{B_s}\times\hat{u}_{H_c}|} \tag{C.10}$$

$$\text{(scalar)}\ \ |\vec{p}_{H_c}| = \frac{(\vec{p}_{D_s^+}\cdot\hat{u}_{B_s})(\hat{u}_{H_c}\cdot\hat{u}_{B_s}) - (\vec{p}_{D_s^+}\cdot\hat{u}_{H_c})}{(1-(\hat{u}_{H_c}\cdot\hat{u}_{B_s})^2)} \tag{C.11}$$

Such relations don't assume any value for the H_c mass and are valid also in the case the H_c decays to three charged pions and additional neutral or undetected particles. In such cases, however, the B_s vertex position needs to be corrected to account for the undetected particles. We temporarily assume the same correction that was found in the $R(D^*)$ analysis which was found using simulation[2]. This correction allows to recompute the B_s vertex position and updated approximated values for the *scalar* and *vectorial* B_s and H_c moments.

[1] Here we use $\vec{p}_{H_c} = \vec{p}_{B_s} - \vec{p}_{D_s^+}$ and $\vec{p}_{H_c}\cdot\hat{u}_{H_c} = |\vec{p}_{H_c}| = \vec{p}_{B_s}\cdot\hat{u}_{H_c} - \vec{p}_{D_s^+}\cdot\hat{u}_{H_c}$

[2] $dz = \exp(p_0 + p_1\cdot Y_M)$ with $p_0 = 1.03$ and $p_1 = -0.002$

Appendix D

Inclusive Monte Carlo background characterisation

	count	Percentage
Category		
Xc_signal_Ypis_B_vertex_fromBs	285302	29.75
Xc_background	157220	16.39
Xc_signal_Ypis_diffVertex_doubleCharm_OneFromB	130226	13.58
Xc_signal_Ypis_displaced_fromBs_fromDs	78099	8.14
Xc_signal_Ypis_displaced_fromB0_fromDp	65190	6.80
Xc_signal_Ypis_nomatch_doubleCharm	38950	4.06
Xc_signal_Ypis_displaced_fromBp_fromD0	30025	3.13
Xc_signal_Ypis_nomatch_Prompt	26858	2.80
Xc_signal_Ypis_diffVertex_doubleCharm_TwoFromB	23271	2.43
Xc_signal_Ypis_displaced_fromLambdab_fromLambdac	22561	2.35
Xc_signal_Ypis_displaced_fromBs_fromDp	16615	1.73
Xc_signal_Ypis_diffVertex_CharmStrange	12252	1.28
Xc_signal_Ypis_B_vertex_fromOtherB	11730	1.22
Xc_signal_Ypis_displaced_fromBs_fromTau	9767	1.02
Xc_signal_Ypis_displaced_fromBp_fromDp	8568	0.89
Xc_signal_Ypis_diffVertex_normlike	8426	0.88
Xc_signal_Ypis_diffVertex_doubleCharm	7933	0.83
Xc_signal_Ypis_displaced_fromB0_fromD0	6635	0.69
Xc_signal_Ypis_displaced_fromB0_fromDs	4702	0.49
Xc_signal_Ypis_nomatch_charmStrange	2865	0.30
Xc_signal_Ypis_diffVertex_SomeFromPV	2764	0.29
Xc_signal_Ypis_diffAncestorYXc	2562	0.27
Xc_signal_Ypis_displaced_fromBs_fromD0	1909	0.20
Xc_signal_Ypis_displaced_fromBp_fromDs	1504	0.16
others	1389	0.14
Xc_signal_Ypis_displaced_fromBs_fromDs_fromTau	1263	0.13
Xc_signal_Ypis_displaced_fromLambdab_fromDs	542	0.06

Table D.1: Detailed categorisation of inclusive MC data for 2012 after cut on B_M $<$ 5000 MeV/c^2

	count	Percentage
Category		
Xc_signal_Ypis_displaced_fromBs_fromDs	47066	20.19
Xc_signal_Ypis_displaced_fromB0_fromDp	46669	20.02
Xc_signal_Ypis_diffVertex_doubleCharm_OneFromB	35243	15.12
Xc_background	20449	8.77
Xc_signal_Ypis_displaced_fromBp_fromD0	18506	7.94
Xc_signal_Ypis_nomatch_doubleCharm	15667	6.72
Xc_signal_Ypis_displaced_fromBs_fromDp	10747	4.61
Xc_signal_Ypis_displaced_fromLambdab_fromLambdac	6433	2.76
Xc_signal_Ypis_displaced_fromBp_fromDp	5843	2.51
Xc_signal_Ypis_diffVertex_doubleCharm	4607	1.98
Xc_signal_Ypis_displaced_fromB0_fromD0	3804	1.63
Xc_signal_Ypis_displaced_fromBs_fromTau	3489	1.50
Xc_signal_Ypis_diffVertex_CharmStrange	3373	1.45
Xc_signal_Ypis_displaced_fromB0_fromDs	2108	0.90
Xc_signal_Ypis_diffVertex_doubleCharm_TwoFromB	1298	0.56
Xc_signal_Ypis_B_vertex_fromBs	1149	0.49
others	1015	0.44
Xc_signal_Ypis_displaced_fromBs_fromD0	1005	0.43
Xc_signal_Ypis_displaced_fromBs_fromDs_fromTau	985	0.42
Xc_signal_Ypis_diffAncestorYXc	917	0.39
Xc_signal_Ypis_nomatch_charmStrange	874	0.37
Xc_signal_Ypis_nomatch_Prompt	703	0.30
Xc_signal_Ypis_displaced_fromBp_fromDs	686	0.29
Xc_signal_Ypis_diffVertex_SomeFromPV	474	0.20

Table D.2: Detailed categorisation of inclusive MC data for 2012 after cut B_Y_SEP < -4.5. It also includes a cut on q2_2 > 0 to ensure validaity of the data.

Appendix E

Double charm suppression BDT

This chapter details the study of the BDT used to suppress the main double charm background.

E.1 BDT inputs

As mentioned in Sec. 6.5 Monte Carlo samples of signal and inclusive b-hadron decays (both incl_$H_b \to D_s 3\pi X$ and incl_$H_b \to D_s H_c(\to 3\pi)X$) are used. Among possible inputs, those observables that show a discrimination power between signal and background are used. They consist in kinematic variables computed assuming the signal decay and after the correction for the undetected neutrino (see Sec. 6.2.1):

- `B_correctedMass` and `Y_correctedMass`, *i.e.* the mass of the B_s and τ candidates,
- `missing_mass_2` and `missing_pY_mass`, *i.e.* the reconstructed missing mass of the B$_s$ and τ candidates,
- `B_pT_Bdir`, *i.e.* component of the B momenta transverse to the B reconstructed direction (from the PV to the B decay vertex),
- `log(Y_PE)`, *i.e.* logarithm of the τ candidate energy,

The following observables instead are computed assuming the decay $B_s^0 \to D_s H_c(\to 3\pi X$ (see Appendix. C):

- `mN2V`, *i.e.* the squared mass of the reconstructed neutral vector,
- `log(sqrt(abs(nDs2vn)))`, *i.e.* function of the reconstructed mass of the H_c system,
- `log(abs((PBsn-PBvn)/PBvn))`, *i.e.* the normalised difference between the different estimates of the B_s candidate momentum (scalar and vectorial),
- `log(abs(PBvn/B_P))` and `log(abs(PBv/B_P))`, *i.e.* the ratio between the reconstructed B_s momentum and the visible one, with or without the corrected B_s vertex information, respectively,

- `log(abs(PBsn))`, *i.e.* the B_s momentum reconstructed using the scalar product method and the corrected B_s vertex,

The following observables complete the list:

- `B_M`, *i.e.* B_s invariant mass reconstructed from the momenta of the visible final state particles, without any correction applied,
- `max_m2pi` and `min_m2pi`, *i.e.* maximum and minimum values of the invariant mass squared for the two $\pi^+\pi^-$ combinations,
- `Y_BPVVDR`, *i.e.* radial distance of the τ candidate decay vertex from the PV,
- `BDT_Iso`, *i.e.* the output of the isolation-BDT described in Sec. 6.2,
- `Y_0_*_#c_PZ`, *i.e.* z component of the sum of momenta of the neutral (#=nc) and charged (#=cc) particles detected in a cone of angle "*" (0.20, 0.30 and 0.40 in units of $R \equiv \sqrt{\Delta\phi^2 + \Delta\eta^2}$) around the τ candidate,
- `Y_0_*_#c_mult`, *i.e.* multiplicity of neutral (#=nc) and charged (#=cc) particles in a cone of angle "*" as defined above.

Figures E.1, E.2 and E.3 show the distributions of the inputs for signal and double charm background.

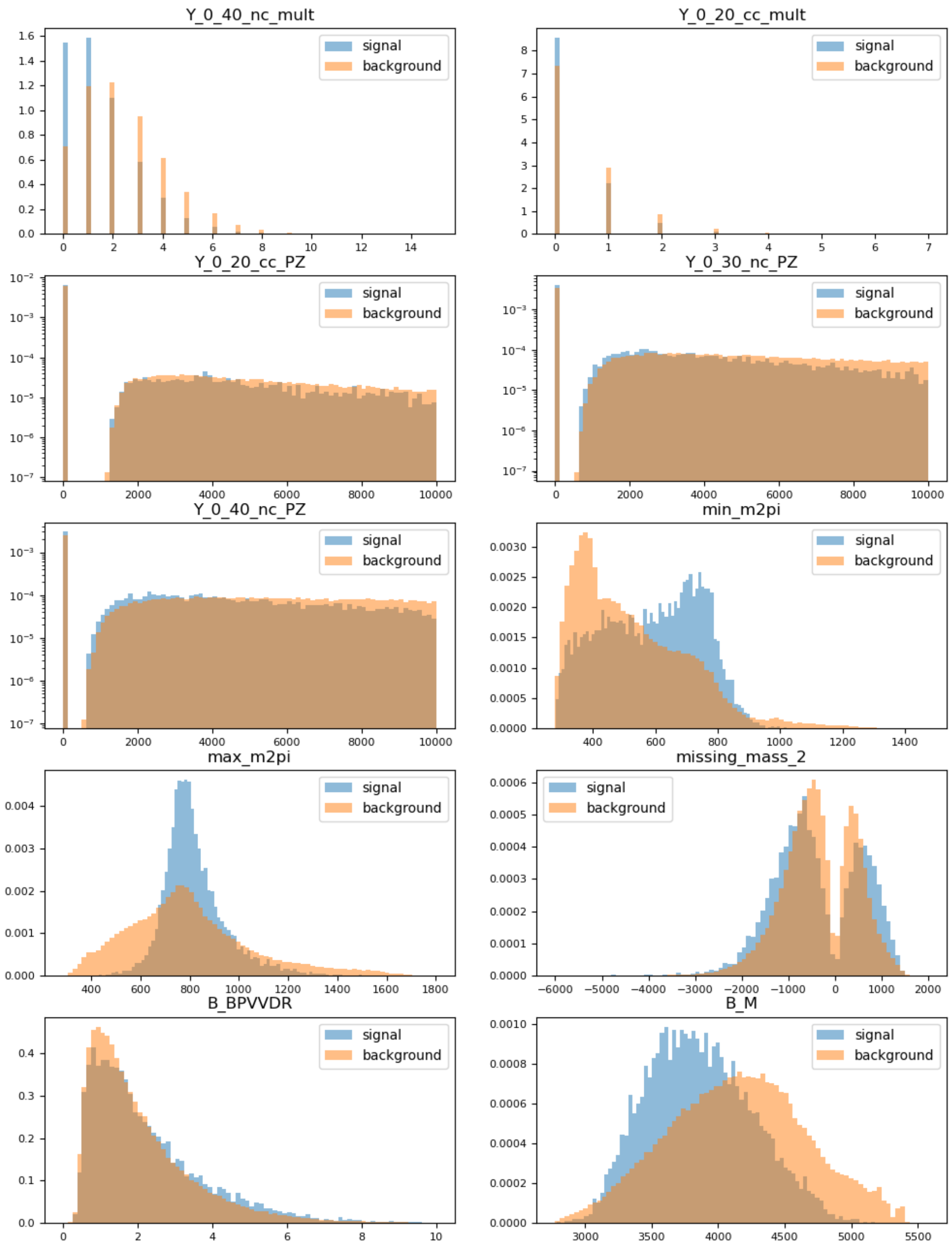


Figure E.1: Distributions of the input variables for the BDT to suppress the double charm background.

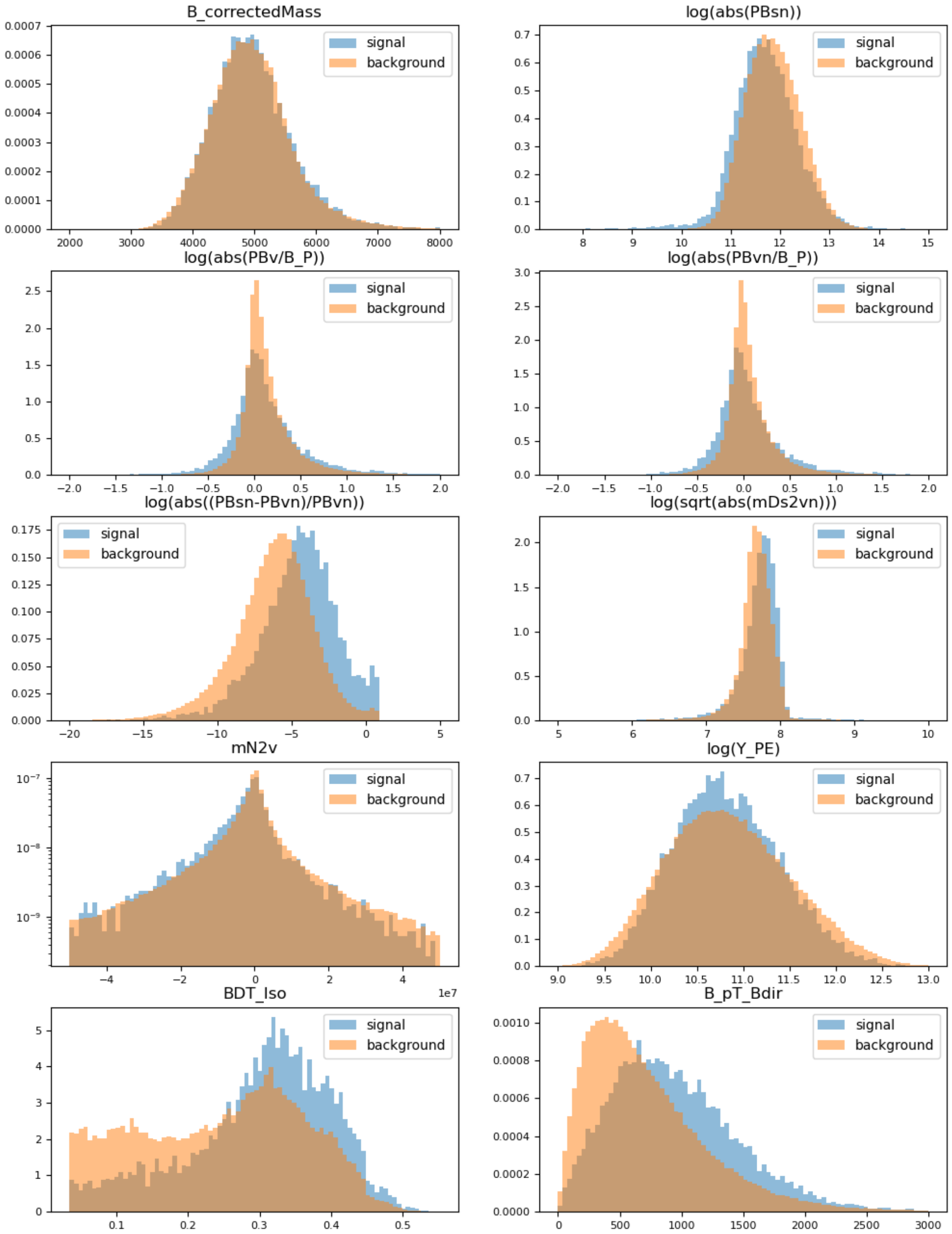


Figure E.2: Distributions of the input variables for the BDT to suppress the double charm background.

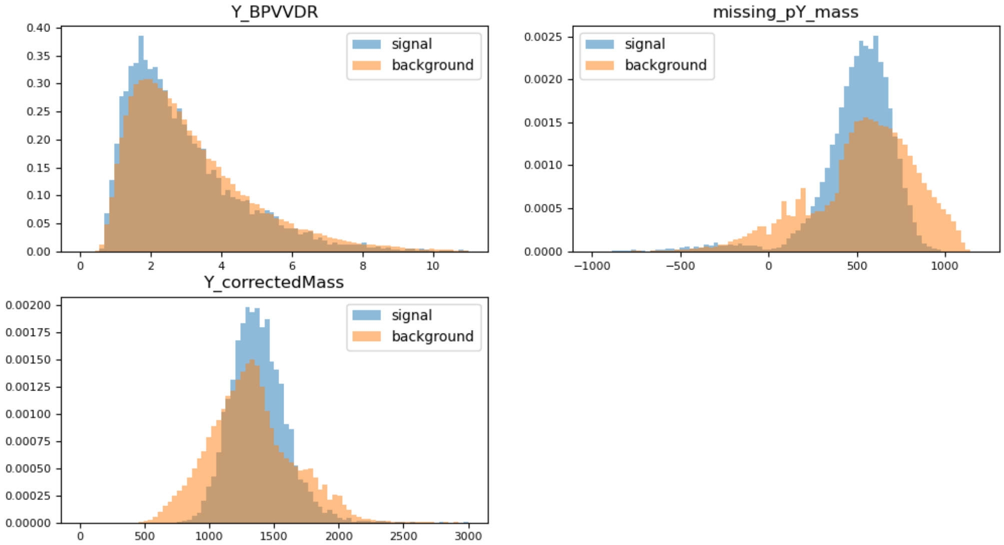
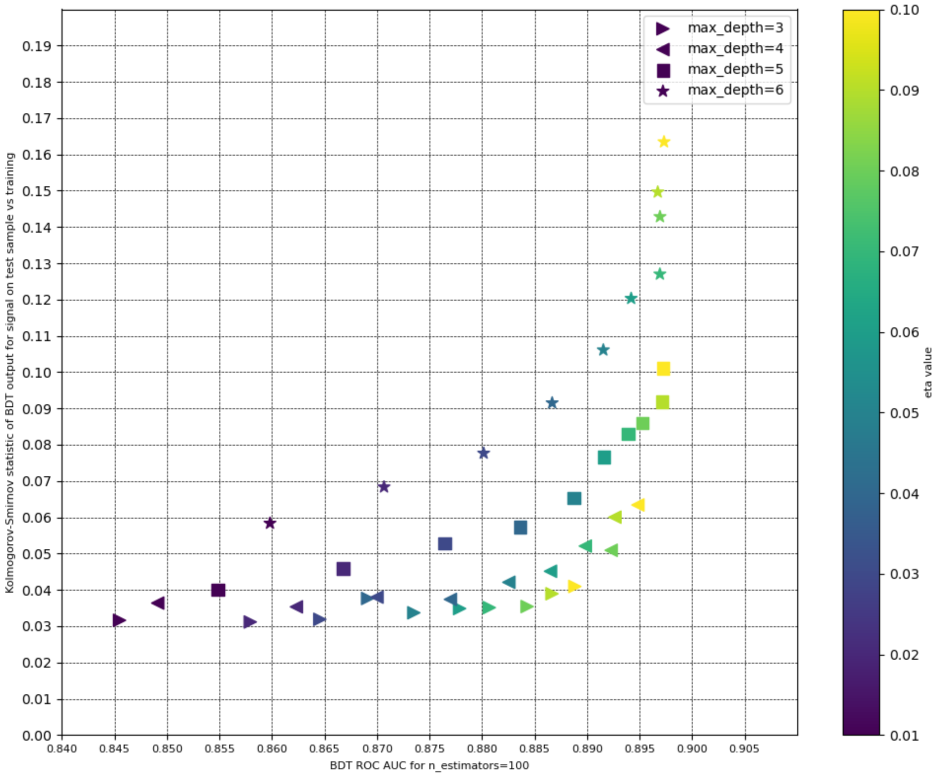


Figure E.3: Distributions of the input variables for the BDT to suppress the double charm background.

E.2 BDT tuning

In order to find the best combination of parameters to tune XGBoost to separate double charm decays in our dataset, various combinations of the XGBoost tuning parameters were investigated, as shown in Figures E.4 E.5.



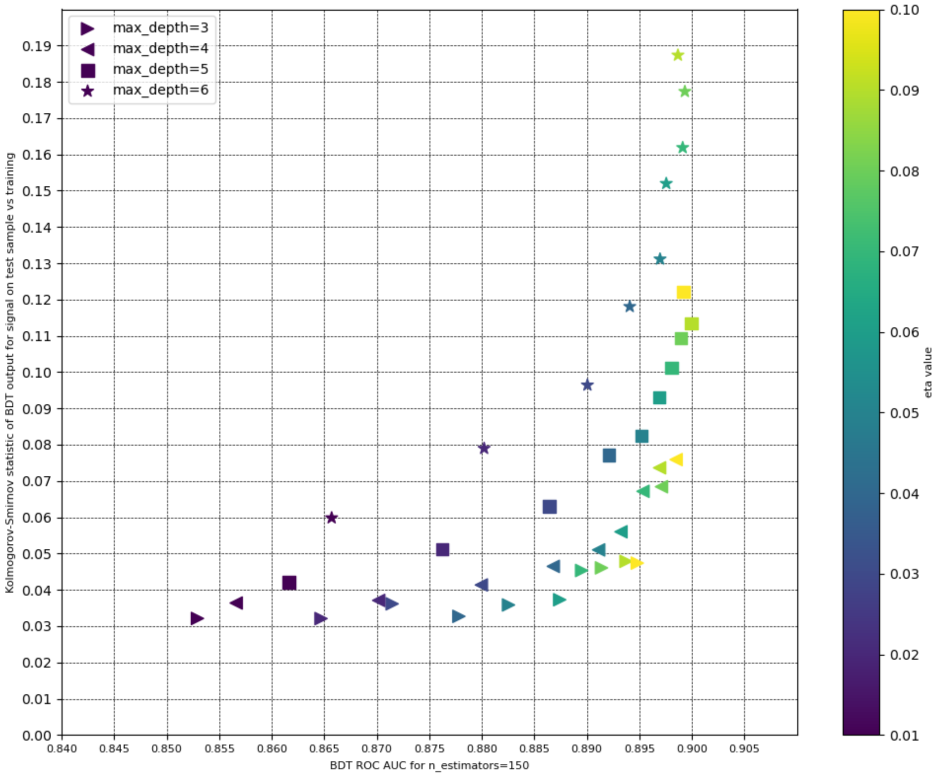


Figure E.4: BDT overfitting as a function of performance with various combinations of *eta* and *max_depth*, with*n_estimators* equal to 100 (top) or 150 (bottom).

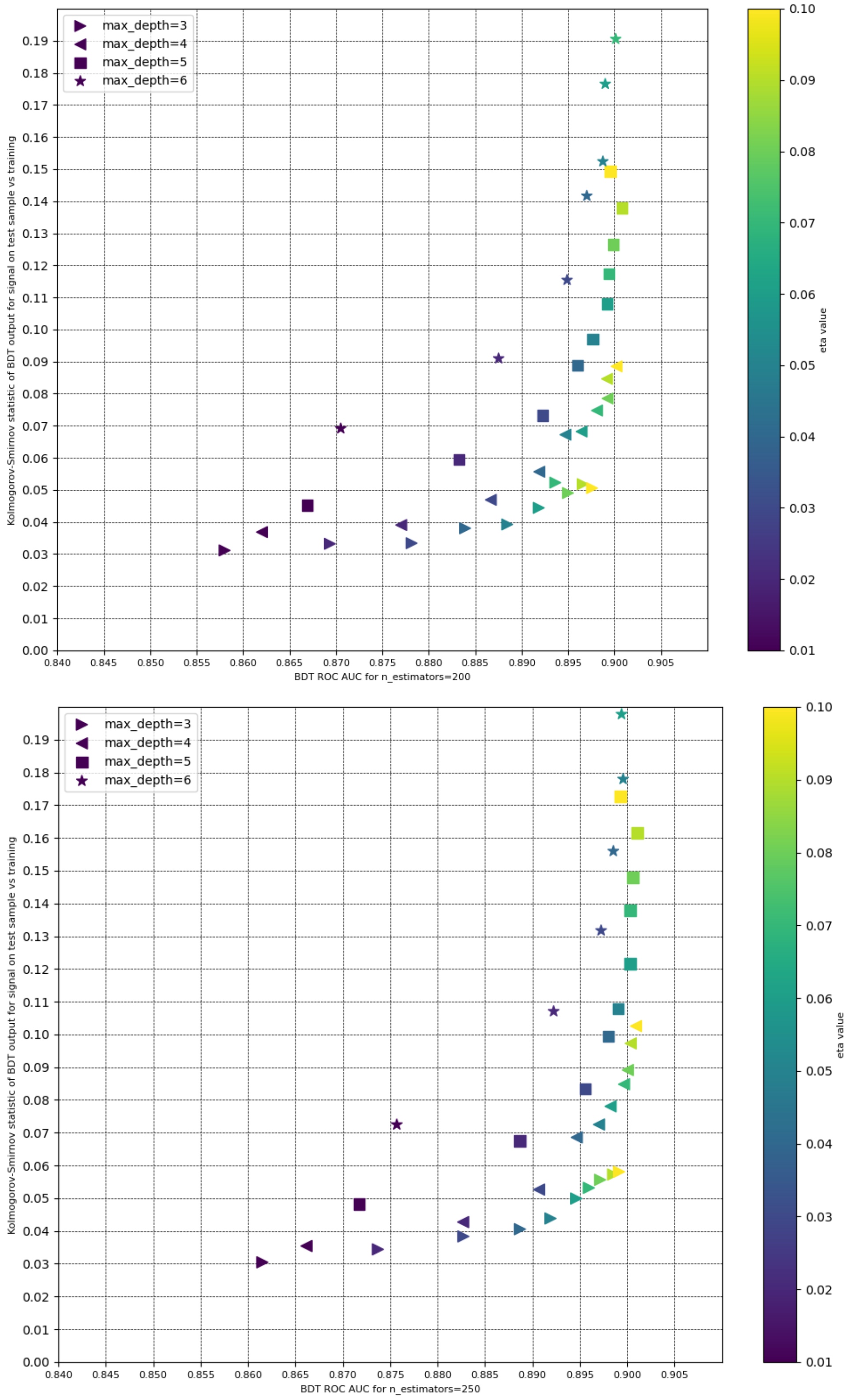


Figure E.5: BDT overfitting as a function of performance with various combinations of *eta* and *max_depth*, with*n_estimators* equal to 200 (top) or 250 (bottom).

This give an idea of the impact of the parameters on the performance and overfitting of the model: increasing the *max_depth* leads to overfitting, while the maximum ROC AUC is clearly around 0.9. The Optuna optimisation software was subsequently used to fine tune the BDT parameters, leading to the configuration of *n_estimators* of 150, a *max_depth* of 3 and *eta* of 0.04, leading to an area of the ROC curve of 0.88. The loss function evolution during the training is shown in Figure E.6, with in blue the training loss which is not significantly different from the test loss.

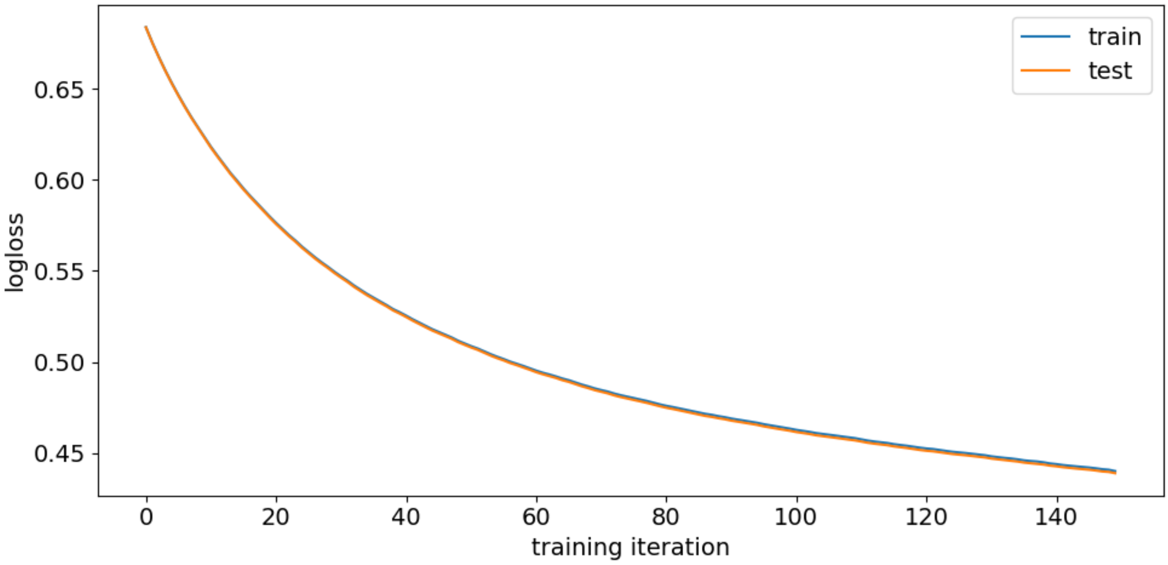


Figure E.6: Evolution of the loss function during the training.

E.3 Results obtained training on different double charm background species

Despite the similarities of the double charm background, each specific charm hadron features some characteristics that could eventually be exploited to reach a better discrimination power with respect to the signal. Such differences are, for example, the lifetime, the relative branching fractions in three or more charged mesons, the decay dynamics, etc. We therefore performed a test where different BDTs were trained on background samples split by the charm-hadron species associated to the Y candidate: D_s^-, D^-, D^0.

The same XGBoost configuration was used to train the BDTs against specific background categories, and the result are shown in Figures E.7, E.8, E.9 and E.10. While of course each specifically trained BDT is optimal to separate its target decay, they do not generalise well to the whole sample, while the BDT trained against all double charm candidates tend to be relatively efficient on each subcategory.

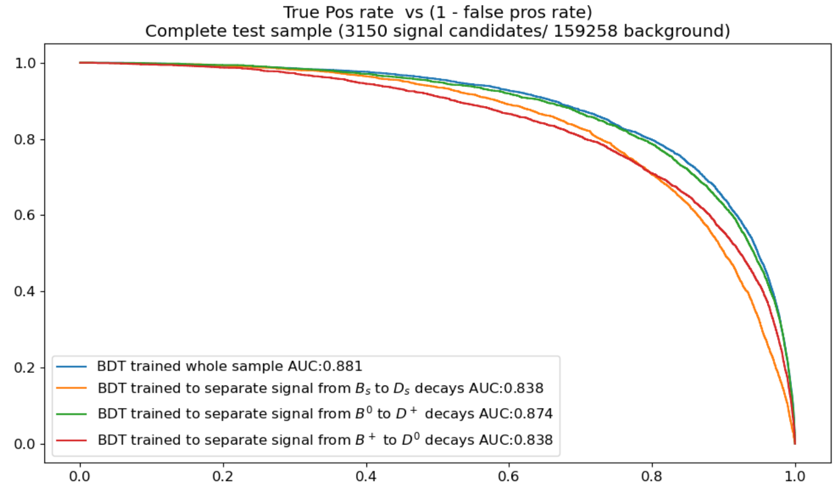


Figure E.7: Comparison of the ROC curves for BDTs trained against specific categories

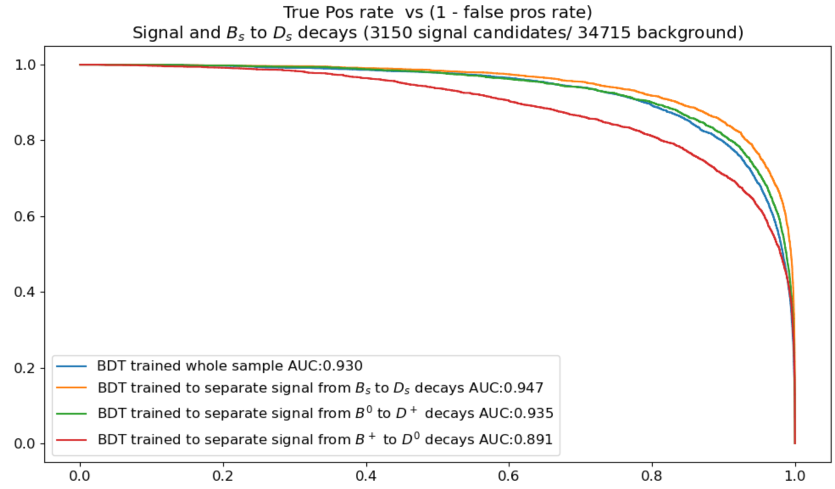


Figure E.8: Comparison of the ROC curves for BDTs trained against specific categories

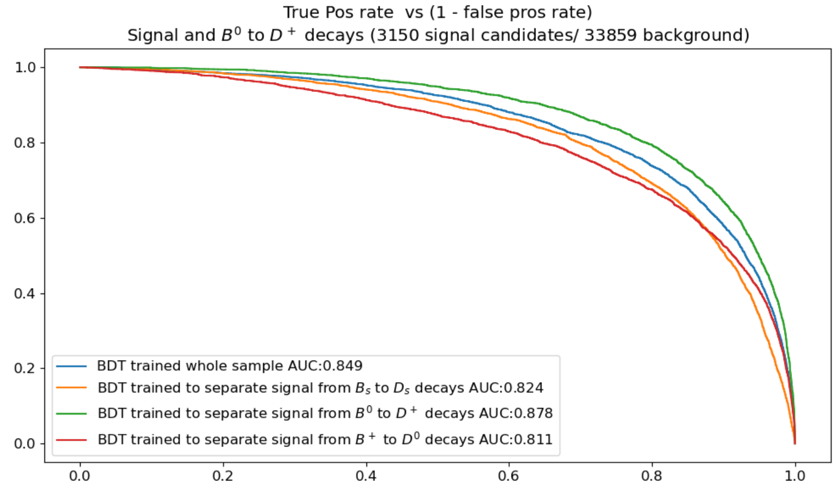


Figure E.9: Comparison of the ROC curves for BDTs trained against specific categories

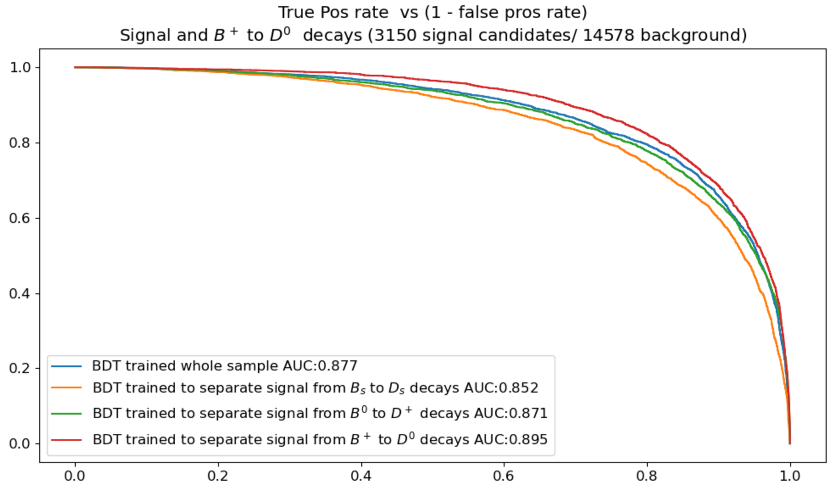


Figure E.10: Comparison of the ROC curves for BDTs trained against specific categories

E.4 Template grouping

In this section are reported the distributions of the observables used in the fit. Each plot correspond to a group of background categories that have similar distributions based on a loose cut of the p-value of 0.05 for the Kolmogorov-Smirnov test. Categories containing less than 100 events are grouped a-priori in the group “others”.

The chosen criteria indicates 5 groups based on the q^2 distributions, 4 groups based on the τ distributions, and two based on the BDT distributions. The final grouping is obtained by combining the information of the three groups. It consists of 10 groups (Table 6.11, each corresponding to categories that have similar distributions in all three observables.

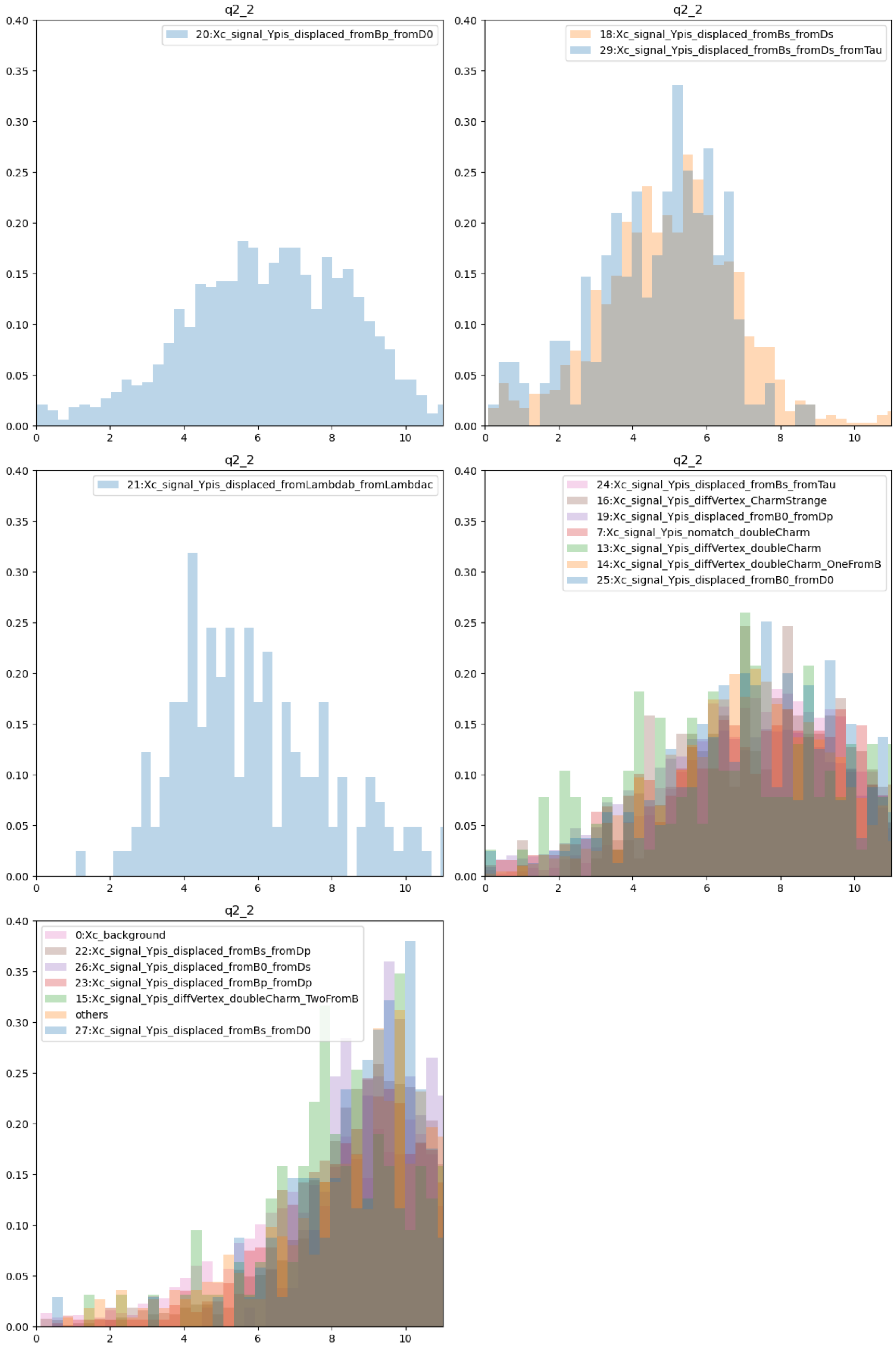


Figure E.11: q2_2 histograms for CUT1 grouped by similarity with threshold 0.05

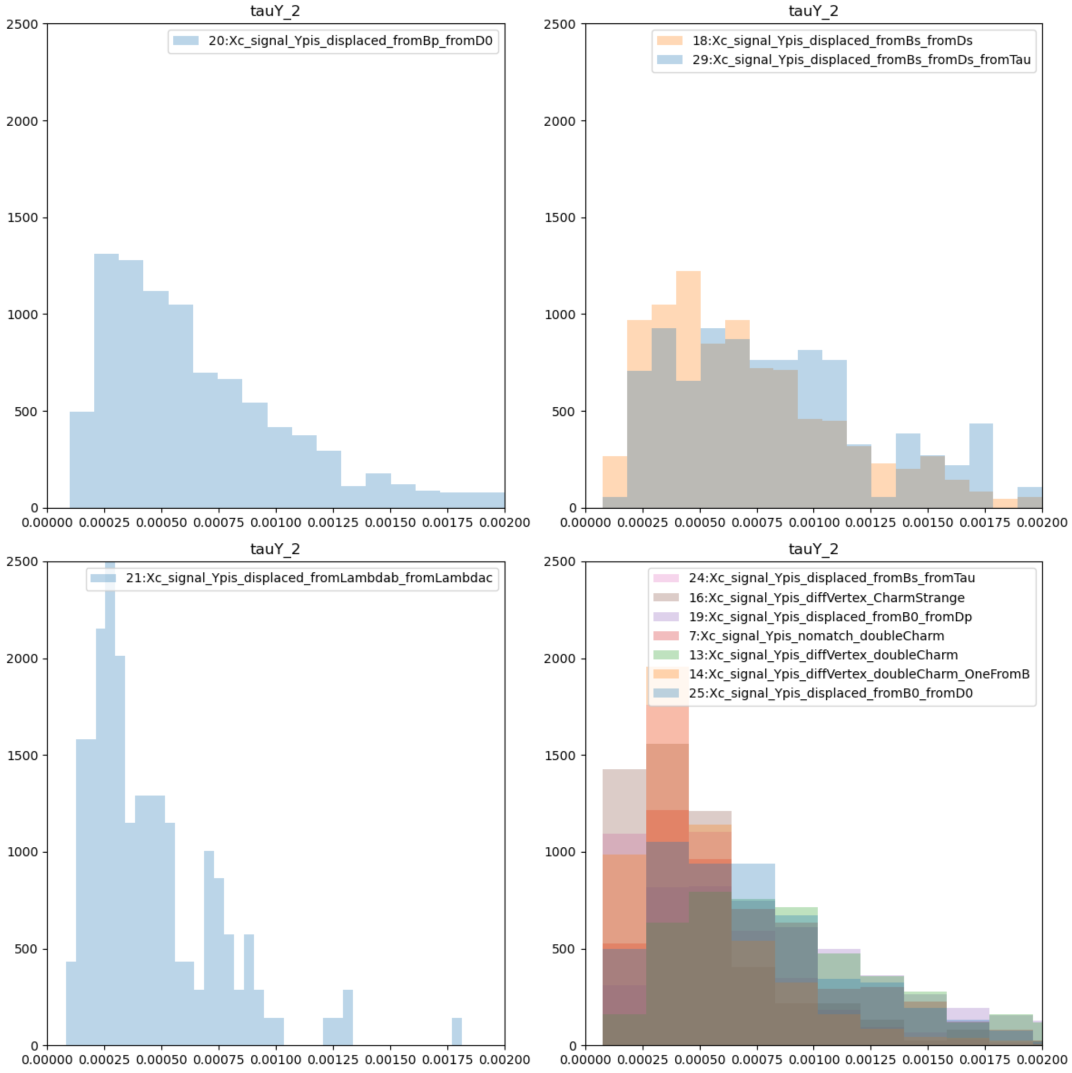


Figure E.12: tauY_2 histograms for CUT1 grouped by similarity with threshold 0.05

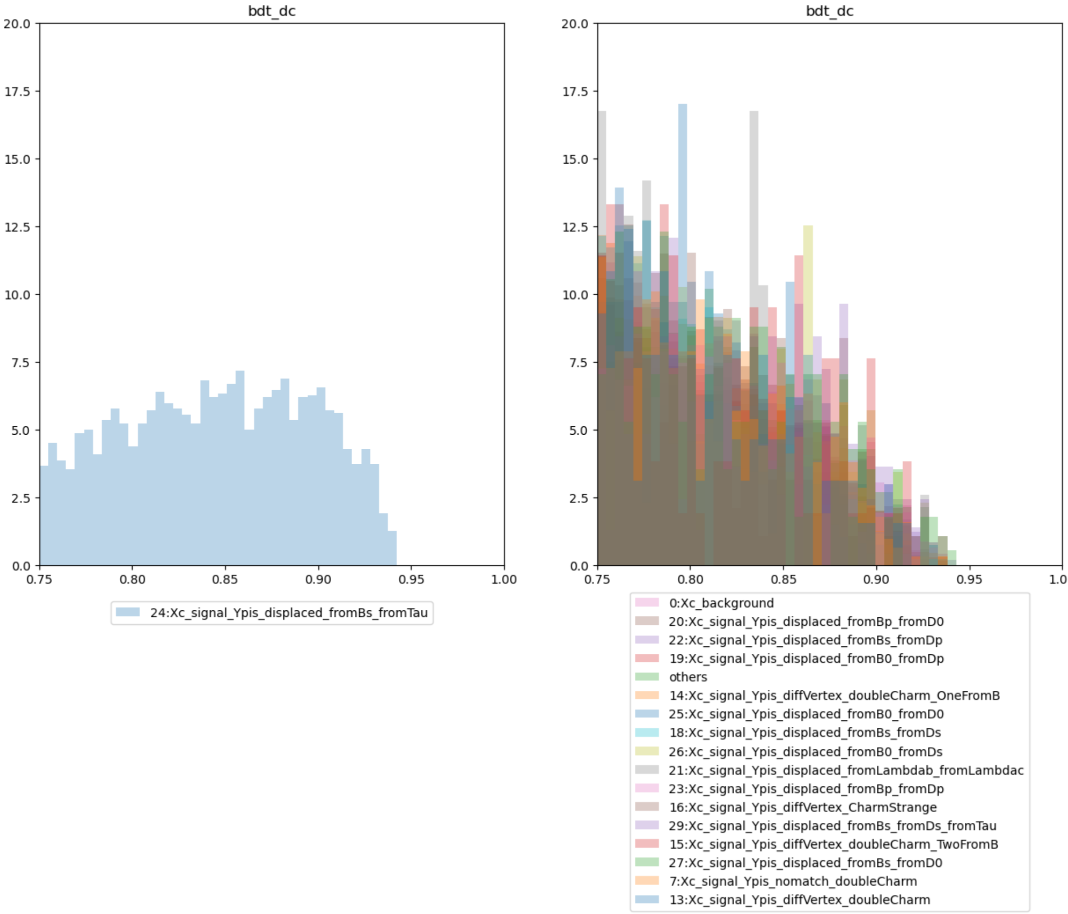


Figure E.13: bdt_dc histograms for CUT1 grouped by similarity with threshold 0.05

E.5 Checking the correlation between the template grouped

The grouping of the templates done in Section 6.6.1 was done by comparing the projection of the three dimension histogram onto each of the fit variables for each category. In order to check that this makes sense, we also checked that the groups made sense by checking that the 2D histograms are consistent, as shown by Figure E.14, E.15, E.16 and E.17.

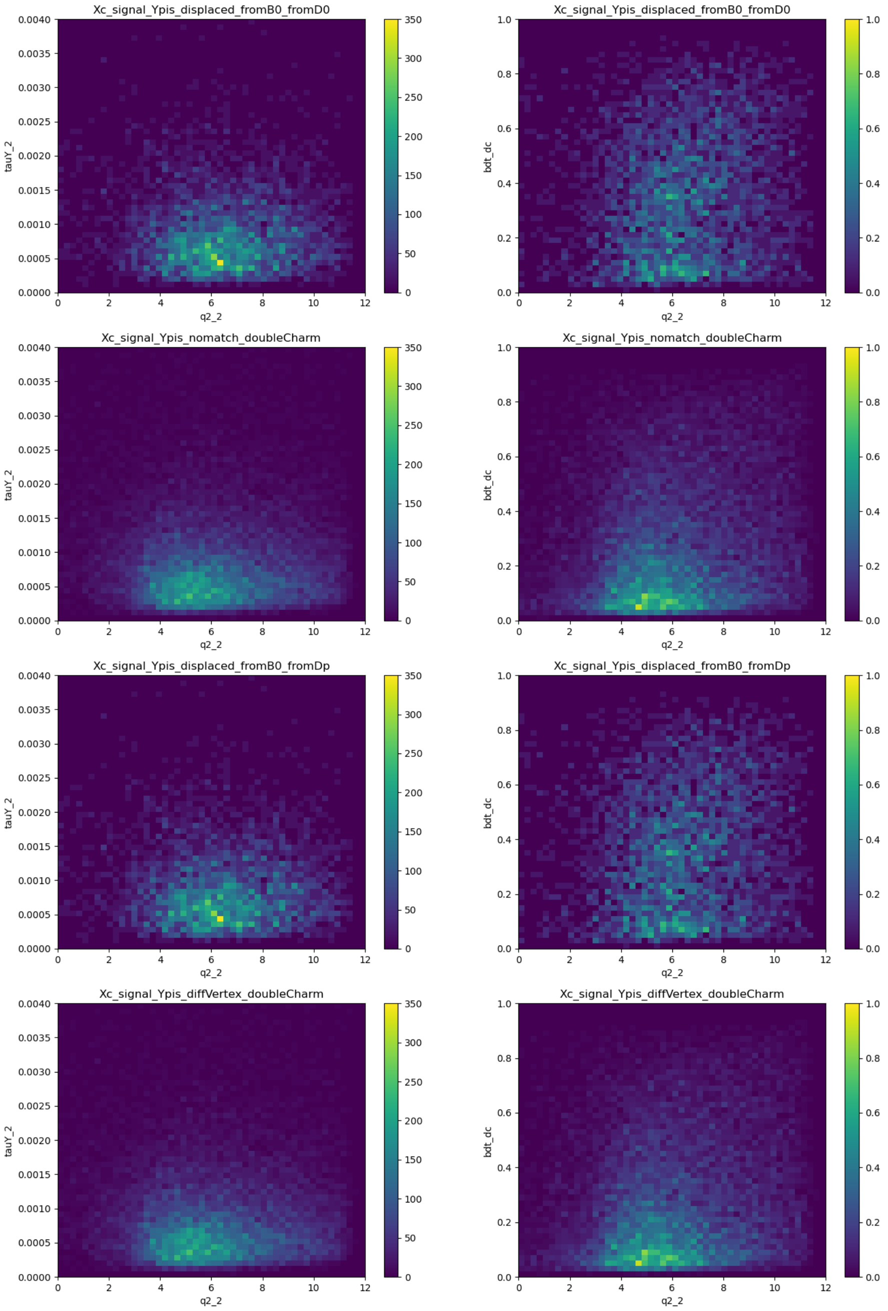


Figure E.14: 2D scatter plot of the variables used in the fit for categories Xc_signal_Ypis_displaced_fromB0_fromD0, Xc_signal_Ypis_nomatch_doubleCharm, Xc_signal_Ypis_displaced_fromB0_fromDp, Xc_signal_Ypis_diffVertex_doubleCharm

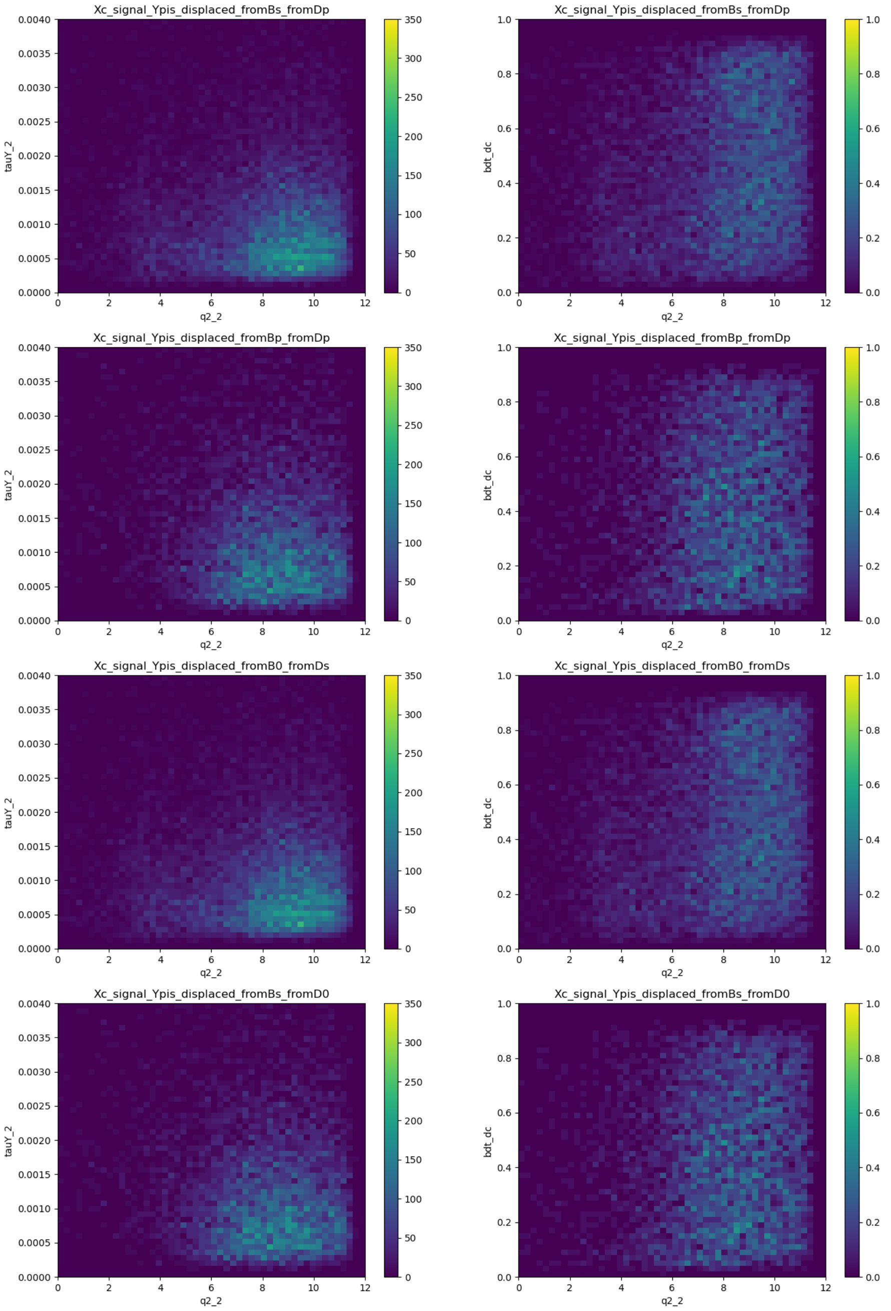


Figure E.15: 2D scatter plot of the variables used in the fit for categories Xc_signal_Ypis_displaced_fromBs_fromDp, Xc_signal_Ypis_displaced_fromBp_fromDp, Xc_signal_Ypis_displaced_fromB0_fromDs, Xc_signal_Ypis_displaced_fromBs_fromD0

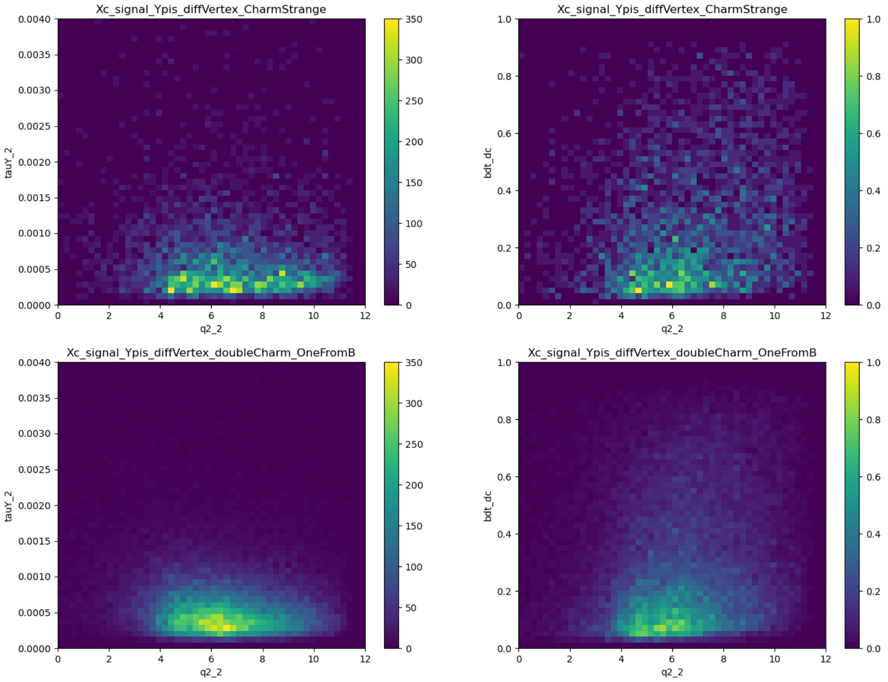


Figure E.16: 2D scatter plot of the variables used in the fit for categories Xc_signal_Ypis_diffVertex_CharmStrange, Xc_signal_Ypis_diffVertex_doubleCharm_OneFromB

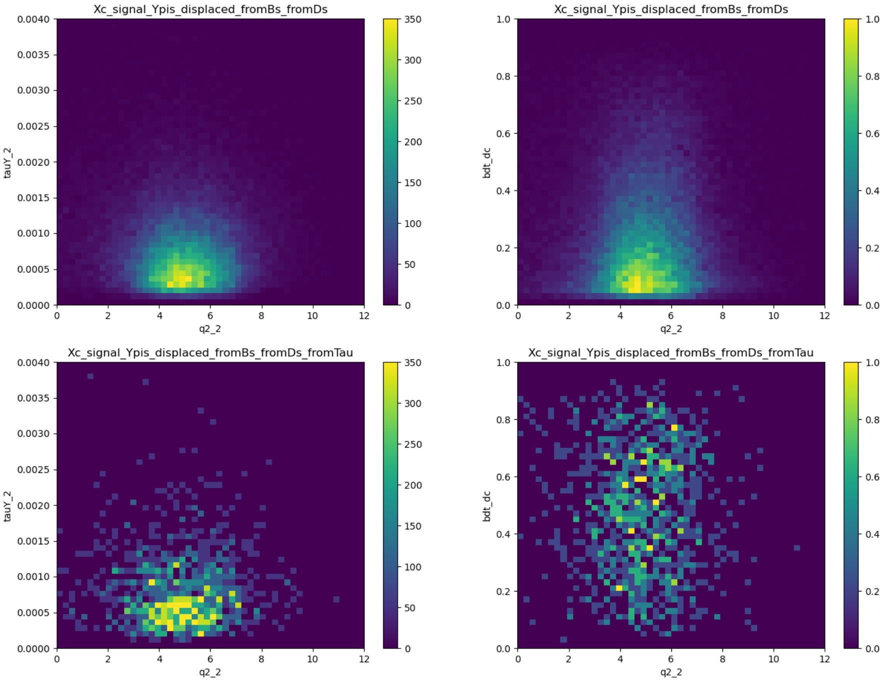


Figure E.17: 2D scatter plot of the variables used in the fit for categories Xc_signal_Ypis_displaced_fromBs_fromDs, Xc_signal_Ypis_displaced_fromBs_fromDs_fromTau

Università degli Studi di Ferrara

Sezioni

Dottorati di ricerca

Il tuo indirizzo e-mail

ctrbjm@unife.it

Oggetto:

Dichiarazione di conformità della tesi di Dottorato

Io sottoscritto Dott. (Cognome e Nome)

Couturier Benjamin

Nato a:

Tourcoing

Provincia:

France

Il giorno:

07/01/1974

Avendo frequentato il Dottorato di Ricerca in:

Fisica

Ciclo di Dottorato

36

Titolo della tesi:

Lepton Flavor Universality and analysis frameworks

Titolo della tesi (traduzione):

Universalità leptonica e sistemi di analisi dati

Tutore: Prof. (Cognome e Nome)

Bozzi Concezio

Settore Scientifico Disciplinare (S.S.D.)

FIS/01

Parole chiave della tesi (max 10):

Modello standard Universalita Leptonica LHCb FAIR

Consapevole, dichiara

CONSAPEVOLE: (1) del fatto che in caso di dichiarazioni mendaci, oltre alle sanzioni previste dal codice penale e dalle Leggi speciali per l'ipotesi di falsità in atti ed uso di atti falsi, decade fin dall'inizio e senza necessità di alcuna formalità dai benefici conseguenti al provvedimento emanato sulla base di tali dichiarazioni; (2) dell'obbligo per l'Università di provvedere al deposito di legge delle tesi di dottorato al fine di assicurarne la conservazione e la consultabilità da parte di terzi; (3) della procedura adottata dall'Università di Ferrara ove si richiede che la tesi sia consegnata dal dottorando in 1 originale cartaceo e 1 in formato PDF/A caricata sulla procedura informatica Esse3 secondo le istruzioni pubblicate sul sito: http://www.unife.it/studenti/dottorato alla voce ESAME FINALE – disposizioni e modulistica; (4) del fatto che l'Università, sulla base dei dati forniti, archivierà e renderà consultabile in rete il testo completo della tesi di dottorato di cui alla presente dichiarazione attraverso la pubblicazione ad accesso aperto nell'Archivio Istituzionale dei Prodotti della Ricerca IRIS-UNIFE (www.iris.unife.it) oltre che attraverso i Cataloghi delle Biblioteche Nazionali Centrali di Roma e Firenze; DICHIARO SOTTO LA MIA RESPONSABILITA': (1) che la copia della tesi depositata presso l'Università di Ferrara in formato cartaceo è del tutto identica a quella caricata in formato PDF/A sulla procedura informatica Esse3, a quelle da inviare ai Commissari di esame finale e alla copia che produrrò in seduta d'esame finale. Di conseguenza va esclusa qualsiasi responsabilità dell'Ateneo stesso per quanto riguarda eventuali errori, imprecisioni o omissioni nei contenuti della tesi; (2) di prendere atto che la tesi in formato cartaceo è l'unica alla quale farà riferimento l'Università per rilasciare, a mia richiesta, la dichiarazione di conformità di eventuali copie; (3) che il contenuto e l'organizzazione della tesi è opera originale da me realizzata e non compromette in alcun modo i diritti di terzi, ivi compresi quelli relativi alla sicurezza dei dati personali; che pertanto l'Università è in ogni caso esente da responsabilità di qualsivoglia natura civile, amministrativa o penale e sarà da me tenuta indenne da qualsiasi richiesta o rivendicazione da parte di terzi; (4) che la tesi di dottorato non è il risultato di attività rientranti nella normativa sulla proprietà industriale, non è stata prodotta nell'ambito di progetti finanziati da soggetti pubblici o privati con vincoli alla divulgazione dei risultati, non è oggetto di eventuali registrazioni di tipo brevettale o di tutela. PER ACCETTAZIONE DI QUANTO SOPRA RIPORTATO

Firma del dottorando

Ferrara, li 4 marzo 2024 Firma del Dottorando ________________________________

Firma del Tutore

Visto: Il Tutore Si approva Firma del Tutore _______________________________________