Towards implementing and combining two simplified analyses of the Higgs boson decay to four leptons on the PUNCH4NFDI framework

Ferran Ortiz Terol

Bachelorarbeit in Physik
angefertigt im Physikalischen Institut

vorgelegt der
Mathematisch-Naturwissenschaftlichen Fakultät
der
Rheinischen Friedrich-Wilhelms-Universität
Bonn

August 2024

Ich versichere, dass ich diese Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt sowie die Zitate kenntlich gemacht habe.

Bonn, 16/08/2024

Datum Unterschrift

1. Gutachter:	Priv.-Doz Dr. Philip Bechtle
2. Gutachterin:	Prof. Dr. Klaus Desch

Acknowledgements

I would like to thank Priv.-Doz. Dr. Philip Bechtle for giving me the opportunity to write my bachelor thesis on this very interesting topic. I would also like to thank Murillo Rebuzzi Vellasco for all his supervision, patience and invaluable help throughout this thesis. I must also thank all other members of the group for their guidance and friendly working environment throughout the development of this work. Finally, I want to thank my family and friends for motivating me to keep going.

Contents

CHAPTER 1

Introduction

High energy physics experiments such as CMS (Compact Muon Solenoid) and ATLAS (A Toroidal LHC ApparatuS) at CERN's Large Hadron Collider (LHC) have made some of the most important discoveries in particle physics, such as the identification of the Higgs boson [1, 2]. However, these experiments generate vast amounts of complex data that must be analysed and interpreted to advance our understanding of the universe. Increasingly, researchers in this and other data-driven fields are recognising the importance of not only collecting and analysing data, but also making it accessible and reusable by others.

In this context, the PUNCH4NFDI (Particles, Universe, Nuclei and Hadrons for the National Research Data Infrastructure) consortium plays a central role in building a robust research data infrastructure for particle physics, astroparticle physics and hadron physics. A key objective of PUNCH4NFDI is to create the appropriate infrastructure for the implementation of a specific set of digital products. These are called Data Research Products (DRPs) and they store all the information necessary to perform a specific workflow, such as code, datasets, tools and even metadata [3]. They offer many advantages in the context of data analysis: reproducibility, reusability, heterogeneity... But one capability that is particularly important in the context of this thesis is the ability to interact between them. This capability is of great value as researchers may wish to use the results of one DRP to perform the execution of another, creating complex chains of analysis that span multiple experiments and data sources.

The primary objective of this thesis is to demonstrate how two DRP prototypes containing the information to perform two separate particle physics data analyses - one with data from CMS and one with data from ATLAS - can be integrated within the PUNCH4NFDI framework. The motivation for this objective is to work towards the creation of a framework capable of combining analyses/DRPs. As such, the combination is expected to be simple and illustrative rather than thorough and scientifically rigorous. In addition to demonstrating the integration of the two DRP prototypes, this thesis explores the process of building the DRP prototypes themselves.

This thesis is structured as follows: Chapter 2 provides an overview of the relevant theoretical and experimental background, while Chapter 3 contextualises and introduces the ATLAS and CMS Open data analyses used during the thesis. The PUNCH4NFDI consortium is introduced in Chapter 4, along with the associated services and resources relevant to this thesis. Chapter 5 describes the implementation of the open data analyses as naive DRPs and discusses the steps forward in their integration, paying special attention to any setbacks encountered in this regard. Finally, the conclusion and outlook for future studies are discussed in Chapter 6.

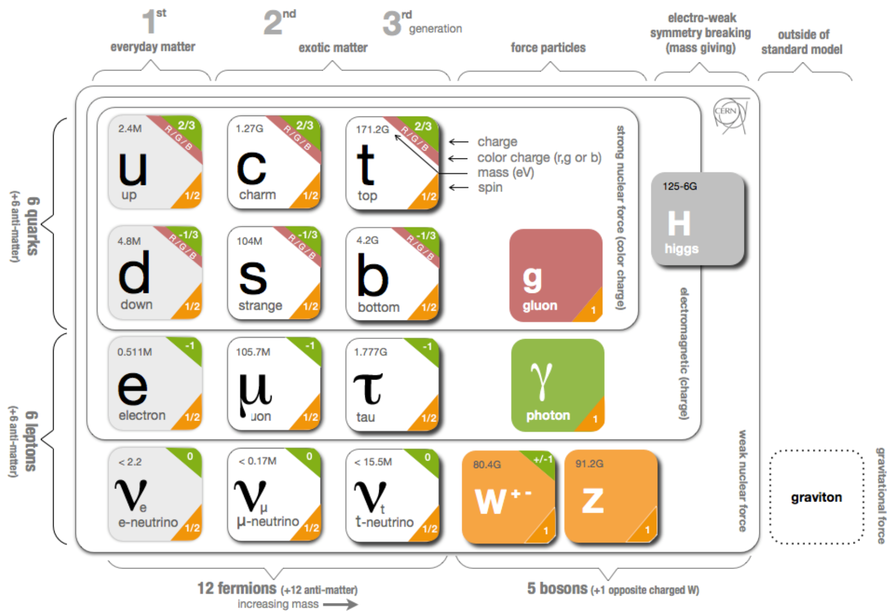
CHAPTER 2

Theoretical and experimental background

2.1 The Standard Model of particle physics

The Standard Model of particle physics (SM) describes all the known building blocks of our Universe, the elementary particles, and almost all the interactions between them. The following overview of the SM is based on reference [4], unless otherwise stated.

An illustration of all known elementary particles can be found in Fig. 2.1, and, as it is already shown there, they can be separated into two main groups: fermions and bosons.



1st everyday matter
2nd exotic matter
3rd generation
force particles
electro-weak symmetry breaking (mass giving)
outside of standard model
2.4M 2/3 R/G/B u up 1/2
1.27G 2/3 R/G/B c charm 1/2
171.2G 2/3 R/G/B t top 1/2
charge
color charge (r,g or b)
mass (eV)
spin
4.8M -1/3 R/G/B d down 1/2
104M -1/3 R/G/B s strange 1/2
4.2G -1/3 R/G/B b bottom 1/2
g gluon 1
125-6G H higgs
0.511M -1 e electron 1/2
105.7M -1 μ μon 1/2
1.777G -1 τ tau 1/2
γ photon 1
< 2.2 0 ν_e e-neutrino 1/2
< 0.17M 0 ν_μ μ-neutrino 1/2
< 15.5M 0 ν_τ τ-neutrino 1/2
80.4G +/-1 W+- 1
91.2G Z 1
graviton
strong nuclear force (color charge)
electromagnetic (charge)
weak nuclear force
gravitational force
6 quarks (+6 anti-matter)
6 leptons (+6 anti-matter)
12 fermions (+12 anti-matter)
increasing mass
5 bosons (+1 opposite charged W)
CERN

Figure 2.1: Overview of elementary particles in the SM [5].

Fermions are the basic components of matter, they all have spin $^{1}/_{2}$ and can also be separated into two groups, leptons (ℓ) and quarks (q). The "lepton" group consists of particles like the electron (e^-) with a negative electric charge ($Q = -1$) and the nearly massless electron neutrino (ν_e) which is electrically neutral ($Q = 0$). The quarks are particles with fractional electric charge like the up-quark (u), with $Q = +2/3$, and the down-quark (d), with $Q = -1/3$. They also carry the Quantum chromodynamics (QCD) equivalent to electric charge, the colour charge. Both of these groups can be further separated into generations: the first generation, composed by the four particles already mentioned, and the second and third generations, also composed by four particles each that are almost exact copies of the first generation ones, only differing in their masses. Finally, for each of these twelve fermions there exists an antiparticle with exactly the same mass but the opposite electric charge and for quarks also the opposite colour charge (anticolours).

The interactions between particles, the forces, are described by the exchange of spin-1 particles called gauge bosons. The photon (γ) intermediates the interactions of Quantum Electrodynamics (QED), while the gluon (g) is the mediator of Quantum Chromodynamics (QCD), and the Z and $W^\pm$ bosons are responsible for the weak neutral- and charged-current interactions, respectively. Finally, the SM contains the Higgs boson (H), which is the only elementary spin-0 particle discovered to date and its existence is a consequence of the Higgs mechanism. In the following section, this mechanism is described in more detail.

2.2 The Higgs mechanism

At first, one would expect all SM gauge bosons to be massless, since corresponding mass terms in the Lagrangian would break the required gauge invariance. This is not a problem in QED and QCD, where the corresponding gauge bosons are massless, but it is in apparent contradiction with the observed large masses of the Z and $W^\pm$ bosons.

In fact, fermions should not have mass neither. This can be shown by writing a fermion mass term in the Lagrangian in terms of the chiral particle states as

$$-m\bar{\Psi}\Psi = -m(\bar{\Psi}_L\Psi_R + \bar{\Psi}_R\Psi_L).$$

Under the $\mathrm{SU}(2)_L$ gauge transformation of the weak interaction, left-handed states transform differently than right-handed states, such that the mass term defined above would also break the required gauge invariance.

As a solution to this puzzle, Higgs [6], Englert and Brout [7] proposed the model now known as the Higgs Mechanism. This model starts by considering all the gauge bosons massles to conserve the gauge symmetry of the Lagrangian and then introducing a two complex scalar fields in weak isospin doublet, $\phi = \begin{pmatrix}\phi^+ \\ \phi^0\end{pmatrix}$ with the potential

$$V(\phi) = \mu^2\phi^\dagger\phi + \lambda(\phi^\dagger\phi)^2.$$

One must then analyse the vacuum state, which is the lowest energy state of the field ϕ and corresponds to the minimum of the potential given before. For the case where the potential has $\mu^2 > 0$ the minimum appears at $\phi = \vec{0}$ and the vacuum state has an unique value, like shown in Fig. 2.2 (a). On the other hand, for $\mu^2 < 0$, the minimum potential does not occur at $\phi = \vec{0}$ and the field is said to have a non-zero vacuum expectation value. This non-zero expectation value together with the symmetry of the potential make the vacuum state to be degenerated, as seen in Fig. 2.2 (b). However, the choice of a specific

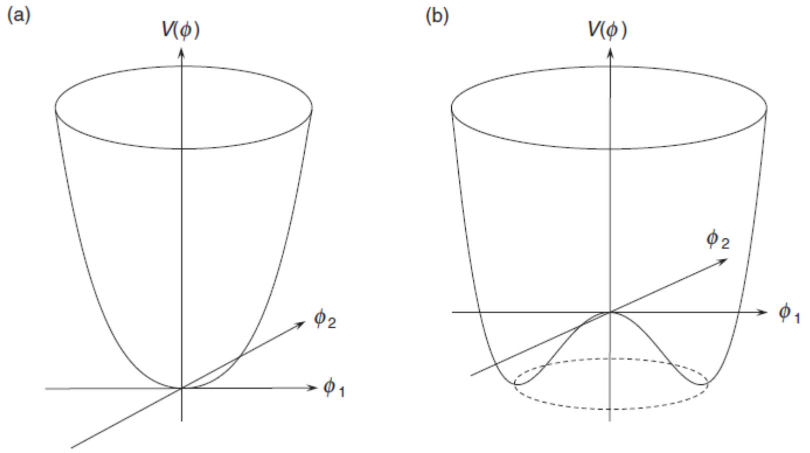


Figure 2.2: The Higgs potential $V(\phi) = \mu^2\phi^2 + \lambda\phi^4$ for single complex scalar field and for (a) $\mu^2 > 0$ and (b) $\mu^2 < 0$ [4]. The representation for the potential of a single complex scalar was used instead of the complex doublet of the SM, since we cannot visualize the SM one because it has four real variables.

value for the vacuum state breaks the symmetry of the Lagrangian, a process known as spontaneous symmetry breaking. Through the spontaneous symmetry breaking of the Lagrangian and then expanding the scalar field about the chosen vacuum state it is possible to hide the underlying gauge symmetry of the Lagrangian without removing it. This leads to the emergence of mass terms in the Lagrangian corresponding to the gauge bosons Z and $W^{\pm}$, without having to break the gauge symmetry. For the fermions, their mass terms will be generated through a spontaneous symmetry breaking of a Yukawa coupling of the same scalar field [4].

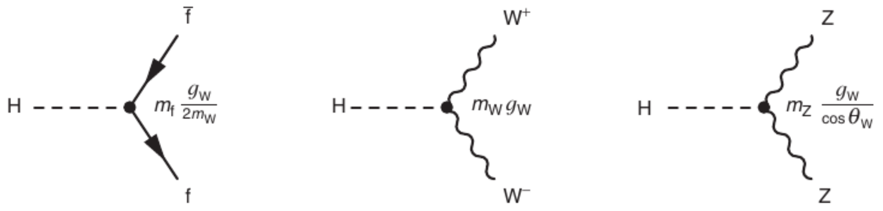

$\bar{\mathrm{f}}$
H
$m_{\mathrm{f}} \frac{g_{\mathrm{W}}}{2m_{\mathrm{W}}}$
f
W^+
H
$m_{\mathrm{W}} g_{\mathrm{W}}$
W^-
Z
H
$m_{\mathrm{Z}} \frac{g_{\mathrm{W}}}{\cos\theta_{\mathrm{W}}}$
Z

Figure 2.3: Feynman diagrams and coupling strengths for the lowest-order decay modes of the Higgs [4].

Other terms will also appear in the Lagrangian corresponding to the mass and interaction of a new scalar boson: the Higgs boson. The Higgs boson couples to all fermions and to the Z and $W^{\pm}$ bosons, as shown with the Feymann diagrams in Fig. 2.3. This figure also depicts how the coupling strength with the Higgs is proportional to the mass of the particle coupled, so the Higgs boson will couple preferentially to the most massive particles that are kinematically accessible [4].

2.3 The Large Hadron Collider

The Large Hadron Collider (LHC) located at CERN, in Geneva, is today's largest and most powerful particle accelerator in the world. It consists of a 27 km ring of superconducting magnets, where two beams of protons are accelerated close to the speed of light by a number of accelerating structures. The high-energy particle beams are then made to collide at four specific locations around the accelerator ring, corresponding to the positions of the four particle detectors of the LHC: the ATLAS, the Compact Muon Solenoid (CMS), A Large Ion Collider Experiment (ALICE) and the Large Hadron Collider beauty (LHCb) [8]. When the beams collide at these locations, the interactions between the high-energy particles create new particles that fly out in all directions from the collision point. The detectors then try to detect and measure the properties of the particles produced, with the aim of reconstructing the primary particles produced in the interaction. An overview of the CERN accelerator complex is shown in Fig. 2.4.

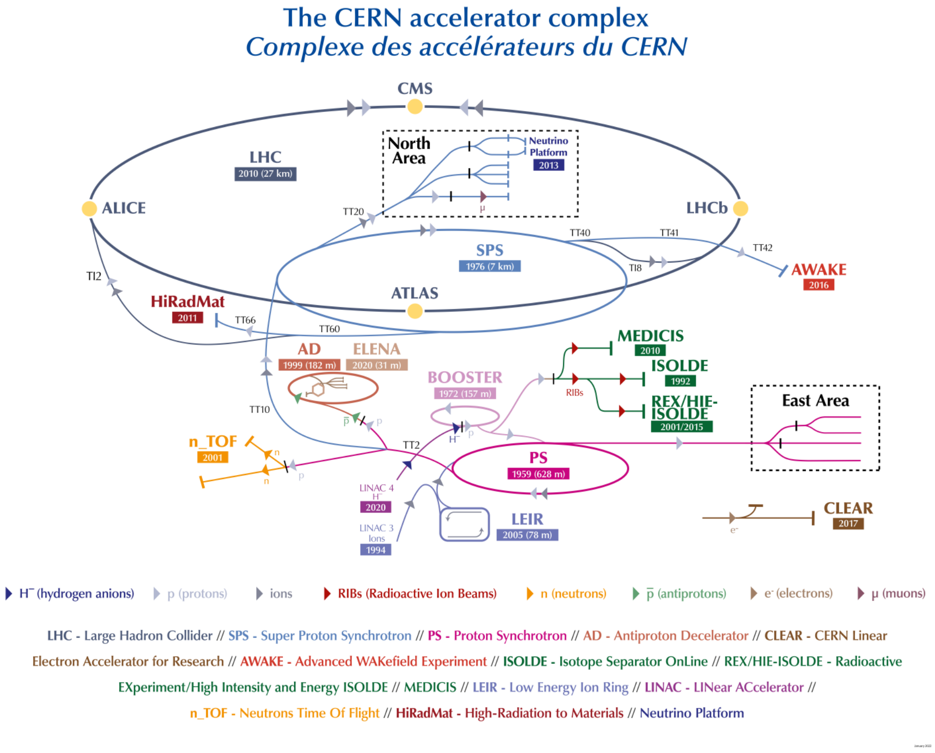

The CERN accelerator complex
Complexe des accélérateurs du CERN
CMS
LHC
2010 (27 km)
North Area
Neutrino Platform
2013
μ
ALICE
LHCb
TT20
TT40
TT41
TT42
SPS
1976 (7 km)
TI8
AWAKE
2016
TI2
HiRadMat
2011
TT66
ATLAS
TT60
AD
1999 (182 m)
ELENA
2020 (31 m)
MEDICIS
2010
ISOLDE
1992
BOOSTER
1972 (157 m)
RIBs
REX/HIE-ISOLDE
2001/2015
East Area
TT10
p̄
p
H⁻
n_TOF
2001
n
TT2
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 2.4: The acceleration complex and experiments at CERN [9].

2.4 The ATLAS and the CMS experiments

ATLAS and CMS are the two general-purpose detectors at the LHC. Both of them independently discovered the Higgs boson in 2012 [1, 2]. Since then, they have been investigating a wide range of physics processes, from the properties of the Higgs boson to topics on physics Beyond the Standard

Model (BSM) like extra dimensions and particles that could make up dark matter [10, 11]. Illustrations of both detectors can be found in Figs. 2.5 and 2.6.

2.4.1 Coordinate system

The coordinate system defined for both detectors takes the nominal interaction point as the origin. The z-axis points towards the beam direction and the x- and y-axis define the transverse plane to the beam direction, with the x-axis pointing to the center of the LHC ring and the y-axis pointing upwards. In cylindrical coordinates, the azimuthal angle ϕ is measured from the x-axis around the beam axis and the polar angle θ is measured from the beam axis. In hadron-hadron collisions, like the ones happening at the LHC, the event kinematics need to be described by three variables, and these variables can be related to experimentally well-measured quantities [4]. Two quantities that are usually used are the transverse momentum

$$p_{\mathrm{T}} = \sqrt{p_x^2 + p_y^2},$$

defined in the x-y plane, and the azimuthal angle ϕ. The third one is the pseudorapidity η, defined as

$$\eta = -\ln\left(\tan\frac{\theta}{2}\right).$$

Differences in η are invariant under boosts along the beam direction, which is also true for ϕ. It is therefore convenient to use the distance in the pseudorapidity-azimuthal angle space

$$\Delta R = \sqrt{\Delta\eta^2 + \Delta\phi^2},$$

as a distance measure between particles in high energy collider physics [12]. The Lorentz invariance of this quantity is useful for hadron-hadron collider physics, where the centre-of-mass frame of the hadron-hadron collisions is not the same as the centre-of mass of the colliding partons. As a result, the final states of the collisions are usually boosted along the beam axis, and so it becomes advantageous to use parameters that are not affected by this unknown longitudinal boost of the parton-parton system [4].

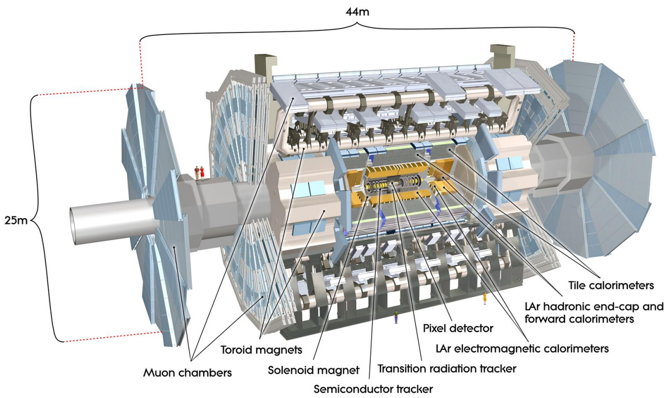

44m
25m
Tile calorimeters
LAr hadronic end-cap and forward calorimeters
Pixel detector
LAr electromagnetic calorimeters
Toroid magnets
Muon chambers
Solenoid magnet
Transition radiation tracker
Semiconductor tracker

Figure 2.5: Computer generated image of the ATLAS detector [13] showing its layout

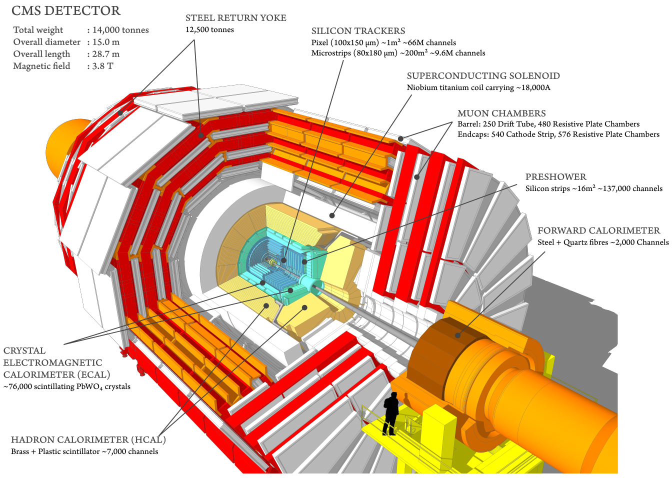

CMS DETECTOR
Total weight : 14,000 tonnes
Overall diameter : 15.0 m
Overall length : 28.7 m
Magnetic field : 3.8 T
STEEL RETURN YOKE
12,500 tonnes
SILICON TRACKERS
Pixel (100x150 μm) ~1m² ~66M channels
Microstrips (80x180 μm) ~200m² ~9.6M channels
SUPERCONDUCTING SOLENOID
Niobium titanium coil carrying ~18,000A
MUON CHAMBERS
Barrel: 250 Drift Tube, 480 Resistive Plate Chambers
Endcaps: 540 Cathode Strip, 576 Resistive Plate Chambers
PRESHOWER
Silicon strips ~16m² ~137,000 channels
FORWARD CALORIMETER
Steel + Quartz fibres ~2,000 Channels
CRYSTAL ELECTROMAGNETIC CALORIMETER (ECAL)
~76,000 scintillating PbWO$_4$ crystals
HADRON CALORIMETER (HCAL)
Brass + Plastic scintillator ~7,000 channels

Figure 2.6: Computer generated image of the CMS detector [14] showing its layout.

2.4.2 Components of the detectors

Both detectors consist of several layers of components wrapped concentrically around the collision point to measure the properties of the produced particles. The disposition and characteristics of each detector is different and presents its own unique characteristics. However, the general structure and the motivation behind each component is mostly the same. In the innermost part of the detectors, just near the beam line, there is a tracking system. This section is composed of several layers of different sensors that track the positions of the particles as they pass through. These positions are then used to reconstruct the path of the particles and measure their charge and momentum. This last step is possible due to the fact that the inner detector is immersed in a magnetic field parallel to the beam line. This field is generated by a magnet system on the detectors and its objective is to bend the path of charged particles emerging from the collisions. The more momentum the particles have, the less their path will bend, and depending on their charge they will bend in one direction or the other. This allows one to calculate the properties of the charged particles by tracking their bent paths [14, 15].

Just outside of the tracking system are calorimeters. They are designed to absorb and measure the energy of most of the particles produced in the collisions. There are two main types of calorimeters: The Electomagnetic Calorimeters (ECAL), which measure the energy of electrons and photons, and the Hadron Calorimeters (HCAL), which are usually located outside the ECAL and measure the energies of hadrons. Photons and electrons are mostly stopped at the ECAL and Hadrons at the HCAL, with charged hadrons also leaving small energy deposits in the ECAL [14, 15].

Muons on the other hand are the only particles (beside the neutrinos) that are not stopped by the calorimeters, although they still leave small energy deposits [4]. To be able to correctly measure them, a special set of muon detectors is located in the outer part of the experiment, where they are the only particles that may produce a signal [14, 15].

CHAPTER 3

CERN Open Data analyses

While this thesis does not set out to perform an in-depth analysis of the $H \to ZZ^{(*)} \to 4\ell$ decay mode, it does seek to implement a simplified analysis within the context of the PUNCH4NFDI. As such, the analysis itself represents a fundamental component of the thesis, and will therefore be discussed in greater detail in the following sections.

To provide further detail, the entirety of the work presented in this thesis is based on two illustrative analysis codes [16, 17]. The two codes are simplified representations of an analysis that reconstructs the Higgs boson decaying to two Z bosons from events with four final-state leptons. One code is based on the original ATLAS Higgs to four leptons analysis [1], while the other is based on the original analysis published by CMS [2]. An overview of both publications and their results is provided in Section 3.1, while a more detailed examination of the $H \to ZZ^{(*)} \to 4\ell$ analysis is presented in Section 3.2. All data and simulated events are sourced from CERN Open Data, representing a subset of data recorded in 2012 with each respective detector. The CERN Open Data Policy is introduced in brief in Section 3.3. Finally, the analysis codes themselves and their characteristics are studied in detail in Sections 3.4 and 3.5, including the coding language used, the cuts applied, the characteristics of the data used, the outputs, and so forth.

3.1 Original observation of the Standard Model Higgs Boson

As previously stated in the preceding chapter, in 2012, both the ATLAS and the CMS Collaborations presented the findings of their searches for the SM Higgs boson in proton-proton collisions with the respective detectors at the LHC. The datasets used at ATLAS (CMS) corresponded to integrated luminosities of approximately 4.8 fb^{-1} (5.1 fb^{-1}) collected at $\sqrt{s} = 7$ TeV in 2011 and 5.8 fb^{-1} (5.3 fb^{-1}) at $\sqrt{s} = 8$ TeV in 2012. The primary decay modes employed in the searches were: (i) $H \to \gamma\gamma$, (ii) $H \to ZZ^{(*)} \to 4\ell$ (where $\ell = e$ or μ), (iii) $H \to WW^{(*)}$, (iv) $H \to \tau^+\tau^-$, and (v) $H \to b\bar{b}$. Following the application of the requisite cuts to the various channels, both the ATLAS and CMS experiments observed an excess of events above the anticipated background at a mass value of approximately 125 GeV, indicative of the production of a new particle [1, 2].

In the ATLAS experiment, the signal was observed with a significance of 5.9 standard deviations, and the particle mass was determined to be (126.0 ± 0.4 (stat.) ± 0.4 (sys.)) GeV. The observed significance corresponded to a background fluctuation probability of 1.7×10^{-9} and the results were consistent with the production and decay of the Standard Model Higgs boson [1].

For the CMS collaboration, the signal had a significance of 5.0 standard deviations and the particle mass was measured to be $(125.3 \pm 0.4\,(\text{stat.}) \pm 0.5\,(\text{sys.}))$ GeV. This observation was also compatible with the production and decay of the Standard Model Higgs boson, which had an expected significance of 5.8 standard deviations for this mass value [2].

Although all the channels mentioned above contributed to the discovery of the Higgs in both detectors, the signals were most significant in two specific decay modes, $H \to \gamma\gamma$ and $H \to ZZ^{(*)} \to 4\ell$, thanks to their high sensitivity in the mass region of the Higgs and their excellent mass resolution [1, 2]. For the purposes of this work, only the $H \to ZZ^{(*)} \to 4\ell$ analysis was considered.

3.2 $H \to ZZ^{(*)} \to 4\ell$

The search for Higgs boson candidates via the $H \to ZZ^{(*)} \to 4\ell$ decay is done by selecting two pairs of isolated leptons, each consisting of two leptons with the same flavour and opposite charge. All leptons are also required to have good kinematics (sufficiently large p_{T} and sufficiently small η values) and to come from the same primary vertex. Once the leptons are selected, both pairs are required to have a total invariant mass within some specific range. This range is more restrictive for the leading lepton pair (the lepton pair with invariant mass closest to the Z boson mass) than for the subleading lepton pair (the other lepton pair). Many other cuts and selection rules are applied in the original ATLAS and CMS analysis, but are beyond the scope of this thesis. It is also worth noting that the values for all these thresholds were chosen with the intention of optimising the results of each analysis, and therefore they do not have the same values for the CMS analysis as for the ATLAS analysis. Moreover, the thresholds change even within the same analysis, depending on the particles or the subchannels being considered.

The objective of these cuts is to enhance the signal-to-background ratio. The largest background source is continuum $ZZ^{(*)}$ production. Additionally, non-negligible background contributions also arise from Z + jets (predominantly $Z + b\bar{b}$) and $\tau\bar{\tau}$ events. The aforementioned background sources are more readily amenable to reduction than those originating from $ZZ^{(*)}$ production, given that they are quite distinct from the signal.

The final plots, obtained after applying all the necessary cuts, for the analysis of the $H \to ZZ^{(*)} \to 4\ell$ channel in the original CMS and ATLAS papers are shown in Figs. 3.1 and 3.2. Both plots were made using a combination of $\sqrt{s} = 7$ TeV and $\sqrt{s} = 8$ TeV data.

3.3 CERN Open Data Policy

UNESCO defines open research data as "the publishing of the data underpinning scientific research results so that they have no restrictions on their access and usage. Openly sharing data opens it up to inspection and re-use, forms the basis for research verification and reproducibility, and opens up a path to broader collaboration" [18]. CERN has been committed to follow this and other open science principles since the CERN Convention over 60 years ago. This commitment was reaffirmed in the European Strategy for Particle Physics (2020) and formalised with the CERN Open Data Policy for the LHC Experiments. This programme was accepted by the collaboration boards of these experiments (ALICE, ATLAS, CMS and LHCb) with the objective of enabling and motivating them to open and preserve experimental data [19].

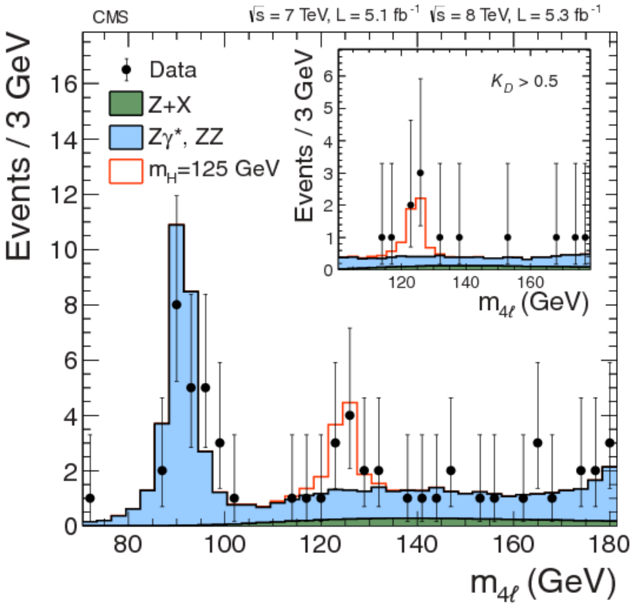


Figure 3.1: Distribution of the four-lepton invariant mass for the Higgs to four leptons analysis, as presented in the original CMS paper [2]. The different background expectations are represented as filled histograms and the data as points. The signal expectation appears as an open histogram and corresponds to a Higgs boson of mass $m_H = 125\,\text{GeV}$. The inset also depicts the four-lepton invariant mass distribution but after selection of events with $K_D > 0.5$, where K_D represents the probability ratio of the signal and background hypotheses, defined as $K_D = \mathcal{P}_{sig}/(\mathcal{P}_{sig} + \mathcal{P}_{bkg})$.

The open data is released through the CERN Open Data Portal [20] which will be supported by CERN for the lifetime of the data [19]. The data produced by the LHC experiments are typically categorised into four different levels identified by the Data Preservation and Long Term Analysis in High Energy Physics (DPHEP) Study Group [21]:

- Level 1 data provides more information on published results in publications, such as extra figures and tables.
- Level 2 data includes simplified data formats for outreach and analysis training, such as basic four-vector event-level data.
- Level 3 data comprises reconstructed collision data and simulated data together with analysis-level experiment-specific software, allowing to perform complete full scientific analyses using existing reconstruction.
- Level 4 data covers basic raw data (if not yet covered as level 3 data) with accompanying reconstruction and simulation software, allowing the production of new simulated signals or even re-reconstruction of collision and simulated data.

The CERN Open Data Portal is primarily concerned with the release of event data from levels 2 and 3. In addition, the LHC collaborations may also provide small samples of level 4 data [22]. The data is

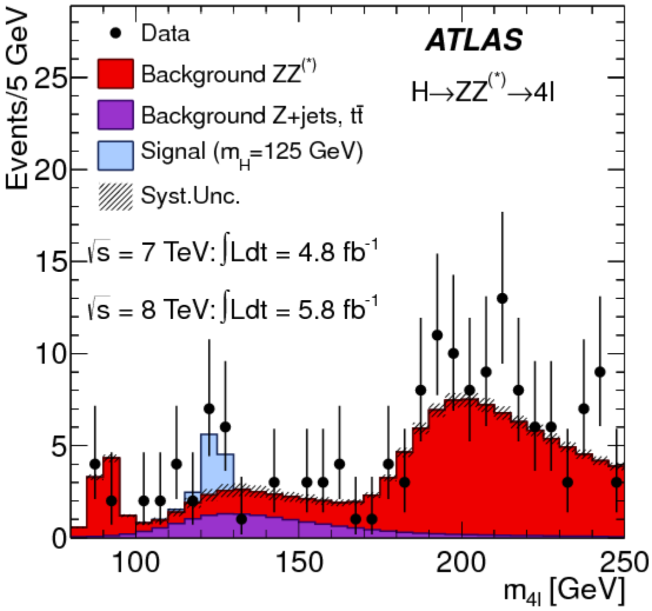


Figure 3.2: Distribution of the four-lepton invariant mass for the the Higgs to four leptons analysis, as presented in the original ATLAS paper [1]. It compares selected data candidates to the background expectation. Additionally, the signal expectation for a Standard Model Higgs with m_H = 125 GeV is also shown.

made available a certain number of years after its collection, allowing sufficient time for members of each experiment to perform the requisite analyses. The ultimate goal is to have the complete datasets accessible by the end of the collaboration. [19] .

It is also important to present the ATLAS Open Data project in this thesis. This is a project run by the ATLAS Collaboration in cooperation with the CERN Open Data team [23]. It has two main goals: firstly, to provide data and tools to high school, undergraduate and graduate students, as well as teachers and lecturers, in order to facilitate their education and enable them to practise physics analysis techniques used in experimental particle physics; and secondly, to provide researchers with high-quality data recorded by the ATLAS detector, thus enabling them to conduct state-of-the-art analyses in particle physics [24]. As mentioned above, the project is working in collaboration with the CERN Open Data team. The intention of the project is to replicate all ATLAS open data resources on both sites [24].

3.4 CMS reduced analysis

The CMS analysis used for this thesis was obtained from the GitLab repository [16] as one of the tutorials available to learn how to use the PUNCH REANA infrastructure, which will be introduced in the next chapter. It consists of a C++ code called `df103_NanoAODHiggsAnalysis.C`, which uses ROOT, an open-source data analysis framework very extended in high energy physics [25], to perform the analysis. The code is an adaptation of the ROOT tutorial [26], which itself follows a simplified re-implementation of parts of the original CMS Higgs to four lepton analysis [2], published on the CERN Open Data Portal [27].

As mentioned in the introduction of the chapter, the data and simulated events are taken from the

CERN Open Data Portal. However, only 50% of the available LHC Run I samples are used for this analysis, as this is the fraction of data that is public according to the CMS Open Data policy [28]. This reduces the statistical significance compared to what can be achieved with the full dataset. Nevertheless, the results from the original paper [2] were also obtained using only a portion of the Run I statistics roughly equivalent to the luminosity of the public sets, but with only partial statistical overlap [27].

For each sample of data and simulated events, the following steps are performed to reconstruct the Higgs boson from the selected muons and electrons:

1. Selection of interesting events using multiple cuts on event properties, e.g., number of leptons, kinematics of the leptons and quality of the tracks.

2. Perform reconstruction of two Z bosons, of which only one on-shell, from the selected events and apply cuts on them.

3. Finally, take the remaining Z boson candidates and reconstruct the Higgs boson. Calculate its invariant mass.

The results of the reconstruction are then employed to generate plots illustrating the invariant mass of the selected four lepton systems in the various decay modes (four muons, four electrons and two of each kind) and in a combined plot indicating the decay of the Higgs boson with a mass of about 125 GeV. For the purposes of this thesis, only the latter plot, shown in Fig. 3.3, is of consequence. The plots are saved as ROOT files; however, the tutorial from which the code was obtained [16] also includes a code file called "PrintHistos.C" that can be used to convert the histograms into PDF files.

The analysis may be conducted independently by downloading the requisite code from the GitLab repository [16] and then, on the same directory as the code file, using the command

```
mkdir -p results && root -b -x -q df103_NanoAODHiggsAnalysis.C,
```

in a terminal environment with ROOT available. To convert the results to PDF, it is sufficient to download "PrintHistos.C" file, move it to the same directory as the analysis code and then use the command

```
root -b -x -q 'PrintHistos.C("results/filename")',
```

with filename being the name of the ROOT file containing a histogram that is to be converted.

3.5 ATLAS reduced analysis

The ATLAS analysis code used is part of a set of educational Jupyter notebooks provided by ATLAS Open Data [29]. The code can be accessed from the referenced website by selecting the 'Higgs to ZZ' notebook, which directs the user to the GitHub repository containing it [17].

The analysis code is loosely based on the original ATLAS Higgs to four leptons analysis [1], and it employs data from the 13 TeV proton–proton collisions collected by the ATLAS detector and provided by the ATLAS Open Data project. It should be noted, however, that due to the educational purpose of the analysis, the data used for the analysis has already undergone a series of pre-processing steps to facilitate the interpretation of the analysis code. Consequently, the only cuts explicitly applied in the code are to select two pairs of isolated leptons, each of which is comprised of two leptons with the same flavour and opposite charge, on each sample, since all the other necessary cuts have already been applied to the

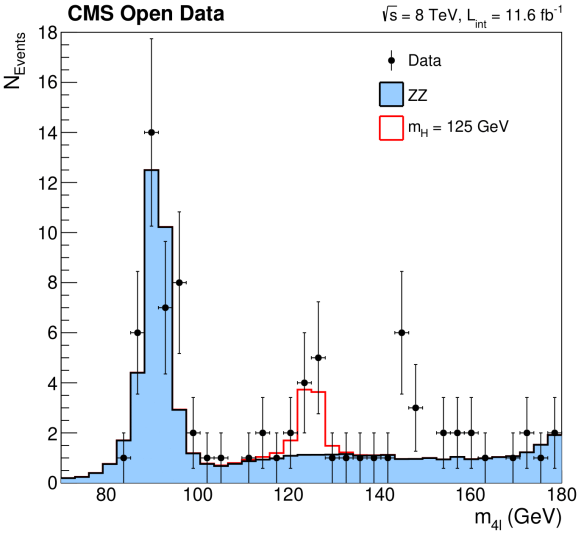


Figure 3.3: Distribution of the four-lepton invariant mass for the Higgs to four leptons analysis obtained from the simplified CMS analysis used for this thesis [16]. It can be observed that this plot is less complex version of the plot on Fig. 3.1 but with different values for the histograms and the data points.

data sets. Additionally, since there are no cuts to be applied to the reconstructed Z bosons, there is no requirement to reconstruct them.

As in the CMS analysis code, the samples to be processed are located within ROOT files. However, in this instance, the analysis code utilises the Python programming language in place of C++. To that end, the Uproot library is employed [30], which facilitates the reading and writing of ROOT files in pure Python and NumPy.

Given the simplicity of the cuts that can be applied, there is no need to separate the samples into the different decay modes (four muons, four electrons and two of each kind). Consequently, a single plot is generated, as illustrated in Fig. 3.4, which depicts the four-lepton invariant mass of all the selected samples. The code displays the plot in the notebook but does not save it in any format; this is addressed in subsequent implementations.

It is important to note that the code in question cannot be executed without the inclusion of another supplementary code file, `infofile.py`, which can be located in the same repository as the ATLAS code [17]. This is a Python file comprising a dictionary containing all the dataset ID numbers (DSID), the number of events, the reduction efficiencies, the sums of weights and the cross-section for each Monte Carlo (MC) simulation. It is necessary for this file to be located in the same repository as the analysis code in order for the analysis to be executed correctly.

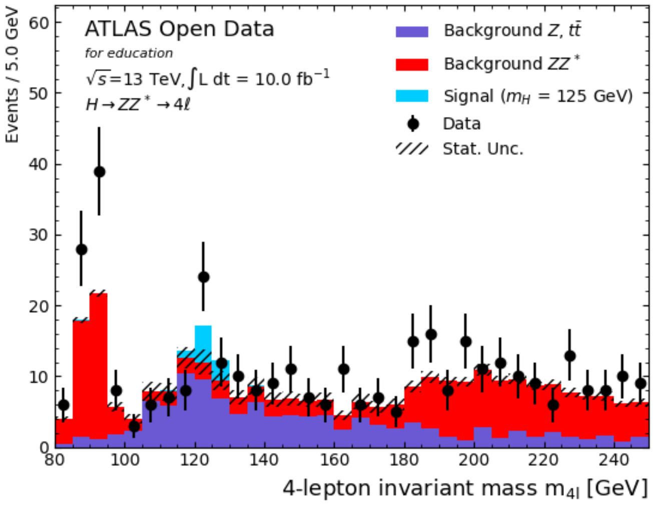


Figure 3.4: Distribution of the four-lepton invariant mass for the Higgs to four leptons analysis obtained from the simplified ATLAS analysis used for this thesis [16]. It displays similar features as the plot on the original ATLAS paper. depicted in Fig. 3.1. However, it uses $\sqrt{s}$ = 13 TeV data instead of a combination of $\sqrt{s}$ = 7 TeV and $\sqrt{s}$ = 8 TeV data.

CHAPTER 4

The PUNCH4NFDI consortium

4.1 PUNCH4NFDI and the PUNCH Science Data Platform

The Particles, Universe, NuClei and Hadrons (PUNCH) for the NFDI (PUNCH4NFDI) consortium is a collective of researchers specialising in particle, astroparticle, astro-, hadron and nuclear physics, established by the National Research Data Infrastructure (German: Nationale Forschungsdateninfrastruktur – NFDI). It represents approximately 9,000 scientists holding doctoral qualifications from universities and research institutes, including the Max Planck Society, the Leibniz Association, and the Helmholtz Association in Germany. The scientific fields that form the consortium engage in data-intensive research at large facilities, and thus they typically confront analogous data analysis challenges. The consortium's primary objective is to facilitate collaboration between these communities and to derive benefits from the sharing of experiences, tools, solutions, and other resources [31].

PUNCH4NFDI aims to establish a science data platform that adheres to the FAIR principles, which stipulate that the data must be

1. **F**indable: It should be easy to find for both humans and computers.
2. **A**ccessible: The user should be informed on how to access the data.
3. **I**nteroperable: One should be able integrate the data with other data. Also it should be compatible with the necessary applications and workflows.
4. **R**eusable: Data and metadata should be described in a way that is possible to reuse it with different settings [32].

The PUNCH Science Data Platform (PUNCH-SDP) will provide the necessary infrastructure and interfaces to access and utilise the diverse resources contributed by participating communities in the project, as outlined in the PUNCH website. These resources will be represented as services offered by the platform, and can be grouped into five classes of services [33]:

- The PUNCH central infrastructure, which basically refers to the PUNCH-SDP itself.
- Data access and management services.
- Analysis software and data irreversibility services.

- Metadata services.
- Computing and storage resource services.

In order to provide the aforementioned services, it is essential to use diverse set of specialised environments, engines, systems and other resources which the consortium possesses, which will serve as the foundation for the scientific platform. The most pertinent ones for this thesis are presented in the following sections.

4.2 Compute4PUNCH and Storage4PUNCH

As previously stated, the PUNCH-SDP aims to facilitate access to the diverse range of resources provided by participating communities across Germany. These resources include a broad and heterogeneous assortment of storage and computing systems, which can be utilized to address the varying needs of the involved communities. To ensure a seamless, unified, and federated access to these resources, the Compute4PUNCH and Storage4PUNCH concepts are being developed [31].

4.2.1 Federated Heterogeneous Compute Infrastructure – Compute4PUNCH

The concept of Compute4PUNCH (C4P) refers to a nationwide federated heterogeneous compute infrastructure that is currently under development with the objective of providing seamless access to the considerable amount of compute resources that are being made available by the PUNCH4NFDI institutions, which include High-Throughput Compute (HTC), High-Performance Compute (HPC) and Cloud resources [31]. A fundamental prerequisite for the establishment of this infrastructure is the dynamic integration of all the compute resources into a unified HTCondor [34]-based Overlay Batch System (OBS), which functions as a single pool of compute resources. Multiple entry points into this resource pool are provided by utilising the same Helmholtz authentication and authorisation infrastructure (AAI). Another fundamental element for Compute4PUNCH is the COBalD/TARDIS metascheduler, which is responsible for determining whether to increase or decrease the number of integrated resources of a certain kind based on the current demand for them [31].

The provision of operating systems and software environments to the PUNCH4NFDI communities is guaranteed through the utilisation of state-of-the-art container technology and the CERN Virtual Machine File System (CVMFS). Gitlab supplies the requisite tools for the hosting of container build instructions (Dockerfiles) and a registry of such containers. CVMFS facilitates a scalable distribution of the containers [31].

A diagram illustrating the proposed federated heterogeneous compute and storage PUNCH4NFDI infrastructure is presented in Fig. 4.1 .

4.2.2 Federated Storage Infrastructure – Storage4PUNCH

In order to use the available storage resources and to supplement the distributed computing infrastructure outlined above, a distributed storage infrastructure for the long-term storage of data products is also being developed. This infrastructure has been designated 'Storage4PUNCH (S4P). Tokens are also provided from the PUNCH AAI (hosted by the Helmholtz AAI [35]) to facilitate access to the storage components. Consequently, dedicated data access protocols are required. The WebDAV and XRootD protocols have

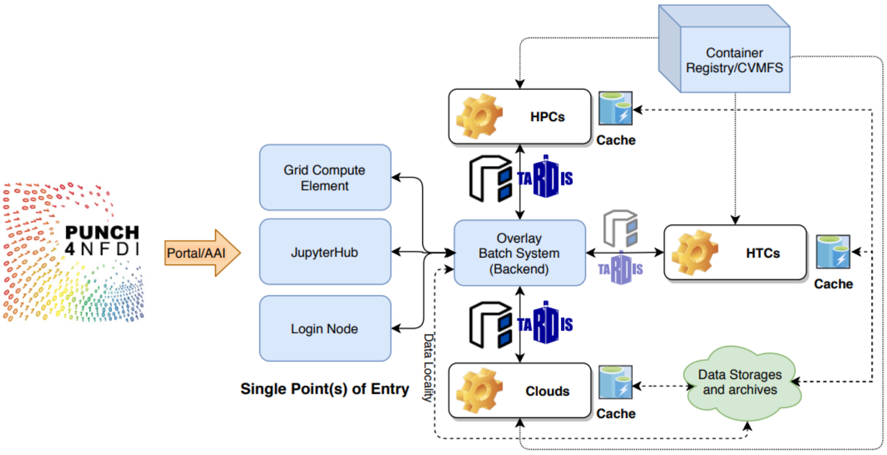


Figure 4.1: Architecture diagram of the intended federated heterogeneous compute and storage PUNCH4NFDI infrastructure [31]

been selected as the preferred options [31]. The current prototype installations for Storage4PUNCH utilise two types of instances: a dCache instance at DESY and XRootD instances at both the University of Bonn and GSI [31].

In the meantime, prospective applications of Storage4PUNCH are being contemplated. These include the development and incorporation of a universal metadata catalog, the evaluation of Rucio, a prevalent data management instrument in High Energy Physics (HEP), for smaller research collectives, and the possibility of federating the storage system [31].

4.3 Workflow engines and REANA

The integration of the previously outlined computing and storage facilities would provide the PUNCH-SDP with the fundamental infrastructure necessary to enable access to the majority of resources available within the PUNCH4NFDI consortium. Nevertheless, the PUNCH-SDP would not yet be complete, as its objective is not merely to provide access to a set of distributed resources; it also aims to offer an appropriate environment for their correct utilisation. In order to achieve this, it is necessary to implement a workflow engine.

A workflow engine is responsible for the management of all constituent steps within a workflow. To illustrate, should a user wish to undertake an analysis utilising data and computing resources from the PUNCH-SDP, a series of steps must be completed. These include the collection of data and its placement in a cache, the submission of a request for the necessary computing resources, and finally, the retrieval and storage of the results. Each individual step can be completed using the available storage or computing facilities. However, to perform the entire analysis, all the steps must be linked together into a single workflow. This is the responsibility of the workflow engine [36].

The option chosen for PUNCH-SDP is REANA, which is a "reusable and reproducible research data analysis platform. It helps researchers to structure their input data, analysis code, containerised environments and computational workflows so that the analysis can be instantiated and run on remote compute clouds" [37]. The platform was originally designed for particle physics analysis, but can be applied to any scientific field. One of its key features is the ability to re-use and re-interpret data analysis, even several years after it was published [37]. Another useful feature of REANA is that it works as a workflow engine, capable of using several workflow languages: Common Workflow Language (CWL), Serial, Yadage and Snakemake [37].

Prior to undertaking an analysis on REANA, it is first necessary to create a reana.yaml file. The file provides a comprehensive description of the analysis structure, including all the requisite elements for its implementation. The fundamental elements that must be defined for any given analysis are as follows:

- The source code with the actual analysis to be performed,
- The environment that is going to be used to run the analysis. Such environments are typically encapsulated within Docker container images.
- The actual workflow, including the computational steps necessary to execute the code.

Additionally, it is common to specify supplementary elements, such as output files containing the results or secondary input files (e.g., data sets or supplementary code). Furthermore, the choice of computing backend for the analysis can be specified. These features indicate a clear path to implementing the computing and storage resources of PUNCH4NFDI into the REANA workflows. These resources can be integrated into the workflows with additional specifications to the "reana.yaml" file corresponding to the analysis workflow. Several tutorials have been published [38], providing examples of different codes prepared to be run on REANA with their respective reana.yaml files. Fig. 4.2 illustrates the integration of REANA in PUNCH-SDP.

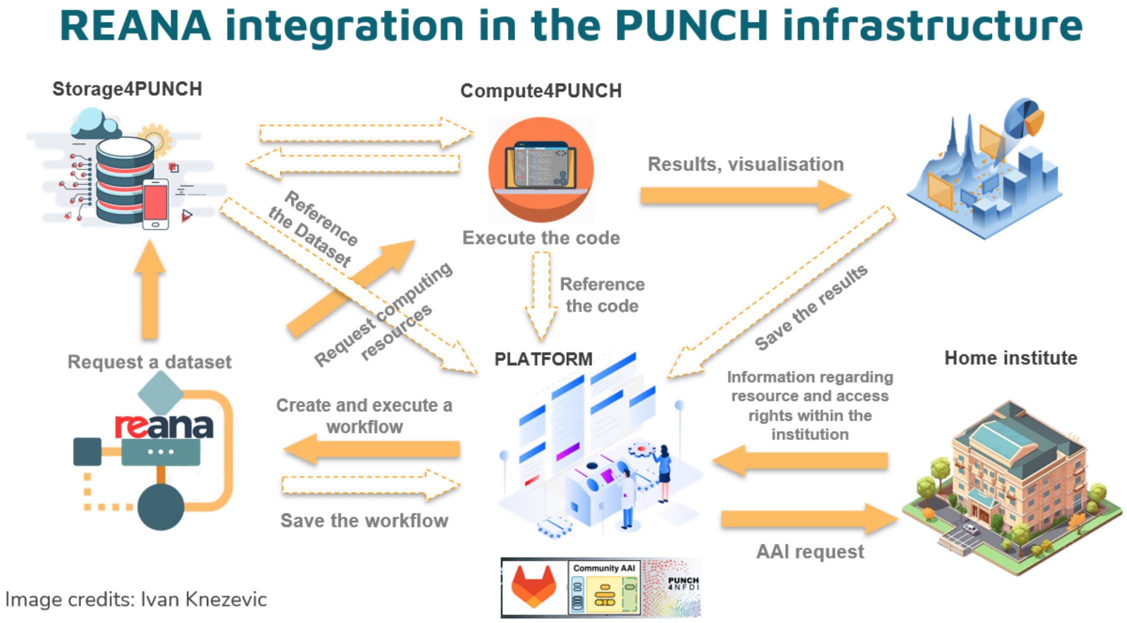


Figure 4.2: Diagram depicting the implementation of REANA on the working infrastructure of PUNCH-SDP. Created by Ivan Knezevic from PUNCH4NFDI

4.4 Digital research products

Before moving on to the next chapter, it is important to talk about one of the fundamental services that PUNCH-SDP plans to offer, and a very important concept in the context of this thesis: Digital Research Products (DRP). In simple terms, a DRP contains all the information necessary to execute a workflow (the "metadata description of a workflow" [3]). This includes datasets, simulations, tools, source code, metadata, publications and any other resources associated with the workflow. All this information is defined with the appropriate identifiers [3], packaged, and stored in a database [36]. The concept of DRP is strongly related to the idea of reproducibility and reusability of existing workflows [39]. DRPs are also presented as a way to store the intermediate status and results of scientific work during its realisation [36]. In addition, DRPs are expected to fulfil an interoperability function, as PUNCH4NFDI intends to provide interoperable DRPs, including combined analysis using data from different sources and experiments [39], which pretty much reflects the purpose of this thesis. The packing, unpacking and storage of these DRPs will be done on the PUNCH-SDP [3, 36]. A Digital Research Product Registry is being developed for this purpose and should be ready in the near future.

CHAPTER 5

Implementation and combination of the DRP prototypes

This chapter provides an overview of the work conducted in the context of this thesis. As previously stated in the introduction, the primary objective of this work was to demonstrate the integration of two distinct DRPs into a unified one within the existing PUNCH4NFDI framework. The initial two DRPs are structured around the Open Data analysis presented in Chapter 3. Consequently, this procedure represents the combination of two similar analyses, using data from different sources (ATLAS and CMS), into a single unified analysis. It was not the aim of this thesis, however, to perform a highly rigorous combined analysis of CMS and ATLAS open data. Rather, the objective was to develop the framework to enable the combination of analyses/DRPs and to provide an illustrative example based on interesting physics. There are several reasons for pursuing this objective. From one perspective, combining different analysis of the same physics process is an effective method for increasing statistics. For this case, it could result in a more pronounced and clear peak for the Higgs signal, which yields significant benefits. Conversely, the capacity to connect and operate multiple DRPs in parallel represents a valuable feature by itself, as it enables the use of the outputs from one DRP as inputs to another. Furthermore, this demonstration offers an nice way for assessing the capabilities of the existing PUNCH4NFDI framework and ascertaining the potential outcomes that can be attained with it.

The chapter is structured as follows: Section 5.1 describes the integration of the analysis codes on REANA. Then, Section 5.2 explains how the integrated REANA analyses were adapted to work using Compute4PUNCH and Storage4PUNCH.Finally, Section 5.3 presents steps towards a simplified combined analysis.

5.1 Implementation on REANA

As previously stated in Chapter 4, the prerequisites for implementing and running an analysis code on REANA are as follows: Primarily, the source code must be available, the environment must be encapsulated inside a Docker container image, and any additional resources required by the analysis must be accessible, such as input databases or a directory for storing results. Subsequently, all this elements are delineated within a YAML file, accompanied by the computational steps that constitute the workflow. Finally, the required steps for conducting an analysis on REANA are to be followed. This is, to set up the reana-client in a Python environment as detailed in the GitLab repository provided by the PUNCH4NFDI team [38] and then running the analysis as shown in the basic tutorials of the same repository.

The implementation was already performed for the CMS open data analysis code by the PUNCH4NFDI team, as this analysis is used as one of their REANA example tutorials [16], as previously mentioned in Chapter 3. The tutorial provides a comprehensive explanation of the steps required to perform the analysis on REANA, which can be carried out by any individual with a REANA account. The `reana.yaml` file, which defines the structure of the workflow, can also be found there. There were no significant difficulties encountered in performing the CMS analysis on REANA, other than the learning curve for using the REANA environment to run workflows. However, this was not a major issue, as there are numerous tutorials available on the aforementioned PUNCH4NFDI GitLab to assist with learning the basics of REANA [38].

The ATLAS analysis presented a more complex scenario. As previously stated in Chapter 3, the ATLAS analysis code is part of a set of educational Jupyter notebooks provided by ATLAS Open Data and is publicly accessible [29]. However, the ATLAS analysis had not already implemented on REANA, as was the case with the CMS analysis. Consequently, it was necessary to implement it in accordance with the steps outlined at the beginning of this section.

The source code was the ATLAS analysis itself, which was available on [17]. Almost no alterations were made to the code. The only modification was adding a code line to save the final plot once the analysis is done, but this will be explained in greater detail later on. Furthermore, the supplementary file, `infofile.py`, must also be included as input to the workflow, as previously outlined in Chapter 3.

In regard to the execution environment, the initial plan was to use one of the numerous available environments provided by PUNCH4NFDI [40]. However, none of those environments satisfied all the requisite criteria for the ATLAS analysis. Therefore, it was necessary to develop a custom Docker image with the instructions for the creation of a Docker container capable of running the analysis. The custom Docker image for the ATLAS analysis was created in accordance with the instructions set forth in the advanced tutorial "Create custom Images" [38]. Even the same base image, `jupyter/scipy-notebook`, was employed. This is due to the fact that the Docker image in question had been prepared to be used with Jupyter Notebooks, just like the ATLAS open data analysis.

The only difference with respect to the tutorial were the packages and libraries included in the python environment inside the Dockerfile. All the libraries and packages necessary to run the ATLAS code, instead of the ones used on the tutorial, were included there. In addition, the `papermill` package was also included, as it is a necessary tool for parameterizing and executing Jupyter Notebooks. The Docker image can be now found as one of the environments provided by PUNCH4NFDI [41].

To use a new environment in a REANA analysis, one has to go inside the repository were that environment is stored, go to `Deploy > Container registry` and then copy the image path of the desired container. This is the path that has to be included in the `workflow/steps/environment` section of the analysis YAML file.

Once all these elements are ready, the YAML file defining the analysis must be created. All the before explained elements were included there with the computational steps that define the analysis workflow. The `reana.yaml` file, together with the ATLAS Open Data analysis code, can be found on a GitHub repository made for this purpose [42]. The `reana.yaml` file itself is quite simple. It defines the source code `HZZAnalysis.ipynb` and `infofile.py` as input files. Then it defines some parameters to make the file more comprehensible and easy to read. After that it defines the type of workflow, the environment used for the analysis (the one made on purpose for this one) and the computational steps to perform.

The first step is the creation of the directory `results` where the outputs will be stored. However, for the output plot produced by the ATLAS analysis to be stored here, the code line `plt.savefig('results/ATLAS_plot.pdf'` `format='pdf', dpi=150)` had to be added to the defined `plot_data` function inside the analysis code. The last step uses the tool `papermill` to run the analysis code. Finally, the `reana.yaml` file defines the resulting plot of the ATLAS Open Data analysis as the output of the workflow.

With the instructions and the material provided on this chapter, any user with a working REANA account should be able of running both the CMS and the ATLAS analysis using CERN Open Data on REANA, and then, retrieving the resulting plots from the users REANA profile.

5.2 Implementation of Compute4PUNCH and Storage4PUNCH

Integrating the two analysis on REANA was the first step towards the simplified combination of both of them into a single unified analysis. Nevertheless, this is not yet enough to achieve the objective of the thesis, as both analysis are not yet integrated into the PUNCH4NFDI framework. To solve this, both REANA analysis need to be modified to make use of the current available computing and storage resources on PUNCH4NFDI: Compute4PUNCH and Storage4PUNCH.

Just like in the last section, the implementation for the CMS example was performed by the PUNCH4NFDI team, as it served as an example of the actual integration of Compute4PUNCH and Storage4PUNCH in an already working REANA analysis. For this, a new YAML file was created. This one is called reana-c4p.yaml and it can be found on the same repository of the basic REANA analysis without C4P and S4P [16]. There are multiple new things on this YAML file. Firstly, a stage_out script is added as an additional input file and a new stage_out step is added to the workflow. This will take care of storing the final plots into S4P. Secondly, a new environment, named `wlcg-wn:latest`, is used. This environment contains some extra libraries that are needed to submit jobs to C4P. Another new detail that is easy to miss can be found on the analysis step. Here, on the code line which runs the analysis,

```
root -b -x -q 'code/df103_NanoAODHiggsAnalysis.C+(false, true)',
```

the second parameter of the analysis function now is set to be true. By taking a look at the CMS analysis code on the REANA repository, one realises that setting this parameter as true instead of false, which is the default value, indicates the code to read the data from S4P instead of directly from CERN Open Data. Finally, a new element was introduced on the YAML file, the `compute_backend`, which allowed to select C4P as the computing infrastructure for this analysis. Putting together all these modifications, it was possible to run the CMS Open Data analysis on REANA while using S4P and C4P.

Then again, the ATLAS Open Data analysis presented more difficulties. Similar modifications were applied to the YAML file, and the source code was also modified to get the data for a special directory made in S4P with the data to run the ATLAS analysis, instead of directly from ATLAS Open Data. However, new setbacks appeared. At first there were problems getting access to S4P, as errors kept appearing saying that it was necessary to install and import some additional packages into the analysis code to be able to access the data files, like `fsspec-xrootd`. For this it would be necessary to create a new Docker image including this packages. Before doing this, and to make sure there were no more needed modifications to the environment, it was decided to keep testing the analysis code getting the data from ATLAS Open Data instead of S4P, but still trying to use C4P. And in fact, an additional problem related with the environment appeared. As explained with the CMS analysis implantation on C4P, additional libraries are needed to submit jobs to C4P, and this libraries are already contained on the `wlcg-wn:latest`, which is an special environment created by the PUNCH4NFDI for this purpose. However, this environment did not provide the necessary libraries and packages to run the ATLAS analysis, like `papermill`. After discussing with people involved on these aspects of the PUNCH4NFDI development, it was concluded that the solution was not use git-based images, but to update their images to include the libraries and packages already used on the working REANA implementation of the ATLAS analysis. Unfortunately, it was not possible for the PUNCH4NFDI team to perform this modifications before the end term of this thesis, as there are OS incompatibilities that, even though they will be overtaken in the future, it will take time as they are building a quite complex infrastructure. In any case, all the work done towards the implementation of C4P and S4P on the ATLAS analysis on REANA can be found in the provided repository [42]. The adapted YAML files and the modified source codes found there are the main products of this thesis implantation work, and even though is not finish, it is hoped that it will be of great use for those who try to finish the job once new solutions and resources are available.

5.3 Simplified combined analysis

Although the original intention was to present a simplified combined analysis of both the CMS and ATLAS Open Data analyses, it was not possible to do so before the end of the thesis. This was largely due to lack of time, but also because some of the resources that were to be used for the combination needed more time to develop. However, it is still possible to present the possible steps forward a simplified combined analysis. This may be useful to visualise how this combination would have been carried out, and may even serve as a guide for those who take up the work after the completion of this thesis.

The first idea was to combine the results of the analyses by means of a "workflow of workflows". The idea was to create a REANA workflow capable of running the ATLAS and CMS REANA workflows, i.e. to run two additional workflows within the main workflow. This main workflow would then access the results of the ATLAS and CMS analyses and perform the combination with them. So with this method, instead of creating a REANA workflow to perform exactly the same analyses as the workflows already created to then fuse the results, one could have a workflow to actually reuse those workflows to perform the combination. The intention behind this was to experiment to see if it was possible to make the ATLAS and CMS workflows more interoperable and reusable. However, a "workflow of workflows" is not currently possible on the REANA platform.

The alternative chosen was to create a workflow, which will be called a combination workflow from now on, that only replicates the last part of the previous idea. That is, to take the results of the CMS and ATLAS workflows and combine them. In order for this workflow to work, one first has to run the CMS and ATLAS workflows manually and store the results somewhere where the combination workflow can access them. This is obviously a naive way of doing a combined analysis, but that does not make it any less valuable. A simple combined analysis like this is really useful for defining the basics of performing this type of workflow, and it could serve as a starting point for a more sophisticated implementation in the future.

However, before creating the combined workflow, it was necessary to make some modifications to the implemented ATLAS analysis. One unavoidable problem was that the ATLAS analysis only produces a PDF of the final plot, which was not sufficient to perform a combined analysis. The first idea was to add a new function to the ATLAS code to store the histograms as `TH1D` classes, which is a ROOT class that basically corresponds to a 1-d histogram, inside a ROOT file. The `TH1D` object class was chosen because ROOT provides tools to perform summations of this type of histogram, which would be useful for combining the results. However, it was found that `uproot` alone does not allow ROOT `TH1D` classes to be created and stored directly. An additional library, such as `pyRoot`, would be needed to do this. This would imply a further modification of the ATLAS REANA environment to include this library and be able to run the code on REANA. One could also try to run the modified code locally, but then it would be necessary to install `pyRoot`. This was not possible here for reasons of time and complexity.

Another issue that needed to be addressed was the number of bins and the range of the x-axis on the histograms for each analysis. The simple combination of the analysis results that was intended was basically to add the bin contents of the result histograms of the same type (signal, background or data) to a third histogram. For this operation to be possible, the bin edges of the histograms in both analyses must be the same. To ensure this, the number of bins and the minimum and maximum values of the x-axis in the ATLAS plot were modified to be the same as in the CMS plot. All these changes to the ATLAS code can be found in the repository [42], although the `pyRoot` problem still needs to be resolved. No changes were needed to the CMS analysis code, as it already saves a ROOT file with all the plotting information.

This was the last real section of implementation work that could be carried out during the term of this thesis. However, before moving on to the conclusions, it is still possible to present some steps towards the completion of the work started here. Once the ATLAS analysis is ready to produce its histograms inside a ROOT file, the code for the combination workflows has to be written. This code needs to retrieve the histograms from the ROOT files from both analyses as `TH1D` histograms. Getting the histograms from the ATLAS ROOT file should be straightforward as they are already stored as `TH1D` classes. However, the CMS analysis actually saves a `canvas` ROOT class instead. For a closer look at this ROOT class, please refer to the documentation [43]. The only thing that should be known about this class here is that one should be able to retrieve the histograms from a `canvas` object using the ROOT function `GetPrimitive`. Further steps to fix this and other implementation/combining problems are given in Chapter 6.

CHAPTER 6

Conclusion

In this thesis, two different analyses of the Higgs decay to four leptons, one using data from the ATLAS detector and the other from the CMS detector, were selected and integrated into the REANA data analysis platform. This allowed users with access to REANA to run the provided analyses, using real collision data from the LHC provided by CERN Open Data, and retrieve scientifically interesting results. The successful integration of the analyses within REANA demonstrates the potential of this and similar platforms to perform data-intensive research and to process data from different sources (e.g. ATLAS and CMS), all while supporting the principles of FAIR and Open Science.

Steps have also been taken to implement these analyses in the PUNCH4NFDI infrastructure. Although there have been many difficulties along the way, many of the setbacks encountered have been caused by issues related to the ongoing development of the PUNCH4NFDI project, which have either been resolved or are expected to be resolved in the near future. However, one problem that has not yet been solved is the integration of both C4P and S4P on the ATLAS analysis, and although the integration with S4P could be done with a small modification of the environment used, the C4P problem will still take some time to be solved. In any case, the work done here contributed to the PUNCH4NFDI infrastructure as a showcase of its accessibility and reproducibility capabilities, and even though it was not possible to fully integrate the ATLAS analysis, the foundations were laid for doing so.

In addition to carrying out the analyses on REANA and the PUNCH4NFDI infrastructure, efforts were made to create a more comprehensive combined analysis. However, the work faced many challenges. Firstly, the idea of building and running a 'workflow of workflows' had to be replaced by the plan to build a simple analysis that would take the histograms already prepared from each analysis and combine them. Many other challenges were encountered, mainly in dealing with the differences in the way each analysis handled and stored histogram data. The ATLAS analysis was the most problematic, as the code had to be modified, including functions from a new library,`pyRoot`, to produce an output ROOT file with the required `TH1D` histograms. Despite the progress made, unresolved technical barriers, particularly in the implementation of `pyRoot`, limited the resolution of the ATLAS analysis problems, and so it was not possible to achieve full realisation of this combination within the timeframe of this thesis, leaving it as a clear avenue for future work.

Looking ahead, there are several key areas where further research and development is required. An immediate priority is to resolve the `pyRoot` related issues that would allow the seamless combination of the CMS and ATLAS histograms. This can be done by creating a new environment for the ATLAS analysis, including the `pyRoot`. In addition, the necessary libraries to access the data from S4P can be implemented at the same time to try to solve the problems with the integration of S4P. Once the ATLAS analysis is able to store histograms as `TH1D` objects, the unified framework can be fully realised, allowing cooperative analysis between CMS and ATLAS data on the REANA platform. This would fulfil the original aims of the thesis and serve as a clear example of the interoperability capabilities of the PUNCH4NFDI project.

Bibliography

[1] The ATLAS Collaboration, *Observation of a new particle in the search for the Standard Model Higgs boson with the ATLAS detector at the LHC*, Physics Letters B **716** (2012) 1, ISSN: 0370-2693, URL: https://www.sciencedirect.com/science/article/pii/S037026931200857X (cit. on pp. 1, 5, 8, 9, 11, 12).

[2] The CMS Collaboration, *Observation of a new boson at a mass of 125 GeV with the CMS experiment at the LHC*, Physics Letters B **716** (2012) 30, ISSN: 0370-2693, URL: https://www.sciencedirect.com/science/article/pii/S0370269312008581 (cit. on pp. 1, 5, 8–12).

[3] *Science Data Platform and Digital Research Product. Presentation*, URL: https://escience.aip.de/ag2022/AG2022-H_Enke-PUNCH4NFDI-TA4.pdf (visited on 16/08/2024) (cit. on pp. 1, 19).

[4] M. Thomson, *Modern Particle Physics*, Cambridge University Press, 2013 (cit. on pp. 2, 4, 6, 7).

[5] A. Purcell, *The Standard Model infographic*, (2012), URL: https://cds.cern.ch/record/1473657/files/SMinfographic_image.png (visited on 20/07/2024) (cit. on p. 2).

[6] P. W. Higgs, *Broken Symmetries and the Masses of Gauge Bosons*, Phys. Rev. Lett. **13** (16 1964) 508, URL: https://link.aps.org/doi/10.1103/PhysRevLett.13.508 (cit. on p. 3).

[7] F. Englert and R. Brout, *Broken Symmetry and the Mass of Gauge Vector Mesons*, Phys. Rev. Lett. **13** (9 1964) 321, URL: https://link.aps.org/doi/10.1103/PhysRevLett.13.321 (cit. on p. 3).

[8] *The Large Hadron Collider*, URL: https://home.cern/science/accelerators/large-hadron-collider (visited on 20/07/2024) (cit. on p. 5).

[9] E. Lopienska, *The CERN accelerator complex, layout in 2022. Complexe des accélérateurs du CERN en janvier 2022*, (2022), General Photo, URL: https://cds.cern.ch/record/2800984 (visited on 20/07/2024) (cit. on p. 5).

[10] *ATLAS*, URL: https://home.cern/science/experiments/atlas (visited on 20/07/2024) (cit. on p. 6).

[11] *CMS*, URL: https://home.cern/science/experiments/cms (visited on 20/07/2024) (cit. on p. 6).

[12] M. C. Hansen, *Studies into measuring the Higgs CP-state in $H \rightarrow \tau\tau$ decays at ATLAS*, PhD Thesis: Universität Bonn, 2020, URL: https://hdl.handle.net/20.500.11811/8456 (cit. on p. 6).

[13] J. Pequenao, *Computer generated image of the whole ATLAS detector*, (2008), URL: https://cds.cern.ch/record/1095924 (visited on 21/07/2024) (cit. on p. 6).

[14] *The CMS Detector*, URL: https://cms.cern/detector (visited on 21/07/2024) (cit. on p. 7).

[15] *The ATLAS Detector*, URL: https://atlas.cern/Discover/Detector (visited on 21/07/2024) (cit. on p. 7).

[16] *REANA CERN Open Data Tutorial*, URL: https://gitlab-p4n.aip.de/compute4punch/tutorials/reana-cern-open-data-tutorial/-/tree/main?ref_type=heads (visited on 06/08/2024) (cit. on pp. 8, 11–14, 21, 22).

[17] *HZZAnalysis.ipynb*, URL: https://github.com/atlas-outreach-data-tools/notebooks-collection-opendata/blob/master/13-TeV-examples/uproot_python/HZZAnalysis.ipynb (visited on 07/08/2024) (cit. on pp. 8, 12, 13, 21).

[18] *Open research data*, URL: https://www.unesco.org/en/open-science/open-research-data (visited on 05/08/2024) (cit. on p. 9).

[19] *CERN Open Data Policy for the LHC Experiments*, tech. rep., CERN, 2020, URL: https://cds.cern.ch/record/2745133 (cit. on pp. 9–11).

[20] *CERN Open Data portal*, URL: https://opendata.cern.ch/ (visited on 05/08/2024) (cit. on p. 10).

[21] Z. Akopov et al., *Status Report of the DPHEP Study Group: Towards a Global Effort for Sustainable Data Preservation in High Energy Physics*, 2012, arXiv: 1205.4667 [hep-ex], URL: https://arxiv.org/abs/1205.4667 (cit. on p. 10).

[22] *CERN Open Data portal description*, URL: https://opendata.cern.ch/docs/about (visited on 05/08/2024) (cit. on p. 10).

[23] *CERN Open Data: About ATLAS*, URL: https://opendata.cern.ch/docs/about-atlas (visited on 07/08/2024) (cit. on p. 11).

[24] *ATLAS Open Data*, URL: https://opendata.atlas.cern/ (visited on 07/08/2024) (cit. on p. 11).

[25] *ROOT. Data Analysis Framework*, URL: https://root.cern/ (visited on 16/08/2024) (cit. on p. 11).

[26] *df103_NanoAODHiggsAnalysis.py File Reference*, URL: https://root.cern.ch/doc/v622/df103__NanoAODHiggsAnalysis_8py.html (visited on 06/08/2024) (cit. on p. 11).

[27] Jomhari, Nur Zulaiha, Geiser, Achim and Bin Anuar, Afiq Aizuddin, *Higgs-to-four-lepton analysis example using 2011-2012 data*, CERN Open Data Portal (2017), (visited on 06/08/2024) (cit. on pp. 11, 12).

[28] CMS collaboration, *2020 CMS data preservation, re-use and open access policy*, CERN Open Data Portal (2020), (visited on 06/08/2024) (cit. on p. 12).

[29] *Physics Searches: Standard Model*, URL: `https://opendata.atlas.cern/docs/13TeVDoc/13tutorial/` (visited on 07/08/2024) (cit. on pp. 12, 21).

[30] *Uproot*, URL: `https://github.com/scikit-hep/uproot5` (visited on 08/08/2024) (cit. on p. 13).

[31] Drabent, Alexander et al., *Federated Heterogeneous Compute and Storage Infrastructure for the PUNCH4NFDI Consortium*, EPJ Web of Conf. **295** (2024) 07020, URL: `https://doi.org/10.1051/epjconf/202429507020` (cit. on pp. 15–17).

[32] *FAIR Principles*, URL: `https://www.go-fair.org/fair-principles/` (visited on 16/08/2024) (cit. on p. 15).

[33] The PUNCH4NFDI Consortium, *PUNCH4NFDI Consortium Proposal*, version v1 without funding tables, 2021, URL: `https://doi.org/10.5281/zenodo.5722895` (cit. on p. 15).

[34] H. Team, *HTCondor*, version 10.7.1, 2023, URL: `https://doi.org/10.5281/zenodo.8230603` (cit. on p. 16).

[35] PUNCH4NFDI Consortium, *PUNCH-AAI registration procedure*, (2023), URL: `https://www.punch4nfdi.de/sites/sites_custom/site_punch4nfdi/content/e117438/e142359/e233013/PUNCH4NFDI.AAI-registration-procedure.pdf` (cit. on p. 16).

[36] H. Enke and T. Schörner-Sadenius, *Science Data Platform and Digital Research Product*, Proceedings of the Conference on Research Data Infrastructure **1** (2023) (cit. on pp. 17, 19).

[37] *REANA GitHub*, URL: `https://github.com/reanahub` (visited on 12/08/2024) (cit. on p. 18).

[38] *REANA Tutorial 2024*, URL: `https://gitlab-p4n.aip.de/p4nreana/tutorials` (visited on 12/08/2024) (cit. on pp. 18, 21).

[39] *PUNCH4NFDI. Service Class 4: Metadata services*, URL: `https://www.punch4nfdi.de/services/service_classes/service_class_4/` (visited on 16/08/2024) (cit. on p. 19).

[40] *Environments for REANA provided by PUNCH4NFDI*, URL: `https://gitlab-p4n.aip.de/p4nreana/reana-env` (visited on 14/08/2024) (cit. on p. 21).

[41] *Environment for the ATLAS open data analysis*, URL: `https://gitlab-p4n.aip.de/p4nreana/reana-env/-/tree/atlas-c4p?ref_type=heads` (visited on 14/08/2024) (cit. on p. 21).

[42] *ATLAS-Open-Data-analysis-implementations*, URL: https://github.com/Forti7/ATLAS-Open-Data-analysis-implementations (visited on 16/08/2024) (cit. on pp. 21, 23, 24).

[43] *TCanvas Class Reference*, URL: https://root.cern.ch/doc/master/classTCanvas.html (visited on 16/08/2024) (cit. on p. 24).

List of Figures