

Front-End development of the graphical user interface for NTuple Wizard

CERN Summer Student Programme 2021

Author:

Donijor Tropmann(RWTH Aachen)

Under supervision of:

Adam Morris(Universität Bonn)

Dillon Fitzgerald(University of Michigan)

Chris Burr(CERN)

September 15, 2021

Abstract

The aim of this report is to cover the work done during my placement at CERN as a Summer Student. It is related to the frontend development of NTuple Wizard — a system to access large-scale open data from LHCb.

1 Introduction

A common problem, which is acute for data in high energy physics but increasingly an issue in other fields as well, is the sheer size of datasets stored in custom file formats. Extensive compute infrastructure has been set up by the countries participating in this type of research. Replicating such an infrastructure to allow the public to handle the respective data would not only require dedicated expert knowledge, it would also duplicate existing infrastructure. It is therefore natural to provide the public access to data and develop a system which leverages the existing compute infrastructure available to the LHCb collaboration in order to enable third party users to request specific data queries. A web interface of the system allows user to discover accessible datasets and guides the user through the process of specifying a request.

1.1 Data Levels

Data produced by the LHC experiments are usually categorised in four different levels :

- **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

1.2 LHCb data flow

Figure 1 shows an overview of the data processing pipeline in LHCb. Level 3 data has been defined as the output of the “Stripping” processing step. The Stripping consists of a large number of selection algorithms called “lines”, which are designed to filter the data and sort events into several collections, which are called “streams” in LHCb terminology. Each stream consists of a set of files, which contain the data selected by an associated set of selection algorithms. Streams are defined according to common physics signatures and aim to collect selections with significant overlaps into a common set of files, to reduce duplication of the data. An important technique to reduce data size during the stripping step, is to only keep those physics objects, which have been associated with at least one of the selections running the stripping. The rest of the event data is stripped from the stream (hence the name Stripping).

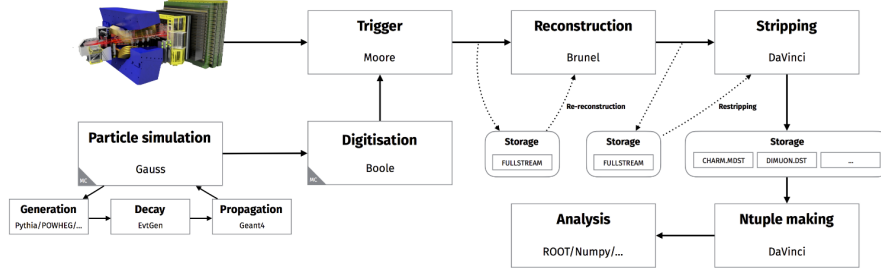


Figure 1: LHCb data flow in Run I and II. The output of the Stripping step will be made public through the CERN Open Data portal

2 NTuple Wizard

NTuple Wizard allows a generation of Ntuples without knowledge of the LHCb software stack. The input provided by the user is converted into a set of configuration files. LHCb analysis production uses these configuration files in order to create Ntuples and return them to the user.

Figure 2 shows a detailed overview of the architecture of the NTuple Wizard. The metadata describing the available data, preselections, as well as the available selection operations is generated from the original provenance traces of the data and the selection code. The web interface presents this metadata to the user in an instructive way, that facilitates data discovery and formulation of the query. The input provided by the user is converted into a set of configuration files, which can be used to configure the internal LHCb analysis productions system. Since the configuration files are generated by the system, the user is isolated from the production environment, which prevents the injection of potentially malicious code.

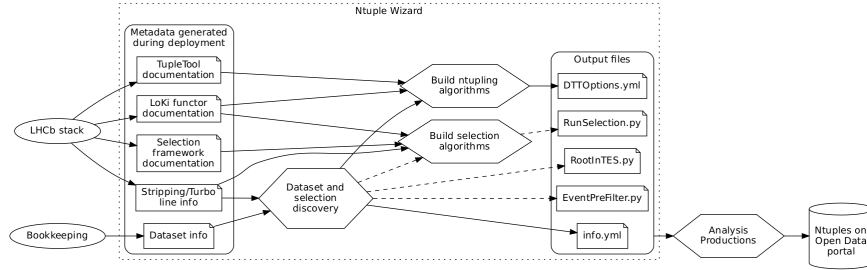


Figure 2: Architecture of the NTuple Wizard.

3 NTuple Wizard frontend

The frontend part of NTuple Wizard consists of a sequence of web pages that guide the user through the configuration steps. This is designed as a client-side dynamic web page that reads metadata acquired at deployment-time and served as static files. The metadata and documentation will be delivered to the user at the optimal stage of the wizard in order to ensure the proper bookkeeping locations are selected and necessary knowledge of configurable algorithms is provided. Once a dataset(s) is selected based on particle/decay candidate criteria, candidates in consideration are assigned to nodes of a configurable graph, in which variables to be written into NTuples can be added to specific nodes, or a collection of nodes satisfying certain criteria. Users interact with the nodes of the graph in order to configure algorithms and selection requirements for various particle/decay candidates

3.1 Decay descriptor selection

The functionality of this page allows user to load decay descriptors dynamically from large data files. Descriptors are converted by MathJax to user-friendly mathematical formulas. In addition, Stripping lines and datatypes are displayed as metadata. As shown in Figure 3(a) user can filter a large list of decay descriptors according to selected particles in the select field. Figure 2(b) demonstrates that user can select multiple descriptors for further processing. After the selection of desired decay descriptors, the user can start the lines selection process.

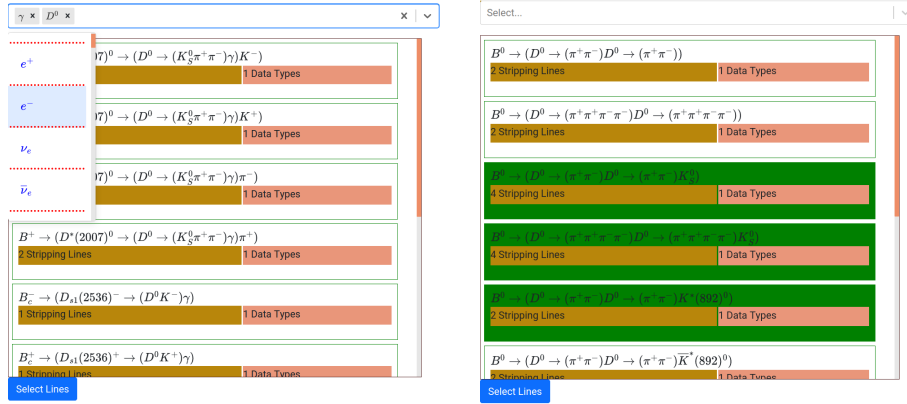


Figure 3: (a) select box for particle selection; (b) selection of multiple decay descriptors

3.2 Stripping lines selection

On this page, user can further process the selected descriptors from the previous step. As shown in Figure 4, the user can specify which line and datatype can be selected for particular decay. User can also exclude from consideration decays by removing them from the table.

Lines Search

Decay	Line	Datatype	
$B^0 \rightarrow (D^0 \rightarrow (\pi^+ \pi^-) D^0 \rightarrow (\pi^+ \pi^-) K_S^0)$	StrippingB02D0D0KSD02HHD02K3PILLBeauty2...	2011	Remove
$B^0 \rightarrow (D^0 \rightarrow (\pi^+ \pi^+ \pi^- \pi^-) D^0 \rightarrow (\pi^+ \pi^+ \pi^- \pi^-) K_S^0)$	StrippingB02D0D0KSD02HHD02K3PILLBeauty2...	2011	Remove
$B^0 \rightarrow (D^0 \rightarrow (\pi^+ \pi^-) D^0 \rightarrow (\pi^+ \pi^-) K^*(892)^0)$	Search variables	Select...	Remove

[Build Tree](#)

Figure 4: Lines selection

3.3 Building decay tree graphs

After selecting stripping lines on the previous step user can start a building of a directed acyclic graph for each decay. Figure 4 shows variables assigning page. A left sidebar contains a search field for variables and TupleTools. Hier user can find desired variable or TupleTool and get a short description as a pop-up hint over it(Fig.5).

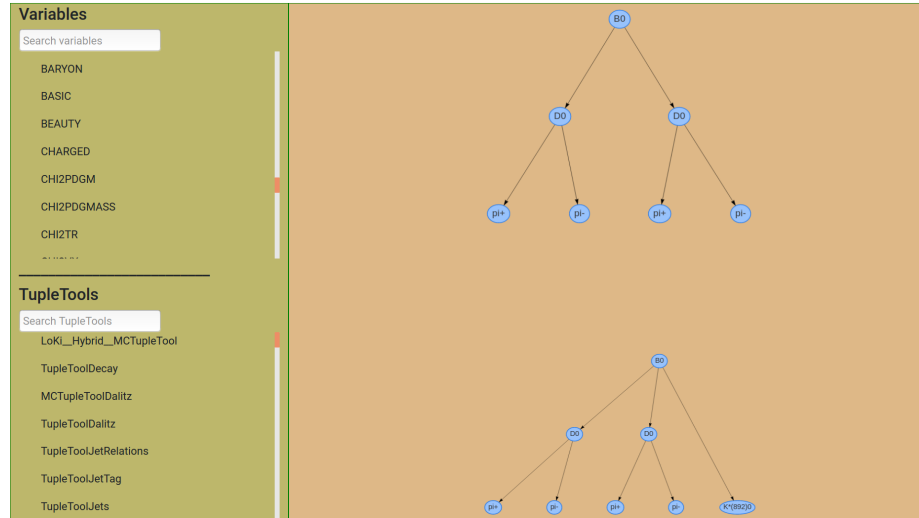


Figure 5: Variables assigning

3.3.1 TupleTools configuration

Users can interact with the nodes of the graph in order to configure TupleTools for various particle/decay.

By clicking on a particular graph node user can call a modal window with a list of assigned TupleTools. As shown in Figure 6, the user can select a new TupleTool from the select field and add it to the list of TupleTools. By clicking on the "Edit Tool" button user can edit selected TupleTools parameters.

The figure displays two screenshots of the TupleTools configuration interface. The top screenshot shows a list of tools with their descriptions and 'EDIT TOOL' and 'DELETE TOOL' buttons. The 'EDIT TOOL' button for 'TupleToolANNPID' is highlighted with a blue box. The bottom screenshot shows the same interface but with the 'Parameters of TupleToolANNPID' section expanded, displaying a table with parameters: ExtraName, Verbose, and MaxPV (set to 100).

Tool	Description	EDIT TOOL	DELETE TOOL
TupleToolANNPID	Fill ANN PID information.	EDIT TOOL	DELETE TOOL
TupleToolGeometry	Fill geometry related information for DecayTreeTuple.	EDIT TOOL	DELETE TOOL
TupleToolEventInfo	Event and Run number for DecayTreeTuple.	EDIT TOOL	DELETE TOOL

Parameters of TupleToolANNPID

Parameter	Value
ExtraName	
Verbose	
MaxPV	100

Figure 6: TupleTools configuration

4 Acknowledgments

I would like to sincerely thank my supervisors and colleagues: Adam Morris, Dillon Fitzgerald, Chris Burr, and Sebastian Neubert, who guided me throughout this project. I am thankful to CERN for giving me this amazing opportunity to participate in the Summer Student Programme.