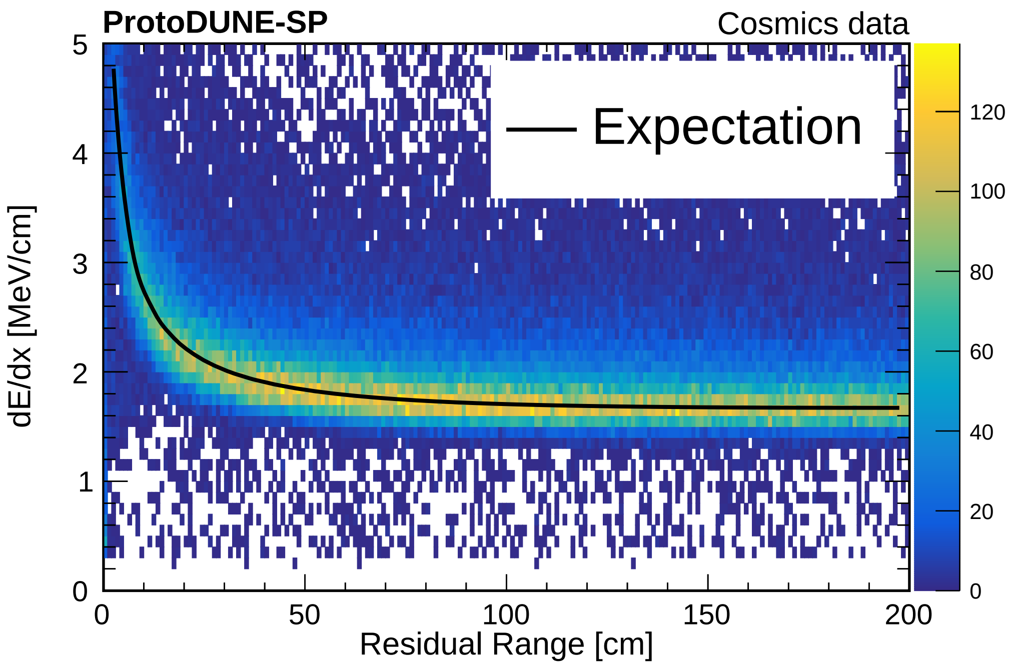


Figure 3.13: dE/dx vs residual range for a stopping muon sample in ProtoDUNE–SP data. Figure from [57].

3.8 ProtoDUNE–SP Online Monitoring System

As well as monitoring the stability of the detector and DAQ systems, the quality of the collected data has to be constantly monitored. This assures the shifter that the data is of high quality, and prevents long runs of low quality data being collected. The online monitoring system (OM) is responsible for providing this quality assurance.

As shown in Figure 3.8, the OM is a part of the ProtoDUNE–SP DAQ system. The OM is responsible for processing the data and displaying the results to the shifter in the control room as soon as possible after the event was triggered. It consists of three main components:

- Analysis processes which decode and analyse the raw data from each detector subsystem.
- Merging processes which collate monitoring data from each subsystem.
- A web interface which displays monitoring data.

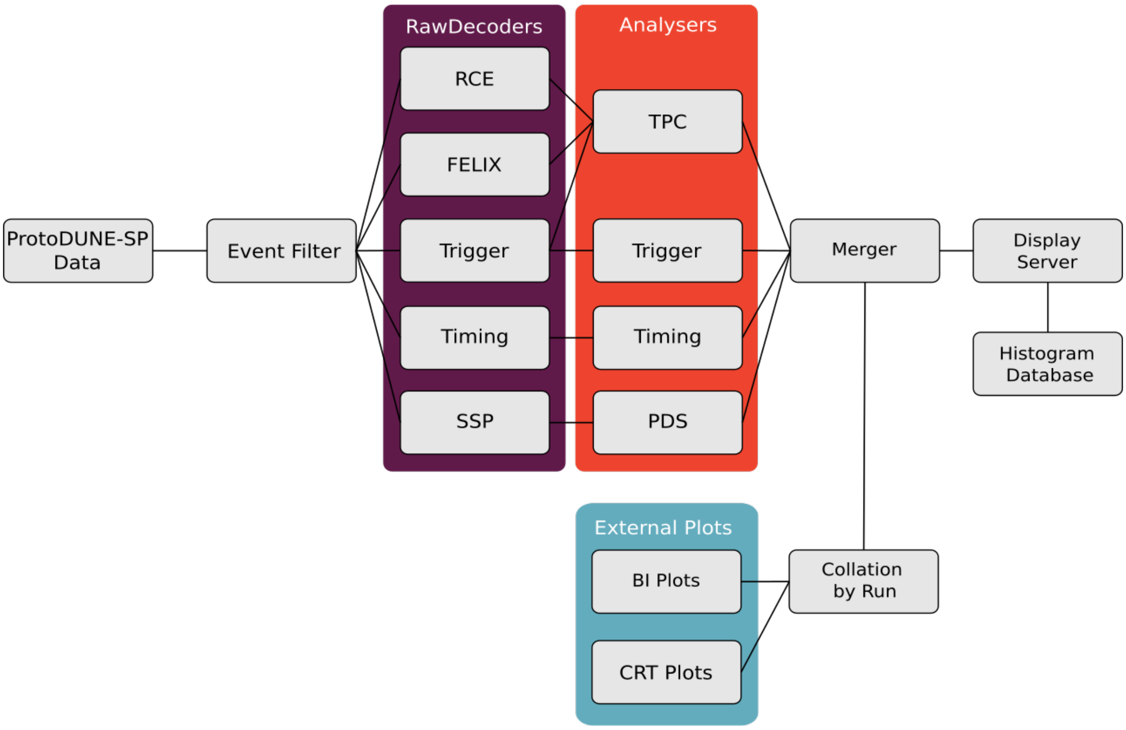


Figure 3.14: Data flow in the ProtoDUNE–SP online monitoring system.

An overview of the data flow in the OM is shown in Figure 3.14. Data from the detector is first filtered before being run in any OM processes. The data which is passed onto the OM processes is then split up and decoded by the relevant RawDecoder which reformats the data ready for analysis. A number of Analyser modules then make use of this data to make plots, which are merged along with plots from external systems, and sent to a web interface to be displayed in the control room. The following sections will provide a brief summary of these stages, as well as examples of the plots which are produced in the OM.

3.8.1 Data Processing

The data processing for the OM is based on Fermilab's art and artdaq software frameworks [60, 62], the only exception is the beam instrumentation data which is analysed outside of the OM system by the CERN beam group. The beam instrumentation plots are merged with the rest of the OM plots after data processing.

The first step of the data processing is event filtering, the main purpose of this step is to control the flow of data into the OM. Two types of filtering take place sequentially; first a random filter is used to cut the data rate into the OM to a manageable level, then a second filter which aims to increase the likelihood that processed events were triggered by the beam instrumentation or cosmic–ray tagger. Different OM processes take different amount of processing time per event, therefore multiple filters are used which control the data rate into each process separately.

After filtering the events are ready to be processed. The data coming into the OM system is in its raw form, it arrives in small pieces known as Fragments which have to be decoded before they can be used by the OM. The RawDecoders are responsible for interpreting the headers and data streams from each detector component, and restructuring the data ready for processing. The details of the decoding vary based on the readout system under consideration; each readout system defines a class which details the contents of each fragment, the RawDecoders use the contents of this class to decode the fragments to prepare the data for processing.

After the data has been decoded it is ready to be analysed, this is done by a number of Analyser modules which analyse the data from different detector components and produce ROOT [72] plots as output. Details of the processing done for each detector component will be given below.

Time Projection Chamber

As the largest data source in ProtoDUNE–SP the TPC data was analysed in several small steps: basic data checks, pedestals and noise, Fourier analysis, and event displays. Examples of some of the plots produced by the TPC analysis are detailed below.

The basic data checks are intended to quickly spot any fundamental issues with the incoming data. An example of a basic check is to check which FEMBs are active in the monitored events, Figure 3.15a shows an example of the number of events recorded by each FEMB on APA 1 for a sample of 10 analysed events.

The pedestals and noise are continually evaluated by the online monitoring, this data is displayed in the control room and used later in the monitoring chain to flatten the background in the event displays. Basic hit removal is used to ensure that the pedestals and RMS are only calculated in the regions of the readout corresponding to noise signals. The RMS of all channels can be represented on a single plot by arranging the channels into a 2D grid and displaying the RMS as the colour scale on a 2D histogram, an example of this plot is shown in Figure 3.15b.

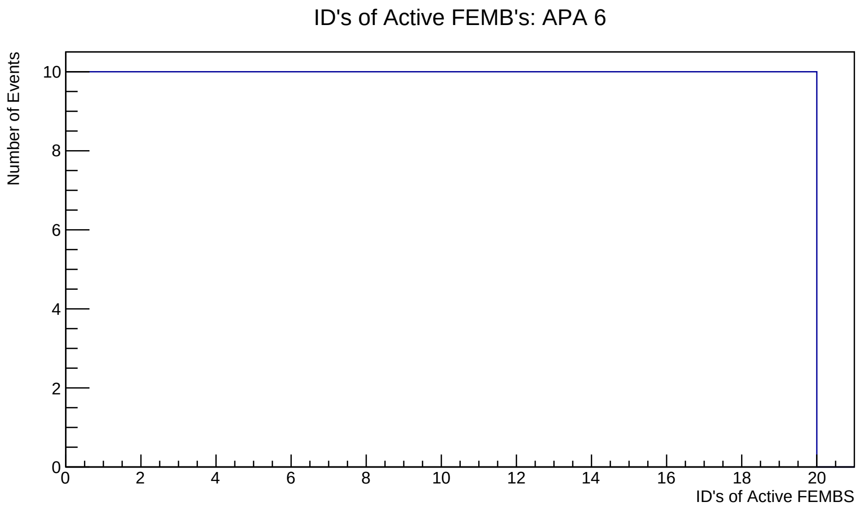
To identify noise sources in the APA it is useful to study the Fourier transform of the signal distribution, this allows the frequencies to be identified and the Fourier distributions can be studied under different conditions to identify noise sources; the cameras in the TPC where identified as a noise source in this way. The OM therefore provides fast Fourier transforms (FFT) for each APA, an example of the FFT for a single FEMB is shown in Figure 3.15c.

Event displays display all the data from all the TPC readout channels simultaneously, and as such they provide the most general check of TPC performance in the OM. During the beam run of ProtoDUNE–SP a number of issues in the data were first identified in the event display, for example issues in the channel mapping and timing synchronisation. They are also particularly useful for checking that the beam trigger results in beam particles in the TPC. As a result a significant effort was made to make event displays available in the ProtoDUNE–SP OM system; in particular changes to the display server were required, these will be discussed later.

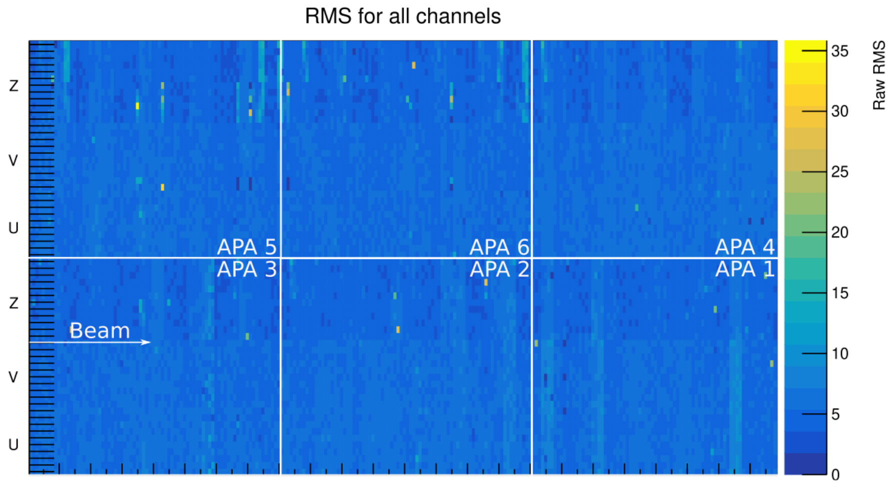
Event displays were offered for all views in all APAs, as well as a specific beam window event displays in each view, and stitched event displays for all of the APAs on each side of the TPC. As the slowest plot to make in the OM, only one set of event displays was made per OM output file; to maximise the number of beam particles in the event displays the trigger information was included during processing. An example of a beam window event display is shown in Figure 3.16; the timing synchronisation issue mentioned previously is visible here, this issue affects a single FEMB and is mitigated during offline reconstruction.

Photon Detection System

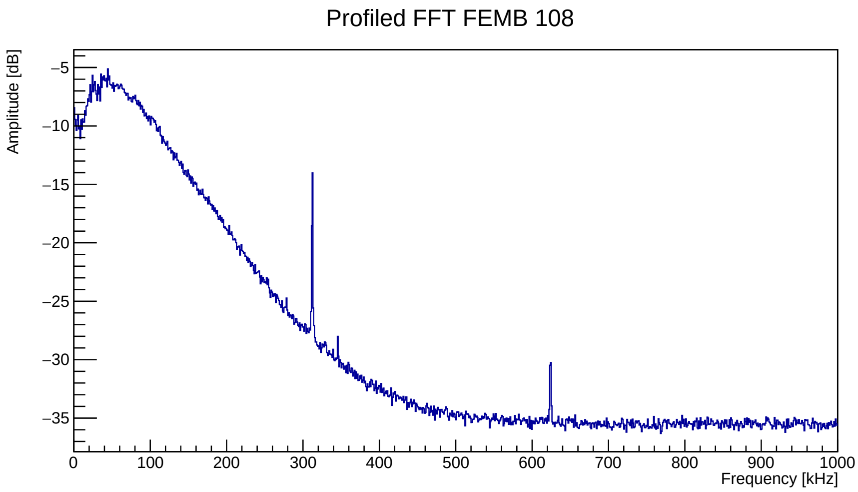
Similarly to the TPC data the PDS data is analysed for pedestal, RMS, and FFTs; in addition the raw waveforms for each channel are accumulated over a number



(a) Number of events recorded by each FEMB on APA 6.



(b) RMS for all channels.



(c) FFT for a single FEMB.

Figure 3.15: Examples of online monitoring plots for the TPC data.

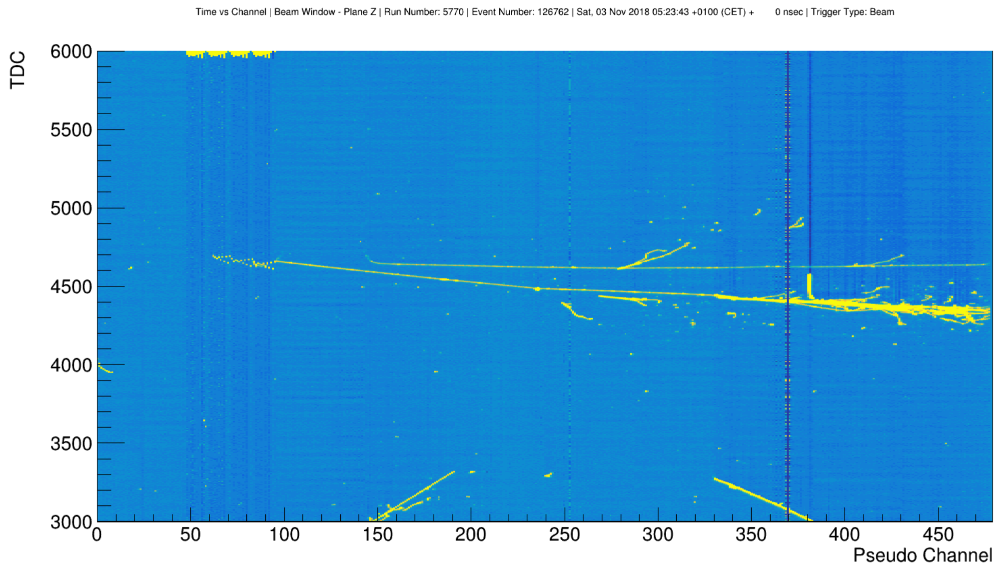


Figure 3.16: An example of a beam window event display from the ProtoDUNE–SP online monitoring system.

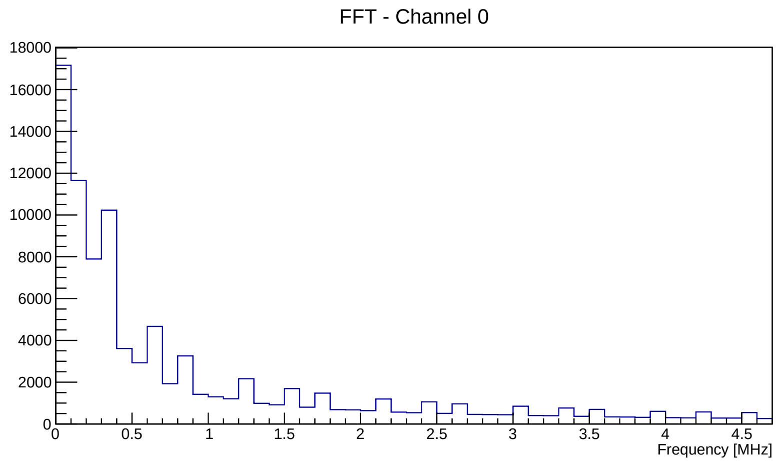
of events and displayed in the online monitoring. Examples of some of the PDS monitoring plots are shown in Figure 3.17.

Trigger and Timing

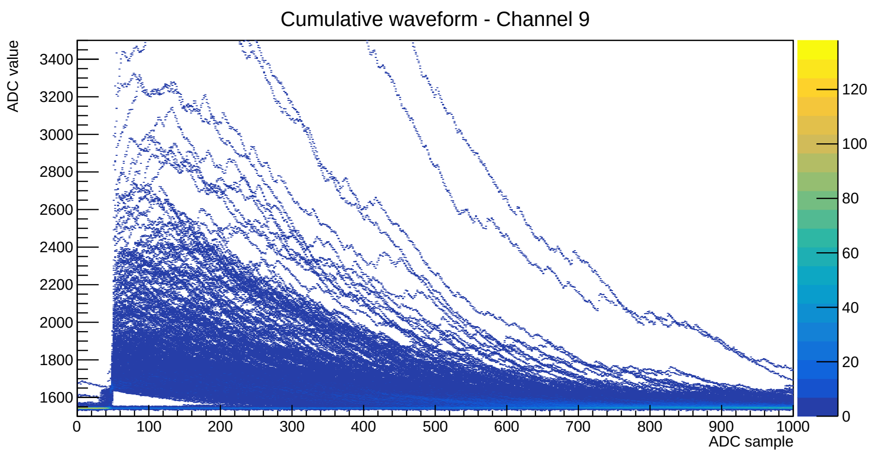
The trigger and timing systems share much of the same data, because the timing system is used to distribute the triggers from the trigger system, so their monitoring is related. Some examples of useful timing and trigger plots include time–stamp difference distributions, and trigger records. The time–stamp delta plots show the difference in time–stamp between consecutive events, this distribution can be used to monitor the stability of the trigger rate during triggered operation. Trigger records are 2D histograms which details of the time and type of trigger issued by the trigger board. Figure 3.18 contains examples of these two plots.

Beam Instrumentation

The beamline instrumentation was monitored by CERN, who produced beamline monitoring plots [63]. The plots where continually overwritten on a time schedule which was decoupled from the ProtoDUNE–SP running schedule, the OM was



(a) FFT for a single PDS channel.



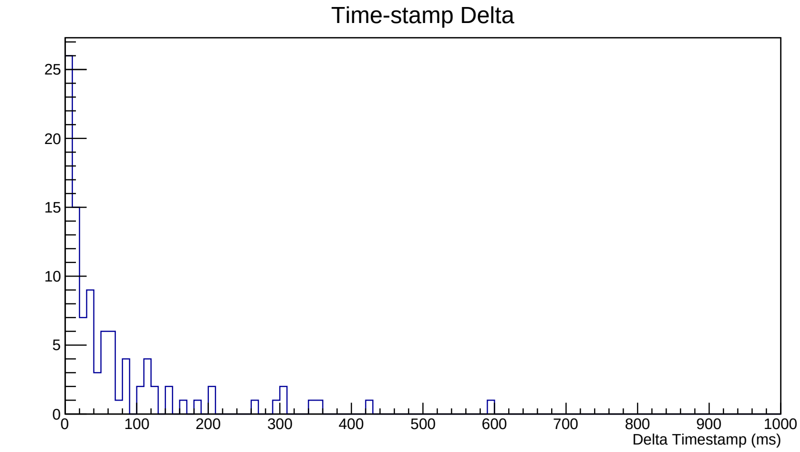
(b) Cumulative waveforms for a single PDS channel.

Figure 3.17: Examples of online monitoring plots for the photon detection system data.

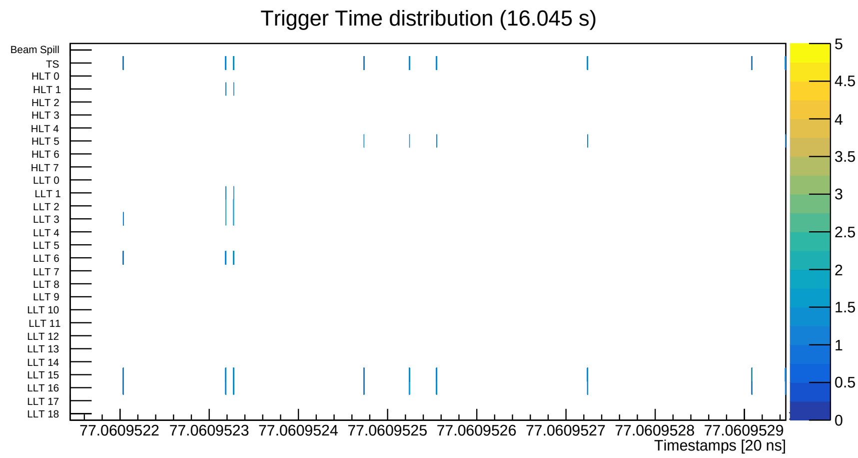
responsible for collating the output of these plots in sync with the ProtoDUNE–SP run schedule, such that the collated beam information for each run could be monitored. After collation the beam instrumentation plots where merged with the rest of the OM plots. The beam instrumentation plots included momentum, and time of flight, examples of which are shown in Figure 3.19.

Cosmic Ray Tagger

A bug in the DAQ meant that the CRT was run separately during the main beam run. Therefore, the CRT data was analysed separately from the rest of the ProtoDUNE–SP systems and merged in the OM similarly to the beam instrumentation data.



(a) Time–stamp delta distribution.



(b) Trigger records.

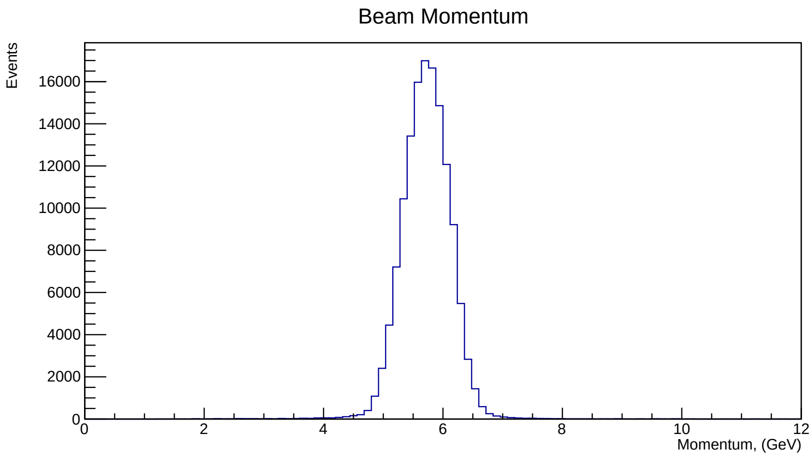
Figure 3.18: Examples of online monitoring plots for the timing and trigger data.

Some examples of CRT monitoring plots include plots of the rate of hits for each channel, and the mean ADC for each channel, these are shown in Figure 3.20.

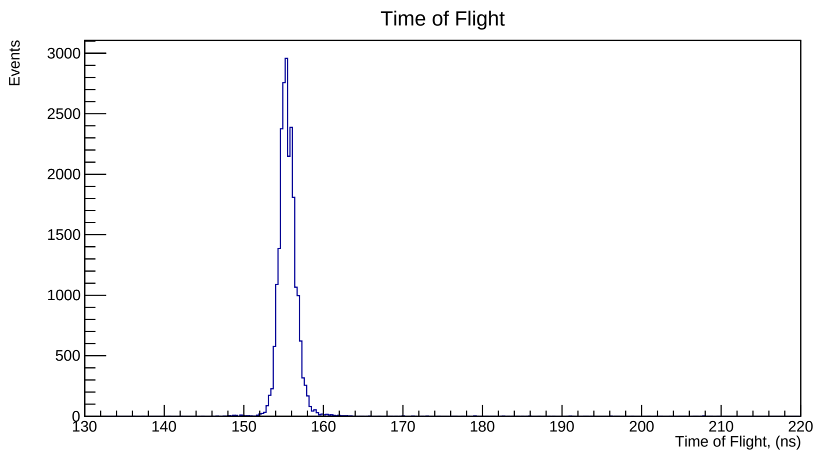
3.8.2 Monitoring Web Interface

The plots from the data analysis are stored in a ROOT file on the ProtoDUNE–SP DAQ servers, this data is viewable from anywhere in CERN via a web interface which was adapted for ProtoDUNE–SP from LHCb's Monet [73]. Monet consists of a Flask web application [74] which uses Bokeh [75] to render plots for display, ROOT [72] and NumPy [76] are used to process the monitoring data in preparation for display.

Two important modifications were made to Monet for ProtoDUNE–SP: efficiency



(a) Beam momentum distribution.



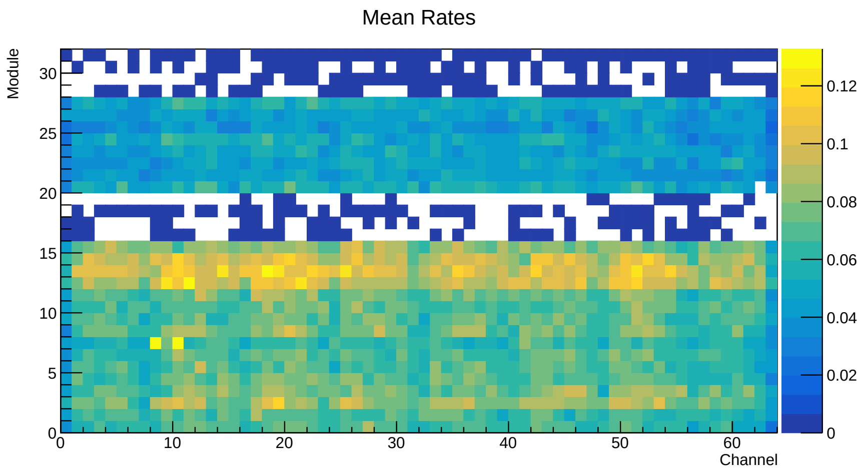
(b) Beam time of flight distribution.

Figure 3.19: Examples of online monitoring plots for the beam instrumentation.

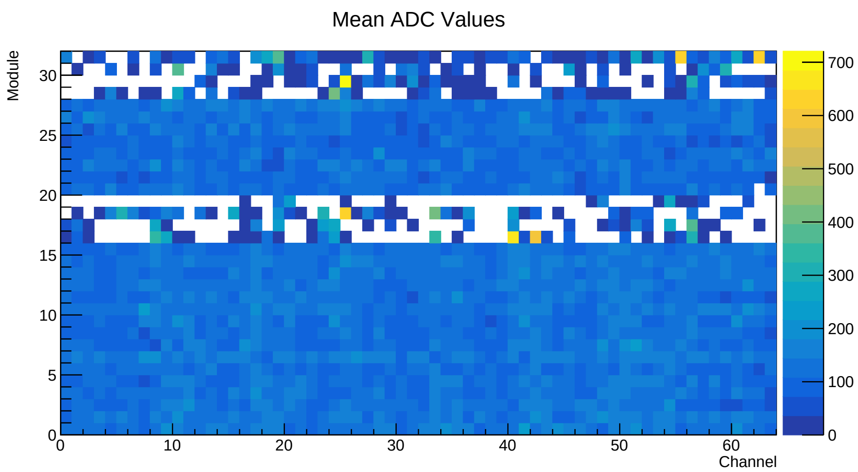
improvements which were required to handle the large amounts of data in the event displays, and the addition of a separate rendering server with additional capabilities.

The components of the web interface, and their connections, are outlined in Figure 3.21. When a user makes a request, which could be a request to open a new page of plots or a request to interact with a plot, the request is sent to the display server which decides what to do with it. Depending on the nature of the request the display server will then interact with the histogram database and/or the rendering server, before updating the plots and displaying them to the user.

The plots in the monitoring files are organised into pages in Monet. Each page can contain any number of plots which are organised into an array on the screen,



(a) Hit rate per channel.



(b) Mean ADC per channel.

Figure 3.20: Examples of online monitoring plots for the cosmic ray tagger.

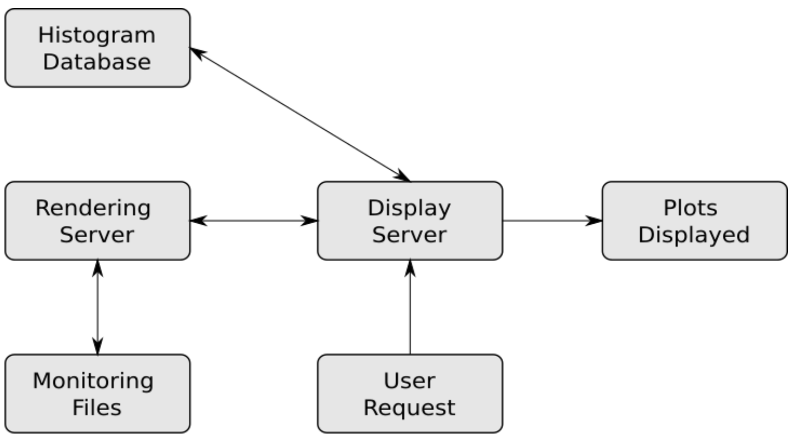


Figure 3.21: Data flow diagram for the web interface in the ProtoDUNE–SP online monitoring system.

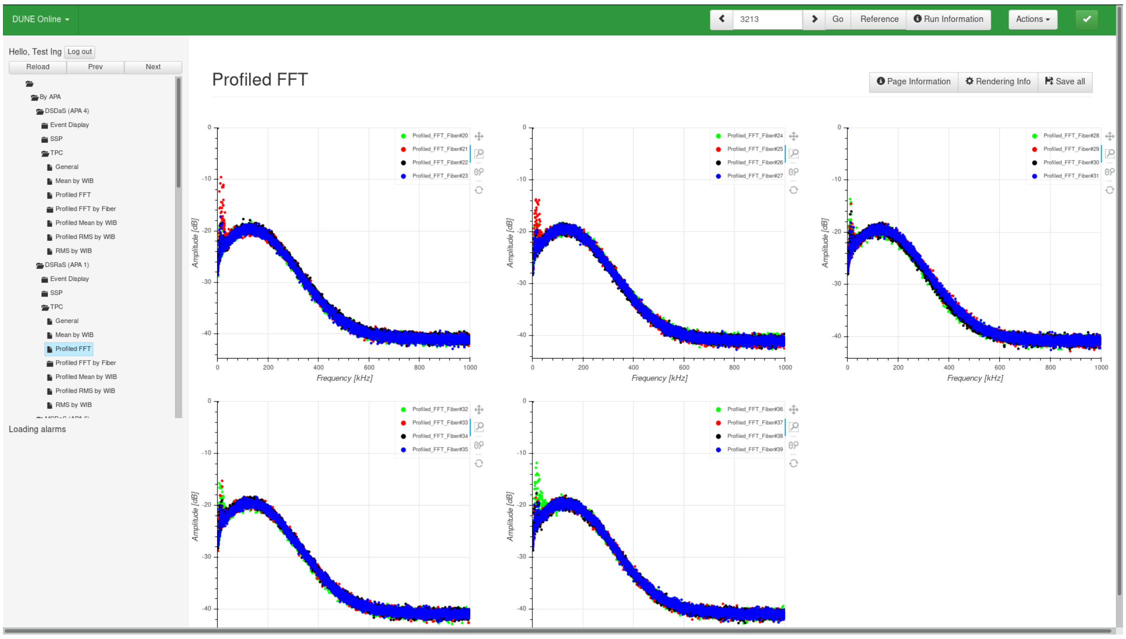
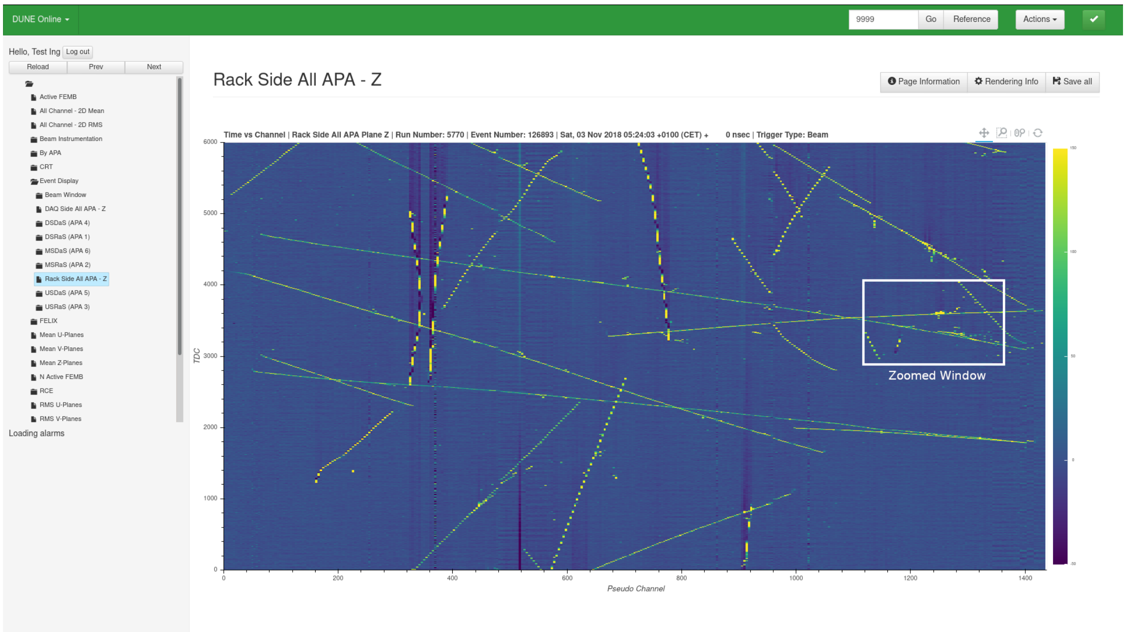


Figure 3.22: An example of an online monitoring page, profiled FFTs for APA 1.

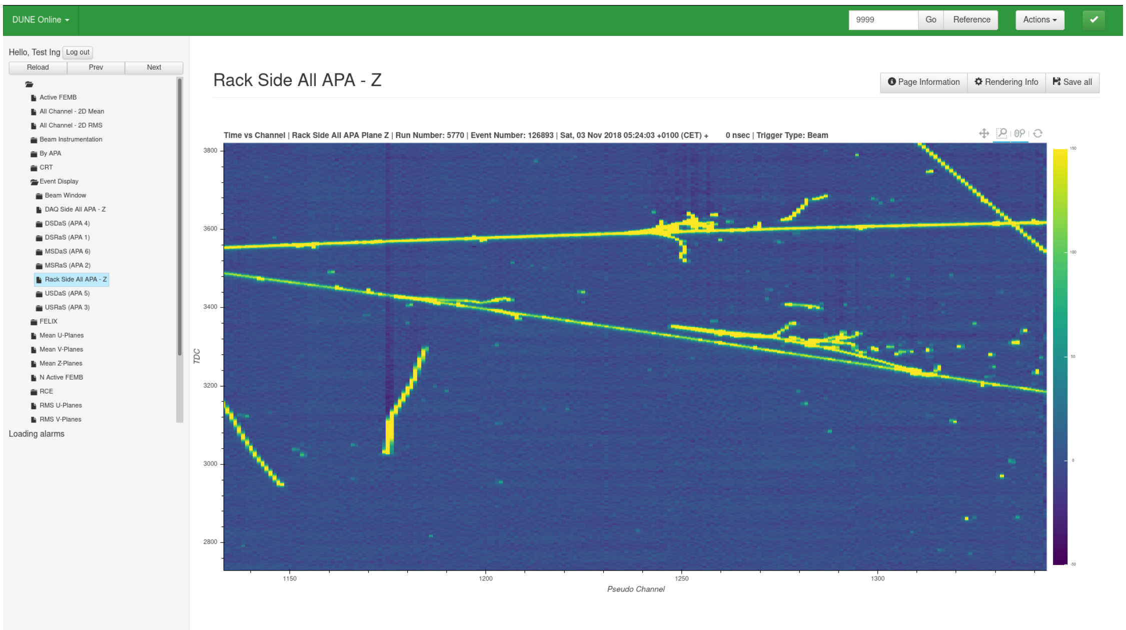
an example of this is shown in Figure 3.22. The histogram database is responsible for the details of each of these pages, this includes the plots on each page, their locations, and the relevant rendering options for each plot. When a new page is requested the display server requests these details from the histogram database before requesting the rendering server to render the plots.

The rendering server is responsible for rendering the plots from the monitoring ROOT files, it is capable of rendering all 1D and 2D ROOT plots. In LHCb's Monet rendering is handled directly on the display server, however in ProtoDUNE–SP an additional rendering server was added. By separating the rendering processes onto their own server additional functionality was possible. The main advantage of the rendering server was that it enabled the OM to utilise a dynamic binning algorithm to scale the resolution of the displayed images in response to user interaction. This is mainly used for the event displays; the event display is initially loaded in a low resolution, but if a user is interested in a specific region of the event display then they can zoom in on the region of interest, which is then displayed at a higher resolution by the dynamic binning algorithm, as demonstrated in Figure 3.23.

The other significant modification to Monet was to modify the data handling to



(a) Full APA view. The white square represents the zoomed region in Figure 3.23b.



(b) Zoomed region.

Figure 3.23: Example of the dynamic binning algorithm in the ProtoDUNE–SP online monitoring system.

significantly improve the efficiency of data transfer to the renderer. When Monet was first adapted for ProtoDUNE–SP it was too slow to render some of the larger plots such as event displays, it would take minutes to render a single event display. By improving the efficiency of the data transfer from ROOT to Bokeh, the rendering time was reduced to around 4 seconds per event display.

In the ProtoDUNE–SP Technical Design Report [54], the proposed online monitoring system was described as being able to provide data quality assurance in real–time, in practice the response time is around one minute. During this time each set of plots will have contain data from 10 events, however, the event displays and TPC FFTs each contain data from one event. TODO: Should I include a discussion of possible improvements?