Status of the Online QA system for the experiment

Ilnur Gabdrakhmanov

VBLHEP, JINR

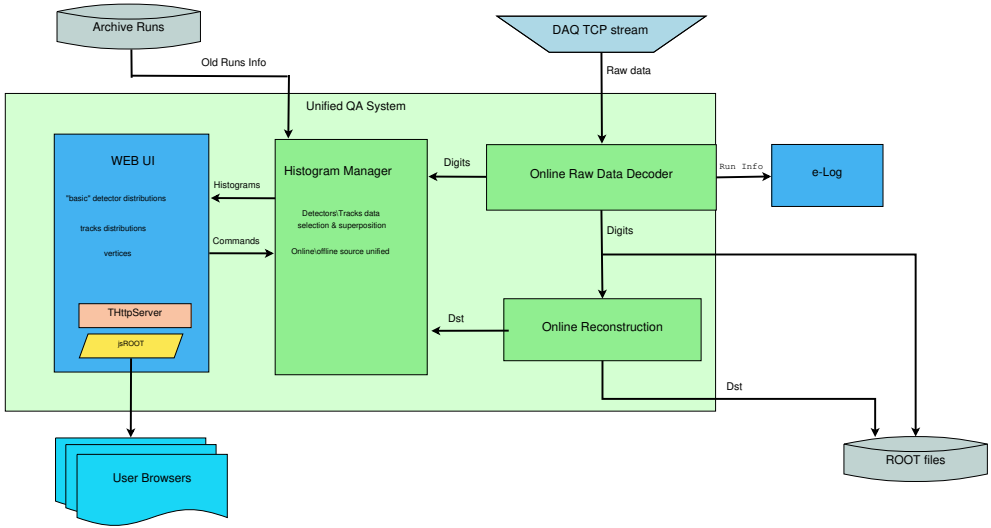
October 2021

8th Collaboration Meeting of the BM@N Experiment at the NICA Facility

Primary tasks (since last run)

- Implement full reconstruction chain inside monitoring workflow
- Make monitoring system flexible in terms of physical setups and histogram sets
- Refactor the system to be more modular and scalable

Preliminary system scheme



Data processing

Status:

- ✓ Data receiving from the DAQ TCP stream is implemented (class BmnOnlineDecoder)
- ✓ Converting and simplified online decoding of the all detector systems raw data is implemented
- ✓ Online reconstruction is partly implemented (class BmnOnlineReco)
- ☐ Refactoring and unification is needed
- ☐ Offline signal noise filtering in the strip detectors directly is inapplicable to the online case

Problem:

The noise reduction in the strip detectors (Silicon/GEM/CSC) implies iterative gathering of pedestal data with consecutive excluding and marking noisy channels.
It is rather slow and causes no valid strip data in the considerable first part of the each run.

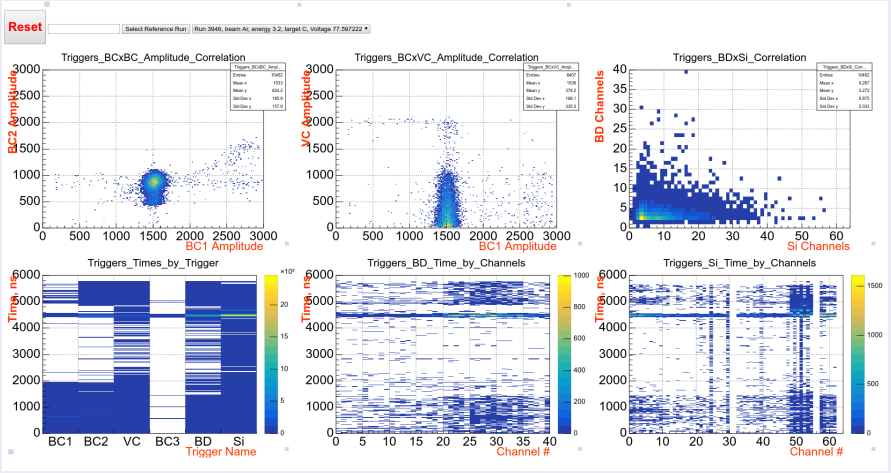
Solution to apply:

Implement "sliding" pedestal set continuously updating during the run and being saved between runs.

Representation part

Interface example with ref run selection options:

Trigger distributions during the 7 period:



QA frontend with hardcoded histograms

Implementation details:

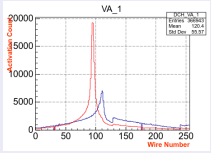
- The data processed and transferred from the previous stage is used to fill ROOT histograms. Which in turn are sent to the end users via http.
- CERN jsROOT library is used to transform the ROOT object to the html histograms.
- Base class for histogram sets BmnHist is used in:
  - BmnHistTrigger
  - BmnHistGem
  - BmnHistToF
  - ... ...

Thus addition of the new detector histogram set is rather simple.

Reference run:

- ✓ Ref run imposition implemented
- ✓ Autoselection of similar runs is implemented
- ☐ Unification is needed

Imposition example:



Fine grain selection:

It is possible to select distributions for specific station/plane/strip

Custom (user defined) histograms

Main objectives:

- ◇ Ability to add histograms during the run without recompile
- ◇ Make addition of histogram simple and flexible
- ◇ If possible make the same for filling logic

Current Implementation status:

- ✓ Class BmnPadGenerator converts structure (from json file) to recursively nested BmnPadBranch objects. Which in turn can be drawn on the canvas.
- □ Web interface for adding histograms
- ? User defined logic for histogram filling

Test code example:

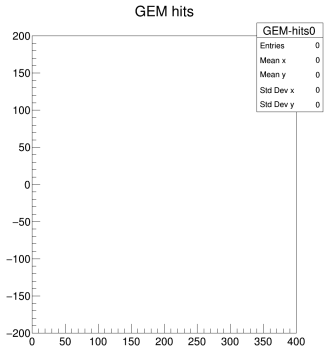
```
BmnPadGenerator *g = new BmnPadGenerator();
g->LoadPTFrom(FileName);
BmnPadBranch * br = g->GetPadBranch();
TCanvas* can = new TCanvas("canHits", "", 1920, 1080);
g->PadTree2Canvas(br, can);
BmnHist::DrawPadTree(br);
```

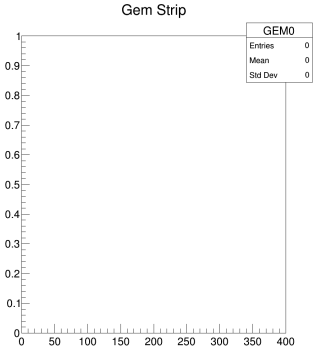
Simple configuration example

JSON scheme:

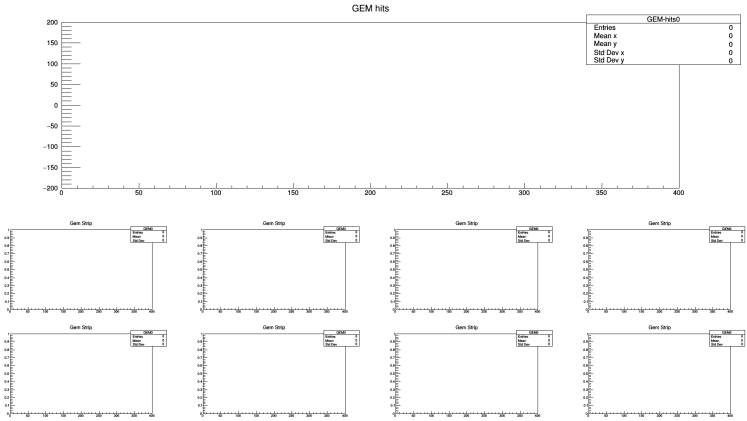
```
{
  "Name": "GEMS",
  "Title": "GEM Canvas",
  "DivX": "2",
  "DivY": "1",
  "Pads": [
    {
      "Class": "TH2I",
      "Name": "GEM-hits0",
      "Title": "GEM hits",
      "Options": "colz",
      "Dimensions": [
        200,
        0,
        400,
        400,
        -200,
        200
      ]
    },
    {
      "Class": "TH1F",
      "Name": "GEM0",
      "Title": "Gem Strip",
      "Dimensions": [
        200,
        0,
        400
      ]
    }
  ]
}
```

Canvas structure:

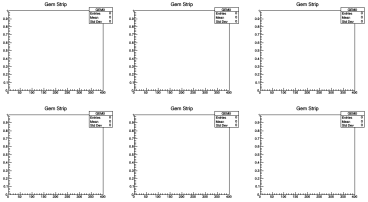


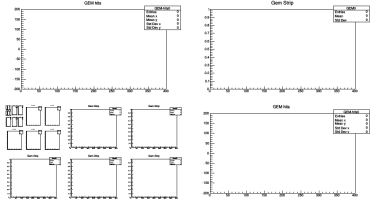


Another configuration example



Another configuration example





Overall Status (green - ready, lime - in progress, gray - not implemented)

Data Processing

Raw Data Decoding

- stream receiving
- decoding(refactoring&unification is needed)
- results transfer

Hits/Tracks reconstruction

- stream receiving
- reconstruction
- results transfer

Histograms manager

Fixed histogram sets

- ref. run load
- ref. run auto selection
- unified composition (Mon/DST)
- addition of DST histograms

Custom histograms

- Generation from config
- Logic description
- Logic implementation

Webpage

- Basic Digit distributions
- Ref. run selection
- Fine grain data selection
- DST histograms
- Custom histogram interface

Question section

Thank you!