QA histograms for TPC online monitoring

ALEXANDER KRYLOV

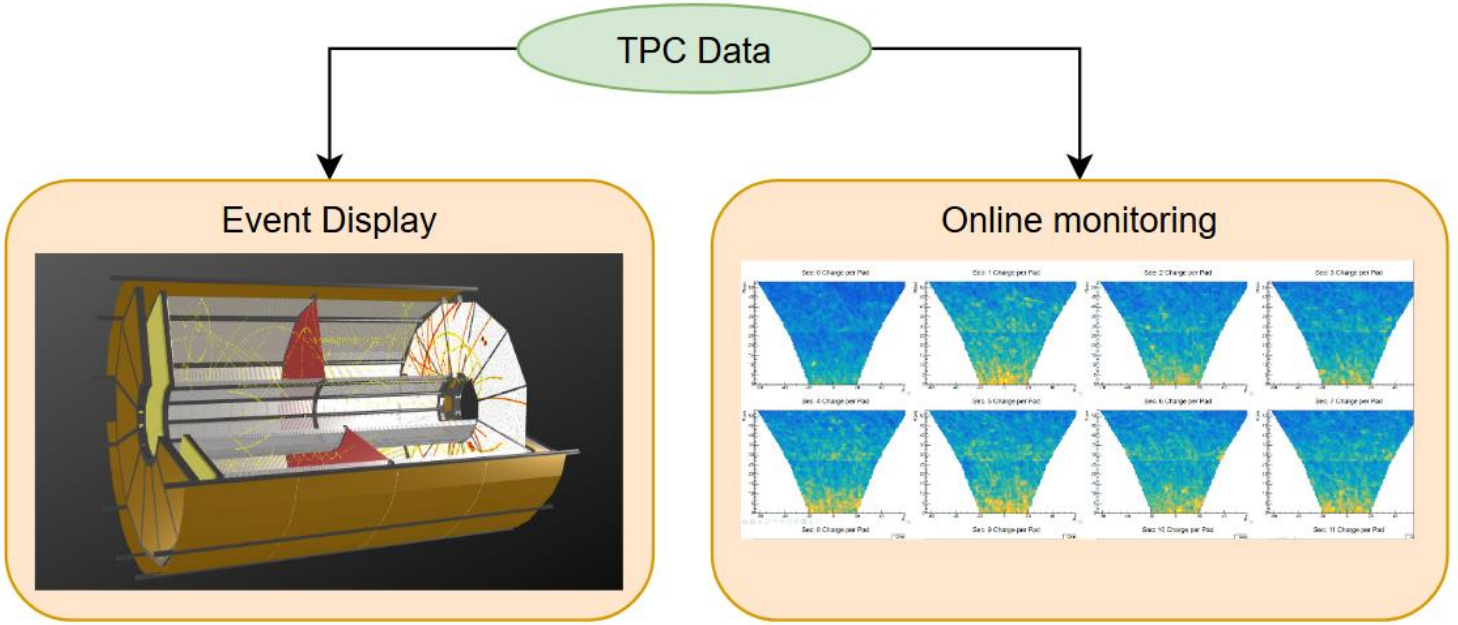
DUBNA/JINR

EMAIL: AVKRYLOV@JINR.RU

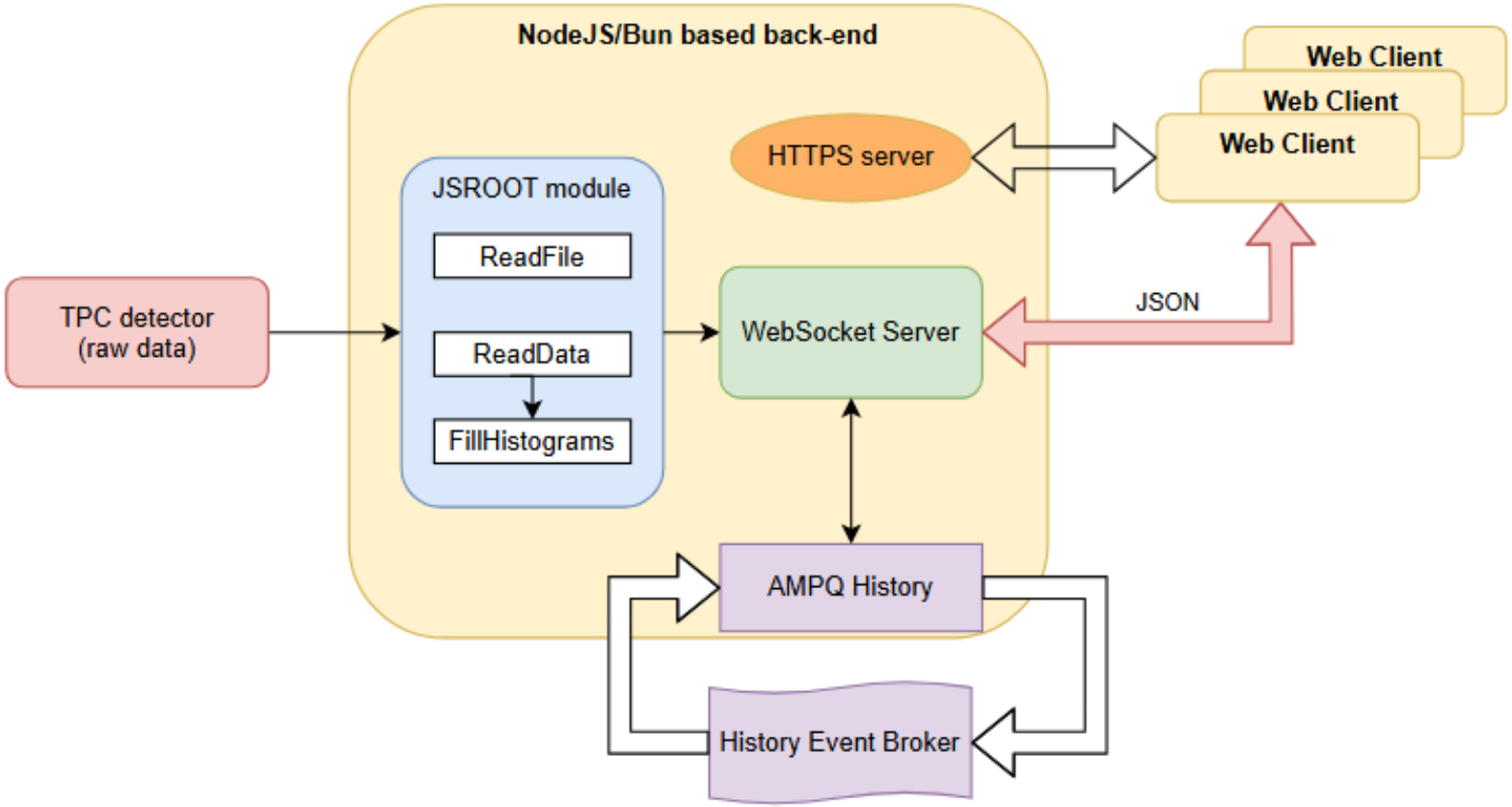
Event display and online monitoring

Event display focuses on visualizing individual events in detail, showing 3D TPC geometry and information after some reconstruction methods.

Online monitoring aims to oversee the overall performance of the experiment and the quality of data being collected in real time.



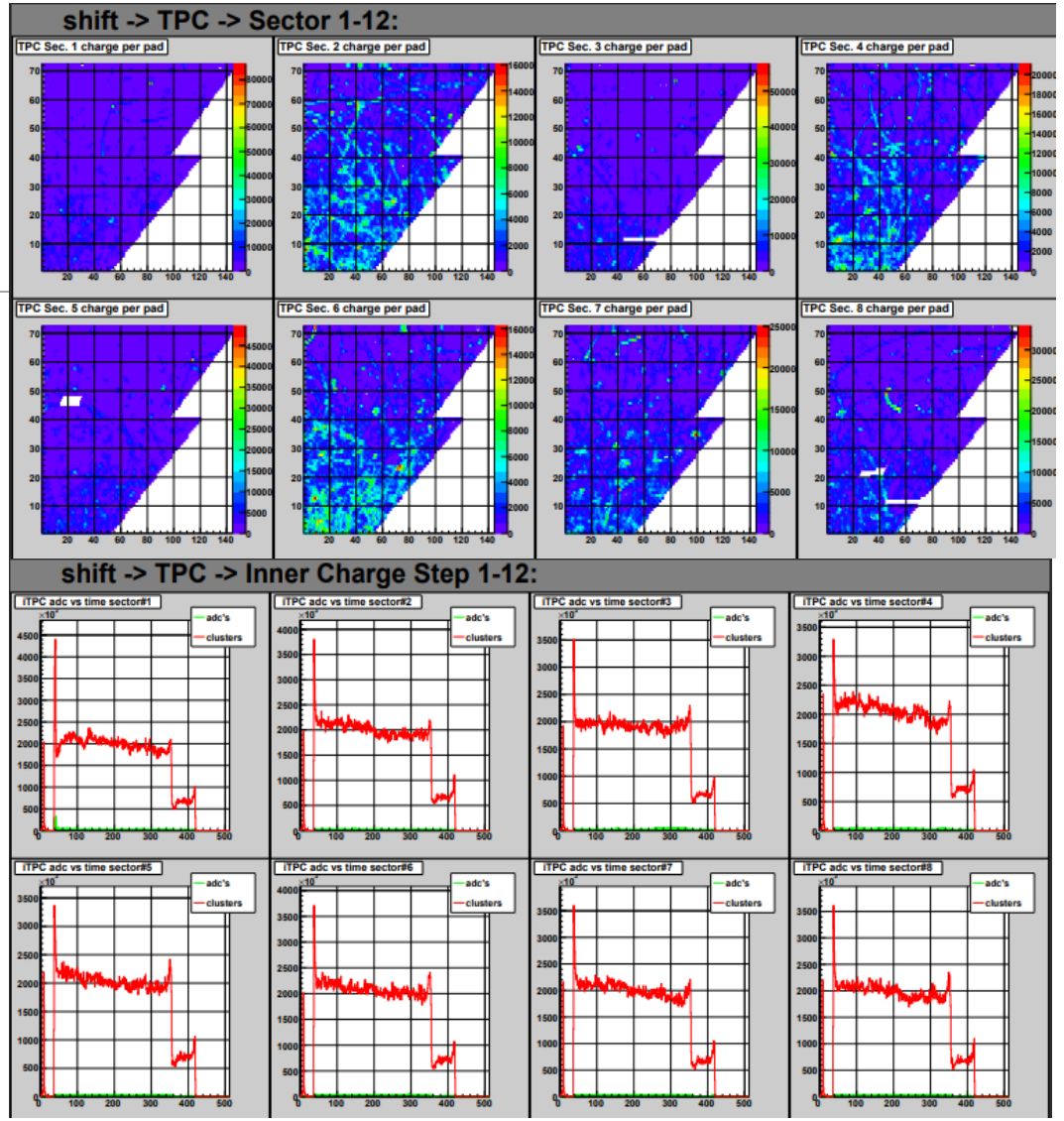
Preliminary TPC monitoring scheme



Quality assurance

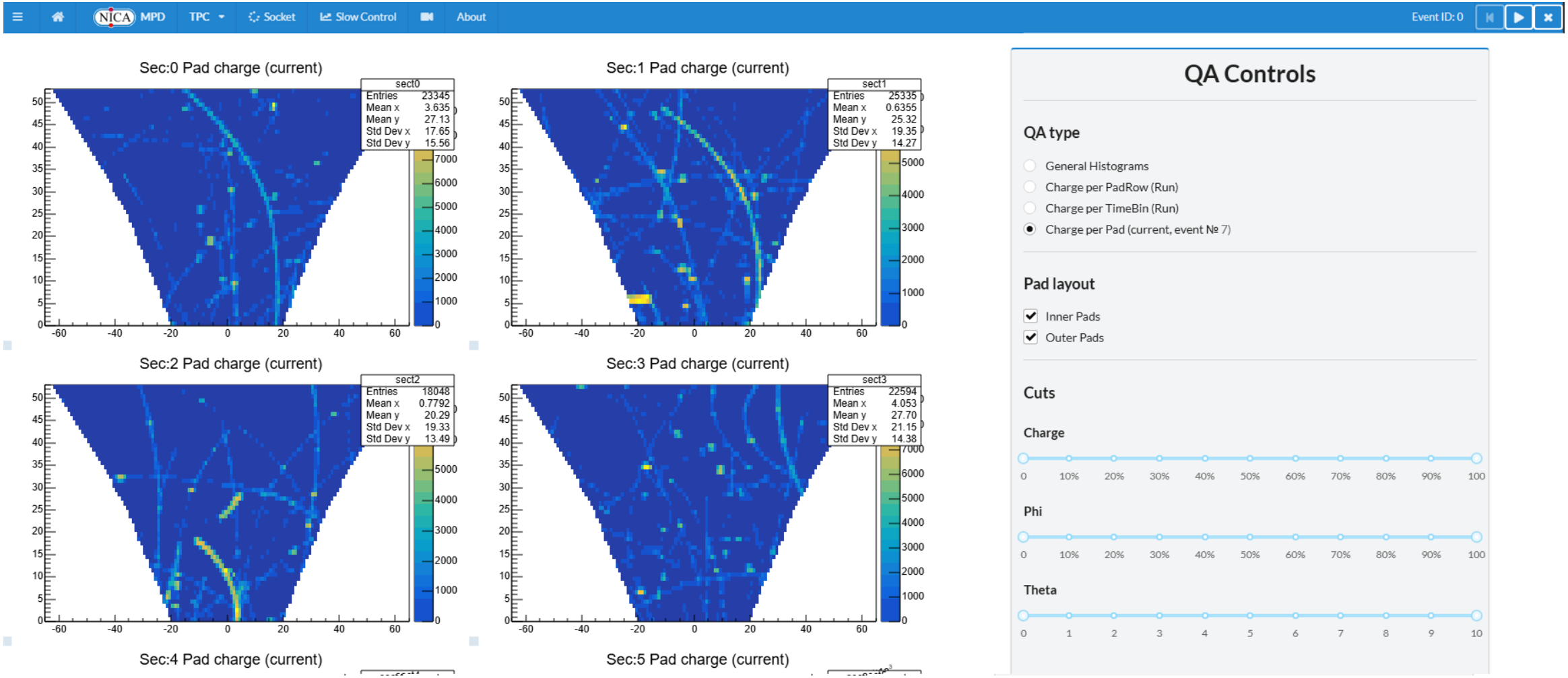
- Inner pads ADC per sector – 24 histograms
- Outer pads ADC per sector – 24 histograms
- Inner pads ADC per timebucket – 24 histograms (per each sector)
- Outer pads ADC per timebucket – 24 histograms (per each sector)
- Inner pads ADC for current event – 24 histograms (per each sector)
- Outer pads ADC for current event – 24 histograms (per each sector)
- General clusters information – 6 histograms

Total number of histograms > 150

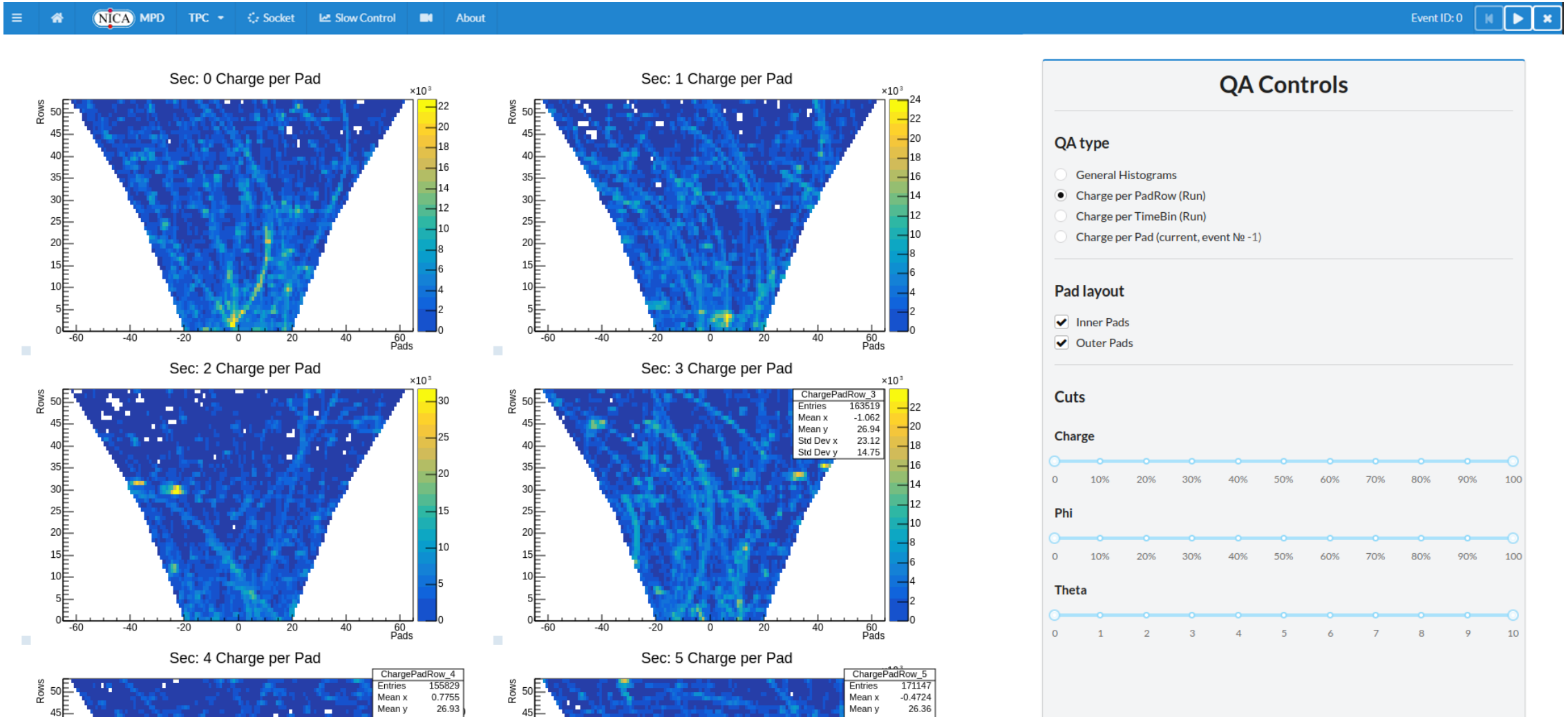


Quality assurance from STAR experiment

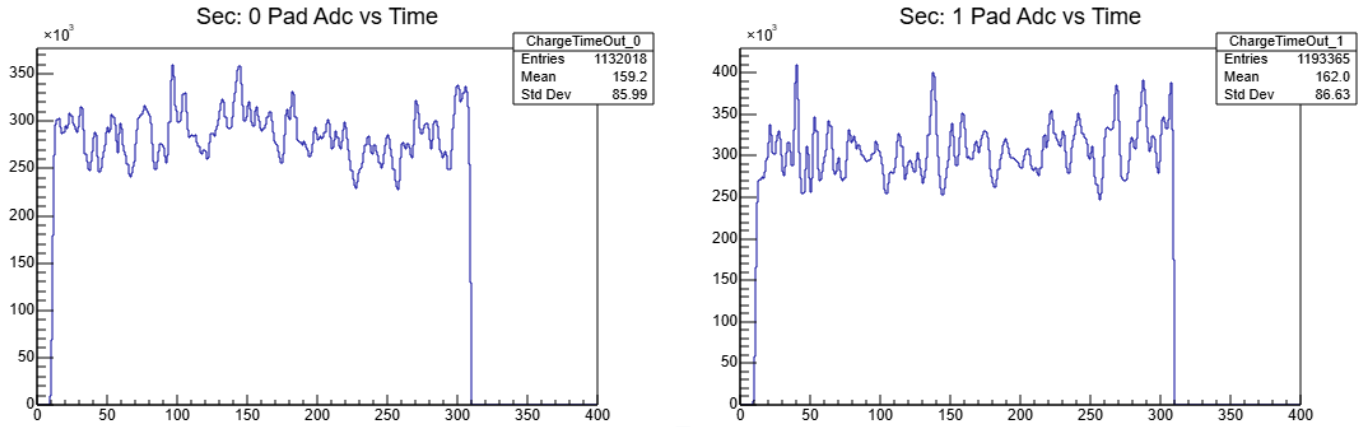
ADC distribution per sector (current event)

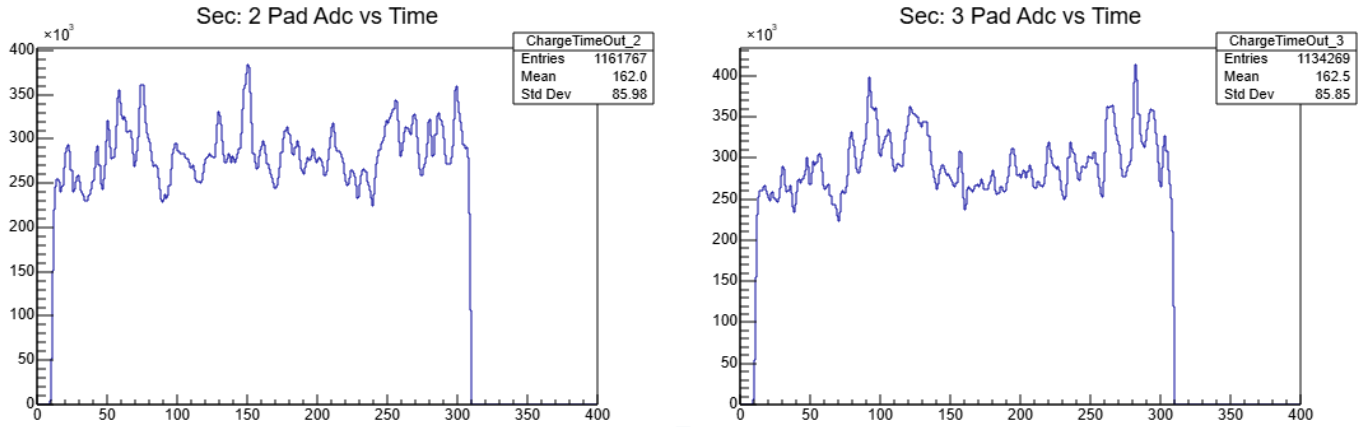


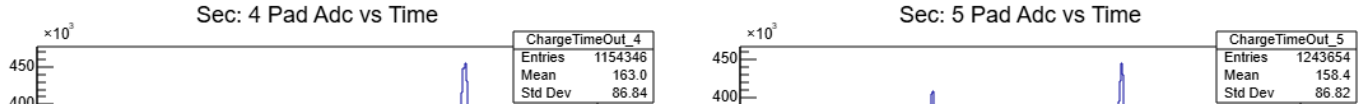
ADC distribution per sector (storage)

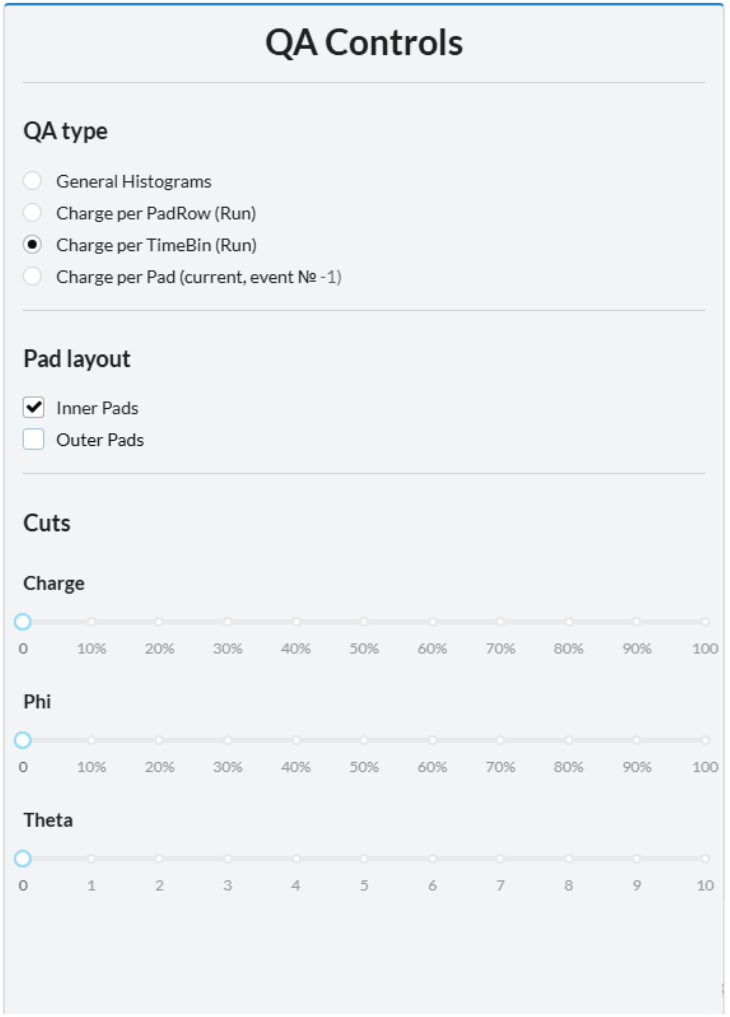


ADC distribution per time (Inner pads)

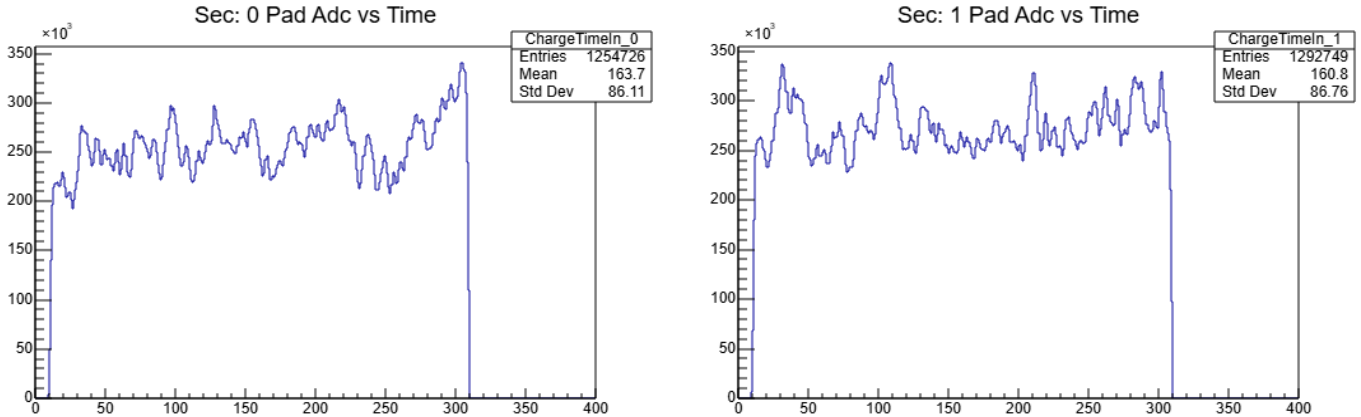


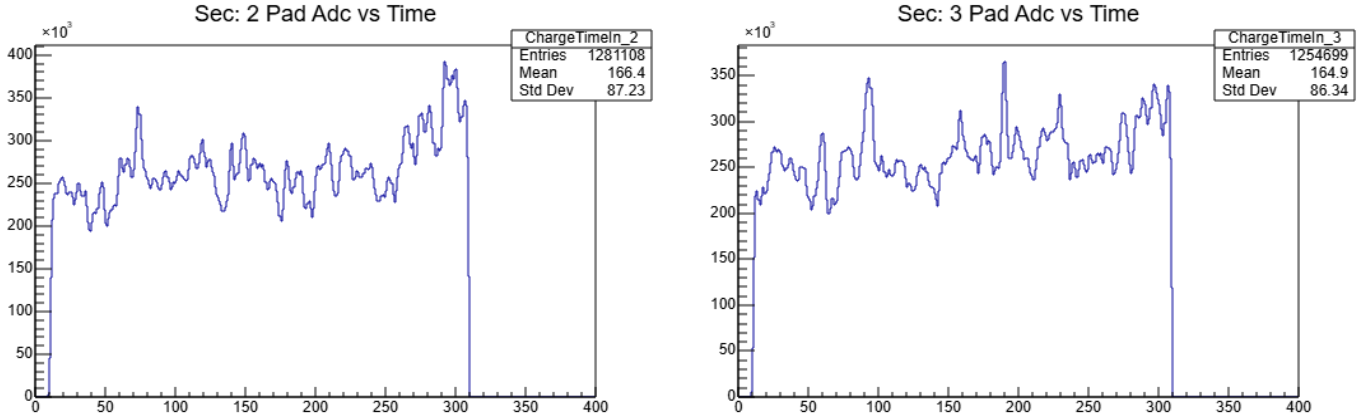


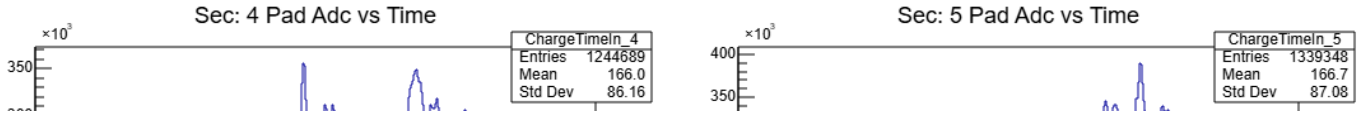


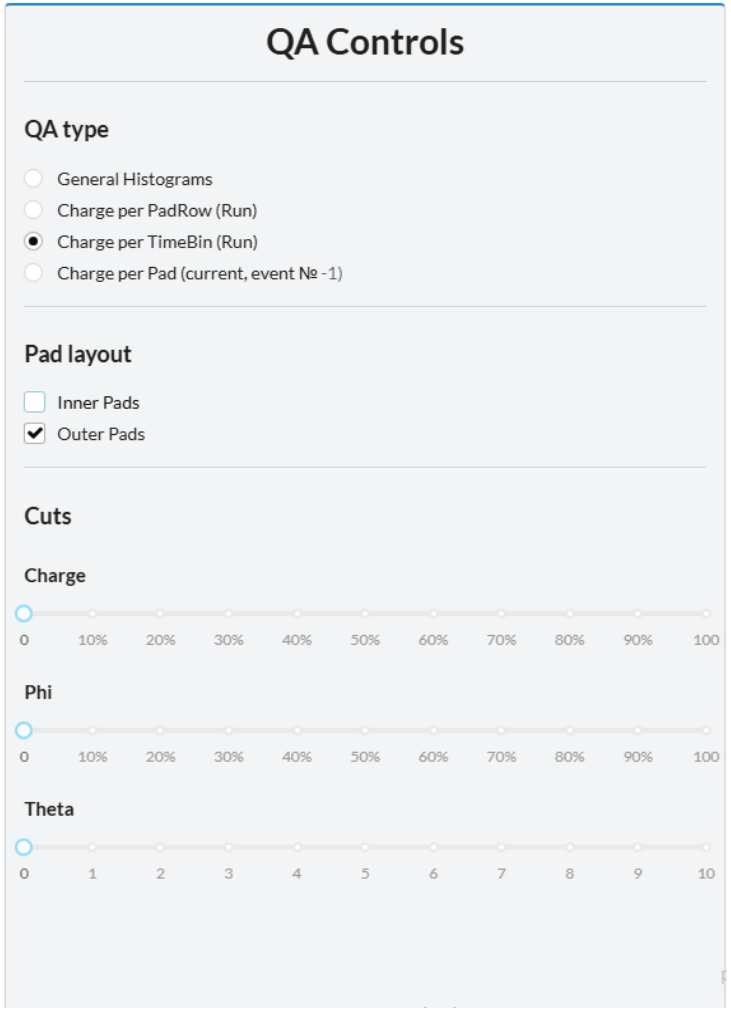


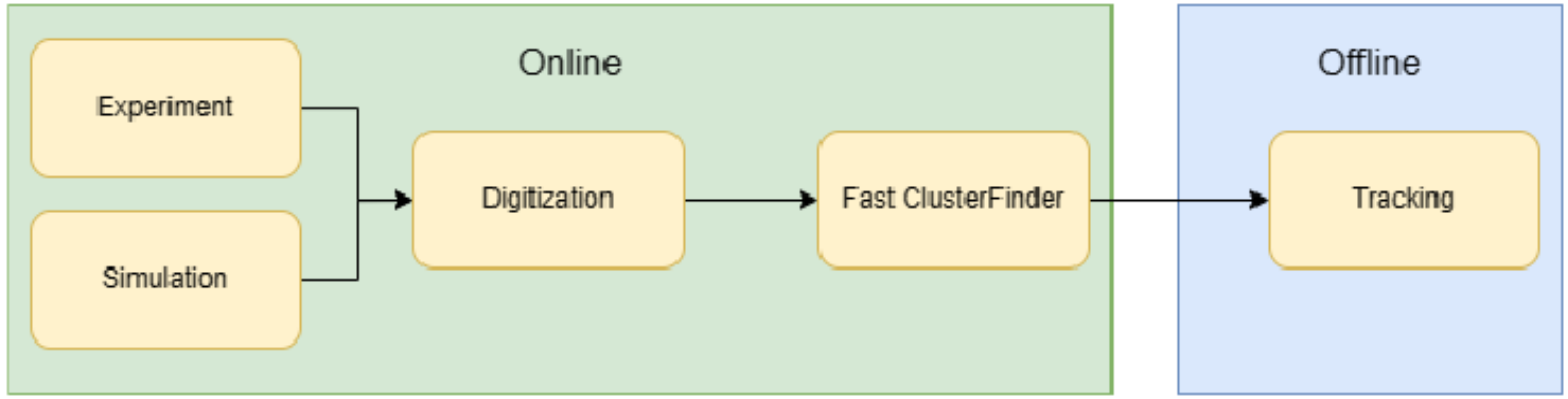
ADC distribution per time (Outer pads)

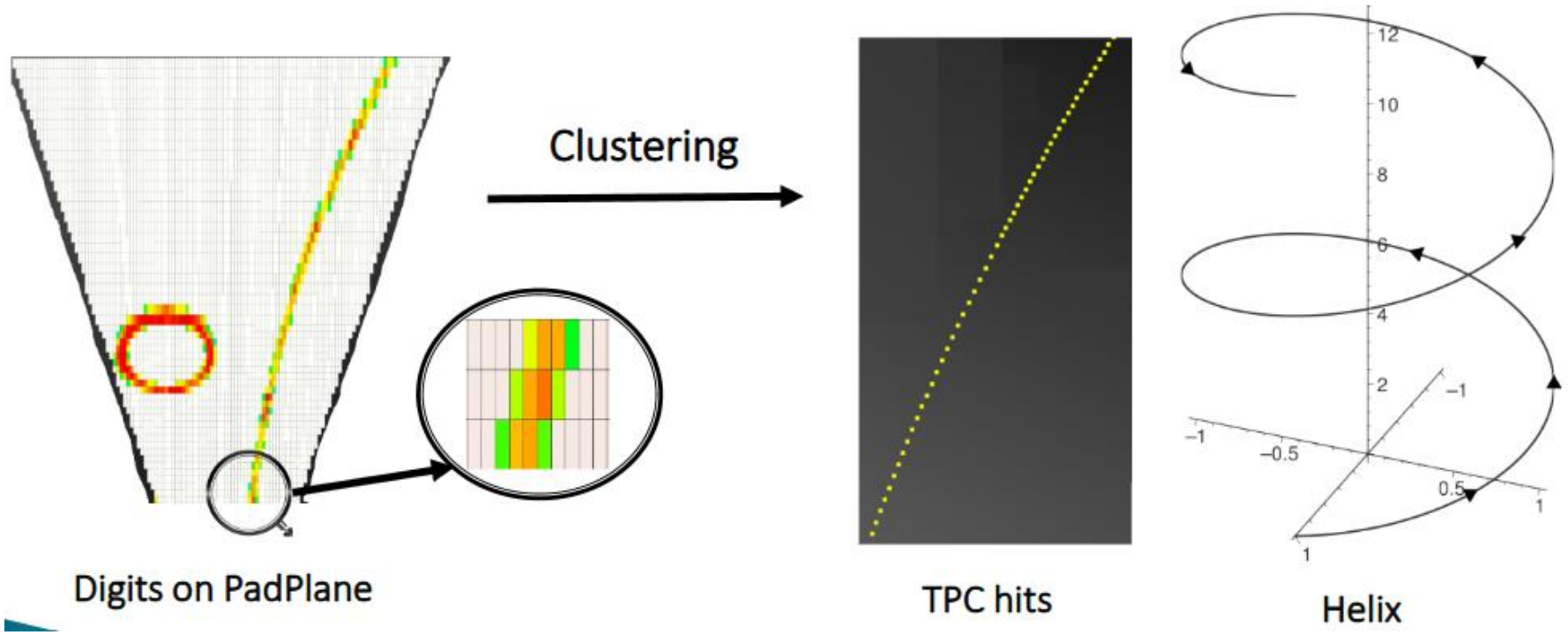




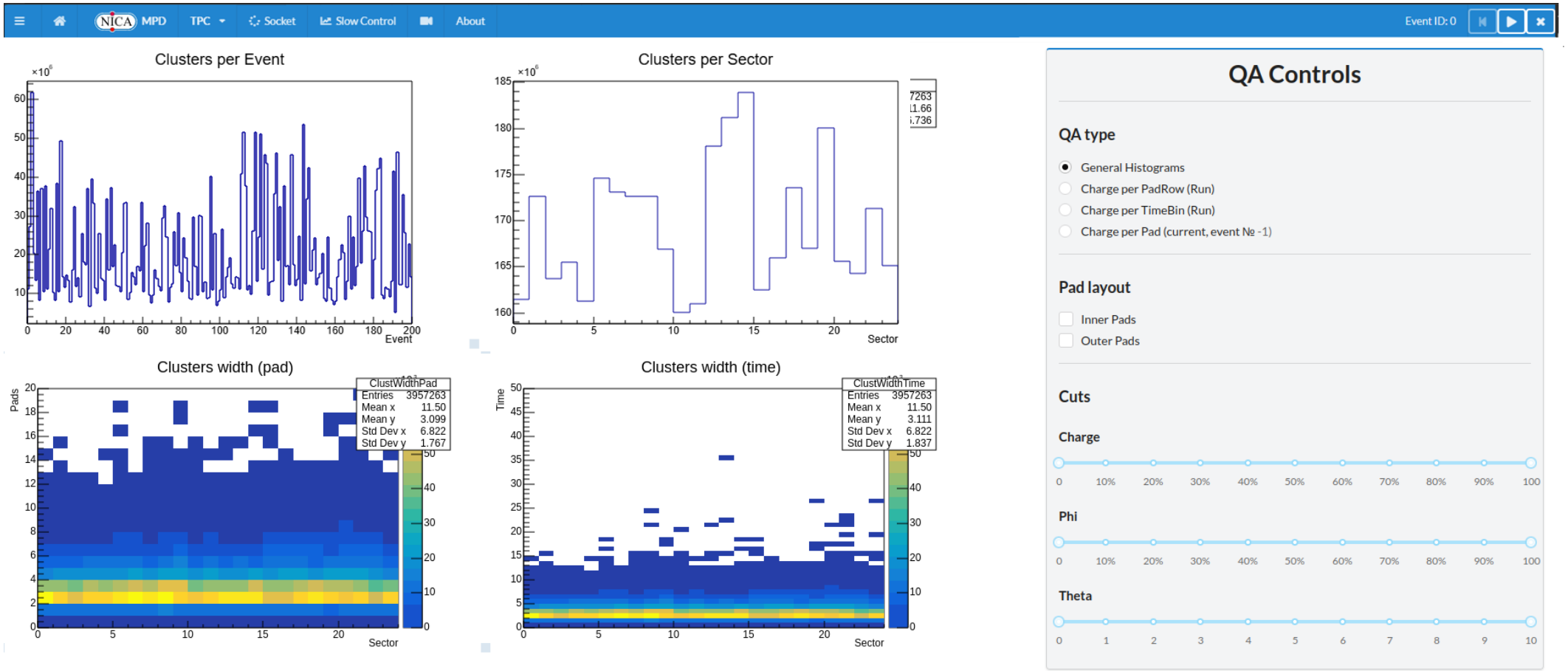





Online
Experiment
Simulation
Digitization
Fast ClusterFinder
Offline
Tracking


Clustering
Digits on PadPlane
TPC hits
Helix

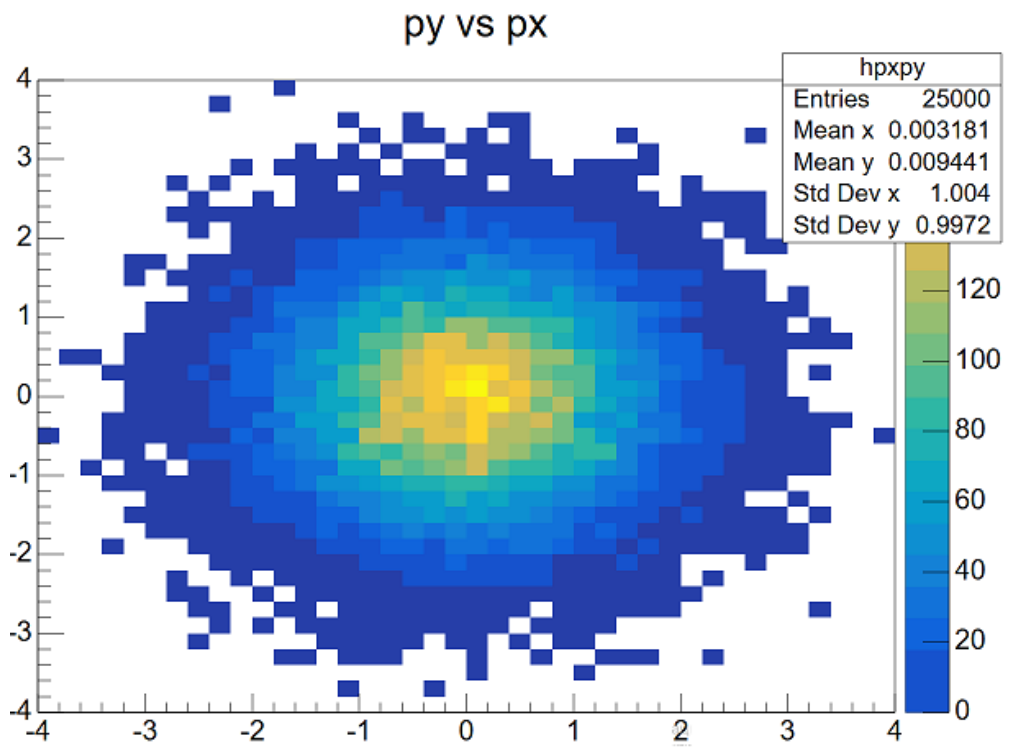
Clusters info



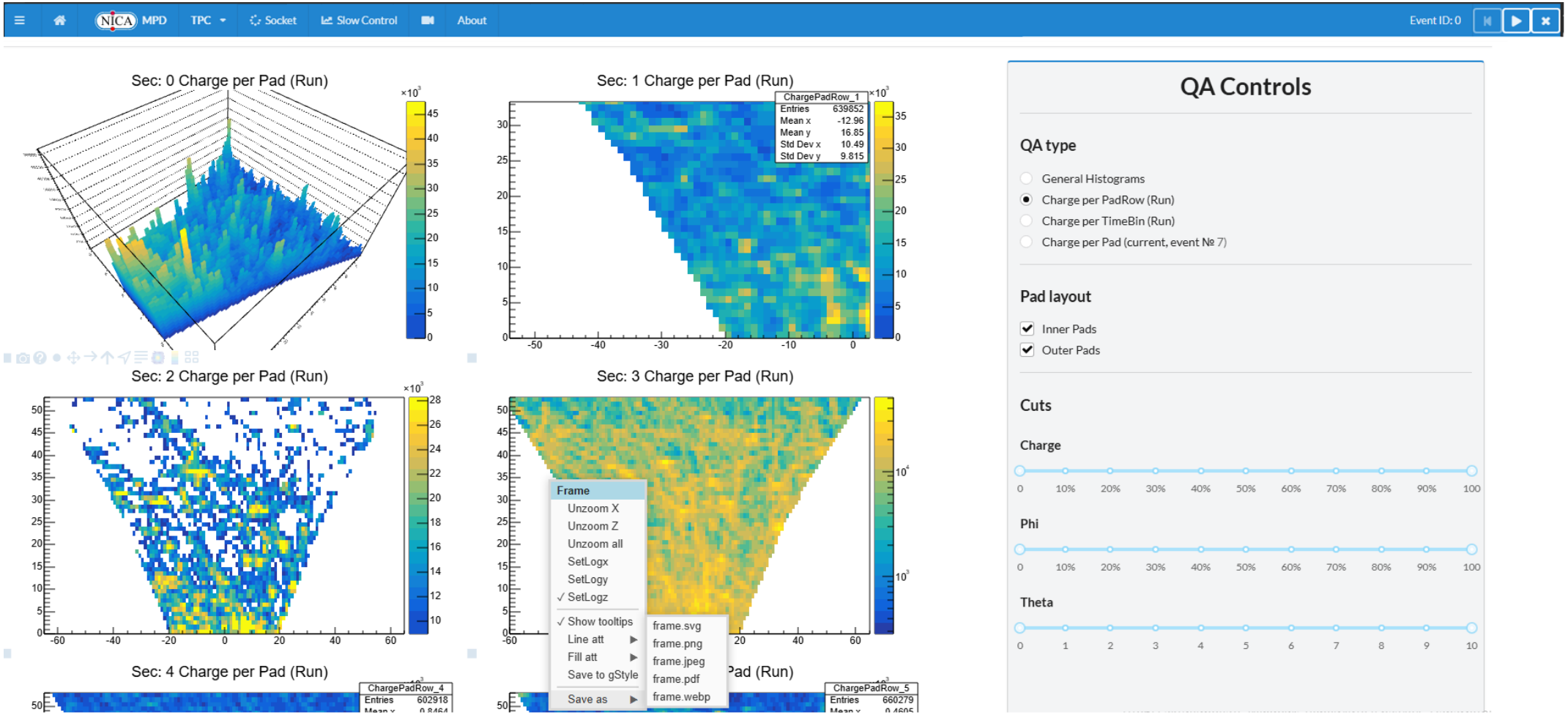
JSROOT library

The JSROOT library allows:

- reading of binary and JSON ROOT files in JavaScript/TypeScript;
- drawing of different ROOT classes in web browsers;
- reading and drawing TTree data;
- interacting with histograms and graphs
- using in node.js.



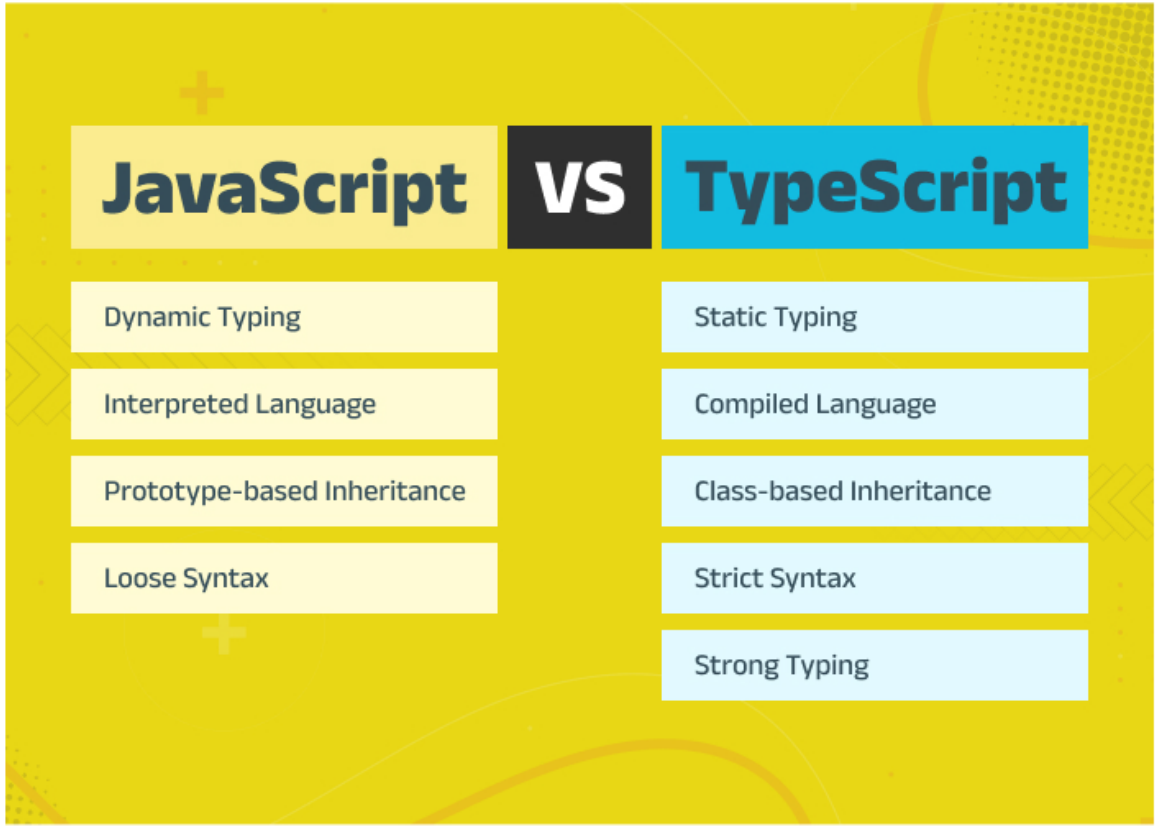
JSROOT possibilities



New Technologies

	New (ED + QA)	Old (Event Display)
FrontEnd	React + TypeScript	React + JavaScript
BackEnd	Bun	NodeJS
Server	Elysia	Express
Store	Modern Redux	Redux
Bundler	Vite	WebPack
Connection	WebSocket	WebSocket
Graphics	ThreeJS + JSROOT	ThreeJS

TypeScript vs JavaScript



Functional components (React)

Simplicity: They are just a function, which makes them simpler to write and understand. There's no need to worry about **this** keyword, which can confuse some developers.

Performance: Functional components can be slightly more efficient than class components. They avoid the overhead of class objects and are more accessible for React to optimize.

Ease of Testing: Functional components are generally easier to test since they are just functions without side effects.

Conciseness: They often result in less code, making your codebase cleaner and easier to maintain.

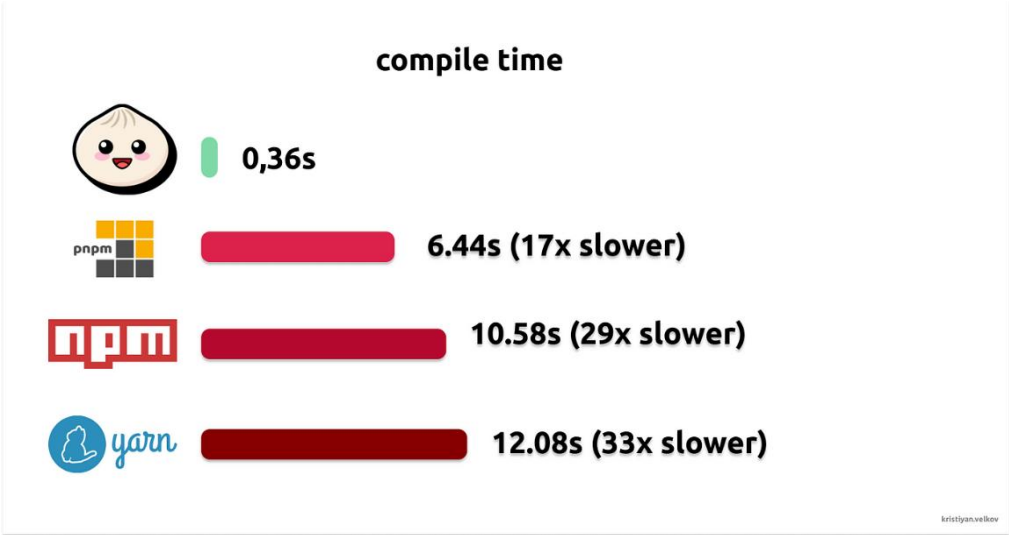
Hooks!!!: With the introduction of React Hooks, functional components can now manage state, side effects, and more, which were previously only possible in class components.

Bun

Bun is a new JS runtime focused on performance and being all-in-one runtime, bundler, package manager and transpiler.

Speed. Bun starts fast and runs fast. It extends JavaScriptCore, the performance-minded JS engine built for Safari.

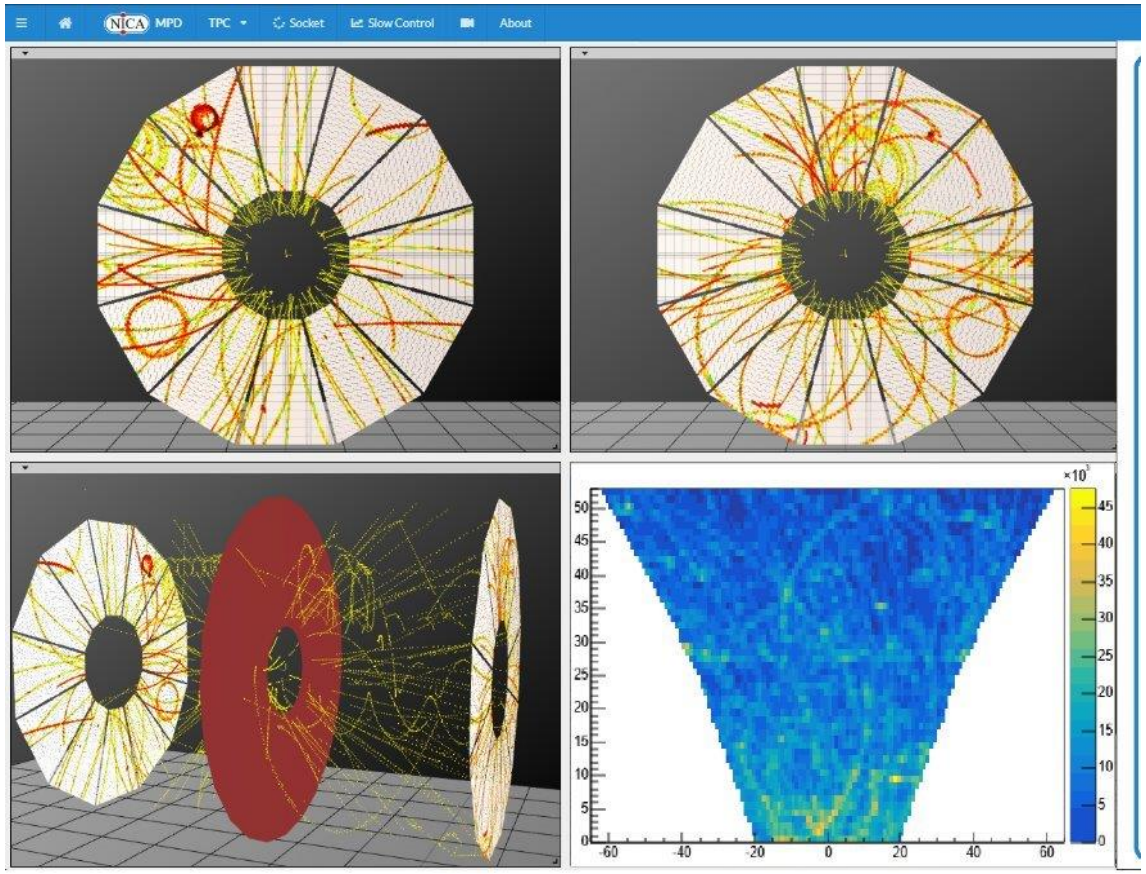
APIs. Bun provides a minimal set of highly-optimized APIs for performing common tasks, like starting an HTTP server and writing files.



Conslusion

Online monitoring as well as event display are one of the most important tools to control quality of data during an experiment.

Its combination ensures that not only data collection process efficient and reliable, but the data itself is rich and informative.



Thank you for your attention!