Development of Web Applications for SRILAC Operation

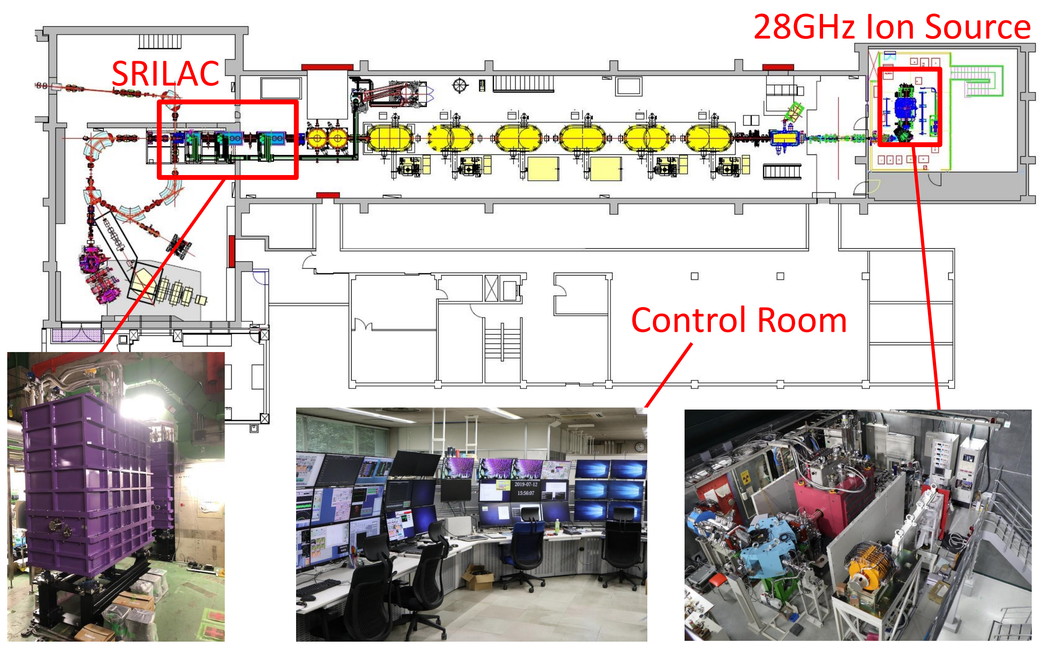
K.Kaneko [A)] A.Uchiyama [B)]
[A)]SHI Accelerator Service Ltd. [B)]RIKEN Nishina Center

Abstract

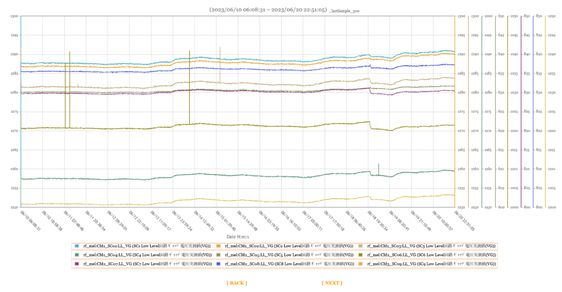
At the RIKEN Nishina Center, the superconducting RIKEN Linear Accelerator (SRILAC) project was started in 2019. SRILAC was added as a booster to the existing normal-conducting Linac. In addition, the new superconducting ion source (28-GHz SC ECR) was installed, and experiments to search for superheavy elements are currently in progress.

The EPICS archiver appliance (AA) was installed as a new epics archiver when SRILAC was installed. The second archiver was added later, currently AA1 archiving about 20,000 records and AA2 archiving more than 200,000 records.

For these reasons, increased the number of various parameters related to SRILAC and the number of monitored MPS signals, which has raised issues for the current operation tools, especially the chart application (COMS Chart) and MPS display (developed in CS-studio and MEDM).



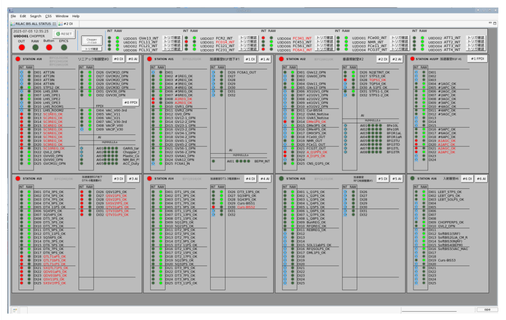
■COMS Chart
・WEB Chart used until now. Developed using jqPlot (jQuery Plugin).
・jqPlot is somewhat old plug-in, and development flexibility of the graph section is limited. So difficult to update in line to operator requests.



-Bad point
・Max 9 series (operator need 10).
・Low development flexibility in developing the graph section.

-Good point
・Works with older browser.
・Capable of displaying non-AA data (MYDAQ2, RIBFCAS).

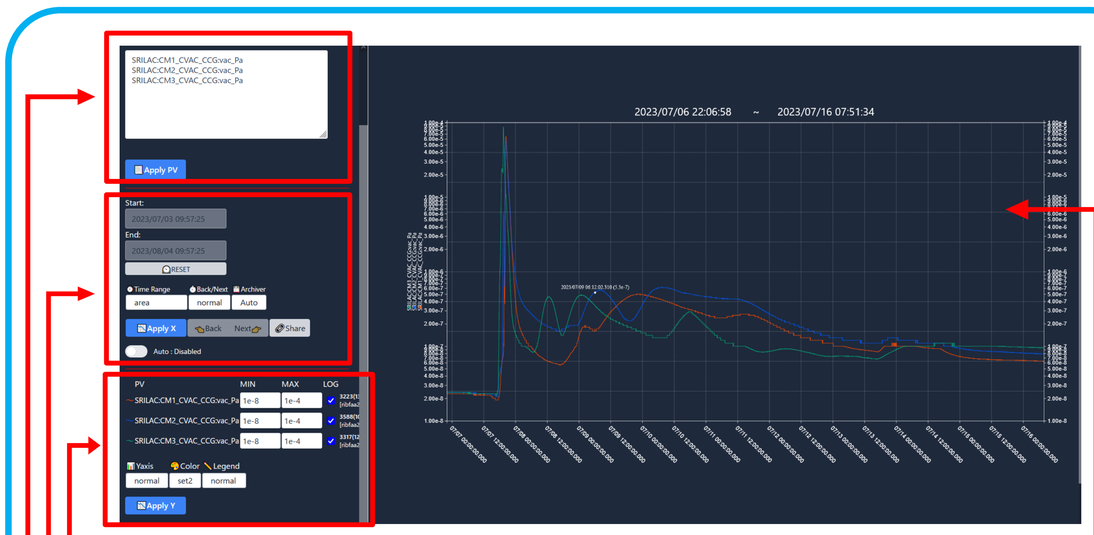
■Displaying MPS monitor
・MPS system is built with PLC, controlled and monitored by EPICS.
・Information is monitored by GUI such as CS-Studio and MEDM.



-Bad point
・It's hard to see because the BITs are just displayed.
・Some MPS signals are automatically triggered and then deactivated, so it may not be possible to confirm which BIT has been activated.
・It is sometimes difficult to confirm the order in which the MPS signal is activated when multiple MPS signals are activated at one time.

< Develop new charting applications >

We chose to visualize data using AA's Web API + D3.JS (https://d3js.org/).
Why D3.JS ? Highly flexible, development , popular, rich apps can be created (The framework uses React + TypeScript).
Since most of the data related to SRILAC and data added in recent years are stored in AA, we developed a charting application specialized for AA (EPICS archivers such as MYDAQ2 and RIBFCAS have been in operation before the installation of AA).



・Paste the PVs you want to display (up to 10).

・Data acquisition within a specified time range.
・"Share Button" to copy the current graph information to the clipboard. And paste in chat to share graphs with team.
・Automatic update enabled, data acquisition every 30 seconds (equipped with trend graph mode).

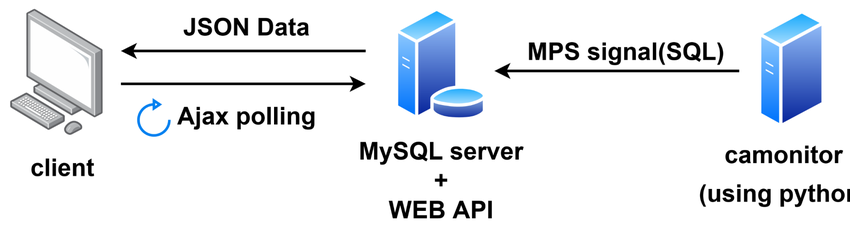
・Y-axis supports log, exponential notation, etc.
・When there are axes in the same range, they are combined into one.
・Line color of graphs can be changed.

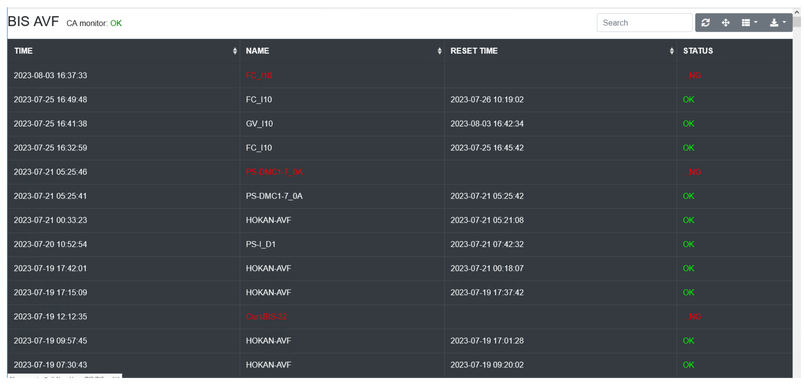
・Use various browser events such as mouse dragging, over, click, etc. So smoothly zoom in on graphs and check values.
・Compatible with ES2015 or later browsers.
・Dark mode available.

< Web App for displaying MPS operating history >

BIT information is stored in RDB using camonitor + MySQL and displayed in a browser. Camonitor is written in python (CaChannel) and the data stored in RDB can be retrieved via WEB API (returns JSON).
Using the similar mechanism, the alarm history was also made into a web application. This is used in combination with CS-Studio's Embedded browser and script.

- For MPS monitoring -





・Real-time updates with Ajax polling.

・Sortable by time, status, etc. Clearly identifies the time of the alarm and reset.

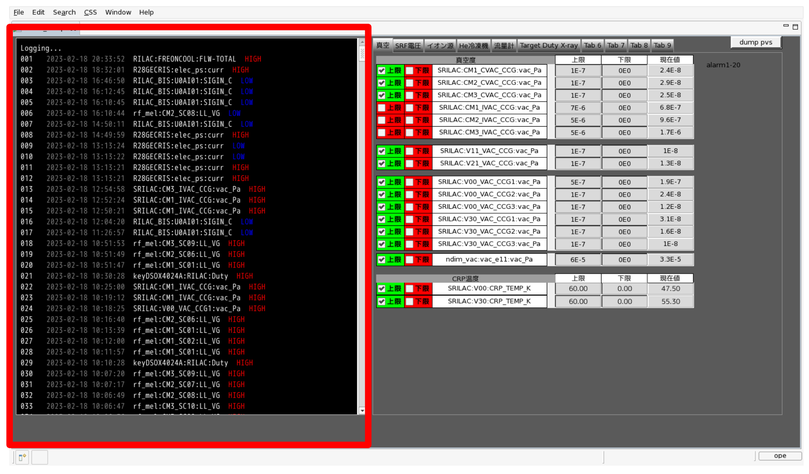
・This app is also used in other accelerators at the Nishina Center.

- For SRILAC alarm -

・CS-Studio's Embedded browser is used to display the history of alarms stored in the RDB.

・Browser Information is updated when an alarm is triggered. Alarms are turned on and off from CS-Studio.

・Use Embedded Script to sound an alarm.



Future Development

■ Update functionality to chart application. ⇒ Display PV descriptions, display non-AA data (RIBFCAS, MYDAQ2, etc.).
■ Problems of SRILAC alarms. (To add or remove alarm targets, the CS-studio and camonitor must be edited.)
⇒ In the future, we aim to develop the new alarm system that can be completed in a browser using WebSocket and other technologies.