Application Development for Archiving Data and Operating Status Analysis at KOMAC

KAERI

Sung-yun Cho#, Jae-Ha Kim, Young-Gi Song, Jeong-Jeung Dang

Korea Multipurpose Accelerator Complex, Korea Atomic Energy Research Institute., Gyeongju, Korea

* This work has been supported through KOMAC (Korea of Multi-purpose Accelerator Complex) operation fund of KAERI by MSIP (Ministry of Science, ICT and Future Planning # sungyun@kaeri.re.kr

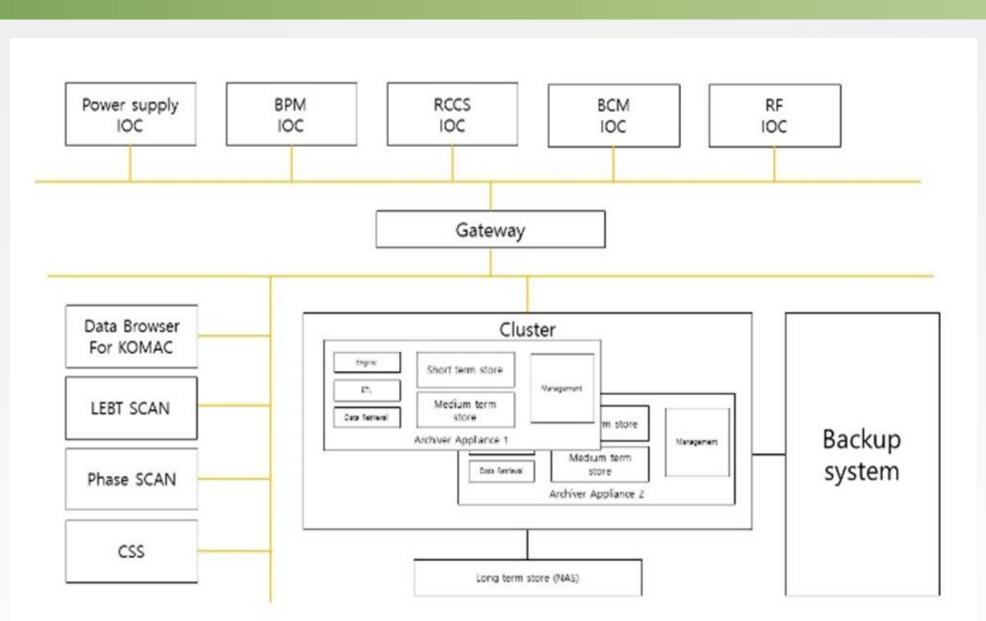
KOMAC and Archiver Appliance System

KOMAC Control System

- 100MeV, 20mA (Peak current) proton linac accelerator
- Based on Experimental Physics and Industrial Control System (EPICS)
- process variable : 18271 ea
- archiving variable : 1100 ea
- Timing system provides pulse until 60Hz repetition that's maximum repetition

Environment of KOMAC Archiver Appliance System

- Centos7 64bit- Storage : 3.5 TB
- Apache-tomcat 7.0
- provide Archiver Appliance Application Programming Interface (API) for external program



Home Reports Metrics Storage Appliances Integration Help This is the archiver appliance management console for the KOMAC archiver. Please contact Jingchen Zhou for any questions regarding these archiver appliances. For support please contact Mural Strankar at 650 xxx xxxx or Bob Hall at 650 xxx xxxx. To check the status of or to archive some PV's, please type in some PV names here. Check Status Archive Archive (specify sampling period) Lookup Pause Resume Check Status Archive Archive (specify sampling period) Lookup Pause Resume PV Name Status Appliance Connected? Monitored? Sampling period Last event Details Quick chart true 1.0 107/12/2020 14.08.13 +09.00 1 107/12/2020 14.08.13 +

Archiver Appliance API

- The data is requested by the URL which includes the information of server address, protocol, process variable name, time
- The received data show the information in JSON format (it is chosen between the CSV, MAT, TXT, JSON, etc)

The Development of Archiving Data Analysis Application

Development Environment

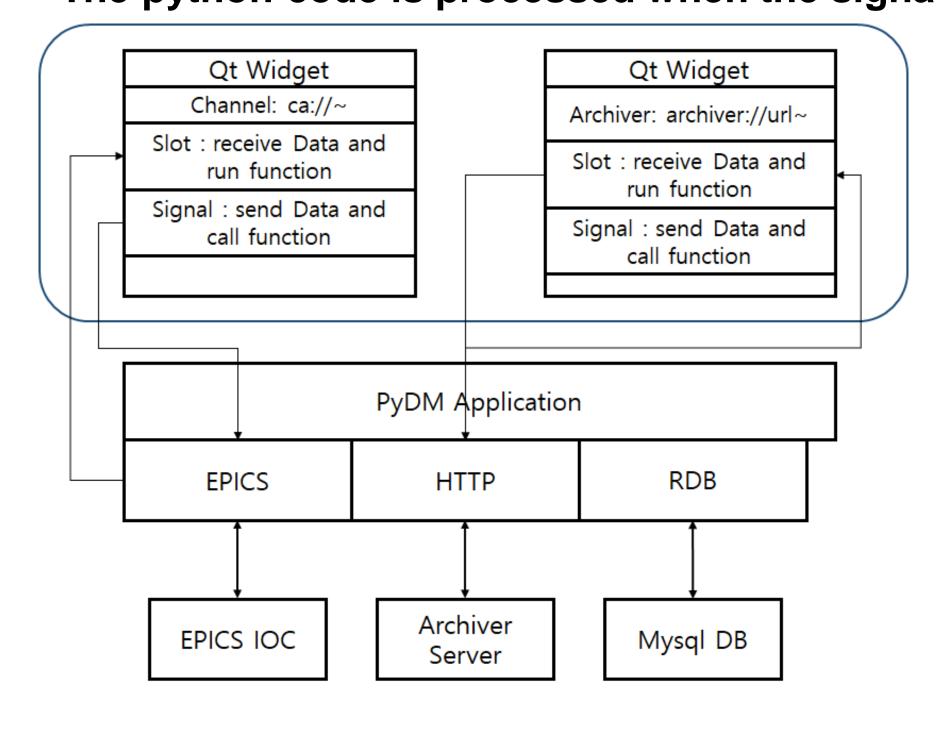
- Centos7 64bit
- Python3.6 (anaconda- environment), it includes the libraries that are pydm, pyepics, pyqt, numpy, etc
- qt 5.12.5 (it includes the qt-designer for UI development)

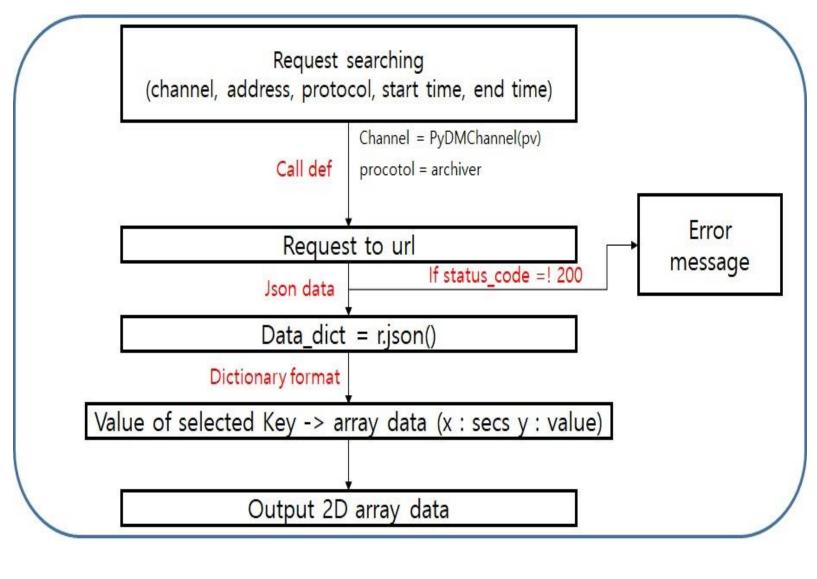
The Application based on PyDM

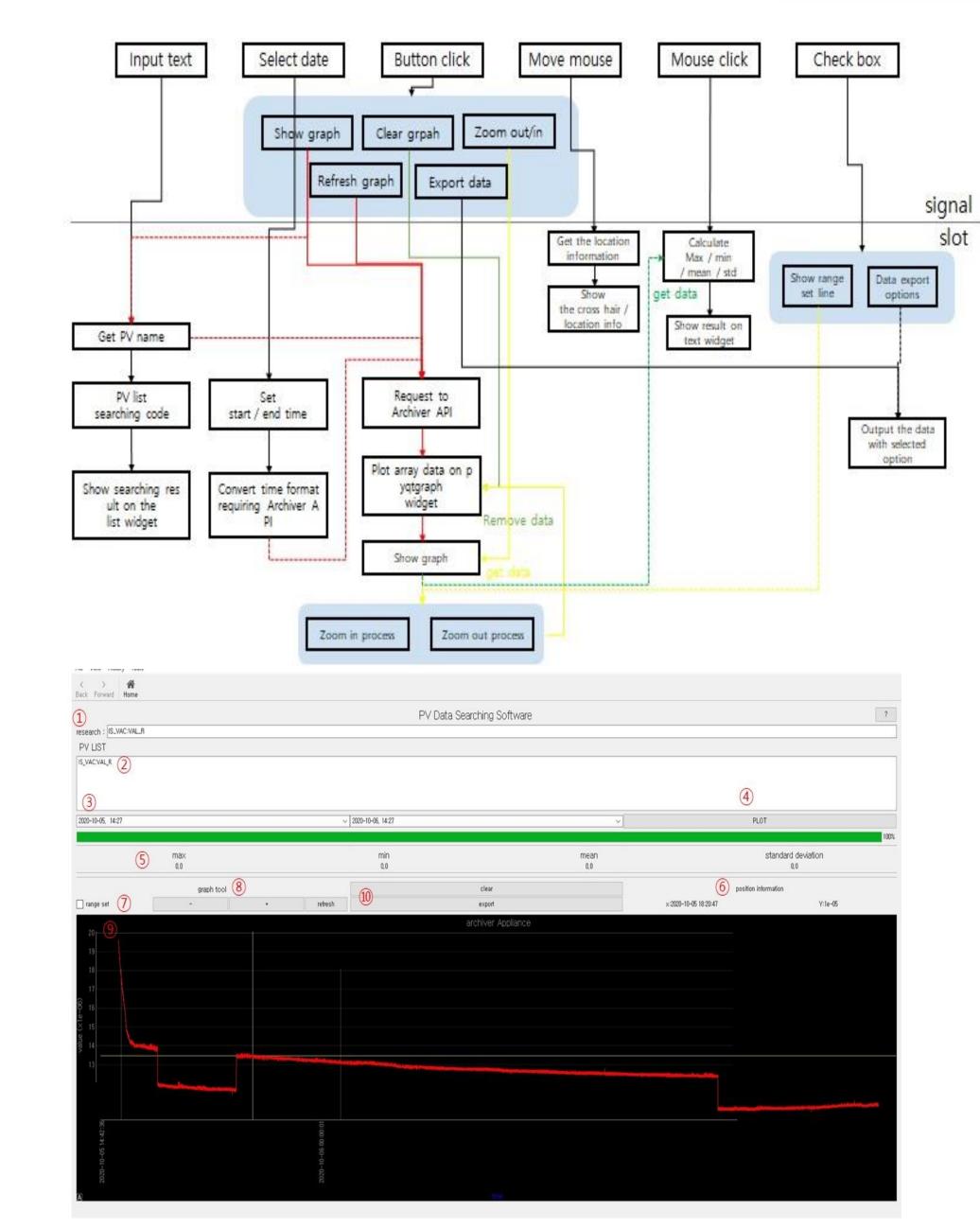
- The PyDM is based on the PyQT framework that's a python binding of the cross-platform GUI toolkit
- It have a function that's a data plugin such as channel access protocol and archiver appliance based on pyepics python library
- The qt-designer is a graphic design tool which provides the function included fast and convenient development

The method of the data requesting

- The data has a JSON format which received by Archiver Appliance API
- The python code is processed when the signal event break out



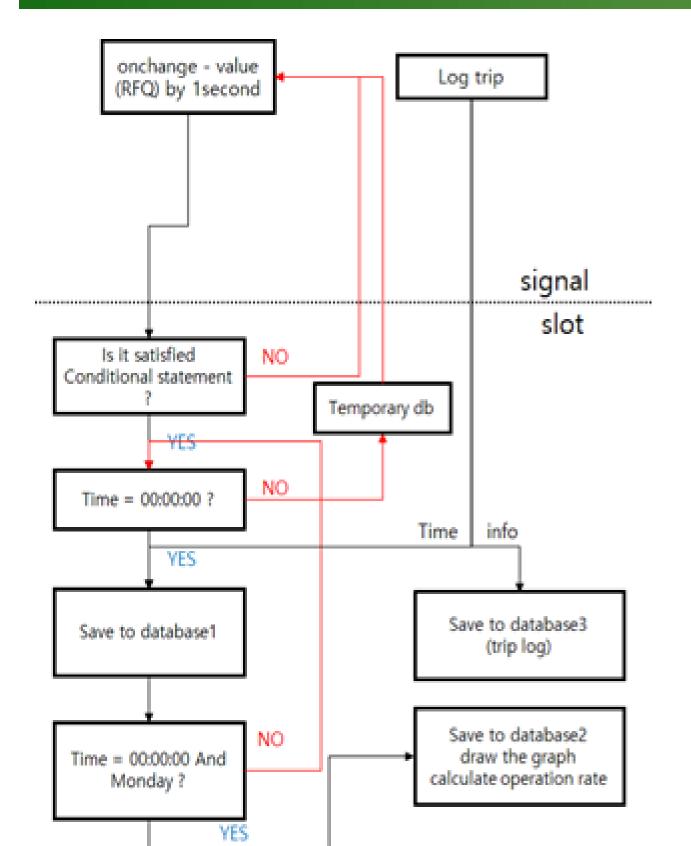


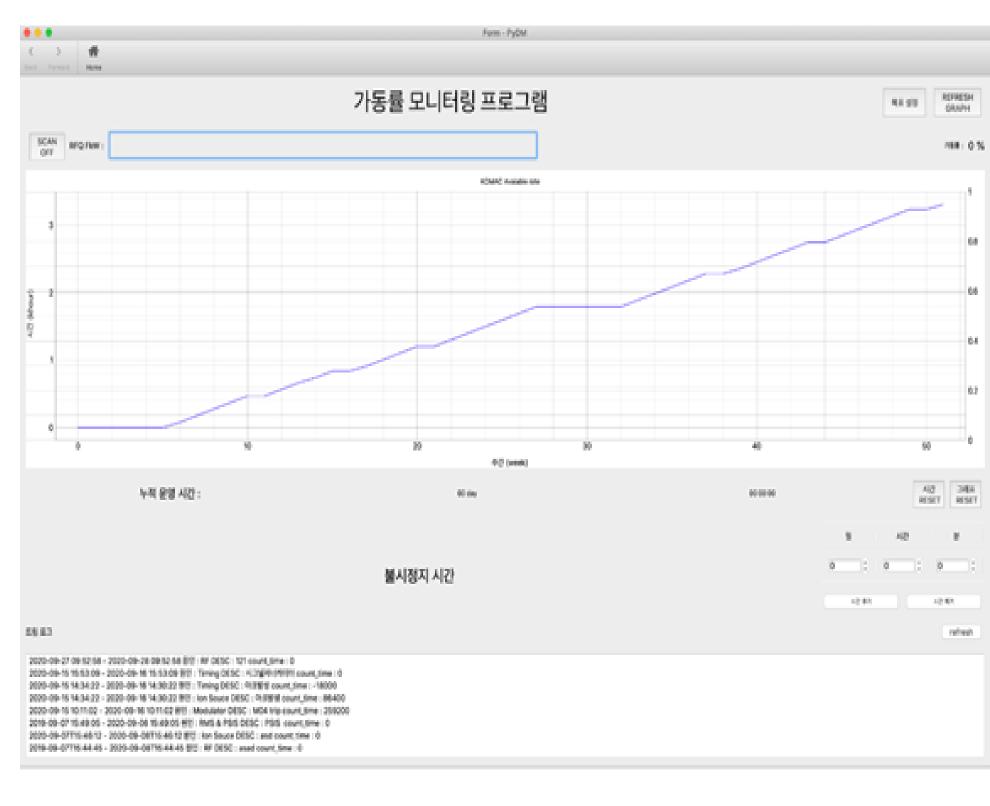


The major features of the graphical user interface

- Autocomplete search: when the user input the name, the code plots every variables related to the entered name in the list
- Select period : select start time, end time with a popup calendar
- Data browsing: auto scale, zoom in, zoom out, refresh, set range, noise filtering, plots the value on the cross-point
- Data export: linear interpolation, noise filtering, select file format

Operating Status Analysis and Vision which includes the WEB development





Operating Status Analysis Application

- The integrated time is calculated if the conditional statement is satisfied which includes Radio Frequency Quadrupole (RFQ) forward power value and user-defined trip source
- The integrated time is reflected on the database at midnight automatically
- From now on, In order to the improvement of the application, the alarm log db will be used and it annunciated the trip source from analysis alarm log

WEB Development

- The WEB interface will provide convenient access and the effective user-management
- The WEB service will be developed based on Django framework with python, html, css, javascript, ajax, etc
- Now, The static interface has been being developed based on HTML, CSS markup language