EPICS PV MANAGEMENT AND METHOD FOR RIBF CONTROL SYSTEM

A. Uchiyama[#], M. Komiyama, N. Fukunishi
RIKEN Nishina Center, Wako, Saitama, Japan

Abstract

For the RIKEN Radioactive Isotope Beam Factory (RIBF) project, the Experimental Physics and Industrial Control System (EPICS)-based distributed control system is utilized in Linux and vxWorks. Utilizing network attached storage (which has a high-availability system) as a shared storage, common EPICS programs (Base, Db, and so on) are shared by each EPICS Input/Output Controller (IOC). From the initial development of RIBF control system, it has continued to grow and consisted of approximately 50 EPICS IOCs and more than 100,000 EPICS records. Because RIBF has been constructed by extending RIKEN Accelerator Research Facility (RARF) in a previous project, the controllers for RARF are also utilized for RIBF control system. In this case, the dependence between the EPICS records and EPICS IOCs becomes complicated. For example, it is not easy to know the accurate EPICS record name information using only the device information. Therefore, we constructed a new management system for the RIBF control system to easily call up the detailed information. In the system, by parsing startup script files (st.cmd) to run EPICS IOCs, all EPICS records and EPICS fields are stored in the PostgreSQL-based database. By utilizing these stored data, we succeeded in developing Web-based management and search tools.

INTRODUCTION

The RIKEN Radioactive Isotope Beam Factory (RIBF) accelerator facility consists of five cyclotrons, including a superconducting ring cyclotron, and two linear accelerators as injectors [1]. For the RIBF, we constructed a control system based on the Experimental Physics and Industrial Control System (EPICS) for magnet power supplies, beam diagnostic instruments, vacuum control systems, and so on [2]. Different types of devices are connected via the EPICS Input/Output Controllers (IOCs) as front-end controllers for accelerator operation. For example, VME-based IOCs are adapted for magnet power supplies, and beam diagnostic and vacuum systems (e.g., Faraday cup and beam profile monitor) are constructed using Linux-based IOCs connected with N-DIMs, which are Ethernet-based control devices originally developed by the RIKEN Nishina Center [3]. Additionally, CAMAC, GPIB, Programmable Logic Controller (PLC), and others are also utilized as control devices with EPICS for the various applications in RIBF control system. The types of control devices and EPICS IOCs are listed in Table 1.

Currently, the EPICS-based RIBF control system consists of 51 IOCs and approximately 110,000 EPICS records. We have constructed the EPICS runtime database

---

#a-uchi@riken.jp

using an Oracle-based database. In this case, using the Oracle database has some advantages from the viewpoint of management because it is available to automatically make EPICS runtime database files easily using a program. In case of a large number of EPICS runtime database files for the same purpose, the abovementioned approach is efficient. Since RIBF control system has been constructed by extending the control system of RIKEN Accelerator Research Facility (RARF), which is the previous project, the relationship among their controllers are complicated. Therefore, we often manually develop EPICS runtime database files without using Oracle-based database.

For system maintenance and development in this situation, we need to identify the IOC connected to the process variables (PVs) because the RIBF control system does not use the UDP broadcasts for *ca_search* from EPICS clients on the subnet. However, the relationship between the IOCs and the PVs is confused, and an efficient system environment was not provided for developers and operators. Therefore, we have constructed a system to manage the EPICS PV to solve the complication.

Table 1: Current Status of the Type of EPICS IOC in the RIBF Control System

IOC Platform	Connected Control Device	Type	Number of IOCs
Linux x86	N-DIM [3] PLC GPIB Other network-based devices	Soft IOC	24
Linux x86	CAMAC	Embedded IOC	6
Linux f3RP61 [4]	PLC	Embedded IOC	14
vxWorks	NIO [5]	Embedded IOC	7

PV MANAGEMENT SYSTEM

IOC and Shared Storage

In the RIBF control system, common EPICS programs (EPICS-base, application programs, runtime database, and additional extensions programs) are stored in the network attached storage (NAS), and they are shared by all EPICS IOCs using the network file system or file transfer protocol. The system diagram is shown in Fig. 1. The NAS, manufactured by NetApp, provides a centralized system with a high-availability. On the other

hand, the IOCs are mounted common to the EPICS programs as read only, and the common EPICS programs are always compiled on another server, which is mounted in a write-allow configuration for control system development users. In addition to the development server, the management server is also mounted with the shared storage to read common EPICS programs.

Method of Storing PV Information

By reading the startup script files and accessing the EPICS runtime database files, the program can parse the runtime database files. Therefore, we developed a program such that the information is separately stored in the PostgreSQL-based database by parsing the file. The program is also compatible with the EPICS substitution file and macro. The system chart is shown in Fig. 2. Additionally, the information of the network-based devices, which are managed by the IOC, is inserted in the database. The summary of the types of data stored in the database is listed in Table 2. This program for parsing is coded in PHP and is regularly invoked from Crontab services, which run on Linux; thus, the data are always in the latest state.
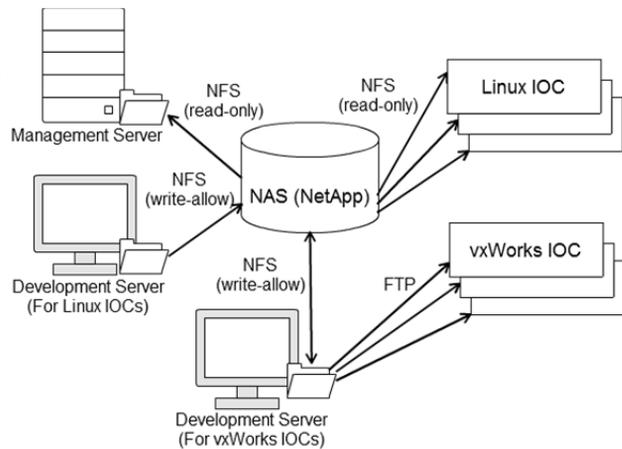

Figure 1: System Diagram for the IOCs and Shared Storage.

Table 2: Database Structure Used for Management System in the RIBF Control System (October 2015)

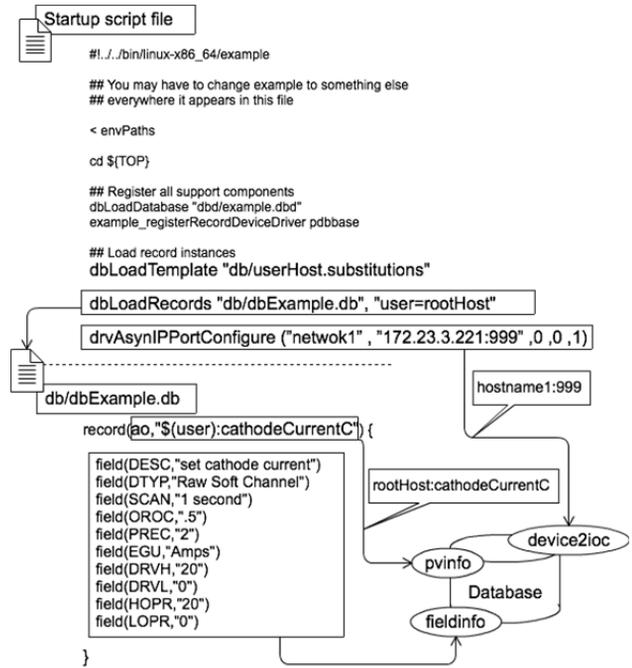| Table Name	Information Stored in the Column	Number of Records
iocinfo	Hostname of IOC	51
	Directory path of EPICS application	
	Startup script file name	
pvinfo	EPICS record name	110,192
	Record type	
fieldinfo	Field	3,151,383
	Field type	
device2ioc	Hostname for network-based device	432

Figure 2: System Chart of the Developed Program and Inserted Data.

STORED INFORMATION USAGE

Command-line Tool

Because the EPICS PVs and fields information are stored in the PostgreSQL-based database, the information is available to use for various purposes by using network communication. Therefore, we developed command-line-based tools to obtain the information, and it was implemented in the operational log system in the RIBF control system. The operational log system is one of the electric log systems for recording and viewing the accelerator operating time and contents of an operated device. To obtain operational information, we use *caMonitor*, which is an event-driven program using EPICS channel access protocol. However, the operational log system requires setting a large number of monitored parameters. Additionally, the specification of the runtime database will be changed for software maintenance. Therefore, manually setting the names of the required EPICS PVs without omission in the previous system is difficult. By using this system, the names of the required PVs are never missed from the list because creating the lists with a program is easy.

Web Application

By developing Web applications, searching the IOC hostname from the EPICS record name becomes possible, which is the same function as *cainfo*, a command-line tool included in the EPICS base software. On the other hand, because this developed system has an autocomplete feature, searching by EPICS record name is possible without requiring the completed EPICS record name. A screenshot of the developed Web application is shown in Fig. 3. Additionally, for all the fields that make up the
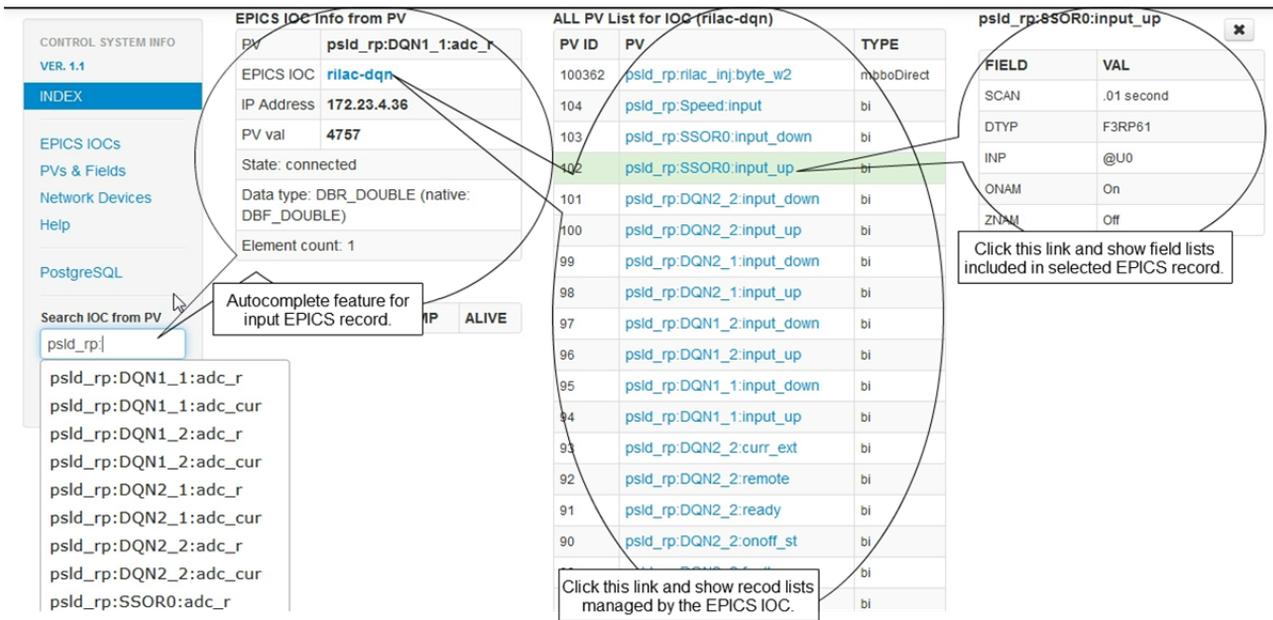
Figure 3: Screenshot of the Developed Web Application to Search EPICS IOC Name from EPICS Record.

EPICS record, accessing them from the record name via Web browser becomes possible. Therefore, we can check the information of the EPICS record and the fields without a source code to manage the system. Because the information of all network devices is also stored in the PostgreSQL-based database from the INP field and startup script file, the Web application has a feature that search the information of network-based devices from EPICS IOC name. Therefore, we can easily observe the relationship of the IOC and the managed network devices.

FUTURE PLAN

Using inserted information, alive monitoring will be implemented in the near future for the RIBF control system. Generally, alive monitoring is performed by a system that checks from the outside whether the computers continuously work. Because the alive monitoring conveys the downtime of the system to the administrator, it is an important system from the point of view of system availability. For EPICS-based system, an infrastructure monitoring system using NAGIOS [6] was introduced at Argonne National Laboratory (ANL) [7]. The NAGIOS-based system obtains information such as IOC from IRMIS [8], which is another management system for EPICS-based system at ANL. The XML files are created as parameters for alive monitoring.

On the other hand, EPICS IOCs are monitored by the *caget* function as a polling program in the NAGIOS-based system. In our system, we plan to implement a monitoring function using *caMonitor*. Additionally, not only IOCs but also all network-based devices will be monitored using the IP address and port number in this system. Therefore, we will be able to check both the port open/close and the ping states.

CONCLUSION

We have developed a system for management of EPICS PVs for the RIBF control system. Because the EPICS runtime database of files and startup scripts are managed in one place, their program analyses and the information of EPICS record, field, and network devices are stored in the database. Using this information, we can check the relationship among the IOCs and EPICS records, fields, and network devices via Web applications as well as the command-line tool. Currently, this feature is used in coordination with the operational log system.

In the future, we plan to implement an alive monitoring system for IOCs and network devices using the information in this management system.

REFERENCES

[1] O. Kamigaito et al., Proc. IPAC'14, Dresden, Germany (2014), p. 800.

[2] M. Komiyama et al., Proc. ICALEPCS'14, San Francisco, CA, USA (2014), p. 348.

[3] M. Fujimaki et al., RIKEN Accel. Prog. Rep. 37 (2004), p. 279.

[4] A. Uchiyama et al., Proc. PCaPAC08, Ljubljana, Slovenia (2008), p. 145.

[5] T. Tanabe et al., Proc. ICALEPCS2003, Gyeongju, Korea (2003), p. 597.

[6] https://www.nagios.org/

[7] D. E. R. Quock et al., Proc. PCaPAC08, Ljubljana, Slovenia (2008), p. 19.

[8] D. A. Dohan et al., Proc. ICALEPCS2007, Knoxville, Tennessee, USA (2007), p. 82.