DEVELOPMENT OF DATA-LOGGING SYSTEM FOR FFAG ACCELERATOR COMPLEX IN KURRI

Akihiro Osanai, Kyoto University, Kyoto 606-8501, Japan
Minoru Tanigaki, KURRI, Kumatori, Osaka 590-0494,Japan

Abstract

It is important to record parameters of an accelerator sequentially all through the operations not only for storing of experimental data but for maintenance in stable operations. In our control system for an FFAG accelerator complex in KURRI (Kyoto University Research Reactor Institute), a client-server type data-logging system, which consists of a database server and PCs controlling the accelerator complex as client PCs has been constructed. In order to construct a data-logging system with multiple client PCs, some data-transfer methods were attempted. The construction method and the performance of our data-logging system will be introduced in this presentation.

INTRODUCTION

The FFAG accelerator complex has been operated as part of KART project in KURRI [1] [2]. This accelerator complex consists of an ion source, three FFAG accelerators named Injector, Booster and Main Ring, respectively, and beamlines connecting between them. The overview of the FFAG complex is shown in Figure 1.

The control system for this complex is based on conventional PCs and programmable logic controllers (PLC) connected over TCP/IP network. Each PLC is responsible for autonomous control of connected devices such as motors or power supplies, and also responsible for maintaining a parameter database periodically read/written by remote PCs over TCP/IP network. Man-machine interfaces (MMI) and integrated sequences are developed using LabVIEW on these PCs. More details and the latest status of our control system were reported at EPAC2008 [3].

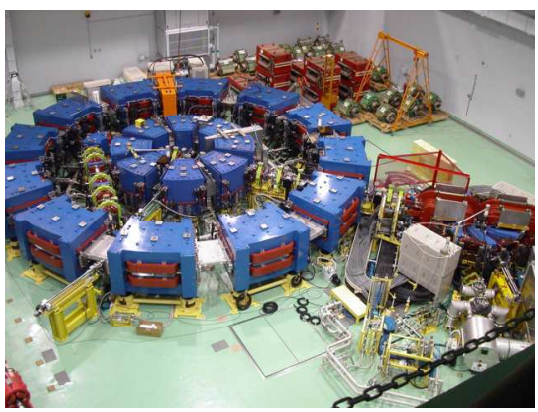

Figure 1: FFAG complex

As for a data-logging system, we have adopted a system based on MyDAQ in SPring-8 [4]. Figure 2 shows the schematic diagram of our system. Acquired data in control PCs is sent to a database server using MySQL and the stored data is browsed through network using Apache and PHP. In the case of multiple control PCs, various data transfer methods between the control PCs and the database server could be considered. More suitable method should be chosen depending on the purpose environment. Some transfer systems between a database server and control PCs installing MMI developed on LabVIEW were attempted and the most effective method was considered in our system. The details and result of our attempt are described in this paper.

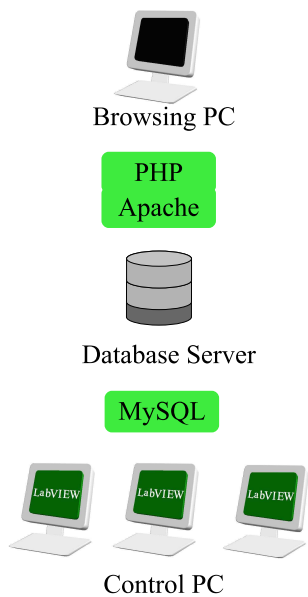

Figure 2: Schematic diagram of the data-logging system

SYSTEM CONSTRUCTION

System Environment and Requirement

The control system of the FFAG accelerator complex consists of about five control PCs communicating with PLCs every 100 $\sim$ 200 msec on the LAN (100 Mbps). Each PC has several MMI VIs developed on LabVIEW.

The data-taking in a high repetition cycle is not required as long as recording irregularities such as a failure of a constant-voltage power supply. 1 minute will be sufficient

as the time interval for the data-taking cycle in such purposes.

System Installation

Three types of installation methods of the data-logging system are examined for the client PC of the ion source in our FFAG accelerator complex. Five parameters appearing on the client PC are logged at the data-taking cycle of 1 sec.

Method I The first method is the simplest. The framework is shown in Figure 3. The client PC directly transfers data to databases on a server using SQL. In this method, there is no data transfer program for data-logging in the server side. Therefore the data transfer cycle and the type and number of data are determined on the client side, while the installation of MySQL and the grant of write privileges to the client PC are required.

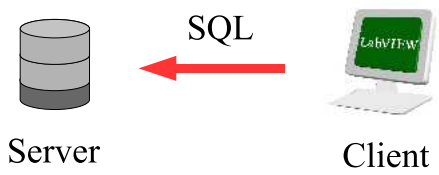

Figure 3: The framework of data transfer method I

Method II The second method is shown in Figure 4. A VI developed on LabVIEW is required to be installed on a server. All data transferred to the server from each PC client is translated by the VI and then sent as SQL commands to databases. The installation of MySQL to the client PC is not required in this method, but a data transfer program should be prepared for communication to the server program. Just like method I, the client side governs all of the data taking processes.

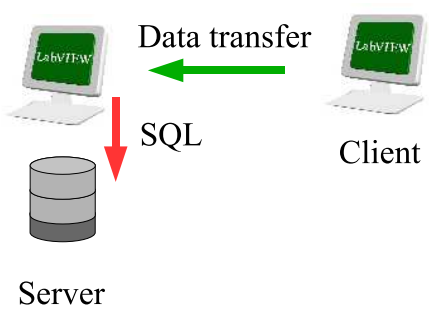

Figure 4: The framework of data transfer method II

Method III The last method is based on the advantages that the current control system is constructed on LabVIEW. The scheme of this method is shown in Figure 5. It is the same framework as the second one, except the data transfer control and realized using VI Server functions of LabVIEW. VI Server is used so as to programmatically control front panel objects, VIs, and LabVIEW, and to dynamically load, edit, and run VIs on a PC or remotely across a network. In this method, the data on the client PC is collected and sent to the databases by a VI on the server. No modification is required in clients side. The data transfer cycle and the type and number of data are determined on the server side. The disadvantage in this method is that every IP address of client PCs and variable names on corresponding VIs must be specified in the server side.

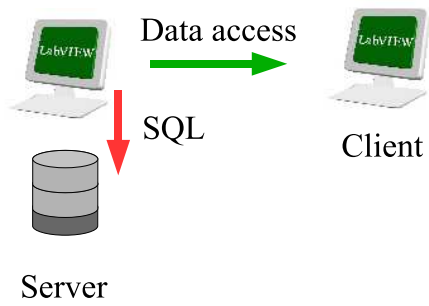

Figure 5: The framework of data transfer method III

RESULT AND DISCUSSION

All of above data transfer methods show sufficient performance on the data transfer in one-sec cyclic period. The method for our data-taking system should be chosen on the basis of other aspects in our circumstances since the shorter data-taking period is not currently required in our case. In addition to the merit and demerit in the installation, those in the system operation and maintenance should be considered. The methods described above are more tightly coupled to the control system as the number increases, in the order of method I, II, and III. Method III is suitable if the interaction between the systems should be avoided. But, the choice is different if the data taking is performed for a large number of client PCs which are independently maintained by different people. In such case, the method in which the data transfer is initiated by client side should be chosen. So method I or II should be adopted. In our case, the information required in method III such as IP addresses or variable names can be easily managed since the control system is rather small and maintained by a limited number of people. Therefore, the method III can be a better choice for its less load on client PCs.

SUMMARY AND FUTURE PLAN

The data-logging system based on MyDAQ has been developed for FFAG accelerator complex in KURRI. In terms of the data transfer method between a server and client PCs, three types of installation methods were examined for the client PC of ion source and showed sufficient performance. The choice of the method mainly depends on the system environment such as system scale, management architecture of a server and client PCs.

As for a transfer cycle, further study will be necessary in cases where more client PCs are worked and more data is treated. In addition, we will attempt to develop the system not caring about programming languages for general use of this data-logging system.

REFERENCES

[1] S. Shiroya, H.Unesaki et al., "Neutronics of Future Neutron Source Based on Accelerator Driven Subcritical Reactor Concept in Kyoto University Research Reactor Institute (KURRI)", Int. Seminar on Advanced Nucl. Energy Systems toward Zero Release of Radioactive Wastes, 2nd Fujiwara Int. Seminar, Nov. 6-9, 2000, Shizuoka Japan, $abstracts$ p. 58

[2] S. Shiroya, H. Unesaki et al., Trans. Am. Nucl. Soc., 2001 Annu. Mtg., June 17-21, 2001, Milwaukee, Wisconsin, p. 78

[3] M. Tanigaki, K. Takamiya et al., "Control System for a 150 MeV FFAG Complex in KURRI", TUPP014, Proceedings of EPAC2008, Genoa, Italy

[4] Akihiro Yamashita, Toru Ohata, "MyDAQ, A Simple Data Logging and Display Server", Proceedings of PCaPAC2005, Hayama, Japan