Implementation of an Operator Intervention System for Remote Control of the RIKEN 28GHz Superconducting ECR Ion Source[a)]

A. Uchiyama,[1,b)] K. Furukawa,[2] Y. Higurashi,[3] and T. Nakagawa[3]

[1]*The Graduate University for Advanced Studies, 1-1 Oho, Tsukuba, Ibaraki 305-0801, Japan*
[2]*High Energy Accelerator Research Organization, KEK, 1-1 Oho, Tsukuba, Ibaraki 305-0801, Japan*
[3]*Nishina Center for Accelerator Based Science, RIKEN, Hirosawa 2-1, Wako, Saitama 351-0198, Japan*

(Presented XXXXX; received XXXXX; accepted XXXXX; published online XXXXX)

The control system for the RIKEN 28GHz superconducting electron cyclotron resonance ion source (28GHz SC-ECRIS) consists of a distributed control system based on the experimental physics and industrial control system (EPICS). To maintain the beam quality for the long beam-service time at the RIBF, beam tuning to prevent subtle changes in the 28GHz SC-ECRIS conditions is required. Once this is achieved, it should then be possible to check conditions and operate the ion source at any time. We have designed a web-based operational interface (OPI) to remotely control the ion source, but for access and control from several locations, suitable access security, policies, and methods are required. We thus implemented an operator intervention system that makes it possible to safely access the network externally with the permission of on-site accelerator operators in the control room.

I. INTRODUCTION

The performance of the ion source is an important factor in determining the overall performance of a heavy ion accelerator. At the RIKEN radioactive isotope beam factory (RIBF), the 28GHz superconducting electron cyclotron resonance ion source (28GHz SC-ECRIS) was constructed in 2009 in an effort to increase the uranium-ion beam intensity[1]. The stability of the beam is important in terms of beam service availability for experiments at the RIBF[2], and periodic fine tuning of parameters such as the RF power and gas pressure is required to ensure a high-intensity beam that is stable over a long period. It is therefore common for ion source developers to telephone the on-site accelerator operator to check the status of the 28GHz-SCECRIS while they are not present at the facility. However, ion source developers could obtain more detailed information about the status of the source, relay operational instructions, and maintain the beam quality if it was possible to access the control system from outside the facility. As there will be differences between the terminals used in the accelerator control room and the remote locations, a cross-platform software environment should be adopted (for example, less common software such as the X Window System may not necessarily be installed on a remote computer, as it is on the control-room Linux PCs). In this regard, web-based technology has an advantage in that any device with a browser, e.g., a PC, mobile phone, tablet, or TV, can be employed, and web-based technology has been shown to be a useful means of disseminating accelerator and beam status information from the RIBF control system [3].

Traditional web applications using asynchronous JavaScript and XML (Ajax) lack the interactivity needed to act as an operator interface (OPI) for some of the accelerator hardware[4]. However, the WebSocket protocol can be implemented to improve the interactive performance. This protocol achieves bidirectional communication between a web server and browser, unlike the periodic polling access of Ajax. The bidirectional communication can be utilized to develop an OPI that can both monitor and control ion-source devices, and we have successfully developed a WebSocket-based client system to monitor the real-time action of the EPICS-based system via a web browser [5].

While there are clear advantages to the developed client system, there are still some problems that need addressing, namely hazards associated with access via WebSocket and operational mistakes. Focusing on the latter, operational mistakes can arise because it is difficult to give a full picture of the status over the telephone (seeing the RIBF parameters on the control-room PC displays is of more benefit) and because operating the 28GHz SC-ECRIS with no real-time response is difficult, even if the remote operator can access snapshots of the graphical user interface (GUI) panels as image files. To prevent operational mistakes from occurring, the accelerator conditions including the 28GHz-SCECRIS need to be understood as a whole. As the accelerator operators in the control room have the best understanding of the RIBF accelerator conditions, their judgment is

[a)] Contributed paper published as part of the Proceedings of the 15th International Conference on Ion Source, Chiba, Japan, September, 2013.
[b)] a-uchi@riken.jp.

indispensable to ensure that devices are controlled safely. The on-site accelerator operator must then be allowed to intervene in the remote operation. The WebSocket-based client system is designed so that output control by the remote ion source developer always requires the permission of an on-site accelerator operator who can intervene at any time.

II. ARCHITECTURE OF THE 28GHz SC-ECRIS CONTROL SYSTEMS

The 28GHz SC-ECRIS control system is an experimental physics and industrial control system (EPICS)-based distributed control system that is part of the RIBF control system[6]. EPICS is an open-source software tool kit for the development of distributed control systems for large experimental physics facilities such as accelerators and telescopes[7,8]. The tool kit was initially developed by the Los Alamos National Laboratory and the Argonne National Laboratory and continues to be developed in collaboration with various other laboratories. The EPICS-based system was adopted at the RIKEN Accelerator Research Facility (RARF) in 2001[9]; this facility is now the injector section of the RIBF.

The EPICS-based client system sends or receives data to/from an input/output controller (IOC) using a dedicated protocol known as channel access (CA). Process variables (PVs) corresponding to input/output commands for hardware device are manipulated in the IOC. The GUI of the OPI runs in the X Window System, while the electronic logbook and software to view archived data are web-based systems[10].

III. WEBSOCKET-BASED CLIENT SYSTEM

WebSocket was originally designed to be implemented as a part of HTML5 and was standardized as RFC6455 by the IETF in 2011[11]. WebSocket realizes an interactive response using bidirectional communication and thus compensates for the disadvantage that could not be eliminated in traditional HTML. Web browsers such as Firefox, Chrome, and Internet Explorer can be used as a cross-platform environment with the implementation of a WebSocket-based OPI. This allows the user to access the OPI from not only a PC-based web browser but also other devices such as Android and Apple mobile phones. After satisfying the conditions for interactive access to the EPICS-based system, we implemented the WebSocket server, which connects to the EPICS IOC via CA, as a web-based OPI. Figure 1 shows a chart of the whole system. The WebSocket-based client system consisted of HTTP server, HTML and JavaScript files, and WebSocket server. The WebSocket server gets the parameters as PVs from EPICS IOC by calling CA API.

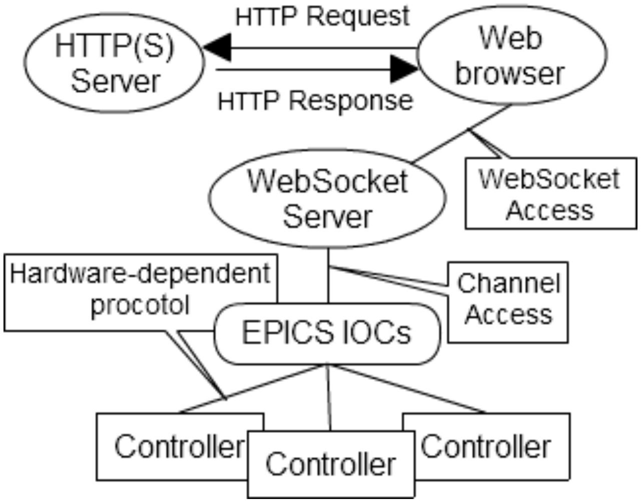

FIG 1: System diagram of WebSocket-based OPI in EPICS-based Control System.

IV. IMPLEMENTATION OF THE OPERATOR INTERVENTION SYSTEM

The main part of operator intervention system consists of a PV gateway provided by the EPICS collaboration, a database, and web applications[12]. An outline of the system is as follows (See figure 2). The on-site accelerator operator first gives permission to every output of the PVs corresponding to the hardware of the 28GHz SC-ECRIS control system. If the PVs have not been granted permission, then the operator intervention system always rejects the output command in the CA protocol layer. The system policy of the operator intervention system can be summarized as follows.

● For monitoring the status of the 28GHz SC-ECRIS using the WebSocket-based OPI, the permission of the on-site operator is not necessary as long as authentication with an SSL connection has been completed.
● When the remote user controls the outputs via WebSocket, they send EPICS PVs requests to the operator intervention system.
● The on-site operator judges whether the requested PVs are available for control.
● If control is possible, a command to accept is sent; otherwise, a refuse command is sent.
● In sending the accept command, the on-site operator set the duration time and controllable upper and lower limits.
● The on-site operator can stop the remote operation at any time if, for example, problems with the accelerator arise.
● Logs of all the outputs via the operator intervention system and WebSocket access are checked by the on-site accelerator operators.

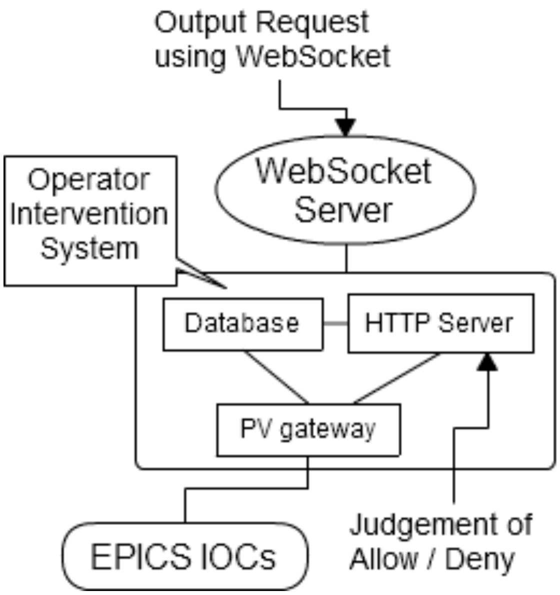

FIG 2: Outline of the Operator Intervention System. It was implemented between WebSocket server and EPICS IOCs.

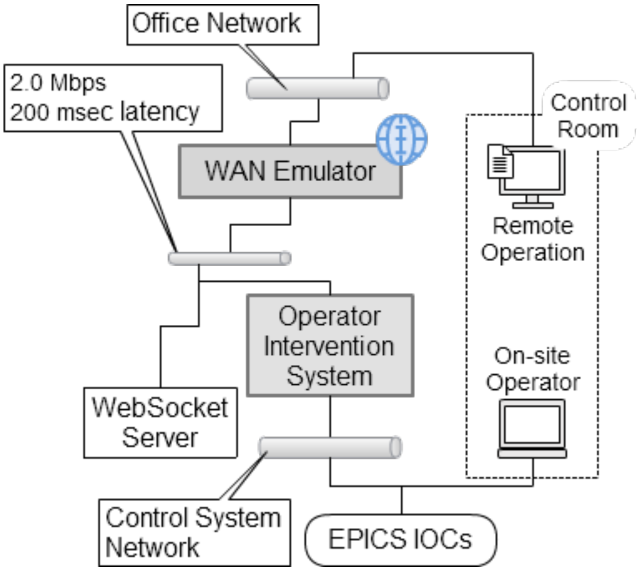

FIG 3: Test environment with another network emulated WAN. The gray area show the implemented system.

V. RESULT OF THE TEST SYSTEM

Remote operation of the 28GHz SC-ECRIS control system was tested using a wide area network (WAN) emulator with both the remote user and on-site operator in the same control room but accessing different network environments (See figure 3). The WAN emulator creates another network with a performance of 2.0 Mbps and 200 msec latency. As a result, we were able to successfully operate part of the 28GHz SC-ECRIS (gas valves, RF power, position of an electrode, and so on) remotely without serious problems. We confirmed that the output instructions were not reached to EPICS IOC without on-site operator's permission. The networking performance is tolerable level so that our system can be used from networks provided by almost all Japanese internet service providers[13]. We have to note that the debugging of HTML and JavaScript programs proves to be more important to prevent the operational mistakes than other general web applications. In case of EPICS native OPI written in C language, the compiler should verify the incomplete syntax of the source code. On the other hand, even when the HTML or JavaScript codes have a bug, the code may be run on the web browser in some situations.

VI. CONCLUSION

We have implemented an operator intervention system for the EPICS-based 28GHz SC-ECRIS control system for an evaluation. The on-site operators manage the digital and analog outputs for remote operation, and access will be denied in the network protocol layer if certain criteria are not met. The purpose of our work was to realize safe remote control of the 28GHz SC-ECRIS via a WebSocket-based OPI, and after confirming the usability and safety of the system in September 2013, we expect that the operator intervention system will be adopted for the 28GHz SC-ECRIS control system. We also expect that this system may be implemented in other ECRIS and RIBF control systems.

[1] Y. Higurashi, J. Ohnishi, T. Nakagawa, H. Haba, M. Tamura, T. Aihara, M. Fujimaki, M. Komiyama, A. Uchiyama, and O. Kamigaito, Rev. Sci. Instrum. **83**, 02A308 (2012)
[2] H. Okuno, N. Fukunishi, O. Kamigaito, Progress of Theoretical and Experimental Physics, 03C002. (2012).
[3] A. Uchiyama, R. Koyama, M. Komiyama, M. Fujimaki, N. Fukunishi, in Proc. ICALEPCS2011 (Grenoble, France, 2011) p.1161
[4] Y. Furukawa, in Proc. ICALEPCS2011 (Grenoble, France, 2011) p.673
[5] A. Uchiyama, K. Furukawa, Y. Higurashi, T. Nakagawa in Proc. PCaPAC2012 (Kolkata, India, 2012) p.7
[6] M. Komiyama, M. Fujimaki, N. Fukunishim, A. Uchiyama, J. Odagiri, in Proc. ICALEPCS2009 (Kobe, Japan, 2009) p.275
[7] L. R. Dalesio, J. O. Hill, M. Kraimer, S. Lewis, D. Murray, S. Hunt, W. Watson, M. Clausen, and J. Dalesio, Nucl. Instrum. Methods Phys. Res. A **352**, 179 (1994)
[8] K. Furukawa, in Proc. PAC07 (Albuquerque, New Mexico, USA, 2007) p.873
[9] M. Komiyama, I. Yokoyama, M. Fujimaki, M. Kase, J. Odagiri, in Proc. CYCLOTRONS2004 (Tokyo, Japan) p.478
[10] A. Uchiyama, K. Furukawa, Y. Higurashi, M. Komiyama, T. Nakagawa, K. Ozeki in Proc. ECRIS2012 (Sydney, Australia, 2012) (to be published)
[11] I. Fette and A. Melnikov, The WebSocket Protocol, IETF HyBi Working Group. 2011.
[12] A. Uchiyama, K. Furukawa, Y. Higurashi, T. Nakagawa, in Proc. The 10th Annual Meeting of Particle Accelerator Society of Japan (Nagoya, Japan), (to be published)
[13] Akamai Technologies, Inc., The State of the Internet, volume 6, number 1, 1st QUARTER, 2013 REPORT,
URL: http://www.akamai.com/stateoftheinternet/