Development of wide area environment accelerator operation and diagnostics method

Akito Uchiyama^{*}

*RIKEN Nishina Center, 2-1 Hirosawa, Wako, Saitama 351-0198, Japan
and SOKENDAI (The Graduate University for Advanced Studies), 1-1 Oho, Tsukuba,
Ibaraki 305-0801, Japan*

Kazuro Furukawa[†]

*High Energy Accelerator Research Organization (KEK), 1-1 Oho, Tsukuba, Ibaraki 305-0801, Japan
and SOKENDAI (The Graduate University for Advanced Studies), 1-1 Oho, Tsukuba,
Ibaraki 305-0801, Japan*

(Received 15 April 2015; published 17 August 2015)

Remote operation and diagnostic systems for particle accelerators have been developed for beam operation and maintenance in various situations. Even though fully remote experiments are not necessary, the remote diagnosis and maintenance of the accelerator is required. Considering remote-operation operator interfaces (OPIs), the use of standard protocols such as the hypertext transfer protocol (HTTP) is advantageous, because system-dependent protocols are unnecessary between the remote client and the on-site server. Here, we have developed a client system based on WebSocket, which is a new protocol provided by the Internet Engineering Task Force for Web-based systems, as a next-generation Web-based OPI using the Experimental Physics and Industrial Control System Channel Access protocol. As a result of this implementation, WebSocket-based client systems have become available for remote operation. Also, as regards practical application, the remote operation of an accelerator via a wide area network (WAN) faces a number of challenges, e.g., the accelerator has both experimental device and radiation generator characteristics. Any error in remote control system operation could result in an immediate breakdown. Therefore, we propose the implementation of an operator intervention system for remote accelerator diagnostics and support that can obviate any differences between the local control room and remote locations. Here, remote-operation Web-based OPIs, which resolve security issues, are developed.

DOI: 10.1103/PhysRevSTAB.18.082801

PACS numbers: 07.07.Tw

I. INTRODUCTION

In large experimental physics research centers, such as accelerator or telescope facilities, control systems are operated from a control room using applications such as user interfaces (UIs). Control rooms are generally located less than 10 km from control systems; in fact, in some accelerator facilities, the control room is located near the accelerator room. However, for equipment maintenance, to perform collaborative physics measurements on different machines, and for several other reasons, some users require the ability to conduct remote operations from distant locations. By utilizing modern control systems with high-performance networks, control system engineers have realized some of the requisite features of remote operation. In the control system technology field, several studies and

attempts at implementing such remote operation have been reported.

A. Remote accelerator control room

Various remote accelerator control rooms have been planned and implemented. For example, for remote operation of the International Linear Collider (ILC) [1], implementation of the global accelerator network (GAN), a dedicated control system for large projects, has been planned since March 2000 [2]. The concept underlying GAN is the construction of remote accelerator control rooms that are virtually equivalent to the local control room as regards performance and features, and that can be utilized by remote users of the facility. In May 2005, a system similar to GAN, the Global Accelerator Network Multipurpose Virtual Laboratory (GANMVL), was implemented in the European Design Study toward a Global TeV Linear Collider (EUROTeV) project [3].

B. Remote experiment systems

Remote operation for experiments involving the Large Hadron Collider (LHC) [4] is achieved using control

^{*}a-uchi@riken.jp[†]kazuro.furukawa@kek.jp

Published by the American Physical Society under the terms of the Creative Commons Attribution 3.0 License. Further distribution of this work must maintain attribution to the author(s) and the published article's title, journal citation, and DOI.

systems. At the Compact Muon Solenoid (CMS) experiment at the LHC, located at the European Organization for Nuclear Research (CERN) in Switzerland, installation of a virtual control room at the Fermi National Accelerator Laboratory (Fermilab) in the U.S. [5] is planned. For the AToroidal LHC Apparatus (ATLAS) experiment at the LHC, a remote system has been implemented, similar to the CMS experiment [6]. By using remote monitoring rooms located at the American ATLAS institution, communication between American ATLAS members and scientists at CERN has been improved. For the Photon Factory (PF) experiment at the High Energy Accelerator Research Organization (KEK) in Japan, a new beamline with video conference system capabilities and, in some cases, remote operation is utilized, to create a new “collaboratory” research network [7]. The Wide-area Remote Experiment System (WRES) has been developed for safe experimentation through remote operation at SPring-8, a synchrotron radiation facility located in Japan [8]. The first experiment using WRES was successfully performed from the RIKEN Wako campus, located 480 km from SPring-8, in October 2010 [9].

C. Motivation

In previous studies, remote control systems have been constructed for environments with sufficient computer resources. However, situations exist in which remote operation is required under limited resources. For example, experts are occasionally required to troubleshoot their equipments while they are away on business trips. This study aims to implement remote operation in environments with limited computer resources. Therefore, we aim to control some accelerator components and to include troubleshooting features, even with limited computer resources. In order to achieve these goals, we believe that Web technology is the primary candidate for operator interfaces (OPIs), because Web applications have the following advantages: platform-independence, rapid system development, and ease of system maintenance.

For the remote operation of control systems, certain security systems must be considered, as the remote operation of an accelerator via wide area network (WAN) faces a number of system security challenges. The term “system security” includes both computer security and system safety, e.g., machine protection, human protection, and human error prevention. Any errors in the operation of remote control systems could result in immediate equipment breakdown, and could cause serious human protection problems without sufficient communication. To prevent operational errors, the accelerator parameters must be completely understood by all project participants, particularly by the on-site operators. Thus, improvement of system security during remote operation is also an aim of this study.

II. CONTROL SYSTEM FRAMEWORK AND WEB-BASED SYSTEMS

A. Control system framework

Modern accelerator control systems, such as the Experimental Physics and Industrial Control System (EPICS) [10], Distributed Object Oriented Control System (DOOCS) [11], Message and Database Oriented Control Architecture (MADOCA) [12], and Taco Next Generation Objects (TANGO) [13] are constructed based on corresponding frameworks.

Of these, EPICS has one of the most prevalent control system frameworks. Hence, EPICS is adopted as the control system framework in this study. Figure 1 shows that the system configuration diagram for the EPICS Input/Output Controller (IOC), the runtime database, and the channel access (CA) client. By reading and writing the values in the EPICS runtime database, EPICS-based systems can control hardware devices [14]. In this model, the record type is described within the record itself. For example, when a record works simply by controlling analog output (ao) devices, a line that indicates the record ao is included in the database file. On the other hand, when a record has fields, its behavior regarding the setting of upper and lower limit values, for example, is defined by coding the fields.

EPICS has adopted a client server model. In an EPICS-based control system, the EPICS Channel Access (CA) protocol is implemented to enable communication between the EPICS IOC and CA client. The CA protocol [15] represents the presentation layer of the standard model. Through software development according to the CA protocol specifications, various types of hardware-dependent protocols can be unified by EPICS IOC in the client system.

B. Web-based systems for accelerator operation

From the 1990s to approximately 2007, almost all Web-based systems in control systems were used for static access, such as for applications for archive viewers and log

FIG. 1. Typical EPICS configuration with IOC, runtime database, and CA client.

notebooks, where fast response is not required. WebOPI was developed at SNS/ORNL in 2011, as one of the features of the Control System Studio [16]. WebOPI aims to provide Web access to EPICS-based OPIs that are created in CSS/BOY, without modification. In WebOPI, various kinds of information such as numerical values and alarm-level colors are displayed on the Web in real-time using Asynchronous JavaScript and XML (AJAX) technology [17]. To publicize information via the Internet, it is only necessary to install Apache Tomcat [18] as the Web server with a JAVA servlet container, and modification of the OPI file created in CSS/BOY is not required.

As described previously, Web technology is used for dynamic-access applications such as OPI; however, AJAX interactive performance is not sufficient via the Internet, because of the synchronous nature of HTTP protocol. This is especially true under a low-bandwidth network. In addition, given that general accelerator control systems are implemented with dedicated local networks, the control systems must also improve remote operation and network security when remote users gain access from the Internet.

III. WEBSOCKET INTERFACE FOR EPICS

A. Bidirectional communication using Web

During beam operation and maintenance, accelerator control from different locations and from the central control room is often required. However, it is impractical to replace the GUI-based OPI with a Web-based system using AJAX technology, because of interactive performance issues. Therefore, as a next-generation OPI over the Web using the EPICS CA protocol, we have developed a client system based on WebSocket, which is a new protocol provided by the Internet Engineering Task Force (IETF) for Web-based systems. WebSocket is a Web technology that provides bidirectional, full-duplex communication channels over a single TCP connection.

B. Development background

One of the advantages of an EPICS-based system is the unified communication protocol between a front-end controller and the client systems provided by the CA protocol. Thus, even when various types of controllers are used as control systems, all client systems, such as OPIs, can be developed based on the CA protocol without hardware dependencies. In general, a method that uses the CA libraries or a display manager supported by EPICS collaboration, such as CSS/BOY, is adopted for GUI development in the EPICS-based system. On the other hand, WebSocket-based development methods for the main OPI were proposed at SPring-8 and a prototype was implemented, which was constructed using a MADOCA control system framework [19]. Given that real-time Web applications have many advantages, we developed a corresponding WebSocket server for the EPICS CA and

implemented Web applications using WebSocket for the EPICS-based control system.

C. WebSocket architecture

WebSocket is a new protocol for achieving bidirectional communication between a WebSocket server and a Web browser, originally in HTML5. Through draft versioning, it was formulated as an RFC6455 by IETF in December 2011 [20]. WebSocket can be used for real-time applications because of its low protocol overhead. Its API specifies how to create and access the protocol within a Web browser: A WebSocket connection is established via JavaScript, and then becomes available for the sending/receiving of messages and execution of error handling through a connection.

D. EPICS channel access for Node.js

After investigating the development of such a protocol, we utilized Node.js to aid in the development of the WebSocket server. Node.js [21] is a server-side JavaScript language developed based on the Google V8 JavaScript Engine [22]. To implement the WebSocket server with a CA connection, Node.js must call the CA API. Therefore, we developed add-on software to interface CA with Node.js, called NodeCA. Since Node.js can call C++ functions, NodeCA utilizes the CA library provided by the EPICS base. To call CA from Node.js, `caGet`, `caPut`, and `caMonitor`, which are basic functions in EPICS, were prepared (see Fig. 2).

E. WebSocket-based OPI

After satisfying the need for interactive access for WebSocket using an EPICS-based system, we implemented NodeCA and the WebSocket server as a Web-based OPI (as shown in Fig. 3). The OPI was found to have adequate accelerator operation abilities, compared with the traditional EPICS-based OPI. Moreover, we confirmed that comparatively-new browsers are capable of obtaining and displaying numerical values from the EPICS input/output controller (IOC) via WebSocket.

FIG. 2. Outline of EPICS-based OPI using WebSocket.

FIG. 3. WebSocket OPI UI (right) and operator intervention system UI (left). For system testing, both are displayed on the same PC.

F. WebSocket-based OPI network bandwidth

For remote operation performance comparison, the WebSocket-based OPI network bandwidth was measured. In the OPI, 30 numerical values were monitored at intervals of 0.1 s. Figure 4 shows that the bandwidth requires 84.39 kbps with an average of 10 min downstream

(from LAN A to LAN B), for a single client. However, Fig. 5 shows that the upstream network bandwidth (from LAN B to LAN A) requires 22.42 kbps only with an average of 10 min. Hence, we believe that WebSocket-based OPI provides the most efficient remote operation.

FIG. 4. WebSocket OPI downstream bandwidth.

FIG. 5. WebSocket OPI upstream bandwidth.

IV. OPERATOR INTERVENTION SYSTEM

A. Development background

Several scenarios require cooperation between experts and remote operators across WAN links for accelerator operation. For accelerator problems and beam-change scenarios (for example, beam emittance from ion sources), on-site accelerator operators require expert directions over the telephone with regard to accelerator components, such as the rf, vacuum, beam diagnostic, control system, and ion source. In such cases, the ability to provide the accelerator operators with precise directions depends on the level of information required for remote troubleshooting.

WebSocket-based OPIs are used to resolve problems via WAN links. In addition to monitoring accelerator parameters from locations far from the accelerator control room, remote users can operate output accelerator devices, even if no concerns regarding radiation safety, human physical safety, and network access security, etc., exist.

In order to prevent operational mistakes, the accelerator parameters must be completely understood. Therefore, a safety system with on-site accelerator operator intervention in order to control output via remote access must be considered. Note that a safety mechanism is indispensable when the remote operation must monitor parameters and control output, in order to prevent accelerator problems caused by operational mistakes. To address safety issues and improve the usability of remote operations, we developed an operator intervention system for WebSocket-based OPIs for EPICS.

B. Security policy

To protect accelerator networks from external intrusion, we considered the following. For accelerator network access from the WAN, we ensured secure WebSocket access using virtual private network (VPN) and authentication via a secure sockets layer (SSL). Also, accelerator operators must have full information on users attempting to access the network (login ID, etc.) and the access route (full Internet provider domain, etc.), before WebSocket control is allowed. Therefore, all WebSocket control access logs should be visible to accelerator operators.

Next, we considered accelerator operation instructions to the EPICS IOC from the Web browser via WebSocket. For caPut, which provides instructions to each device or component during accelerator operation, accelerator operators must understand its role and behavior perfectly. However, the operators are not required to understand certain simple functions in detail, such as caGet and caMonitor, which are used to monitor or obtain numerical values for accelerator parameters. If some accelerator parameters are changed using the Web application from outside the internal networks without considering beam tuning, the accelerator operating conditions may be

FIG. 6. Operator intervention system procedure.

complicated in many cases. Thus, when using caPut, a policy between accelerator operators in the control room and remote users exerting control over the Web is required.

C. Operator intervention system concept

The purpose of the proposed operator intervention system is to ensure safe remote output control of EPICS via the WebSocket server. To simply monitor the status of the accelerator parameters using the WebSocket-based OPI, permission from the on-site operator is unnecessary, provided authentication has been accomplished through a SSL connection. The WebSocket-based client system is designed such that output control via remote operation always requires the permission of an on-site operator, who can intervene at any time. The main component of the proposed operator intervention system consists of a process variable (PV) gateway provided by EPICS collaboration [23], a MySQL database, and AJAX Web applications. The procedure for the implementation of the proposed operator intervention system is as follows (see Fig. 6): (i) The remote user sends a request command to the on-site operator via the operator intervention system; (ii) The on-site operators grant permission to every PV output corresponding to the EPICS hardware devices. The on-site operators can determine the upper and lower remote control operation limits before sending permission commands; (iii) The remote user can control the operation via the WebSocket-based OPI with permission to access the PV outputs. If this permission has not been granted, the operator intervention system always rejects the output command at the CA protocol layer.

If accelerator operators decide to interrupt operation, they discontinue WebSocket operations as soon as possible. Also, if a certain period of time elapses, WebSocket operation permission will be canceled through timeout.

FIG. 7. Key services, system management flow, accelerator control flow, and information logging flow in operator intervention system.

D. Operator intervention system design

1. Required services

The operator intervention system comprises important services that run on several distributed hosts. A chart of the system is depicted in Fig. 7. In this system, the PV gateway is used as a proxy system for the CA connection to reduce the network load in the back-end EPICS IOCs.

Also, access restriction provided by the EPICS access security group (ASG) is implemented in the PV gateway and the EPICS IOC. EPICS ASG is a security function that implements access restrictions to the EPICS IOC and PV gateway using a configuration file [24]. Using ASG, it is possible to realize access permission and read/write access through a host name and a login name.

In the proposed operator intervention system, the WebSocket server always connects to the EPICS CA server via the PV gateway. Further, a MySQL service is necessary to store the flag value via an internal network. The Apache Web server and network file system (NFS) are run on an application server. The NFS is used to share log files among the WebSocket and Apache Web servers, the PV gateway via the internal network, and the MySQL service. WebSocket server programs developed using Node.js run on a separate host, which connects to the MySQL database and EPICS IOCs via the PV gateway.

2. Upper/lower limit setting

In general, EPICS utilizes DRVH/DRVL fields to set the upper and lower limits of the analog output (ao) records [24] (see Fig. 1). In this case, however, utilizing the DRVH/DRVL fields of the EPICS IOCs to control the system network is inadvisable because it may cause problems, such as insufficient parameter settings, in normal operation with remote users. Therefore, we implemented this function by modifying the source code of the PV gateway. Consequently, the customized PV gateway is possible to

set the upper and lower limits of ao records, without changing the field values of real EPICS IOCs.

3. User interface

We developed the operator intervention system interface as a Web application using AJAX, because such an interface can be less responsive than the WebSocket-based OPI. The UI of the operator intervention system is shown in Fig. 3. On-site operators can verify the requested EPICS, completed, and in-operation PVs on a single screen. Similarly, all access and caPut logs can be verified by the on-site operator at any time.

E. System implementation

At the RIKEN Radioactive Isotope Beam Factory (RIBF) [25], we implemented an operator intervention system and WebSocket-based OPIs on the EPICS-based RIKEN 28 GHz superconducting ECR ion source (28 GHz SC-ECRIS) control system for evaluation [26]. We confirmed that output instructions did not reach the EPICS IOC without permission from the on-site operator. Also, using a WAN emulator, an implementation test environment with approximately the same network latency as that obtained between Japan and the USA (200 ms) was constructed. As a result, remote operation was performed satisfactorily with a network latency of 200 ms. On the other hand, the WebSocket-based system discussed in Sec. III F has been shown to function with a small network bandwidth. Therefore, networking performance tolerance should allow this system to be used over almost all the networks provided by Japanese Internet service providers, facilitating stable and sustainable operation of the machine [27].

V. CONCLUSION

In this paper, we proposed an operator intervention system and WebSocket-based OPI for implementation in an accelerator control system. For remote operation using limited computer resources, we developed a server that allows connection with the EPICS CA by WebSocket. In this system, real-time Web-browser-based control is enabled. To prevent various far-remote operation problems, a safety system is necessary. This system ensures secure remote operation, including output device operation, by allowing the intervention of an on-site operator. The implementation of the operator intervention system facilitates remote real-time Web-based vacuum and beam-current monitoring using numerical values. Also, output control can be performed without any lapse in operation safety.

However, if even a single problem related to the accelerator components occurs, it may not be possible to provide beam service to experiment users. In all accelerator facilities, beam availability [28] is an essential factor determining the performance. Therefore, rapid troubleshooting using remote operation is a useful method for achieving higher

beam availability. As regards troubleshooting, the system developed in this study can solve external problems using limited resources. As a result, this technology should contribute to enhanced accelerator operation without the use of an EPICS-based control system.

ACKNOWLEDGMENTS

We extend our heartfelt appreciation to Dr. Yoshihide Higurashi and Dr. Takahide Nakagawa, who provided carefully considered feedback and support for the system implementation in this study. Finally, we are deeply grateful to Prof. Noboru Yamamoto, whose comments and suggestions throughout the course of our study were of immeasurable value.

-
- [1] G. Aarons, T. Abe, J. Abernathy, M. Ablikim, H. Abramowicz, D. Adey, C. Adloff, C. Adolphsen, K. Afanaciev, I. Agapov *et al.*, International Linear Collider reference design report volume 2: Physics at the ILC, SLAC National Accelerator Laboratory (SLAC), 2011 (unpublished).
 - [2] A Global Accelerator Network: ICFA Task Force Reports, International Committee for Future Accelerators, 2001 (unpublished).
 - [3] S. Gabrielli, M. Hodapp, and R. Ranon, in *Second IEEE International Conference One-Science and Grid Computing*, 2006 (IEEE, Amsterdam, 2006), p. 139.
 - [4] S. Chatrchyan, G. Hmayakyan, V. Khachatryan, A. Sirunyan, W. Adam, T. Bauer, T. Bergauer, H. Bergauer, M. Dragicevic, J. Erö *et al.*, *J. Instrum.* **3**, S08004 (2008).
 - [5] L. Taylor, E. Gottschalk, K. Maeshima, and P. McBride, *J. Phys. Conf. Ser.* **119**, 072029 (2008).
 - [6] I. Alexandrov, A. Amorim, E. Badescu, M. Barczyk, D. Burckhart-Chromek, M. Caprini, J. D. S. Conceicao, J. Flammer, B. Di Girolamo, M. Dobson *et al.*, *IEEE Trans. Nucl. Sci.* **51**, 578 (2004).
 - [7] N. Yamamoto, in *9th International Conference on Accelerator & Large Experimental Physics Control Systems*, Gyeongju, Korea (JACoW, 2003), p. 603.
 - [8] Y. Furukawa, K. Hasegawa, D. Maeda, and G. Ueno, in *12th International Conference on Accelerator & Large Experimental Physics Control Systems*, Kobe, Japan (JACoW, 2009), p. 615.
 - [9] Y. Furukawa, K. Hasegawa, and G. Ueno, in *13th International Conference on Accelerator & Large Experimental Physics Control Systems*, Grenoble, France (JACoW, 2011), p. 1193.
 - [10] 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., Sect. A* **352**, 179 (1994).
 - [11] P. Pucyk, T. Jezynski, W. Koprek, T. Czarski, K. T. Pozniak, and R. S. Romaniuk, in *Wilga-DL Tentative. International Society for Optics and Photonics* (JACoW, 2005), p. 52.
 - [12] R. Tanaka, T. Fukui, K. Kobayashi, T. Masuda, A. Taketani, T. Wada, and A. Yamashita, in *International Conference on Accelerator & Large Experimental Physics Control Systems*, 2001, Beijing, China (JACoW, 1997).
 - [13] J. Chaize, A. Götz, W. Klotz, J. Meyer, M. Perez, E. Taurel, and P. Verdier, in *8th International Conference on Accelerator & Large Experimental Physics Control Systems*, 2001, San Jose, California (JACoW, 2001).
 - [14] L. R. Dalesio, M. Kraimer, and A. Kozubal, in *International Conference on Accelerator & Large Experimental Physics Control Systems*, Tsukuba, Japan (JACoW, 1991), p. 92.
 - [15] J. Hill, E. Williams, R. Lange, A. Johnson, J. Sage, K. Furukawa, and S. Hunt, in *9th International Conference on Accelerator & Large Experimental Physics Control Systems*, Gyeongju, Korea (JACoW, 2003), p. 377.
 - [16] K.-U. Kasemir, X. Chen, J. Purcell, and K. Danilova, in *13th International Conference on Accelerator & Large Experimental Physics Control Systems*, Grenoble, France (JACoW, 2011), p. 1178.
 - [17] J. J. Garrett *et al.*, Ajax: A new approach to web applications (2005), https://courses.cs.washington.edu/courses/cse490h/07sp/readings/ajax_adaptive_path.pdf.
 - [18] M. Moodie and K. Mittal, *Pro Apache Tomcat 6* (Springer, New York, 2007).
 - [19] Y. Furukawa, in *13th International Conference on Accelerator & Large Experimental Physics Control Systems*, Grenoble, France (JACoW, 2011), p. 673.
 - [20] I. Fette and A. Melnikov, The websocket protocol (2011), <https://tools.ietf.org/html/rfc6455>.
 - [21] S. Tilkov and S. Vinoski, *IEEE Internet Computing* **14**, 0080 (2010).
 - [22] G. Richards, S. Lebresne, B. Burg, and J. Vitek, *ACM Sigplan Notices* **45**, 1 (2010).
 - [23] K. Evans and M. Smith, in *Proceedings of the 21st Particle Accelerator Conference*, Knoxville, TN, 2005 (IEEE, Piscataway, NJ, 2005), p. 3621.
 - [24] M. R. Kraimer, J. B. Anderson, A. N. Johnson, W. E. Norum, J. O. Hill, R. Lange, B. Franksen, and P. Denison, Epics application developer's guide (to be published).
 - [25] Y. Yano, *Nucl. Instrum. Methods Phys. Res., Sect. B* **261**, 1009 (2007).
 - [26] A. Uchiyama, K. Furukawa, Y. Higurashi, and T. Nakagawa, *Rev. Sci. Instrum.* **85**, 02A904 (2014).
 - [27] A. Technologies, Akamai's state of internet Q1 2014 report, Vol. 7 (2013), p. 14.
 - [28] H. Okuno, N. Fukunishi, and O. Kamigaito, *Prog. Theor. Exp. Phys.* (2012) 03C002.