DOI: 10.17199/NOBUGS2016.70

IROHA2: Standard instrument control software framework in MLF, J-PARC

Takeshi Nakatani[1], Yasuhiro Inamura[1], Takayoshi Ito[2] and Kentaro Moriyama[2]

[1] Materials and Life Science Division, J-PARC Center, Japan Atomic Energy Agency (JAEA), 2-4 Shirakata, Tokai-mura, Naka-gun, Ibaraki, Japan
[2] Neutron Science and Technology Center, Comprehensive Research Organization for Science and Society (CROSS), 162-1 Shirakata, Tokai-mura, Naka-gun, Ibaraki, Japan

E-mail: takeshi.nakatani@j-parc.jp

Abstract. We have developed the MLF standard instrument control software framework which is called "IROHA2" by upgrading IROHA. IROHA2 consists of the device control server for operating, monitoring and logging devices, the instrument management server for measurement management, configuration of an instrument and user authentication, the sequence management server for an automatic measurement and the integrate control server for administration of an instrument. Because each software component of IROHA2 has a web interface, we are able to use its all functions under multi-platform environment via a web browser. For the remote access environment in MLF, IROHA2 is also connected to and cooperated with several external systems which are the measurement monitoring system, the authentication system and the information linkage database.

1. Introduction

Neutron and muon experimental instruments in Materials and Life Science Experimental Facility (MLF), Japan Proton Accelerator Research Complex (J-PARC) carry out many kinds of measurements and produce an enormous number of experimental data, such as raw data and metadata representing measurement conditions, by a lot of external users from not only academic but also industry. It is important to be user-friendly and automated control software to perform efficient measurements. In the beginning of MLF, we have developed the MLF standard instrument control software framework for the integration of data acquisition (DAQ) and device control [1, 2]. The name of the software framework is "IROHA". Origin of the name IROHA is the old order of the Japanese alphabet like "ABC" in English. After the usage of IROHA for several years, we have upgraded IROHA to the next generation control software framework which is called "IROHA2" [3]. IROHA2 consists of four core software components, i.e. the device control server to control and monitor each device, the instrument management server to authenticate a user, manage a measurement and configure an instrument setup, the sequence management server to do an automatic measurement and the integrate control server to unify instrument control and monitoring. We can use all the functions of these software components with its web interface. We realized the multi-platform control environment via a web browser [4]. Currently, IROHA2 has supported many kinds of devices (over 50 devices including subtle difference), for example, a beam narrower, a chopper, a goniometer and a temperature controller and has been installed and operated in several neutron and muon experiment instruments in MLF.

2. Software components

The common architecture and implementation of IROHA2 were written in the previous paper [3]. IROHA2 can be run on a relatively low performance computer shown in table 1.

Table 1. Required specifications for IROHA2.

Item	Details
CPU	Xeon 1.86GHz
Memory	8GByte
Hard disk drive	2TB (with log file)
Operating system	Linux (64bit)
Runtime environment	Python 2.6

There are four core software components in IROHA2 shown in figure 1. The functions of the device control server and the instrument management server were mentioned in the previous paper [4]. The functions of the sequence management server and the integrate control server are as follows. The data linkage between the instrument management server and the MLF external systems will be mention in the next section.

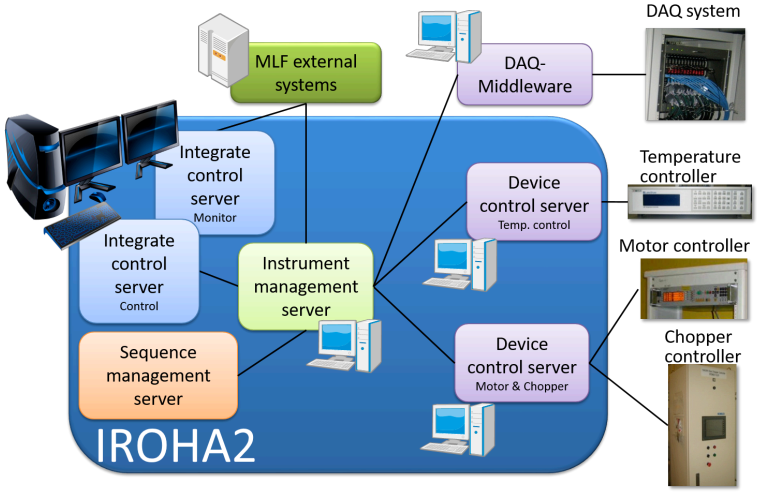

Figure 1. IROHA2 software components (Device control server, Instrument management server, Sequence management server and Integrate control server)

2.1. Sequence management server

The sequence management server has many kinds of functions which is necessary in an automatic measurement. We can edit a sequence for an automatic measurement with a drag and drop operation, graphically confirm a sequence by visualization of changing parameters, run control (Begin / End / Pause / Resume) a sequence and monitor progress of a sequence on a web user interface. In addition, if a new device is attached to an instrument, we can generate operating commands of the device by obtaining the parameter list from its device control server via a web browser. The sequence is available for nested loop structure by numerical increasing parameters as well as list data in Python. Because each command of the sequence can return values of the result, we can do feedback measurement. Because the sequence is running in background process by the Python Debugger, we can change the parameters of the sequence dynamically.

2.2. Integrate control server

The role of the integrate control server is integration of an instrument. We can run control not only a DAQ system but also the sequence management server, operate the devices and monitor all of the information which is necessary in a measurement, for example, the DAQ status, the device status of the present values and the trend graphs, the proton beam power and the facility status. Because the web user interface of the server is able to be customized flexibly, the instrument scientists can perform access control (visible / invisible and operational / not operational) by each parameter of the devices in their instrument.

We can select the operation mode when the integrate control server is started. The modes are two types, which are "Control mode" and "Monitor mode". The "Control mode" is that the server is run as mentioned above. The "Monitor mode" is that the server collects the status and the parameters from the other servers and produces the static web page files at stated periods. The files are transferred to the measurement monitoring system which is accessible from Internet. The users can monitor the present measurement status with a web browser on their own Internet connection device after the authentication.

3. Data linkage

One of the main functions of the instrument management server is producing run information which is recorded each measurement result. The run information is included the location of the raw data, the user, the theme, the sample, the devices with their parameters and the measurement time, which are typically written in an experimental logbook. In MLF, the location of the raw data is configured by the configuration file of the DAQ middleware, the user is authenticated by the login process to IROHA2, the information of the theme and the sample is obtained from the users' manual inputs or the user proposal and support system in J-PARC Users' Office, the information of the devices is collected from the device control servers and the time is recognized from the system clock adjusted by NTP. The data flow between the server and the external systems is shown in figure 2. When the user logins to IROHA2, the instrument management server connects the authentication system by LDAPS, load the user data and certificates the user. To obtain the information of the theme and the sample, the server connects the information linkage database in cooperation with the user proposal and support system and load the data of the theme and the sample by the Oracle Instant Client [5]. After the run information recorded by the server is archived and cataloged in the MLF experimental database [6], the users can access their run information via a web browser from inside of the facility as well as their home laboratory.

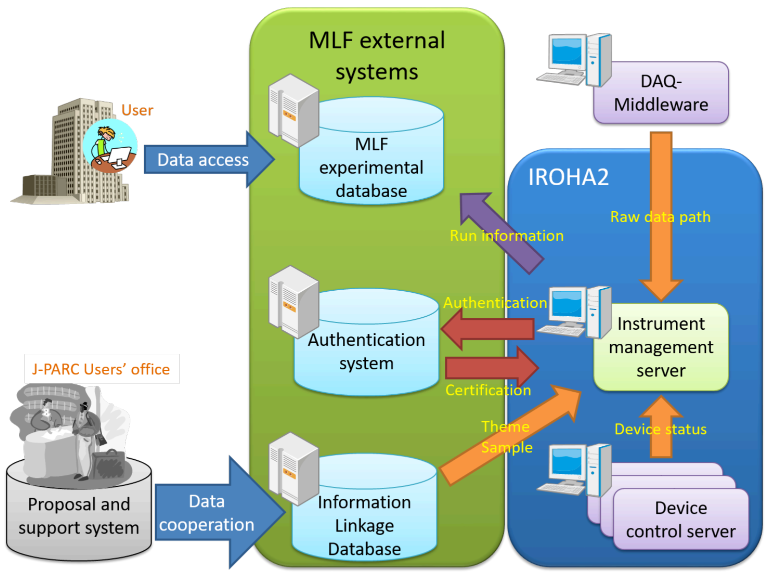

Figure 2. Data flow diagram between IROHA2 instrument management server and the MLF external systems.

4. Conclusion

We have upgrade and operated the next generation MLF standard instrument control software framework called "IROHA2". Because IROHA2 can automatically produce the information written in an experimental logbook, we will realize that the users in MLF can do "logbook-less" experiments.

Acknowledgments

We acknowledge the valuable discussions with and suggestions of the computing environment group in MLF.

References

[1] Nakatani T, Inamura Y, Ito T, Harjo S, Kajimoto R, Arai M, Ohhara T, Nakagawa H, Aoyagi T, Otomo T, Suzuki J, Morishima T, Muto S, Kadono R, Torii S, Yasu Y, Hosoya T and Yonemura M 2009 *Proc. of ICALEPCS2009* 673-675.

[2] Nakatani T, Inamura Y, Ito T, Moriyama K and Otomo T 2014 *JPS Conf. Proc*. 1, 014010.

[3] Nakatani t, Inamura Y, Ito T and Otomo T 2015 *Proc. of ICANS-XXI (JAEA-Conf 2015-002)* 493-496.

[4] Nakatani T, Inamura Y, Ito T and Otomo T 2015 *JPS Conf. Proc.* 8, 036013.

[5] http://www.oracle.com/technetwork/database/features/instant-client/index.html

[6] Moriyama K and Nakatani T 2015 *Proc. of ICALEPCS2015* 834-837