2017 International Conference on Cyberworlds

IoT Remote Group Experiments in the Cyber Laboratory:
A FPGA-based Remote Laboratory in the Hybrid Cloud

Norihiro Fujii
Faculty of Health and Medical Science
Teikyo Heisei University
Toshima-ku, Tokyo, JAPAN
nfujii@thu.ac.jp

Nobuhiko Koike
Faculty of Computer and Information Sciences
Hosei University
Koganei-shi, Tokyo, JAPAN
koike@k.hosei.ac.jp

***Abstract*— In accordance with the resent advancement in Internet of Things (IoT), the needs for IoT experiment platform have been ever increasing. IoT system consists of various technologies such as networking, sensor controller, edge-side computing, server-side big data collections, analysis and their visualizations. An experimental environment that can handle the development and experiments of such an IoT system become important. In the IoT system, a highly flexible system structure for applications using Field Programmable Gate Array (FPGA) is required. The authors propose the Remote Laboratory System for handling IoT experiments in the Cyber Laboratory, which is an educational FPGA-based remote laboratory for under-graduate university students. It enables not only to use available board-level small computers but also to use FPGA boards for prototyping IoT edges. It can also organize the IoT cloud-server side programs in the hybrid cloud. The FPGA based edge-side computing approach can have much more freedom and flexibility to implement various sensor controls those can be customized for specific IoT applications. The use of free micro-processor IP-core and re-organizing the available FPGA CAD design platform allow us to reduce the burden of design and implementation efforts for the construction of new Cyber Laboratory to accommodate IoT designs and experiments. It also contributed to reduce the students' amount of efforts to conduct their own IoT design and experiments, where students are required to have various skills in Information Technologies (IT): hardware design, edge-side computing, server-side computing, networking and infrastructure construction. The use of the Docker container/Swarm and the Docker File contributed to construct their own IoT experiment platforms for every student automatically, in the form of "Infrastructure as Code". Furthermore, these separately designed IoT experiment platforms can be combined to conduct a group of group experiment simultaneously. The paper showed the Cyber Laboratory's usefulness and applicability for IoT kinds of Remote experiments.**

Keywords- Remote laboratory; FPGA hardware design laboratory; IoT System design laboratory; Group-Group experiment; Hybrid cloud; Docker container

I. INTRODUCTION

The development of the IoT System is said to be the Fourth Industrial Revolution (Industry 4.0) [1]. IoT is getting its popularity and more and more IoT devices have been applied and utilized for various kinds of industrial fields. IoT system contains many IT technologies such as hardware design, edge computing, cloud server-side programing and networking technologies, as shown in the Figure 1. So, IoT experiment laboratory should become a suitable theme, which should be included in the curriculum for under graduate computer science students.

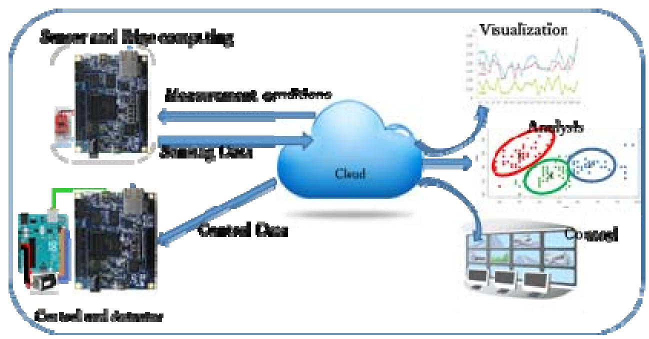

Figure.1 IoT System

The paper proposes the realization of a number of IoT experiment systems contained in the Cyber Laboratory [2, 3], which is a FPGA-based hardware laboratory implemented in the hybrid cloud. The original Cyber Laboratory takes advantage of public cloud's scale out capability and on premise private cloud's free connectivity of any I/O devices, namely IoT devices implemented in FPGAs [4]. In order to conduct IoT design and experiment, various kinds of IT skills are required. For IoT experiment, FPGA plays important rolls both for edge side and for cloud server side processing. The sensor controller in the FPGA receives commands from the server and obtains sensor data, and sends them to the server via the Internet [9]. Then, the cloud servers with the help of FPGAs, perform accumulation, analysis and visualizations of the measured data [10].

In order to realize such an IoT system level experiment environment, each student has to prepare IoT edge device designs described in Verilog-HDL and edge side computer programs for built-in IP cores first. Also, the cloud side IoT server programs should be prepared. For the skilled students, server side FPGAs can be useful to accelerate their server tasks. The infrastructure construction phase comes

DOI 10.1109/CW.2017.29

next and it used to be an obstacle. Because, the infrastructure construction task has to be handled by administrators by making use of the paper documents provided by students. The infrastructure work includes many cumbersome tasks such as, multiple IoT device setups, cloud-side server setups and connection among them. In order to build such an IoT experiment system automatically, the combined use of Docker Containers [5], Docker Swarm [11] and Docker Files has been employed. If the course administrator once prepares a kind of template as the Docker File written in Go codes, every student can easily understand the basic system construction processes in the form of the templates written in GO code and can easily customize them for his/her own specific IoT system.

The Docker Swarm can handle available FPGA setup/run containers, which are connected separate FPGA boards, and allocate these idle FPGA board and container pares to every students in a space division fashions [7]. Also, the Docker Swarm finds out available cloud side servers and allocate student server containers, these can be joined in the student IoT experiment system. In case of existing remote laboratory projects where FPGA devices should be exclusively occupied by singleton users for entire experiment periods and they cannot resolve the device usage conflicts for a long time and results in poor FPGA device usages and long waiting queues [6, 8]. On the other hand, as the Cyber Laboratory [2] allows on demand FPGA uses and corresponding singleton containers occupy FPGAs during FPGA run periods only, an efficient FPGA usage can be realized.

The gang scheduling can prevent conflicts in utilizing of IoT devices in a space division fashion. It is possible to share FPGAs without spatially overlapping. Furthermore, by combining sharing FPGAs in time division fashions with queueing mechanism, an effective utilization of finite number of FPGA devices can be achieved.

After all IoT edge side containers and server side containers have been started, they will all wait for the arrival of the experiment start signals from the Docker Controller in the form of the Web Services or Web API/Socket. In this way, students can build up their skills related to FPGA hardware design, sensors/actuators, networking, server-side programing, and big-data, without going into cumbersome system infrastructure details.

II. Cyber Laboratory and the Experiment for IoT System

Figure 2 shows Cyber Laboratory for IoT system as the remote experimental environment. The use of the Docker Containers also contributed to reduce the system development cost and man-power for realizing the new Cyber Laboratory. For the design and preparation phase, a free FPGA design automation tool is inevitable and useful. However, as available software are assembled as a monolithic software process, it should be separated into a number of individual services, and realized as a set of separate micro services, these can be implemented in separate containers. The connections among micro services can be realized in the form of the Web Services/Socket or the Web APIs. It becomes so called "micro service architecture", and an optimal micro service allocation between the public cloud and the on-premise private cloud, these are determined according to the "closeness" to the specific devices. Most micro services except FPGA setup/run services can be allocated to the public cloud. In this way, the Cyber Laboratory can cope with the dynamically changing student workload by taking scale-out or shrinking strategies. On the other hand, the device dependent micro services, such as FPGA setup/run service should be allocated for the on-premise private cloud. As for the IoT cloud server containers, they can be allocated either for the private cloud or for the public cloud. As the work load of the on-premise private cloud computers are rather light during the experiment run time, they can handle the IoT server side containers as well at the same time. However, if the IoT events should become huge, it is possible to migrate these server containers to the public cloud side as well.

As plural users should use the system at the same period of time, where each user would utilize a number of edge devices (FPGAs) and a number of cloud servers. Each physical computer contains up to 8 FPGA devices and associated device setup/run containers. If the users should prepare the cloud servers as well, corresponding server containers can be allocated as well. The Docker Swarm controller handles the container allocations across the on-premise laboratory servers. It also sets up the networking and connections among containers. When all requested containers have been allocated, the Swarm controller issues a set of experiment start messages to all participating containers via either the Web Services/Socket or Web APIs.

If a group of students' group experiments is requested, the Swarm controller put each student's requests into the rendezvous queue until all participating student's requests should arrive. When all requests arrived, all participating containers are allocated to the laboratory servers in the gang scheduling fashion. Thus, the group of group IoT experiment can be realized.

III. System Design and Its Configuration

Students can design and experiment their own IoT systems using quite a few FPGA boards and cloud servers in the hybrid cloud environment. Students can develop and conduct experiments independently or in a group experiment fashion.

Each student is requested to design and experiment his/her own group experiment which should contain a few IoT edges and a few cloud servers. For IoT edges, the student should prepare Verilog-HDL for each FPGA and the IoT edge control programs as well. For the cloud servers, the

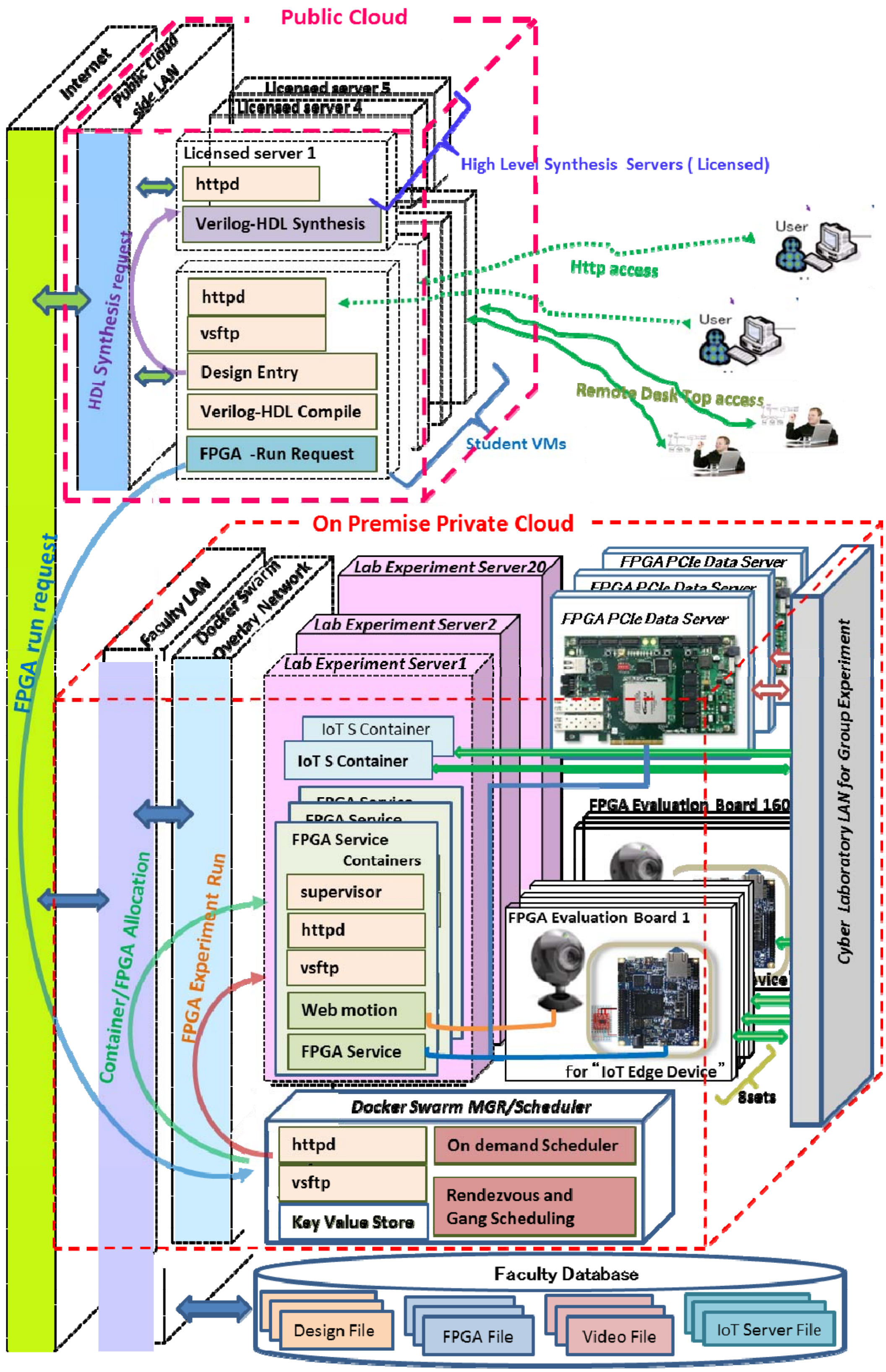

Figure.2 IoT Remote Group Experiments system in the Cyber Laboratory System

event reception and processing programs become necessary. Rest of the works including networking and the infrastructure organization can be automatically handled by the Docker File that realizes the “Infrastructure as Code”. For skilled students, it is possible to include new features and functionalities for both edges and cloud servers, by modifying the basic provided Docker Files. Thus, continuous integration and deployment can be realized.

The experimental environment for the assumed IoT system can contain various kinds of IoT edges implemented in different FPGAs, these would be separately designed by quite a few students. So, that becomes the group of group experiments these should be handled by the Docker Swarm controller. Each student’s group experiment request is first put into the rendezvous queue. When all the group experiment requests have arrived, the controller allocates necessary containers to laboratory servers and set them up to run. The actual group of group experiment should be handled by the Swarm controller by issuing a set of messages in the form of the Web Services/Socket via the http channel. The use of Docker Container and Docker Swarm contributed to construct a flexible and lightweight virtual machine PC cluster and can overcome the physical number limitation of laboratory servers. Also, the micro service architecture contributed to offload the workload of the on-premise private cloud to the public cloud, without changing the design automation process structure. After the monolithic design automation system have been decomposed into a collection of micro services, a flexible and easy service allocation can be realized across the public cloud and the on-premise private cloud in both functional distributed and load-balanced ways. The device dependent services namely the FPGA setup/run containers should still be kept in the on-premise public cloud, where each singleton container should make a pair with each attached FPGA board.

Since the workload of a FPGA setup/run container is rather lightweight, up to eight such FPGA board and its setup/run container pairs can be serviced by single laboratory server. That makes up to 160 FPGAs as a whole and they can be available to participate in an experiment, which should be more than enough for students’ group of group IoT experiment.

IV. CONCLUSIONS

The authors proposed the remote experimental environment for IoT system design and experiment in the Cyber Laboratory. It is possible to perform remote group experiments simultaneously. This functionality is inevitable for IoT system experiment, where a number of FPGAs and cloud servers are necessary. Cyber Laboratory makes it possible to utilize FPGAs effectively in both development phase and experiment of Complex structure of IoT System. The Cyber Laboratory have realized an efficient and effective FPGA based remote laboratory for IoT experiments.

In the IoT system, edge device implements edge computing and sensor controls, these handle various sensor devices. Furthermore, these IoT group experiments can be grouped together again to form a group of group experiments. They are put in the rendezvous queue until all group experiment requests have arrived. Then they are all allocated to the physical laboratory servers in the gang scheduling fashion.

The micro service architecture together with the Docker container/Swarm realized an efficient and scalable FPGA-based IoT system remote laboratory in the hybrid cloud.

REFERENCES

[1] Industry4.0: https://www.weforum.org/about/the-fourth-industrial-revolution-by-klaus-schwab, retrieved at April, 2017

[2] Nobuhiko Koike, “Concurrent Remote Group Experiments in the Cyber Laboratory: A FPGA-based Remote Laboratory in the Hybrid Cloud” *Springer-Verlag Berlin Heidelberg 2011*

[3] Yuichi Toyoda, Nobuhiko Koike and Yamin Li, “An FPGA-based Remote Laboratory: Implementing Semi-Automatic Experiments in the Hybrid Cloud”, *14th international Conference on remote engineering and Virtual Instrumentation (REV2016)*

[4] T. Gomes, S.Pinto, A. Tvares and J.Cabral, “Towards an FPGA-Based Edge Device for the Internet of Things”, *Emerging Technologies & Factory Automation (ETFA), 2015 IEEE 20th Conference on*

[5] Docker: https://docs.docker.com/, retrieved at April, 2017

[6] Jasveer Singh T. Jethra, Sachin B. Patkar and Shamik Datta, “Remote Triggered FPGA based Automated System”, *11th International Conference on Remote Engineering and Virtual Instrumentation (REV2014),* pp.309-314, 26-28 Feb. 2014

[7] Nobuhiko Koike, “Cyber laboratory: Migration to the hybrid cloud solution for device dependent hardware experiments”, *Information Technology Based Higher Education and Training (ITHET), 2014*, pp.1-5, 11-13, Sept. 2014

[8] F. Morgan, S. Cawley, M. Kane, A. Coffey and F. Callaly, “Remote FPGA Lab Applications, Interactive Timing Diagrams and Assessment”, *Irish Signals & Systems Conference 2014 and 2014 China-Ireland International Conference on Information and Communications Technologies (ISSC 2014/CIICT 2014). 25th IET*, pp.221-226, 26-27 June 2013

[9] P. G. Lopez , A. Montresor, D. Epema, A. Datta, T. Higashino, A. Iamnitchi, M. Barcellos, P. Felber and E. Riviere, “Edge-centric Computing: Vision and Challenges”, *in ACM SIGCOMM Computer Communication Review*, Volume 45, Number 5, October 2015, pp.37-42

[10] Adrian M. Caulfield, Eric S. Chung, Andrew Putnam, et al., “A Cloud-Scale Acceleration Architecture”, *2016 49th Annual IEEE/ACM International Symposium on Microarchitecture (MICRO)*

[11] Docker Swarm: https://docs.docker.com/engine/swarm/, retrieved at April, 2017