Softwarized and Open OLT Architecture for Flexible Optical Access Network

Keita Nishimoto, Takahiro Suzuki, Kota Asaka, Jun-ichi Kani, and Jun Terada
NTT Access Network Service Systems Laboratories, 1-1 Hikarinooka Yokosuka-shi, Kanagawa 239-0847, Japan
keita.nishimoto.wr@hco.ntt.co.jp

Abstract: Recently, many telecom carriers are promoting the re-architecture of access networks and COs by utilizing SDN/NFV and OSS. We present our research relevant to the software PON-OLT architecture that we proposed for further flexibility. © 2020 The Author(s).
OCIS codes: (060.4510) Optical communications, (060.4250) Networks

1. Introduction

Telecom carriers have been actively promoting research into raising the flexibility of their own optical access networks and central office (CO). NTT, for instance, proposed a next generation access network equipment architecture aiming at flexible service instantiation on the network by combining various functional components [1]. AT&T and ONF (Open Networking Foundation) also announced the launch of CORD (Central Office Re-architected as Datacenter) project in 2016 [2]. CORD aims at re-architecting COs, originally realized as dedicated hardware, to achieve cost-efficient and flexible optical access networks by utilizing commodity hardware as well as SDN/NFV technologies.

2. Softwarization and Disaggregation of PON-OLT

It is important to softwarize PON-OLT and open up APIs between hardware and software for realizing flexible optical access networks and COs. PON-OLT controls and manages its ONUs in the subscribers' home, and connects them to the network in CO. We define the term "Softwarize" in this paper as a concept that transforms a hard-coded/firmware-coded function coupled with dedicated hardware into software that can run on a commodity platform (e.g. x86 server) and chipset (e.g. x86 CPU, GP-GPU, etc.), and further disaggregates the software from dedicated hardware (e.g. ASIC) platform logically or physically (Fig. 1 (a)). Softwarization enables easy development for new functions and services, and also cost-efficient network equipment management without caring about the specification in the hardware and communication method. For example, AT&T and ONF succeeded in disaggregating the OLT management function from its hardware in the R-CORD (Residential CORD a.k.a. SEBA) project. Specifically, they softwarized OLT's management layer such as OMCI stack and ONU authentication functions etc., and set them in a part of C-Plane software that connects OLT devices via open APIs [3]. They succeeded in rapidly developing these C-Plane software components called VOLTHA and ONOS through the support of the open software community. We believe telecom carriers/operators can enjoy the following two benefits by opening up those APIs and promoting community-based software developments after softwarizing their network equipment: (1) enabling rapid and cost-efficient development for new services, (2) expanding hardware options.

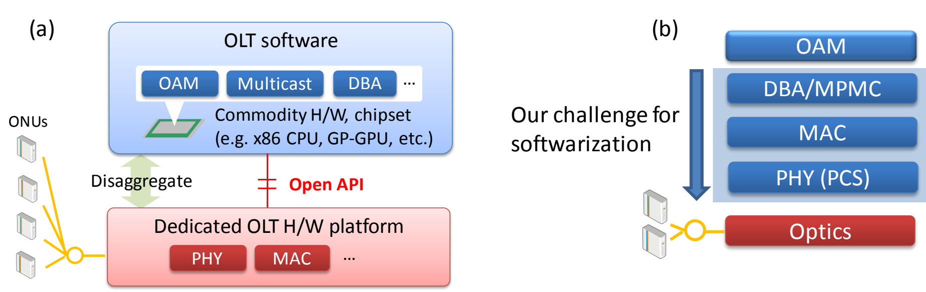

Fig. 1. (a) Concept of PON-OLT softwarization, (b) Our challenge for "extreme" softwarization.

We, NTT Access Network Service Systems Laboratories [1, 4-7], and some other research groups [8, 9] have also tackled the softwarization of PON-OLT. We have advanced existing studies/projects with work on softwarizing and standardizing the APIs not only for the management layer, but also PON-basic functions such as Dynamic Bandwidth Allocation (DBA), Multi-Point Mac Control (MPMC), and PHY processing (Fig. 1 (b)). Our research aims to softwarize PON-OLT functions to the "extreme" level so as to achieve further flexibility including physical processing and a chance to select various QoS functions to satisfy customer demands. Also, expanding the software-implemented part will lead to the cost-reduction of OLT hardware, and will minimize its size. We expect these advantages will enable us to build optimal PON systems in various environments such as mobile front haul, offices, power-plants, Data Centers. Our proposed flexible PON-OLT architecture can meet the varied requirements of these sites, which are quite different from FTTH/FTTO (i.e. the main target of conventional PON system).

We address the following three technical problems in the next section towards PON-OLT softwarization: (1) Guarantee of time-accuracy, (2) Accelerated PHY layer processing, (3) Integrated management for distributed hardware and software components. We introduce our solutions to the problems, and elucidate the technical issues that remain to be solved.

3. Technical problems and solutions

3.1. Guaranteed time-accuracy

Among the basic TDM-PON functions, DBA is the function that most strongly impacts the subscriber's QoS (e.g. throughput, latency, fairness, etc.) since it controls the upstream bandwidth of each ONU [10]. For example, some researchers have proposed extremely low-latency DBA for realizing the TDM-PON-based mobile front haul that is quite different from existing systems [11]. We thus consider the DBA softwarization that enables flexible algorithm swapping or alteration is essential for providing optimized QoS levels in various environments.

However, the time-critical PON-OLT functions, such as DBA and power-saving functions, have mainly been implemented on dedicated hardware to achieve the strict time-accuracy requirements as regards processing time. We need to guarantee the time-accuracy when implementing DBA as software and running it on commodity platforms that work more slowly with larger jitter than dedicated ones. DBA function must send out bandwidth-grant messages (GATE) to ONUs and receive bandwidth-request messages (REPORT) from ONUs with low jitter (A) and low latency (B). The upstream signals can collide due to the jitter in the upstream transmission of each ONU if DBA does not satisfy requirement (A) [4]. Regarding to requirement (B), its violation increases the upstream queueing time triggering buffer overflows because each ONU has to wait a long time for its GATE message after sending its REPORT message to DBA.

For meeting requirement (A), we have proposed the use of DPDK, it enables software-packet handling with low jitter [5], and a hardware/software cooperation method [7]. In the cooperation scheme, the dedicated hardware side undertakes GATE/REPORT-handling (i.e. time-stamping, RTT calculation, etc.) while the software side calculates and schedules the upstream bandwidth allocated to each ONU. This function split approach implements the architecture named "Mini-PON" (Figure 2(a)) and thus retains the flexibility offered by softwarization while preventing upstream signal collision [7]. We have standardized the information structure exchanged between hardware and software in DBA-API TR-402 [13] / 403 [14] in BBF in order to ensure the cooperation of these loosely-coupled components.

Disaggregating DBA software from hardware makes it difficult to meet requirement (B), especially when we distribute hardware sets and ONUs distant from DBA software. Some researchers including us have tried to solve this technical issue by developing predictive DBA methods that grant additional bandwidth to ONUs just prior to the arrival of upstream traffic [6], [12]. These methods allow ONUs to rapidly forward newly arriving upstream packets even if requirement (B) is not satisfied.

We consider there are several technical challenges for implementing DBA as software in real networks although we have demonstrated the satisfaction of both requirements (A) and (B) by our prototypes and simulations. For instance, future work must realize the QoS assurance of control messages in the network between softwarized DBA and hardware platform when those messages share physical links with user-data traffic, as well as improving the scalability to handle greater numbers of OLTs and ONUs connected to the network.

3.2. Accelerated PHY-Layer processing

PHY Layer processing is a significant issue in the realization of the full softwarization of PON-OLT functions. Those functions such as FEC, scrambling, frame-synchronization are usually implemented on ASICs for adequate processing speed. We thus need a technology that accelerates software-processing of those functions on general purpose hardware. We selected the GP-GPU as an accelerator and proposed several methods specific to it, for instance, a DMA-transfer method between PON-IF and GP-GPU, task distribution for parallel processing, and CPU/GPU cooperation method [15] (Fig. 2 (b)). Our prototype successfully demonstrated it could attain the throughput necessary for 10GE-PON PHY layer even though it performed all processes in software. As a next step, we are now trying to reduce the processing latency.

3.3. Integrated management of distributed hardware and software

Softwarization will give us an option to physically distribute the software and hardware forming PON-OLTs. Furthermore, the types of hardware components will further diversify given the adoption of commodity CPU in Section 3.1, and GP-GPU in Section 3.2. We thus need a method for managing the softwarized OLT architecture, i.e., how to compensate the differences in those components and integrate them into a single virtual OLT (Fig. 2 (c)). At present, ONF and BBF are developing VOLTHA (Virtual OLT Hardware Abstraction) [3] and OB-BAA (Open Broadband-Broadband Access Abstraction) [16], respectively, for realizing both the vendor-agnostic and access technology-agnostic management of access network devices. We are now investigating a management method for the softwarized OLT by utilizing these architectures.

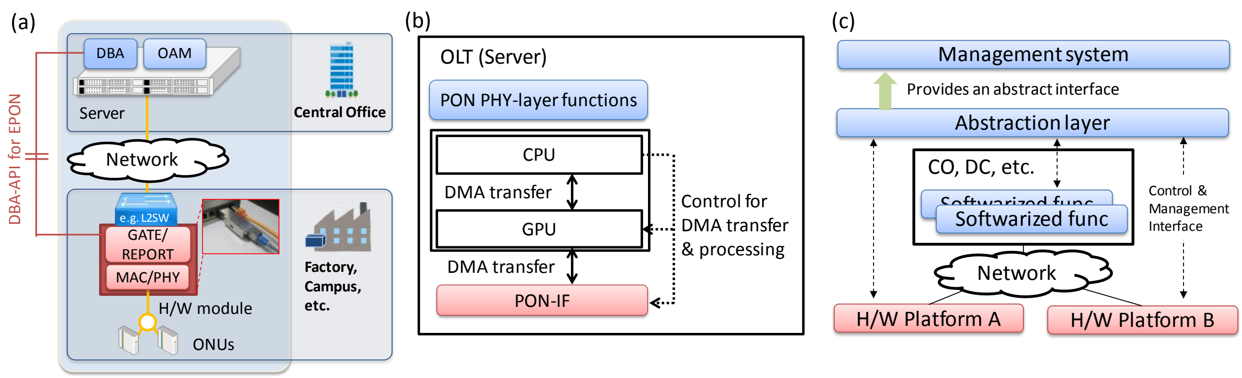

Fig. 2. (a) Concept of Mini-PON, (b) CPU/GP-GPU-based PON-OLT PHY layer processing architecture, (c) Management architecture for distributed hardware/software components for OLTs.

4. Conclusion

This paper showed how telecom carriers have been improving the flexibility of optical access networks and COs, and the trends in PON-OLT softwarization as a key-technology. We then introduced our research progress towards the softwarization of PON-OLT's basic functions (i.e. DBA, PHY processing) for further flexibility in optical access networks and remarked on the future challenges.

5. References

[1] J. Kani et al., "Flexible Access System Architecture (FASA) to Support Diverse Requirements and Agile Service Creation," JLT, 36 (8), pp. 1510-1515 (2018).
[2] L. Peterson et al., "Central office re-architected as a data center," IEEE Com. Magazine, 54 (10), pp. 96-101 (2016).
[3] S. Das, "SEBA: SDN Enabled Broadband Access," ONF Connect 2018 (2018).
[4] K. Nishimoto et al., "Virtualization of EPON OLT functions and collision suppression techniques for Multi-Point MAC control," Proc. OFC 2016, W3F (2016).
[5] K. Nishimoto et al., "Implementation of Software-based EPON-OLT and Performance Evaluation," IEICE ComEX, 6 (7), pp. 467-472 (2017).
[6] K. Nishimoto et al., "Predictive Dynamic Bandwidth Allocation Based on the Correlation of the Bi-directional Traffic for Cloud-based Virtual PON-OLT," Proc. IEEE CQR 2017, pp. 1-6 (2017).
[7] T. Tochino et al., "First demonstration of Mini-PON by using standard-based DBA-API and software components on commodity server apart from modular OLT," Proc. ECOC, TuA3 (2019).
[8] M. Ruffini, F. Syne, "Moving the Network to the Cloud: The Cloud Central Office Revolution and Its Implications for the Optical Layer," JLT, **37** (7), pp. 106-1716 (2019).
[9] S. Okamoto et al., "Logical optical line terminal technologies towards flexible and highly reliable metro and access integrated networks', Proc. SPIE, 10129 (2017).
[10] G. Kramer et al., "Interleaved Polling with Adaptive Cycle Time (IPACT): A Dynamic Bandwidth Distribution Scheme in an Optical Access Network," Photonic Network Communications, 4 (1), pp 89–107 (2002).
[11] T. Tashiro et al., "A novel DBA scheme for TDM-PON based mobile fronthaul," Proc. OFC, Tu3F (2014).
[12] R. Yasunaga et al., "Modular Dynamic Bandwidth Allocation for a Flexible PON: Concept and Evaluation," Proc. ECOC 2018, 2018.
[13] TR-402, "Functional Model for PON Abstraction Interface," (2018).
[14] TR-403, "PON Abstraction Interface for Time-Critical Applications," (2018).
[15] T. Suzuki et al., "Software Implementation of 10G-EPON Downstream Physical-Layer Processing Adopting CPU-GPU Cooperative Computing for Flexible Access Systems," IEEE Access, 7 (1), pp.33888-33897 (2019).
[16] TR-384, "Cloud Central Office Reference Architectural Framework," (2018).