Configuration and Monitoring of the Optical Physical Layer Using Software-Defined Tools

Yuliya Verbishchuk, Cormac Sreenan, Fatima Gunning

1. *Tyndall National Institute, Lee Maltings Complex Dyke Parade, Cork, Cork, T12 R5CP2.*
2. *Department of Computer Science / 3. Department of Physics, University College Cork, Cork, Ireland*

yuliya.verbishchuk@tyndall.ie

Abstract: In this paper we discuss the need for implementation of SDN-based elements and tools towards disaggregated physical layer devices for optimum management of resources.

OCIS codes: 060.4510(Optical communications), 060.4256 (Networks, network optimization)

1. Introduction

Abstraction in computer science neither appreciates or distinguishes between devices that manipulate photons or electrons. Instead it treats them as abstract resources that store, transfer and manipulate data. Communications networks are similar above layer 3. From an applications and services perspective, particularly with the deployment of 5G, a high-performance physical layer infrastructure is assumed as a given. The intricacies of how the physical layer works and the nature of its constituent parts are less important; the abstraction of data storage, transfer and manipulation are more important. Optical networks have a quasi-static physical layer (milliseconds) compared to the fast-dynamic rates of bit manipulation on computing devices (ps). The reconfiguration of data transmission paths across optical fibres – each one providing several dozen THz of bandwidth and servicing eighty or more wavelength carriers (channels) of 6.25-100GHz apart is a slow process. Each wavelength carrier can host a multitude of clients' services that are *coarsely-switched* by ROADMs and more *finely-forwarded* by electronic packet switches. Control and management of the data routing is straightforward from the physics/engineering perspective, via distributed control protocols such as Multiprotocol Label Switching (MPLS) or Generalized Multiprotocol Label Switching (GMPLS) and more recently centralised, software-defined network controllers. The flow management of data traffic was optimised in the recent past with proprietary, vendor-specific software, to maintain service level agreements. But now multi-vendor, open architectures are common and promotes interoperability issues across a variety of equipment types and functions shown in Figure 1 below.

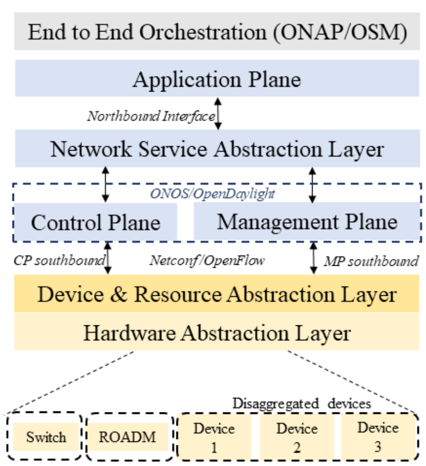

Figure 1. System Architecture within SDN framework defined in [1]

But data volumes are increasing, and the network infrastructure must keep pace and incorporate a reconfigurable and adaptive physical layer that can respond to demand. Vendor lock-in must be deprecated; common device configurations promoted; configuration and operational data separated; and support for network-wide device- and service-level atomic transactions and rollback capabilities exposed [2] to simplify network management. The Software defined networks (SDN) approach is tackling these challenges head-on: it aims to have "the capacity to initialize control, change, and manage network behaviour dynamically via open interfaces" [1]. Open APIs and open software applications are now able to oversee the network by creating appropriate abstraction layers that distinguish between the data forwarding plane; the control plane, and the management plane allowing for fast innovation.

From our computer science (CS) perspective, we demonstrate how SDN can expose and 'unlock' current and legacy component-level interfaces within open, disaggregated physical network functions (PNFs). We also suggest how this can fashion novel, energy efficient flexible photonic/electronic devices. Allowing dynamic and reconfigurable PNFs which can respond to bandwidth demand proportionate to energy consumption -- a feature of datacentre computing servers for over a decade [3]. In this talk we will review current trends in optical disaggregation and illustrate through lab experiments with disaggregated devices utilising protocols such as Netconf; YANG models and the ONOS controller.

2. Software defined networks (SDN) and hardware abstraction disaggregation

Figure 1 describes how SDN is stratified into three planes: (1) *application* – responsible for services that specify network behaviour; (2) *control / management* – responsibilities include configuration, telemetry, fault processing/resolution and decision making with regards to switching wavelengths and/or forwarding packets; and communicating these decisions to and from data plane devices. This is the realm of the controllers like ONOS and Open Daylight; (3) *data* - composed of network device elements, both physical and/or virtual, to forward and switch data [1].

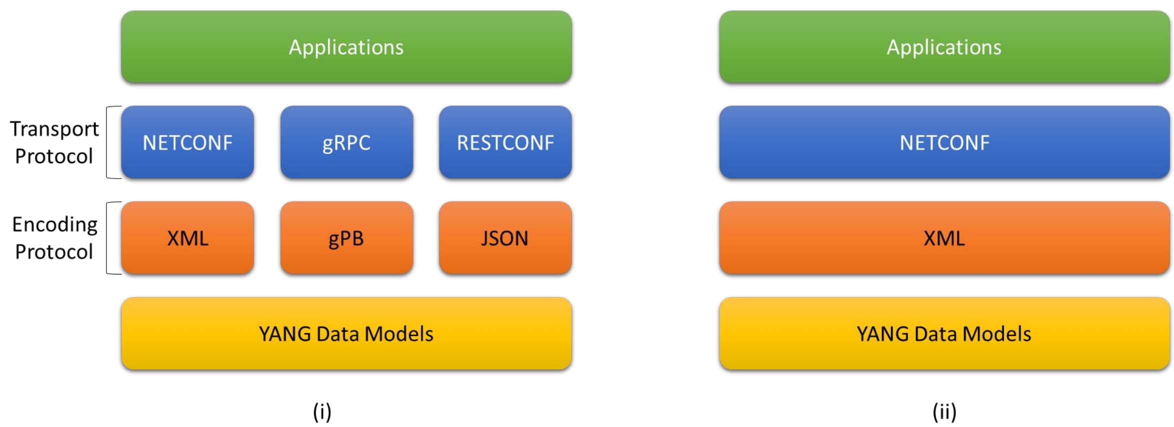

Figure 2. (i) Architecture choices for transport and encoding; (ii) One Implementation.

Typically, the atomic devices and components that collectively inform the behaviour of a data plane device element are grouped and abstracted through a hardware abstraction layer (HAL) that is vendor-specific. The underlying data model is closed, often opaque and so is limited to what is uncovered through a restricted application programming interface (API). The YANG language and the associated data model libraries will change this. They provide a framework to expose and exploit open data models that can describe a service, or a data plane device, and its constituent components. Through YANG extensions (or augmentation) atomic functions can be defined and hence addressed by the application plane to read from, or effect change to, particular parameters of the data plane device element. This is invaluable to aid the process of orchestrating the behaviour of novel disaggregated components on the lab-bench for incorporation within generic open switches, transponders and ROADMs that are now entering the marketplace.

There are several network transport protocols (such as NETCONF, RESTCONF or gRPC) that support the carriage of serialised encoded messages whether formatted as XML, JSON or Google protocol buffers (gPB). The architectural framework sketched in Figure 2 (i) is now a reality. In the work we describe it was convenient to use NETCONF as the transport protocol and XML for message encoding as shown in Figure 2 (ii). This approach supported the use of open, SDN tools that included Netopeer, for its NETCONF toolset; and Sysrepo for the configuration and monitoring of the YANG device models that we created. These included legacy disaggregated devices (e.g. optical power monitoring) via GPIB; Transceivers via SFF specifications over the I^2C bus that we have described previously for the management of tuneable transceivers and open ROADMs [4]. Although modest in scope, this approach is compatible with existing end-to-end open, optical frameworks such as ONFs ODTN initiative [5].

3. Conclusion

A palate of possibilities is now possible by harnessing disaggregated hardware controlled and managed by open software. This approach extends the management and control of both legacy devices and novel-devices for incorporation in emerging open, disaggregated hardware platforms. This provides more flexibility and greatly aids component migration and upgrade without vendor lock-in. Most importantly it is consistent with harnessing the abilities of the cohort of CS millennials, already well-versed in programming “open” hardware Arduinos [6] and RaspberryPi’s [7], with open software; to incorporate legacy and innovative optical devices within dis-aggregated server-based hardware platforms.

This work was supported by Science Foundation Ireland under grant 13/CDA/2103, and IPIC (12/RC/2276) and CONNECT (13/RC/2077).

4. References

[1] E. Haleplidis, et al. “Software-Defined Networking (SDN): Layers and Architecture Terminology,” 2015. https://tools.ietf.org/html/rfc7426.
[2] J. Schoenwaelder, et al. “Overview of the 2002 IAB Network Management Workshop,” 2003. https://tools.ietf.org/pdf/rfc3535.pdf.
[3] L. A. Barroso and U. Hölzle, "The Case for Energy-Proportional Computing", IEEE Computer, 40, pp. 33—37, December 2007.
[4] S. Ahearne, et al. “Software Defined Control of Tunable Optical Transceivers Using NETCONF and YANG,” Eur. Conf. Networks Commun. EuCNC 2018, pp. 81–86, 2018.
[5] A. Campanella, et al., "ODTN: Open Disaggregated Transport Network. Discovery and control of a disaggregated optical network through open source software and open APIs.," OFC2019, paper M3Z.4.
[6] https://www.arduino.cc/
[7] https://www.raspberrypi.org/