An Open Controller for the Disaggregated Optical Network

Marc De Leenheer, Yuta Higuchi, Guru Parulkar

***Abstract*— The Open and Disaggregated Transport Network is an operator-led initiative to build data center interconnects using disaggregated optical equipment, common and open standards, and open source software. We discuss the project objectives, roadmap, and design.**

***Index Terms*—Open source software, Optical fiber networks, Software defined networking, Standards**

I. Introduction

SERVICE providers are constantly trying to simplify and automate network operations. The combination of hardware disaggregation and the use of advanced software tools to control, configure and observe networks is expected to be a major driver towards this goal. To this end, we founded the ODTN project (short for Open and Disaggregated Transport Networks), an industry-wide and operator-led initiative to build an open source reference platform and deliver production-ready solutions using an innovative supply chain model.

The Open Networking Foundation has launched this project to rally service providers, hardware vendors and system integrators around the following objectives:

1. Build a reference implementation using (a) open source software, (b) open and common data models, and (c) disaggregated hardware devices.
2. Perform lab and field trials using the reference implementation.
3. Identify supply chain gaps and propose solutions.

In particular the project will focus on disaggregated DWDM systems, including but not limited to transponders and Open Line Systems, amplifiers, multiplexers, all-optical switches and ROADMs.

II. Objectives and Roadmap

The project addresses increasingly complex network scenarios, starting with relatively simple point-to-point open line systems and ending with a meshed network consisting of disaggregated optical equipment.

Initial focus in Phase 1 is on what is commonly referred to in the industry as the *Open Line System* [1]. Typically in the form of a simple point-to-point topology, such deployments are increasingly popular for cloud and content providers. These companies operate at extreme limits of networking performance and scale, and require high levels of flexibility and programmability. We expect the popularity of the OLS will only increase over time. Indeed, many traditional telco companies are starting to deploy and operate their own telco cloud infrastructure. Concrete initiatives such as CORD (Central Office Re-architected as a Data center) [2], and the more general multi-access edge computing and edge cloud projects reflect the growing importance of this trend. As such, interconnecting these Central Offices is the prime use case for our software stack.

Phase 2 will introduce integrated ROADM devices, leveraging open APIs to control and configure such equipment. Such deployments are necessary to support meshed topologies. The final Phase 3 will perform disaggregation of the ROADM itself; it is as of yet to be decided what *granularity of disaggregation* will be performed. The complete disaggregation into basic optical components as shown in the figure is unrealistic for reasons of performance and complexity of integration. It is expected that some intermediate form of disaggregation will offer acceptable performance while greatly improving flexibility and programmability. This is an area of active research in both academia and industry; guidelines and best practices will be folded into the project as these become available.

Each phase produces a set of deliverables that consist of the following:

1. Reference implementation composed of complete open source software stack, set of applications, and device drivers.
2. Integration report detailing measured, expected and potential gaps for performance levels and feature sets. Of particular interest are limitations brought about by a multi-vendor environment.
3. Detailed plan for operator-led deployment (lab or field trial), including system design, gap analysis, vendor selection, and resource commitments.

Paper first submitted on February 6th, 2018.

Marc De Leenheer is with the Open Networking Foundation (e-mail: marc@ opennetworking.org).
Yuta Higuchi is with NEC Corporation America (e-mail: yuta.higuchi@necam.com).
Guru Parulkar is with the Stanford University and the Open Networking Foundation (e-mail: parulkar@stanford.edu).

978-3-903176-07-2 © IFIP 2018

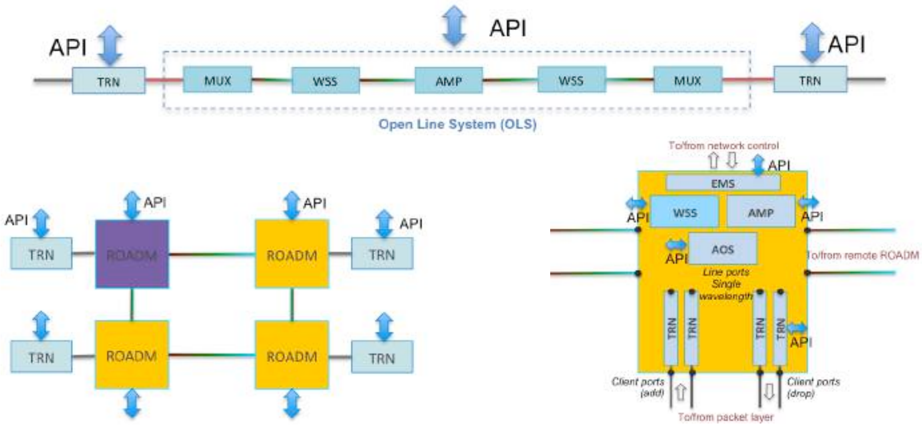

Fig. 1. Three phases to go from simple point-to-point deployments to fully disaggregated, meshed optical networks. Phase 1 addresses the OLS, initially controlling the terminal equipment (Phase 1.0), afterwards adding explicit control of the line system itself. Phase 2 introduces control of integrated ROADM nodes. Phase 3 disaggregates the ROADM further into more basic components.

III. Open Source Software Stack

The high-level design of our reference implementation is shown in Figure 2. It essentially breaks down into three parts: the northbound or service layer API, the network OS, and the southbound or device layer API.

On the northbound side, Transport API is an open standard for configuration and control of transport networks, offering a variety of services such as topology, connectivity, path computation and others [3]. The standard has achieved significant industry traction – in the form of field trials and interoperability testing – with numerous service providers and the support of a variety of equipment vendors. Many options are available in terms of northbound protocol specification; our initial implementation supports RESTCONF-based interaction between service orchestrator and the network operating system.

In turn, the southbound protocol is foremost based on NETCONF due to the limited availability of mature implementations on hardware devices. Of greater interest is the choice of standard device models. The importance of using open and common standards for optical equipment cannot be overstated, as it is the chief way to finally diminish vendor lock-in. OpenConfig is a set of vendor-neutral network management models, driven by actual operational use cases from network operators [3]. Another option is the OpenROADM multi-source agreement, which focuses on disaggregated models for optical ROADMs, with a strong emphasis on interoperability in multi-vendor environments. For Phase 1 at least, ODTN is developing a solution based on the OpenConfig models, while Phases 2 and beyond will evaluate the applicability of OpenROADM [4].

The translation between north- and southbound models occurs inside the network OS, in this particular case the Open Networking Operating System or ONOS. ONOS is an open-source SDN operating system architected to meet the stringent availability, scalability and performance demands of service provider networks. It was built from the ground up with a distributed architecture and introduces innovative state management techniques necessary to build a robust distributed SDN controller.

ONOS provides APIs that enable applications to view the network topology and to inspect and control the devices that compose it. However, many application developers would rather operate at a network-level, rather than a device-level, and do not need specific control of all parameters. To this end, ONOS championed the *intent abstraction*, which frees application developers from the need to specify and control low-level parameters, leaving them for the network operating system to optimize as network conditions change. Additionally, ONOS offers a converged topology abstraction, in which multiple network layers and technologies are presented as a single logical graph. Next to these network-centric abstractions, ONOS also offers a sub-system that can span both network and device-level state management. Indeed, the Dynamic Configuration Sub-system or DCS allows application developers to interact with a configuration database, offering advanced features such as distributed state management, transactions, and automated protocol handlers. Finally, although ONOS comes with a built-in multi-layer PCE, the modular system design allows external PCEs to be plugged in and out on-demand.

A concluding remark is that the complete reference implementation, including network OS, applications and drivers, is being made available under an Apache 2 software license. This is an open source software license that is relatively liberal in how. This deliberate choice increases the chance for further adoption by the industry, and ultimately improves the community to take ownership and maintenance of the code base.

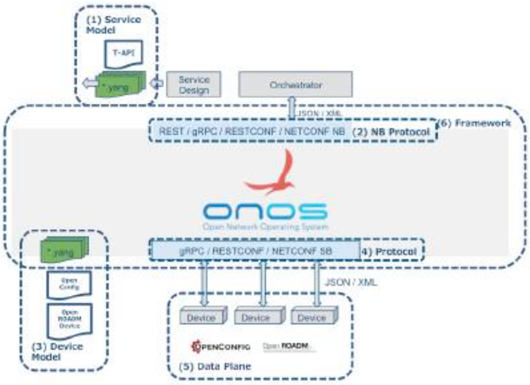

Fig. 2. Software stack based on the Open Networking Operating System (ONOS), detailing both north- and southbound models.

IV. Service Application

As a network OS, ONOS' first mission is to manage network hardware and offer common services for network applications. In ODTN, the core functionality to translate between north- and southbound operations is handled by the service application depicted in Figure 3. It shows how different ONOS sub-systems work together to take (1) a network-level service request, (2) semantically interpret the request, (3) compute a set of abstracted device configurations, and finally (4) generate concrete device control and config instructions. The intermediate abstract device configurations are necessary to allow flexibility in terms of device models, both in terms of differing/competing models or revisions of existing ones.

V. Conclusions

The ODTN project is an industry-wide effort to bring openness and innovation to the optical networking space. It does this by putting service providers in the driver's seat, allowing them to prioritize use cases and hopefully maximize the opportunity for production deployments. The project is built on three core principles: (1) disaggregated hardware, (2) open and common data models, and (3) open source software. All of these are essential to rally the industry eco-system, rebuild the supply chain, and bring about much-needed disruptive changes. It is our sincere hope that the community will step up and help us realize this vision.

References

[1] Mark Filer, Hacene Chaouch, and Xiaoxia Wu, "Toward Transport Ecosystem Interoperability Enabled by Vendor-Diverse Coherent Optical Sources Over an Open Line System," Journal of Optical Communications and Networking, 10(2):A216–A224, February 2018.

[2] Larry Peterson, Ali Al-Shabibi, Tom Anshutz, Scott Baker, Andy Bavier, Saurav Das, Jonathan Hart, Guru Parulkar, William Snow, "Central office re-architected as a data center," IEEE Communications Magazine, 54(10):96–101, October 2016.

[3] Open Transport Configuration and Control, https://www.opennetworking.org/projects/open-transport/

[4] OpenConfig, http://www.openconfig.net/

[5] OpenROADM MSA, http://openroadm.org/home.html

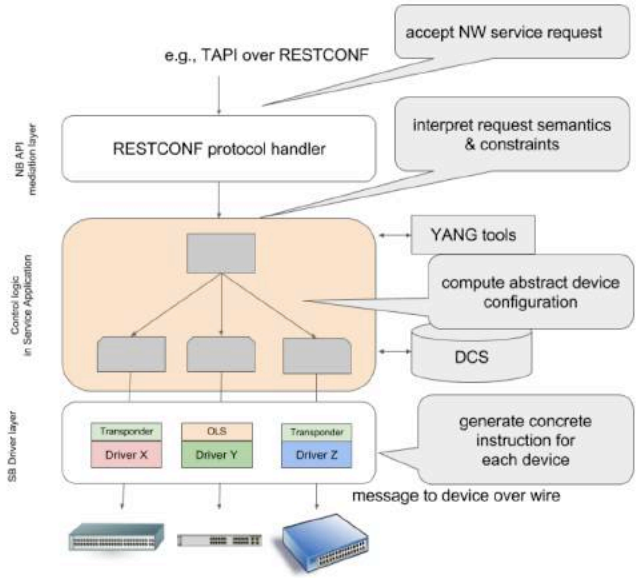

Fig. 3. Service application that deals with translating service-level requests into device operations. The application relies extensively on key ONOS sub-systems, such as the Dynamic Config Sub-system (DCS), ONOS YANG tools, Intent Framework and the Driver & Behaviour mechanisms.

Marc De Leenheer is a Member of Technical Staff at the Open Networking Foundation, an operator-led consortium that is transforming networks into agile platforms for service delivery. Marc leads the teams for packet/optical convergence and Enterprise CORD, and his software platforms are being trialed by some of the largest service providers on the planet. He received the PhD degree in computer science from Ghent University, Belgium in 2008, and completed post-doctoral stays at UCDavis and Stanford University.

Yuta Higuchi received the Master's degree in Information Science from Nagoya University. He joined NEC Corporation in 2007, where he has been undertaking development of a platform framework for operation management software stacks. He is currently with the Optical IP Development Division, NEC Corporation of America. His research interests include architecture design of scalable frameworks for Software-Defined Networking.

Guru Parulkar is Executive Director of Open Networking Foundation (ONF), Stanford Platform Lab, and Consulting Professor of EE at Stanford University. At ONF he leads open source projects ONOS and CORD.

Guru has been in the field of networking for over 25 years. He joined Stanford in 2007 as Executive Director of its Clean Slate Internet Design Program. At Stanford Guru helped create three programs: OpenFlow / Software-Defined Networking, Programmable Open Mobile Internet 2020, and Stanford Experimental Data Center Laboratory.

Prior to Stanford, Guru spent four years at the National Science Foundation (NSF) and worked with the broader research community to create programs such as GENI, Future Internet Design, and Network of Sensor Systems. Guru received NSF Director's award for Program Management excellence.

Before NSF Guru founded several startups including Growth Networks (acquired by Cisco) and Sceos (IPO'd as Ruckus Wireless). Guru served as Entrepreneur in Residence at NEA in 2001 and received NEA's Entrepreneurship Award.

Prior to this Guru spent over 12 years at Washington University in St. Louis where he was a Professor of Computer Science, Director of Applied Research Laboratory and the head of research and prototyping of high performance networking and multimedia systems.

Guru received his PhD in Computer Science from the University of Delaware in 1987. Guru is a recipient of the Alumni Outstanding Achievement award and the Frank A. Pehrson Graduate Student Achievement award.