

面向 IP 和 OTN 协同的 SDN 管控技术

刘亚峰

(中国电信江苏分公司 省操作维护中心, 南京 210017)

摘要: 针对近些年网络新业务和新技术应用不断增长的趋势, 介绍了光网领域应用软件定义网络(SDN)技术的重要性。重点研究分析了 2 种面向 IP 和光传送网(OTN)协同的 SDN 技术——ONF Transport API(TAPI)和 OpenDayLight Transport PCE(TPCE), 包括它们具体的技术细节以及如何实现跨域跨厂家多层网络管控, 并对它们的技术特点进行了比较。最后展望了这 2 种 SDN 技术未来的发展方向。

关键词: 光传送网; 软件定义网络技术; YANG 数据模型

中图分类号: TN914 文献标志码: A 文章编号: 1002-5561(2021)01-0042-06

DOI: 10.13921/j.cnki.issn1002-5561.2021.01.009

开放科学(资源服务)标识码(OSID):



SDN management and control technology for IP and OTN collaboration

LIU Yafeng

(China Telecom Jiangsu Branch provincial operation and maintenance center, Nanjing 210017, China)

Abstract: In view of the increasing trend of new network services and new technology applications in recent years, the importance of software defined network (SDN) technology in optical network field is introduced. This paper focuses on the research and analysis of two SDN technologies for IP and optical transport network(OTN) collaboration: ONF transport API(TAPI) and OpenDayLight transport PCE (TPCE), including their specific technical details and how to achieve cross domain and multi-layer network management and control, and compare their technical characteristics. Finally, the future development direction of these two SDN technologies is prospected.

Key words: optical transport network; software defined network technology; YANG data model

0 引言

传统通信是一个按需建设、静止且封闭的网络, 由于承载业务的多样化, 网络也越来越复杂。随着云计算、物联网等新兴业务的出现和发展, 传统网络模式已经难以为继, 并严重阻碍了网络技术的发展和革新。软件定义网络(SDN)的出现, 打破了传统网络以硬件为主的发展模式, 软件开始成为了网络发展的主要驱动力。SDN 的概念开始在运营商、设备商和互联网厂商之间广为传播, 并掀起了一股研究和应用热潮。

收稿日期: 2020-07-13。

作者简介: 刘亚峰(1974—), 男, 江苏宜兴人, 硕士, 通信工程师, 毕业于南京邮电大学, 现工作于中国电信江苏分公司省操作维护中心, 主要负责江苏电信一二级传输干线维护和技术支撑, 以及传输网络规划和优化工作。参与了省内多个一二级干线传输系统工程的可研、设计和建设。



SDN 本质上是一种网络编程方法, 通过提供开放的标准化接口给上层应用(软件), 实现由软件来动态地驱动、控制、调整和管理网络。SDN 最初只是针对数据网络设计的, 利用控制平面和转发平面的分离以及控制功能的逻辑化集中实现网络流量的灵活调度。随着 SDN 研究的不断深入, 该项技术已经延伸到了光网领域, 并已经成为了当前 SDN 技术应用研究的主要方向。本文深入研究当前光传送网(OTN)中 2 种最主要的 SDN 技术——ONF Transport API(TAPI)和 OpenDayLight Transport PCE(TPCE)。

1 SDN 技术在 OTN 中的应用需求

本文从运维优化、网络演进等方面分析了 SDN 技术在 OTN 中的应用需求:

①摆脱传统模式束缚, 加速 OTN 创新和网络演进的要求。传统的光传输网络由设备硬件、操作系统和

应用功能三部分紧密耦合在一起,组成一个封闭系统,这三部分相互依赖,每一部分的创新和演进都要求其余部分做出相应的升级。与可以跨厂家混合组网的路由器、交换机这样的数据设备不同,不同厂家的光传输设备之间互通性和互操作性非常糟糕,既无法混合组网,也不能统一管理。这些都限制了网络创新步伐,减慢了网络演进的速度。

②优化多层网络资源、提高维护效率的要求。OTN 主要用于承载高速率大带宽 IP 业务,即 IP 网络通常叠加在传输网络之上,而这 2 个网络的控制平面并不互通,无法实现 IP 网络和传输网络的统一调度和多层优化。此外,IP 网络和传输网络在传统上是 2 个完全独立的网络,无论是网络规划、设计和维护都是由不同的专业技术人员完成,这势必造成过度配置(数据和传输在各自的网络上都对同一条业务提供冗余和保护,从而造成网络资源浪费)和维护效率低下的问题。

③跨网业务动态调整和路径优化的要求。运营商网络中往往存在多个由不同厂家设备组建的 OTN,由于没有统一控制手段,无法实现跨网业务的端到端监测和全局性的动态调整。而且,跨网业务往往只能保证在单个 OTN 域内业务路径的最优化,而无法确保端到端全程业务路径都能最优化。

④未来电信业务的承载要求。根据 5G 乃至 6G 的应用场景,可以看到未来的电信业务承载网必须满足更大的带宽、更低的时延、更高精度的时间同步、更灵活的组网和功能更强大的网络切片的要求,而这些都需要通过在 OTN 中引入 SDN 技术来实现。

2 SDN 技术在 OTN 中的应用前提

2.1 解决网络的可编程性

SDN 是通过应用软件程序动态地对网络设备进行编程,从而实现控制整个网络行为的能力。网络可编程是 SDN 技术的核心,解决网络的可编程性,需要有更加灵活的电层和光层。

2.1.1 电层

灵活的电层用以适配各种速率、各种类型业务的动态接入、调度和调整问题:

- 除了支持固定速率的 ODU_k ($k=1,2,3,4\cdots$)之外,还支持 ODU₀ (1.25 Gb/s)、灵活速率 ODUflex (CBR/GFP)以及 ODUflex (GFP)的无损调整 (ITU-T G.7044),为多种不同类型和不同速率的客户信号的接入提供了映射、复用和动态调整手段。

- 通过引入基于分组 (OIF 的 OFP 协议)的统一

交叉矩阵和 OTN 混合线卡概念,实现了分组 (Packet)、SDH (VC)和 ODU_k 的统一交叉,从而可以对不同类型的业务颗粒进行统一灵活的调度。

2.1.2 光层

灵活的光层用以解决 OTN 光层可编程问题:

- 可调谐波长转换元器件的使用,实现了 Xponder 波长软件可调。

- 无色、无方向和无竞争功能的可重构光分插复用器 (CDC ROADM)的使用,实现了 OTN 光层的灵活组网和光层业务的灵活调度,尤其是最新一代的 ROADM 中可编程硅基液晶 (LCOS)技术的应用,实现了波长调度软件可调。

- 灵活栅格 (flex-grid)、超级信道 (super-channel)以及不同的相干波长调整技术的应用,可以为 100G 以上 (B100G)系统不同的业务数据速率制定灵活的调制方案,实现光通道带宽软件可调。

2.2 解决网络的互操作性

解决网络的互操作性需要有统一的模型和规范。网络的互操作性是跨厂家设备组网的关键,有效避免了厂家锁定,提升组网灵活性和减少网络的投资成本。

当前 OTN 使用的标准和规范都是 ITU-T 制定的。由于 OTN 电层承袭了 SDH 技术特点,因此不同厂家的 OTN 设备可以在电层上互连、互通。但对于 OTN 光层而言,不同厂家的设备所支持波长数量、单波带宽和收发光功率范围等都有所不同,最主要的是光线路使用的前向纠错 (FEC)标准不一致,OTN 设备在光层并不能跨厂家混合组网,也无法实现互操作。要实现 OTN 在光层的互操作性,则需要有统一的模型和互操作规范。目前,专注于 OTN 中 SDN 技术应用的标准组织及其工作内容如下:

- IETF。最主要的网络标准化组织,致力于提供 OTN 标准化参考协议 (RFC)、网络信息模型 (UML)和数据模型 (YANG/SMI)等。

- OpenConfig^[1]。为光传输系统定义了通用的、但不完整的 YANG 数据模型。

- ONF。推出了支持 OTN 的 OF-Switch 光传输协议扩展 (TS-022)^[2]和实现 OTN 抽象的 TAPI (当前版本 2.1)接口^[3]。

- OpenROADM MSA^[4]。以实现跨厂家 ROADM 设备互操作性为目的,定义了 OpenROADM 统一的通用的网络、设备和业务 YANG 数据模型。

- TIP^[5]。Facebook 推出了支持白盒化 Xponder 硬件的开源软件平台 Voyager。

3 OTN 网络中的 SDN 技术

ONF 的开放式网络操作系统(ONOS)^[6]和 LFN 的 ODL(OpenDayLight)^[7]是当前最普及也是最主要的 2 种开源 SDN 控制器技术,都受到了运营商和设备商的广泛支持 and 应用。ONF 和 LFN 针对 OTN 的 SDN 化分别推出了 ONF TAPI 套件^[3]和 OpenDayLight TPCE 技术^[8]。

3.1 ONOS 和 TAPI

ONOS 是 ONF 推出的开源 SDN 控制器项目^[6],其旨在白盒系统上创建灵活的高性能可编程运营商网络。ONOS 采用 API 驱动的服务抽象层(AD-SAL)方式,使用 API 接口或插件来南向控制设备和北向提供服务。通过 ONF 的 OpenFlow 协议以及不同厂家提供的专有南向协议(TL1/SNMP/PCP),ONOS 可以控制由不同厂家设备组成的使用不同网络协议的多层硬件网络,ONOS 核心模块(bundles)和设备厂家无关,所有的网络协议以及和厂家专有的信息都隐藏在南向接口驱动程序(provider)中。

TAPI 是 ONF OTCC 项目组在 ONF SDN 架构和 ONF 核心信息模型(CIM)基础上针对 OTN 开发的一套北向接口 API 软件研发工具包(SDK),其中包含了 UML 信息模型、YANG 数据模型和用于 RESTCONF 协议下 CRUD(Create、Read、Update、Delete)操作定义的 OpenAPI 规范。TAPI 将 OTN 功能抽象为一组业务 API 接口(TR-527)^[9],提供给上层应用程序调用以及分层 SDN 架构之间的递归调用,包括:

- 拓扑业务 API。提供网络拓扑检索业务。
- 连接业务 API。提供连接信息检索以及连接请求业务。
- 路径计算业务 API。提供业务路径计算请求和业务路径优化请求业务。
- 虚拟网业务 API。提供虚拟网信息检索和虚拟网请求业务。
- 通知业务 API。为 API 客户端提供告警、性能越限(TCA)、对象创建/删除、属性值更改(AVC)、状态更改等事件报告的订阅和过滤业务。
- OAM 业务 API。通过 OTN 光层和电层开销提供告警和性能监测、故障发现和网络维护手段。
- 设备清单业务 API。提供网络设备信息检索业务。

ONF TAPI 功能架构如图 1 所示。

通过 TAPI 可以在上层控制器上,或在应用层上实现跨域跨厂家设备的互操作,如图 2 所示。

TAPI YANG 数据模型是 RESTCONF 协议下 CRUD

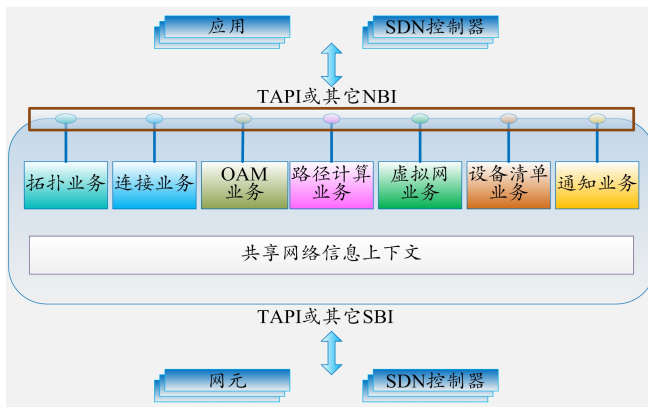


图 1 ONF TAPI 功能架构

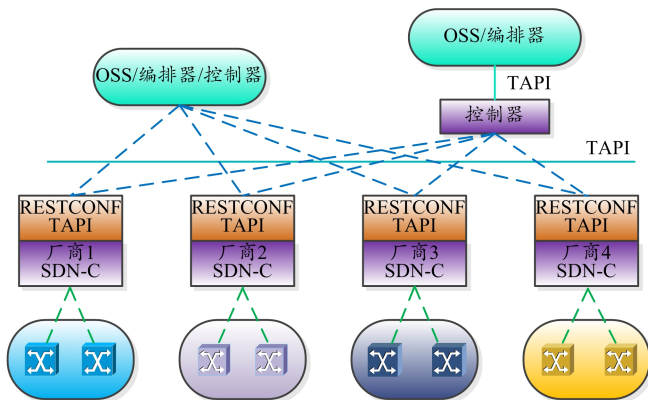


图 2 通过 TAPI 实现跨域跨厂家设备的互操作

操作的核心,它是由 TAPI UML 信息模型通过 ONF Eagle 项目的开源工具 xmi2yang 自动生成的。

ONOS 作为 ONF 大力主推的开源 SDN 控制器平台,其内置了对 TAPI YANG 数据模型的支持,既可以作为厂家专有 SDN 控制器的替代,也可以作为厂家专有 SDN 控制器的上层控制器。

将 ONOS 作为 OTN 的 SDN 控制器使用时,可以使用 NETCONF 作为南向协议,利用 ONOS 内置的 TAPI YANG 数据模型来抽象 OTN,以实现跨域跨厂家 OTN 的统一管控和互操作。在多层网络(IP/OTN)环境下,也可以使用 ONOS 作为集中式多层网络控制器,其中 IP 网络统一使用 OpenFlow 南向协议,从而实现跨域跨厂家多层网络(IP/OTN)的集中管控,如图 3 所示。

在将 ONOS 作为统一的 SDN 控制器时,理论上只要 OTN 设备支持 OpenFlow 1.4(及以上版本)协议和“OF-Switch 的光传输协议扩展”(TS-022),IP 网络和 OTN 则都可以使用 OpenFlow 作为南向接口协议来实现上面所述的跨域跨厂家多层网络的统一管控和互

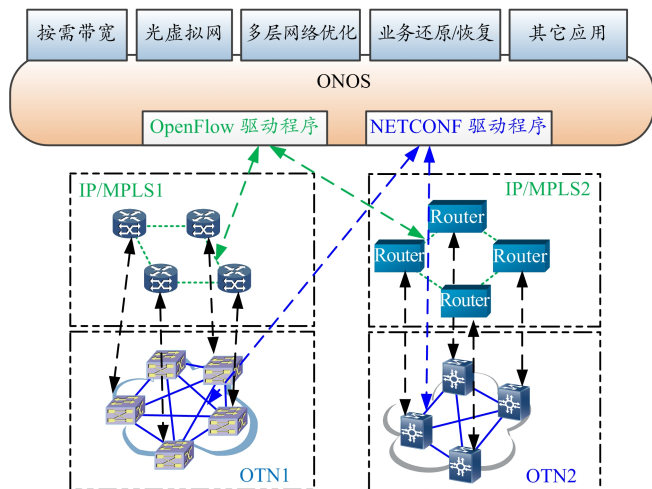


图3 使用 ONOS 作为 SDN 控制器实现多层网络的集中管控

操作。但是,经过多年实践,OTN 设备采用 OpenFlow 流表模式建模难度极大,除了在模拟器中(LINC-OE)有所实现之外,尚无设备厂家有成熟的产品推出,因此,在 OTN 这种复杂的网络中南向接口使用 NETCONF 协议已经成为业界的共识。

3.2 ODL 和 TPCE

ODL 是 LFN(Linux Foundation Networking)的开源 SDN 控制器项目^[7]。与 ONOS 项目主要由运营商推动不同,ODL 项目主要由设备厂家推动,旨在构建一个高可靠性、模块化和灵活可扩展的多协议 SDN 控制器基础结构,可以适配多厂家设备的 SDN 部署,并允许用户可以在该平台上轻松编写跨厂家设备和南向协议的网络应用程序。ODL 是基于模型驱动的服务抽象层(MD-SAL)框架的,因此 YANG 数据模型在 ODL 中至关重要,用于创建数据存储结构、生成各种接口函数和数据结构对象的 Java 代码,以及生成供上层应用调用的 REST API(RESTCONF)函数。此外,关于网络、设备和服务的各种消息、远程过程调用(RPC)操作也都是在 YANG 数据模型中定义的。各个厂家的设备通过受 ODL 支持的统一的 YANG 数据模型(例如 OpenFlow 和 OpenConfig)建模,可以实现跨厂家设备的组网和控制,如图 4 所示。此外,ODL 通过加载各种模块化插件(bundles)可以灵活地添加对当前现有的各种南向接口协议(OpenFlow/NETCONF/SNMP/PCEP)的支持。

TPCE 是 LFN 在 ODL 框架内基于 OpenROADM MSA 规范开发的一套管理 OTN、设备和业务的应用程序^[8],其核心内容是为 OTN 提供一个端到端业务路径计算引擎,从而实现为上层应用提供各种 OTN 服务,

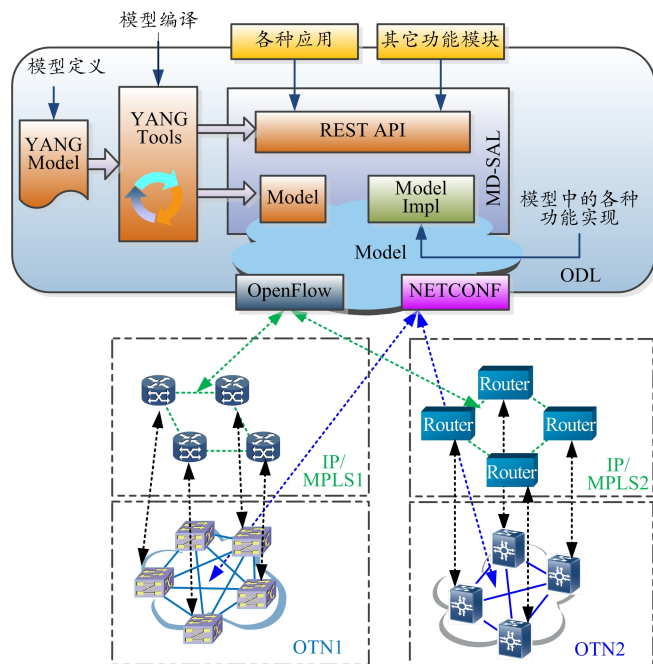


图4 使用 OpenDayLight 作为 SDN 控制器实现多层网络的集中管控

包括保护服务(端到端业务保护、故障后恢复以及快速重路由)和光虚拟网(OVPN)服务,与使用 PCE 作为集中控制器的体系架构(IETF RFC8283)有点类似。TPCE 体系架构如图 5 所示。图中虚线表示 YANG 数据模型映射,实线表示直接的 API 调用和进程间通信。

TPCE 中的拓扑管理模块通过访问 OpenROADM 注册网元,使用 LLDP 协议来自动发现节点、链路,并根据节点和拓扑构建 OpenROADM 网络拓扑。

业务处理程序 SH(Service Handler)模块处理来自北向 RESTCONF API 的业务处理请求,包括业务的创建、删除、恢复、重配置和重路由等,将业务实现请求发送给 Renderer,并从 Renderer 接收业务实现响应。

PCE 从 SH 接收路径计算或业务可行性检查请求,负责根据约束条件计算 A 端到 Z 端的最优业务路径,并将计算结果返回给 SH。

Renderer 从 SH 接收业务实现请求,将路径描述的抽象元素(TP/Node)转换为设备元素(电路盘、端口等),并通过 NETCONF 接口来配置业务路径上的所有设备,即配置接口和创建、删除交叉连接。

光纤线路终端(OLM)模块用于对光线路的光功率进行设置和调整,以确保 OTN 光层线路的光信噪比满足要求。

TPCE 之所以选择基于 OpenROADM MSA 规范开发^[4],是因为 OpenROADM MSA 为 OTN 提供了一组通

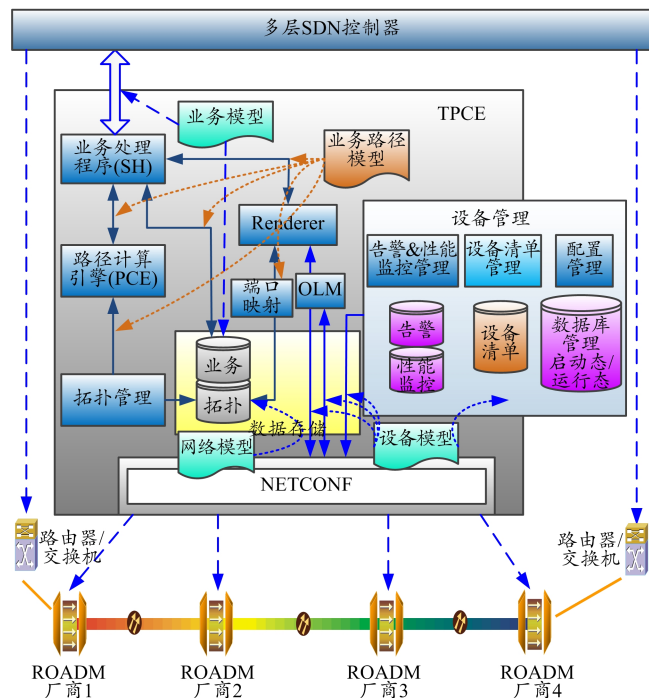


图5 TPCE 体系架构

用的、完备的 YANG 数据模型,包括了网络模型、业务模型和设备模型。其中,OpenROADM 网络模型^[10]是在 IETF RFC8345 的 IETF-network 模型基础之上扩充了对 OTN 光层和 OTN 层网元和链路的支持,为 SDN 控制器上的路径计算引擎 PCE 提供网络拓扑信息;业务模型^[11]由业务相关的数据存储、RPC 和通知组成,用于新建、删除、变更业务以及获取与这些操作相关的业务信息和通知;设备模型^[12]中定义了 CDC ROADM、直插式光放(ILA)、Xponder 和可插拔光元器件的设备和功能设计规范,用于跨厂家设备的统一管理、控制和配置,并且这些功能模块之间都使用了统一的开放接口,以实现跨厂家设备之间的互连和互操作。

TPCE 中的业务处理程序 SH 模块和上层应用对接的北向 API 使用的是 MSA 的业务模型;Render器 和 OLM 模块通过南向接口(NETCONF/YANG)管理配置 OTN 设备使用的是 MSA 的设备模型;拓扑管理模块使用的则是 MSA 的网络模型。

由于 TPCE 是基于 OpenROADM MSA 规范之上的,通过 OpenROADM MSA 统一的 YANG 数据模型和互操作性规范可以实现 OTN 设备的跨厂家互连和互操作,而不是仅仅只能在控制层面之上实现统一管控和互操作。

对于 OpenDayLight 这种 MD-SAL 架构来说,OpenROADM MSA 的网络模型、设备模型和业务模型

才是 TPCE 真正的核心和关键。

3.3 2 种技术的比较

综上所述,本文将 TAPI 和 TPCE 这 2 种 SDN 技术主要的异同点比较如下:

3.3.1 相同点

- SDN 控制器使用的框架相同。ONOS 和 ODL 都是基于 OSGi 框架(一种 Java 动态化模块化框架)的,并且都是用 Apache Karaf(一种轻量级 OSGi 容器)来动态部署各种功能组件和应用程序。

- 适用场景(用例)相同。TAPI 和 TPCE 都是为了实现 OTN 这样大型的、复杂的多层网络的可编程性,以实现跨厂家、跨层、跨域实现全网资源和业务的优化,并能在 OTN 网络中快速引入新服务,加速 OTN 网络创新步伐。

- 最终的目标一致。TAPI 和 TPCE 的最终目标都是实现 OTN 网络及设备的硬件开放式(遵循统一的技术规范和数据模型设计和制造)、软件开源化(SDN 控制器和南北向接口 API 都基于开源项目),从而实现跨厂家设备的混合组网和统一管理,以增加网络的灵活性,从而最终降低网络的投资成本(CapEx)和运营成本(OpEx)。

- 都可以跨控制器平台移植。由于 ONOS 和 ODL 使用了相同的容器架构(OSGi),因此 TAPI 和 TPCE 可以在这 2 个控制器平台上平滑移植。

3.3.2 不同点

- 服务抽象层使用的驱动方式不同。ONOS 沿用了 AD-SAL(API 驱动服务抽象层)方式,目前 ODL 使用的是 MD-SAL(模型驱动服务抽象层)方式。

- 上游依赖(upstream dependencies)性不同。TAPI 主要依赖于 OpenFlow 项目,OpenFlow 项目自身独立性较强,版本更迭影响很小;TPCE 严重依赖 NETCONF 项目,而 NETCONF 项目本身又依赖很多 IETF 内核项目,版本改动和版本更迭的影响较大。

- 推荐使用的南向接口协议不同。TAPI 推荐使用 OpenFlow 协议控制网络、设备和业务(对于 OTN 设备而言,采用 OF-Switch 建模难度很大),使用基于 NETCONF 的 OF-Config 协议管理和配置设备;TPCE 推荐使用 NETCONF/YANG 控制、管理和配置网络、设备和业务。

- 跨厂家设备互连和互操作性不同。TAPI 在控制层面上实现了对支持 TAPI 和 OpenFlow 设备的互连和互操作性,但是对底层 OTN 设备的跨厂家互连以及互操作性方面未作要求;TPCE 则要求底层的 OTN 设

备必须支持 OpenROADM MSA 模型和规范,从而保证了传送层上物理设备的跨厂家互连和互操作性。

- 为上层应用提供的服务不同。TAPI 为上层应用提供了一整套完备的 OTN 业务 API 以供调用。TPCE 主要解决的是 OTN 业务路径计算和业务管理功能,诸如 OVPN 服务等应用还需要在此基础上作二次开发。

4 结束语

本文对 TAPI 和 TPCE 这 2 种 OTN 中的 SDN 技术进行了研究,并在此基础上分析了它们的技术要点。通过比较可知,ONF TAPI 是一套完整的 OTN 中 SDN 解决方案,其技术核心是 OpenFlow 协议。2016 年,ONF 和 OIF 曾一起联合组织了全球多个运营商、设备厂商以及科研机构进行了基于 TAPI 1.0 的 OTN 互联互通测试。2018 年,ONF 和 OIF 又再次组织了基于 TAPI 2.0 的互联互通测试,也验证了 ONF TAPI 的可行性。与 ONF TAPI 不同,LFN TPCE 是 OpenDay-Light 控制器上针对 OTN 的一个平台服务,它的技术核心是 OpenROADM MSA 模型和规范,上层服务和应用使用 REST API 基于 OpenROADM 的 YANG 数据模型来管理 OTN、设备和业务。与 TAPI 技术相比,TPCE 技术较新,目前还处于概念验证(PoC)阶段。

这 2 种 OTN SDN 技术其实也分别代表了 ONOS 和 ODL 控制器技术发展方向。ONOS 虽然可以通过加载不同的南向驱动模块来支持 NETCONF、PCEP 等非 OpenFlow 南向协议,甚至可以是设备厂家专有的南向接口 API,但 ONOS 控制器的核心还是 OpenFlow,只有通过 OpenFlow 才能掩盖各种网络技术的细节和不同厂家设备之间的差异,在 OpenFlow 基础之上简化多域多层网络的统一控制,并提供必要的敏捷性和灵活性。ODL 由于主要是由设备厂家推动的,因此更注重技术包容,通过这种包容,ODL 几乎可以支持目前出现的所有南向接口协议,ODL 是 MD-SAL 架构,它不仅内置了对 IETF 和 OpenConfig 通用 YANG 数据模型的支持,而且还允许设备厂家自定义专有的 YANG 模型,这些都意味着 ODL 是一个远比 ONOS 复杂的控制器,控制器和设备之间的关联也更紧密。

对这 2 种 SDN 技术的选择其实也就是对 ONOS

和 ODL 2 种控制器技术的选择。对于运营商而言,ONOS 可能是首选,因为它可以让维护人员从设备维护中脱离出来,更多地关注于网络和业务;而对于设备商而言,应该会首推 ODL,因为它更多地保留了自己的设备“特点”,也提供了更大的“空间”让设备功能得到充分体现。这 2 种网络 SDN 技术都将会彼此交织、长期并存。

参考文献:

- [1] OpenConfig. Published versions of OpenConfig modules [EB/OL]. (2020-05-09) [2020-07-10]. <https://github.com/openconfig/public/tree/master/release>.
- [2] ONF OTCC. Optical Transport protocol extensions Ver.1.0 (TS-022) [EB/OL]. (2015-03-15) [2020-07-10]. https://opennetworking.org/wp-content/uploads/2014/10/Optical_Transport_Protocol_Extensions_V1.0.pdf.
- [3] ONF OTCC. TAPI v2.1.3 Reference implementation agreement (TR-547) [EB/OL]. (2020-06-23) [2020-07-10]. <https://www.opennetworking.org/wp-content/uploads/2020/08/TR-547-TAPI-v2.1.3-Reference-Implementation-Agreement-1.pdf>.
- [4] OpenROADM. Open ROADM whitepaper v2_2 [EB/OL]. (2018-07-20) [2020-07-10]. https://0201.nccdn.net/4_2/000/000/05c/0e7/Open-ROADM-whitepaper-v2_2.pdf.
- [5] Telecom Infra project (TIP). OPEN OPTICAL & PACKET TRANSPORT [EB/OL]. (2016-10-19) [2020-07-10]. https://cdn.brandfolderio/D8DI15S7/as/q60sew-6zd7ls-evhflw/TIP_Project_Group_Charter-OOPT.pdf.
- [6] AYAKA K, CARMELO C, Carmelo C. ONOS tutorials [EB/OL]. (2019-05-07) [2020-07-10]. <https://wiki.onosproject.org/display/ONOS/Tutorials>.
- [7] CASEY C, DHIRAJ S. OpenDayLight controller [EB/OL]. (2019-11-09) [2020-07-10]. <https://wiki.opendaylight.org/display/ODL/Controller>.
- [8] GUILLAUME L. TransportPCE [EB/OL]. (2019-11-09) [2020-07-10]. <https://wiki.opendaylight.org/display/ODL/TransportPCE>.
- [9] ONF OTCC. Functional requirements for transport API (TR-527) [EB/OL]. (2016-06-10) [2020-07-10]. https://www.opennetworking.org/wp-content/uploads/2014/10/TR-527_TAPI_Functional_Requirements.pdf.
- [10] OpenROADM. Open ROADM Network Model [EB/OL]. (2018-10-23) [2020-07-10]. https://0201.nccdn.net/4_2/000/000/06b/a1b/Open-ROADM-Network-Model-Whitepaper-v2-2-1.pdf.
- [11] Open ROADM. Open ROADM Service Model [EB/OL]. (2018-10-01) [2020-07-10]. https://0201.nccdn.net/1_2/000/000/0f6/092/Service-Model-White-Paper-v2_2-0831.pdf.
- [12] OpenROADM. OpenROADM device white paper [EB/OL]. (2018-08-28) [2020-07-10]. https://0201.nccdn.net/1_2/000/000/134/c50/Open-ROADM-MSA-release-2-Device-White-paper-v1-1.pdf.