

SDN 北向接口发展现状与趋势研究

Research on the Development Status and Trends of SDN Northbound Interface

庞涛¹, 魏含宇², 武娟¹, 陈建²/Pang Tao¹, Wei Hanyu², Wu Juan¹, Chen Jian²
(1. 中国电信股份有限公司广州研究院 广州 510630; 2. 华为技术有限公司 深圳 518129)
(1. Guangzhou Research Institute of China Telecom Co., Ltd., Guangzhou 510630, China;
2. Huawei Technologies Co., Ltd., Shenzhen 518129, China)

摘要 随着 SDN 技术的快速发展以及开源 SDN 控制器的日益成熟, SDN 相关技术的标准化进展迅速。各标准组织为 SDN 定义了南北向接口, 在 SDN 接口标准化方面, 控制器与底层网络的南向接口以 OpenFlow 为代表, 通过控制器向上层业务应用开放的北向接口, 其标准化工作正在逐步展开。SDN 北向接口的发展目前主要包括几条主线, 一条是以开源系统为代表的实践中的北向接口发展路线, 一条是由各个标准组织推动的北向接口标准化发展路线, 同时, 学术界对 SDN 北向接口的开放也有自身的考虑, 本文综述了这几个方向的发展现状及趋势。

关键词 SDN; 北向接口; 控制器; API

Abstract With the rapid development of SDN technology and the increasing maturity of SDN open source controllers, the standardization of SDN is making process. Various standards organizations defined southbound and northbound interfaces of SDN, in the SDN standardization of interfaces, there is no effective standard on northbound as OpenFlow on southbound interface. There are many achievements on the research of SDN northbound interface. The development of northbound interface includes some clues, one is open source system, the other is standardization, and academia to open SDN northbound interfaces has its own considerations, the development status and trends of these directions were reviewed.

Key words software defined network, northbound interface, controller, application programming interface

1 引言

网络开放是产业发展的必然趋势, 不但能够为设备和网络带来变革, 还可以进一步细分产业链, 带来新的产业发展机遇。在网络通信领域, 从网络开放性的角度来理解 SDN, 可以将 SDN 抽象出图 1 所示的北向接口、南向接口和东西向接口 3 种接口类别, 其中, SDN 通过南向接口实现数据网络中控制平面与数据平面的分离, 以 ONF 的 OpenFlow 协议和 IETF 的 ForCES、OVSDB、NETCONF+YANG、PCEP 等协议为典型代表; 东西向接口负责解决多个设备控制平面之间协同工作的问题; 北向接口则是本文的研究重点。

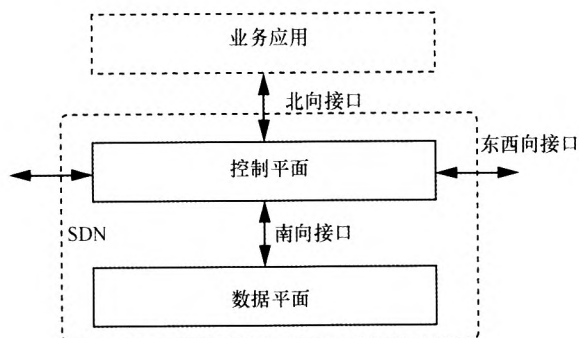


图 1 SDN 接口示意

从应用的角度自北向南看, SDN 北向接口是用户业务以及各种网络业务开发者有效控制和利用网络

的门户,开发者能以软件编程的形式调用各种网络资源;从网络运营的角度自南向北看,SDN 北向接口是通过控制器向上层业务应用开放的接口,SDN 向上提供资源抽象,实现软件可编程控制的网络架构,上层的网络资源管理系统或者网络应用可以通过控制器的北向接口,全局把控整个网络的资源状态,并对资源进行统一调度。

北向接口方案及协议的制定成为当前 SDN 领域竞争的焦点,目前尚未形成统一的标准。不同的参与者或者从用户角度出发,或者从运营角度出发,提出了很多方案,甚至部分传统的网络设备厂商在其现有设备上提供了编程接口供业务应用直接调用,也可被视作北向接口之一。如 Cisco 的 ONE 方案,基本不涉及原有网元设备的智能分离,仅通过管理面编程开放有限的能力,缺陷是创新业务的开发和部署困难,与硬件厂商捆绑缺乏兼容性。总体而言,当前 SDN 北向接口的发展一方面由各开源平台推动,力图通过支持更多网络应用落地,造成事实标准;另一方面则通过各个标准组织对北向接口的分类、框架、协议等进行标准化定义;还有学术界的知名 SDN 专家,站在本领域的研究前沿,也在积极探索北向接口的制定,希望继 OpenFlow 后,再次引领产业界的发展。

2 开源实现的 SDN 北向接口

2.1 OpenDaylight

OpenDaylight 旨在打造一个统一开放的 SDN 平台,如图 2 所示,应用可以通过 OpenDaylight 的北向接口 API 访问网络中的资源和信息。

OpenDaylight 控制器采用了模块化驱动

(model-driven) 的方法抽象出 SDN 控制器各个组件的南北向 API 以及各种服务和组件使用的数据结构,并且使用 YANG 数据建模语言[RFC6020]作为服务和数据抽象的建模语言。

YANG 初始作为 NETCONF 的建模语言被创建出来,在整个 NETCONF 的配置协议体系中,YANG 用来描述具体网元的配置、状态、通知时间、操作维护的数据信息以及之间的约束关系。在 NETCONF+NETMOD 联合配置框架下工作时,网元可以对客户端提交的配置等信息通过 YANG 描述的模型进行校验,保证其准确性,同时,也保证各厂商 NETCONF 协议支持的规范兼容性。一个使用 YANG 来描述的 OSPF 配置的示例,表明了一个 OSPF 的属性配置包括 Area 名称(ID)、启用 OSPF 的接口描述、Metric 的配置范围以及心跳保活检测的时间范围等。YANG 建模示例:OSPF 协议配置项以及约束如图 3 所示。

目前,YANG 的标准化将要完成,其应用范围已经完全脱离了前述的网元配置管理的数据模型描述范畴,部分厂商根据自身的需要,更多地采用 YANG 在 SDN 控制场景中来构建南北向的 API 定义,如上述的 OpenDaylight 社区。OpenDaylight 定义的 REST API,基于 YANG 来构建模型,然后由 YANG Tools 开源工具生成对外的 API 以及部分 Plug-in 框架代码,接口符合 IETF RESTCONF [RFC6241]规范。

2.2 Floodlight

Floodlight 是 Big Switch 控制器的开源版本, Floodlight 架构如图 4 所示,其北向接口也是基于 REST API。Floodlight 不仅提供了基本的网络状态查询、统计、配置接口,还提供了以下 3 种 API。

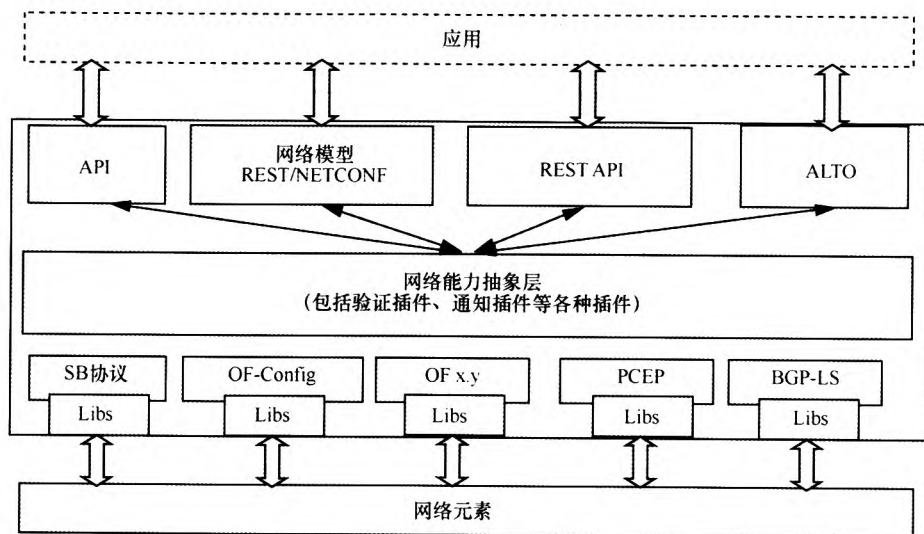


图 2 应用通过 OpenDaylight 北向接口访问网络资源

```

module example-ospf{
  namespace http://example.org/netconf/ospf;
  prefix ospf:
  import network-types//Access another module' s def' ns
    prefix nett;
  }
  container ospf{ //Declare the top-level tag
    list area{ //Declare a list of "area" nodes
      key name; //The key "name" identifies list members
      leaf name{
        type nett:area-id;
      }
      list interface{
        key name;
        leaf name{
          type nett:interface-name;
        }
      }
      leaf priority{
        description "Designated router priority" ;
        type uint8;//The type is a constraint on valid values for "priority"
      }
      leaf metric{
        type uint16{range 1..65535;}
      }
      leaf dead-interval{
        units seconds;
      }
    }
  }
}

```

图 3 YANG 建模示例:OSPF 协议配置项以及约束

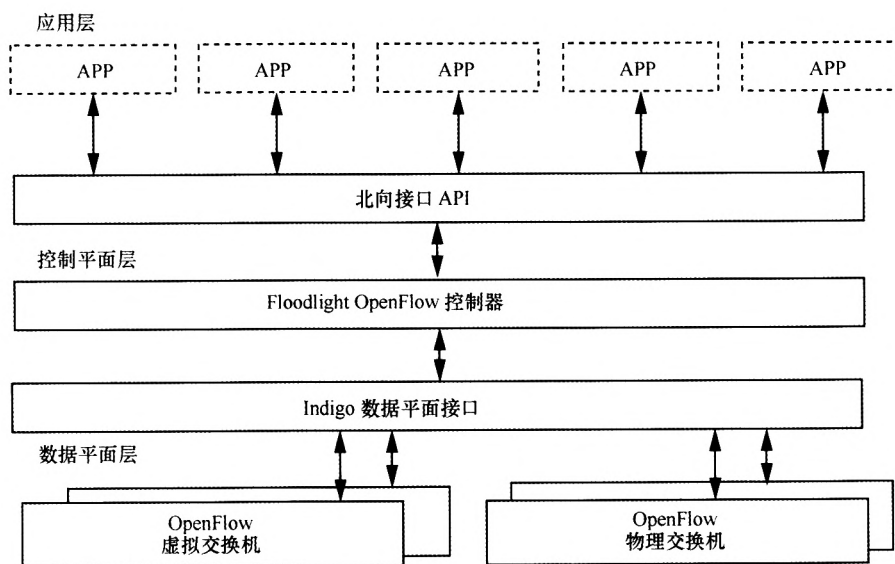


图 4 Floodlight 架构

(1)静态流推送器 API

静态流推送器 API 允许用户手动将流插入到 OpenFlow 网络,包括 OpenFlow 的主动添加和被动添加 2 种方式。

(2)虚拟网络 API

虚拟网络 API 负责创建新的虚拟网络的名称、ID、网关等,将主机加入到虚拟网络中等。

(3)防火墙 API

防火墙 API 负责防火墙的创建、开启、删除等操作。

2.3 Ryu

Ryu 采用组件化的架构,由 Python 语言编写,并提供大量库函数(组件)供 SDN 应用的开发。RyuSDN 框架如图 5 所示,针对北向的用户和应用,Ryu 框架也提供了丰富的 API。

(1)RESTful 管理 API

用户通过相应组件配置管理 SDN 网络的接口,可以通过 OF REST 组件 API 配置 OpenFlow 交换机;通过 Firewall 组件 API 配置防火墙等。

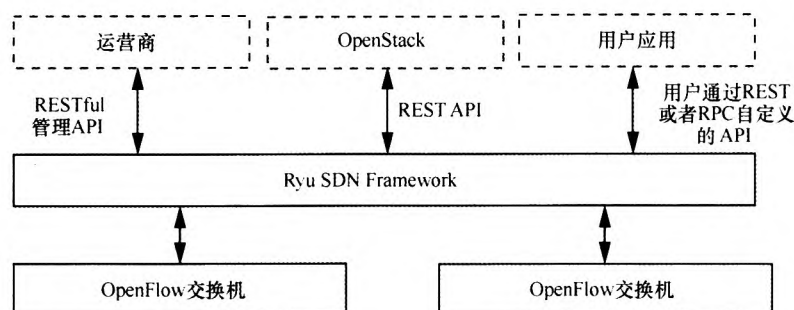


图5 RyuSDN 框架

(2)REST API

REST API 是与 OpenStack 云编排系统(orchestration)对接的接口。

(3)用户自定义 API

用户也可通过自定义的方式实现 REST API 或 RPC API。

2.4 OpenStack

2.4.1 Neutron

OpenStack 是 SDN 控制器上的一个重要应用, Neutron 是 OpenStack 中的“网络即服务”技术,它的作用在于帮助管理员创建、配置并管理软件定义网络体系。OpenStack 通过 Neutron API 向用户和其他服务公开虚拟网络服务接口,包括网络、子网和端口,但它并不实际执行工作,而是由 Neutron 插件负责与底层基础架构交互,以便依据逻辑模型来传送流量,插件可看作是 OpenStack API 的一种后端实现。

主流的 SDN 控制器大多已实现通过北向的 Neutron REST API 与 OpenStack Neutron 插件对接, OpenStack 对网络层面的调度交由对接的控制器完成。

Neutron API 分为以下2类。

(1)API Core

这是核心 API,可以看作是插件功能的最小集合,即每个插件都必须有的功能,也就是对网络、子网和端口的查询、加删和更新操作等。

(2)API Extensions

这是扩展 API,一般是针对具体插件实现的,这样租户就可以利用这些插件独特的功能,如 ACL 和 QoS。

2.4.2 Group-Based-Policy

Group-Based-Policy 是由 Cisco 和 IBM 主导,在 Openstack 平台新增的一个关于北向接口的开源项目。

2013 年 11 月, Cisco 收购 Insieme Networks 公司后,以该公司技术为核心,推出了以应用为中心的基础设施解决方案“ACI”。同时,在其主导的

OpenDaylight 社区创建新的孵化项目 Group Policy, 随后与 IBM 联合提案,在 Neutron 项目中增加以应用为中心的、面向策略的、抽象的接口项目,并最终命名为“基于组的策略抽象(GBP)”。

该项目的根本目标是制定更适用于应用所使用的、开放的北向 API,其出发点认为目前面向网络及其服务的北向接口比较复杂,不利于应用开发者使用,特别是那些不熟悉网络编程的人员,无法面对复杂的网络协议和功能,因此,需要提供面向应用策略的语义接口,能够使上层对网络的使用更简化。

GBP 项目将应用模型进行了抽象,将应用对网络的需求归纳成多个不同 Group 之间所需的、不同的连接关系。其中,Group 指具有相同策略需求的网络端节点(Endpoint)的集合,而不同 Group 之间采取的连接,附着对网络需求的策略,策略本身由一系列规则集合而成,这些规则决定了 Group 之间数据传输的行为与动作。例如,描述用户终端与一组 Web 服务器之间的连接关系为允许 HTTP 80/8080 等端口的 TCP 流量传输。

2.4.3 Congress Project

Congress Project 由 VMware 公司主导,于 2014 年初启动,目的是提供面向策略的北向接口(Policy as a service)。

作为云计算领域领头的 IT 公司,VMware 面对的主要是 IT 设施以及应用层面的开放,Congress 的提出也是对 Cisco(Vendor 厂商)面向应用部署、自动化方面渗透的 GBP 项目的一个应激防御,防止由 Vendor 厂商来主导面向应用层次接口的开放制订。

Congress 希望给应用提供一个关于部署、配置、管理等方面的接口框架,简化应用层对下层细节理解以及使用管理。主要涉及以下几个方面:

- 租户资源的配置(如 VMs, Applications);
- 自动化部署(如 application/VM locations);
- 资源应用自适应(如 Web App scaling based on load);

- 基础设施的管理(如 relationship between hosts、networks、storage)。

它允许应用层用特定的“策略语言”描述其对下层基础设施、资源使用的需求,告诉底层设施需要遵从哪些规则。“策略语言”使用 Datalog,常用于数据库查询语言领域,类似 SQL,但比 SQL 更接近于传统的编程语言。

相对于 GBP 的网络策略抽象,Congress 项目更侧重于 IT 服务的策略抽象,使得 IT 服务的部署配置更加自动化,管理更加便利。从层次上看,Congress 相对于 GBP 更接近于应用层,考虑更多的是应用部署与服务交付本身。

目前,该项目尚处于起步阶段,按照 VMware 的计划,会在近期公布更多的结果以及设计细节。

3 SDN 北向接口标准化

3.1 ONF NBI-WG

ONF 原本在架构与框架工作小组 (architecture framework WG, ARCH-WG) 中有一个研究小组在从事不同 SDN 解决方案的北向接口研究,为了响应产业对 SDN 北向 API 标准化的需求,新成立了北向接口工作小组 (NBI-WG)。

NBI 工作组目前尚未发布相关的标准草案,但是

在工作组的目标文档中提出,对于 SDN 北向接口应给出不同层次的抽象及接口范围(包括 Topology、Connections、Status、Flow、Policy 等),ONF 北向接口层次框架规划如图 6 所示。按照 NBI 工作组的计划,预计 2014 年将完成 NBI Framework 的草案,确定 NBI 体系的参考模型、网络能力、接口语言等定义,同时将发布典型的应用场景文稿。

ONF 期望定义并发展通用的以及一些特定领域的 API,其主要目的是为控制器、网络服务以及开发者提供可扩展且稳定的北向 API,增加软件设计上与 SDN 控制器交互运作的便利性,同时确保控制器供货商开发时能在共通 API 上自由创新。

3.2 IRTF 和 IETF 相关工作组

IRTF 已经成立了 SDN 研究工作组 (software-defined networking research group, SDNRG),工作组草案提出了 SDN 的层次化架构,其中,在管理和控制层面上给出了网络服务抽象层 (network services abstraction layer, NSAL),为应用、服务、控制和管理提供接入,也就是北向接口,具体的接口形式包括但不限于 RESTful APIs、RPC、公开的或特定的协议(如 NETCONF)、CORBA 接口,甚至内部进程间的通信 (IPC) 等。从 SDNRG 定义的 SDN 层次架构来看,相对于其他 SDN 北向接口范围,IRTF 认为北向接口应该具

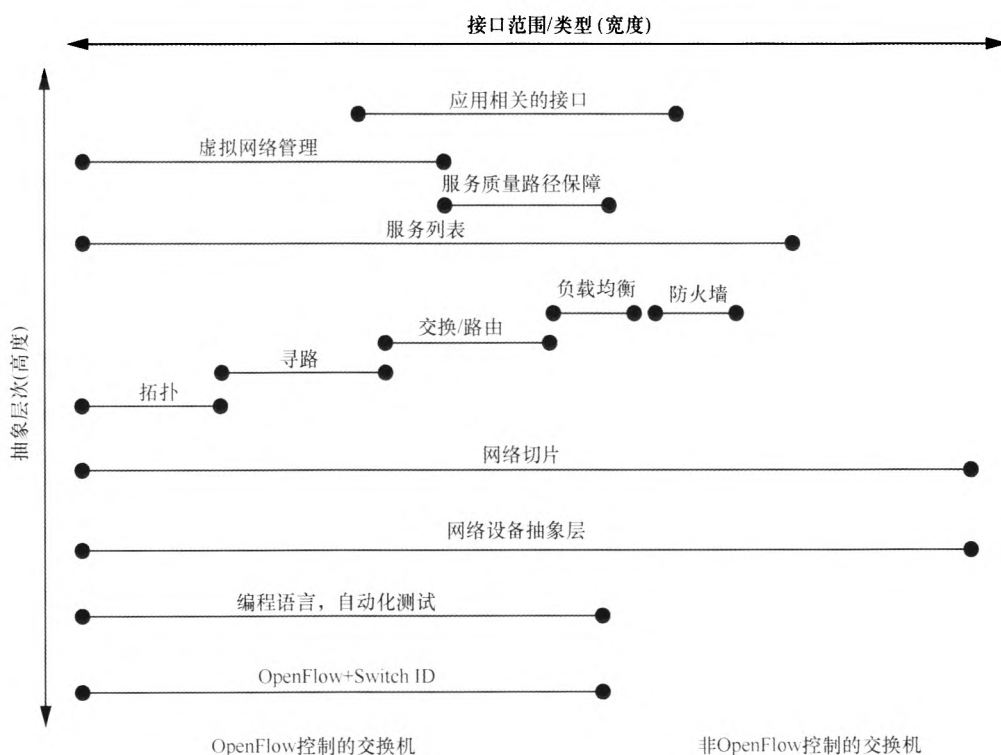


图 6 ONF 北向接口层次框架规划

有更多信息,整个设备、资源的管理平面也纳入到SDN北向接口的考虑范围中。SDNRG定义的SDN层次架构如图7所示。

IETF前期对于SDN的标准化多集中于南向接口技术,围绕传统的协议接口进行完善和扩展,并在一定程度上进行少量创新,试图在对传统网络不造成较大冲击的情况下适应SDN的需求,如ForCES[RFC5810]、BFD、PCEP、BGP的修改,新增I2RS以及像NICIRA(已经被VMware公司收购)提交的OVSDB等。北向接口方面,前期的ALTO可作为网络拓扑以及路径计算服务的开放接口,NETCONF管理协议,完成YANG、RESTCONF等一系列与SDN紧密相关的扩展,可作为北向接口的选择之一。另外,考虑到NFV以及NFV与SDN结合的影响,2013年IETF成立了SFC(service function chain)工作组,意图确立各网络功能服务整合的体系架构以及对外的接口,SFC的对外控制接口是SDN北向接口的重要组成部分。

4 学术界关于北向接口的研究

就像SDN概念体系源于学术界一样,学术界几乎对SDN开放的各个方面都有所探索,并且仍然致力于引领新的SDN方向。SDN北向接口也是众多学术研究方向的重点领域之一。

一方面,产业界出资与学术机构合作研究可商业化的原型系统实现,并以原型平台为基础,对业界推介SDN的北向接口定义,如ONIX、NOX、BEACON等都是类似这样的SDN控制器平台,进而自定义一套北向接口,在方式上类似目前开源界北向接口的制定思路。这类控制器平台接口更接近于传统控制器平台接口,能够或多或少地解决一些其他控制器平台无法涵

盖的问题。

另一方面,针对OpenFlow,在POF这类网络编程方面接近汇编式语言可能的缺陷,如暴露太多底层细节,模拟设备的转发行为,需要编程者在类似规则重叠、优先级冲突、数据面状态一致性等细节上做更多考虑,因此,编程效率低且容易出错。一些研究者致力于创造新的网络编程语言的研究,希望提供一个更高层次的、简化的转发设备编程接口,支持快捷的业务创新和代码重用,避免程序员过多地陷于网络细节,将更多精力投入到业务逻辑本身中。这方面研究的代表有Frenetic、Pyretic、Maple、Netcore等,解决的问题涵盖转发规则命令自动生成、编译效率、优先级冲突检测、冗余规则合并等诸多方面,对SDN控制器平台如何保证北向到南向的高质量转换提供了有价值的参考。

5 结束语

从技术实现上看,在开源组织、标准化组织,甚至工业领域,RESTful API是当前普遍认同的接口技术,其具备以下4个特点。

(1)可寻址性强

对应用而言,需要使用到的数据或算法片段,都会具有唯一的、可访问的资源地址。

(2)资源表述与操作

资源以JSON、XML、HTML等格式进行表述,通过表述操作资源。

(3)无状态

对每个请求而言,彼此之间是隔离的,服务器不保存“应用状态”,客户端无需事先获知如何和服务器交互。

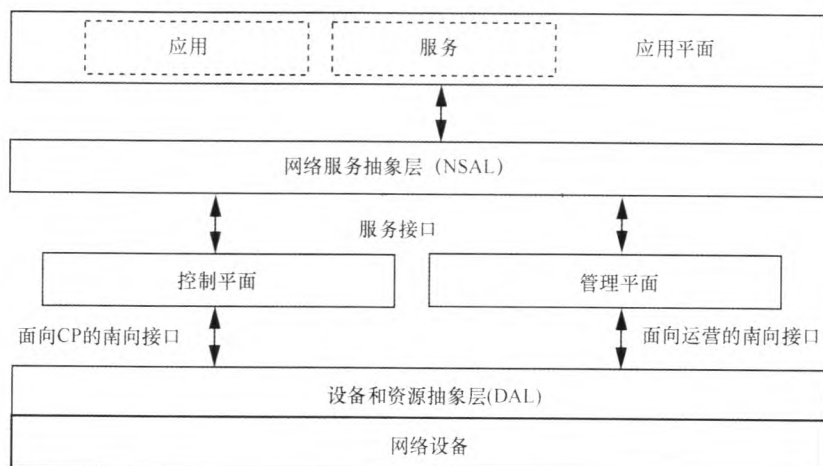


图7 SDNRG定义的SDN层次架构

(4)消息自描述

通过消息交互,每条消息包含足够的信息描述该条消息应该如何处理。

由于存在受控制器提供的接口能力的较大限制、在某些场合执行效率低等问题,RESTful API 不一定是所有情况下的最佳选择,因此,多数 SDN 控制器的架构和接口方案并不局限于 RESTful API,其他常见的 RPC、JAVA 接口、CORBA 接口等都是可选方案。另外,学术界也更多地偏向传统编程接口,也有部分研究者提出了新的面向 SDN 网络的编程语言。

网络应用主要是为了解决网络运作或管理上的问题而生,由于移动网络、数据中心、一般企业网络等网络环境的不同以及公众、金融、医疗、能源等网络应用领域不同,造成各类应用对于网络服务的需求与范围差异很大,ONF 也因此提出了多层次化、不同范围的接口架构,对接口层次的划分和聚类抽象是下一阶段北向接口标准化工作的重点内容。

从北向接口的发展趋势来看,当前 SDN 北向接口更多的是从网络的角度,如网元、链路、端口、网络业务实体等视角出发提供 API,随着这些“基础”的北向接口不断完善,从业务视角出发,按照业务应用描述、编排、流程化等需求对底层接口进行封装,接口形式像低级编程语言到高级语言一样演进,将会是北向接口技术的发展方向。

参考文献

- 1 程莹,张云勇.SDN 应用及北向接口技术研究.信息通信技术,2014(1)
- 2 Sarwar R, David L.Open Networking Foundation North Bound Interface Working Group (NBI-WG) Charter, <https://www.opennetworking.org>, 2014
- 3 Denazis S, Pentikousis K.SDN Layers and Architecture Terminology, <http://datacenter.ieee.org/doc/draft-haleplidis-sdnrg-layer-terminology>, 2014
- 4 Diego K, Fernando M V, Ramos. Software-Defined Networking: A Comprehensive Survey, <http://arxiv.org/abs/1406.0440>, 2014
- 5 Teemu K, Martin C, Natasha G, et al. Onix: A Distributed Control Platform for Large-scale Production Networks. Proceedings of the 9th USENIX Conference on Operating Systems Design and Implementation, 2010
- 6 Natasha G, Teemu K, Justin P, et al. NOX: Towards an Operating System for Networks. ACM SIGCOMM Computer Communication, 2008, 38(3)
- 7 David Erickson. The Beacon OpenFlow Controller. Hot SDN'13 Proceedings of the Second ACM SIGCOMM Workshop on Hot

Topics in Software Defined Networking, 2013

- 8 Nate F, Rob H, Michael J F, et al. Frenetic: A High-Level Language for OpenFlow Networks. <http://hdl.handle.net/1813/19310>, 2010
- 9 Reich J, Monsanto C, Foster N, et al. Modular SDN Programming with Pyretic. USENIX, 2013, 38(5)
- 10 Andreas V, Wang Junchang, Yang Y R, et al. Maple: Simplifying SDN Programming Using Algorithmic Policies. SIGCOMM'13, 2013
- 11 Christopher M, Nate F, Rob H, et al. A Compiler and Run-Time System for Network Programming Languages. POPL'12 Proceedings of the 39th Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, 2012

作者简介



庞涛

现就职于中国电信股份有限公司广州研究院,工程师,主要研究方向为云计算、SDN、业务平台与承载网络技术。



魏含宇

现就职于深圳市华为技术软件有限公司南京研究所,工程师,主要研究方向为 SDN/NFV 架构以及接口标准化、云计算、网络领域的增值业务。



武娟

现就职于中国电信股份有限公司广州研究院,高级工程师,主要研究方向为云计算、SDN、互联网及电信网络技术、大数据。



陈建

现就职于深圳市华为技术软件有限公司南京研究所,工程师,主要研究方向为 SDN/NFV、网络架构、网络增值业务。

SDN北向接口发展现状与趋势研究

作者: [庞涛](#), [魏含宇](#), [武娟](#), [陈建](#), [Pang Tao](#), [Wei Hanyu](#), [Wu Juan](#), [Chen Jian](#)
作者单位: [庞涛, 武娟, Pang Tao, Wu Juan\(中国电信股份有限公司广州研究院 广州510630\)](#), [魏含宇, 陈建, Wei Hanyu, Chen Jian\(华为技术有限公司 深圳518129\)](#)
刊名: [互联网天地](#)
英文刊名: [China Internet](#)
年, 卷(期): 2014 (9)

本文链接: http://d.wanfangdata.com.cn/Periodical_hlwtd201409012.aspx