Software-Defined Networking in Access Networks: Opportunities, Challenges, and Choices

Chen Tian, Huazhong University of Science and Technology, Wuhan, China
Jie Wu, Fudan University, Shanghai, China
Haibin Song, Huawei, Nanjing, China

ABSTRACT

Software-Defined Networking proposes to fundamentally change the current practice of network control. The two basic ideas are Centralized State Control and Uniform Device Abstraction, which support the Software-Defined promise. SDN has made significant progress. The opportunities of SDN in carrier access networks have been largely ignored by both industry and academia. In access networks, Quality-of-Service (QoS) oriented bandwidth management is more critical; the flexible QoS provisioning could be the most important opportunity for SDN. In this position paper, the authors show that the unique characteristics of access networks pose significant challenges to the two basic ideas. Contrary to the common agreement on “match-action” abstraction, the authors argue that the object-oriented abstraction might be a better choice for access networks to make a better software-defined implementation.

Keywords: Access Network, NETCONF, Object-Oriented Abstraction, Openflow, Quality-of-Service, Software-Defined Networking, YANG

INTRODUCTION

Software-Defined Networking (SDN) proposes to fundamentally change the current practice of network control (McKeown 2008; Shenker 2011). In recent years, there are significant research and implementation efforts for SDN from both industry and academia (Casado 2007; Sherwood 2009; Canini 2012; Monsanto 2013; Yu 2013; Jain 2013). The most prominent SDN achievement is the B4 Project (Jain 2013), where Google uses SDN to perform traffic engineering for its global inter-datacenter networks. Shown in Figure 1 (Sezer&Scott-Hayward, 2013), the SDN architecture has 3 layers: Application layer, Control layer and Infrastructure layer. The philosophy of SDN is that basic state distribution primitives should be implemented only once rather than separately for every control task.

DOI: 10.4018/IJWSR.2015010101
Leading by Open Networking Foundation (ONF), SDN promotes two basic ideas (Sezer&Scott-Hayward, 2013):

- **Centralized State Control**: For Control layer, a physically separated and logically centralized control platform handles state collection from all devices, makes decisions, and distributes the control state to them.

- **Uniform Device Abstraction**: For Infrastructure layer, devices of the forwarding plane could be controlled by a uniform open interface, which also removes the danger of vendor lock-in.

These two ideas together support the *Software-Defined* promise in Application layer: a fully programmatic interface upon which developers could build network management applications on (Koponen&Casado, 2010).

SDN has made significant progress. ONF advocates OpenFlow as the standard southbound interface defined between the Control and Infrastructure layers. In Control layer, there are many controllers emerged such as NOX (Gude&Koponen, 2008), ONIX (Koponen 2010) and Maestro (Cai&Cox, 2010). The OpenDaylight project promises to unify the northbound API between the Control and Application layers (Gopal, 2013). An additional OpenFlow management and configuration protocol is also proposed to remotely configure the control channel between the controllers and switches.

Besides its success in data-center and enterprise networks, people also realize the value of SDN in carrier networks (Elby, 2011), mostly in transport area (McDysan, 2013, pp.28-31). As a comparison, opportunities of SDN in access networks have been largely ignored by both industry and academia.

Access networks are different from data-center and enterprise networks. Devices in those networks (*i.e.*, switches and routers) focus on forwarding, which is well suited for the common "match-action" SDN abstraction. While in access networks, Quality-of-Service (QoS) oriented bandwidth management among

---

Copyright © 2015, IGI Global. Copying or distributing in print or electronic forms without written permission of IGI Global is prohibited.
Related Content

Aspect-Oriented Framework for Web Services (AoF4WS): Introduction and Two Example Case Studies
[www.igi-global.com/chapter/aspekt-oriented-framework-web-services/26082?camid=4v1a](www.igi-global.com/chapter/aspekt-oriented-framework-web-services/26082?camid=4v1a)

Transactional Composite Applications
[www.igi-global.com/chapter/transactional-composite-applications/26079?camid=4v1a](www.igi-global.com/chapter/transactional-composite-applications/26079?camid=4v1a)

Migrating Web Services in Mobile and Wireless Environments
[www.igi-global.com/article/migrating-web-services-mobile-wireless/4101?camid=4v1a](www.igi-global.com/article/migrating-web-services-mobile-wireless/4101?camid=4v1a)
A Service-Based Approach to Connect Context-Aware Platforms and Adaptable Android for Mobile Users
Valérie Monfort, Sihem Cherif and Rym Chaabani (2013). Adaptive Web Services for Modular and Reusable Software Development: Tactics and Solutions (pp. 302-332). www.igi-global.com/chapter/service-based-approach-connect-context/69480?camid=4v1a