RAON: Recursive Abstraction of OpenFlow Networks

Sae Hyong Park, Byungjoon Lee, Jaeho You, Jisoo Shin, Taehong Kim, Sunhee Yang
ETRI (Electronics and Telecommunications Research Institute), Korea
{labry, bjlee, jaehoyou, jshin, taehong.kim, shyang}@ etri.re.kr

***Abstract*— This paper describes RAON which is a novel solution for tackling scalability in OpenFlow networks. This approach recursively abstracts its underlying networks as OpenFlow switches to reduce complexity and to increase manageability. The design and early prototype of RAON shows very promising and graceful use cases of various scenarios.**

Keywords— *SDN, OpenFlow, Recursive, interconnection, domain, heterogeneity, abstraction*

I. INTRODUCTION

In software defined networking (SDN), a controller provides a unified view of the network. This unified view offers simplified configuration and management of the network. Further, in WAN settings, this unified view enables unprecedented high utilization of the links by managing end-to-end paths [1]. However, as the size of a managed SDN network grows, providing a global view causes scalability issues in terms of complexity. There are many attempts to solve SDN scalability issues: 1) logically centralized controller: multiple master controllers only solve controller performance issue, but it does not address the core architectural issue with an increase in complexity as an SDN network scales. 2) WE-Bridge [2] which is a SDN peering mechanism offers a way to interconnect SDN networks, but does not guarantee the advantages of SDN's centralized view. Centralized architecture, in nature, best suits with integration of heterogeneous networks within the same ownership since inter-domains integration of different owners necessitates restricting and hiding of many internal topology and settings. These existing proposals can only be remedies, but cannot be offered as a solution to the scalability problem in SDN networks.

In this paper, we propose RAON as a fundamental solution to SDN scalability issue. RAON stands for recursive abstraction of OpenFlow networks, but it also means "domain" in Scottish Gaelic. RAON, in a nutshell, recursively abstracts a single controller domain as a large OpenFlow switch. "Abstract" means representing selected characteristics, while hiding or summarizing characteristics as the administrator of the domain desires. This capability provides reduced complexity, centralized view, and graceful integration of OpenFlow networks, which means super controllers are oblivious to the existence of underlying controllers and transparently controls underlying domains as if it were large OpenFlow switches. In other words, it does not require any changes in super controllers in terms of architecture and protocols.

II. ARCHITECTURE OF RAON

In this section, we propose a system called RAON which gracefully integrates unit OpenFlow networks while maintaining the advantages of centralized view. RAON consists of the following components: (1) unit OpenFlow networks, (2) super controllers, and (3) OpenFlow gateways. A unit OpenFlow network is an OpenFlow network controlled by a single physical or logical OpenFlow controller. A super controller is a relative concept in a hierarchical OpenFlow network that indicates a higher-level OpenFlow controller. However, the super controller sees an underlying OpenFlow controller as an OpenFlow switch since in RAON architecture, a lower-level controller exports its controlled domain as an OpenFlow switch to its high-level controller, and the higher-level controller can transparently handle the underlying domain as it were a real OpenFlow switch. OpenFlow gateways are intermediary devices between two or more OpenFlow domains that either perform tunnelling for compatibility with legacy networks or provide interconnection to another OpenFlow domain.

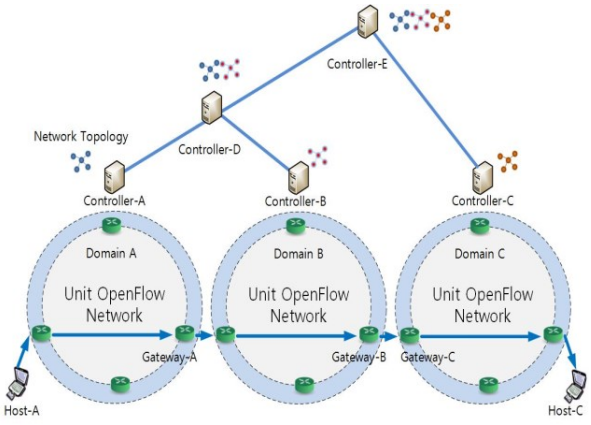

Figure1. Integration of unit OpenFlow networks

In Figure 1, we illustrate a workflow of transmitting packets from host A to host C for clarity's sake. When transmitting packets, host A sends packets destined for host B. The OpenFlow switch which host A is attached to detects a table-miss and relays the packets to the controller A. The controller A also relays the packets to the controller D and to the controller E. The controller E finally knows whole topology sends flow modification to its underlying controllers as if they were OpenFlow switches, and the underlying controllers translate received flow modification to appropriate flow modifications for its network. After the flow rules are all set-up, host A and C can communicate without sending more packets to any controllers.

A. *Network Zoning*

A Traditional OpenFlow network does not provide any means to partition its network into many areas or zones like Open Shortest Path First (OSPF) network [3] for si-

978-1-4799-6919-7/14 $31.00 © 2014 IEEE
DOI 10.1109/EWSDN.2014.29

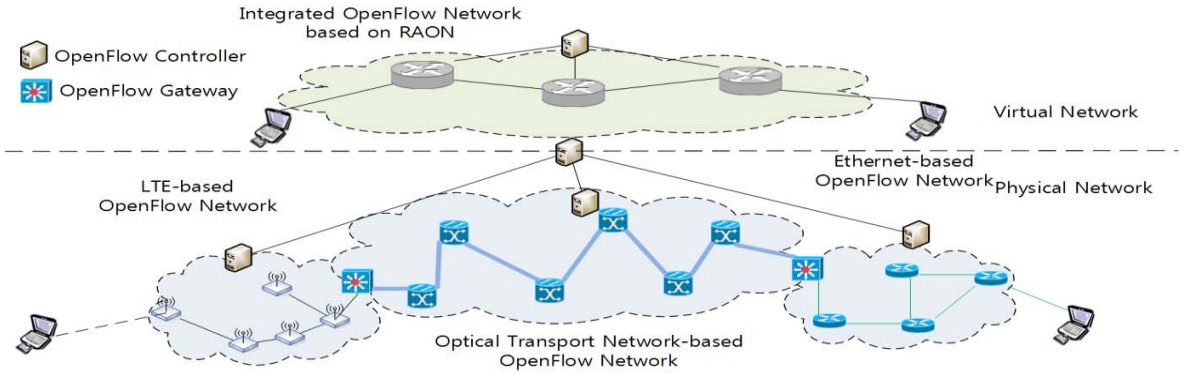

Figure 2. RAON for integration of heterogeneous networks

mplifying administration or optimizing traffic and resource utilization. However, as the complexity of one administrative domain increases, the need for partitioning becomes evident. RAON is particularly well-suited for such a case since RAON naturally partitions the administrative domain per controller while maintaining the centralized view at the highest controller.

B. *Integration of Heterogeneous Networks*

Traditional 2G/3G mobile networks were circuit switched centric networks, but mobile networks are evolving to all-IP based network as it becomes 4G LTE and beyond. Recent trend in networking such as carrier-grade Ethernet and MPLS over Ethernet makes OpenFlow based network integration more plausible. This integration of heterogeneous networks can greatly reduce CAPEX and OPEX in enterprise and carrier-grade networks. Figure 2 depicts RAON-based integration. RAON-based integration is best suited for an intra-Autonomous System (AS) domain for higher link utilization and optimization of networks, but it can also be applied to inter-AS domains since network administrator can hide internal network topology as much as needed.

III. Early Prototype and Outlook

RAON implementation has two parts: a RAON switch and a RAON OpenFlow controller abstraction module that communicates with the RAON switch. We implement the RAON switch based on BigSwitch's Indigo Virtual Switch (IVS) [4] and the OpenFlow controller abstraction module as a module in OpenIRIS [5]. IVS is virtual switch composed of Indigo switch agent, Open vSwitch, and Loci OpenFlow libraries. The RAON switch basically replaces Open vSwitch from IVS with an interface module for communication with OpenIRIS. Therefore, to a higher controller, underlying controller is viewed as an OpenFlow switch. The RAON controller module is in charge of abstracting its controlled domain and sending it to a higher OpenFlow controller as OpenFlow messages. In other words, the controller's both south and north interfaces become OpenFlow and theoretically there is no limit to the number of levels in the controller hierarchy. When OpenIRIS receives OpenFlow messages, the RAON module acquires the messages through PacketIn pipeline or inter-module communication channel and performs abstraction (i.e., XID, cookie, and table mapping, hiding detailed groups, ports, tables and meters, etc.) on the received messages. After processing, the module transmits the messages to the RAON switch. Upon reception, the RAON switch updates its internal data structures with received messages. To reduce the inter-controller overhead, we are implementing local termination mechanism for many inter-controller messages especially for statistical requests and replies. However, the RAON switch relays command messages such as packet_out, flow_mod, group_mod, table_mod, and meter_mod. We are also designing a mechanism to deal with OpenFlow version difference between domains with OpenFlow message conversion mechanism. Currently, we are implementing OpenFlow 1.0 to 1.3 conversions to minimize the loss of information during conversion. The implementation is still ongoing.

We are expecting RAON to solve many of the OpenFlow scalability issues, and assist SDN and OpenFlow to gain more momentum by providing a means to have unified control over heterogeneous networks.

Acknowledgment

This research was funded by the MSIP (Ministry of Science, ICT & Future Planning), Korea in the ICT R&D Program 2014

References

[1] S. Jain et al. B4: Experience with a globally-deployed software defined WAN. In SIGCOMM, 2013.
[2] P Lin et al. WE-Bridge: West-East Bridge for SDN Inter-domain Network Peering. In INFOCOM, 2014.
[3] Thomas II T. OSPF network design solutions. Cisco Press, Indianapolis, 1998.
[4] http://www.projectfloodlight.org/indigo-virtual-switch, by Big switch
[5] http://openiris.etri.re.kr/, ETRI
[6] James McCauley, Aurojit Panda, Martin Casado, Teemu Koponen, and Scott Shenker. Extending SDN to large-scale networks. In ONS, 2013
[7] B. Ager, N. Chatzis, A. Feldmann, N. Sarrar, S. Uhlig, and W. Willinger. Anatomy of a large european ixp. In Proc. of ACM SIGCOMM, 2012.