M2cloud: Software Defined Multi-site Data Center Network Control Framework for Multi-tenant

Zhongjin Liu, Yong Li, Li Su, Depeng Jin, Lieguang Zeng
Department of Electronic Engineering, Tsinghua University, Beijing 100084, China
liyong07@tsinghua.edu.cn

ABSTRACT

A significant concern for cloud operators is to provide global network performance isolation for concurrent tenants. To address this, we propose M2cloud, a software defined framework providing scalable network control for multi-site data centers (DCs). M2cloud employs two-level controllers with decoupled functions, providing each tenant with flexible virtualization support in both intra- and inter-DC networks.

Categories and Subject Descriptors

C.2.1 [**Network Architecture and Design**]

General Terms

Design

Keywords

SDN; multi-tenant; multi-site; data center networks

1. INTRODUCTION

Cloud service providers possess multiple data centers in distributed places, over which they provide diverse applications and resilient infrastructures. In order to provide high available services to global users, especially deliver the fast-growing mobile content to mobile users[2, 3], large volumes of data need to be moved from one site to another, making the inter-DC traffic to be a major portion of backbone links.

Data center tenants deploy their applications in collections of virtual machines (VMs) placed in one or more sites. Cloud operators have a strong requirement to enforce inter-DC traffic optimization and cross-site performance isolation for tenants, but current solution cannot provide joint intra- and inter-DC network control ability [1]. In this paper, we present a software defined multi-site data center control framework supporting virtualization across multiple data centers, which allows tenant specific performance optimization. Two-level controllers are introduced with layered functions including device configuration, network virtualization and decoupled APIs, which balance between scalability and performance. Simulation results show that the inter-DC bandwidth utilization can be improved, and the global workload can be balanced through M2cloud.

SIGCOMM'13, August 12–16, 2013, Hong Kong, China.
ACM 978-1-4503-2056-6/13/08.
http://dx.doi.org/10.1145/2486001.2491725

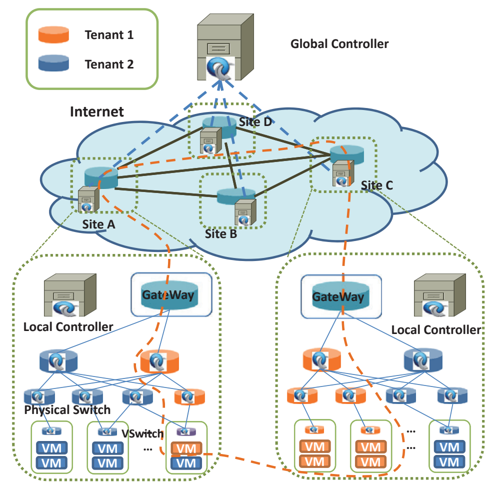

Figure 1: M2cloud Architecture

2. DESIGN PRINCIPLE

2.1 Design Goals

M2cloud is designed with the purpose of enabling flexible network control in both intra- and inter-DC networks, which provides exclusive isolated network environment for each tenant. Besides flexibility brought by the software defined approach, M2cloud also intends to strike a balance between scalability and performance.

2.2 Network Control Architecture

The key idea of M2cloud lies in two-level controllers. We note that the network functions required by data center applications can be decomposed into two categories: functions such as forwarding table configuration, which require low-latency and high-performance processing, should be handled locally; while functions such as inter-DC traffic engineering and global workload balancing, which benefit from the global view of data centers, should be processed on top of inter-DC network. Based on this observation, M2cloud adopts function-decoupled controllers in intra- and inter-DC networks, whose architecture is shown in Fig. 1.

The geographically distributed data centers are interconnected via programmable border gateways that are controlled by a logically centralized global controller. The traffic of different tenants can be easily distinguished and directed through flow level control to support dedicated tenant performance optimization. The global controller has the full view of interconnection network, which enables global traffic engineering as well as workload balancing among DCs.

Network devices in one data center are controlled by a local controller, which installs flow rules to both physical and software programmable switches. Since VMs are directly connected to software switches, local controller has

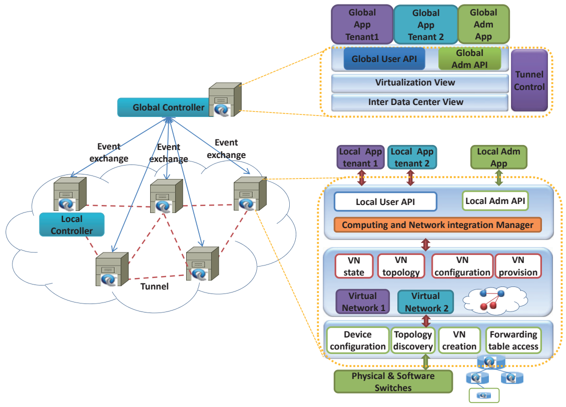

Figure 2: Building Block of M2cloud

the insight of both switches and VMs. Resources such as flow table, traffic, bandwidth and topology can be sliced as what FlowVisor [4] does. Therefore, the controller can slice VMs, traffic and even physical links into a tenant's virtual network.

2.3 Building Blocks

It does not make much sense if controllers work independently of each other, which is impossible to support cross-site virtualization for tenants. Therefore, M2cloud enables interconnection among controllers to support joint intra- and inter-DC resources slicing. The local controllers are connected to global controller, via which they exchange local events and share network view with local controllers in other sites. The global controller and the local controller have the same layered structure but different in functions. The building block of controllers are shown in Fig. 2.

Global controller. The lowest layer of the global controller monitors topology of interconnection network and realtime traffic. Virtualization layer provides each tenant with separate DC-DC pathes. Application program interfaces (APIs) are opened to administrators and tenants for different purposes. Administrator APIs are able to manage substrate devices and build virtual network for tenants. Tenant APIs only manage his virtualized network resource, including topology control and traffic engineering, etc. An important extension to the global controller is the tunneling mechanism, which builds tunnels among several local controllers in different sites. Thus local controllers are able to exchange local information with each other.

Local controller. Network operator is able to exploit functions in lowest network "driver" layer to configure switches, monitor topology, build virtual networks and access flow tables inside a data center. Virtual network control functions are provided in virtual layer. The abstraction layer are integrated with VM management tools such as OpenStack in order to uniformly manage both computing and network resources. APIs are also authorized to administrators and tenants with different scopes of control abilities.

3. PRELIMINARY RESULTS

To illustrate the ability of M2cloud to support cross-site management, we present a use case that migrates VMs across data centers, whose procedure is shown in Fig. 3.

To show the flexibility and efficiency of proposed framework, we evaluate the VM migration performance with numerical simulation. Concurrent 10GB VM migration re-

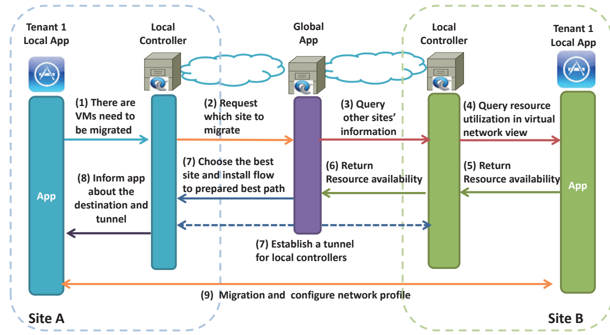

Figure 3: Cross sites VM migration in M2cloud

quests are generated among 30 data centers shared by two tenants. Topology is randomly generated and bandwidth of per link is set to be 100Mbps. Randomly generated background traffic is used to simulate client traffic. In the contrast setting, the migration destinations are set to a random site, and the shortest path routing is used. In M2cloud, VMs are able to migrate to the site with lowest workload, and traffic of different tenants is able to go through the optimal pathes with highest bandwidth individually. The performance comparison is listed in Table 1 with contrast setting compared to M2cloud. The bandwidth utilization can be improved by more than 24%, and the workload tends to be more balanced.

Table 1: Performance Comparison

Tenant No.	Migration time	Bandwidth utilization	Workload std. deviation
1	34.6/17.5min	38.5%/76.2%	23.76%/8.2%
2	34.6/21.3min	38.5%/62.6%	23.76%/10.65%

4. ACKNOWLEDGEMENT

This work is supported by National Basic Research Program of China (973 Program) (No. 2013CB329105), National Nature Science Foundation of China (No. 61171065, No. 61021001, No. 61133015), National High Technology Research and Development Program (No. 2013AA010601 and No. 2013AA010605), and Program for Changjiang Scholars and Innovative Research Team in University (PCSIRT).

5. REFERENCES

[1] M. Banikazemi, D. Olshefski, A. Shaikh, et al. Meridian: an sdn platform for cloud network services. *Communications Magazine, IEEE*, 51(2):120–127, 2013.

[2] Y. Li, P. Hui, D. Jin, et al. Evaluating the impact of social selfishness on the epidemic routing in delay tolerant networks. *IEEE Communications Letters*, 14(11):1026–1028, November 2010.

[3] Y. Li, Z. Wang, D. Jin, et al. Collaborative vehicular content dissemination with directional antennas. *IEEE Transaction on Wireless Communications*, 11(4):1301–1306, April 2012.

[4] R. Sherwood, M. Chan, A. Covington, et al. Carving research slices out of your production networks with openflow. *ACM SIGCOMM Computer Communication Review*, 40(1):129–130, 2010.