SIMON: Seamless servIce MigratiON in Mobile Network

Dung Ong Mau, Jialun Wang, Long Wang, Long Hu,
Yujun Ma, and Yin Zhang(✉)

School of Computer Science and Technology, Huazhong University of Science and Technology, 1037 Luoyu Road, Wuhan 430074, China
{omdung,jialun.cs,longwang.epic,longhu.cs,yujun.hust}@gmail.com,
yin.zhang.cn@ieee.org

Abstract. In the era of cloud and mobile social computing, a plethora of mobile applications demand mobile stations (MSs) seamlessly interact with the cloud anyplace in a real-time fashion. The mobility characteristic may cause a service on MS to be migrated between different Data Centers (DC); otherwise, a packet delay is increased due to the fact that a considerable geographical distance between MS and serving DC. With a current networking architecture, IP address of either MS or virtual machine (VM) is changed because of the VM migration, and an IP session between two peers is released and the service is disrupted as a result. In this paper, we leverage the emerging Content Centric Networking (CCN) as a straightforward solution to these issues. Based on unique content name identification, instead of IP address, service migration can be continuous. Furthermore, a seamless service migration framework is proposed to conduct the user's service request to the optimal DC, which satisfies user requirements, minimizes the network usage and ensures application Quality of Experience (QoE).

Keywords: Data Center · Cloud Computing · Virtual Machine Migration · Content Centric Networking

1 Introduction

Virtualization technology provides a better way to utilize computation resources, improve scalability, reliability and availability while decrease operation costs. With a huge number of advantages, virtualization technology is gaining more and more interest in high-performance computing.

A hardware virtualization technique allows multiple operating systems as known as virtual machines (VM) to run on a single cluster of physical hardware. The visualization hides underlying computing system and presents an abstract computing platform by using a hypervisor (or virtual machine manager) [1]. In data center (DC), the number of physical machines can be reduced by using virtualization that consolidates virtual appliances into shared servers, which help to improve the efficiency of Information Technology (IT) systems.

V.C.M. Leung et al. (Eds.): TridentCom 2014, LNICST 137, pp. 261–270, 2014.
DOI: 10.1007/978-3-319-13326-3_25

Virtual appliances are virtual machine images containing everything needed to run a particular task (operating system, databases, configuration, software, etc.) that can be used as a *"black box"*. Obviously, it allows multiple virtual machines to be run on a single physical machine in order to provide more capability and increase the utilization level of the hardware.

Virtual machine (VM) migration is another important technique that always accompanies with the virtualization technology. Migration of VMs is a useful capability of virtualizing clusters and data centers. VM migration supports for the VM to be moved between physical machines (e.g. between two servers in the same DC) as known as live VM migration technique. Hence, the flexible management of available physical resources by making load balance and maintaining the infrastructure are easy to be done by VM migration and do not affect to available applications located on. VM migration is expected to be fast and VM service degradation is also expected to be low or even non-interruption during migration. In some previous works, the envision of VM migration occurs within internal DC and the VM needs to reserve the IP address [2,3]. However, it is a great challenge to migrate a VM to a different DC, as routine network operations would not allow non-continuous IP assignment.

It should be noted that VM migrates between different DCs only when all DCs are running the same hypervisor platform. Normally, cloud provider distributes many DCs in the large scale due to the growing business or other feasible condition when all DCs are handled by the same cloud provider [4]. In this paper, leveraging the emerging Content Centric Networking (CCN) strategy, we propose a novel method to support the VM migration without any service interruption. Specifically, communication with VM is identified via the name of the service running on it instead of IP address. The routing is found by the service name. In this way, services and VMs are decouple from their locations. Under framework proposed, our research focuses on designing a VM migration protocol between DCs and optimizing the Quality of Experiment (QoE). Furthermore, communications are expected to provide ubiquitous connectivity and continuously between machines and humans with Seamless servIce MigratiON (SIMON), instead of human intervention [5].

The remainder of this paper is organized as follows. In Section 2, we give the motivation and problem statement. Section 3 introduces the proposed seamless service migration framework. Section 4 portrays the envisioned network architecture of the simulation setup and introduces and discusses simulation results. Finally, Section 5 concludes this paper.

2 Motivation and Problem Statement

Cloud computing has been greatly developed in the IT market in recent years. With the multiple advantages and features, cloud computing supports a wide range of services, and the cloud provider upgrades its network by expanding a number of DCs. Thus, DCs are distributed over a wide area network (WAN) to cope with the growing number of business and users.

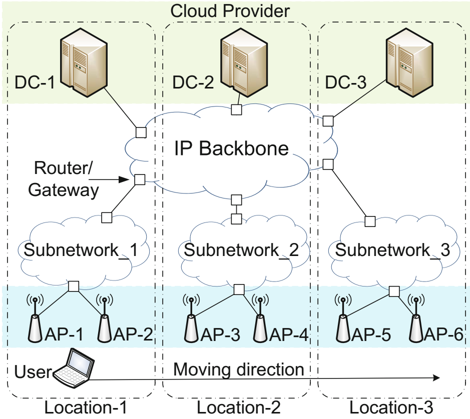


Fig. 1. Distributed Data Center

Fig. 1 presents three separate DCs under the same cloud provider that are geographically distributed and interconnected by the IP backbone network. In general, users in the location-1 access to the service running on DC-1. However, when a user moves to the location-2, it is likely that the user accesses to the subnetwork-2 but still uses the service on DC-1 in location-1 which is far from the user. The longer distance between the user and the appropriate VM, the more communication resources consume. Furthermore, it results in high packet latency and poor Quality of Experience (QoE). This intuitively results in an inefficient connectivity service caused by the absence of an optimal end-to-end connectivity. Therefore, there are following challenges.

– What is the suitable time to request for the optimal VM? Which entity can take a role to find out the optimal VM? And how to find out the optimal VM?
– Suppose that the user had enough information about the optimal VM, then VM is need to migrate. With the current network operation based on IP address, the IP address changing on VM will cause the IP session between user and VM being released.

This paper proposes a solution that address all these issues, defining a seamless service framework that interworks between a distributed DCs. Hereunder, the key feature of the proposed solution is to replace IP address connection by service/data identification in the context of Content Centric Networking (CCN).

3 Seamless Service Migration Framework

In this section, we consider a network topology showed in Fig.1 and we add a new component named *Controller* as illustrated in Fig.2. The controller can be an independent entity, or a software embedded into DCs [6,7].

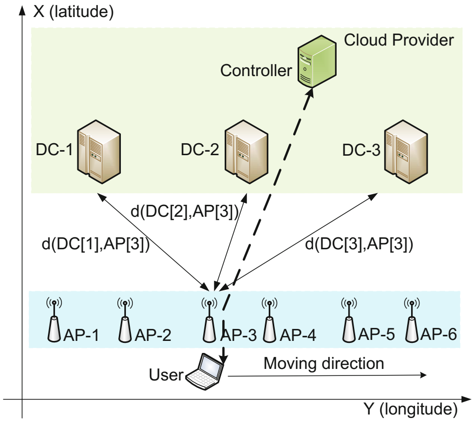


Fig. 2. Controller entity

In this paper, we focus on how to minimize a geographically distance between a mobile station (MS) and the optimal DC. Let $d(DC[i]AP[j])$ denote the distance between DC i and Access Point (AP) j, the controller can select the optimal DC k for AP j based on Equation (1) and (2).

$$d(DC[i]AP[j]) = \sqrt{(X_{DC[i]} - X_{AP[j]})^2 + (Y_{DC[i]} - Y_{AP[j]})^2} \tag{1}$$

$$k_j = argmin\{d(DC[i]AP[j]), \forall i = 1, ..., N\} \tag{2}$$

Functions of the controller include:

- Collecting and updating position, eg. latitude and longitude, of all DCs within a cloud provider.
- Pointing out the optimal VM to the MS based on the minimum distance location between the MS and the DC.
- Decision and management for VM migration between DCs.

By leveraging the concept of CCN [8], we replace IP address by content name identification. A specific application designed for CCN strategy is installed on MS, AP and DC. CCN is not only a theoretical strategy but also a capable of the real-life implementation. Indeed, many projects and prototypes have been implemented with CCN successfully [9–12]. CCN decouples location from identity and communication, and enables continuous communication in dynamic network environment. In CCN, the MS can switch to other network and continue to reserve former session or service regardless IP address. The service is identified by a location-independent unique name instead of the network address. Therefore, the MS can access the service without awareness about the allocation and migration of VMs.

Fig.3 shows the flow of messages and procedures carried out to establish a service session migrated to a different DC. In the initial state, DCs update their

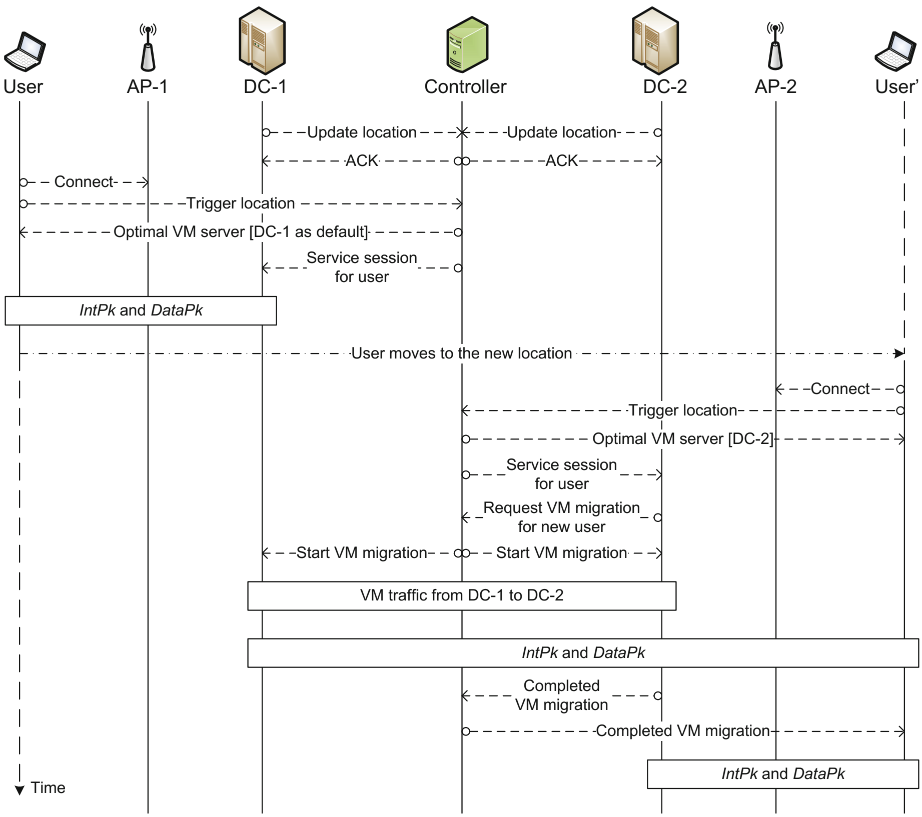


Fig. 3. Flowchart for the procedure of service migration

positions to the controller, which is confirmed by an acknowledgment (ACK) packet. A MS stays within the coverage of a radio station of AP-1. After connected successfully, the MS sends a trigger location packet to the controller. Based on the current location of the MS and a table of DCs' location, the controller selects the appropriate DC (as known as DC-1 in this example) and establishes a service session between MS and DC-1. As a default, we suppose that the service application for the MS is available on the DC-1.

In the second state, during the mobility of the MS, the MS becomes aware of either non-available AP or new AP service set identification (SSID). When connecting to a different AP, the MS will send a trigger location packet to the controller. Once the controller finds that the service is not available at DC-2, the service should be migrated from DC-1 to the optimal DC-2. Then a service session will be established between the MS and DC-2. In spite of a change in the IP address of both the MS and DC server, the service session continues without being torn down as all request and content are identified by the unique name.

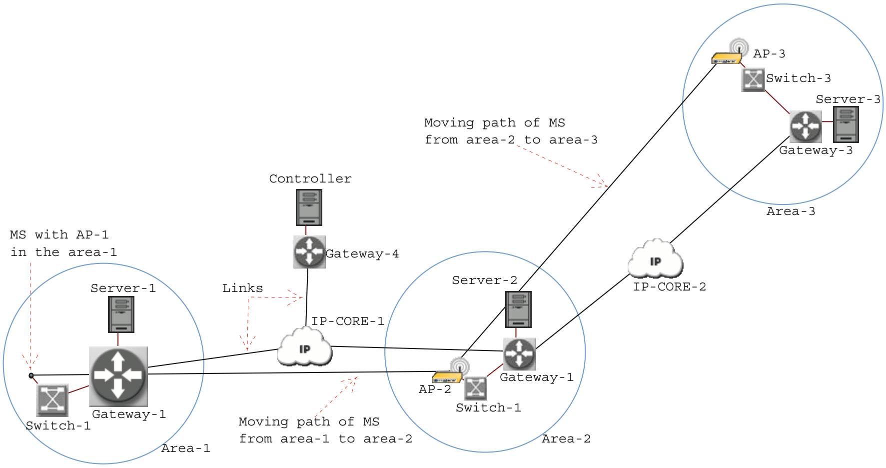


Fig. 4. Network simulation

4 Simulation and Results

To evaluate the performance of proposed service migration framework, we implemented CCN strategy and conducted simulations using the OPNET Modeler 16.0 [13,14]. In the simulations, CCN is overlayed over the IP layer. Indeed, we integrated the CCN processing modules into all network elements, such as MS, AP, routers, servers and IP Cloud.

4.1 Network Simulation

Fig.4 is a network simulation to demonstrate a procedure of the service migration in DC. There are three separate areas connected to IP core, which makes some delay for each packet passed through, e.g. constant distribution 0.01s. Each area includes one AP and one server. It should be noted that the server presents one or more VM servers.

In order to clarify the procedure of service migration in this scenario, there is only one MS moving from area-1 to area-2 and area-3. The trajectory of this MS is presented in Table 1. And Table 2 shows all the simulation parameters.

4.2 Simulation Results

Fig. 5 illustrates the status connectivity of the MS to APs and the time stamp tracking for using optimal server. Because of the mobility of the MS, it connects to APs with the SSID $1; 2; 3$ respectively, while the SSID -1 indicates that non-AP is available. In the dashed line, the value one indicates that MS currently uses the optimal server and the value zero indicates that the MS is waiting for VM migration while it continues using non-optimal server.

Table 1. Trajectory of the MS

(X;Y) (m)	**Distance (m)**	**Traverse time (s)**	**Ground speed (km/h)**	**Wait time (s)**	**Accom. time (s)**
(0;0)	n/a	n/a	n/a	1800	1800
(6200;64)	6200	558	40	1800	4158
(10291;4774)	6239	561	40	1800	6519

Table 2. Simulation parameters

Element	**Attribute**	**Value**
WAN	Link between Gateway	OC-24 data rate
	Link for Switch,AP and server	1000 BaseX
	IP core latency	Constant 10ms
	Wireless interface	IEEE802.11g
CCN	CCN directory	*ccn://epic_lab/myVM_i*
	IntPk interval	50ms
	IntPk size	32B
	DataPk size	1500B
VM	VM size	1GB
	VM transfer rate	$12.10^6 bps$

Fig. 6(a) presents for the Round Trip Time (RTT) and data bit rate received by the MS. In Fig. 6(a), when the MS uses the optimal server, the RTT reaches a minimum delay because of the shortest distance between the MS and optimal server. However, as waiting for VM migration, the MS must continue using non-optimal server. As shown in Fig. 4 about network simulation, when the MS is not in the same area with its serving server, the IntPk and DataPk go through IP Core and got some delay. Therefore, Fig. 6(a) clearly shows that RTT in-case-of non-optimal server is greater than RTT in-case-of optimal server about $0.02s$.

Fig. 6(a) also illustrates the seamless VM migration in the context of CCN. Typically, the IP-connection will be dropped down when either source's or destination's IP are changed. Leverage from the concept of CCN, the data conversation is based on unique name of content and CCN supports well for the mobility of both producers and consumers.

Fig. 6(b) illustrates a traffic transmission between DCs caused by VM migration. In this simulation, we support that the size of a VM is 1GB and be transferred with a constant rate ($12.10^6 bps$). Hence, it takes about $716s$ to complete transfer the VM. It also means that the MS keeps an former conversation with the un-optimal server within the first $716s$ until moving on the optimal server during the service migration.

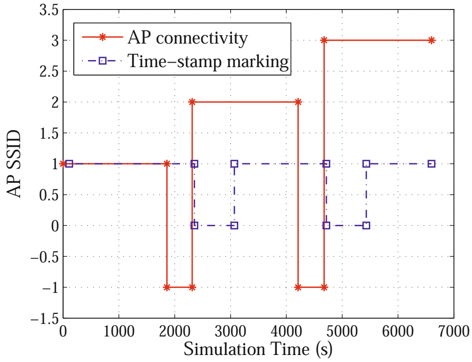
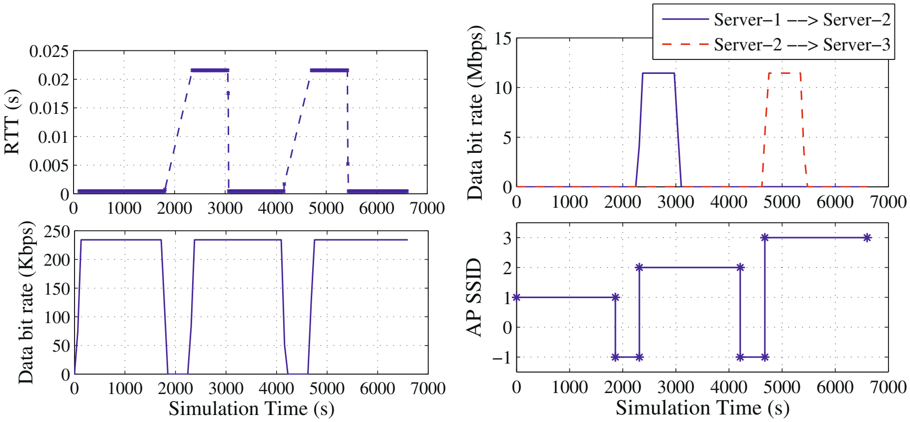


Fig. 5. AP connectivity and the time stamp tracking for using optimal server



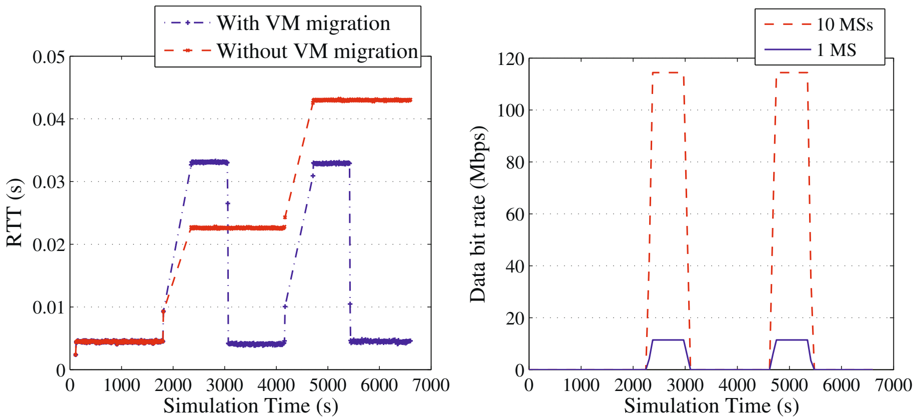
(a) RTT and data bit rate at MS. (b) VM migration traffic for one MS.

Fig. 6. Simulation results for one MS

4.3 Evaluation

In order to evaluate the performance of our proposed framework, we expand the simulation by increasing the number of MSs up to 10. Moreover, we make a comparison between two mechanisms: with and without VM migration. Each MS occupies a separate VM in the server and distinguished by the name of content, (e.g. $ccn : //epic\_lab/myVM\_[i]$), where i refers to the identification of MS. The increasing number of MSs will affect most of simulation results, e.g. RTT and total VM migration traffic.

Fig. 7(a) shows a big gap of RTT between these two mechanisms. Obviously, without service migration, the RTT will increase as along as the MS is moving. In the contrary, with service migration, the MS does not only get the data latency delay because of a long communication, but also gets extra data latency because of the data burst caused by transferring VM traffic between servers.



(a) The gap of RTT at MS between with/without VM migration.

(b) Total VM migration sent and received traffic for 1MS and 10MSs, respectively.

Fig. 7. Simulation results for 10 MSs in two scenarios

Finally, Fig. 7(b) illustrates the VM traffic rate sent and received by servers. In this scenario, all MSs have different services and each MS occupies one VM in the server. Moreover, all MSs have the same trajectory, so their service migration consume the same time.

5 Conclusion

In this paper, we investigate the challenges of seamless VM migration for DC. Then we introduce the framework supports continuity and QoE of mobile client with seamless service migration. It is considered that CCN strategy can be overlayed on any kind of network transmission (e.g. Internet layer is the most popular). Hence, the framework is thus highly feasible, practical, and standards-compliant without any major complexity being added to the current network architecture.

References

1. Zaw, E.P., Thein, N.L.: Improved Live VM Migration using LRU and Splay Tree Algorithm. International Journal of Computer Science and Telecommunications **3**(3), 1–7 (2012)
2. Anala, M.R., Kashyap, M., Shobha, G.: Application performance analysis during live migration of virtual machines. In: 2013 IEEE 3rd International Advance Computing Conference (IACC), pp. 366–372 (February 2013)
3. Li, K.K., Zheng, H., Wu, J.: Migration-based virtual machine placement in cloud systems. In: 2013 IEEE 2nd International Cloud Networking (CloudNet), pp. 83–90 (November 2013)

4. Fernando, N., Loke, S.W., Rahayu, W.: Mobile cloud computing: A survey. The International Journal of Grid Computing and eScience **29**(1), 84–106 (2013)
5. Zheng, K., Hu, F., Wang, W., et al.: Radio resource allocation in LTE-advanced cellular networks with M2M communications. IEEE Communications Magazine **50**(7), 184–192 (2012)
6. Mann, V., Vishnoi, A., Kannan, K., Kalyanaraman, S.: CrossRoads: Seamless VM mobility across data centers through software defined networking. In: 2012 IEEE Network Operations and Management Symposium (NOMS), pp. 88–96 (April 2012)
7. Taleb, T., Hasselmeyer, P., Mir, F.G.: Follow-Me Cloud: An OpenFlow-Based Implementation. In: 2013 IEEE and Internet of Things (iThings/CPSCom) Green Computing and Communications (GreenCom), pp. 240–245 (August 2013)
8. Jacobson, V., Smetters, D., Thornton, J., Plass, M., Briggs, N., Braynard, R.: Networking named content. Communication of the ACM **55**(1), 117–124 (2012)
9. Named Data Networking. http://www.named-data.net/index.html
10. NS-3 based Named Data Networking (NDN) simulator. http://ndnsim.net/index.html
11. Scalable and Adaptive Internet Solutions (SAIL) project. http://www.sail-project.eu/
12. Publish-Subscribe Internet Technology (PURSUIT) project. http://www.fp7-pursuit.eu/PursuitWeb/
13. OPNET Modeler. www.opnet.com
14. Chen, M.: OPNET Network Simulation. Press of Tsinghua University (2004) ISBN 7-302-08232-4