# Study of Data Placement Schemes for SNS Services in Cloud Environment

**Yen-Wen Chen*[1], Meng-Hsien Lin[1,2] and Min-Yan Wu[1]**
[1]Department of Communication Engineering, National Central University
Tao-Yuan City, Taiwan
[2] Chunghwa Telecom Laboratories Taoyuan, Taiwan 32661
*Corresponding author: Yen-Wen Chen
[e-mail : ywchen@ce.ncu.edu.tw]

*Received April 9, 2015; revised June 22, 2015; accepted July 8, 2015;
published August 31, 2015*

## Abstract

Due to the high growth of SNS population, service scalability is one of the critical issues to be addressed. The cloud environment provides the flexible computing and storage resources for services deployment, which fits the characteristics of scalable SNS deployment. However, if the SNS related information is not properly placed, it will cause unbalance load and heavy transmission cost on the storage virtual machine (VM) and cloud data center (CDC) network. In this paper, we characterize the SNS into a graph model based on the users' associations and interest correlations. The node weight represents the degree of associations, which can be indexed by the number of friends or data sources, and the link weight denotes the correlation between users/data sources. Then, based on the SNS graph, the two-step algorithm is proposed in this paper to determine the placement of SNS related data among VMs. Two k-means based clustering schemes are proposed to allocate social data in proper VM and physical servers for pre-configured VM and dynamic VM environment, respectively. The experimental example was conducted and to illustrate and compare the performance of the proposed schemes.

***Keywords:*** Social network services, Cloud computing, Virtual machine, Data placement, K-means clustering

A preliminary version of this paper was presented at ICONI 2014, and was selected as an outstanding paper. This version includes a complete algorithms for different application environment and supporting results on placement performance. This research work was supported in part by the grants from the National Science Council (NSC) and Ministry of Science and Technology (MOST) (grant number: 103-2218-E-194-007), Taiwan, ROC.

**http://dx.doi.org/10.3837/tiis.2015.08.027** **ISSN : 1976-7277**

## 1. Introduction

The rapid development of network technologies enables the increasing of network applications. And various kinds of network services have been deployed and changed the human life style. Among them, in addition to the traditional telephone, VoIP, skype, and messaging services, the social network services (SNS) create an alternative interaction space among people [1-3]. In the social space, people can be easily linked together, share information, interact to each other in real time, etc. Thus, SNS establish the social cyberspace that is implemented in virtual. Generally, the SNS can be simplified to be constituted of people entities, information, and the relationships among them. The relationship may exist between users, information, or either. And the most complex is the dynamic and time varing behaviors of the above components. In order to provide a scalable SNS system, the deployment environment shall be flexible enough to support the needs of SNS.

The cloud computing is designed in common resource pool concept and the services run on cloud environment can request the computing and storage resources as it needs. The virtual machines (VM) and network function virtualization (NFV) approaches enable the flexibility in resource allocation [4, 5]. Therefore, it is quite straightforward to implement SNS in cloud environment to achieve scalability for service deployment. Basically, the cloud computing achieves its scalability and efficient performance through the parallel processing concept. It is easy to figure out that the system performance will be downgraded if bottleneck occurs due to the unbalance resource allocation in the processing components. The SNS data exists correlation properties and can be represented as a social graph where the link represents the correlation between them. The SNS data shall be placed in several data VM through the cloud data center (CDC) network. Therefore, we need to arrange SNS data to be placed in proper data VM so that the processing and transmission overhead can be minimized. Thus, we need to map the social network topology into the CDC network topology as shown in **Fig. 1**. And, as a result, the issue of load balance for both of the physical servers and the transmission network is critical to maximize system performance in cloud computing environment.

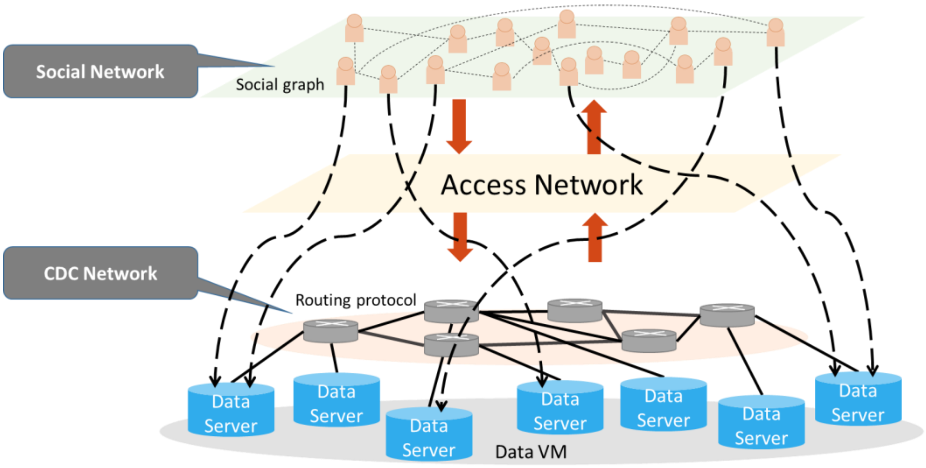

**Fig. 1.** The social network and CDC network

It is well known that the associations among people entities and correlation of information are complex but essential in SNS. The association and correlation properties dominate the behavior of information retrivals in the SNS cyberspace. Therefore, the SNS data shall be properly placed in the cloud environment so that the loads of storage VMs and network links could be balanced to achieve better user experience in using SNS. The main purpose of this paper is to propose a heuristic procedure for SNS data placement to achieve the above objective. The proposed scheme adopts the correlation property to group users information by using k-means clustering and distributes users or data sources among VMs. Two data placement schemes are proposed for both pre-configured VM and dynamic VM situations in this paper. The experimental example is provided to illustrate the performance achievement of the proposed scheme. The rest of the paper is organized as follows. The related works are overviewed in the following section. In Section 3, the proposed SNS data placement scheme is described with the illustrative example. The conclusions of this paper and future works are provided in the last section.

## 2. Related Works

As mentioned in previous section that cloud computing environment provides on demand resource allocation. Therefore, the flexible management of VM and networking resources is one of the critical challenges toward efficient cloud services [6]. Although the VM and NFV can be flexibly adjusted when necessary, it may require much adjustment effort if data is improperly placed. The SNS information inherently has high correlation and association in its nature, therefore, the strategy of data placement is crucial for service quality. The VMPlanner scheme was proposed in [7] to optimize the VM placement and traffic flows so as to achieve energy saving. In [8], they proposed a cost-aware meta-heuristic algorithm to deal with Cost-Aware VM Placement Problem (CAVP). Solving the problem approximates the best trade-off point between electricity cost and wide-area-network (WAN) communication cost while minimizes the geo-distributed cloud system’s operating cost. In [9], to achieve better VM sizing with effective system performance, it proposed a novel dynamic VM placement scheme which utilizing predicted mean and variance together with correlation among VMs in data center management.

In addition to VM placement, recently, several literatures were devoted to the data placement in cloud environment. In [10], the authors proposed a data placement structure by clustering row groups so as to satisfy fast data loading, fast query processing, highly efficient storage space utilization, and strong-adaptive to highly dynamic workload patterns. The data placement scheme, which adaptively balances the data amount among nodes, was proposed in [11]. Their placement scheme distributes data across multiple heterogeneous nodes according to their computing capabilities so as to improve data processing performance. Both of the topology-aware and hardware-aware VM placement schemes were proposed by considering locations of physical machines that related to the associated VMs to achieve load balance [12]. In [13], the adaptive resource management scheme was proposed to provide fast and reliable cloud services.

Although the above schemes achieve some degrees of the load balance in cloud environment, the data correlation and association properties were not well considered. In [14], to meet multiple system objective of data placement over multiple geographically distributed clouds, it designed a scheme to optimize a formula based on their need of system objective. The scheme divides the formula into two sub-problems and using greedy algorithm and graph cuts to deal with them. But it is designed for geo-distributed cloud. In [15], the authors considered the placement and migration of social aware data in opportunistic network. In their proposed model, the link weight represents the association degree between societies. And the K-median approach is applied to maximize the system performance. Hence it is the objective of this paper to study the data placement issue from the view point of SNS data correlations.

## 3. Unsupervised Clustering

The factors that affect the data delivery performance in cloud environment can be briefly classified in to the processing cost and transmission cost. The processing cost is affected by the load of allocated VM as well as the physical server the VM resided. And the transmission cost is determined by the transmission path and the traffic condition. For example, as shown in **Fig. 2**, if the associated data is placed in VM 5 and VM 7, respectively, then the transmission path will go through path 2 which travels for 6 hops. Additionally, the numbers of VMs, which are co-located in the same physical host of VM5 and VM 7, are 3 and 2, respectively. Their load may be higher than VM 3, whose physical host has only one VM. Alternatively, if the correlated data is in VM1 and VM 3, respectively, then the processing load can be improved and the transmission cost, i.e. path 1, is also less than path 2.

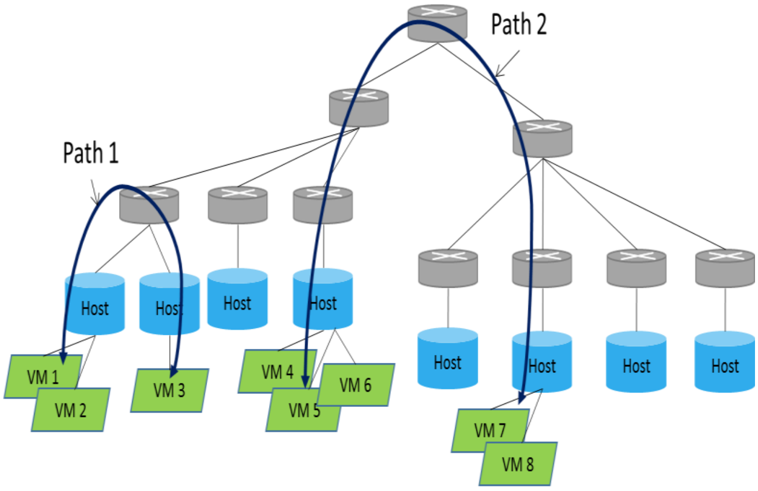

**Fig. 2.** Costs of data placed in different VMs

In the proposed scheme, the SNS is modeled into a correlated graph, where the node represents the activity of the user or data source in the SNS, e.g. the number of associations (or friends) the user or data source has. The link denotes the correlation between two users (or data components), e.g. the number common societies, the interaction frequency, the common interests, the number of common friends, etc. The purpose of the proposed scheme is to properly allocate the nodes to the VMs and associated physical servers so that the transmission cost can be reduced according to the social relationship graph. Two approaches are proposed in this paper: one is the placement for VM pre-figured in physical servers; the other is the placement for flexible VM in physical servers.

**(A) The placement in pre-configured VM environment**

The pre-figured VM approach is suitable for the placement when the number of available VM is fixed in each physical server. Assume that S is the set of servers and there are n physical servers, i.e. S={1, 2, 3, …n}. And there are $v_i$ VMs pre-configured in the physical server i (i=1~n). The proposed scheme is divided into three steps to place the SNS data as shown in **Fig. 3**.

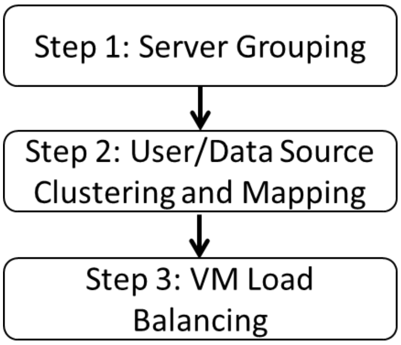
Step 1: Server Grouping
Step 2: User/Data Source Clustering and Mapping
Step 3: VM Load Balancing

**Fig. 3.** Placement procedures for the pre-configured VM environment

(1) Server grouping:
The first step is to group physical servers into groups so that all servers of the same group can communicate to each other within the transmission cost d. For example, if transmission cost is the number of hops between two servers, then the grouping will put the servers, which have d hops to each other, together. Let m be the number of server groups after grouping then

$$G = \{g_i, g_2, ... g_m\} \qquad (1)$$

$$g_i = \{x_{i,1}, x_{i,2} ...\} \qquad x_{i,j} \in S \qquad (2)$$

For the extreme case, when d equals to 0, then it means that each physical server is itself a group and m=n.

(2) User/data source clustering and mapping:
In this step, users are clustered into m categories according to their link weight (i.e. correlation) by using K-means approach first. And the m users/data sources clusters will be allocated to the m server groups in one to one manner. As server groups may have different number of VM, we define $C_{g_i}$ as the capacity of the server group $g_i$ and it is obtained in equation (3).

$$C_{g_i} = \sum_j v_{x_{i,j}} \qquad (3)$$

As the node number of the user cluster conceptually reflects the processing load, therefore, the user cluster with more user nodes is mapped to the server group with higher capacities of server groups. The process is as follows:
- Use k-means algorithm to divide nodes into m user cluster according to link weight;
- Sort the server groups according to $C_{g_i}$;
- Sort the user clusters according to the number of users of the cluster;
- Perform one-to-one mapping between user clusters and server groups.

(3) VM load balancing:
This step is to place the data of the users of the same cluster to the VMs of their associated server group. In order to balance the load among VMs, the node weight is considered during the data placement process so that the load difference among VM can be minimized. Thus we prefer to arrange the weights summations of the nodes being

allocated to VMs are close to the mean value. It can be achieved by sorting the node weights first before placement.

**(B) The placement in dynamic VM environment**

The above placement scheme assumes that all VMs are pre-configured in physical servers, however, in some case; the cloud operator may dynamically rearrange VMs through VM migration tool without interrupting service to achieve better processing performance. Thus, the numbers of VMs and physical servers are fixed; however, we can arrange any number of VMs to each of the physical servers. In order to reduce the transmission cost, the hierarchical k-means clustering schemes proposed. The basic concept is to group nodes into VMs first to obtain the VM correlation topology, and then to cluster VMs for individual physical servers according to the transmission cost among physical servers. The procedure is provided in the following **Fig. 4**.

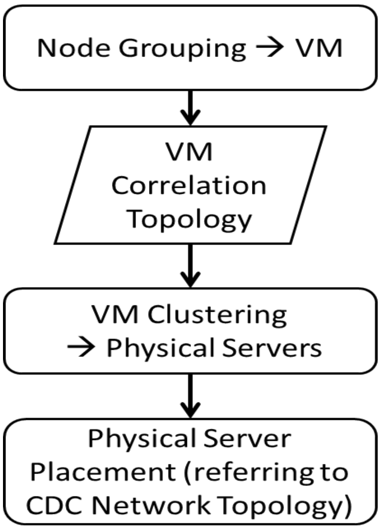
Node Grouping → VM
VM Correlation Topology
VM Clustering → Physical Servers
Physical Server Placement (referring to CDC Network Topology)

**Fig. 4.** Placement procedures for the dynamic VM environment

The number of VM to be placed is $V = \sum_{i=1}^{n} v_i$ . The nodes are grouped into V VMs by using K-means procedure in the first step. We then can obtain the VM topology where the link weight, $l_{(VM)i,j}$ represents the correlation between VM i and j. The value of $l_{(VM)i,j}$ is calculated by the summation of the link weights of the two nodes that each is included in either VM as shown in the equation (4).

$$l_{(VM)i,j} = \sum_{(x \in VMi) \wedge (y \in VMj)} d_{x,y} \tag{4}$$

An example is provided in **Fig. 5.** to illustrate the construction of VM correlation topology. The topology has 7 nodes and is grouped into 3 VMs by using K-means scheme. Then the correlation VM topology is constructed by 3 VM nodes and the link weights are calculated according to equation (4).

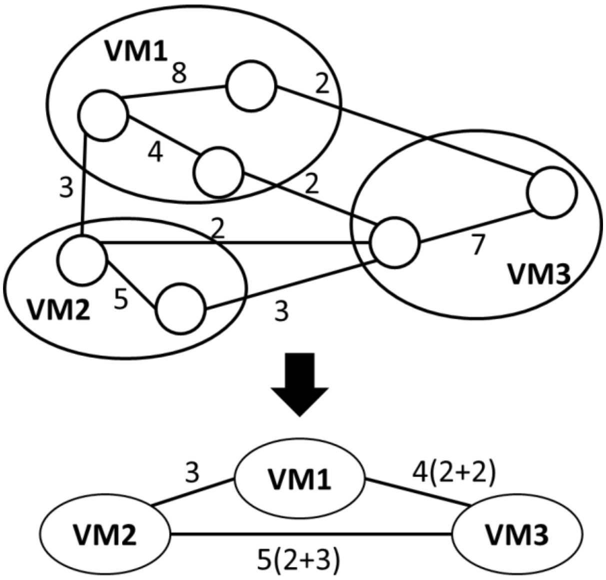

**Fig. 5.** Construction of VM correlation_topology

In the third step, the VM correlation topology is further grouped into n clusters, where n is the number of physical servers. The purpose of this step is to allocate VMs (VM grouping), which are tightly correlated, to the same physical server to reduce the transmission cost. The mapping between clusters and physical servers is performed by referring to the network topology so that the transmission cost can be further minimized.

## 4. Experimental Example and Analysis

In order to investigate the effectiveness of the proposed pre-configured VM and dynamic VM schemes, the following experimental example is provided. Assume that the social graph consisting of 20 users/data sources (nodes) en-numbered as shown in **Fig. 6**. The node weights represent the number of associations of users/data sources and their assumption values are listed in **Table 1**. The link weight, which represents the data access frequency (i.e. correlation) between two users, is specified beside each link.

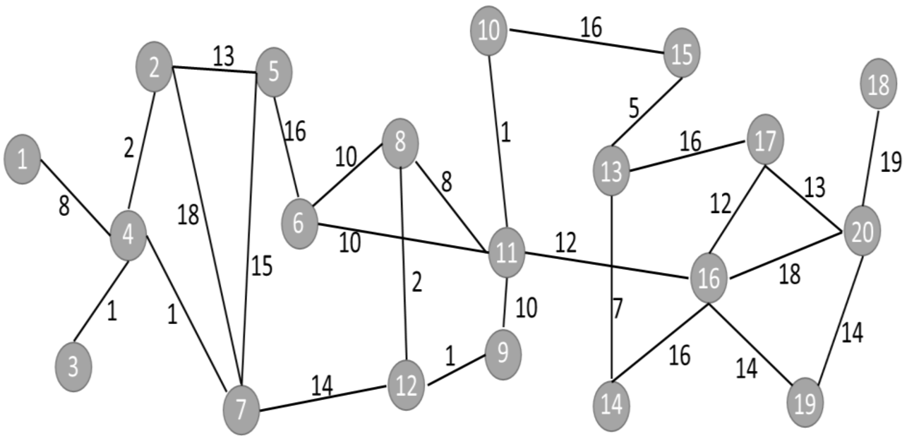

**Fig. 6.** Example of social graph

**Table 1.** Node weights of the example

| ID | Weight | ID | Weight | ID | Weight | ID | Weight |
|---|---|---|---|---|---|---|---|
| 1 | 5 | 6 | 4 | 11 | 8 | 16 | 14 |
| 2 | 7 | 7 | 10 | 12 | 2 | 17 | 2 |
| 3 | 6 | 8 | 5 | 13 | 11 | 18 | 7 |
| 4 | 4 | 9 | 1 | 14 | 13 | 19 | 11 |
| 5 | 8 | 10 | 4 | 15 | 1 | 20 | 3 |

The network topology is assumed to consist of three physical servers and six VMs. For the pre-configured VM scheme, we further assumed the 6 VMs were pre-configured to be 1, 2, and 3, in each physical server, respectively as shown in **Fig. 7**. Here we assume d=0, which means that each physical server is itself formed one server group. The numbers of VM of each physical server are 1, 2, 3, respectively. We took the number of hop counts as part of the transmission cost. It shows that the numbers of hops between physical servers (1, 2), (2, 3), and (1, 3) are 4, 5, and 5, respectively. If two VMs, which are co-located in the same physical server, we assume that it will have no transmission cost between them. And for the dynamic VM case, the network topology, numbers of physical servers and VMs are the same; however, VM can be flexibly allocated to either physical servers.

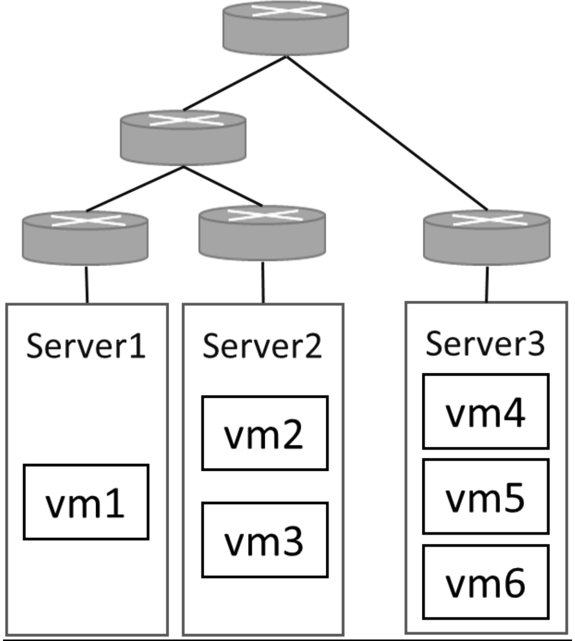

**Fig. 7.** Example of cloud servers topology

### (A) Experimental results of pre-configured VM case

By referring to the step (2) of the proposed VM pre-configured placement scheme, the nodes were clustered into three node clusters through k-means process and then the three nodes were mapped to the physical servers as shown in the **Table 2**.

**Table 2.** Nodes clustering and mapping

| Cluster ID | Nodes | Server ID |
|---|---|---|
| 1 | { 16, 17, 18, 19, 20 } | 1 |
| 2 | { 1, 2, 3, 4, 5, 7, 12, 13, 15 } | 3 |
| 3 | { 6, 8, 9, 10, 11, 14 } | 2 |

And, in the step 3, the nodes of the same server were fairly placed in the associated VM according to the node weights. The final placements of nodes are provided in **Table 3**.

**Table 3.** Placement of nodes in VMs (Pre-configured VM)

| VM ID | Nodes |
|---|---|
| 1 | { 16, 17, 18, 19, 20 } |
| 2 | { 6,14} |
| 3 | { 8,9,10,11} |
| 4 | { 1,4,7} |
| 5 | {2,5,12} |
| 6 | {3,13,15} |

We compare the results to the scheme of even node distribution scheme. The even node distribution scheme randomly allocates almost the same number of nodes to each VM. The effectiveness of the data delivery is compared in terms of the load balance and the transmission cost. And the variance of VM loads is applied to indicate the degree of load balance. The smaller the variance is the more balance of the VM loads. The transmission cost is obtained by the product of the number of hops and the access frequency (i.e. the link weight) between two nodes. The performance of the proposed scheme and the node balance scheme is compared in the following **Table 4**. In addition to the load variance and transmission cost, we calculated the effectiveness by the product of the above two indexes. As it prefers to have smaller load variance and lower transmission cost, the smaller the product value is, the better the effectiveness is.

**Table 4.** Comparisons of the proposed pre-configured scheme and the node balance scheme

| Schemes | Node variance | Transmission cost | Effectiveness |
|---|---|---|---|
| Proposed Pre-configured Scheme | 51 | 828 | 42228 |
| Node Balance | 72 | 1498 | 107856 |

The results of **Table 4** illustrate that the proposed scheme demonstrates better performance than the node balance scheme in either index. For the load balance, although the node balance scheme fairly allocates nodes to each VM, the even distribution of number of nodes does not result in load balance because the node weight is not considered. The proposed scheme carefully considers the correlations of the social graph of SNS and adopts the k-means to cluster tightly related nodes in step 2; therefore, it is the main reason that the transmission cost is greatly reduced in our scheme when compared to the node balance scheme.

**(B) Experimental results of the dynamic VM case**

In the dynamic VM case, VMs are flexible to be arranged in any selected physical server. As there are 6 VMs, the 20 nodes are clustered into 6 groups, one for each VM, in the first step by using K-means calculation. The results of clustered nodes and the associated VM correlation topology are obtained in **Table 5** and **Fig. 8**.

**Table 5.** Clustering of nodes in VMs (Dynamic VM)

| VM ID | Nodes |
|---|---|
| 1 | {2,4,5,7} |
| 2 | {6,8,9,11} |
| 3 | { 1,12 } |
| 4 | {10,15} |
| 5 | {13,14,17} |
| 6 | {3,16,18,19,20} |

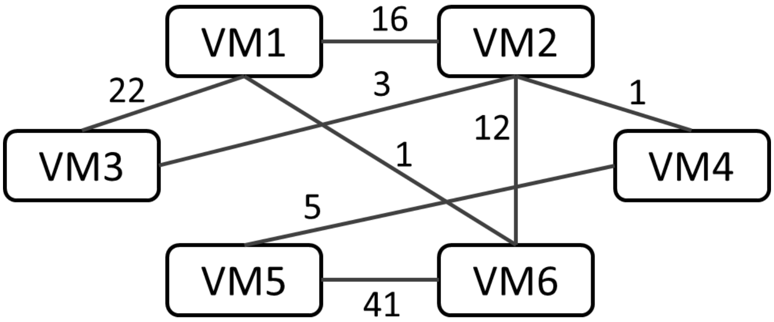

**Fig. 8.** VM correlation topology

In order to allocate the 6 VMs for the three physical servers, the K-means scheme is applied for the VM correlation topology to obtain 3 groups to be allocated for the three physical servers as shown in **Table 6** and its associated topology in **Fig. 9**.

**Table 6.** VM grouping

| Group ID | VM ID | Nodes |
|---|---|---|
| 1 | 1 | {2,4,5,7} |
| | 2 | {6,8,9,11} |
| | 3 | { 1,12 } |
| 2 | 4 | {10,15} |
| 3 | 5 | {13,14,17} |
| | 6 | {3,16,18,19,20} |

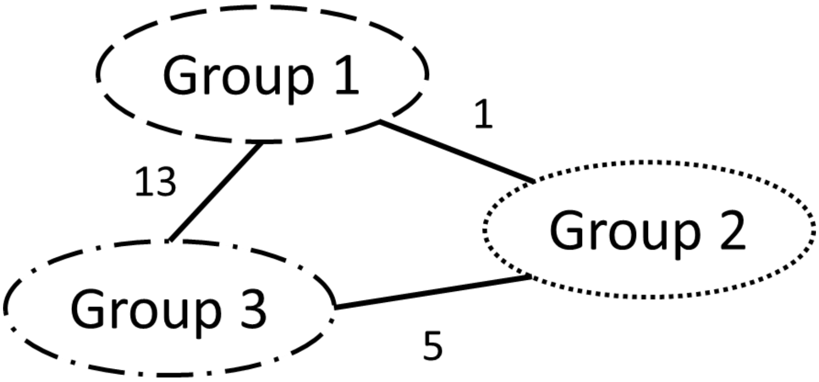

**Fig. 9.** Group topology

Then, in the last step, we need to arrange the above three groups to the three physical servers. As the link weight presents the access frequency between groups, therefore, we need to locate groups, which have the largest link weight, in the physical servers that have the smallest distance. In this example, the numbers of hops between physical servers (1, 2), (2, 3), and (1, 3) are 4, 5, and 5, respectively, and the link between group 1 and 3 has the largest weight, according to the topology in **Fig. 10**, therefore, we locate VMs of group 1 and 3 in the physical server 1 and 2, respectively. The placement result for the dynamic VM situation is illustrated in **Fig. 9**.

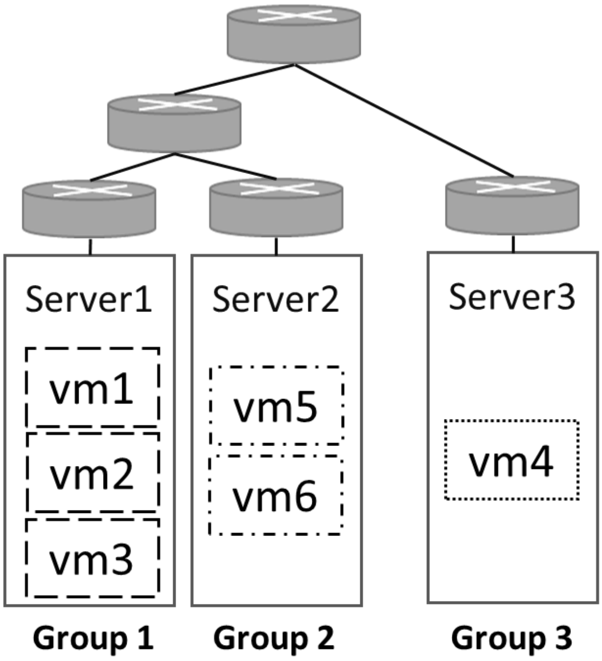

**Fig. 10.** Placement result for the dynamic VM environment

The performance between the pre-configured and dynamic VM scheme are compared in **Table 7**.

**Table 7.** Comparisons of the proposed schemes

| Schemes | Node variance | Transmission cost | Effectiveness |
|---|---|---|---|
| Pre-configured VM | 51 | 828 | 42228 |
| Dynamic VM | 158.33 | 164 | 25966.12 |

The results shown that the dynamic VM scheme achieve much better performance in transmission cost when compared to the pre-configured scheme, however, it increases the variance of node load. Thus the load among nodes is less balance than that of the pre-configured VM scheme. The main reason is that the pre-configured scheme balance the VM load in step 3, however, the dynamic scheme does not consider this issue.

## 5. Conclusions

This paper studies the data placement issue of SNS in cloud environment. The proposed three-step placement scheme deals with this problem in a systematic manner. The proposed scheme applies k-means to group the users/data sources, which are highly correlated, together and allocate them to the server group. And the nodes, which are allocated to the same server group, are evenly assigned to the VM of the service group to achieve load balance. An example is provided to illustrate that the proposed scheme has better performance than the node balance scheme.

This paper also proposes the server group concept to join multiple physical servers within a limited transmission cost d so that the clustering can be more flexible. However, the placement performance, including computation complexity and the delivery performance, may be sensitive to the setting of d. Furthermore, the technology of software defined networking (SDN) may be applicable to balance the load of CDC network of cloud. In addition, the data access behavior and the construction of SNS social graph need to be further characterized through exhaustive observation in real environment. These issues are our on-going research issues.

## References

[1] R. Ackland, "Social network services as data sources and platforms for e-researching social networks," *Social Science Computer Review*, vol. 27, no. 4, pp. 481-492, 2009. Article (CrossRef Link).

[2] Wilson Robert E., Samuel D. Gosling, and Lindsay T. Graham," A review of Facebook research in the social sciences." *Perspectives on Psychological Science 7.3 (2012)*, pp. 203-220. Article (CrossRef Link)

[3] A. Lampinen, V. Lehtinen, A. Lehmuskallio and S. Tamminent, "We're in it together: interpersonal management of disclosure in social network services," in *Proc. of the SIGCHI Conference on Human Factor in Computing Systems*, pp. 3217-3226, 2011. Article (CrossRef Link)

[4] T. Wood, K. K. Ramakrishnan, P. Shenoy and J. Merwe, "CloudNet: dynamic pooling of cloud resources by live WAN migration of virtual machines," in *Proc. of the 7th ACM SIGPLAN/SIGOPS International Conference on Virtual Execution Environment*, 2011, pp. 121-132. Article (CrossRef Link)

[5] Md. Faizul Bari, Raouf Boutaba, Rafael Esteves, Lisandro Zambenedetti Granville, Maxim Podlesny, Md Golam Rabbani, Qi Zhang, Mohamed Faten Zhani "Data Center Network Virtualization: A Survey," *IEEE Commun. Surv. Tut.*, vol. 15, no. 2, pp. 909-928, Sep. 2013. Article (CrossRef Link).

[6] Caton. S, Haas. C, Chard. K, Bubendorfer. K, Rana. O, "A Social Compute Cloud: Allocating and Sharing Infrastructure Resources via Social Networks" *IEEE Transactions on Services Computing*, vol. 7, no. 3, pp. 359-372, 2014. Article (CrossRef Link)

[7] W. Fang, X. Liang, S, Li, L. Chiaraviglio, N. Xiong, "VMPlanner: Optimizing Virtual Machine Placement and Traffic Flow Routing to Reduce Network Power Costs in Cloud Data Centers," *Computer Networks*, vol. 57, pp. 179-196, 2013. Article (CrossRef Link)

[8] Kuan-yin Chen, Yang Xu, Kang Xi, H. Jonathan Chao, "Intelligent Virtual Machine Placement for Cost Efficiency in Geo-Distributed Cloud Systems" *IEEE International Conference on Communications (ICC)*, pp. 3498 – 3503, 2013. Article (CrossRef Link)

[9] Wei Wei, Xuanzhong Wei, Tao Chen, Xiaofeng Gao, Guihai Chen,"Dynamic Correlative VM Placement for Quality-Assured Cloud Service" in *Proc. of IEEE International Conference on Communications (ICC)*, pp. 2573 – 2577, 2013. Article (CrossRef Link)

[10] Y. He, R. Lee, Y. Huai, Z. Shao, N. Jain, X. Zhang, Z. Xu, "RCFile: A fast and space-efficient data

placement structure in MapReduce-based warehouse systems," in *Proc. of IEEE 27th International Conference on Data Engineering (ICDE 2011)*, pp 1199 - 1208, 2010. Article (CrossRef Link)

[11] Jiong Xie, Shu Yin, Xiaojun Ruan, Zhiyang Ding, Yun Tian, James Majors, Adam Manzanares, and Xiao Qin, "Improving MapReduce performance through data placement in heterogeneous Hadoop clusters," in *Proc. of IEEE International Symposium on Parallel & Distributed Processing, Workshops and Phd Forum (IPDPSW)*, pp. 1-9, 2010. Article (CrossRef Link)

[12] A. Gupta, D. Milojicic, V. Laxmikant, "Optimizing VM Placement for HPC in the Cloud," in *Proc. of the 2012 workshop on Cloud services, federation, and the 8th open cirrus summit*, pp. 1-6, 2012. Article (CrossRef Link)

[13] R. Mark, A. Mateo and Jaewan Lee, "Dynamic Service Assignment based on Proportional Ordering for the Adaptive Resource Management of Cloud Systems," *KSII Trans. on Internet and Inofrmation Systems*, vol. 5, no. 12, pp. 2294-2314, Dec. 2011. Article (CrossRef Link)

[14] Lei Jiao, Jun Li, Wei Du, Xiaoming Fu," Multi-Objective Data Placement for Multi-Cloud Socially Aware Services," *2014 Proceedings IEEE INFOCOM*, pp. 28-36, May 2014. Article (CrossRef Link)

[15] P. Pantazopoulos, I. Stavrakakis, A. Passarella and M. Conti, "Efficient Social-aware Content Placement in Opportunistic Networks," *IEEE/IFIP Wireless On-demand Network Systems and Services (WONS)*, pp. 17-24, 2010. Article (CrossRef Link)

**Yen-Wen Chen** received the Ph.D. degree in Electronic Engineering from National Taiwan University of Science and Technology (NTUST) in 1997. During 1983 to 1998, he worked at Chunghua Telecommunication Laboratories, Taiwan and was a project manager of the broadband switching systems. From August 1998 to July 2000, he joined the Department of Information Management, Central Police University. Since August 2000, Dr. Chen has joined the Department of Communication Engineering of National Central University. Currently, he is a professor. His research interests include broadband mobile networks, QoS management, sensor networks, network applications, and multimedia networks. Dr. Chen is a member of the IEEE communication society.

**Meng-Hsien Lin** received the M.S. degree in electrical engineering from National Chung Cheng University, Taiwan, in 2000. He is working toward the Ph.D. degree in ommunication engineering at the National Central University, Taiwan. His research interests include broadband mobile networks, social network services and cloud computing architecture. He is currently with Chunghwa Telecommunication Laboratories, Taiwan

**Min-Yan,Wu** graduate from Department of Communications Engineering National Chung Cheng University in 2013. And, currently, she is studying at the Department of Communication Engineering in National Central University to purchase her Master degree. Miss Wu's research interests include cloud network and applications, network services design, etc.