

Human-centric Computing and Information Sciences

February 2021 | Volume 11



www.hcisjournal.com



Human-centric Computing and Information Sciences (2021) 11:9

DOI: <https://doi.org/10.22967/HCIS.2021.11.009>

Manuscript received July 9, 2020; accepted January 10, 2021; published February 26, 2021.

Performance Evaluation of the vSAN Application: A Case Study on the 3D and AI Virtual Application Cloud Service

Chen-Kun Tsung¹, Chao-Tung Yang^{2,*}, Rajiv Ranjan³, Yong-Lun Chen⁴, and Jean-Huei Ou⁴

Abstract

Although the popularity of server virtualization technology has facilitated the development of numerous applications, the storage input/output performance and the high availability of the backend are the key factors that affect service quality, regardless of the type of application service. Following the proposal of a software-defined technology, solutions to the concepts of software-defined data centers and software-defined storage have also been proposed. A virtual storage area network (vSAN) has a relatively scalable architecture, supports solid-state drives, and provides an expansion mechanism in the form of scale-up and scale-out storage. We conducted an in-depth study in which we measured the performance of the 3D virtual application cloud service (3D-VACS) on a university campus. The 3D-VACS constitutes the distributed file system (DFS), heterogeneous network, and high-performance environments for a vSAN. A greater number of disk stripes was discovered to improve the performance of big block transfers, i.e., when 10 disk stripes were employed, the performance was 2.46 times higher than when 1 disk stripe was used. The highest performance improvement, 32%, was obtained in a test requiring simultaneous reading and writing in a heterogeneous network environment, providing a faster network; when the write-only requirement was enforced, the performance did not improve significantly, although it was greater than that observed for the read/write hybrid requirement. Scale-out improved operational performance by 69% in an environment that required large amounts of reading and writing. In multimedia server data storage or applications in a file server storage environment, scale-up was observed to improve performance by up to 48%.

Keywords

Link Aggregation, Network-Attached Storage, Virtual Storage Area Network

1. Introduction

From the early mainframe and distributed computing to the recent cloud computing, computing platforms have been able to provide increasingly diverse services due to the accumulation of hardware and software resources. The evolution of network transmission capability and storage space has enabled continuous improvements to be made to the performance of cloud computing, giving users a better experience [1, 2]. Regarding the network infrastructure, the breakthrough of full-duplex data transmission

* This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

Corresponding Author: Chao-Tung Yang (ctyang@thu.edu.tw)

¹Department of Computer Science and Information Engineering, National Chin-Yi University of Technology, Taiwan

²Department of Computer Science, Research Center for Nanotechnology, Research Center for Smart Sustainable Circular Economy, Tunghai University, Taiwan

³Computing Science and Internet of Things, Newcastle University, United Kingdom

⁴Computer Center, Tunghai University, Taiwan

has increased convenience for end users. To provide a better experience for multiple users, aggregating data [3, 4] or network ability are common solutions [5, 6]. The recent 10G even uses the 40 G super-high-speed link aggregation network to enable cloud servers to provide a faster rate of data transmission [5–7]. Additionally, since magnetic disks replaced magnetic tapes in storage devices, the performance of data reading has also improved continually. Nowadays, an increasing number of key applications and backup-oriented storage include solid-state drives (SSDs) as one of the core components of high-performance storage devices in response to the high-performance requirements.

In the traditional input/output (I/O) architecture, servers provide computing capability whereas storage provides storage capability, and the two are connected via the high-speed network. This approach is easy to manage and offers great flexibility when resources are increased. However, as the computing architecture of data centers is generally advancing toward x86 platforms, the differences between hardware configurations supported by servers and storages are becoming increasingly small; the installation of storage system software on the server is all required to enable centralized computing and storage capabilities. The first alternative to the traditional I/O architecture was the hyper-converged infrastructure (HCI) [8]. Storage virtualization, computation virtualization, and virtualization management are the three main elements of the HCI [9, 10]. Through virtualization and management mechanisms, storage and computation are aggregated into a single service. Therefore, software-defined technology (SDT) makes server operations highly flexible [11, 12]. Because of the cooperation between HCI and SDT, software-defined everything, virtualization, and clustering have blurred the boundaries between servers and storage devices [13–15].

The user's experience in reading large amounts of data has continued to improve, along with improvements in hardware performance. Take, for example, the virtual desktop [16], which is a teaching or single operating environment; during class, students read from the virtual desktop infrastructure (VDI) server [17, 18] and operate the remote course virtual machine (VM) [19]. The advantage of using the VDI is that it enables service managers to standardize the runtime environment of the user and minimize the problems caused by the runtime environment; the challenge here is how to enable a smooth operating experience for the user [20].

In addition to hardware improvement, management software has evolved, becoming more convenient for cloud service managers. Take, for example, the highly flexible virtual storage area network (vSAN) architecture [21]. vSAN 1.0 supports scale-up expansion of storage space, whereas vSAN 6.0 additionally includes a scale-out mechanism, meeting the requirement for increased VM storage by providing another option, which in turn lowers management complexity and the overall cost of ownership.

This study targeted the 3D virtual application cloud service (3D-VACS) in order to evaluate the performance of the vSAN architecture. The 3D-VACS is a distributed file system (DFS) with heterogeneous networks and a high-performance environment. The 3DVACS allows easy by multiple users simultaneously over the Internet. The users utilize the computation resources from an authorized VM after logging into the 3D-VACS, which provides high performance for large-scale computation because some graphics processing units are equipped to offer a virtual graphics processing unit service. Thus, each user is able to perform some scientific computations and artificial intelligence processes, such as image recognition by the convolutional neural network, model construction by machine learning, event prediction by the long short-term memory network, and 3D model design by AUTODESK AutoCAD.

The main contribution comes from the performance evaluation of the 3D-VACS based on providing a high user experience. The following is an explanation of the results:

(1) **Multinode system:** A distributed system comprises a number of nodes and improves access performance through a distributed data source; it is also the most common architecture at present. We discovered that in a multinode system, the number of disk stripes in the vSAN result positively affects the performance for big block, transfers but does not affect the performance for small block transfers. We designed 60 performance tests for deploying VDI; the vSAN provides less deployment time than traditional network-attached storage (NAS) mechanism [22]. Overall, the greater the number of vSAN nodes, the shorter the deployment time is.

- (2) **Hybrid networking system:** How to use a heterogeneous network effectively is one of the problems faced by network administrators. We examined the data read/write performance, latency, and throughput in the mainstream 10 G network and the super-high-speed 40 G network environments. Overall, a faster network results in superior system operational performance, but in the write test, the improvement in performance is not as high as in the read/write and hybrid operations because the hard disk used for cache and storage differs in the vSAN.
- (3) **Fully SSD system:** SSDs are used in high-performance storage. In addition to having a short response time, SSDs have the major advantage of a fast random-access time. We tested the scale-up and scale-out storage expansion mechanisms in the high-end system environment and discovered that in an environment that is complex, and in which the VM requires numerous read/write operations, scale-out yields greater benefits through high parallelization. In multimedia server data access or applications in a file server access environment, scale-up yields a better performance because of the larger cache space.

2. Related Research

This section introduces the common system architectures as follows:

- (1) **DFS:** The DFS allows files to be stored across multiple hosts on a network [23], and the file access is subject to abstraction and becomes a unified interface, giving users the impression that DFS is like any normal file system. The DFS has high performance, fault tolerance, reliability, availability, and scalability [24, 25]. In addition to dispersing data among different data nodes, each data node has distributed computing capability; as such, program logic and data processing, which in non-DFSs are performed on a single host, can be allocated to several computing nodes. Once the computing nodes have completed their assigned task, the outcomes are aggregated to form the final result. The distributed computing in the DFS improves the processing performance by reducing the queue time for data access and computation. Thus, Yahoo provides the Hadoop DFS to improve web search performance [26, 27].
- (2) **Software-defined service:** The DFS enables the flexible configuration and expansion of storage space, but its operation appears to be limited in other requirements such as computing and network traffic control [28]. In recent years, increases in the amount of data generated have resulted in higher demand for storage space, and the storage capacity of server configurations is becoming increasingly large [29]. However, to ensure the service quality of the system, each storage device can only be accessed by a specified application. If a company uses storage devices of different brands, service integration is a problem; therefore, implementing centralized control of storage resources is extremely difficult [30]. Storage virtualization is a solution to heterogeneous storage devices [12]. Virtualization can be used to centrally manage all storage devices, enabling user storage access that resembles that in the DFS. To strengthen the scalability of heterogeneous storage, the introduction of software-defined storage (SDS) built using the SDT not only retains the abstract interface of DFS hardware resources but also achieves the flexible, scalable architecture of scale-up and scale-out expansions [31]. Conversely, when building a DFS data center, the information personnel must establish the scope of demand, select the hardware, plan the middleware, and design a management mechanism. A goal-oriented design improves the performance of a data center, but the expansion of different applications is limited. SDT, as a design concept for another type of data center, has more flexible expansion than the DFS. The core technology of the SDT is virtualization; all hardware resources are aggregated into a resource pool, and the operation of hardware resources is controlled using software [32].
- (3) **vSAN:** The traditional network storage is divided into storage area networks and NAS; both manage the physical storage by a single storage hardware device through a network. The vSAN was formed after integrating the SDT; once the physical storage component distributed to different hosts is virtualized, the virtualized environment is directly used on a common storage resource. A vSAN is a

cross-host storage space that also uses a hybrid hard drive system in its native machine. Native peripheral component interconnect express (PCIe) Flash or an SSD is employed as the cache for data reading and writing, and data are stored on the traditional mechanical hard drive of the native machine. Therefore, a vSAN improves the I/O performance of a VM's data access in a production environment. For example, the VDI provides a better user experience after considering the vSAN structure [33]. The vSAN datastore is the collection of local storage resources of the virtual host group, and the overall performance is dependent on the flash memory devices of the virtual host group; the size of the storage resources depends on the space on the mechanical hard drive. Thus, when a new virtual host is added to the group, the overall data read/write performance and storage space increase, that is, they scale-out. Scale-up is achieved by directly adding hardware resources to the original host. In the operation framework of a vSAN, at least three ESXi member hosts are required to provide vSAN storage resources, and these three hosts must satisfy the minimum requirement of a vSAN storage framework: one PCIe flash or SSD, and one SAS or SATA hard disk drive (HDD). Once the vSAN storage space has been established, other virtual hosts can be used directly, even if they do not provide storage resources.

A vSAN with SDT provides complete data protection and has high scalability; it is the mainstream operating method at present. However, the configuration method under different requirements and environments still affects the system's performance.

Moreover, automatically allocating and dispatching computation resources is helpful in increasing the user experience. For example, when a task consumes most resources, the cloud infrastructure should dynamically adjust the computing power [34–37]. The automatically increasing the computing power of the VMs is the advanced properties of the cloud infrastructure for the perspective of end users. This paper aims to evaluate the efficiency of the vSAN architecture under different requirements and environments. Although auto-scaling provides a better user experience, we are more interested in the network design. Therefore, evaluating the cloud performance according to different scenarios is the major goal of this paper.

3. System Design

This study aims to evaluate the 3D-VACS platform, whose system architecture is illustrated in Fig. 1(a). As the goal of 3D-VACS is to provide accessibility for 180 users, it is equipped with a total of 72 CPU cores, 1 TB of memory, 25 TB of storage space, three NVIDIA GRID K2 and two NVIDIA GRID K1. To maximize the utilization of each server, the 3D-VACS applies the vSAN network environment and the VMware vSphere to construct the cloud infrastructure. Moreover, the VMware Horizon and a portal are deployed to manage the VDI services and assist users' operations, respectively.

As shown in Fig. 1(b), the user visits the 3D-VACS portal in the beginning. The 3D-VACS reserves and allocates hardware resources, including a specific CPU, RAM, HDD space, virtual network interface card, etc., after the user logs into the system. Then, the 3D-VACS creates the VM and responses the system screen to the user, enabling the user to access the VM. When the user logs out or is idle for a specific amount of time, say 30 minutes, the 3D-VACS will deconstruct the VM and then retrieve the allocated resource.

In the experimental environment prepared for this study, VMware vSphere 6.0 with Horizon View 6 was employed to implement vSAN and virtual desktop scenarios, and the system's performance was evaluated according to the following common scenarios: multinode, hybrid networking, and fully SSD systems. Multinode systems and hybrid networking systems are commonly used system architectures. Earlier systems caused a heterogeneous server and network operating environment after several hardware upgrades. A fully SSD system is a recently designed architecture that enables high application performance; an SSD is employed to improve data access performance. The implementation of each system for the three common architectures is explained.

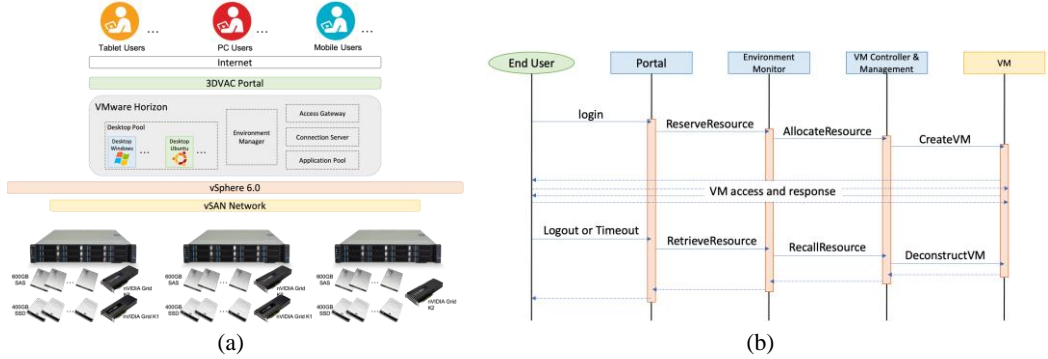


Fig. 1. The system diagram of 3D-VACS: (a) the system architecture and (b) the sequence diagrams.

(1) **Multinode system:** To implement an environment in which multiple computing nodes work together, we used 10 Inventec Zion Servers in the environment test. Because the Inventec Zion Server can support only one hard drive, we used a Y-cable to enable the installation of two physical hard drives on one server in order to meet the implementation requirements of a vSAN. The specifications of the single server are displayed in Table 1. In the multinode system, the objective of the test was to determine the performance of server local data access and the data transmission efficiency of the vSAN. Each Inventec Zion Server provided four 1 GbE network interfaces. We used three network interfaces to implement 802.3.ad with the Link Aggregation Control Protocol to improve the data transmission efficiency of the vSAN. Additionally, the vSAN is a product that meets SDS requirements, as it can implement fault tolerance and splitting and retain the SSD cache ratio of the front-end cache tier through SDT. So, a manager’s ability to improve the vSAN’s transmission efficiency is one of the crucial benefits of the multinode system.

Table 1. Device specifications for the multinode experimental environment

Item	Specification	Count
CPU	Intel E5540	2
RAM	4 GB DDR3 1066 ECC	12
SSD	Intel 730 Series 240 G	1
HDD	WD 2 TB 7.2K	1
Network interface	1 Gb	4

(2) **Hybrid networking system:** To implement a heterogeneous network environment, we used three HP ProLiant DL380 Gen9 servers as the computing nodes. Each computing node employed two disk groups, and each disk group used six SAS hard drives with an Intel 730 SSD to improve the hard drive access performance. In the vSAN test, ensuring that the hard drive access performance of a single node was higher than the network transmission efficiency is necessary; otherwise, measuring the actual performance of the vSAN would have been impossible. A vSAN system requires the use of a better network environment than 10 GbE, so in addition to using the preloaded HP 1 GbE networking interface, we added the HP 10 Gb and Mellanox 40 GbE networking interfaces. The hardware architecture is detailed in Table 2.

(3) **Fully SSD system:** To improve the flexibility of system adjustment, personal computers (PCs) were used as single computing nodes in the fully SSD system, and SATA hard drives and SSDs were employed to build the storage space. The Mellanox 40GbE networking interface was used to form the vSAN environment and to ensure that the experimental results truly reflected the performance of the vSAN. We concatenated four host servers to test the performance growth of scale-up and scale-

out, determining the optimal settings for different usage scenarios. The specifications of the single node system are displayed in Table 3. Four data nodes were deployed in this experiment. This study sought to understand whether the performance of the vSAN storage environment would be strongly or only slightly positively affected if all the hard drives were replaced with SSD flash devices. Additionally, we investigated how the application time for scale-up expansion and scale-out expansion should be selected regarding the expansion of hardware devices in the vSAN storage environment.

Table 2. Device specifications for the heterogeneous network experimental environment

Item	Specification	Count
CPU	Intel E5-2640	2
RAM	16 GB DDR4 2133 ECC	8
SSD	Intel 730 Series 480 G	2
HDD	HP 600 G 10K SAS	12
Network interface 1	HP 1 Gb 331i	4
Network interface 2	HP 10 Gb 560 SFP+	2
Network interface 3	Mellanox MT27520 ConnectX-3	1

Table 3. Device specifications for all-flash storage experimental environments

Item	Specification	Count
CPU	Intel i7-5960	1
RAM	16 GB DDR4 2133	4
SSD	Intel 535 Series 480G	2
HDD	WD 2 TB 7.2K	2
Network interface 1	Intel 82574L 1 Gb	1
Network interface 2	Mellanox MT27520 ConnectX-3	1

We consider the general platform to build the 3D-VACS. The performance of the 3D-VACS could be maximized, but however we do not consider the better platform because of the implementation consideration. As mentioned above, the 3D-VACS is a real-world application, and the student can access the predefined VM over the Internet. Constructing a platform which balances performance and cost is the major purpose of this paper. Therefore, the 3D-VACS provides a good user experience for scholars and students even if there are some hardware and software solutions to maximize the performance of the 3D-VACS, such as using the non-volatile memory host controller interface specification drives instead of SSD or the in-memory caching mechanism.

4. Experiment and Analysis

The principle behind the operation of a vSAN is to concatenate the storage space of different servers and view them as a single resource pool. Stored data are mutually backed up on the hard drives within the disk group. When one of the hard drives fails, the backup drive takes over the work in progress. Theoretically, the operational performance of a vSAN is positively correlated with the number of nodes. Therefore, we investigated the data transmission efficiency of the three common computing frameworks, multinode, hybrid networking, and fully SSD systems, and provided the system settings for optimized transmission efficiency in various scenarios.

4.1 Multinode System

4.1.1 Experimental environment

We first measured the data access performance of the storage space as the baseline for comparison. To the multinode system, we added a server with the same specifications as the baseline considered, but did not add this server to the vSAN storage. Instead, we created a new vSAN storage as the test baseline. Then, an Iometer was used to measure the baseline data access performance. We employed 16 outstanding I/Os and increased the accuracy of the settings in a multiplexed manner. We used a pair to represent the latency and the average number of I/O operations per second (IOPS). The results obtained after running the experiment five times were (1.09, 14693), (4.19, 3818), and (31.87, 501) for 4 kB, 64 kB, and 256 kB full read data, respectively. In the multinode system, we analyzed the effects of network speed and the number of disk stripes on the transmission efficiency of the vSAN. We also deployed numerous VMs to capture the operational performance of real cases.

4.1.2 Effect of the network speed on performance

First, we used an Iometer to obtain the experimental results by using the default configuration of the vSAN; the experimental results and test baseline results are presented in Fig. 2, where the horizontal and vertical axes represent the number of network interface cards (NICs) used and the IOPS in comparison with the test baseline, respectively. A comparison of the results with the test baseline does not reveal significant differences; the performance and even the data size in the 64 kB and 256 kB results were worse than in the baseline test.

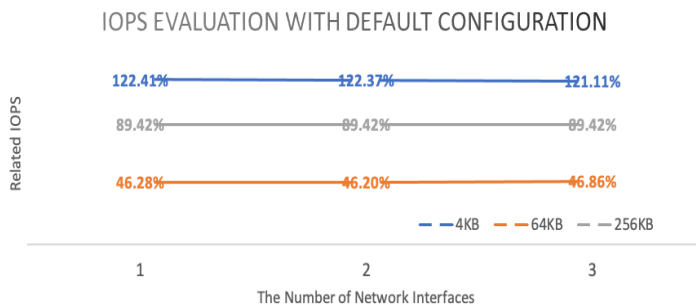


Fig. 2. Experimental results for default configuration.

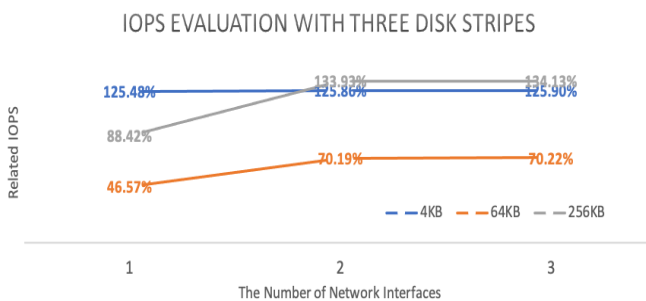


Fig. 3. Experimental results for three disk stripes.

It was surmised that the reading and writing occurred on the same data node. The data access performance of a single node was lower than the 1 GbE network speed, which caused the performance to be lower than in the baseline test. Therefore, in the second experiment, we adjusted the number of disk stripes and stored the data on three nodes to improve the distributed processing. The test results are shown in Fig. 3, where the horizontal and vertical axes represent the number of NICs used and the IOPS in comparison with the test baseline, respectively. In the 4 kB data test, the difference in performance was

slight, whereas the 64 kB and 256 kB experimental results indicated a significant improvement. The performance of the aggregation of two NICs was 70.19% in the 64 kB data test, compared with the previous value of 46.57%; the total performance was thus increased by 50.73%. In the 256 kB data test, the performance was 133.93% up from 88.42%, an increase of 51.47%. By contrast, the performance was not improved considerably when the three NICs were aggregated; the 64 kB data and the 256 kB data increased only slightly, by 0.04% and 0.15%, respectively.

4.1.3 Effect of the number of disk stripes on performance

Increasing the number of disk stripes can improve the transmission efficiency of a vSAN. Therefore, this study investigated the effect of the number of disk stripes on transmission efficiency. We discovered that the improvement was greater than the results illustrated in Fig. 3 the 4 kB and 256 kB data. Therefore, we evaluated the effect of the number of disk stripes on the vSAN transmission efficiency through 1 to 10 settings in the 4 kB and 256 kB data environments. The experimental results are illustrated in Fig. 4. The horizontal axis represents the number of disk stripes, whereas the vertical axis represents the IPOS in comparison with the test baseline.

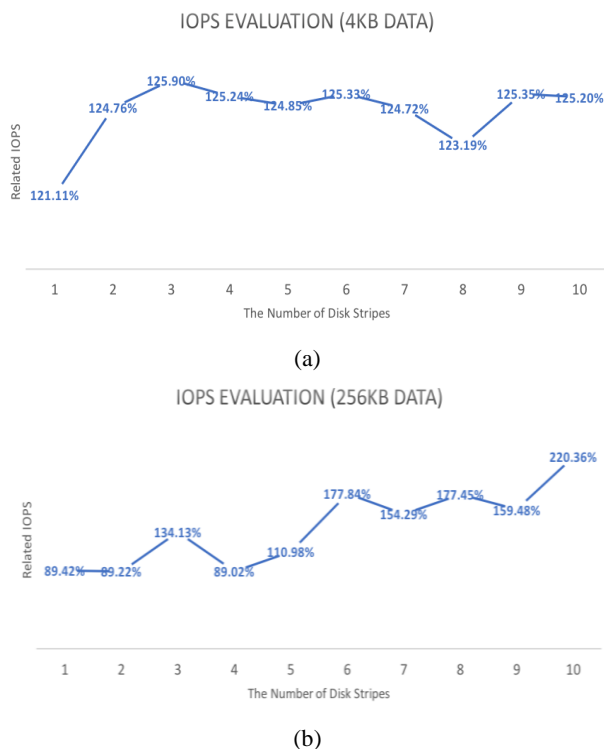


Fig. 4. Results of various data-size experiments: (a) 4 kB data and (b) 256 kB data.

In the 4 kB data experiment, the transmission efficiency was increased by approximately 21%–26%. A system bottleneck was reached on increasing the number of disk stripes to more than two; the performance increased only slightly, an increase of approximately 3%. Therefore, under the transmitting small files, increasing the number of disk stripes does not considerably improve the vSAN transmission efficiency.

In the 256 kB data experiment, we concluded from the results (presented in Fig. 4(b)) that the transmission efficiency of the vSAN was approximately proportional to the number of disk stripes; however, there are disparities in some of the experimental results and inferences. The data actually read through user action can be stored in the SSD or HDD, and the identity of this storage determines the data

transmission efficiency. Overall, however, increasing the number of disk stripes improved the transmission efficiency of the vSAN. The transmission efficiency for 10 disk stripes was 2.46 times that obtained for 1 disk stripe.

4.1.4 Efficiency of VM deployment

Next, we simulated a scenario in which numerous users are online simultaneously, such as in online teaching, when several VMs are initiated at the beginning of a class. This was done to evaluate the difference in the operational performance of the multinode system in NAS and vSAN environments. We used Horizon View 6.1 to generate 60 virtual desktop environments at a time, and ran the experiment under 3, 6, and 10 disk stripes, which, according to Fig. 4(b), yielded the highest performance. The NAS used RAID 10 and was connected using iSCSI. The experimental results are presented in Fig. 5, where the horizontal and vertical axes represent the number of disk stripes and the deployment time in comparison with NAS, respectively.

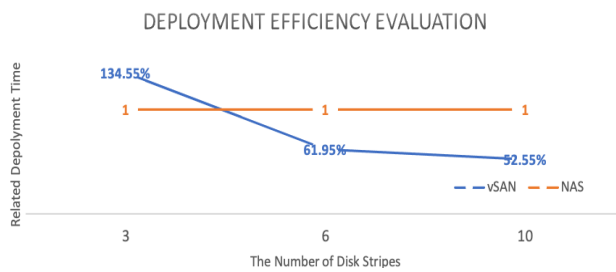


Fig. 5. Deployment time in NAS and vSAN for deploying 60 VMs.

The deployment time of the vSAN with 10 disk stripes was only 52.55% that of NAS; the vSAN with 6 disk stripes had a slightly higher deployment time of 61.95% than that of NAS; finally, the vSAN with three disk stripes required 34.55% more time to deploy than does NAS.

When the number of disk stripes was increased from three to six, the deployment time varied greatly. Therefore, we analyzed the time required for various actions in the deployment process in order to understand the direction in which the system could be optimized. The VM deployment process involved four steps: copy, setting, deployment, and configure. We obtained the time required for each of these four steps, as shown in Fig. 6, where the horizontal and vertical axes represent the actions of the VM deployment process and the required time in comparison with NAS, respectively

The time required for copy and setting did not vary greatly from that in the vSAN; while the runtime for deployment and configure were the most critical. In deployment, the vSAN required only 38.94%–54.78% of NAS time to complete the action, whereas the vSAN with three disk stripes required 73.85% more time than NAS to complete the configuration of the virtual desktop action. Because configuring the virtual desktop consumed a large amount of the SSD cache, when the request exceeded the SSD cache’s capacity, the request was transferred to the HDD, resulting in a long reading time. This phenomenon also occurred in the copy and setting steps, but because the request did not exceed the capacity of the SSD cache to a significant extent, the vSAN did not take a particularly long time. Therefore, increasing the size of the SSD cache can reduce the deployment time, but this is not an effective solution because it is difficult to estimate requests; rather, it is more efficient to increase the number of disk stripes.

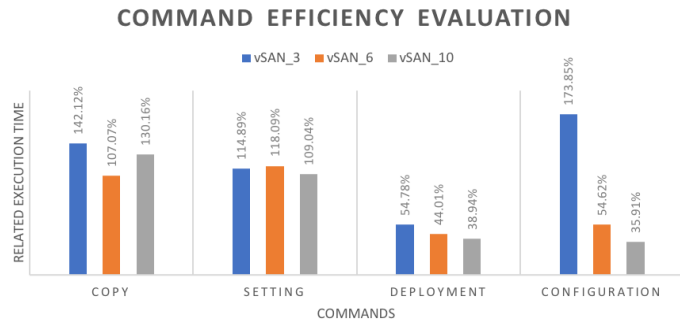


Fig. 6. Time required for each action in the deployment process.

4.2 Hybrid Networking System

4.2.1 Experimental environment

The data transmission efficiency of a distribution system depends on the system's networking performance. In the multinode system experiment, we discovered that aggregating multiple 1 GbE NICs improved the data transmission efficiency. However, not all systems use homogeneous networks; a hardware upgrade or the acquisition of new hardware may create a heterogeneous network environment. Focusing on this phenomenon, we employed 10 GbE and the Mellanox 40 G high-speed optical fiber NIC to evaluate the data transmission efficiency in various scenarios. The Mellanox 40 G NIC used the quad small form factor pluggable interface; and a fast transmission speed of 40 Gbps was obtained by aggregating four 10 Gbps data transmission channels.

Once the infrastructure for the high-speed network has been established, the system configuration affects the data transmission efficiency. To ensure that the network environment could provide 10 Gbps and 40 Gbps transmission capacity, we tested the 10 G and 40 G network environments under system settings of 1, 6, 12, and 24 nodes. The average results obtained using IPerf for the five measurements are displayed in Table 4.

Table 4. Evaluation of transmission efficiency (unit, %)

Number of nodes	Intel 10 G	Mellanox 40 G
1	5.680	7.328
6	9.130	14.180
12	9.122	14.440
24	9.044	14.040

The performance of 10 GpE when the number of nodes was six, was close to the theoretical efficiency, whereas the performance of Mellanox 40 G fell below expectations, with only approximately 14 Gbps. Therefore, we adjusted the data distribution principal settings of ESXi; the test results for the higher dispersion are presented in the ESXi Optimization column shown in Table 5. As the performance of a single node was poor, we only obtained results for 6, 12, and 24 nodes. Although the performance was increased from the original 14 to 32 Gbps after adjusting the ESXi parameter, there was still a disparity of approximately 20% with the theoretical value. Therefore, we opened multithreading in the guest operating system and obtained a transmission efficiency of approximately 36 Gbps, which was closer to the theoretical value of 40 Gbps.

Table 5. Evaluation of the transmission efficiency of Mellanox 40 G (unit, %)

Number of nodes	ESXi Optimization	ESXi Optimization + Multithreading
6	33.5	35.7
12	32.3	37.6
24	32.5	36.3

Moreover, we used the following five test cases to conduct further experiments:

- **Basic sanity test (BST):** A 1-GB test file was generated of each test host, and 70% read and 30% write actions were executed simultaneously. This is the most common test and represents the average proportions of operations in practice.
- **Stress test (ST):** A 1 TB test file was generated of each test host, and multiple read and write actions were executed simultaneously.
- **Fully read test (RT):** A 10 GB test file was generated of each test host, and only read actions were executed on the test file. This test focused on using the front-end cache, and was used to simulate the requirements of a teaching environment.
- **Fully write test (WT):** A 5 GB test file was written to the storage device of each test host. This test method focused on the write cache, and was used to simulate the data writing requirements of the data center.
- **Hybrid test (HT):** Each test host used 64k block size to execute 70% read and 30% write actions with the target of generating a 30 GB test file.

4.2.2 Results of the stress test

It is easy to generate a read and write test under the most common severe conditions that simulate users' behavior. In a multiuser environment, read and write operations are not performed in specific proportions, which is the major consideration of the stress test. So that it is difficult to design the system with high performance for the stress test. The stress system test and experiment design are thus relatively difficult. The experimental results are presented in Table 6. Increasing the network bandwidth directly increased latency and IOPS performance. However, the improvements were not as large as the increase in bandwidth; and the greatest improvement was only 40.00% of the average latency.

Table 6. Results of the stress test on Intel 10 G and Mellanox 40 G

	Intel 10 G	Mellanox 40 G	Improvement (%)
IOPS	4330	5703	31.71
Maximum latency (ms)	14.1	11.9	18.49
Average latency (ms)	10.5	7.5	40.00
Throughput (MB/s)	34.0	45.0	32.35

4.2.3 IOPS test results

Next, we completed all of the scenario tests. The results of the IOPS test are presented in Fig. 7, which plots the IOPS versus the test scenario. In the BST, which was the closest test to the actual conditions, the 10 G network was increased to a 40 G network, and the performance of the vSAN was increased by approximately 17%. Under the extreme conditions of the HT and RT, the systems consider read cache and write cache; therefore, the improvement in the IOPS performance of Mellanox 40 G was only 10% higher than the performance of Intel 10G. In the big block data read and write test, 64k block size was employed for the HT, and 70% read and 30% write actions were simultaneously performed. In addition, a 30 GB test file was generated on each test host in order to evaluate the IOPS performance. The results

of the test revealed that the performance of the vSAN was improved by 13%. The highest improvement of performance (32% higher IOPS) was obtained in the ST, which involved the simultaneous reading and writing of a large volume of data combined with multiplexed executions.

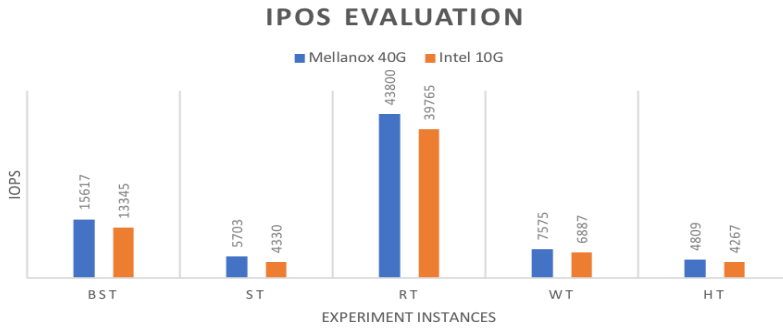


Fig. 7. Results of the IOPS test of Intel 10 G and Mellanox 40 G.

4.2.4 Throughput evaluation

Throughput is the data transmission performance of a metrics networking system; in context, the more data the system can process, the better the user’s experience. The results of the experimental throughput are presented in Fig. 8, where the horizontal and vertical axes represent the test scenario and measured throughput, respectively, in units of MB/s. In the transmission of large blocks (64 kB), as shown in the HT, the highest throughput achieved with the 10 G network interface was 267 MB, while that achieved with Mellanox 40 G was 301 MB; thus, the improvement was approximately 12.73%. However, in the BST, ST, and WT, which were more focused on the writing requirements, the throughput performance was relatively poor, which may be attributable to the cache settings of the vSAN, i.e., 70% of the SSD was set for reading and only 30% for writing.

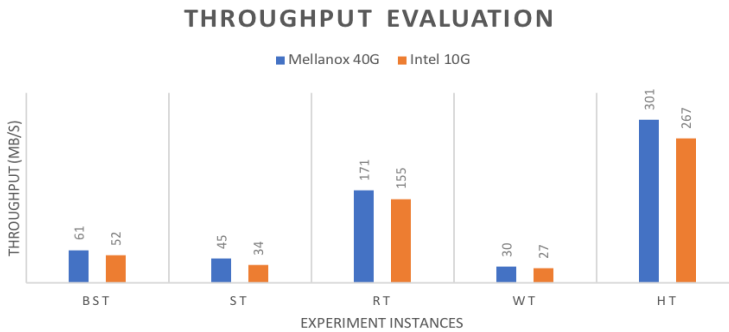


Fig. 8. Results of the throughput test of Intel 10 G and Mellanox 40 G.

4.2.5 Latency evaluation

Latency indicates how users feel about the response of a system. Thus, as lower latency indicates that users feel the system feedback to be relatively instantaneous, minimum latency is the objective. The experimental results are presented in Fig. 9, which plots the measured latency (unit, ms) versus the test scenario. Overall, the latency of the vSAN was relatively low, being only 4 ms at its highest. However, in the ST, the performance of Intel 10 G and Mellanox 40 G was 10.504 and 7.53 ms, respectively, showing a disparity of approximately 39.5%. Because the read and write requirements of the ST mostly

exceeded the design of the system target, the performance of the vSAN was approximately 10 ms. Although a latency of 10 ms in distributed systems is not comparable to 4 ms, it is considered acceptable by users.

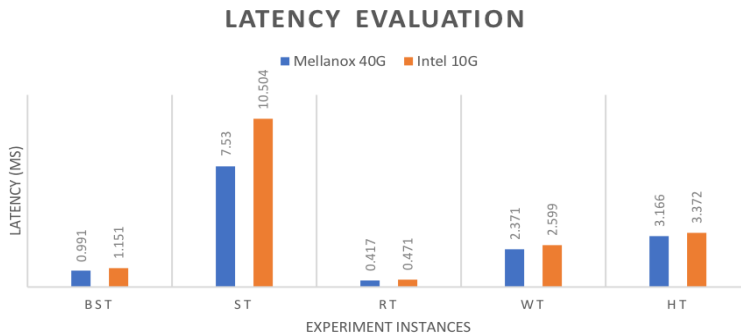


Fig. 9. Latency test results for Intel 10 G and Mellanox 40 G.

4.3 Fully SSD System

4.3.1 Experimental environment

In the experiment presented in Section 4.1, it was discovered that an SSD is responsible for the vSAN cache and that reading uses 70% of the overall space, while writing uses the remaining 30%. Therefore, we have concluded that applying the SSD only in the vSAN cache is a factor that affects the write performance. In this experiment, we upgraded the actual storage data device from the original HDD to an SSD, and expected to discover improvements in the write performance and response speed. Because the vSAN was a fully SSD system, we only evaluated the performance of Mellanox 40 G. Furthermore, as the vSAN supported mechanisms for expanding scale-up and scale-out storage, we added this experiment.

The following four tests were conducted:

- **Small block hybrid test (SHT):** 4k blocks were used to execute 70% read and 30% write actions on each test host in order to generate a 30 GB test file.
- **All-flash test (AFT):** Each test host generated a 1 TB test file on the SSD.
- **Streaming read test (SRT):** A 1 TB test file was generated on each test host to simulate a continuous reading streaming service, such as a media server.
- **Streaming write test (SWT):** A 1 TB test file was generated on each test host to simulate a continuous-writing streaming service, such as a backup server

4.3.2 IOPS evaluation

We used a PC to build a four-node test environment that included one disk group with three hosts. Each node used one SSD in the cache tier and the capacity tier. We employed a similar PC for the control experiment, but used a 7200 rpm SATA HDD in the capacity tier of the back end.

As regards the BST, WT, and RT, and the newly added AFT, the IOPS performance is shown in Table 7 (). The IOPS of all-flash was increased by 24.87% in the BST in comparison with the hybrid system. In the RT, an improvement of only 0.92% was observed, whereas in the AFT, which fully used the SSD, an improvement of 7.51% was obtained. For a large amount of reading, because the test data had priority access to the cache tier data, no considerable increase was observed in the SSD situation. In the WT, because the device for writing to the capacity tier was different, the performance improvement of all-flash was approximately 16.96% compared with the hybrid system.

Table 7. Comparison of IOPS between the all-flash and hybrid structures in the BST, WT, RT, and AFT cases

	BST	WT	RT	AFT
Hybrid (MB/s)	14000	6264.75	73788.75	67237.75
All-flash (MB/s)	17482	7327.50	74469.50	72290.50
Improvement (%)	24.87	16.96	0.92	7.51

Table 8. Comparison of IOPS between ALL-Flash and Hybrid in the ST, SHT, HT, SWT, and SRT cases

	ST	SHT	HT	SWT	SRT
Hybrid (MB/s)	10082	1202.75	2491	179	4905.25
All-flash (MB/s)	10522.25	11532.25	5812.75	164	5424
Improvement (%)	4.37	858.82	133.35	-8.38	10.58

The results presented in Table 7 were not obtained with the full utilization of the advantages of the SSDs. Therefore, we again added the ST, SHT, HT, SRT, and SWT items. The experimental results are shown in Table 8. In the SHT and HT, the IOPS performance reached 858.82% and 133.35%, respectively; these two tests best utilized the advantages of the SSDs. The SHT used the smaller 4K block, whereas the HT used the 64K block. Using smaller blocks to obtain higher operational performance, combined with the characteristics of the SSD, enabled the fully SSD system to achieve an IOPS performance of all-flash that considerably exceeded that of the hybrid system. In the SWT, the all-flash architecture was discovered to be inferior to the hybrid architecture by 8.38%. The test file used in the SWT was 1 TB, but the SSD capacity used by the experimental environment was only 480 GB; therefore, the IOPS performance was affected by the continuous exchange of data.

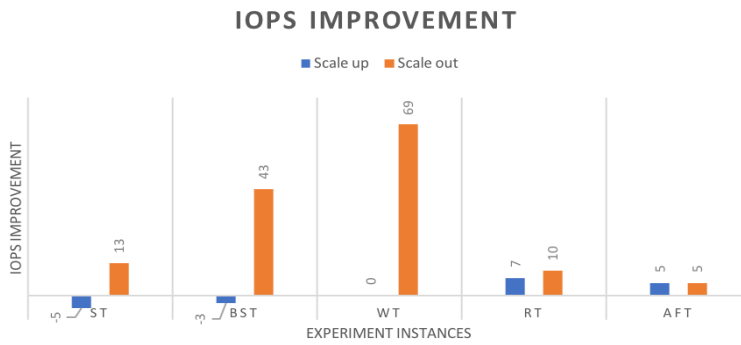
4.3.3 Comparison of the scale-up and scale-out mechanisms

The all-flash architecture could indeed extensively utilize the advantages of SSDs in the SHT and HT. In addition to supporting the all-flash mode, the vSAN 6.0 supported scale-up and scale-out storage expansion. So, we investigated the effectiveness of these two mechanisms and the scenario most suitable for those mechanisms. The baseline of this experiment involved a disk group containing three data nodes. A disk group of four data nodes was used in scale-out, whereas two disk groups were employed in scale-up, with each disk group comprising two data nodes.

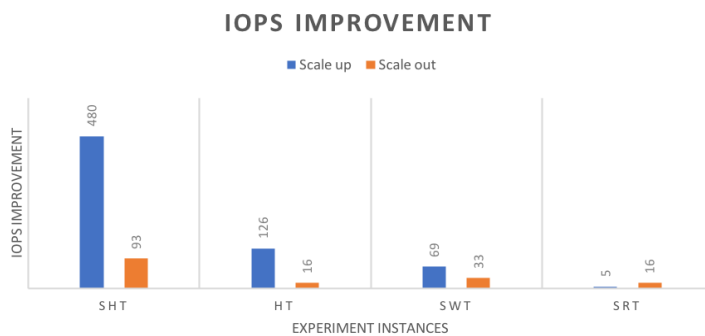
The experimental results obtained under these two conditions and at the baseline are illustrated in Fig. 10(a). In scale-out, direct benefits were obtained, particularly in writing, for which the results were outstanding; while an improvement of 69% was obtained in the WT. So, the effects of scale-up were poor, with performance worsening by 5% and 3% in the ST and BST, respectively.

The remaining results are presented in Fig. 10(b). The scale-up performance improved considerably in all tests (except the SRT) and most particularly for the 4KB block data, for which the performance was excellent (improvement of 480%). The performance of the HT, for which a 64 kB block was used, improved by 126%, thus agreeing with the previous experimental results. The SRT experiment, which focused on continuous reading, showed an improvement of 5%.

In the two experiments, we discovered that the performance of scale-up was dependent on the number of disk groups, because each disk group had a front-end cache tier for acceleration—a characteristic which scale-up could exploit favorably. Therefore, the scale-up performance was superior under the requirement entailing a small block size. Scale-out improved parallelization through the addition of hard drives, thus enabling performance to be improved when the read/write requirements were high.



(a)



(b)

Fig. 10. Result of comparing scale-up with scale-out cases: (a) in the ST, BST, WE, RT, and AFT cases and (b) in the SHT, HT, SWT, and SRT cases.

5. Conclusion

Once virtualization technology became popular, the establishment of cloud services became more flexible, and the utilization of hardware resources also increased. Network usage, system performance, and methods of expanding storage space must be taken into consideration by cloud services during system design. This study conducted a series of experiments focused on the aforementioned three variables, and obtained the following comprehensive results.

- In the deployment of virtual desktops, the execution speed of a vSAN with 10 data nodes is 1.9 times of that in the traditional NAS with RAID 10. The vSAN architecture can also reduce the overall cost of a business and improve the application of virtual desktops.
- Regarding the network infrastructure, the use of 10GbE servers is recommended as employing VMware. We adjusted the network setting and used Mellanox 40 G NIC in the vSAN; and the operational performance was improved by ESXi and guest host settings.
- In storage expansion mechanisms, scale-out produces superior results to scale-up through high parallelization if the environment is complex, and the VM must perform large amounts of data access. In multimedia servers, data access is required in a file server access environment, and scale-up performs better because more caches are employed.

When deploying Mellanox 40 G NIC, we were unable to completely utilize 40 G. If the network transmission capability is not one of the factors that adversely affect the system's performance, the services which the system can provide should be better. In the future, we shall continue to investigate how a vSAN can utilize 40 G networks effectively in order to provide more effective, better performing

system services.

Author's Contributions

Conceptualization, Tsung CK, Yang CT, Chen YL. Writing—original draft, review, editing, Tsung CK, Yang CT, Ranjan R. Data curation, Chen YL, Ou JH. All authors have read and approved the final manuscript.

Funding

This work has been supported in part by the Ministry of Science and Technology (MOST), Taiwan ROC (No. 108-2745-8-029-007, 109-2625-M-029-001, and 109-2221-E-029-020).

Competing Interests

The authors declare that they have no competing interests.

References

- [1] M. A. Khan, "A survey of security issues for cloud computing," *Journal of Network and Computer Applications*, vol. 71, pp. 11-29, 2016.
- [2] A. A. Laghari, H. He, I. A. Halepoto, M. S. Memon, and S. Parveen, "Analysis of quality of experience frameworks for cloud computing," *International Journal of Computer Science and Network Security*, vol. 17, no. 12, pp. 228-233, 2017.
- [3] J. He, L. Cai, P. Cheng, J. Pan, and L. Shi, "Consensus-based data-privacy preserving data aggregation," *IEEE Transactions on Automatic Control*, vol. 64, no. 12, pp. 5222-5229, 2019.
- [4] P. Gope and B. Sikdar, "An efficient data aggregation scheme for privacy-friendly dynamic pricing-based billing and demand-response management in smart grids," *IEEE Internet of Things Journal*, vol. 5, no. 4, pp. 3126-3135, 2018.
- [5] G. Sun, Y. Li, D. Liao, and V. Chang, "Service function chain orchestration across multiple domains: a full mesh aggregation approach," *IEEE Transactions on Network and Service Management*, vol. 15, no. 3, pp. 1175-1191, 2018.
- [6] J. Xiao, Y. Xie, T. Tillo, K. Huang, Y. Wei, and J. Feng, "IAN: the individual aggregation network for person search," *Pattern Recognition*, vol. 87, pp. 332-340, 2019.
- [7] S. Szilagyai, I. Bordan, L. Harangi, and B. Kiss, "Throughput performance comparison of MPT-GRE and MPTCP in the gigabit Ethernet IPv4/IPv6 environment," *Journal of Electrical and Electronics Engineering*, vol. 12, no. 1, pp. 57-60, 2019.
- [8] S. A. Azeem and S. K. Sharma, "Study of converged infrastructure & hyper converge infrastructre as future of data centre," *International Journal of Advanced Research in Computer Science*, vol. 8, no. 5, pp. 900-903, 2017.
- [9] H. Hawilo, M. Jammal, and A. Shami, "Exploring microservices as the architecture of choice for network function virtualization platforms," *IEEE Network*, vol. 33, no. 2, pp. 202-210, 2019.
- [10] A. Roozbeh, J. Soares, G. Q. Maguire, F. Wuhib, C. Padala, M. Mahloo, D. Turull, V. Yadhav, and D. Kostic, "Software-defined "hardware" infrastructures: A survey on enabling technologies and open research directions," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 3, pp. 2454-2485, 2018.
- [11] R. Jain and S. Paul, "Network virtualization and software defined networking for cloud computing: a survey," *IEEE Communications Magazine*, vol. 51, no. 11, pp. 24-31, 2013.
- [12] C. T. Yang, S. T. Chen, W. H. Cheng, Y. W. Chan, and E. Kristiani, "heterogeneous cloud storage platform with uniform data distribution by software-defined storage technologies," *IEEE Access*, vol. 7, pp. 147672-147682, 2019.
- [13] R. Chaudhary, G. S. Aujla, N. Kumar, and J. J. Rodrigues, "Optimized big data management across multi-cloud data centers: software-defined-network-based analysis," *IEEE Communications Magazine*, vol. 56,

- [14] D. Li, K. Ota, Y. Zhong, M. Dong, Y. Tang, and J. Qiu, "Towards high-efficient transaction commitment in a virtualized and sustainable RDBMS," *IEEE Transactions on Sustainable Computing*, 2019. <https://doi.org/10.1109/TSUSC.2019.2890841>
- [15] R. Ranjan, I. S. Thakur, G. S. Aujla, N. Kumar, and A. Y. Zomaya, "Energy-efficient workflow scheduling using container-based virtualization in software-defined data centers," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 12, pp. 7646-7657, 2020.
- [16] P. Casas, M. Seufert, S. Egger, and R. Schatz, "Quality of experience in remote virtual desktop services," in *Proceedings of 2013 IFIP/IEEE International Symposium on Integrated Network Management (IM)*, Ghent, Belgium, 2013, pp. 1352-1357.
- [17] T. Guo, P. Shenoy, K. K. Ramakrishnan, and V. Gopalakrishnan, "Latency-aware virtual desktops optimization in distributed clouds," *Multimedia Systems*, vol. 24, no. 1, pp. 73-94, 2018.
- [18] C. H. Chang, C. T. Yang, J. Y. Lee, C. L. Lai, and C. C. Kuo, "On construction and performance evaluation of a virtual desktop infrastructure with GPU accelerated," *IEEE Access*, vol. 8, pp. 170162-170173, 2020.
- [19] W. L. Encalada and J. L. C. Sequera, "Model to implement virtual computing labs via cloud computing services," *Symmetry*, vol. 9, no. 7, article no. 117, 2017. <https://doi.org/10.3390/sym9070117>
- [20] R. Gracia-Tinedo, J. Sampe, G. Paris, M. Sanchez-Artigas, P. Garcia-Lopez, and Y. Moatti, "Software-defined object storage in multi-tenant environments," *Future Generation Computer Systems*, vol. 99, pp. 54-72, 2019.
- [21] S. M. Basnet, H. Aburub, and W. Jewell, "Residential demand response program: predictive analytics, virtual storage model and its optimization," *Journal of Energy Storage*, vol. 23, pp. 183-194, 2019.
- [22] A. A. Ilham and S. Usman, "Performance analysis of extract, transform, load (ETL) in apache Hadoop atop NAS storage using iSCSI," in *Proceedings of 2017 4th International Conference on Computer Applications and Information Processing Technology (CAIPT)*, Kuta Bali, Indonesia, 2017, pp. 1-5.
- [23] E. Kakoulli and H. Herodotou, "OctopusFS: a distributed file system with tiered storage management," in *Proceedings of the 2017 ACM International Conference on Management of Data*, Chicago, IL, 2017, pp. 65-78.
- [24] K. Shvachko, H. Kuang, S. Radia, and R. Chansler, "The Hadoop distributed file system," in *Proceedings of 2010 IEEE 26th Symposium on Mass Storage Systems and Technologies (MSST)*, Incline Village, NV, 2010, pp. 1-10.
- [25] R. Ramakrishnan, B. Sridharan, J. R. Douceur, P. Kasturi, B. Krishnamachari-Sampath, K. Krishnamoorthy, et al., "Azure data lake store: a hyperscale distributed file service for big data analytics," in *Proceedings of the 2017 ACM International Conference on Management of Data*, Chicago, IL, 2017, pp. 51-63.
- [26] B. F. Cooper, E. Baldeschwieler, R. Fonseca, J. J. Kistler, P. P. S. Narayan, C. Neerdaels, et al., "Building a cloud for yahoo!," *IEEE Data Engineering Bulletin*, vol. 32, no. 1, pp. 36-43, 2009.
- [27] S. Bende and R. Shedge, "Dealing with small files problem in Hadoop distributed file system," *Procedia Computer Science*, vol. 79, pp. 1001-1012, 2016.
- [28] S. Xiao, T. Li, B. Guo, and Z. Huang, "Cloud platform wireless sensor network detection system based on data sharing," *Cluster Computing*, vol. 22, no. 6, pp. 14157-14168, 2019.
- [29] R. Huo, F. R. Yu, T. Huang, R. Xie, J. Liu, V. C. Leung, and Y. Liu, "Software defined networking, caching, and computing for green wireless networks," *IEEE Communications Magazine*, vol. 54, no. 11, pp. 185-193, 2016.
- [30] M. Ndiaye, G. P. Hancke, and A. M. Abu-Mahfouz, "Software defined networking for improved wireless sensor network management: a survey," *Sensors*, vol. 17, no. 5, article no. 1031, 2017. <https://doi.org/10.3390/s17051031>
- [31] R. Gracia-Tinedo, J. Sampe, E. Zamora, M. Sanchez-Artigas, P. Garcia-Lopez, Y. Moatti, and E. Rom, "Crystal: software-defined storage for multi-tenant object stores," in *Proceedings of the 15th USENIX Conference on File and Storage Technologies (FAST)*, Santa Clara, CA, 2017, pp. 243-256.
- [32] J. Prados, P. Ameigeiras, J. J. Ramos-Munoz, J. Navarro-Ortiz, P. Andres-Maldonado, and J. M. Lopez-Soler, "Performance modeling of software-defined network services based on queuing theory with experimental validation," *IEEE Transactions on Mobile Computing*, 2019. <https://doi.org/10.1109/TMC.2019.2962488>
- [33] M. Ghobaei-Arani, A. Souri, T. Baker, and A. Hussien, "ControCity: an autonomous approach for

- controlling elasticity using buffer management in cloud computing environment,” *IEEE Access*, vol. 7, pp. 106912-106924, 2019.
- [34] G. Hou, J. Ma, C. Liang, and J. Li, “Efficient audit protocol supporting virtual nodes in cloud storage,” *Transactions on Emerging Telecommunications Technologies*, article no. e3911, 2020. <https://doi.org/10.1002/ett.3911>
- [35] M. Ghobaei-Arani, R. Khorsand, and M. Ramezani, “An autonomous resource provisioning framework for massively multiplayer online games in cloud environment,” *Journal of Network and Computer Applications*, vol. 142, pp. 76-97, 2019.
- [36] M. Ghobaei-Arani, S. Jabbehdari, and M. A. Pourmina, “An autonomic approach for resource provisioning of cloud services,” *Cluster Computing*, vol. 19, no. 3, pp. 1017-1036, 2016.
- [37] M. Ghobaei-Arani, S. Jabbehdari, and M. A. Pourmina, “An autonomic resource provisioning approach for service-based cloud applications: a hybrid approach,” *Future Generation Computer Systems*, vol. 78, pp. 191-210, 2018.