Construction and Performance Analysis of Unified Storage Cloud Platform Based on OpenStack with Ceph RBD

Weichao Ding
Department of Computer Science and Engineering
East China University of Science and Technology
Shanghai, China
e-mail: weichaoding@126.com

Chunhua Gu, Fei Luo, Yaohui Chang
Department of Computer Science and Engineering
East China University of Science and Technology
Shanghai, China
e-mail: {chgu, luof}@ecust.edu.cn,
chyh_inf@shzu.edu.cn

***Abstract*—The high reliability and scalability of storage system is critical for OpenStack cloud platform system, for the reason of multiple components require the backing of back-end storage. The RADOS BLOCK DEVICE (RBD), which is the block storage device of Ceph (one of the most popular distributed file system) is used as the back-end storage of glance, nova and cinder components of OpenStack in this paper. What's more, the RBD-OpenStack, which is a unified storage cloud platform, has been build. By analyzing the operation process of the virtual machine under the RBD-OpenStack platform, much advantage has been demonstrated such as the quality of the virtual machine service etc. Experiments show that the RBD-OpenStack cloud platform can effectively reduce the time spent in deploying and migrating virtual machines and improve the read and write rate of volumes.**

Keywords-openstack; ceph; block storage; performance analysis

I. INTRODUCTION

OpenStack is an open source cloud platform project, co-sponsored by Rackspace and NASA in 2010, is committed to providing cloud computing services based on standardized hardware for any organization [1]. Because of its loose coupling and modular design concept, OpenStack has quickly gained support from many enterprises and institutions since its inception. The OpenStack foundation in 2016 survey released biennial [2] shows that there are more than 585 enterprises, nearly 40 thousand people supporting the development of this open source projects which has more than 20 million lines code in various ways.

Multiple components of OpenStack cloud platform system，which provide virtual machine service, requires the backing of the back-end storage system, the Glance component requires it to store the various images needed to create the virtual machine; The Cinder component requires it to store a large amount of data in the virtual machine disk; The Nova component requires it to store the running virtual machine system disk [3]. To prevent being kidnapped by some manufacturer, OpenStack does not specify the back-end storage used by its components, it is compatible with many existing back-end storage technology. A default deployment scenario is to use the local file system as the back-end storage for Glance and Nova components, and use LVM as the back-end storage for Cinder component. The advantages of the scheme are low cost and simple configuration, but storing the images and the data in the virtual machine in the local file system without backup is not the best choice, there are persistent, security and other aspects of the hidden dangers, moreover, with the increase of the number of virtual machines, the local storage is not easy to expand, and it is easy to cause the storage bottleneck of computing nodes. The distributed storage system [4] is a good solution, however, using different distributed storage systems as the back-end storage of the components of OpenStack, on the one hand, brings inconvenience to the management and maintenance of the system, on the other hand, a large amount of data exchange between heterogeneous storage systems will have some adverse effects on the performance and stability of the cloud platform.

To solve these problems, this paper apply Ceph [5-6] RBD to OpenStack as the back-end storage of Glance, Nova and Cinder components, accordingly, a unified storage RBD-OpenStack cloud platform is built. Experiments show that OpenStack Unified storage cloud platform based on RBD will effectively improve the quality of virtual machines service.

II. LITERATURE SURVEY

OpenStack, as a kind of open cloud operating system on the IaaS (Infrastructure as a Service) layer, each component adopts the idea of modular design, support all kinds of excellent back-end storage technology. The open architecture keeps OpenStack technically advanced, and it does not bind to specific vendors. Take the Cinder component as an example, the back-end storage that the storage node can use includes not only open source LVM, NFS, Ceph and GlusterFS, but also commercial storage systems such as EMC and IBM. Because commercial storage technology has many problems, such as expensive licensing fees, difficult expansion and so on, most cloud service providers choose open source storage technology as back-end storage for OpenStack components [7]. However, open source storage systems generally provide only one kind of storage service, which can't meet the application requirements of block storage and object storage at the same time, furthermore, the use of multiple back-end storage system to meet the demand

978-1-5386-4301-3/18/$31.00 ©2018 IEEE

of the different components will make cloud platforms complex and unmanageable. To address these issues, Xiaowen Zheng [8] attempts to build a shared storage pool using GlusterFS as the back-end storage for Glance, Nova, and Cinder components, the I/O test of virtual machine disk under different scenes shows that GlusterFS is superior to Swift and local file system in its read-write performance, however, when KVM matched GlusterFS did not do any optimization operation, resulting in lower IOPS values when deploying virtual machines directly to the GlusterFS storage pool; Another author Wenjun Huang [9] uses the Ceph block storage device RBD as the back-end storage for the Glance and Cinder components, moreover, the file storage interface Ceph FS and Nova are integrated so that the virtual machine can be deployed directly to Ceph FS, the experimental results show that the scheme can effectively reduce the deployment and migration time of the virtual machine, however, Ceph FS does not support Clone operation, when you first create a virtual machine, you need to download the image from the Glance to the local, and then upload it to the Ceph cluster from local, which takes a long time; Sheepdog [10] is one of the few storage systems can meet the requirements of block storage and object storage, it adopts the absolute symmetrical structure so that there is none central node like metadata service, however, it has just provided qemu side drivers in block storage services at once, so that lacking of the support of libvirt [11].

From the above, although open source storage technology has the advantages of low cost and self-customization function, it is of a very high technical requirements, and its stability is not very well, therefore, an open source storage system with high availability and supporting multiple storage services is particularly important for OpenStack cloud platforms. In recent years, Ceph, a kind of distributed file storage system, has received extensive attention in OpenStack community due to its excellent structural design and nice scalability [12]. However, in the unified storage cloud platform OpenStack based on RBD, the performance and reliability of virtual machines hasn't been extensively validated. Aiming at this point, this paper attempts to integrate RDB into the Glance, Cinder, and Nova components of OpenStack, by testing and analyzing the deployment and the migration of virtual machines and read-write rate of cloud hard disk, to demonstrate the advantages while using RDB as OpenStack unified back-end storage.

III. CONSTRACTION OF RBD-OPENSTACK

A. Integration of OpenStack and RBD

There are three components in OpenStack that can be integrated with RBD: Nova, Glance, and Cinder. After the integration of Nova and RBD, the image file of the virtual machine will be stored in the Ceph clusters; after the integration of Glance and RBD, the images of the cloud platform will be stored in the Ceph clusters; after the integration of Cinder and RBD, the volumes of the virtual machine will be stored in the Ceph clusters. At present, Ceph developers have developed corresponding drivers based on block storage interfaces to support these components [13], as shown in Fig. 1, RBD's support for Nova and Cinder relies on qemu, which calls the librbd command to implement the virtual machine's management of system volumes and cloud hard disk; RBD's support for Glance is relatively straightforward, the in-out operation of image can be accomplished by writing drivers for the librbd interfaces.

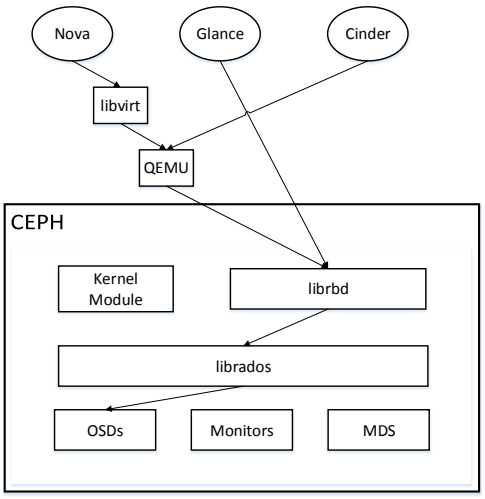

Figure 1. RBD's support for Nova, Glance, and Cinder

B. RBD-OpenStack Cloud Platform Architecture

Based on the integration of RBD and OpenStack, we can build RBD-OpenStack unified storage cloud platform. The system architecture is shown in Fig. 2, the cloud platform can be divided into four types of nodes: control nodes, network nodes, compute nodes and storage nodes. The relevant service components of OpenStack (Keystone, Horizon, Glance, Cinder, Control service of Nova), Monitor of Ceph storage cluster and MySQL database are installed and deployed in the control nodes, it is responsible for receiving external requests, controlling and managing the computing nodes and the storage nodes; The Neutron components has been deployed in network nodes, it's mainly responsible for managing and creating network resources, and distribution network service for virtual machines; Nova-compute (computing service of Nova) has been deployed in the compute nodes to manage the lifecycles of virtual machines; Ceph distributed storage cluster has been deployed in the storage nodes to provide the management and storage service of virtual machine system disks, images and volumes in the cloud platform.

Four types of nodes are connected to each other through three kinds of networks: External Network, Management Network and Storage Network. The external network is responsible for the connection of the virtual machines and the physical nodes to the Internet. The management network is responsible for managing and controlling storage and computing nodes, mainly data for some control flows. The storage network is responsible for the data reading and writing between the computing node and the storage node and storing the mutual transmission of the image or snapshot between storage nodes. Generally, the data traffic in storage

network is large, and it is recommended to use the optical fiber.

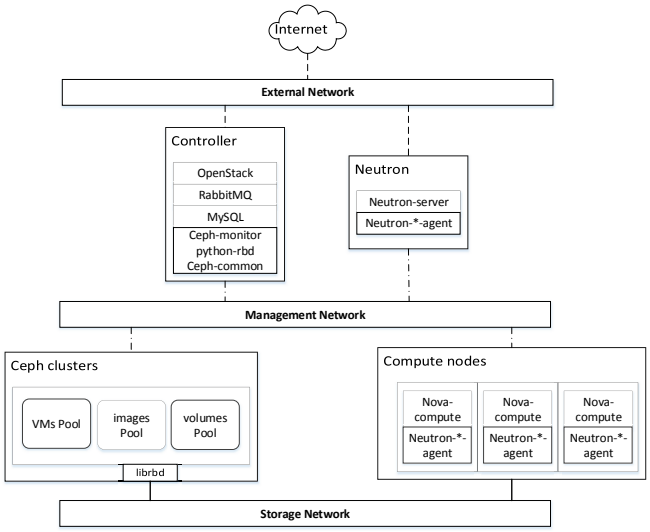

Figure 2. The diagram of RBD-OpenStack cloud platform architecture

IV. Performance Analysis of RBD-OpenStack

As the basic unit of IaaS layer service, the virtual machines' response time to the request, the disk rate of reading and writing and the high availability of the internal data directly determine the quality of service (QoS) of the cloud platform. Therefore, this paper analyzes the superiority of RBD-OpenStack cloud platform compared with traditional storage mode cloud platform (Local-OpenStack) in virtual machine startup time, migration speed and cloud hard disk read and write efficiency in terms of virtual machine related performance.

A. Analysis of Startup Time

Define abbreviations and acronyms the first time they are used in the text, even after they have been defined in the abstract. Abbreviations such as ASME, SI, MKS, CGS, sc, dc, and rms do not have to be defined. Do not use abbreviations in the title or heads unless they are unavoidable.

In OpenStack, the startup of the virtual machine is primarily responsible for the related services of the Nova component. After the Nova-compute receives the request, the instance creation operation is executed, and the specific steps are as follows:

a) Prepare computing resources for instance.
b) Create the overlay file for instance(The overlay file is the file corresponding to the virtual machine startup disk, it will become larger as the number of software installed in the virtual machine increases).
c) Get the virtual network of instance.
d) Create the instance's XML definition file and start the virtual machine.

Since the steps 1), 3), 4) of the virtual machine creation process are independent of the storage modes of the cloud platform, therefore, the difference between RBD-OpenStack and Local-OpenStack in virtual machine startup time lies in the process of creating the overlay file.

In Local-OpenStack, the overlay file of the virtual machine is stored in the local disk of the host, as shown in Fig. 3, after the Nova-compute receives the request to create a virtual machine, it first checks to see if the image is downloaded (If we have already created a virtual machine based on the same image, we do not need to download it). If not, send the HTTP request to Glance and download the image to the local. When the image is downloaded successfully, if its type is qcow2, call the qemu-img convert command to convert it to raw format, and then call the qemu-img resize command to adjust the image to the size specified by flavor. Last, take the image in raw format as backing_file, and call the qemu-img create command to create the overlay file of the virtual machine.

Based on the above analysis, the time spent in Local-OpenStack creating the overlay file for the virtual machine $T^{local}_{create\_overlay}$ can be expressed as:

$$T^{local}_{create\_overlay} = T_{download\_image} + T_{convert\_format} + T_{resize\_image} + T_{generate\_overlay} \quad (1)$$

In (1), $T_{download\_image}$ represents the time it takes to download the image from the Glance to the local (In a production environment, $T_{download\_image}$ usually takes 3-5 minutes); $T_{convert\_format}$ represents the time it takes to convert the image format from qcow2 to raw; T_{resize_image} represents the time it takes to adjust the image size; $T_{generate\_overlay}$ represents the time it takes to generate overlay files in qcow2 format based on backing_file.

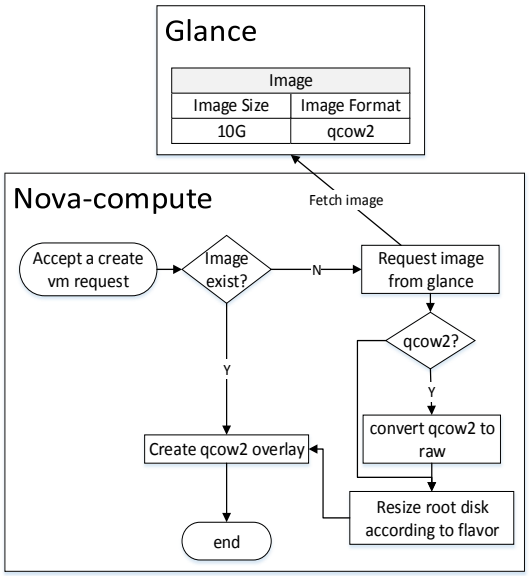

Figure 3. The creation flow chart of the virtual machine overlay file

In RBD-OpenStack, the image of the virtual machines managed by Glance are saved in RBD, therefore, when we create an overlay file, we do not need to download the image from the RDB to the local via Glance, and then upload it to RDB by Nova. In fact, when RBD is used as the back-end storage for Glance to storage the image, Glance defaults to creating a snapshot for each image (image@snapshot). After receiving the creation request of the virtual machine, the Nova-compute first checks whether the back-end storage of the Nova supports the clone operation [14], if this operation is supported, the RBD image backend command is invoked through the Nova driver, it performs clone operations to the image@snapshot on the RBD storage layer, and take the clone file as the overlay file of the corresponding virtual machine. Thus, in a unified storage mode, the time taken for the virtual machine to create the overlay file can be expressed as:

$$T_{create\_overlay}^{united} = T_{clone} \tag{2}$$

The implemented way of clone operation in Ceph is based on copy-on-write, and the relationship between the clone file and the image@snapshot is shown in Fig. 4, the cloned file (child) stores the reference to the snapshot of image (parent), and replication of data objects is triggered only when a write operation is performed on the Clone file, as a result, cloning is proceeding at a very rapid rate (i.e. $T_{clone} \ll T_{download\_image}$) [14]. In particular, when M virtual machines are created based on the same image on N different computing nodes, the time of the Local-OpenStack takes $N * T_{download\_image} +$ $M * \left(T_{convert\_format} + T_{resize\_image} + T_{generate\_overlay}\right)$, while the RBD-OpenStack only needs to perform M clone operations, the time it takes is $M * T_{clone}$, its advantages is more obvious.

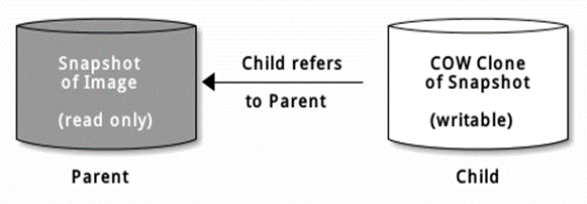

Figure 4. Sketch of relation between clone and snapshot in Ceph

B. Analysis of Migration Time

Transfer operation on OpenStack cloud platform includes Migrate and Live Migrate, where Migrate will execute the Shut Off operation of virtual machine first; While the entire migration process of Live Migrate in the virtual machine only needs to perform a short time Pause operation, and will not affect the operation of the virtual machine service. Due to the Live Migrate operation is relatively flexible and widely used, this paper will compare and analyze RBD-OpenStack and Local-OpenStack based on Live Migrate of virtual machine.

Live Migrate of virtual machine progress can be divided into three stages: Pre_live_migration, Live_migration and Post_live_migration. Among them, the Pre_live_migration stage is mainly responsible for the preparation before the migration, including:

a) Select the target nodes from several computing nodes.

b) If the Nova backend is not unified storage, it is necessary to migrate the image files and virtual network resources of the virtual machine to the target node.

The Live_migration is the execution phase of the virtual machine Live Migrate, including:

a) Perform Pause operations on the virtual machine waiting for the migration.

b) Copy the memory state data from the virtual machine to the target node.

c) Perform Resume operations on the virtual machine on the destination node.

The Post_live_migration phase is responsible for the processing after the migration, including:

a) delete related files of virtual machine on the source node

b) Create the XML definition file on the target node.

In the three stages above, scheduling strategy execution, creating and deleting virtual machine file cost a short time, which users almost do not perceive. Therefore, the whole virtual machine Live migrate time can be expressed as:

$$\begin{aligned} T_{live\_migration} &= T_{copy\_image} + T_{pause\_instance} \\ &\quad + T_{copy\_memory} + T_{resume\_instance} \end{aligned} \tag{3}$$

T_{copy_image} represents the time needed to copy images and other resources from the source node to the target node; $T_{pause\_instance}$ and $T_{resume\_instance}$ represents the time needed to pause and restore the virtual machine; T_{copy_memory} represents the time needed to copy the memory state from the source node to the target node. In traditional storage mode, T_{copy_image} is approximate to $T_{download\_image}$ which usually takes a few minutes. While in RBD-OpenStack, both the source node and the target node can access the virtual machine images directly, so there is no need to copy the images to the target node in the Live_migration phase (i.e. $T_{copy\_image} = 0$), and the migration speed will be significantly improved.

C. Other Advantages Analysis

In addition to the short start-up time and fast migration speed of the virtual machine, RBD-OpenStack also has the advantages of high scalability, high reliability and high efficiency of read and write to cinder volumes.

1) Scalability

In the traditional storage mode, the virtual machine is stored in the local disk of the computing node, and the storage capacity of the single computing node is limited and difficult to expand. It is easy to generate storage bottleneck.

While the Ceph distributed file storage system has better ability of horizontal expansion.

2) Reliability

If the computing node in the cloud platform is out of order (such as power outages or hardware damage), all the virtual machines running on it will stop service. In the unified storage mode, the image file of the virtual machine are stored in the shared storage, and all the virtual machines in the fault nodes can be restored to other nodes through the Evacuate operation provided by OpenStack.

3) Efficiency of read and write to volumes

Cinder provides a block storage service that mounts the volumes by using the iSCSI protocol when the disk space of the virtual machine is insufficient. In traditional storage mode, the computing nodes and storage nodes belong to different physical nodes, and the virtual machine's efficiency of read and write to volumes not only depends on physical disk I/O performance, also the current network traffic of cloud platform. While in RBD-OpenStack mode, the images of virtual machine and cloud disks are stored in Ceph distributed file system. It can effectively improve the reading and writing speed of cloud hard disk when the virtual machine do reading and writing operations to cloud the hard disk, which is process of data transfer in unified storage in actual.

V. EXPERIMENT

A. Experiment Setup

To evaluate the advantages of RBD-OpenStack in virtual machine related QoS (i.e. startup time and migration speed of virtual machine and read-write efficiency of volumes), 3 cloud platforms (RBD-OpenStack, Local-OpenStack, GSR-OpenStack) with different storage backend are constructed according to Table I and II.

TABLE I. DIFFERENT STORAGE BACKEND MODELS

Components / Storage mode	Nova	Glance	Cinder
Local	***LocalFS***	***LocalFS***	***LVM***
GSR	***GlusterFS***	***Swift***	***RBD***

The characteristics of the servers are given in Table II. Nodes of Controller, Neutron, Compute1and Compute2 are used to deploy OpenStack Liberty. Nodes of Storage1 and Storage2 are used for deployment back-end of different storage systems (the version of Ceph is Jewel and kernel configuration parameters of cluster refer to [15]). During the experiment, the storage network uses a 10-gigabit switch, the management network and the external network use gigabit switches, and the virtual machine information used for the experimental test is shown in Table III. In order to simulate virtual machine requests in IaaS cloud environment, a random method is used to generate the specified type and number of virtual machines in the virtual machine startup time experiment. The number of virtual machine types follows a normal distribution. Virtual machines of Windows7_x86_64 are adopted in migration speed experiment. Virtual machines of Ubuntu_14.04_Server are adopted in cloud hard disk mount experiment. Tool for disk reading and writing is FIO [16].

TABLE II. PHYSICAL HOST SPECIFICATIONS

Service	Server Name	VCPU/ Cores	Memory/GB	Disk/ T
OpenStack (Liberty)	Controller	***8***	***32***	***0.5***
	Network	***8***	***32***	***0.5***
	Computer1	***48***	***256***	***2***
	Computer2	***48***	***256***	***2***
Ceph, LVM, GlusterFS, Swift	Storage1	***4***	***64***	***4***
	Storage2	***4***	***64***	***4***

TABLE III. CONFIGURATION OF VIRTUAL MACHINE

Operating System	Image_size	VCPU/ Cores	Memory/ GB	Disk/GB
Ubuntu_14.04_Server	***760MB***	***2***	***4***	***30***
CentOS_7_x86_64	***540MB***	***2***	***4***	***30***
Windows7_x86_64	***16GB***	***2***	***4***	***50***

B. Experiment Result

Evaluation criterion for 3 different storage modes of OpenStack cloud platform are: comparison of startup time of virtual machines on the same request scale, comparison of migration speed of the same type of virtual machine, comparison of the rate of reading and writing of cloud hard disk on the same type of virtual machine. Fig. 5 shows the time required for startup of the virtual machine in a cloud platform with different storage modes. RBD is the shortest, GSR takes a little longer and Local takes the longest. This is because in RBD storage mode, Nova and Glance use a unified back-end storage, the process of virtual machine creation does not need to download the images. In GSR storage mode, different computing nodes of the virtual machine share GlusterFS storage pool, to create the same type of virtual machine in N different nodes only needs to download images once, while Local storage mode needs to download N times. But because of Nova and Glance in GSR using the back-end storage of different storage systems, virtual machines need to download the images from Swift to local by Glance and upload to GlusterFS by Nova for the first time. So GSR is slower than Local when the size of the virtual machine is small. It is worth noting that, with the increase of the size of the virtual machine, growth of the time to start the virtual machine will be slow gradually in Local storage mode. This is because it had created the virtual machine based on the same image, computing nodes will retain the image cache. The virtual machine behind is running directly without the need to download the image.

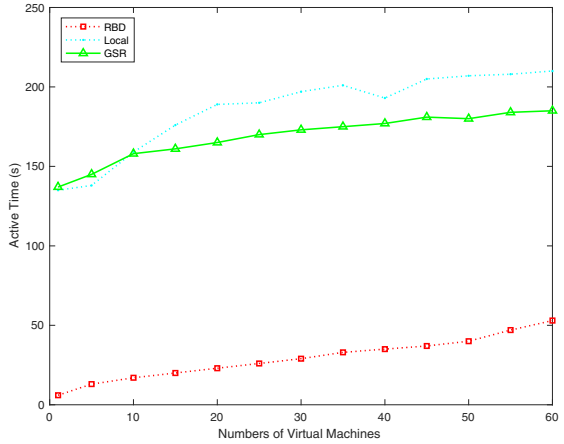

Figure 5. Comparison of instance launch time with different storage backend

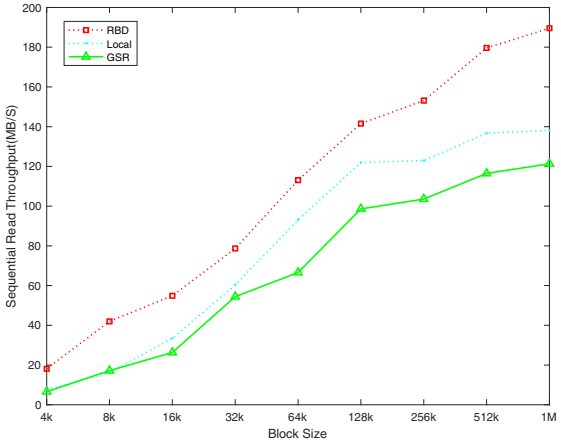

Figure 6. Comparison of sequential read throughput with different storage mode

Table IV shows the time it takes for the virtual machine to perform Live migration operations between two computing nodes in different storage modes. It can be found that in shared storage mode (RBD, GSR), the virtual machine has a very live migration process (just take 1 second). While in the traditional storage mode, since the storage of the virtual machine image is local, the process of live migration needs to copy the image files between two computing nodes, so it takes a long time.

TABLE IV. MIGRATION TIME WITH DIFFERENT BACKEND

Number / storage	1	2	3	4	5	average
Local	*117s*	*121s*	*108s*	*133s*	*128s*	*121.4s*
GSR	*1s*	*1s*	*1s*	*1s*	*1s*	*1s*
RBD	*1s*	*1s*	*1s*	*1s*	*1s*	*1s*

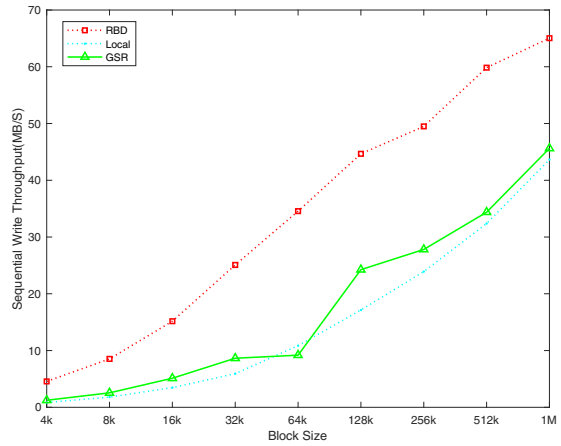

Figure 7. Comparison of sequential write throughput with different storage mode

Figs. 6 to 9 show a comparison of sequential reading and writing, random reading and writing performance of a virtual machine to a volume in different storage modes. Because of large amount of sequential data is needed in continuous reading and writing, the performance of sequential reading and writing is concerned. Therefore, data throughput is considered as a target. While random reading and writing pays attention to random reading and writing performance, IOPS is considered. It can be found that RBD mode has better performance of virtual machine reading and writing than Local and GSR. In RBD storage mode, the virtual machine to read and write the volumes is actually the process of data transfer in the Ceph storage cluster. While in Local and GSR, it not only depends on physical disk I/O performance, also the current network traffic cloud platform.

The results of this experiment are in good agreement with the performance analysis of RBD-OpenStack cloud platform in Chapter IV, which proves the effectiveness of RBD-OpenStack in solving the problem that the virtual machine starts up for a long time, the migration speed is slow and the efficiency of reading and writing on the hard disk is low.

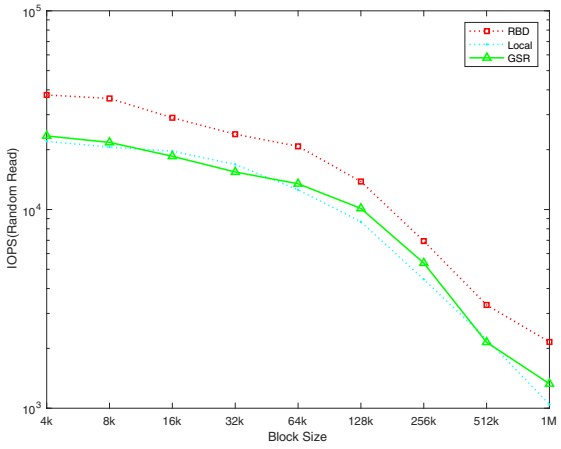

Figure 8. Random reading IOPS of volume with different storage modes

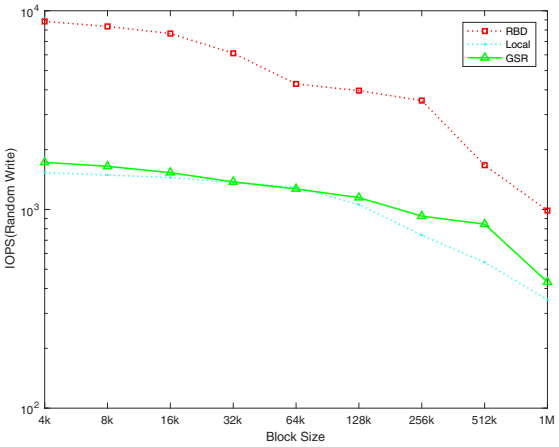

Figure 9. Random writing IOPS of volume with different storage modes

VI. CONCLUSION

In this paper, Ceph RBD is integrated with Nova, Glance and Cinder in OpenStack for the problem that the default back-end storage of the related virtual machine service components in OpenStack is not strong, the reliability is not high, and the management is complex. Then build a unified storage cloud platform: RBD-OpenStack. Through experiments and analysis with a real OpenStack cloud platform in the other two storage modes(Local and GSR), the advantages of RBD-OpenStack in reducing the deployment of virtual machines, migrating time and improving the efficiency of virtual machine reading and writing are validated.

In addition to Nova, Glance, Cinder components in OpenStack, storage services Swift and file sharing services Manila need back-end storage system support, the next step will try to Ceph object storage service RADOSGW and file storage service CephFS applied to OpenStack , The specific work is as follows:

a) RADOSGW integrates with Keystone and replaces Swift to provide object storage services.

b) Use CephFS as Manila backend storage, providing file sharing services.

ACKNOWLEDGMENT

This work is supported by the National Natural Science Foundation of China (Grant NO.61472139).

REFERENCES

[1] Xiaoke Li. The Research of Virtual Machine Placement Strategy Based on OpenStack Cloud Platform [D]. East China University of Science and Technology, 2016.

[2] OpenStack foundation. A snapshot of OpenStack users' attitudes and deployments [R]. 2016.

[3] Fitfield T. Introduction to OpenStack [J]. Linux Journal, 2013, 2013(235): 4.

[4] Chuanhui Yang. Large-scale Distributed Storage System: Principles and Architectures [M]. China Machine Press, 2013.

[5] Weil S A. Ceph: reliable, scalable, and high-performance distributed storage [J]. Santa Cruz, 2007.

[6] Weil S A, Brandt S A, Miller E L, et al. Ceph: a scalable, high-performance distributed file system[C]. Symposium on Operating Systems Design and Implementation. USENIX Association, 2010:307-320.

[7] Rackspace Cloud Computing. OpenStack Powers Demanding Production Workloads Worldwide. http://www.openstack.org/user-stories/ [EB/OL].

[8] Xiaowen Zheng. Design and Implementation of OpenStack Cloud Computing Platform Based on The GlusterFS [D]. Dalian University of Technology, 2014.

[9] Wenjun Huang. Research on Integrating OpenStack with Ceph [D]. Zhejiang University, 2014.

[10] Maciel P, Matos R, Callou G, et al. Performance evaluation of sheepdog distributed storage system[J]. 2014:3370-3375.

[11] Haomai Wang. Viewing the world of block storage from OpenStack. http://www.infoq.com/cn/articles/block-storage-overview/ [EB/OL].

[12] Wei Kong, Yu Luo. Multi-level image software assembly technology based on OpenStack and Ceph [C]. 2016 IEEE 2nd Information Technology, Networking, Electronic and Automation Control Conference. Chongqing, 2016: 307-310.

[13] CEPH: a unified, distributed storage system. http://docs.ceph.com/docs/master/ [EB/OL].

[14] Maizi Mai, Analytic CEPH: The cloning of Librbd. http://www.wzxue.com/ceph-librbd-clone/ [EB/OL].

[15] Karan Singh, Learning Ceph [M]. Birmingham:Packt Publishing Limited, 2015:157-161.

[16] E1-Harake H N, Schoenemeyer T. Detailed Performance Analysis of Solid State Disks [J].