

Deployment Scheme of Video Conferencing MCU Based on OpenStack

Haifeng Han ^a, Jianxin Song ^b

Nanjing University of Posts and Telecommunications, Nanjing 210000, China

^a15062200778@163.com, ^biamsongjx@139.com

Keywords: Video Conferencing, MCU, Cloud Computing, OpenStack.

Abstract. The multipoint control unit (MCU) is a very important component of the video conferencing system. In the current video conferencing system, the hardware performance of MCU equipment has more demand with the different requirements of video conferencing type. At the same time, with the physical location of the participants is different, flexibility of the MCU geographical location is also higher. Room size of video conferencing and the number of participants can't match well, resulting in waste of resources. With the emergence of cloud computing, its advantages (on-demand self-service, broad network access, resource pooling, rapid elasticity, measured service) provide new ideas for solving the problems faced by traditional video conferencing MCU. This paper implements a scheme to deploy software MCU on the cloud platform OpenStack based on the fuel environment, analyzes the advantages of this scheme, and provides a valuable idea into the deployment of MCU to cloud computing.

1. Introduction

1.1 Multipoint Control Unit (MCU)

In a multipoint video conference system ^[1], MCU ^[2] is equivalent to a media switch, which is composed of a multipoint controller (MC) and a multipoint processor (MP). MC, as the media control module, is responsible for coordinating and controlling video transmission between multiple terminals, such as call and control of conference; MP, as the media processing module, is responsible for features ^[2] including multi-picture, audio mixing, stream forwarding and monitoring. MCU can classify and extract the information streams transmitted by different conference terminals, and then process them through internal audio and video processing modules. After that, all kinds of information are reassembled and forwarded to the terminal of the corresponding venue. The relationship between the MCU and the endpoints of the participants is shown in Figure 1.

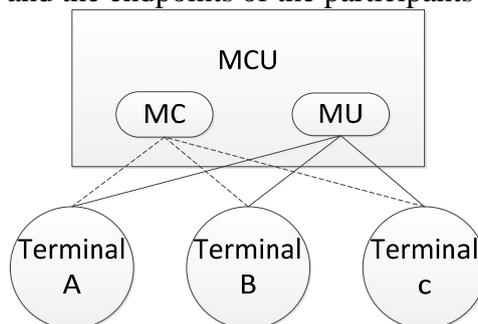


Fig. 1 The relationship between the MCU and the endpoints of the participants

Annotation: --- A control channel, used to transport control messages.

A media logical channel, used to transmit media information.

1.2 OpenStack

OpenStack is an open source project developed by NASA and Rackspace aimed at providing software for both public and private clouds. OpenStack is a IaaS ^[3] layer software that aims to provide reliable cloud deployment scenarios and good scalability to achieve cloud infrastructure services ^[4] like Amazon. OpenStack has three core modules ^[5]: computing (Nova), object-based storage (Swift) and Image Management (Glance). Each module can be installed and run independently. Nova is a cloud platform controller that provides a tool to deploy the cloud, including running instances, managing the network, and controlling access to the cloud by users and other modules. Swift is an

extensible object-based storage system. Object-based storage supports a variety of applications, such as copying and archiving data, images or video services, storing secondary static data, and creating cloud based on elastic storage for web applications. Glance is a storage, query, and retrieval system for virtual machine images that allows users to query the virtual machine mirroring metadata through HTTP requests, and retrieve actual mirroring. Keystone provides certification services for OpenStack. Horizon provides the Web front-end interface for users using the cloud platform.

2. Deployment of OpenStack Cloud Platform

The scheme is installed and configured with OpenStack cloud platform through Fuel [6] environment. Deployment of OpenStack platform is more complex, usually contains a Fuel_controller node, a Fuel_master node and multiple Fuel_compute nodes. Each node can be deployed in one or more physical machines, taking full advantage of the idle hardware resources. This paper implements an OpenStack cloud platform by deploying a Fuel_controller node, a Fuel_master node, and two Fuel_compute nodes into the VirtualBox. The OpenStack cloud platform architecture, shown in Figure 2, consists of three virtual network cards (net 1~3) and four virtual machines.

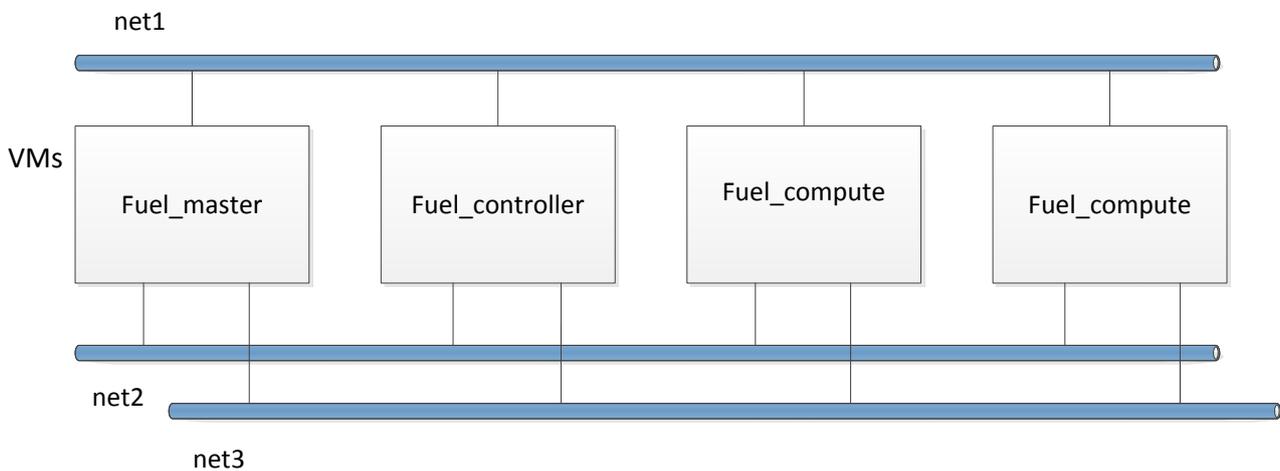


Fig.2 OpenStack architecture

The installation steps are as follows:

i. Installation Preparation

- Enabling virtualization technology support: Turn on virtualization support in the BIOS settings to support related options.

- Hardware configuration: 1 CPU quad core, 8 GB of memory and 480GB of SSD disk.

- Virtualization tools: Oracle VirtualBox 5.

- Installation package: fuel-community-8.0.iso.

ii. Implementation Process

- Establish three virtual network cards, corresponding to net 1~3, the original network card does not move, pay attention to turn off the DHCP service.

Net1: administer/management network

Net2: public network

Net3: storage/private network

- Set up the Fuel_master node, configured as follows:

VCPU: 1 Memory: 3GB Disk: 30GB Network: net1, net2, net3

- After the installation is complete, the OpenStack environment is configured by accessing the web interface.

- Establish a Fuel controller node and two Fuel_compute nodes in turn, referring to the Fuel_master configuration.

- By accessing the web interface provided by the Fuel_master node, the new three nodes are ordered to configure the environment required by the OpenStack.

- The network is set up for each node and then verified. After that, the deployment of nodes can be carried out. The OpenStack administration interface is shown in Figure 3.

The OpenStack cloud platform deployed in this article provides the Web management interface [7] that is simple, easy to use, and rich in functionality. The login user name and password are the same as those used when registering services in Keystone. The administrator's permissions include the following: create users and project groups and manage them, modify the instance of the user group and use the quota, alter available mirror types, view the usage of the application platform. The intermediate user has the right to operate on the virtual machine instance, including start, break, hang up, close, assign and reclaim floating IP addresses, assign and recover the cloud hard disk, and so on. Ordinary users can use the services provided by virtual machines.

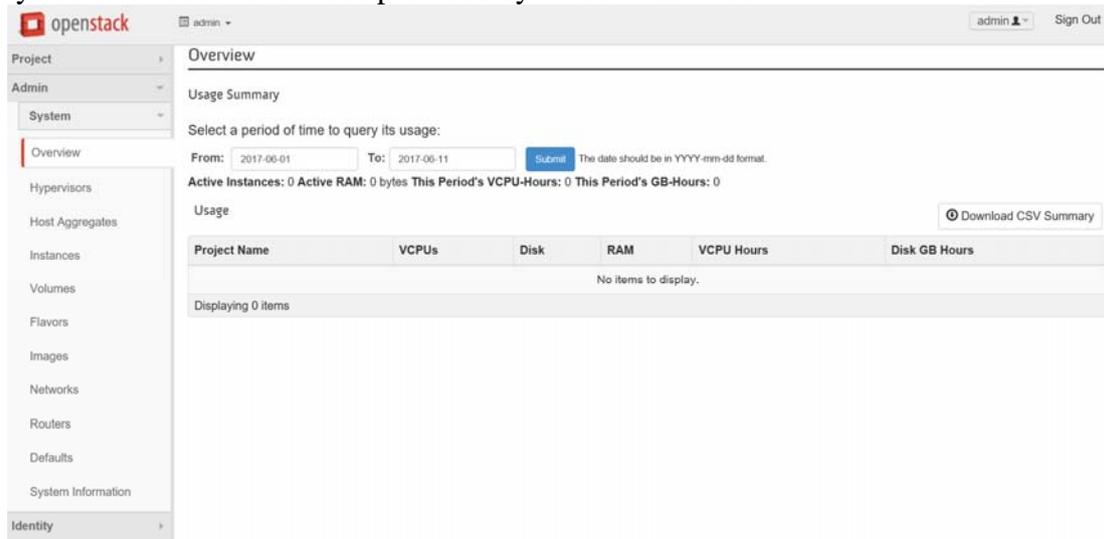


Fig.3 OpenStack administration interface

When cloud conferencing users demand, cloud conferencing service providers provide the video resolution, the audio sampling rate, FPS and other parameters^[7] for users to arrange different matching program, and calculate bandwidth, CPU, memory and other physical resource requirements according to user's choice and number of participants. As the reference platform of an instance of virtualization, the internal scheduling mechanism selects the best computing nodes to visualize instances.

3. Deployment of MCU on OpenStack

This paper implements a deployment of software MCU by deploying OpenMCU^[8] to the OpenStack platform. OpenMCU is an open source multipoint control unit (software MCU) that uses H.323, SIP, and RTSP protocols.

OpenMCU contains the following features: configured by a web interface on port 1420, requires no codec hardware to operate, supports all plugin audio codecs supported by OpenH323, supports H.261, H.263 and H.264 video, can accept multiple connections simultaneously, several different conferences can take place at the same time, using the 'rooms' feature-display statistics on calls in progress, initiate calls from the MCU to remote endpoints, and support the use of a gatekeeper (for example GnuGk)

The installation process is as follows:

open required ports in iptables or turned off the firewall:

Service iptables stop

in some cases, you may need to disable SELinux:

/usr/sbin/setenforce 0

required dependencies:

Yum install libstdc++ freetype libjpeg-turbo

- download the RPM package for OpenMCU and install it:

Rpm –ivh openmcu-ru-4.1.6-1448.el6.i386.rpm

Finally, the OpenMCU operation interface is obtained by accessing the web page, as showed in Figure 4.

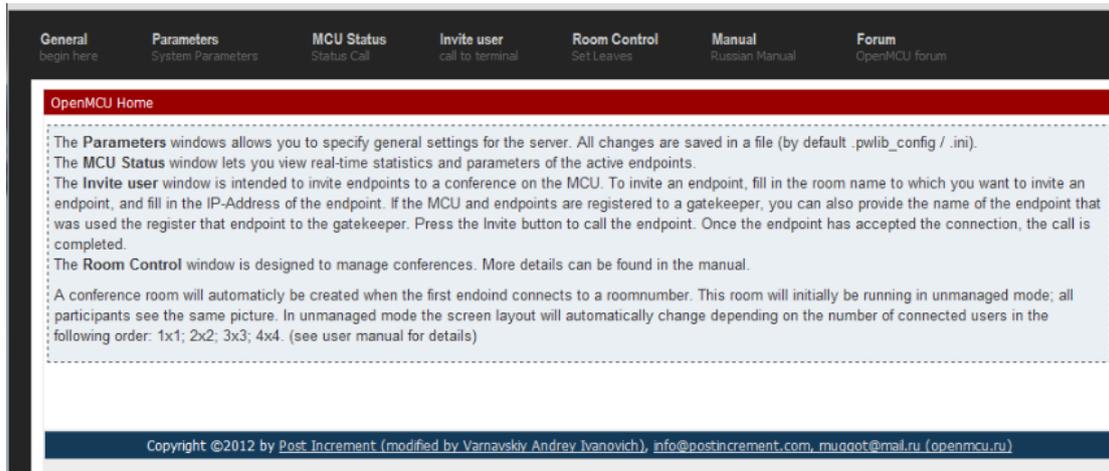


Fig.4 the OpenMCU operation interface

4. Advantage

An MCU component may require different computing characteristics depending on the number of participants, session conditions (recording, forwarding, transcoding or composing), and the physical location of the participants. Furthermore, these conditions of operability may vary dynamically during the session. As a result, deploying MCU to the OpenStack cloud platform has the following advantages:

i. The size of the meeting and the changes in the number of people during the meeting will affect MCU's demand for hardware resources. MCU operation also depends on the type of session it runs, and it could perform a variety of tasks: forwarding, recording, mixing (composing) and transcoding. Each of these features requires different computation capacities. Therefore, by evaluating the computing power required by the MCU, a virtual machine with corresponding computing power can be applied on the OpenStack cloud platform to satisfy the above problems. We can make MCU better adapt to changes in program requirements by increasing or decreasing the hardware configuration of the virtual machine (such as the number of memory, CPU and kernel) or directly turning off / on a new virtual machine.

ii. In practice, different types of session perform different tasks with different computation capacities. Different functional modules are installed into different virtual machines that are applied from OpenStack, so as to make better use of existing resources. For example, the Transcoder, mixer and broadcaster are installed on the virtual machine 1, and the recorder is installed on the virtual machine 2 to realize the different tasks of the two virtual machines to complete the MCU.

iii. The flexibility of location is another advantage of the program. By configuring the OpenStack network in different ways, we can implement a private cloud or public cloud platform. It allows us to build MCU devices in different geographical locations by applying virtual machines via web pages. Moreover, the scheme allows MCU in different regions to be interconnected and optimizes the quality of videoconferencing.

5. Conclusion

This paper implements an implementation plan for deploying MCU on a cloud platform by deploying OpenMCU to the OpenStack cloud platform. The established cloud platform has the characteristics such as simple usage, on-demand selection of instance types, and rapid deployment of virtual machine instances, thus meeting the needs of MCU in terms of geographical location flexibility and the scalability of hardware requirements [9]. The MCU deployment in the cloud is a

promising field of research, and it is necessary to make further use of the important advantages of cloud computing for practical applications.

References

- [1]. Liu Hao, Wang Dan, Li Zhe. 'Technical Characteristics and Development Trend of Videoconference Systems'. *TIANJIN Science & Technology*. Vol. 42 (2015) No. 9, p.34-39.
- [2]. Huang Yun, Zhu Zhixiang, Pei Changxing. 'Design and Implementation of MCU in Videoconferencing System Based on IP Network'. *Computer Engineering*. Vol. 30 (2004) No. 14, p.138-140.
- [3]. Wang Jingzheng. 'Overview of cloud computing technology', *Computer Knowledge and Technology*. Vol. 12 (2016) No. 20, p.52-54.
- [4]. Fang Jun. 'Multimedia and Cloud Computing'. *Computer Applications and Software*. Vol. 28 (2011) No. 10, p.291-296.
- [5]. . Rakesh Kumar, Neha Gupta, Shilpi Charu, Kanishk Jain, Sunil Kumar Jangir. 'Open Source Solution for Cloud Computing Platform Using OpenStack'. *International Journal of Computer Science and Mobile Computing*. Vol. 3 (2014) No.5, p.89-98.
- [6]. Jiang Zhongyuan. 'On the Establishment and Deployment of OpenStack Big Data Course Teaching Platform Based on Fuel Environment'. *The Science Education Article Collects*. (2017) No. 377, p.67-68.
- [7]. Chen Rongxian, Zhang Dongdong, Zhang Yaying. 'Virtualization solution of cloud conference system based on OpenStack'. *Journal of Computer Applications*. (2014) No. 34, p.82-85, 89.
- [8]. OpenMCU User Manual: <http://manualzilla.com/doc/5681651>
- [9]. Álvaro Alonso, Pedro Rodríguez, Joaquín Salvachúa, Javier Cerviño. 'Deploying a Multipoint Control Unit in the Cloud: Opportunities and Challenges'. *The Fourth International Conference on Cloud Computing, GRIDs, and Virtualization*. (2013) p.173-178.