Design and Implementation of IaaS platform based on tool migration

Wei Ding

State Key Laboratory of Networking and Switching, Beijing University of Posts and Telecommunications, Beijing, 100876, China

Email: dingweitc@bupt.edu.cn

Keywords: Cloud computing platform; service innovation; tool migration

Abstract. Based on cloud computing architecture and future network service resource architecture, using the OpenStack open source framework, design an Service Innovation Platform which can manage development tools and application programming interface in many aspect. Maximum collecting the dispersed, isolated service resources and tools to provide an open, one-stop service innovation environment for the developers. In this paper, we will explain how to building the innovation cloud platform based on tool migration and how to realize tool cloud migration, how to realize the second development, how to realize automatic expansion of cloud host and load balance. Provide a reliable cloud environment for experiment finally.

Introduction

With the development of the future network, more and more experimental personnel need to build their own environment to complete their research. However, due to the limited service resources and service tools, the types of experiment services are relatively simple; in addition, due to the lack of effective development tools to support, the efficiency of development is relatively low. Various service resources and service development tools are scattered in the hands of different providers, and have not been fully utilized. In order to solve this problem, it is necessary to integrate these scattered, isolated service resources and tools to provide an open, one-stop service innovation environment. With convenient service resources and efficient development tools, the environment can automatically or semi automatically generate related services efficiently and provide the testing and running environment, which will be important to promote the development of China's future network. Therefore, it is necessary to examine and verify the design of the future network from the perspective of application and service, and build a new type of service verification experiment platform for the future network [1], provide the design, development, deployment and testing environment for the service development or testing personnel based on the future network environment.

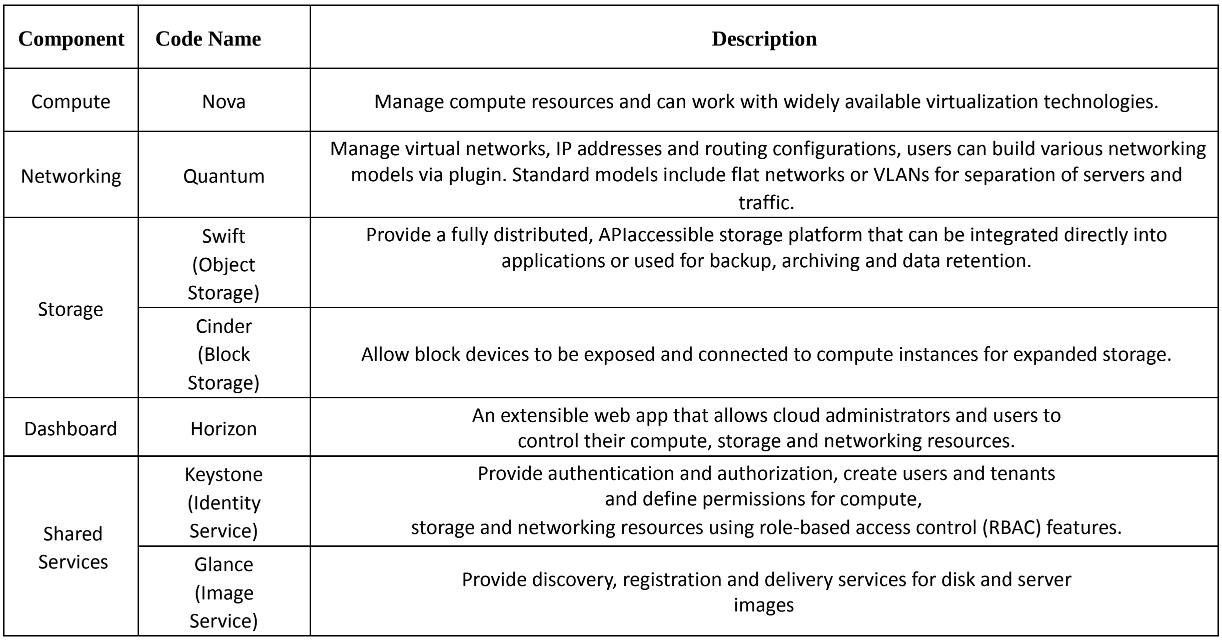
At present, varieties of business services cloud platform are provided, such as the cloud platform based on application provided by Sina company and the computing cloud platform provided by Alibaba [2]. All this platforms are provided for commercial users to provide hosting and storage services, users can do some simple deployment in these platforms, however users can’t obtain the service tools directly, let alone to publish service, to do the second development and to manage version and to share resources. In this paper, we will introduce the construction and implementation of service innovation cloud platform based on tool migration, users can obtain services, publish services, unified call service interface, and do second development, do version control, and obtain their own test results in this platform.

.

© 2015. The authors - Published by Atlantis Press

Related Works

In today's cloud computing platform, Amazon's AWS is typical and widely used, Amazon's EC2 (Cloud Computing) and Amazon S3 (simple storage service) is the core component of these services [3]. In the open source community, there are a lot of cloud computing platform. OpenStack [4] is an open source cloud computing management platform project, which consists of several major components to complete specific work, almost all types of cloud environment, project objectives are to provide the implementation of simple, scalable, rich, standard unified cloud computing management platform, using a variety of complementary services to provide infrastructure namely services (IaaS) solutions, each service provides API to provide software for the construction and management of public and private cloud. Its community has more than 130 companies and 1350 developers, these organizations and individuals will be OpenStack as a general purpose infrastructure that serves (IaaS) resources. According to the target that to build an open cloud platform and hardware environment, we use OpenStack as the basic cloud computing platform.



Component	Code Name	Description
Compute	Nova	Manage compute resources and can work with widely available virtualization technologies.
Networking	Quantum	Manage virtual networks, IP addresses and routing configurations, users can build various networking models via plugin. Standard models include flat networks or VLANs for separation of servers and traffic.
Storage	Swift (Object Storage)	Provide a fully distributed, APIaccessible storage platform that can be integrated directly into applications or used for backup, archiving and data retention.
	Cinder (Block Storage)	Allow block devices to be exposed and connected to compute instances for expanded storage.
Dashboard	Horizon	An extensible web app that allows cloud administrators and users to control their compute, storage and networking resources.
Shared Services	Keystone (Identity Service)	Provide authentication and authorization, create users and tenants and define permissions for compute, storage and networking resources using role-based access control (RBAC) features.
	Glance (Image Service)	Provide discovery, registration and delivery services for disk and server images

Fig.1. Core Components of OpenStack (IceHouse)

At the present stage, the cloud application migration service has gradually become a trend, lots of enterprise applications and test environments have begun to migrate to the cloud, and some cloud migration technology is relatively mature, however, The currently cloud migration services are all staying in the private local services cloud migration, and did not share their service; the existing programs and applications which based on open source code are all in the sharing platform such as Github [5], users lack a direct operational environment, all of those source program requires to be built and run in local place, which make users great difficult to use, firstly, This approach need to consume local physical resources, and secondly, building environment might spend a lot of manpower and resources, and easy to have compatibility problems and so on, In order to achieve an open, open source faced sharing cloud platform, in this paper, we migrated local tools and applications to the cloud using virtual machine, users can use the resources created by mirror and Snapshot directly and also can improving resource, published resources. Different from the traditional local service, cloud migration has a lot of advantages [6] [7] [8]:

a) cost saving

Traditional services are implemented by local physical machine, the performance of the physical machine has stringent requirements. By using cloud services, we can take advantage of the distributed hosts, using the cluster approach to ensure performance and cost.

b) Scalability
With the limit of performance and capacity on local machine when the service pressure is high, local service can't expand capacity and performance to handle the pressure, meanwhile, when the pressure is reduced, local service can't reduce capacity to save resources. By using the cloud services, we can do elastic expansion and contraction.

c) Flexibility
When services id down, local machine is not conducive to mobility and migration, to solve this problem, we have to use other machine as a copy, it's a big waste of resources, but the cloud host can achieve dynamic migration and resources changed which make greatly improved on flexibility.

d) Shareable
For different users, due to geographical barrier, we can't achieve real-time sharing, due to the impact site environment, local resources are difficult to reproduce. By using cloud host, we can solve this problem perfectly, achieve a shared development environment.

e) version control and monitoring
In the cloud, users can build version control systems, and even, users can run different version of program to compare the performance and call a unified platform interface to monitoring instances of their services.

Implementation of Service Innovation Cloud Platform

In this chapter, we describe how to build the service innovation platform, we will divide the platform into two parts, underlying part is the physical structure and there are different virtual virtualization technology to manage physical resources. OpenStack as the cloud system layer supports a variety of virtualization technology, such as KVM, Xen, HpverV, VMware, and etc. In addition, OpenStack provides several services to manage different resources. Nova is the services which manages computing resources. Quantum manages network resources. Swift and cinder are the object storage and block storage services [9].

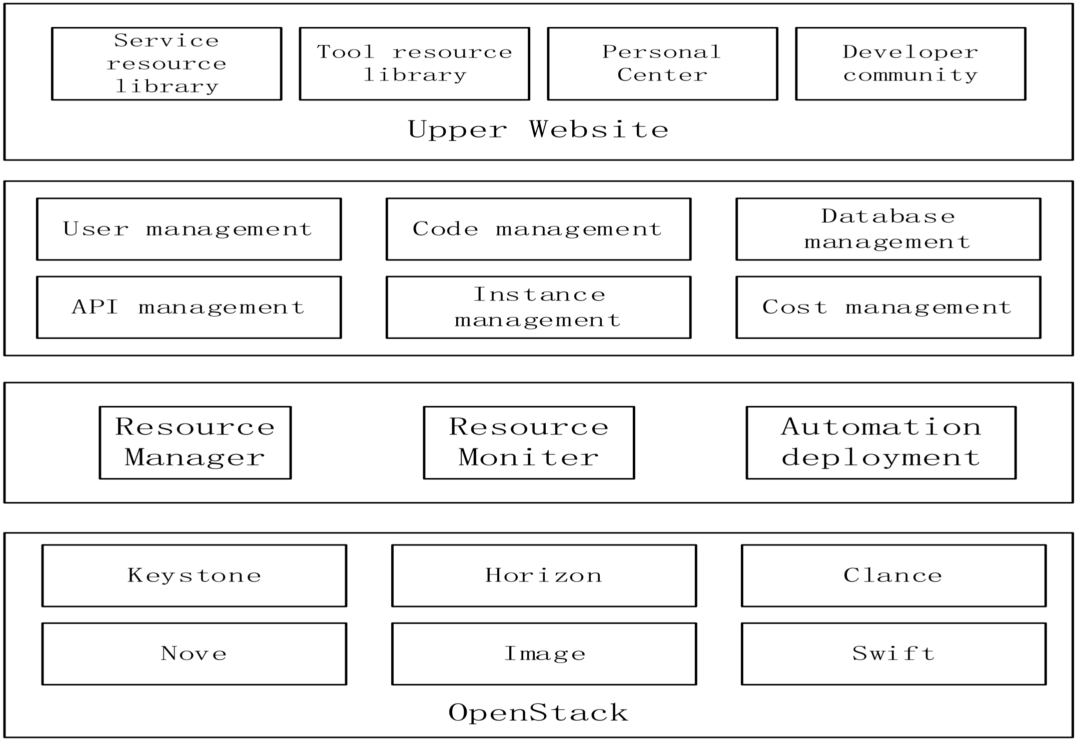
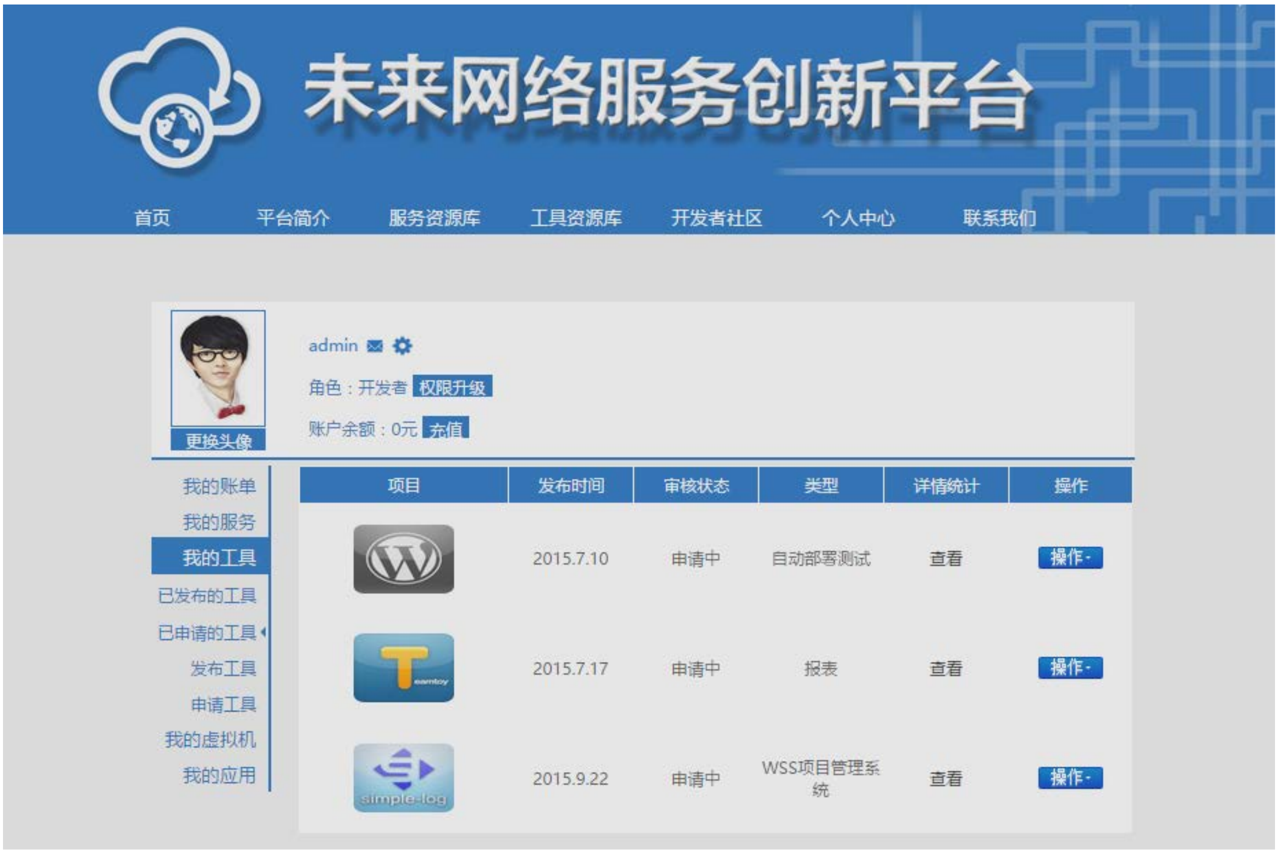


Fig.2. Architecture of Service Innovation Platform

The upper part is our website system, it contains a complete web systems, an automated deployment systems and a network monitoring system. The web system is consist of services resource library, tool resource library, individual centers and developers Forum. Services resource library contains various popular API, such as social services, electronic payment services, audio transcoding services, voice text conversion service etc. At the same time, when the user want to share your API service, you can register to publish it for others. Tools Library is the most important part of the innovative service platform, we have a collection of tools for platform migration, platform registered users, only need to apply the tools, can create a virtual machine with the tool, then they do the remote connection to log on to do the secondary development on their virtual machine, in addition, when the user have done a secondary development or have a new tools , they can publish it and the platform will save the current virtual machine by snapshot in order to be used by others. In personal center, users can manage their own tools, applications and virtual machines. The developer community, which is developed as a forums, give a place for users to exchange experience and discuss.



admin

角色：开发者 权限升级

账户余额：0元 充值

项目	发布时间	审核状态	类型	详情统计	操作
	2015.7.10	申请中	自动部署测试	查看	操作
	2015.7.17	申请中	报表	查看	操作
	2015.9.22	申请中	WSS项目管理系统	查看	操作

Fig.3. Website Implementation

Apart from the website module, there also have several modules, such as the Automated Deployment module. When the user apply a virtual machine on the platform, using the open-source tool called Puppet, the platform provide users the automated deployment of the basic tools, such as JDK, Tomcat, MySQL and other infrastructure development services, while users can dynamic switching to version, all this can be found in the personal center of the website module, it's a great convenience to users.

Another module is Resource monitoring module. The platform monitors the system for various parameters. By using the OpenStack Neutron API, monitor module can get the outside network flow. For single virtual machine, platform use the Libvirt [10] interface to get the network, the virtual machine memory usage, the virtual machine disk read and the write speed, the virtual machine CPU usage rate and other parameters and then store it into MySQL database. This information will be used to display in the web pages and provide a guarantee for virtual machine expansion and load balance.

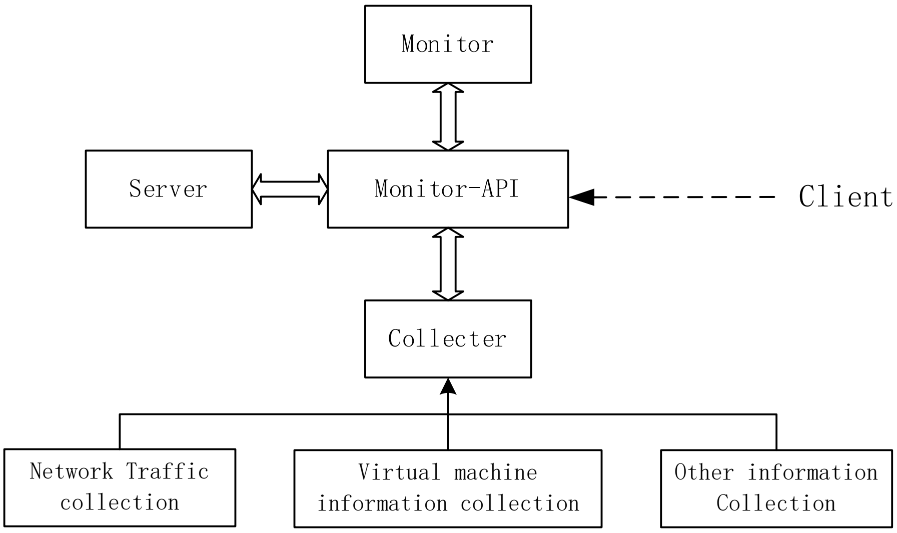

Monitor
Server
Monitor-API
Client
Collecter
Network Traffic collection
Virtual machine information collection
Other information Collection

Fig.4. Architecture of Monitor Module

Platform provide an load balance deployment interface, when use want to create an load balancing machine cluster, platform will do this using the Apache reverse proxy, then the monitor module will monitor the virtual machine in the cluster, we have an algorithm to calculate whether the network flume, memory used, CPU rate are over limited, if the current cluster can’t server the request, platform will make a new instance by the snapshot and add it to the load balance machine cluster.

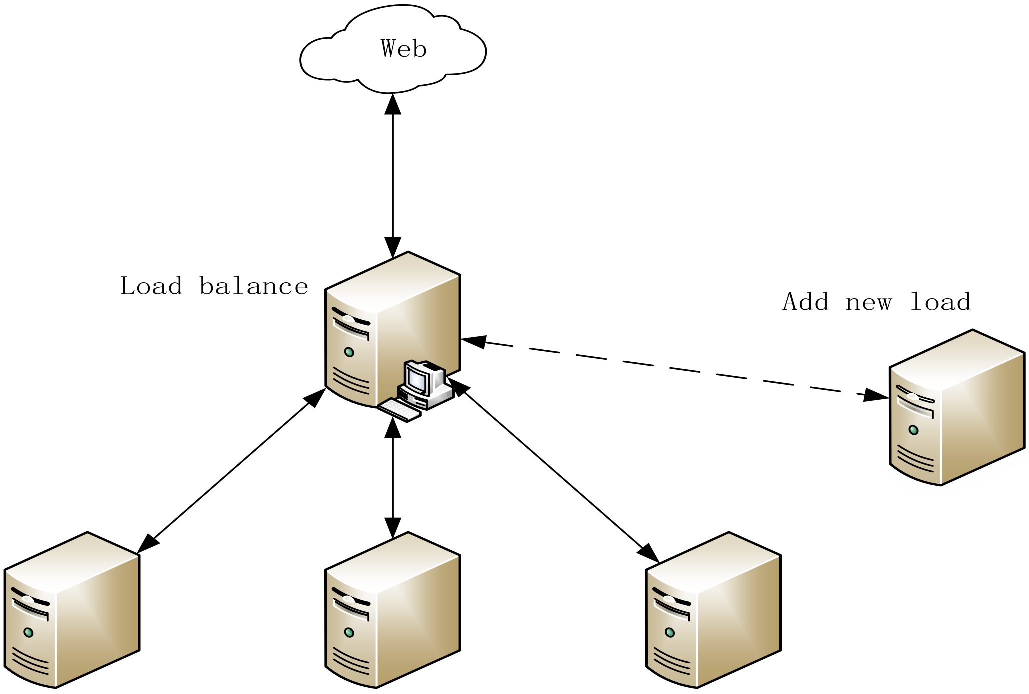


Fig.5. Architecture of Load Balance

Conclusion

The Service Innovation Platform achieved the migration of basic tools，automated deployment of basic development tools, the basic information collection and monitoring of virtual machine, the automatic expansion and load-balancing of virtual machine cluster. It provide Developers and testing personnel a good platform to exchange and sharing their development. After verification, the platform is stable and efficient performance, reached the original idea perfectly. However, the security issues of migration tool is not guaranteed right now, for those who want to share their services but do not want to share their open source, we do not have an interface to access. So this problems will be thought and solved in next step.

Acknowledgement

In this paper, the research was sponsored by the Service Identification and Migration mechanism of Future Internet (No. 2012CB315802), National Key Basic Research Program of China (973 Program)

References

[1] Pan Y, Kumar P, Kim J, et al. Firefly: illuminating future network-on-chip with nanophotonics[C]//ACM SIGARCH Computer Architecture News. ACM, 2009, 37(3): 429-440.

[2] Zhang W, Tang H, Jiang H, et al. Multi-level selective deduplication for vm snapshots in cloud storage[C]//Cloud Computing (CLOUD), 2012 IEEE 5th International Conference on. IEEE, 2012: 550-557.

[3] Cloud A E C. Amazon web services[J]. Retrieved November, 2011, 9: 2011.

[4] http://www.OpenStack.org/

[5] http://www.github.com /

[6] Qiu W, Zheng Z, Wang X, et al. Reliability-based design optimization for cloud migration[J]. Services Computing, IEEE Transactions on, 2014, 7(2): 223-236.

[7] van den Hengel A, Pietikainen M, Wang Y Y, et al. International Journal of Automation and Computing[J].

[8] Morshedlou H, Meybodi M R. Decreasing impact of SLA violations: A proactive resource allocation approachfor cloud computing environments[J]. Cloud Computing, IEEE Transactions on, 2014, 2(2): 156-167.

[9] Sefraoui O, Aissaoui M, Eleuldj M. OpenStack: toward an open-source solution for cloud computing[J]. International Journal of Computer Applications, 2012, 55(3): 38-42.

[10] http://libvirt.org/